Takumi Toyama

Towards Wearable Attention-Aware Systems in Everyday Environments

Dissertation

genehmigt vom Fachbereich Informatik der Universität Kaiserslautern

zur Verleihung des akademischen Grades

Doktor der Ingenieurwissenschaften (Dr.-Ing.)

Dekan:
Prof. Dr. Klaus Schneider, Universität Kaiserslautern

Berichterstatter:
Prof. Dr. Prof. h.c. Andreas Dengel, Universität Kaiserslautern
apl.-Prof. Dr. habil. Marcus Liwicki, Universität Kaiserslautern

Vorsitzender der Promotionskommission:
Juniorprof. Dr. Christoph Garth, Universität Kaiserslautern

Kaiserslautern, den 03. November 2015

D 386

Abstract

Attention-awareness is a key topic for the upcoming generation of computer-human interaction. A human moves his or her eyes to visually attends to a particular region in a scene. Consequently, he or she can process visual information rapidly and efficiently without being overwhelmed by vast amount of information from the environment. Such a physiological function called visual attention provides a computer system with valuable information of the user to infer his or her activity and the surrounding environment. For example, a computer can infer whether the user is reading text or not by analyzing his or her eye movements. Furthermore, it can infer with which object he or she is interacting by recognizing the object the user is looking at. Recent developments of mobile eye tracking technologies enable us to capture human visual attention in ubiquitous everyday environments. There are various types of applications where attention-aware systems may be effectively incorporated. Typical examples are augmented reality (AR) applications such as *Wikitude* which overlay virtual information onto physical objects. This type of AR application presents augmentative information of recognized objects to the user. However, if it presents information of all recognized objects at once, the overflow of information could be obtrusive to the user. As a solution for such a problem, attention-awareness can be integrated into a system. If a system knows to which object the user is attending, it can present only the information of relevant objects to the user.

Towards attention-aware systems in everyday environments, this thesis presents approaches for analysis of user attention to visual content. Using a state-of-the-art wearable eye tracking device, one can measure the user's eye movements in a mobile scenario. By capturing the user's eye gaze position in a scene and analyzing the image where the eyes focus, a computer can recognize the visual content the user is currently attending to. I propose several image analysis methods to recognize the user-attended visual content in a scene image. For example, I present an application called *Museum Guide 2.0*. In Museum Guide 2.0, image-based object recognition and eye gaze analysis are combined together to recognize user-attended objects in a museum scenario. Similarly, optical character recognition (OCR), face recognition, and document image retrieval are also combined with eye gaze analysis to identify the user-attended visual content in respective scenarios. In addition to Museum Guide 2.0, I present other applications in which these combined frameworks are effectively used. The proposed applications show that the user can benefit from active information presentation which augments the attended content in a virtual environment with a see-through head-mounted display (HMD).

In addition to the individual attention-aware applications mentioned above, this thesis presents a comprehensive framework that combines all recognition modules to recognize the user-attended visual content when various types of visual information resources such as text, objects, and human faces are present in one scene. In particular, two processing strategies are proposed. The first one selects an appropriate image analysis module according to the user's

current cognitive state. The second one runs all image analysis modules simultaneously and merges the analytic results later. I compare these two processing strategies in terms of user-attended visual content recognition when multiple visual information resources are present in the same scene.

Furthermore, I present novel interaction methodologies for a see-through HMD using eye gaze input. A see-through HMD is a suitable device for a wearable attention-aware system for everyday environments because the user can also view his or her physical environment through the display. I propose methods for the user's attention engagement estimation with the display, eye gaze-driven proactive user assistance functions, and a method for interacting with a multi-focal see-through display.

Contributions of this thesis include:

- An overview of the state-of-the-art in attention-aware computer-human interaction and attention-integrated image analysis.
- Methods for the analysis of user-attended visual content in various scenarios.
- Demonstration of the feasibilities and the benefits of the proposed user-attended visual content analysis methods with practical user-supportive applications.
- Methods for interaction with a see-through HMD using eye gaze.
- A comprehensive framework for recognition of user-attended visual content in a complex scene where multiple visual information resources are present.

This thesis opens a novel field of wearable computer systems where computers can understand the user attention in everyday environments and provide with what the user wants. I will show the potential of such wearable attention-aware systems for everyday environments for the next generation of pervasive computer-human interaction.

Acknowledgements

First of all, I would like to express my sincere appreciation to all those who supported me throughout the Ph.D program. I would like to express special gratitude to my advisor Prof. Dr. Prof. h.c. Andreas Dengel for your guidance and support throughout entire process. I sincerely thank you for giving me an opportunity to work in German Research Center for Artificial Intelligence and to write my thesis on such an interesting topic. Additionally, I would also like to thank apl.-Prof. Dr. habil. Marcus Eichenberger-Liwicki for your guidance and brilliant comments for improving this dissertation. I would like to express my deep appreciation to you because you helped me on various occasions not only research work but also daily life when I started living in Germany.

Furthermore, I would also like to acknowledge with much appreciation the support of Prof. Dr. Koichi Kise, who encouraged me to go to Germany and recommended me as a Ph.D student for the Technical University in Kaiserslautern. Without the collaboration with your Intelligent Media Processing Group in Osaka Prefecture University, my dissertation would not have been possible.

I would like to extend thanks to my colleagues in DFKI. Especially, I would like to thank Dr. Daniel Sonntag, who lead me in several projects. I got many inspirations from you, which were the fundamentals of many systems presented in this dissertation. Additionally, I thank Dr. Thomas Kieninger and Dr. Faisal Shafait, who supervised me during my master thesis and the beginning of my Ph.D process. I would like to give a gratitude to all the members in Knowledge Management department and Multimedia and Data Mining group. Especially, many thanks go to Dr. Ralf Biedert, who is currently working in Tobii, for your prior developments in the field of eye tracking and fruitful discussions. Furthermore, I also thank Markus Weber for your encouragements during writing this dissertation. I would like to express my special appreciation to Brigitte Selzer and her former student assistants Christian Walter and Ramon Plank, who supported me a lot to make it possible to live in Germany. You helped me when I had troubles because of language and different customs and cultures.

A special thanks goes to all co-authors of publications and partners of collaborative projects. In particular, I would like to thank Dr. Masakazu Iwamura, Dr. Kai Kunze, Dr. Kiyoshi Kiyokawa and Prof. Andras Lorincz for many valuable discussions.

In addition, I would like to sincerely appreciate all the work done by students. Takuya Kobayashi, Wakana Suzuki, Takahiro Matsuda, Takahiro Kashiwagi, Yuki Shiga, Hisham Mohamed Ramzy Ismail, and Loai Ghoraba, each of you developed a wonderful system or recognition algorithm. Without your contributions, I would not have completed the systems presented in this dissertation. Additionally, I would like to give a special gratitude to Jason Orlosky, who worked with me in many collaborative projects and inspired me a lot for developing several novel applications. I also thank you for proofreading this dissertation.

Furthermore, a thank you to Museum Pfalzgalarie in Kaiserslautern for letting me use pictures of their exhibitions. Also, I thank Dr. Arnd Rose and SensoMotric Instruments for their flexible responses to our requirements to eye tracking technologies.

Last but not least, many thanks to my family. My parents and my sister, you always help me and love me. Although words cannot be enough to express my gratitude to you, I would like to say a big thank you to all of you.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Attention is a primitive physiological function that primates innately possess [Tre01]. It has been developed since the birth of our neural systems over time in order to perceive scenes and environments more quickly and efficiently for hunting prey or escaping from predators. By giving our attention to a particular stimulus in an environment, we can boost the ability of sense and the acuity is the greatest at the point-of-attention. Consequently, our brain can specifically process necessary things rapidly without being overwhelmed by vast amount of information from the environment.

Many studies on human visual attention have been contributed to reveal the mechanisms of eye gaze control during image and scene perception [Hen+03; Ray+09; Dor+10; Sch+11]. Although a comprehensive mechanism of human eye gaze control is still undiscovered, many researchers found potential of human visual attention in terms of Computer-Human Interaction (CHI) [Duc02]. Indeed, there is a growing interest in visual attention in computer technologies because of the following two advantages. First, as human beings exclude irrelevant stimuli from environments in order to process necessary information, a computer system can process a limited area of an image in order to boost the performance, accuracy, or efficiency on only necessary information. Typical examples are active robot vision systems that incorporate selective attention [HE08; But+08]. By attending to a particular stimulus in an image, robots can recognize objects and environments more rapidly and efficiently, which results in improvements of the overall performance. Secondly, measuring the eye gaze of the user to detect his or her attention in a scene is essential for context-awareness in computer-human interaction [Bul+11b]. In order to understand what the user is doing, what the user is planning to do, or in which environment she or he is acting, i.e., the user context, recognition of user visual attention plays an important role. Users can benefit from such computer systems, which know what the user needs, when it is needed, and how it is needed, because they can provide the user with automatic assistance with an adequate way.

Eye gaze is merely an informative source for analyzing visual attention of a viewer. Our eyes are controlled by the visual attention system; they are normally directed to a point-of-attention. To measure human eye movements, eye tracking technologies have been developed over decades [Hen+03]. Hence, eye trackers available nowadays have become light-weight and usable in mobile scenarios, thus, pervasive. Consequently, attention analysis and applications that integrate such analysis also have become feasible in everyday ubiquitous environments.

As computer technologies become more pervasive in our daily life, we have several challenges to make them more attentive and intelligent. A typical challenge appears when the complexity of the environments is high where various types of information content are present. Attention-awareness is one of the solutions for such a challenge. In the following section, I describe the challenges.

1.1 Attention-Awareness in Everyday Environments: Motivations

Figure 1.1 shows a typical daily scene from the viewer's perspective where multiple information resources are present. As one can see in this image, various types of visual content that may be meaningful for the viewer exist. Conventional user assistance systems may not be able to provide with adequate support in situations like this since the viewer may be involved in various types of situational contexts. The viewer may *talk to a friend*, *look for a shopping item*, or *check the price of item*. For respective action, he or she needs to attend to individual information resources; e.g., he or she attends to the face of the friend to talk to her, attends to shopping items (objects) to find the item he or she wants, or attends to price tags (text) to compare the prices. Let us consider a conventional augmented reality (AR) application that can provide supportive information of present content in a scene (without attention-awareness). For example, it can present a list of prices of similar items as virtual image overlays if a price tag is present in an image. Because this application does

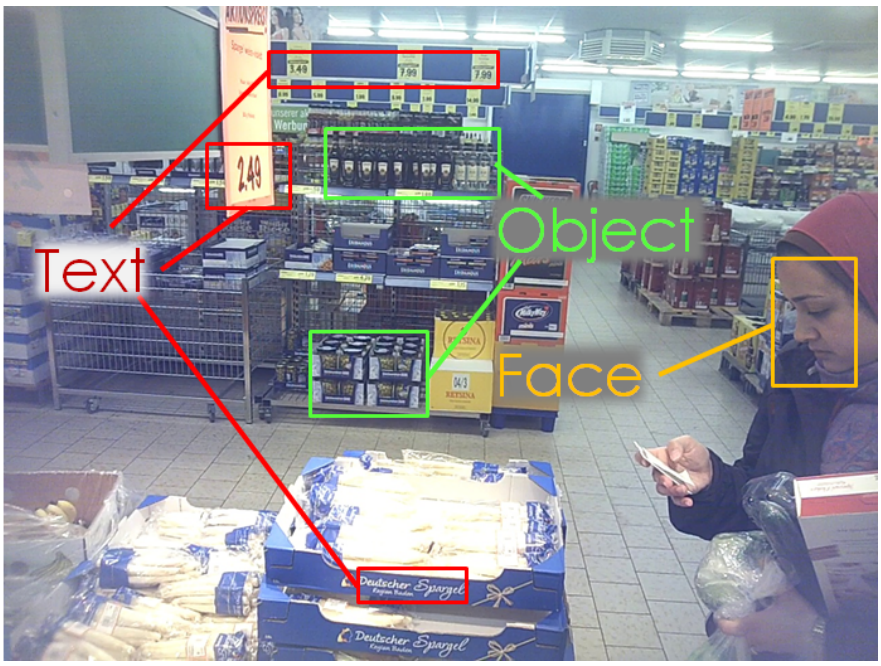


Figure 1.1: Daily scenes are often quite complex. Various types of information resources are present in the same scene (e.g., human faces, items in a supermarket, text on price tags, etc.). In these types of complex daily scenes, to ascertain which content the user is attending to is a challenging issue.

not consider in which content the user is interested, it would present the list of prices to the user as output even if the user is talking to the friend. However, such information is obtrusive because the user is not currently interested in the list of prices. If the computer system can recognize to which visual content the user is attending, it can assist him or her in a more adequate way; for example, it can present a list of prices when he or she is viewing the price tag.

In general, the more complicated an image is, the more computation is required to understand the entire image. However, it is often the case that necessary information in a complex scene is only partial. By ignoring irrelevant visual information resources in a scene, a computer system can process the scene rapidly to find relevant resources.

Attention-aware systems refer to a specific type of computer systems that can understand the user attention in the environment and process the information accordingly [RT06; BK06; Anc+12]. In this thesis, I focus on two fundamental research questions regarding attention-aware systems in everyday environments:

- *How can a computer recognize the visual content in a scene that the user is attending to?*

- *What is an adequate way to present information to the user?*

By answering these questions, I will show that users can have improved access to information, make more informed decisions about real life activities, and that virtual content becomes less obtrusive. These benefits represent a significant step towards improving wearable interfaces in ubiquitous environments. Particularly, I have two hypotheses regarding the aforementioned questions:

- By combining image analysis technologies with eye gaze analysis technologies, a computer can recognize the visual content the user is attending to.

- Analysis of user eye gaze is useful to present information of attended content in an adequate way.

To validate them, I propose several methods and frameworks for recognition of *user-attended visual content*, which represents the visual content in a scene that is attended by the user and for information presentation with proper timing and a form. As an information presentation tool of wearable attention-aware system, I propose to use a see-through head-mounted display (HMD). Analysis of eye gaze in the see-through HMD can enhance effective interaction and information presentation. I demonstrate the feasibilities and benefits of the proposed systems and frameworks in various practical *everyday applications*.

1.2 Attention-Awareness in Everyday Environments: Applications

This thesis presents applications for everyday environments. These applications effectively utilize attention-awareness to support users in various daily scenes especially where extensive computer assistance is helpful.

Museum and City When a visitor browses around a museum or a city, he or she would often need help of a *guide*. Normally, a human guide accompanies the visitor and explains the attractions (such as art objects, historical buildings, etc.) when the visitor wants information. I propose applications for a museum or city which mimic such *attentive human guides*. They monitor the visitor's eye movements to detect attention on objects. When attention to a particular object is detected, they present information about the object to the visitor.

Professional Environment In many professional environments, adequate machine support is appreciated. For example, when a doctor examines a patient, he or she sometimes needs to look up a previous record for the patient that was examined by another doctor. I propose a system that automatically learns and recognizes the patient face and presents the record of the patient in a wearable display. Another professional example is in a factory. When a worker in a factory has to carry out an infrequent task, such as system maintenance, he or she needs to ask colleagues about the process or read a manual. I also propose an application that identifies the factory system that the user is handling using the proposed attended visual content recognizer and can provide proper guidance for the task.

Reading Assist When one reads a document, he or she often uses a dictionary or encyclopedia to look up terms that are unknown to him or her. I propose an application which can assist the user in this type of situation. It monitors the reader's eye movements during reading and detects his or her attention to words. According to the attention to a word, annotations or translations are presented in the see-through HMD. Attention analysis enables the system to present supportive information unobtrusively without disturbing the user's reading process.

Language Support A traveler in a foreign country sees many signs or navigations written in the language spoken there. To understand what is written there, he or she may need to ask someone who can translate or look up the meanings of the words using a dictionary. However, letters and characters are not the same everywhere. The traveler may not even be able to read what is written there. I propose an application for helping the user in this type of problem. Using the proposed application, the user can get translations of signs more easily. Once he or she gazes at the sign, the translation is immediately presented in the display he or she wears.

Memory Aid Humans often forget events or certain types of information, sometimes even when the information is important. Also, there are some patients who suffer from brain disease that results in memory loss, such as dementia. I propose an application which can be used to aid user's memory. By analyzing the visual attention of the user in a daily scene, the application can log his or her daily memory as a collection of attended images or episodic events. Such event logs are used later by the user to recall the memory.

1.3 Contributions of this Thesis

Contributions of this thesis are summarized as follows:

- An overview of the state-of-the-art in attention-aware computer-human interaction and attention-integrated image analysis (Chapter 2):

  - Eye tracking for visual attention and human perception analysis (Section 2.1).
  - Eye gaze-based user cognitive state and activity recognition (Section 2.2).
  - Eye gaze-based computer interfaces and applications (Section 2.3).
  - User context-awareness in CHI (Section 2.4).
  - Image analysis with integrated attention direction (Section 2.5).
  - Gaze-based interfaces in virtual, augmented, and mixed reality (Section 2.6).

- Methods for the analysis of user-attended visual content in various scenarios (Chapter 4 and 5):

  - User-attended visual content analysis for objects (Section 4.1).
  - Detection of user attention to arbitrary objects (Section 4.2).
  - User-attended visual content analysis for human faces (Section 4.3).
  - User-attended visual content analysis for natural scene text (Section 5.1).
  - Gaze gestures for triggering natural scene text translation (Section 5.1).
  - User-attended visual content analysis for paper documents (Section 5.2).

- Demonstration of the feasibilities and benefits of the proposed user-attended visual content analysis methods with practical user-supportive applications:

  - Museum Guide 2.0 and Talking Places: Attention-driven machine museum (city) guide (Section 4.1 and Section 8.1).
  - Visual Diary: Logging system for user-attended visual content (Section 4.2).
  - ERMed: Augmented reality in medicine using multi-modal interfaces (Section 4.3).
  - Gaze-triggered natural scene text translator (Section 5.1).
  - Attention-driven augmented document (Section 5.2).

- Methods for interaction with a see-through HMD using eye gaze (Chapter 6):

  - A method for attention engagement estimation with a see-through HMD using gaze depth and vestibulo-ocular reflex (Section 6.1).
  - Proactive user assistance functions for a see-through HMD using eye gaze-based cognitive analysis (Section 6.1).
  - A new gaze-based interface for a multi-focal (semi-volumetric) wearable see-through display (Section 6.2).

- A comprehensive framework for recognition of user-attended visual content in a complex scene where multiple visual information resources such as text, objects, and human faces are present (Chapter 7):

  - A method for cognitive state classification using eye movements.
  - A method for selecting a proper image analysis module based on a user's cognitive state class (attention-driven image analysis).
  - A method for fusing multiple visual content recognition results.

1.4 Outline of this Thesis

Chapter 2 presents an overview of closely related work. It includes surveys of research on visual attention and human visual perception analysis, eye-based activity and cognitive state recognition, eye gaze-based user interfaces and applications, image analysis with integrated attention direction, and eye gaze-based systems in virtual, augmented, and mixed reality. Next, I present the technological backgrounds and the proposed architecture overview in Chapter 3. Then, Chapter 4 and 5 present user-attended visual content analysis methods. In addition to the analysis methods, I discuss the experiments and studies that I conducted to evaluate the proposed methods and frameworks. As a complement to image analysis, I discuss gaze-based interaction with a see-through HMD in Chapter 6. In this chapter, several approaches for user-display interaction using eye gaze are presented. In Chapter 7, I present a method for cognitive state recognition using eye gaze and a comprehensive framework for user-attended content recognition. Furthermore, Chapter 8 summarizes the applications proposed in this thesis (Museum Guide 2.0, ERMed, and others) and presents the results of the user study for Museum Guide 2.0. Finally, in Chapter 9, I conclude the thesis.

Chapter 2

Related Work

This chapter presents an overview of closely related work to the subject of this thesis. It starts with introduction of traditional approaches for eye tracking. Then, it is followed by the relations of eye tracking with the research on human visual attention and perception analysis (Section 2.1). I then discuss the research on how eye gaze is connected to mental or cognitive processes of humans (Section 2.2). Prior work showed that activities and cognitive states of the subject could be predicted by analyzing eye movements. In this section, approaches for such prediction are introduced, referring to the connection between eye gaze and human behaviors. Next, approaches and applications of gaze-based interfaces in the field of CHI are presented (Section 2.3). I provide with an overview of how eye gaze can be utilized as an input modality for a computer interface. Furthermore, I also discuss the computer systems which take context-awareness into account in terms of CHI (Section 2.4). Here it is addressed how attention-awareness contributes to context-awareness, as well as other types of context-aware computing systems. In addition to the interaction domain, I also discuss how attention direction could be utilized in image understanding or analysis frameworks (Section 2.5). In this section, I address which kind of roles visual attention play in the field of computer vision. Lastly, gaze-based interfaces in different types of simulated reality, i.e., virtual, augmented, and mixed reality are presented (Section 2.6).

2.1 Eye Tracking for Visual Attention and Perception Analysis

Eye tracking technologies have developed over a century for analysis of human visual perception [Dod08]. In the beginning of the eye tracking study, researchers mainly focused on text reading, which only requires horizontal eye movement tracking. Buswell, who was one of the first investigators of the analysis of human visual perception, developed an early prototype of the eye trackers that we can see today [Bus35]. Although the eye tracking apparatus used in the study was somewhat large and the setting was prototypical, it advanced the research on studying the human visual perception during viewing pictures (an example of eye movements during image perception is shown in Figure 2.1). One of the prominent findings of such a study was that human gaze fixates on an informative region in a picture, rather than randomly selecting a fixation point. Following such an early study, Yarbus [Yar67] showed the differences of human eye gaze scan-paths in image perception depending on the task in which the subject is engaged. However, despite contributions of such previous work, it is still questioned that: "Why do we look where we do?" [Sch+11]

Figure 2.1: Eye movements during an image viewing task. The viewer was told to freely study the image. Each circle depicts a fixation (the longer the duration is, the larger the circle becomes.) and each line depicts a saccade. Fixations located on informative regions (faces, pillars, etc.). The picture was downloaded from http://en.wikipedia.org/wiki/Alessandro_Magnasco.

and the nature of our attention systems and eye movement mechanisms are not completely revealed. Recent studies on human eye movements however suggest several mechanisms for describing how eye movements are controlled. In [Sch+11], a layered structure is proposed for accounting for the control of saccadic target selection in visual perception. According to the layer, eye gaze is controlled by inter-playing different level of control circuits, which consists of salience, object recognition, value, and plans.

An outcome from prior eye movement studies is a profound development of eye tracking technologies [HJ10]. The precision and the sample rate of eye tracker becomes increasingly high. The size of whole apparatus becomes small and even portable. Such a development of mobile eye tracking technologies opens up the opportunities for analysis of eye gaze movements in daily environments [Bul+09], including outdoor scenes [Eva+12]. Nowadays, users do not have to sit in front of a desk to benefit from eye tracking applications.

In recent literature, two types of visual attention are mainly addressed [Hen03]. One is bottom-up visual attention and the other is top-down. Bottom-up visual attention is referred as a type of attention that is triggered by visual stimuli from environments. It is known that this type of visual attention is modeled effectively by a computational saliency map [Itt+98]. A saliency map is computed by combining multiple low-level feature maps (e.g. colours, intensity, orientation, and others) generated from an input scene image. This map depicts salient regions in the image (see Figure 2.2), which attract human visual attention and are likely candidates for fixation points in the image [FU08]. On the other hand, top-down visual attention is more task-oriented, i.e., the eye movements are driven by cognitive objectives such as object recognition, plans, and value. Although this type of top-down visual attention is rather complicated and still undiscovered compared to the bottom-up one, studies on individual tasks suggested several important factors [Ray09]. For instance, research on eye movements in reading has been a central issue of task-oriented attention studies [Ray98]. In reading, gaze regressions have been treated as a characteristic

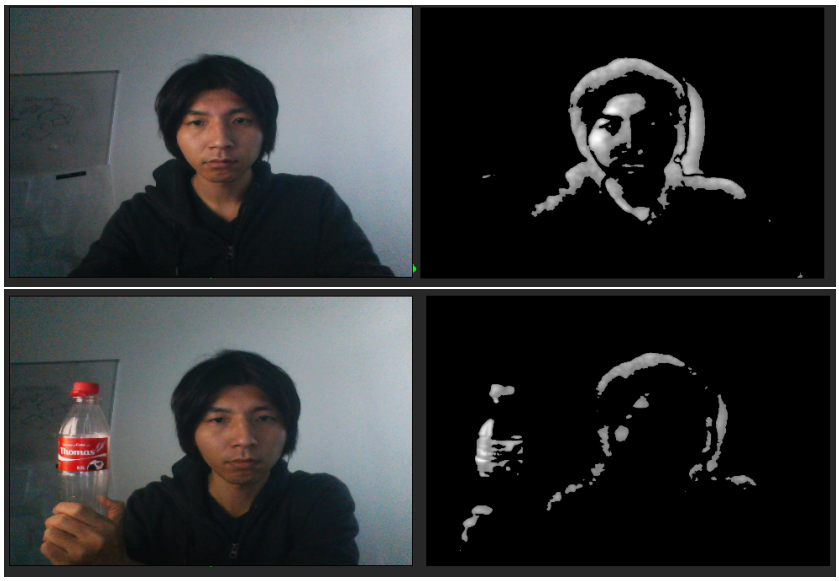


Figure 2.2: Examples of a saliency map. In the top image, the facial area is more salient than the other image region, whereas the label of a bottle is more salient than the facial area in the bottom image. We can clearly see the salient regions would draw human attention.

component for analysis in addition to saccades and fixations [Ray78]. Reader's eyes move backward when the reader has a difficulty to understand the text, misinterprets the text, or overshoots his target. Furthermore, another finding shows that saccade size in reading depends upon character spaces and not visual angle [MR81].

Another classical issue for analysis of top-down attention is eye movements in scene and object recognition [Hen+03; KT06; Sch+09; Ray+09; NH10]. An early study conducted by Parker [Par78] investigated eye movement patterns when viewers were trying to find differences between previously learned and currently viewed scene images. Based on the study, he reported that information is encoded from a wide area of the field of view, not only from the fixated area. It was found when the viewers could detect deletions of objects without fixating the regions. However, a recent study also showed contradictory results that peripheral regions of a fixation cannot sufficiently recognize objects [Hen+03]. The results indicate that fixations must be occurred on the object-of-interest in order to recognize the presence and the identity. Another interesting experiment also showed that presence of objects can predict fixation positions, regardless of the task that the viewer is engaged in [Ein+08a; NH10]. This indicates that eyes are fixated to the locations where objects exist rather than salient regions during recognition. In addition to the studies on image and scene perception, Schütz et al. have presented a study on letter recognition task while targets were moving [Sch+09]. It showed the influence of smooth pursuit and foveating saccades to a high-level object recognition task.

Visual search is also another main topic for analysis of top-down control of attention [DP04; NG05; Ein+08b; Eck11]. In the early 1960s, Neisser conducted a primary experiment for investigating cognitive operations involved in a visual search task [Nei64]. Two-stage models are introduced by such studies [Nei64; Hof79], which suggest that coarse processes are done in parallel in the first stage and it is followed by fine discrimination in

visual search. Such parallel mechanisms of visual search are also found in saccadic target selection [Fin97]. The ability of visual search shows that humans can process the visual information without actually directing the eye gaze to the stimuli. Importantly, visual search is one of the most influential paradigms to study covert visual attention [Eck11], which account for a type of vision awareness of the region on which the viewer is not fixating [Pos80].

As stated above, a number of studies have been contributed to understand the nature of humans' attentional system and eye behaviours. The findings from these studies are the basis for computer-human eye gaze interaction. In the following sections, I discuss how attention and human eye gaze could be used in applications of computer systems.

2.2 Eye Gaze-Based User State and Activity Recognition

Studies on eye movements have also contributed to discover the characteristics of eye gaze during individual activities and tasks [JC76; Ray95; LH01; Cas+09]. Historically, saccades and fixations, which are typical categorizations of gaze roles, have been mainly focused for analysis of eye gaze behaviour during various cognitive tasks and activities. In early work, Buswell already showed differences of viewer's fixations between different picture viewing tasks; the number of fixations and fixation duration differ when the viewer looks at picture with a particular instruction from free viewing [Bus35]. Rayner also showed characteristics of fixations and saccades during reading and information processing [Ray78; Ray98]. Besides, one of the challenging topics for the study of task-related eye movements is about gaze analysis in a non-visual task [Ehr+07]. In early studies, it was already investigated how eye movements differ between visual and non-visual cognitive states [Gaa66]. Prior studies already showed close relations of auditory stimuli to visual attention [Spe+00]. Based on these findings, one may conclude that even when a subject is not attending to vision, eye movement analysis supplies very meaningful information to understand the user's context. Covert orienting of attention [Pos80], however, may occur during several everyday activities. We should also take this type of gaze behaviour into account in order to trace a cognitive state of the subject. An important factor is that covert shift of attention to non-visual elements may occur even when a fixation remains in the same position.

Meanwhile, such studies on task-related eye movements established computational frameworks for prediction of the user state using eye gaze. Recent work on eye gaze-based CHI shows that a computer can predict in which type of cognitive processes, activities, and tasks the user is engaged based on his or her eye movement patterns. A typical application of such a system is reading detection using an eye tracker [CM01; Kea+03]. As shown in Figure 2.3, reading is a very characteristic activity of eye gaze. Campbell and Maglio [CM01]

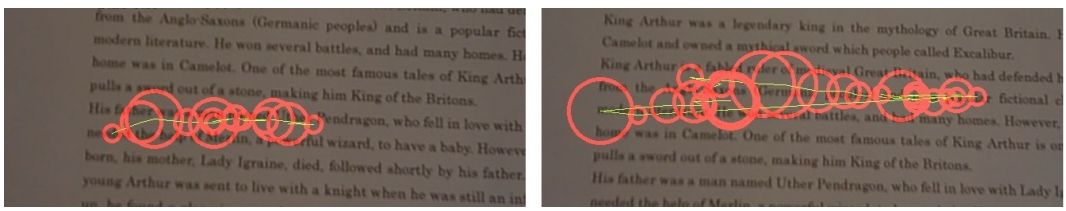


Figure 2.3: Eye movements during a reading task. The images are drawn with the same manner as that in Figure 2.1. Eye gaze saccades move normally horizontally during reading. Also, compare the difference to Figure 2.1.

developed an algorithm for reading detection using directions and distances of eye saccades. Keat et al. [Kea+03] proposed a recognition method using finite-state machines (FSMs) based on eye gaze movement directions. Recently, Biedert et al. [Bie+12] showed that a robust differentiation between "reading" and "skimming" could be possible. From their results, one can learn that a more fine-granular inference of user's cognitive task seems to be feasible. Another approach proposed by Bulling et al. [Bul+08] demonstrates reading activity recognition in a wearable scenario. They showed that robust reading recognition is possible even with a cheep eye movement measurement device so-called electrooculography (EOG).

Not only for reading, but also for other types of cognitive processes including visual search, scene memorization, and others, eye gaze analysis is very useful [Gid+13]. For instance, a study on eye movements during tasks with a desktop interface showed the characteristics of eye movements on each task [IB04]. In [Hen+13], it is presented that a classification of cognitive states (scene memorization, scene search, and text reading) is feasible using eye gaze features. A contradictory result was however reported by Greene et al. [Gre+12], where they showed that identification of images are possible with scan-paths but not classification of cognitive tasks. However, Coco et al. [CK14] claimed that it might be because of fixed viewing time for eye movement analysis and showed the classification is actually feasible. These prior approaches commonly employ gaze features of fixation duration, the number of fixations, saccade amplitudes, saccade directions and others.

User activity recognition using eye tracking is a relatively new challenge. As discussed in [LH01], eye movements could imply some daily activities. A couple of approaches were proposed for recognition of user activities in several scenarios. Courtemanche et al. proposed an activity recognition method during interaction with a computer interface using a machine learning approach called Layered Hidden Markov Models (LHMMs) [Cou+11]. In [Bul+11a; Oga+12], activities in desk work are recognized using an EOG. Asteriadis et al. [Ast+09] proposed a method to identify the user state (the level of interest and engagement) using eye gaze analysis in combination with head pose tracking. Furthermore, by combining image-based features with eye gaze movement patterns, Fathi et al. [Fat+12], Hipiny and Mayol [HMC12], and Shiga et al. [Shi+14] developed a recognition system for daily activities or actions.

Although inference of subject's high-level mental states, such as a language understanding level, is a very challenging task, Martinez-Gomez et al. showed that it could be recognized to some extent [MGA14]. Similarly, Kunze et al. proposed a method to spot a difficult word for the reader from his or her eye movements during reading [Kun+13b]. In addition to such cognitive states and processes, the user intention and uncertainty could also be estimated by eye movements [Pre+09].

This thesis presents a method for recognition of the user's cognitive state (Section 6.1 and Chapter 7) using eye gaze patterns. Recognition of the user's cognitive state could effectively drive analysis of user attention, especially for selecting an appropriate image analyzer for particular resources. Furthermore, it can also be used to infer a right moment for information presentation in an HMD. The cognitive state analysis method proposed in this thesis follows those prior approaches for activity and cognitive state recognition using an eye tracker.

2.3 Eye Gaze-Based Computer Interfaces and Applications

The development of eye tracking technologies brought a profound benefit for computer-human interaction: gaze as a computer interface. In early eighties, Bolt presented early work of gaze-based interfaces for use-computer communication [Bol82]. Several years later, Starker and Bolt proposed a system that employs human eye gaze input to control an object in a computer screen [SB90]. This system demonstrated the great potential of gaze as an interface for computers. In the meantime, Jacob also proposed an interaction technique using eye gaze, employing a dwell-time based approach [Jac90]. In his work, Jacob also addressed a common problem of gaze-based interaction so-called the "Midas Touch" problem, where the user activates an unintentional command by chance, as the Greek mythological king turned everything he touched into a gold regardless of his intention. Because eye gaze is always there unless we close our eyes, a good interaction system needs to take into account whether the user really intends to activate the command or not. Otherwise, unintentional outputs or activation would be obtrusive for the user.

Following the early work, a number of approaches and methods have been proposed for gaze-based interfaces [Duc02]. Two main distinctions are addressed here; i.e., gaze-based interfaces could be categorized into two different types: an *explicit* (*intentional* or *unnatural*) input or *implicit* (*unintentional* or an *natural*) input [JK03].

When one uses eye gaze as an *explicit* input manner, predefined gaze actions must be learnt by the user. So-called *gaze gestures* are a typical example for this type of gaze interaction [DS07; Wob+08; Roz+11; HR12]. Because such types of eye movements do not occur in natural interaction (see also Figure 2.4), the user could communicate with a computer without being interrupted by unintentional command activation. It is also important to mention that designing of gestures trades off the naturalness against the fatigue [Wob+08]. That is, the more unnatural the gesture is, the more inconvenient it is for the user. It was reported that the users fatigued with a traditional dwell-time-based gaze keyboard more than a letter shape-based gesture input method. This result shows that dwell-time-based gestures are unnatural for many users; thus, they are less acceptable to the users.

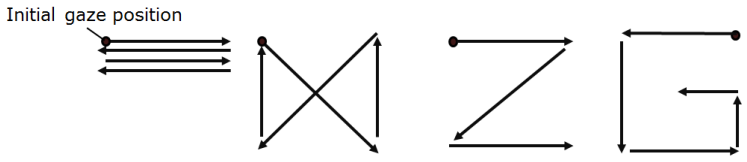


Figure 2.4: Examples of gaze gestures proposed in the previous work by [DS07; Roz+11]. By moving gaze focus following the path of a respective gaze gesture, the user can activate a command or input a letter.

On the other hand, users do not need special training steps for an *implicit* gaze interface. As discussed in the previous section, computers can predict which kind of the tasks or activities the user is engaged in based on natural eye gaze behaviour. In *implicit* inputs, the user interacts with a computer or an environment naturally and the system activates an appropriate command according to the tasks or activities. A typical application that incorporate implicit gaze input is implicit relevance feedback [Sal+03; Bro+06; Bus+08]. By analyzing reading behaviour of the user, a retrieval system can present more relevant

documents to the user for the focused topic. The idea of attentive documents [Bus+12] demonstrated the potential of a novel gaze-based interaction scheme that utilizes the analysis of user's natural reading behaviour. Not only text, a similar feedback framework could also be applied to images [Kla+08]. Also, slightly different but very relevant work by Yoshitaka [Yos13] proposed that similar natural eye movement analysis could also be used for image or video indexing, retrieval, and summarization.

Neither *explicit* nor *implicit* gaze input, one might need to consider *semi-implicit* gaze interaction methods. With semi-implicit interaction methods, the user activates commands with natural interactions but he or she knows how to activate them. For instance, iTourist [QZ05] is an application that outputs information of city attractions according to the user's interest measured by an eye tracker. This type of applications that dynamically present information to the user (in real-time) are designed to track the user natural eye gaze and attention. However, the user sometimes moves his or her eyes to deliberately activate the function because they can predict what would happen in the consequence of their gaze behaviour (e.g., looking at an object longer than normal in order to activate information provision).

Gaze-based interfaces can be found in wide areas of CHI applications. Typical examples are assistant systems for people who have handicaps: gaze-typing [MR02; Han+04], communication interfaces [Hor+06; Cal+13; V¨+14] and others. Since eyes are merely a communication source for those who cannot speak and move bodies, gaze-based interfaces are spotlighted intensively. Furthermore, for interaction with virtual environments [TJ00; Vat+05; HB09] or virtual characters [Bee+10b; Bee+10a], gaze-based interaction is widely used. Nowadays, not only for usability, research, or efficiency purposes, but also for an entertainment purpose, gaze-based interfaces are used. The eyeBook [Bie+09] showed that reading experiences can be enhanced by integrating a recognition framework of user's reading behaviour. Apart from desk-mounted settings, we also have various opportunities for gaze-based interaction in mobile or wearable scenarios. Especially, eye gaze-controlled environments [Bon+09; Shi+07] and wheelchairs [Mat+01; Bar+08] draw growing attention in the field of mobile gaze-based interaction. In life-event logging [O'H+08], mobile eye tracking is a practical tool. Using an eye tracker, we can log various types of daily events that can be associated with our attentional behaviours: such as reading events [Kim+13; Kun+13a].

To summarize this section, I conclude that a number of eye gaze-based interfaces proposed to date showed the benefits and feasibilities. The above-mentioned prior work strongly shows the advance of the gaze-based interfaces and the promises. Many techniques and methods proposed in this thesis are inspired by the prior work.

2.4 User Context-Awareness in Computer-Human Interaction

Depending on the application, context may have a different meaning. A context-aware system is understood as a computer system that understands the user context and can process the data accordingly. In many applications, context-aware systems could be employed, especially where users need a special assistance such as healthcare, driving, tourism, etc. [Mak+09; MN13]. Existing systems show that context-awareness can enhance effective computer-human interaction.

Although context has historically been a key topic for Natural Language Processing,

recent developments of mobile computing technologies have brought the idea of context-awareness in the field of CHI [Sch+99]. In mobile computing, the physical environment surrounding a user is monitored by various types of sensors. By processing the information acquired from sensors, computers are able to recognize the user activities, tasks, locations, and other types of user contexts. A typical example of such a system is an activity recognition application using an inertial measurement unit (IMU) integrated in a mobile phone [Kwa+10]. Since many people recently own (a) mobile phone(s), such a type of mobile applications have great potential. In addition to such a classical activity recognition framework, state-of-the-art approaches for context-aware computing adapt various types of wearable sensors. For example, an electroencephalography (EEG) is a popular brain-computer interface for recognition of the user's mental (emotional) state [Lot+07; PH10]. Li et al. presented a recognition method for oral activities using a sensor embedded in teeth [Li+13]. A depth camera, which can be even worn, is also used to recognize egocentric activities [Mog+14]. In early applications, only a sole sensor was used. However, in recent applications, muliti-modal sensors are largely integrated into a system for collaboratively collecting information as much as possible [Mae+10; LC12].

Attention-awareness is a growing research topic [RT06; BK06] and also a key factor for context-awareness [Anc+12]. CHI always involves interaction with users. Thus, user attention must be intensively addressed for attentive user interfaces (AUIs) [Ver03]. Furthermore, an attention-aware system can be directly associated with a location-aware system if the system is also aware of the spatial environment [ST13a; Orl+14a]. A mobile eye tracker can be used to relate the user visual attention with his or her needs for mobile multi-modal context-aware computing [Bul+11b]. A fusion of multi-modal inputs including a mobile eye tracker can already be used in a practical scenario of context-awareness [Son+13; Web+13].

In this thesis, I discuss eye tracking-based attention-awareness, which directly contributes to context-awareness. Knowing to which type of visual content in a scene the user is attending, one can infer which type of cognitive context the user is engaged. For example, if the user attention is directed to text, we may infer that the user is reading. This thesis presents the benefit of attention-awareness, especially focusing on cognitive context-awareness.

2.5 Image Analysis with Integrated Attention Direction

As previously discussed, humans direct attention to a particular stimulus in a scene in order to perceive the scene more efficiently and rapidly. The ability called *active vision* is also simulated in a robotic vision system, where attention does not normally exist [But+08; Ras+10]. In many situations, a technical system has to recognize an object which is only partial in the field-of-view, for taking a subsequent action, i.e. decision making. For such active vision systems, a saliency map is widely used. Using the map, the system can effectively predict a relevant region-of-interest. Furthermore, recent studies showed that saliency maps are even useful for scene text [Sha+12] and face [Ban+04] detection. In [Sha+12], a couple of saliency maps are evaluated in terms of scene text detection.

Although saliency maps can model human visual attention very well with a bottom-up manner, it cannot be used to analyze top-down (task-oriented) visual attention. In order to overcome this weakness, we utilize an eye tracker. With an eye tracker, a gaze position in a scene can be captured easily. Captured gaze positions are used to track the user's visual attention, including top-down one. In many existing systems, eye gaze is used to estimate

the region-of-interest in a scene image [Shi+07; Ish+10; Beu+12; Beu+14]. Consequently, the systems can recognize what is in the user's focus based on analysis of the region-of-interest. That is, image features are extracted from a limited area (region-of-interest) to recognize the visual content attended by the user. This kind of user attention-guided image analysis could be applied to various types of visual information resources: objects [Ish+10; Shi+07; Beu+12], faces of people [Beu+14], etc. Furthermore, also activities of the user could be classified using similar user attention-guided image analysis approaches [Fat+12; HMC12]. Unlike eye tracking on a computer display or on a tablet PC, computer systems do not have a priori knowledge of the relevant content in a natural scene unless we set up a controlled environment. The prior work showed that user attention-guided image analysis can play a very important role for attention-aware systems in natural scene images.

The framework proposed in this thesis relies on the image analysis methods which integrate attention direction as introduced above. In the proposed framework, we need to identify to which visual content in natural environments the user is attending. Thus, eye gaze and image analysis must be tied.

2.6 Gaze-Based Interfaces in Virtual, Augmented, and Mixed Reality

The early work by Bolt [SB90] already revealed the potential of eye tracking as an interface for computer systems. In his proposed system, a virtual reality environment can be controlled by user eye gaze. Traditionally, gaze has been used as an effective input modality for interaction in virtual reality [JK03]. Several approaches were proposed for gaze-based interaction in virtual reality [Jim+08]. Recent studies also showed that accuracy and latency of-state-of-the-art eye tracking are good enough for interaction in virtual reality both with a monocular [Pfe08] and binocular [Pfe+08] setup. Also, eye tracking is widely used in a specific application of virtual reality where researchers simulate a professional task such as aircraft inspection [Duc+00]. An advantage of eye tracking in virtual reality is that the system knows the entire environment (it can also be referred as a controlled environment) in the virtual space [Bar+11] unlike natural (physical) scenes. Thus, as long as eye gaze is accurately tracked, the system can correctly identify the object or content that is attended by the user.

Recent developments of wearable displays further extend the potential of gaze-based interface integrated augmented and mixed reality (AR/MR) applications in mobile scenarios. For such wearable AR/MR systems, gaze-based interfaces are helpful since eye gaze can be controlled intuitively by the user without moving hands of arms, which might reduce fatigue [Zel+05]. Eye gaze input is for instance effectively incorporated in order to present information in an appropriate position in an AR view [IJ11]. By combining an eye gaze-based information presentation management method with an image-based management method such as [Orl+13a], we can develop a more effective information presentation system for AR from an ergonomics point of view. Nilsson et al. presented a video see-through AR system that employs eye gaze input [Nil+09]. Although the presented system was still prototypical, they showed the promise of eye gaze-based see-through AR systems. The idea is extended by Lee et al. [Lee+11] where they used an optical see-through HMD. The system allows the user to interact with content in the display using eye gaze. Another prototypical application proposed by Ajanki et al. [Aja+11] augments the contextual information of daily scenes in

a video see-through HMD based on the user's attentional behaviours. In their pilot study, a dwell-time-based method is used to detect attention on particular content (faces of known people, AR markers, and speech) in a physical scene. When a certain type of attention is detected, the contextual information is overlaid onto the physical content using the HMD. Such an AR system can also assist the user when he or she is involved in a particular task. An important point is also addressed here; when the user is gazing at the augmentative information in the display, the eye gaze behaviours become different from those in natural interactions. This finding tells us that one should carefully design gaze-based AR systems when he or she also integrates a user cognitive state or activity inference framework. Because information provision in the AR might distract the user attention, it could alter natural eye gaze behaviours.

In this thesis, I present new AR systems that employ gaze input. Particularly, I use a see-through type of HMD where gaze depth can be effectively used to analyze the user's attention engagement with the display. The findings from the prior studies and the approaches used in the existing systems are the foundation of the proposed frameworks in this thesis.

Chapter 3

Background and Overview

This chapter presents an overview of the proposed architectures and backgrounds, including the hardware settings used throughout the thesis and state-of-the-art technologies. First, I summarize the basics of eye tracking technologies and describe the eye tracker used in this thesis (Section 3.1). Then, a framework of gaze-guided image analysis is presented (Section 3.2). By limiting the image region according to the user's gaze position, one can analyze the visual content that is attended by the user. Next, the hardware settings of the see-through HMD are described (Section 3.3). I also present how the mobile eye tracker and the see-through HMD are combined to construct a prototype of eye-trackable see-through HMD. Additionally, an overview of see-through wearable display technology is presented. Finally, an overview of the comprehensive framework that combines all recognition modules and information presentation modules is presented (Section 3.4). This comprehensive system demonstrates the proposed wearable user attention-aware interaction systems as a whole.

3.1 Eye Tracking

3.1.1 Overview of Eye Tracking Technologies

A number of eye tracking approaches have been proposed. Many of them use light sources to capture reflections in the eye(s). For instance, an early method by Buswell [Bus35] uses a six-volt ribbon-filament lamp as a light source and the reflection in the cornea is recorded in films. In general, an eye tracking system calculates the direction of the eye pupil from the reflection of (a) light source(s). The direction of the eye pupil is then mapped to a respective space such as a computer screen or a physical environment. Although it is also possible to detect pupil direction without using light sources, the accuracy is much higher when we use them. Nowadays, many eye trackers use infra-red light sources in order to track the eye movements with eye-reflected images [MM05]. Recent do-it-yourself eye tracking approaches show potential [Man+12]. However, the off-the-shelf eye tracker products available today are still more accurate and reliable. In this thesis, an off-the-shelf mobile eye tracker called the SMI Eye Tracking Glasses (ETG) is used. The specifications of the ETG are summarized in Section 3.1.5.

In addition to such light reflection-based eye tracking, there are other types of eye tracking approaches. For instance, an EOG is used as a reasonably available eye movement measurement apparatus in a mobile scenario [Bul+08]. Although the EOG cannot pinpoint

the user's point-of-gaze, it can track the muscular movements of eyes. It is mainly used for recognition of gaze activities. Another approach for eye tracking is proposed by Nishino and Nayar [NN06]. They use a corneal reflection image of the user's eyes without using a particular light source.

3.1.2 Calibration

In general, calibration is a prerequisite process before actual eye tracking. It is especially required in point-of-gaze eye tracking. In a calibration process, the system calculates internal parameters for eye tracking such as individual eye ball sizes and eye ball center positions in the images. A couple of approaches were also proposed for eye tracking calibration. In this section, I explain a typical calibration process. As shown in Figure 3.1, an eye tracking system calculates the parameters based on the relations between the known calibration points and the current gaze positions on the calibration plane. In the calibration process, the system shows each calibration point successively on the screen. The viewer is requested to look at each of them for a certain amount of time in turn. Thereby, the system estimates the user eye positions in the eye tracking system.

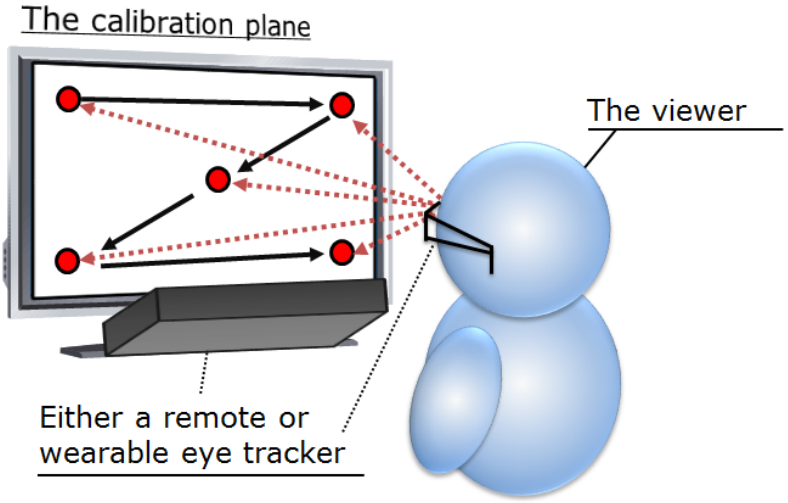


Figure 3.1: Eye tracker calibration process. The viewer looks at each point successively. This calibration process is almost the same for both a wearable and a remote eye tracker.

Generally, the accuracy of eye tracking is guaranteed as long as the viewer is gazing at the calibrated plane. This is the main reason that eye tracking on a computer screen maintains the accuracy compared to that in a natural scene. As shown in Figure 3.2, in a 3D space, the off-set of gaze position is large when the user gazes at a far plane. Principally, it is very difficult to perform perfect calibration (to calculate the precise parameters) because viewers cannot keep fixating on the exactly same position. Gaze drifts slightly even if the subject tries to keep fixating on one position. Even small differences of calculated parameters can make large differences on non-calibrated planes as shown in the figure. Additionally, when one uses a monocular eye tracker, the gaze position on far or near planes deviates largely. The problem is also known as the *parallax problem* (described later in Section 3.1.4).

By increasing the number of calibration points, one can increase the accuracy of eye

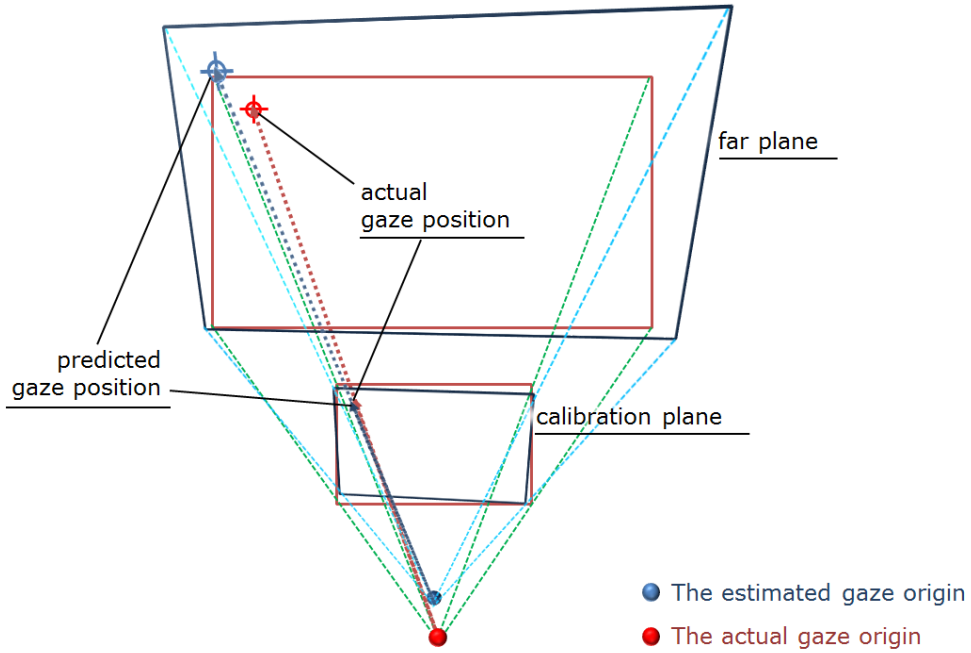


Figure 3.2: Eye tracking calibration problem in a 3D space. Although the off-set of the predicted gaze position is very small on the calibration plane, it becomes large on the far plane.

tracking. However, it brings extra burden to the users. Although several approaches have been proposed for calibration-free eye tracking [Koh+08; Nag+09], a calibration process is still necessary for accurate point-of-gaze eye tracking.

3.1.3 Remote (Stationary) vs. Wearable (Mobile)

Since the eye tracking apparatuses in early decades were somewhat major and the processors were not as powerful as the current ones, most studies were performed with a stationary setting. For such a stationary setting, remote eye trackers have been developed. With a remote eye tracker, experimental subjects do not need to wear the eye tracker. Instead, an apparatus for eye tracking is installed in the environment (usually mounted on a desk) as shown in Figure 3.3 (left). With such a remote eye tracking setting, eye movements are measured stably within a confined space, which is usually a computer screen. However, the subjects normally have to keep their head positions fixed during the eye tracking experiments or studies. When the head position or orientation changes, a re-calibration process is required because the parameters also change. Some eye trackers can also track head movements [CM06] in order to compensate gaze position shifts caused by head movements (free head motion).

Recent developments of wearable computing devices enable us to use eye trackers in a mobile scenario. Just recently, eye-glasses type eye trackers (shown in Figure 3.3 (right)) were developed by several eye tracker manufacturers. These kinds of wearable eye trackers usually calculate the gaze position in a scene image captured by an outward (egocentric) scene camera. Advantages of wearable eye tracker are as follows: i) it can be applied to mobile environments and ii) the subject can move his or her head. However, as mentioned

Figure 3.3: Remote eye tracker (left) and wearable eye tracker (right). A remote eye tracker is usually mounted on a desk and captures eye gaze on a computer screen. On the other hand, a wearable eye tracker captures a gaze focal point in a natural scene.

previously (Section 3.1.2), the accuracy of a gaze position in a 3D space on non-calibrated planes is not as high as the calibrated one. Furthermore, the gaze position with a wearable eye tracker is relative to the head movements, i.e., the same coordinates in scene images do not necessarily indicate the same location in the physical space. If one needs to track absolute gaze positions in the physical space, he or she must either track the scene images (or head motions) as well as eye gaze or map the relative gaze position into the physical space.

3.1.4 Monocular vs. Binocular

A monocular type eye tracker calculates the direction of a single eye whereas binocular one calculates the directions of two eyes, as shown in Figure 3.4.

In optics, a human eye and a camera can be treated similarly since both are image sensing devices which have pupils and lenses. Therefore, the image in human eye is modeled similarly by the camera model of computer vision. Particularly, a binocular (stereoscopic) view, where one has two cameras (or eyes) for capturing the same scene with different perspectives, may be modeled by *epipolar geometry*. In epipolar geometry (also refer to Appendix A) , a point in a camera image corresponds to an epipolar line in another camera image. Thus, the gaze position, which shows the point-of-focus of human eyes, is principally projected as the epipolar line in the eye tracker's scene camera. If we calibrate the gaze position on a

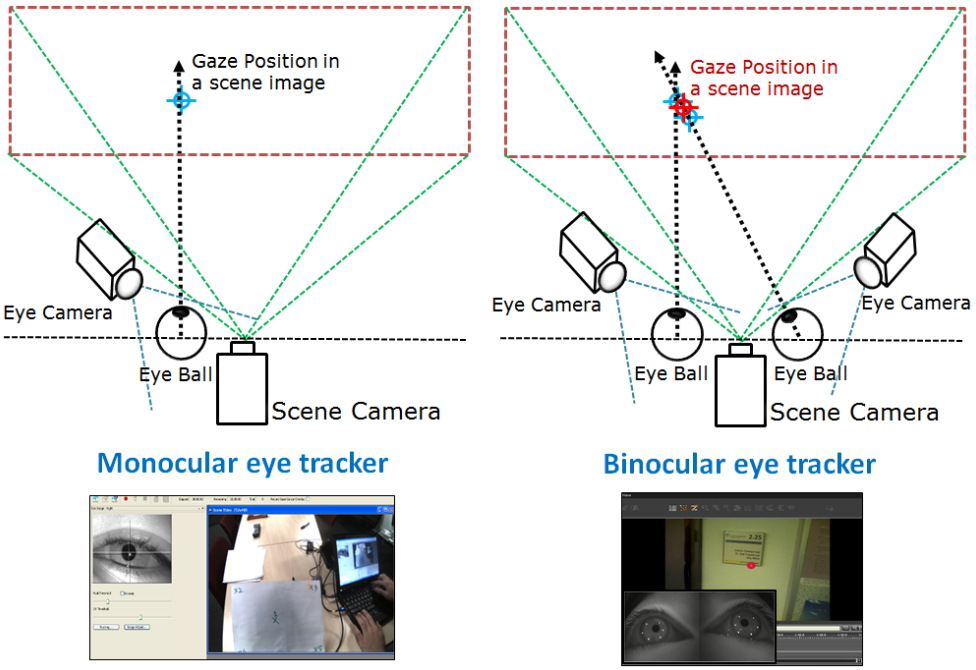


Figure 3.4: Monocular eye tracker (left) and binocular eye tracker (right).

particular plane, the system can calculate the point-of-focus as the point on the calibration plane (not as the epipolar line). Accordingly, a monocular eye tracker can determine the gaze position using only one eye direction.

However, a monocular eye tracker inherently has the *parallax problem*. The parallax problem refers to the problem that the positions in two field-of-views differ when they have different point-of-views. A monocular system cannot calculate accurate gaze positions on non-calibrated planes since the system knows the difference of point-of-views (between an eye and a camera) on the calibrated plane only but not on other planes. Consequently, gaze positions on non-calibrated planes cannot be as precise as the calibrated one, which is a similar problem that we have in a mis-calibration case as shown in Figure 3.2. Using two eye directions in binocular eye tracking, one could compensate such a problem [Kwo+06]. The intersection of two gaze vectors provides us with the 3D coordinate of point-of-focus in the given space. Once we calibrate the two eye cameras with respect to the scene camera, the intersection of gaze vectors in the 3D space can be mapped to the point-of-gaze in the scene camera image. Thus, we can calculate a gaze position more accurately with a binocular eye tracker.

In addition to the compensation for accurate eye tracking, a binocular eye tracker has another advantage. One can also calculate the focal depth of eye gaze. Gaze depth can be effectively used for gaze-based interaction. In this thesis, I discuss gaze-depth-based interaction methodologies using see-through type displays (Chapter 6). Using a see-through display, virtual images can be overlaid onto the physical environment at a different depth but in the same line of sight from the user's view. In order to determine with which focal interaction plane the user's eyes are engaged, I propose a focal depth calculation method using a binocular eye tracker. As a result, we can detect whether the user is focusing on the content in a virtual display or in a physical environment.

3.1.5 Own Eye Tracking Setup

As previously stated, I use the SMI Eye Tracking Glasses[1] (ETG) as my own setup for the eye tracker throughout this thesis. Pictures of the ETG are shown in Figure 3.5. The ETG

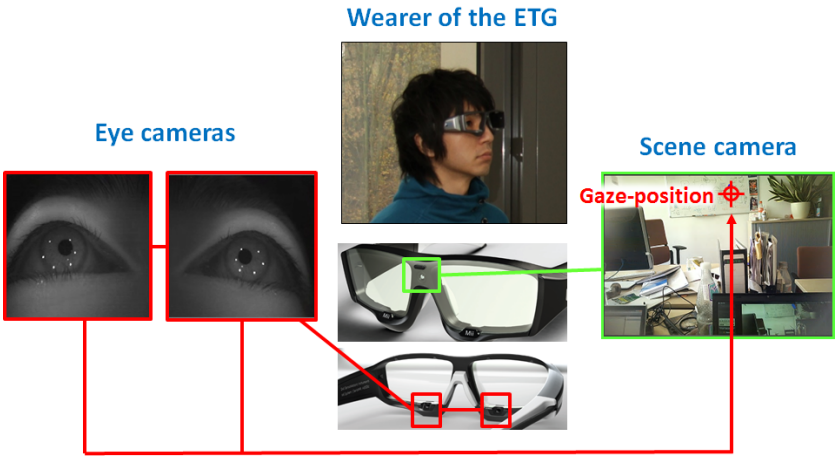


Figure 3.5: SMI Eye Tracking Glasses. Each eye is illuminated by six infra-red light sources.

are a binocular wearable eye tracker. As shown in the figure, each eye is illuminated by six infra-red light sources. The ETG calculate the pupil positions and eye gaze directions based on the reflections of these light sources. Then based on the spatial relations between eye balls and the scene camera, two gaze vectors in the space are mapped to the gaze position in a scene image (such as shown in Figure 3.4).

The specifications of the ETG are summarized as follows:

Glasses weight:	78 g
Sampling rate:	50 Hz binocular
Gaze tracking accuracy:	0.5° over all distances (typ.)
Gaze tracking range:	80° horizontal, 60° vertical
Scene camera resolution & frame rate:	1280x960 at 24 fps
Scene camera field of view:	60° horizontal, 46° vertical

For calibration of the ETG, we can select three options. The first option is the one-point calibration which requires the user to look at one calibration point. For many users, this option works quite well. The second option is the three-points calibration. This calibration is used when the one-point calibration does not perform well. Although one can expect higher accuracy with the three-points option than the one-point one, the calibration process normally takes longer. As a special feature of the ETG, the SMI has developed the so-called '0-point calibration', which is actually a calibration-free option. However, the gaze position with the 0-point calibration has a large off-set with many users. To be on the safe side, I use the one-point or the three-points calibration in all experiments conducted in this thesis. The ETG's gaze tracking accuracy is 0.5°. This value indicates that we might have an off-set with 1cm at a distance approx. 1m (114cm).

[1] http://www.eyetracking-glasses.com

3.2 Basic Framework for Gaze-Guided Image Analysis

The system proposed in this thesis consists of several components that analyze the user attention and image content in various scenarios. The most fundamental components for user attention analysis are respective gaze-guided image analysis modules. In this section, I present a basic framework that is adopted by most of the proposed image analyzers.

3.2.1 Scene Image and Gaze Coordinate: Recording and Streaming

Once we start the eye tracking process with the ETG, the ETG starts capturing the scene images and calculating the gaze coordinate in each scene image as shown in Figure 3.6. For offline analysis, we record a scene image video with synchronized eye gaze data. The video data and eye gaze data are saved separately. When we perform an analysis, the recorded video data and eye gaze data are loaded and coupled again.

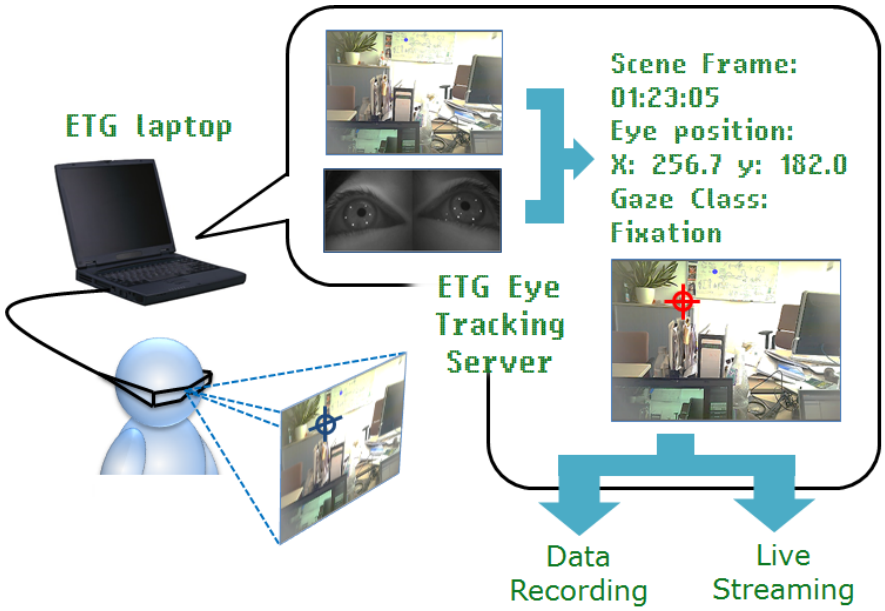


Figure 3.6: Eye Tracking recording and streaming.

Additionally, we also have a live streaming option with the ETG for online (real-time) application usages. SMI also released the APIs for online streaming, which can be integrated in one's own codes. Calling the API modules from one's own codes, he or she can process scene images and the eye gaze data in real-time. I use these APIs in the proposed applications which run in real-time.

3.2.2 Image Crop

For recognition of the object or image content at which the user is gazing, I propose an image cropping step which limits the entire scene image to a local gaze region only. In Figure 3.7, I illustrate examples of an image cropping step. As discussed in Chapter 1, everyday natural scene images may contain various types of visual content. To recognize the visual resource which is relevant to the user's current context in such a complex scene,

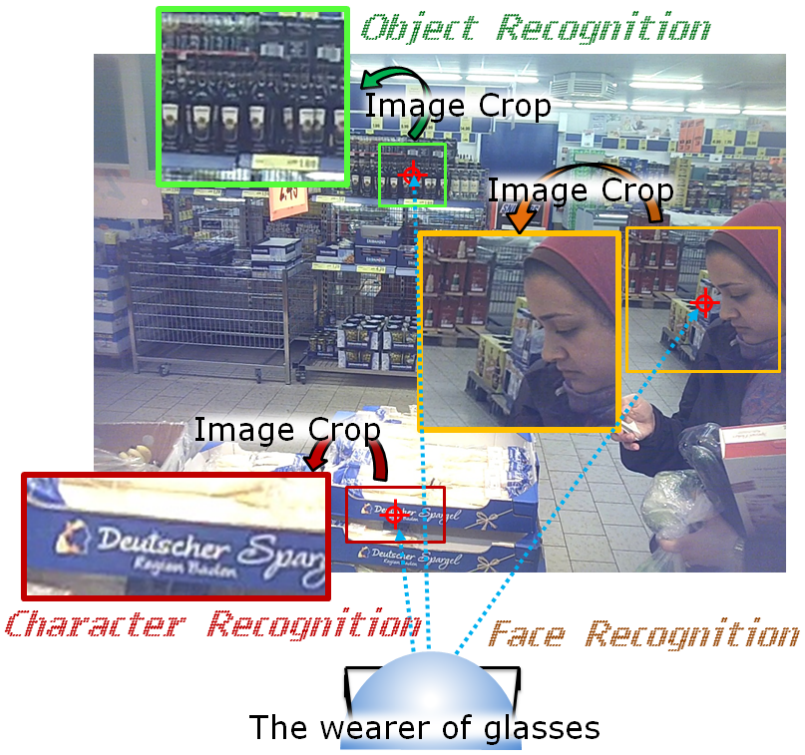


Figure 3.7: Gaze-guided image analysis. The local gaze regions are cropped for image analysis of individual content.

we prioritize the eye gaze input. I crop a local image region from the entire image so that the gaze position comes center in the local image as shown in the figure. In the proposed *gaze-guided image analysis* modules, only cropped regions are used for the recognition of image content. The sizes of local regions are adjusted according to the content type. For example, if the content type is "English text", a horizontally wide region is cropped because English text is usually aligned horizontally. I present a couple of image analysis methods that utilize this gaze-guided approach in Chapter 4 and 5.

By limiting the image region only to the user attended area, we can i) speed up the image analysis process and ii) identify the user attention in a complex image. In Chapter 4 and 5, I discuss advantages and disadvantages for individual gaze-guided image analysis methods.

3.3 Eye Gaze with a See-Through Wearable Display

Using a see-through type of display, one can present augmentative information to the user as superimposed virtual images onto the physical world. I combine a see-through display with an eye gaze-based interactive system. A great feature of see-through type displays is that the user's field of view is not occluded by the display; the user can see the physical environment through the virtual display. This section presents an overview of see-through wearable display and the setup I used in this thesis.

3.3.1 Overview of See-Through Wearable Displays

An early head-worn display already existed in late sixties [Sut68]. It allowed the user to see a stereoscopic image in a virtual screen. Historically, displays that can be worn on a head have been called a head-mounted display (HMD) [JH07]. Because early HMDs were large and heavy (usually large devices must be mounted on a helmet), they cannot really be worn by the users. Recently, rapid developments of wearable displays have been seen. Many HMDs available nowadays are wearable and portable. One of the most spotlighted gadgets of recent wearable displays was the GoogleGlass[2] produced by Google. The advent of such a light-weight wearable computer shows the potential of applications of wearable displays in everyday scenarios.

Additionally, see-through features are integrated in many recent wearable displays. Not only immersively viewing a virtual world, but the user can also see the physical world with augmentations. A wide area of AR/MR applications have been proposed using see-through wearable displays [Zho+08].

3.3.2 Optical See-Through vs. Video See-Through

Overall, there are two distinctions of see-through display. One is called "optical" see-through and the other is called "video" see-through. The differences between these two types of see-through display are summarized in Figure 3.8. As we can see in this figure, a virtual image is superimposed onto the physical world with an optical see-through display. Thus, the user can see the physical world directly as well as the overlaid information with a virtual image. Normally, a virtual image projection is reflected by a semi-transparent mirror for

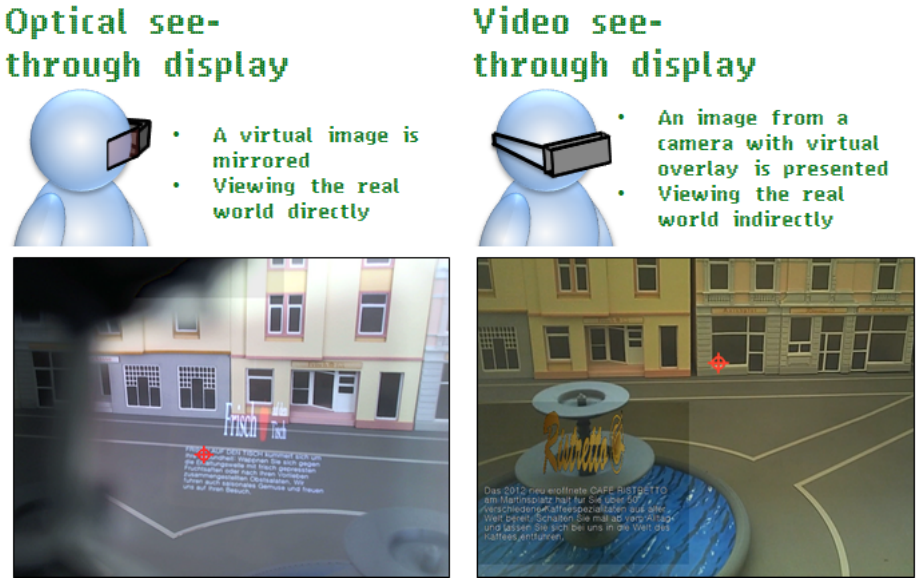


Figure 3.8: Differences between an optical and video see-through display. The user can see the physical world directly with the optical one whereas the user see camera-captured images in the immersive display with the video one. In this optical see-through display example, the virtual image projection is reflected by a semi-transparent mirror.

[2] https://www.google.com/glass/start/

optical see-through as shown in the figure. Major examples of optical type see-through displays are: Epson Moverio[3], Google Glass, and Brother AirScouter[4].

On the other hand, the user can see the physical world through a camera image using a video see-through display. This type of see-through display can be implemented more simply than the other one since it does not require any special mirror. However, with a basic setup, a field-of-view of video see-through display is relatively narrow compared to an optical one and the image is not stereoscopic (The user can only see a planer camera image). An advantage of video see-through is that the virtual image overlays onto physical objects are easier than the optical one since it can be directly overlaid onto the camera image. Video see-through displays commercially available are for example Vuzix Wrap 920AV[5] and Oculus Rift[6] (stereoscopic).

In the existing AR/MR applications using see-through displays, a video see-through has been widely used. An optical see-through display is still expensive and a calibration of a virtual display is also challenging. In order to overlay a virtual image onto a particular location in the physical environment, one needs to calibrate the display, i.e., calculate the spatial position of the display in the environment. Although several approaches have been proposed for such display calibration [TN00], it is not as easy as a video see-through display, which usually does not require display calibration.

3.3.3 Own Setup: Eye-Trackable See-Through Display Eye-Wear

I use the Brother AirScouter as the see-through display in this thesis. The AirScouter is an optical see-through monoscopic HMD. The specifications of the AirScouter are summarized as follows:

Hardware weight:	106 g
Display resolution:	800x600
Angle of view:	22.4° diagonal
Focal length:	$0.3 - 10.0$ m (adjustable)

One of the goals of this thesis is to implement an attentive information presentation system combining the eye gaze analysis and the see-through HMD. Thus, I needed to build a prototypical eye-wear that can track the user's eye movements and presents information through the display. In order to build the own eye-trackable see-through display, I assembled the ETG and the AirScouter using a 3D-printed snap-on frame as shown in Figure 3.9. This apparatus allows the system to track the user's eye movements in both virtual and physical environments and to overlay virtual images onto the physical world unobtrusively. From the user's field of view, the virtual display can be seen as Figure 3.8 (left).

For dynamic information overlay, we also need to calibrate the display with respect to the eye tracking system. In Section 5.2, I present an approach for the display calibration method using an image analysis technique.

[3] http://www.epson.de/moverio
[4] http://www.brother.com/en/news/2011/airscouter/
[5] http://www.vuzix.com/UKSITE/consumer/products_wrap920av.html
[6] https://www.oculus.com/ja/

Figure 3.9: Assembling own eye trackable see-through display. 1) The ETG and the AirScouter. I printed a snap-on frame for fixing these two devices. 2) and 3) The ETG and the AirScouter is combined with the snap-on frame. 4) A complete eye trackable see-through wearable display. The position of the AirScouter could also be switched to left-eye.

3.3.4 Eye Gaze Measurement in the HMD

For eye gaze interaction with a wearable display, we need to detect eye gaze in the display. Moreover, we also need to check whether the user is focusing on the display or the physical environment.

Figure 3.10 is an illustration of eye gaze measurement in a see-through display. The focal length of the AirScouter can be adjustable from $0.3 - 10.0$ m. When the focal plane of the display is near the calibration plane of the eye tracker, the gaze position on the display can be linearly projected using the scene image coordinate. On the other hand, if the focal plane of the display is not near the calibration plane, the gaze position in the 3D space must be mapped to the screen coordinate based on the *homography* between the camera space and the display space. Hence, I propose two approaches for eye gaze detection in an HMD depending on the scenarios. First, an eye gaze direction-based approach is employed when the focal length of the display is near the calibration plane. This approach only uses the eye gaze coordinate in a scene image that is calculated by the eye tracker. The other approach called an eye gaze depth-based approach uses focal depth calculation (as we discussed in Section 3.1.4). I present these methods in Chapter 6 and evaluate them as well.

Once we establish the method for eye gaze measurement in the wearable display, we can implement several *attention-driven AR display functions* that utilize eye gaze in the display. For example, when the user is not attending to the display, a virtual image would be obtrusive to him or her to view the physical world. An *automatic dim* is one of the proposed interaction functions that automatically dims the virtual image in the display when the user is not attending to the display. Additionally, an *eye-con* is another proposed interaction function which activates a respective predefined command according to the user attentional

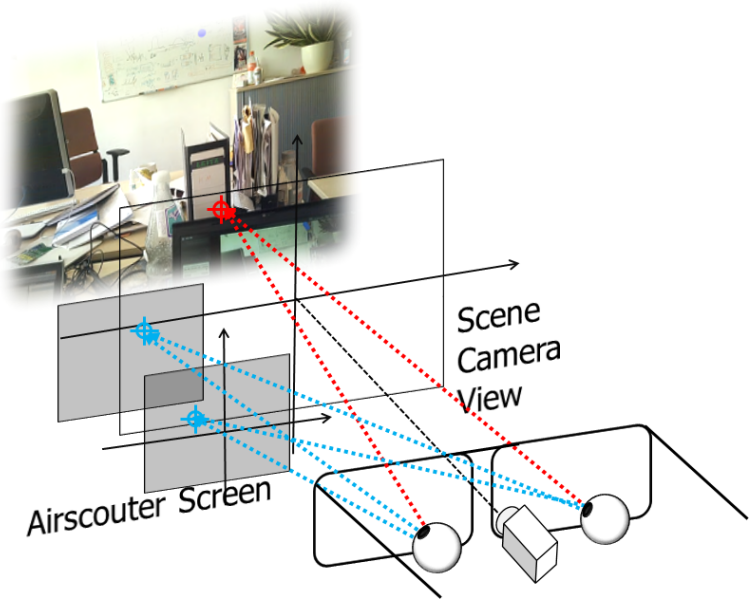


Figure 3.10: Eye gaze in the display or in the scene. When the focal plane is near the calibration plane of the eye tracker, we can directly use the coordinate in the scene image. When it is not, we must calculate gaze position in the 3D space.

gaze on icons in the display. I also present these interaction functions in Chapter 6.

3.4 Comprehensive Framework for Attention-Aware System

In this thesis, I present approaches and methods for recognition or analysis of user attention in individual everyday scenarios and for computer-user interaction using eye gaze. Although each recognition or interaction component such as the *gaze-guided image analysis* or *attention-based AR display functions* can work standalone for individual applications (e.g., Museum Guide 2.0, ERMed, etc.), I also propose a comprehensive framework for demonstration of the intelligent attention-aware interactive system as a whole.

3.4.1 Attention-Aware Interactive System

The proposed attention-aware interactive system allows a flexible and intelligent interaction exploiting individual attention and image analysis modules. The main feature of the comprehensive framework is that image analysis is driven by the user's cognitive state predicted from the eye gaze patterns. That is, the system predicts the type of the user's attention, recognizes the attended visual content, and provides the user with appropriate information regarding the content. For instance, when the user reads textual descriptions of an art exhibit in a museum, the system recognizes reading behaviour and starts a text recognizer. If any supplementary information is found with respect to the recognized text, the system presents the information in the see-through display at an appropriate position and moment. The comprehensive framework consists of several components for attention analysis, image analysis, attentive interactions, and information presentations.

Users can benefit from the proposed interactive system in various daily scenarios, especially in a complex situation where various types of visual information resources such as text, objects, and others are present.

3.4.2 Architecture

Figure 3.11 shows the architecture of the comprehensive framework. As stated above, the system consists of several attention and image analysis, and gaze-based interactive components. Individual components are described in respective chapters (Chapters 4 - 7). Roughly speaking, the architecture could be separated into four sections: the *eye gaze*

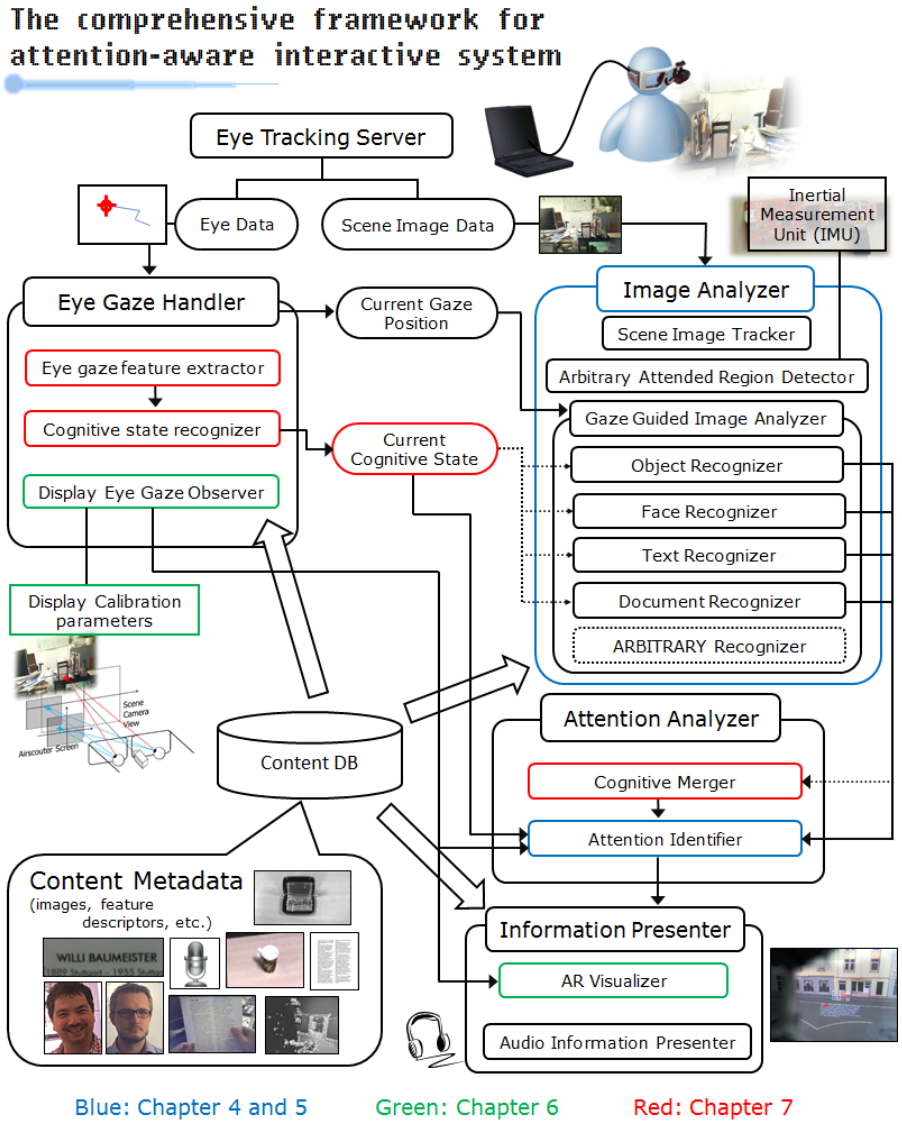


Figure 3.11: Architecture of the proposed comprehensive attention-aware interactive system. The entire architecture consists of several components proposed in this thesis. Each component is also presented in each chapter.

handler, the *image analyzer*, the *attention analyzer*, and the *information presenter*.

- Eye gaze handler: receives eye gaze data, extracts gaze features, recognizes the user's cognitive state, and analyzes eye gaze in the display.
- Image analyzer: processes image data and recognizes content in an image.
- Attention analyzer: analyzes the user attention to content and classifies the cognitive state the user is engaged with.
- Information presenter: provides the user with augmentative information, either auditory or visually (as an image).

As shown in the figure, an additional *ARBITRARY* image recognizer could be incorporated as an add-on in this proposed architecture. Individual components load respective data, such as images, features, meta-information from the content database (DB). The meta-information is used to present the user with augmentative information by the information presenter. The user can have access to information as a virtual image overlaid onto a scene or as audio output through earphones or speakers. In Chapter 7, I discuss the comprehensive framework for user-attended visual content analysis.

Chapter 4

User-Attended Visual Content Analysis: Objects and Faces

This chapter presents user-attended visual content analysis (VCA) approaches for non-textual content. As prominent cases of non-textual visual content, I focus on objects and faces in this thesis. In contrast to textual content, fixations during perception of these types of visual content are located on rather unpredictable regions since the cognitive tasks involved in these types vary (e.g., memorization, recognition, free-view, etc.). Thus, I mainly focus on duration of time where the user attends to the content rather than eye gaze movement paths drawn on the content to recognize attention on objects and faces. In particular, I propose a method for detection of *attentional gaze* (AG) on visual content.

First, I discuss the user attention detection on objects in a controlled setup (Section 4.1). I focus on a museum scenario where various types of objects are present in a scene. Museum visitors would attend to art exhibits for a certain amount of time. I propose a method to detect the user attention to individual exhibits, combining eye tracking and object recognition systems. Adopting the proposed detection method, I also propose an application that presents augmentative information to the visitor regarding the attended exhibit. I present the experiments for evaluating the proposed method in a museum scenario.

Next, I focus on the user's attention to arbitrary objects in an uncontrolled scenario (Section 4.2). Since objects are unknown to the system in an uncontrolled scenario (i.e., objects are not in the predefined database), we cannot rely on object recognition unlike the aforementioned museum scenario. This section presents a method for detection of attention to arbitrary visual content based-on a spatial eye gaze analysis combining an eye tracker with an inertial measurement unit (IMU). In the proposed system, we may also apply an online image recognition API to the detected attended content in order to infer the object label.

Finally, I discuss the face recognition and learning system driven by eye gaze (Section 4.3). Similar to the object recognition system, eye gaze is used to identify the face in a scene image attended by the user. This identification is also useful for an online face learning system. I present an online face recognition and learning system which can aid user's memory for facial images.

Each section in this chapter is based on the work presented in [Toy11; Toy+11; Toy+12a], [Toy+12b], and [Toy+13a; ST13b], respectively.

4.1 User-Attended Object Recognition

In this section, I present i) basics of the object recognition framework, ii) an object recognition method guided by the user's eye gaze, iii) a ground-truth generation approach for the user's attentional gaze, iv) evaluation methodology, and v) experiments and results.

4.1.1 Introduction

As discussed in Chapter 2, research over the last century has contributed to understanding the nature of human attention by analyzing eye movements using eye tracking [Bus35; Hen+03; Yar67]. As a result, eye tracking itself has emerged as a new technology to interact with computers. Since people generally prefer simple and intuitive interaction mechanisms to complicated or incomprehensible ones, any kind of interface available today could be replaced by a simpler and more intuitive one. From this viewpoint eye tracking is a highly remarkable technology due to its immediate connection to human intuition.

Wearable eye trackers available today provide a lot of opportunities to interact with the surrounding environment intelligently, for instance by using eye tracking with AR. An AR system presents a view of the real world whose elements are "augmented" by computers in several ways (such as embedding signs, sounds, etc.). Recent smartphone applications like Wikitude[1] or Google Goggles[2] present a platform to overlay information about things in the physical world onto a mobile phone display.

These advances are due to recent progress of image-based object recognition technologies. The objective of image-based object recognition is to recognize the objects present in an image or in a video stream in the same way as humans do. Early studies in object recognition started to employ global features such as color or texture histograms [Har+73]. However, since such global features were not robust enough to illumination or perspective changes and occlusions, methods based on local features became more common [Zha+06]. Local features, which are extracted from small patches of an image are widely utilized nowadays [LA08]. In particular, Scale-invariant feature transform (SIFT) [Low99; Low04] is broadly used due to its invariance to scale, orientation, and affine distortion. Based on these methods, recognition systems can be developed that have excellent robustness against lighting and position variations, background changes, and partial occlusion [Pon+06; RW08].

We investigate how human gaze can be used for attention-aware image analysis. First, we develop algorithms for guiding object recognition by using fixation points. Then, we present how to detect a user's attention in a controlled scenario, given raw eye tracking data and the corresponding object recognition results. Finally, we develop a novel application called *Museum Guide 2.0* that utilizes eye tracking as an interactive interface and recognizes

[1] http://www.wikitude.org
[2] http://www.google.com/mobile/goggles/

objects in a real environment. The proposed application demonstrates the feasibility of our algorithms in practice.

Note that there are several related applications that also integrate the object recognition system with an eye tracking application, such as [Bon+09; Ish+10]. However, evaluations of the benefits of the integration were not discussed deeply in the previous work. Here we present a new approach for triggering the information provision and the evaluations of the approach including a user study in a practical use-case.

4.1.2 Scenario - Museum Guide 2.0

The basic idea of *Museum Guide 2.0* is that visitors of a museum wear a head mounted eye tracker while browsing museum exhibitions. Whenever the visitor looks at any of the exhibits for a certain duration, the system automatically presents corresponding AR meta-information in a certain way (such as a human guide might do). Figure 4.1 shows an abstracted image of the Museum Guide 2.0 scenario. The benefit of using gaze in this scenario is that the

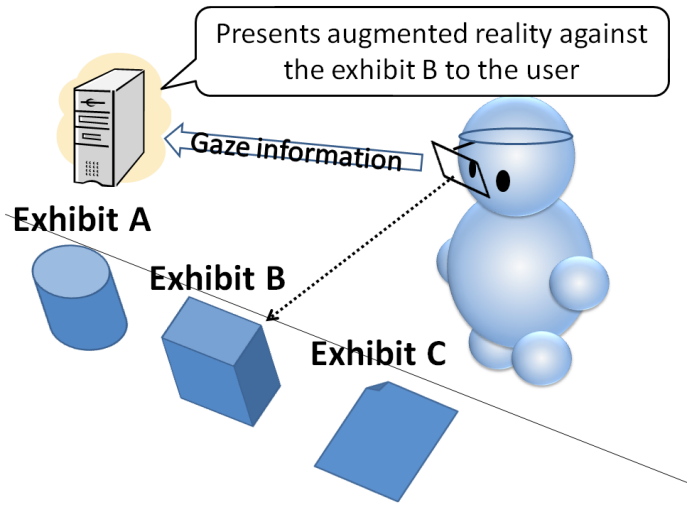


Figure 4.1: Museum Guide 2.0. The system detects attention of the user on a particular object and presents additional information regarding the respective object using augmented reality.

system does not require the user to perform any other action but looking at the object to get the information. The main considerations that inspired us to this application are:

- A known set of objects: All exhibits in a museum are known and we can obtain as much training data as we need.
- Controlled environment: All exhibits in a museum are well illuminated, static background, and are not cluttered.
- Limited perspective: Users usually look at exhibits from some typical directions only.

Considering these aspects, we can start from a restricted scenario that still contains a lot of challenges. In Section 4.2, I discuss how we can extend the proposed framework to remove the restrictions.

Museum Guide 2.0 works as follows: The scene camera captures the front view of the user while the eye camera observes the eye movements. When a fixation is detected, the image from the scene camera is piped to the object recognition framework together with the coordinates of the fixation. The recognition framework returns the name of the object the fixation points to. Our attention detection algorithm judges if the user actually attends to the very object or his eye gaze fixates on the position rather occasional. If the former is the case, Museum Guide 2.0 presents AR meta-information of object to the user.

4.1.2.1 Objects in Museum Guide 2.0

There are many types of exhibits in museums. One can categorize these types into "2D large-sized object", "2D small-sized object", "3D large-sized object" and "3D small-sized object" as shown in Figure 4.2. In the experiment, we focus on *3D small-sized objects*

(a) 2D large-sized object (e.x. painting on a wall)

(b) 2D small-sized object (e.x. picture)

(c) 3D large-sized object (e.x. dinosaur skeleton)

(d) 3D small-sized object (e.x. ancient pottery)

Figure 4.2: Four different categories of exhibits in a museum. We separate large-sized objects and small-sized objects whether the object is larger than human or not.

because of the following two reasons. First, recognition of a 2D object is relatively simple compared to that of a 3D object. One only needs one (or a few) image perspective(s) for 2D object recognition, whereas a multiple perspectives might be necessary for 3D object recognition. Thus, in order to test the image recognition approach in a more challenging environment, we use 3D objects. On the other hand, when the object is too large, the user attention may be captured by a local part of the object, rather than the entire object itself. In this work, we consider that the user attends to an entire object for simplicity of the evaluation.

4.1.2.2 Challenges

Although Museum Guide 2.0 is a simple and uncomplicated scenario, it contains many challenges.

First, ordinary object recognition systems often suffer from high computational cost and cluttered backgrounds. A significant difference of the proposed system to ordinary object recognition systems is that we have a fixation point which is directly connected to the user's interest point in the image. By taking this advantage, we extend a basic object recognition method to *fixation guided object recognition* to speed up the recognition process and to increase the recognition accuracy.

Second, the most primitive way to evaluate the recognition system is to judge if the system outputs the correct name of the object indicated by the fixation for each frame. However, such a way of evaluation does not consider a well known problem, i.e., the so-called Midas Touch problem [Jac90]. Since the eyes are one of our key perceptual organs, they provide a large amount of information to the human. Besides, the movements of the eye (fixations and saccades) strongly reflect the mental process of viewing. Many saccades and fixations are not caused by the user on purpose but are rather subconscious. If the application responds for each frame individually, the overflow of the user with irrelevant information would not lead to any acceptance towards the application. Therefore, one needs to define another criteria to evaluate the system based on the user's *attentional gaze* (AG) which can be observed as a sequence of fixations on a particular object rather than a fixation for each frame. Furthermore, to satisfy such an attention-based evaluation method, we propose the methods to detect the existence of AG on a particular object by using the object recognition result of consecutive frames.

Third, since image processing generally requires high computational cost, one might need to reduce it when he or she applies the system in a real-time environment, where system reactions to user behavior should be triggered with minimal delay. The majority of processing time however is occupied by the SIFT feature extraction and matching. While the processing for one fixated image area is done by the system, other fixations might occur in the meanwhile. Queuing these events for later processing is not suitable for a real-time system. To catch up with real-time, we propose a compensation approach, i.e., by counting the number of frames during the recognition, the system keeps the latest information.

4.1.3 Overview of Object Recognition in Computer Vision

In this subsection, I summarize object recognition approaches in literature, together with the background and the state-of-the-art.

The goal of object recognition is to identify (an) object(s) present in an image. Object recognition has been one of the most challenging topics for computer vision. To name an object present in an image only having that visual properties (appearance) is not trivial to computers while humans can do that very naturally. Raw image data has very simple representation: each pixel value shows color intensity of a very small region in a scene. Although a computer has data of the pixel values of an image, it cannot tell "what" it is from such simple representation. One could imagine that the simplest method for object recognition is to compare each pixel value of a given image with images for references (image database). However, such a simple approach may not work when image properties change: scales and poses of the object in the image, light conditions of the image, and

others. Thus, we need to extract *image features* from the image, which would account for traits of the object in terms of image representation and train image classifiers in order to let the computer recognize the object.

In early work, holistic image features such as texture histograms [Tam+78], color histograms [Nib+93; NS92] and eigenspaces [MN95] were employed for object recognition. Such approaches enabled computers to recognize objects and scenes robustly even images have minor differences. However, such holistic image features could not solve recognition problems when images have considerable viewpoint and light changes, occlusions or cluttered backgrounds. Especially, occlusions and cluttered backgrounds are main challenges for such holistic feature-based recognition approaches. If there are other irrelevant objects included in an image, the system cannot compute proper features. We may apply an *image segmentation* to an image; however, known as the *chicken and the egg dilemma*[3], it is hard to segment image regions without knowing the objects in the image and vice versa.

Meanwhile, local (part) feature-based approaches showed their robustness against scale changes, image occlusions, and cluttered backgrounds. Detection of interest keypoints enables computers to match similar image parts between images [Web+00]. Thus, objects in the image can be detected even a part of the object is visible. In particular, SIFT and Speeded-Up Robust Features (SURF) [Bay+06] are well-known feature descriptors for such local features. Image retrieval and object classification problems were advanced in recent years due to the invention of such local image feature detectors and descriptors.

Recent studies showed potential of global image representation based on local image descriptors. So-called "bag-of-keypoints" (bag-of-visual-words) approaches [Csu+04] are a typical example of such methods. They are effective for image (object) categorization or classification, whereas part-based local feature matching approaches are more effective for object instance identification or detection.

In addition to the developments of image feature extraction methods, a number of machine learning methods have been presented for object recognition. The Support Vector Machine (SVM) is one of the most popular approaches in the object classification domain [PV98]. The state-of-the-art researches show that neurologically inspired learning structures such as Deep Neural Networks (DNNs) [Cir+12] and Multiple Kernel Learning (MKL) [Yan+12] are effective for challenging object classification tasks. Successes of these classification models that deal with complicated feature models suggest that a massive amount of data may be essential for complex images or object classification tasks.

In the proposed framework, we adopt a rather simple framework using local feature matching due to computational efficiency. Because we focus on a particular scenario, there are not so many objects; thus, a complicated architecture is not necessary and a simple framework would work sufficiently.

4.1.4 Method

First, we describe the method for basic object recognition (used as the base of our proposed object recognition method). Then, we describe our *real-time gaze-based object recognition* method for Museum Guide 2.0.

[3] "Which comes first, the chicken or the egg?", here it is questioned that which is needed first, the segmentation or the recognition.

4.1.4.1 Basic Object Recognition Method

In the object recognition framework, we adopt SIFT with the DoG as the feature extraction and description method. These features are used to identify the object in a query image from a pre-built database. To achieve fast computation, we also use the Approximate Nearest Neighbour (ANN)[4] method for matching features.

A brief model of the basic object recognition process is shown in Figure 4.3. First of all, we build a database consisting of SIFT features from images of all objects to be displayed in the museum. Object recognition is processed by finding the most similar features from the database when features from the query image is given. The name of the object which has the majority of matched features is returned as the result.

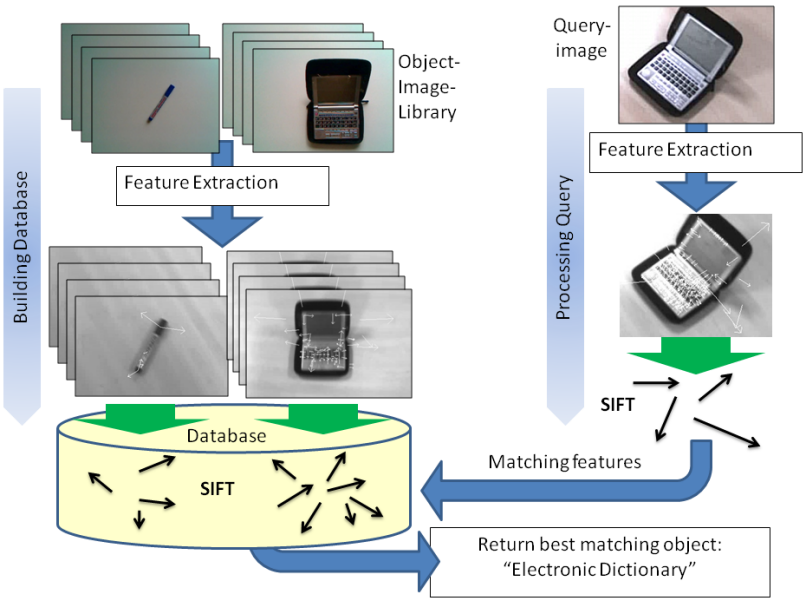


Figure 4.3: Object recognition process. SIFT features are extracted from images and matched between the database and the query.

SIFT We detect interest keypoints using DoGs (Difference of Gaussians) and describe the detected keypoints using SIFT. In Figure 4.4, I show an illustration of keypoint detection using DoGs. To detect interest keypoints of SIFT using DoGs, a given image is first converted to a gray-scale. Then, the gray-scale is filtered by Gaussian Kernels with multiple scales of Gaussian parameter σ to generate blurred images. By changing σ, one can control the blurriness of generated images. The convolution $L(x, y, \sigma)$ of the given image $I(x, y)$ with a Gaussian function $G(x, y, \sigma)$ is given by

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y),$$

where $*$ is the convolution operation and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right).$$

[4] http://www.cs.umd.edu/~mount/ANN/

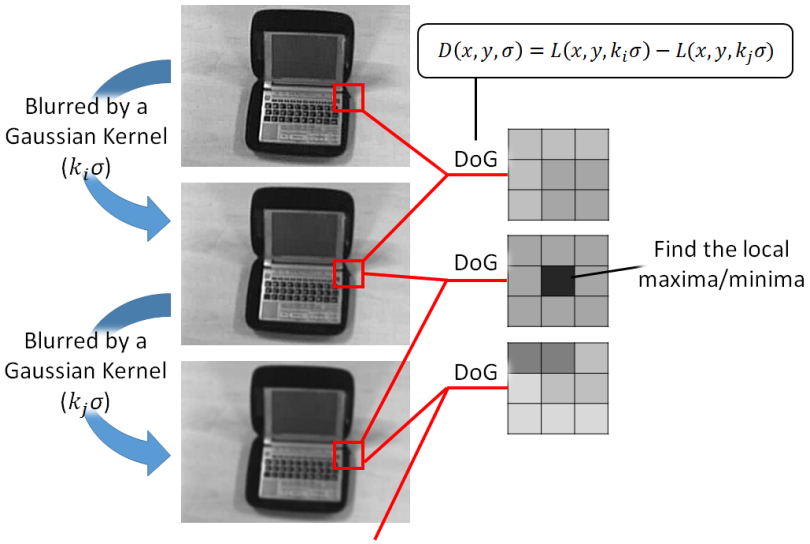


Figure 4.4: Interest keypoint detection using DoGs.

Then, candidates of interest points are detected as the local maxima/minima of the Difference of Gaussian (DoG) between the ith scale $k_i\sigma$ and jth scale $k_j\sigma$. A DoG is given by

$$D(x, y, \sigma) = L(x, y, k_i\sigma) - L(x, y, k_j\sigma).$$

Since some candidates that are located on the edge or whose surrounding areas are low contrast are sensitive to distortions, such unstable candidates are removed after the detection process.

On the description stage, a squared region (a 16×16 window) around each interest point is cropped at first. Then, the region is rotated according to the dominant gradient orientation within the region to acquire rotation invariance. The region is divided into $4 \times 4 = 16$ sub-regions and for each sub-region a histogram of the gradient of 8 orientations is computed. Thus, each feature is represented as a vector that has $16 \times 8 = 128$ elements. To enhance affine invariance, the vector is normalized to unit length. For SIFT features, more detailed descriptions and evaluations can be found in [Low04].

Database for Object Recognition For object recognition, one has to build a database which contains the features of the objects to be recognized. Since we assume a museum scenario, pictures for creating a database can be taken under similar conditions that are later given for the run-time system as described in Section 4.1.2, i.e., with similar illuminations and spatial arrangements. Additionally, to keep up with an identical environment to the use case, we also use the same camera and the same resolution.

As shown in Figure 4.5, people usually have typical viewing perspectives when they watch exhibits in a museum. Pictures of objects are taken only from these typical perspective angles.

In the following, I describe the process of building a database.

1. Place the object on a table.

2. A person wearing the eye tracker walks around the table, thereby directing the scene camera to the object.
3. Record a video using the scene camera.
4. After recording a video, find images taken with different viewing angles and extract SIFT features from the selected images manually.
5. Label images with the identity of the object.
6. For all objects, repeat the procedure from 1 to 5.

To have a good object recognition performance, the images must be taken from several viewing angles. It is especially needed when an appearance of object changes drastically due to the viewing angle.

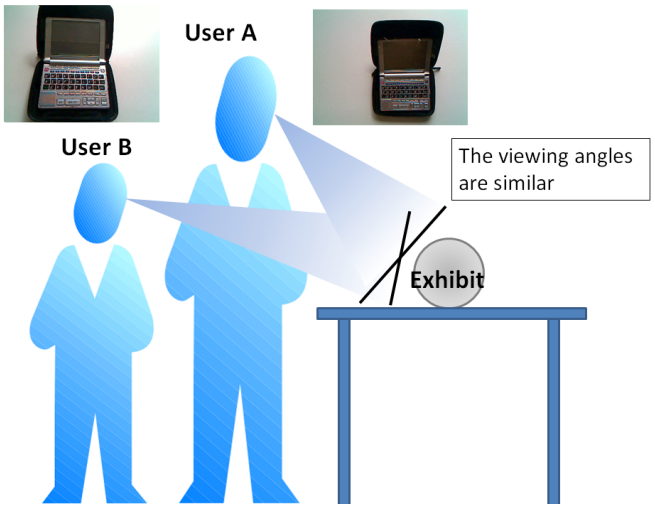


Figure 4.5: Typical angles for viewing exhibits in a museum. We take pictures of objects from these angles.

Matching In the recognition phase, an image of an unknown object is given as the query to the retrieval system. After extracting SIFT features from the query image, our object recognition method retrieves the nearest features in the database for the individual SIFT features from the query. As the measurement of the feature distance, we use the Euclidean distance metric. The object identity of the closest match for each feature is retrieved from the database and a histogram is built representing how frequently a particular identity was retrieved. The histogram is normalized to unit length to remove disproportion of the number of features. Eventually, if the highest value in the histogram exceeds a threshold value, the identity of the corresponding object is returned as the recognition result. If none of the entries in the histogram exceeds a threshold value, no recognition result is returned.

The computational cost of the presented object recognition method using local descriptors does not only depend on the number of features to be retrieved for each image, but also on the number of stored or indexed features in the database. The larger the number of indexed features is, the longer the processing time is required. To accomplish real-time

processing, it is necessary to reduce computational cost of individual feature matching. We use an ANN search method to reduce computational cost for matching features. In the ANN search, the nearest feature to a query is returned with a certain error bound ϵ [IM98]. As the value of ϵ increases, the retrieval becomes faster but the probability of having a retrieval error also becomes higher. Thus, it is normally required to find a suitable value of ϵ depending on the size of the database and the number of queried features per image.

4.1.4.2 Real-Time Gaze-based Object Recognition

In this subsection, we propose *real-time gaze-based object recognition* that overcomes the problems stated in Section 4.1.2.2. The main objective here is to develop a computational approach that detects the existence of the user's attention, i.e., AG on a particular object using eye gaze information and a scene image from the eye tracker and to enable it to meet a real-time requirement.

Fixation Guided Object Recognition A quite distinct point of our object recognition system compared to an ordinary camera-based one is that we have not only images from the scene camera but also fixation points. A typical object recognition system has to deal with images that have complex background. In such a complex image, it is not easy to locate where the object-of-interest is. Hence, for example, when an image is highly cluttered, the recognition task becomes quite difficult. Unlike such an ordinary recognition system, we can take an advantage of fixation information which often indicates the location of the object-of-interest in the image.

Ideally, we want to locate only the object-of-interest, including the object contour. However, although the methods for estimation of a contour of an object (image segmentation) have developed recently and is one of the active topics in computer vision [Zha+08; Arb+11], the technology is not mature enough to be utilized in real-time. Therefore, we simply crop a rectangular region from the image centered on the fixation point and extract SIFT features from that local region. The region is chosen to be large enough to contain sufficient interest points for reliable object recognition.

Generally, performance of local feature-based object recognition relies on the number of the features extracted from the query [Kis+10]. Here, we select n features closest to the fixation point for use in object recognition. For example, when 50 is a given number for features, the 50 closest features to the fixation point are used for object recognition as shown in Figure 4.6. Limiting the number of features not only enables the object recognition module to work "locally" on the object of interest, it also speeds up the recognition process for complex objects. Assuming that k features were originally extracted from the rectangular region around the fixation point, the number of features actually used for object recognition would be $\min(n, k)$.

In addition, by expanding this *non-weighting fixation guided recognition method* described above, we propose another method called *SIFT feature weighting fixation guided recognition method* that reasonably utilize geometrical configuration of features.

The eye position is considered as the point where the user is mostly interested at the moment. In other words, the interests of the viewer decreases as the distance from the gaze position increases. This insight gives us the idea to weight SIFT features according to the distance from the fixation point. Hence, when building the histogram (see Section 4.1.4.1), more weight is given to the features close to the fixation point as compared to those far

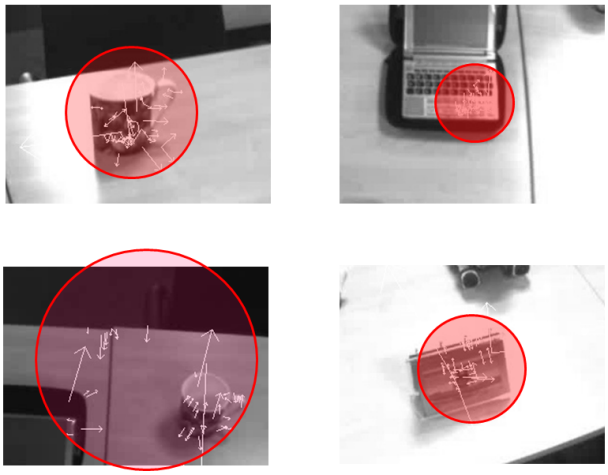


Figure 4.6: SIFT features extracted from local image regions. The circles in the images represent the approximate region in each image. Note that the distribution varies for each object depending on the complexity of the object.

away. In this *SIFT weighting method*, we employ a Gaussian function to weight the vote in the histogram, i.e. the weight w of the feature that has a Euclidean distance d from the fixation point is given by

$$w(d) = \exp\left(-\frac{d^2}{a}\right),$$

where a is a given parameter.

Each weight is added to the corresponding identity in the histogram. Finally, the histogram is normalized to unit length as before.

Estimation of the Eye Position on a Recognized Object Using the above-mentioned recognition method, we can recognize the identity of the fixated object, i.e., "on which object" in the scene the user is fixating. In addition to recognition of the object identity, we could also estimate the fixated position of the attended object. By matching local features between the query image and the reference image in the database, we estimate from which perspective the user is viewing the object. Then, the gaze position in the query image is simply mapped to the reference image.

Suppose $\boldsymbol{fq}_i = (fq_{ix}, fq_{iy})$ is the coordinate of the ith matched (the closest) feature in the query against the reference feature $\boldsymbol{fr}_i$. The $\boldsymbol{fr}_i$ is a feature extracted from reference image X. Our goal is to estimate the homography $\mathbf{H}$ between the query image and the reference image X, which projects the coordinate as follows:

$$\boldsymbol{fr}_i = \mathbf{H} \cdot \boldsymbol{fq}_i.$$

Using the random sample consensus (RANSAC) approach [Li+05], we find the most likely homography $\mathbf{H}$ of the given correspondences $\boldsymbol{fq}_i$ and $\boldsymbol{fr}_i$. Thus, the eye position in the

query image $\boldsymbol{E_q}$ is projected to the eye position in the reference image X, by

$$\boldsymbol{E_r} = \mathbf{H} \cdot \boldsymbol{E_q}.$$

Attentional Gaze-Based Ground Truth Processing In order to apply any kind of benchmarking or evaluation to the system results, one needs to define the so-called *ground truth* – a manually labeled result that represents the ideal system output. We need to model the time intervals, in which the user really attends to a specific object presented, not an unconscious glance. The primitive manual tagging however, which is made on the basis of frames, needs to deal with noise which occurs through unconscious eye-movements and respective fixations. As the data that we manually tag with labels are the individual frames, the frames representing noise will also be labeled. In order to judge, whether a fixation to a specific object can be considered as noise or as actual attention, we need to define where is the border of attentional and unattentional fixations.

To identify the event of a user attending to one specific object, we analyze the stream of fixations based on the following observations:

- When we attend to an object, the *duration* is usually longer than that for any unattentional fixation or glance.
- Fixations are not necessarily restricted to the object of interest for the whole duration that a person is gazing at it. The eyes might also fixate other objects or backgrounds for short periods without consciousness. These fixations might be considered as *noise*.

Hence, in this context we propose a computational method for the presence of an attentional event based on a sequence of fixations on one specific object X. The number of fixations on that object X must be longer than the *duration threshold* T_{dur}[5] but may also contain a certain amount of *noise*, which we consider as fixations on objects other than X.

Attentional gaze (AG) is the basic element to trigger information provision for a particular object. Therefore, we explain how AG is detected from manually labeled video frames in detail in the following.

Suppose we have a video stream containing scene images with corresponding gaze coordinates. Each frame is manually labeled as the identity of the object being indicated by the corresponding fixation. If there does not exist any fixation for that frame or the fixated object is not in the database, the frame is labeled as "undefined". From frame number zero, successively the labels of the frames are examined. When an inspected label is a defined object X, the algorithm starts to count the number (duration) of the frames F_X that have the label X. While counting up the X-frames, if the number of consecutive frames that are *not* labeled as X (considered to be "noise") F_{noise} exceeds the noise threshold T_{noise}, the sequence is dropped (F_X is set to zero). As soon as the duration F_X exceeds the duration threshold T_{dur}, the sequence starting at the first frame with label X (where the recent counting started) is recognized as AG on object X. This AG ends at the last frame with the label X when the noise F_{noise} exceeds the noise threshold T_{noise}.

Figure 4.7 shows an example of AG detection given a sequence of labeled video frames. In this example, we set the noise threshold value $T_{noise} = 2$ and the duration threshold

[5] To find an optimal value of this threshold T_{dur}, we conducted experiments in which the user had to give explicit verbal feedback when he was looking at some object with consciousness. We evaluated, which threshold values yielded best result w.r.t. this spoken ground truth.

value $T_{dur} = 3$. At frame number 3, the label *"mug"* appeared for the first time and we thus start counting up F_{mug}. Until frame number 8 (where the duration reaches T_{dur}), it does not contain any consecutive noise frames more than 2, thus it is recognized as AG on the object *"mug"*. But for the next sequence of *stapler*-labels, the noise F_{noise} exceeds T_{noise} before the duration $F_{stapler}$ reaches T_{dur}. Consequently, the sequence of the frames is dropped, $F_{stapler}$ is set to zero.

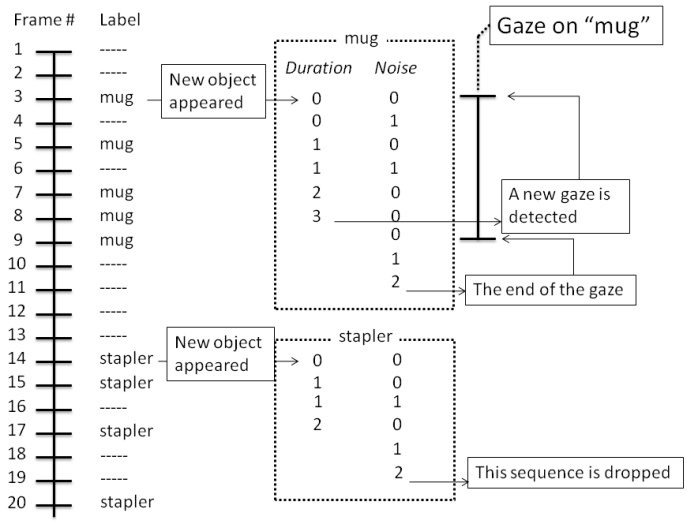


Figure 4.7: An example of AG detection with the noise threshold value $T_{noise} = 2$ and the duration threshold value $T_{dur} = 3$. The label "- - - - -" represents "undefined". Note that AG was detected for the object "mug", whereas no AG was detected for the object "stapler" owing to the higher amount of noise in the stapler sequence.

We investigate videos and eye tracking data with varying T_{noise} and T_{dur} thresholds to evaluate to what degree detected AG by this algorithm matches expressed consciousness within our ground truth data of attention. These experiments and their evaluation allowed finding suitable values of these thresholds.

Attentional gaze-based ground truth is obtained as a series of AG on each object which is detected by the described process and *real-time gaze-based object recognition system* aims to recognize them using results from *fixation guided object recognition* so that Museum Guide 2.0 can automatically start to present proper AR for an object X.

Attentional Gaze Detection Methods Based on Recognition Results As criteria for evaluation, *attentional gaze-based ground truth* is obtained by the process stated in the previous subsection. Now we need to discuss how to detect the existence of AG from results of the fixation guided object recognition framework. Let us for now disregard the real-time requirement and assume that our object recognition process can be processed for each frame that has a fixation. As a result, every such frame then contains a respective machine generated (recognized) label denoting the object in focus. In the following we need to verify whether the user's gaze is attending to that object or whether it can be considered as an unconscious glance or noise.

Therefore, we propose one plain method and two different sophisticated methods to compute existence of AG from a sequence of fixation guided recognition results (respective labels). Figure 4.8 shows the differences between individual methods.

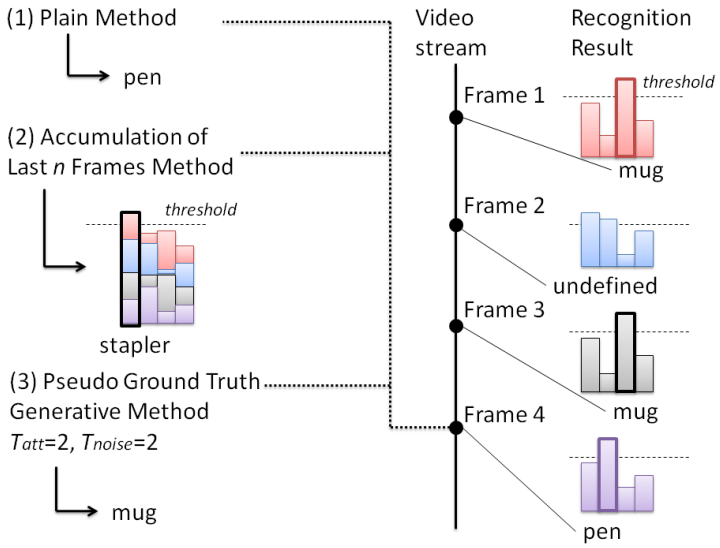


Figure 4.8: Comparison between three methods. Though the results from fixation guided object recognition are same, output of each detection method differs.

1. **Plain Method**: This method directly returns the latest result from the fixation guided object recognition. This method actually does not detect AG and therefore this method is considered as the baseline method.

2. **Accumulation of Last n Frames Method**: In this method, we directly accumulate the normalized histograms of best matches of SIFT features from each frame. The result is returned as the identity of the object that has the highest value in the accumulated histogram, but only if it exceeds a given threshold (otherwise as "undefined").

3. **Pseudo Ground Truth Generative Method**: In this method, the same process that is used to post-process manually labeled ground truth data is applied to the recognition results. The algorithm counts the number of frames that have the same label of object X. When the number F_X exceeds the *duration threshold* T_{dur} before noise F_{noise} exceeds *noise threshold* T_{noise}, AG is returned for the object X.

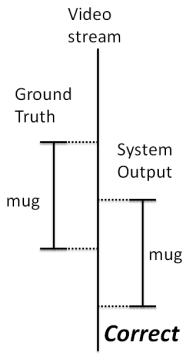
In Figure 4.8, assume we have four objects (a *stapler*, a *pen* a *mug* and a *tin*). When processing the frame number 4, the plain method directly returns the result from the current frame thereby returning *pen* as output. The accumulated histogram in (2) returns *stapler* as the result of thresholding. The pseudo method ($T_{dur} = 2$, $T_{noise} = 2$) returns *mug* since the count of this object reaches 2 in one frame before.

Evaluation Methodology In Museum Guide 2.0, once the user's AG is detected by one of these methods, the system starts to present AR meta information. The presentation of AR meta-information is not stopped unless new gaze on another object X' (with $X' \neq X$) is detected, i.e., as long as these methods return the either the name of the same object X or "undefined", the presentation of AR meta-information remains active.

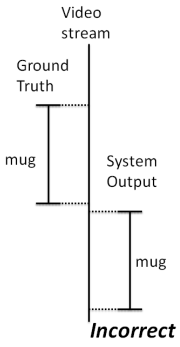
We evaluate the system by comparing the output from each of these methods with attentional gaze-based ground truth that was obtained from manual labels on each video stream by our processing method. Processed ground truth represents a time interval, in which the user attends to a specific object. Therefore, as shown in Figure 4.9a, if there is a chronological overlap between the detected AG and the ground truth, it is considered as a correct output. On the other hand,

- if there is no overlap as shown in Figure 4.9b,
- the name of object is not same as shown in Figure 4.9c, or
- the AG is already detected correctly as shown in Figure 4.9d,

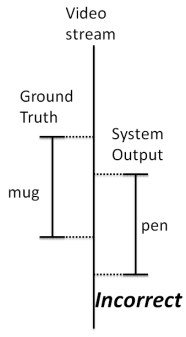
these outputs are considered as incorrect.



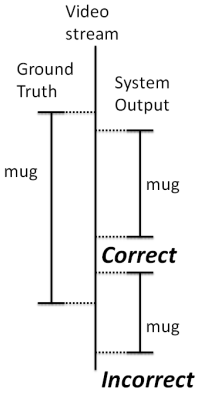
(a) Example of a AG that is correctly detected by the system.



(b) Example of incorrect system output: The system output is too late.



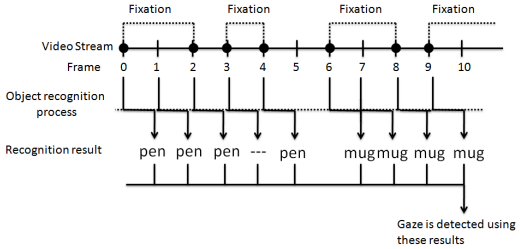
(c) Example of incorrect system output: The name of object is incorrect.



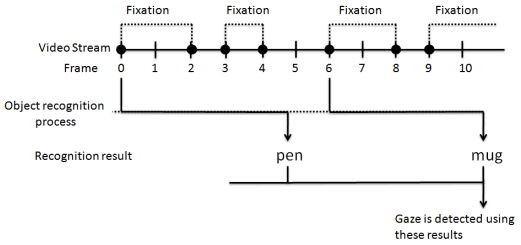
(d) Example of incorrect system output: Multiple AG events are detected, instead of a single one.

Figure 4.9: Examples of correct and incorrect system output.

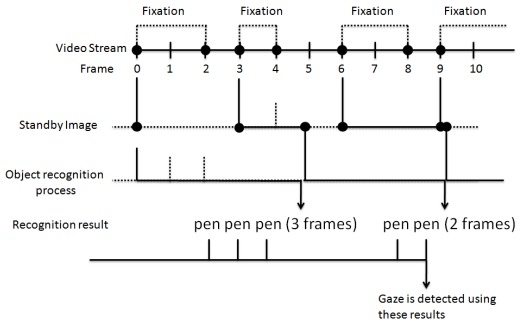
Compensational Approach for Real-time Processing Our intended application is characterized by strong real-time requirements: The user wants to get AR presentations right at the time that he attends to the object. Ideally, the process of object recognition is required to be fast enough, so that the entire process catches up the real-time frame rate as shown in Figure 4.10a. However, the process of a given query-image (local fixated region in the image) by the SIFT based retrieval system takes too long to process all frames (at 25 frames per second) that are delivered by the eye tracker. Consequently, as shown in Figure 4.10b, not all fixation events can be processed and this system cannot detect AG without further ado.



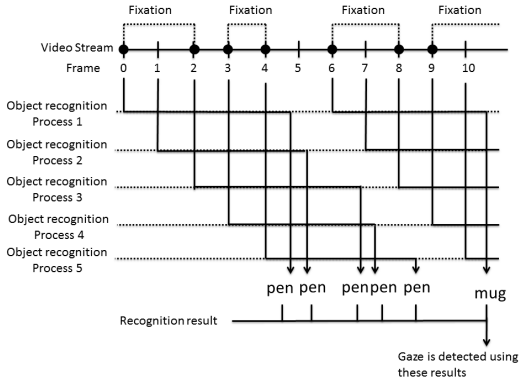
(a) Ideal case. The process for recognition is fast enough to process the data on each frame.



(b) Actual case. The process takes longer than a single frame time interval so that it cannot catch up the frame rate in real-time.



(c) Proposed compensation approach. Keeping a standby image and counting the number of frames having the same fixation within the process to catch up the real-time environment.



(d) Another possible approach only when we can use a lot of CPU cores. Each object recognition process is done by each core.

Figure 4.10: The advantage of compensation approaches for achieving better performance in real-time.

To resolve this problem, we propose a *compensation approach for real-time processing*. In this approach, we prepare a standby image to catch up the real-time environment and to minimize the loss of information. As shown in Figure 4.10b, a standby image is stored at the interface between the video stream of the eye tracker and the object recognition processor. When a new fixation is detected, the corresponding image is stored as the current standby image. If the object recognition processor becomes idle after the recognition process, the

image is piped to the recognizer to start over the recognition immediately. Simultaneously, the number of frames having the same fixation is counted up and the recognition result is multiplied by it when the recognition process on that fixation is ended. This way, the recognition processor of our system is kept busy as much as possible and thus produces as many labels for fixated images as possible, while at the other hand the system always analyzes the newest fixation image. Thus, if AG is recognized and AR is presented, it is based on the newest possible data.

There is also another possible approach that uses multiple CPU cores as shown in Figure 4.10d. Object recognition is processed in individual cores in this approach. However, we discard this approach in this work because it is highly dependent to performance of the hardware.

4.1.5 Experiments

To thoroughly evaluate different aspects of our real-time gaze-based object recognition framework, we conducted a series of experiments. All experiments were performed with the Museum Guide 2.0 use case.

1. Methods for *fixation guided object recognition* were presented. We investigated the performance of each of these methods by comparing two baselines.

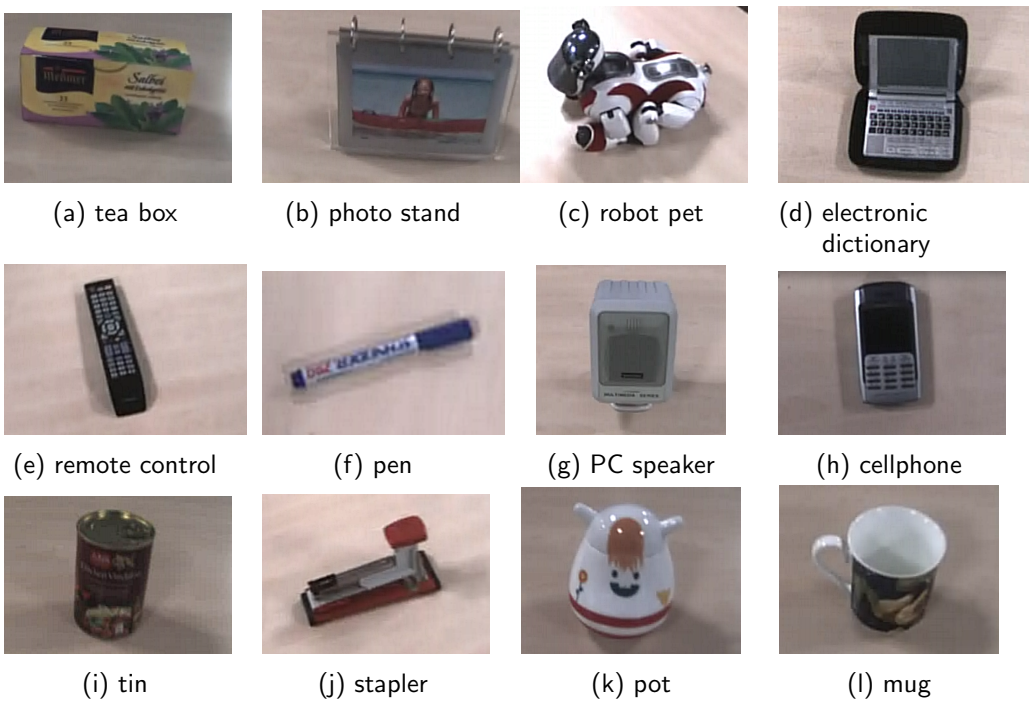
2. We proposed the *attentional gaze-based ground truth processing* algorithm. Thus, we conducted real-world experiments with different users to evaluate suitable threshold values for our processing algorithm.

3. Using the suitable threshold values obtained in experiment (2), we obtain *gaze-based ground truth* which are aimed to be detected by our methods proposed. We evaluated each of the *attentional gaze detection methods based on recognition results* using the evaluation method stated in the chapter. All methods and parameters were optimized for Museum Guide 2.0 based on this evaluation.

4. We evaluated the performance of the system in a real-time environment using the *compensation approach for real-time processing* introduced.

In the experiments (1) to (3), we ignore the constraints of a real-time environment, i.e. there is sufficient time to process for *each frame* as shown in Figure 4.10a that we call *off-line analysis*. The parameters and methods in experiment (4), which is processed in real-time, are optimized based on the results from the *off-line analysis*.

Before conducting the experiments, we designed our museum for the entire experiments. As stated in Section 4.1.2.1, we focused on 3D and small-sized objects. The objects we used were a *tea box*, a *photo stand*, a *robot pet*, an *electronic dictionary*, a *remote control*, a *pen*, a *PC speaker*, a *cellphone*, a *tin*, a *stapler*, a *pot* and a *mug*. The example images of these objects are shown in Figure 4.11. These objects were placed on a long table sparsely and all recordings and experiments were done under the same light setting. In order to test different object placement situations, we had two different object layouts as shown in Figure 4.12.

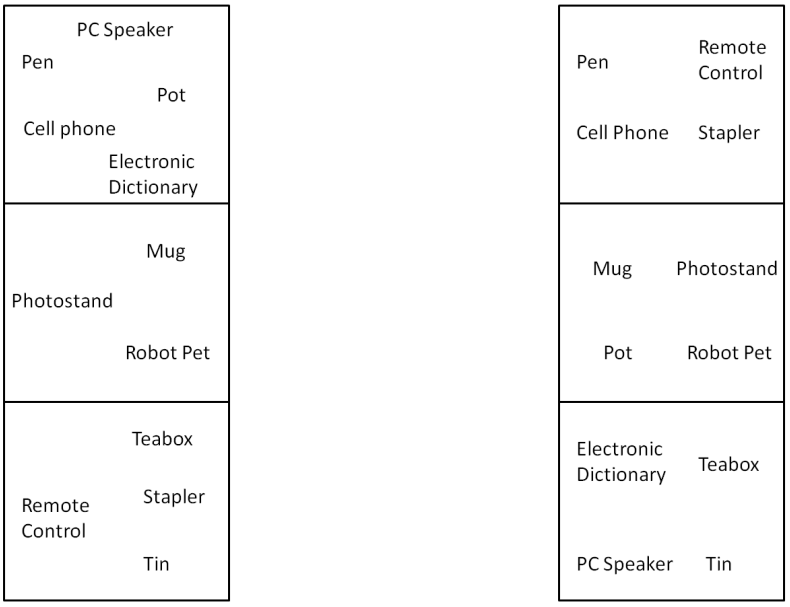


(a) tea box (b) photo stand (c) robot pet (d) electronic dictionary

(e) remote control (f) pen (g) PC speaker (h) cellphone

(i) tin (j) stapler (k) pot (l) mug

Figure 4.11: Example images of objects in the museum.



(a) Layout A (b) Layout B

Figure 4.12: Two different object layouts of the museum.

4.1.5.1 Performance Evaluation of Fixation Guided Object Recognition

First, we investigated the performance of fixation guided object recognition methods by comparing the SIFT method with and without weighting of the eye position and two other SIFT based methods that work on the camera image without concerning the eye position (baseline methods).

The fixation guided object recognition process returns the identity of the object where the fixation exists. Precision P and recall R of this experiment are given by

$$P = \frac{tp}{tp + fp}, R = \frac{tp}{tp + fn},$$

where tp is the number of true positives, fp is the number of false positives, and fn is the number of false negatives. Precision and recall are calculated frame by frame on a per class basis and then averaged over all classes.

To evaluate the benefit obtained by using fixation positions in object recognition, we compared our system with two baseline methods. As a first baseline method, we used an entire image given by the scene camera for the recognition process. If the object of interest occupies major part of the scene image, using the scene image directly for retrieval should already give good retrieval results. As the second baseline method, we cropped the center area of a given image and used that for retrieval. The size of the entire image was 752×480 and the size of the cropped area used both in fixation guided method and the center area method was 280×210. This method would indicate how much performance can be achieved by approximating the center part of the image as the area of interest if we have no gaze information.

Experimental Setup The test video files and gaze data were recorded from six participants while they were strolling in our museum and browsed objects according to their interests. Ten video files were recorded from them (four in layout A and six in layout B). The length of the recorded videos varied from 1 min to 2.5 min. The recorded video files were then labeled manually such that for each frame the identity of the object located at the fixation position was marked. When the fixation position did not correspond to any object in the Museum, the frame was left unlabeled (corresponding to *undefined*). The labeled data was then cross-checked for correctness by another person. In total, the videos contained $27,128$ frames out of which $17,046$ frames were labeled with the identity of the object being gazed at when the frame was captured. All the recorded frames were passed to the object recognition module as query images. Note that the evaluation here is done *frame by frame* such as explained as the primitive way in Section 4.1.4.2.

Preliminary Experiments and Results To evaluate the effect of each parameter (number of SIFT features, ϵ and size of the database), we did several preliminary experiments.

First, the maximum number of SIFT features to be used for retrieval from each image was varied from 10 to 100. Figure 4.13 shows the results with each upper bound value of number of SIFT features. The performance did not drop until the upper bound value decreased to 40 and then the performance started to drop. From this result, 30 to 50 are considered to be suitable values. We chose 50 for the upper bound value of number of SIFT features.

Next, ϵ was varied from 0.0, 2.0 and 5.0 in the validation (refer to Section 4.1.4.1). Table 4.1 shows the obtained recall with each ϵ value and the processing time per image

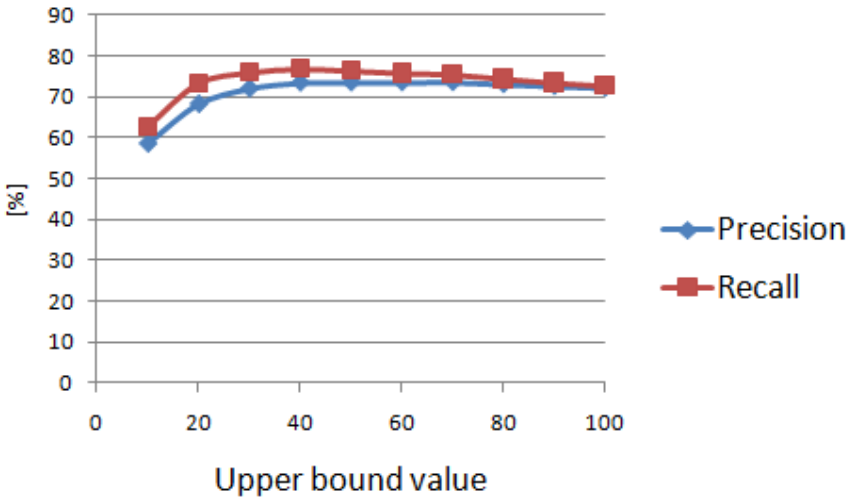


Figure 4.13: Recognition performance with each upper bound value of number of features. The performance dropped under 40.

Table 4.1: Effect of changing values of parameter ϵ of ANN approach on the recognition speed and recall.

	$\epsilon = 0.0$	$\epsilon = 2.0$	$\epsilon = 5.0$
Processing time	546.9 msec	49.9 msec	8.0 msec
Recall	43.10 [%]	42.85 [%]	37.40 [%]

without feature extraction. $\epsilon = 0.0$ was too slow but $\epsilon = 2.0$ with its response time of 49.9 msec was acceptable as one has to consider that processing time for the extraction of SIFT features which took at least another 150.0 msec even if there were only few features (10 or even fewer features). Thus, we used $\epsilon = 2.0$ for the entire experiments.

Figure 4.14 shows the results from four different sizes of database. Here we used 8408 features as the smallest database and the others are approximately double, quadruple and octuple size of the smallest one. Processing time of queuing each feature grew as the size of database was increased. However, even with the smallest size of database it achieved competitive recognition performance to other large-sized database.

Results of Fixation Guided Object Recognition Figure 4.15 shows the Precision-Recall curves obtained by changing output thresholds from 0.5 to 0.9 for the highest value in the histograms computed by object recognition processes. In the *entire image*, an entire image from the scene camera is used for each recognition process. Also, the center area of a given image is used in the *center area*. These two methods were used as the baseline methods. The fixation guided object recognition methods (non-weighting and SIFT weighting methods) completely outperformed the baseline methods indicating that eye position indeed helps in improving the object recognition system. Though the SIFT weighting method scored slightly higher recall than non-weighting method on the same threshold, the overall performance

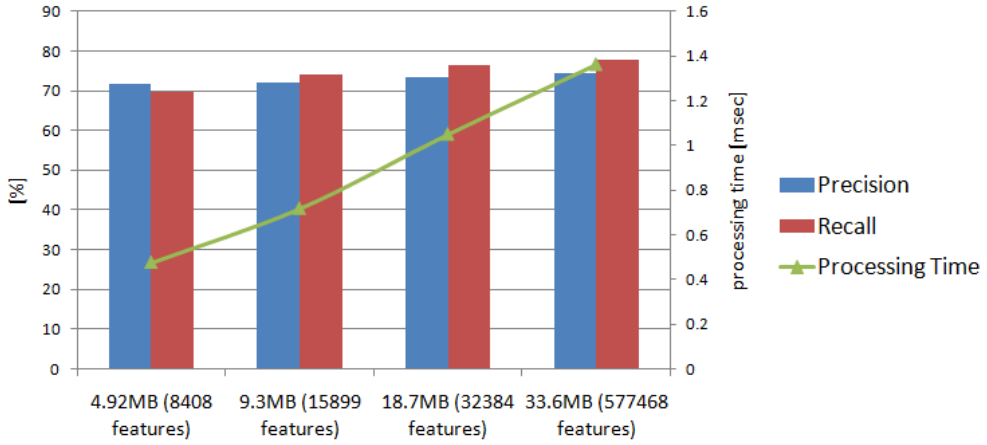


Figure 4.14: Recognition performance with each database size. Processing time is the average time of queuing each feature. The smallest size of database achieved competitive performance to others.

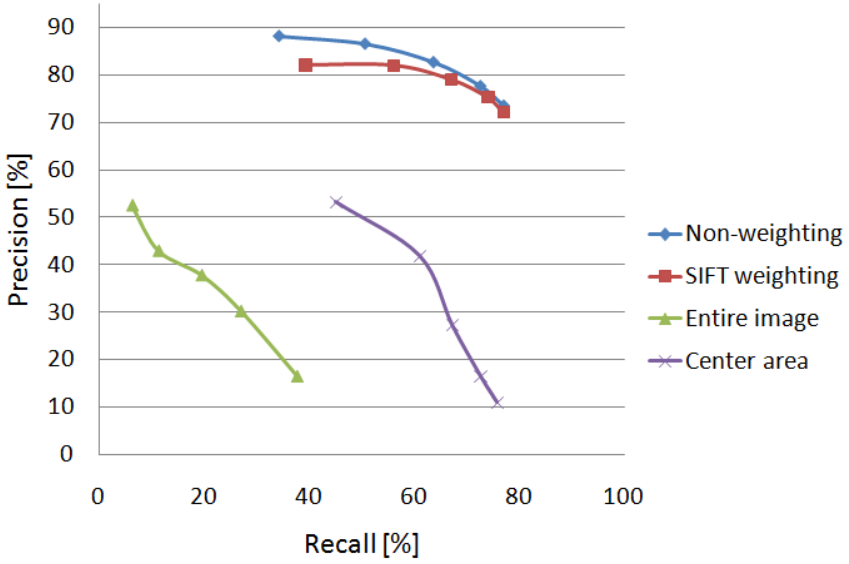


Figure 4.15: Results of fixation guided object recognition, showing that methods using fixation information (non-weighting and SIFT weighting) clearly outperformed simple object recognition methods (Entire image, Center area) that did not use any eye tracking information.

of this was inferior. Since this method weights the features close to the gaze point more, mistakes of matching of these features are critical. Therefore, in the case that features nearby the fixation point are matched to an irrelevant object, the precision is decreased. On the other hand, if only features nearby the fixation point are correctly matched, the system returns a correct answer even if further features are matched to irrelevant objects.

Next, we investigate the recognition performance on each object. Here we only show the results from the SIFT-weighting method with threshold 0.5. Figure 4.16 shows the system outputs for each object identity. The length of the graph where the system output and the ground truth is same represents precision on each object in this figure. Objects that have complicated texture such as *tea box* or *electronic dictionary* (as shown in Figure 4.11) achieved higher precision compared to *PC speaker* or *cell phone*. Sometimes *tin* was recognized as *robot pet*.

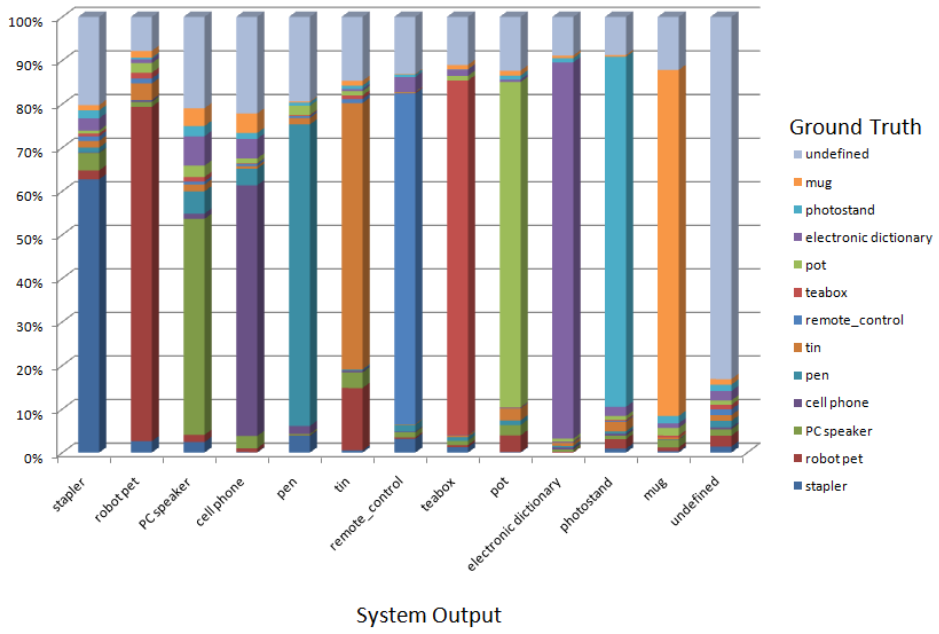


Figure 4.16: Ground truth label rates for each system output. We can observe precision of each object identity from these graphs. *Teabox* or *electronic dictionary* achieved higher precision than *PC speaker* or *cell phone*. Sometimes *tin* was recognized as *robot pet*.

Figure 4.17 shows the recognition results for each ground truth label. Similar to the previous result, recall of each object is observed from these graphs. The recall score of *PC speaker* as well as the precision score was the lowest. However, the recall score of *cell phone* was not as low as the precision score.

4.1.5.2 Validation of Gaze-based Ground Truth Processing

In this experiment we aimed to find suitable threshold values for our attentional gaze-based ground truth processing by analyzing video and gaze data.

Experimental Setup In this analysis, five objects (*a tin, a pen, a cellphone, a PC speaker* and *a tea box*) were placed on a table and we asked the participants to provide spoken feedback (e.g. *"Now, I am watching a pen."*) when they were watching objects *consciously*. These explicit verbal feedback represent the ideals of attentional gaze-based ground truth. Six test persons took part in this experiment and they were asked to watch objects at least 20 times in total. In contrast, we also asked the participants to act as if they browse around in

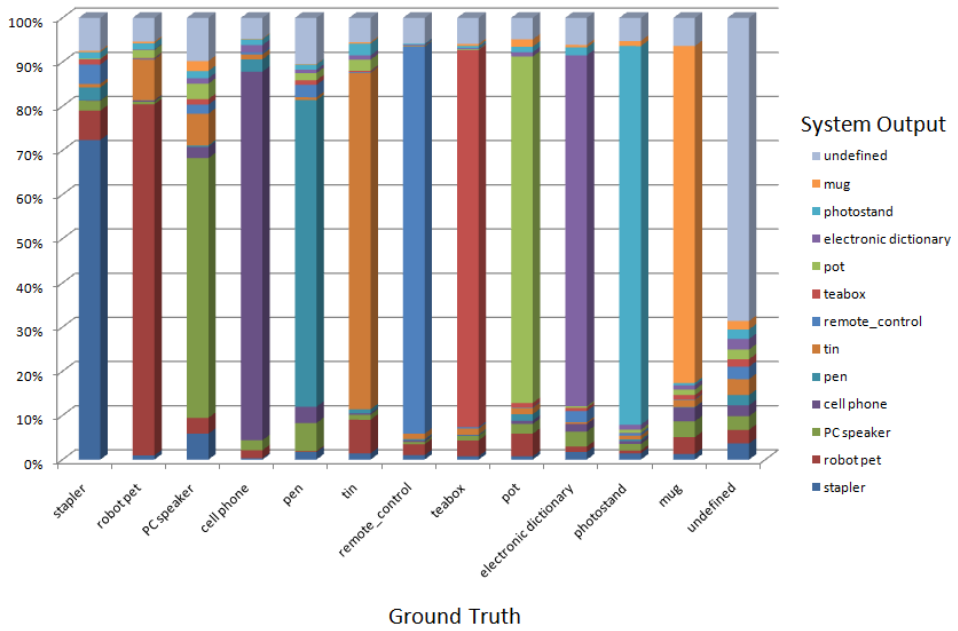


Figure 4.17: System output rates for each object ground truth label. We can observe recall of each object from these graphs. *PC speaker* had the lowest recall score in similar to precision. However, the recall score of *cell phone* was not as low as the precision score.

a real museum, where some objects are only watched for short time but also with no real consciousness. We evaluated with which threshold values the algorithm obtains the best overlapping result with respect to the spoken ground truth.

Results and Analysis We applied our attentional gaze-based ground truth processing with changing duration threshold value T_{dur} and noise threshold value T_{noise} to all the manually labeled frames. If the number of obtained AG events by a particular combination of threshold values is close to the number of verbal feedback counts, these AG events are considered to be correctly reflecting our attentional behavior. Thus, we compared the number of AG events obtained by the algorithm to the number of the verbal feedback counts from the test persons.

We would like to investigate general tendency rather than variation between individuals. Therefore, we average the number of the obtained AG events for each test person. Figure 4.18 shows the average number of obtained AG events by changing T_{dur} for $T_{noise} = 3$, $T_{noise} = 18$, and $T_{noise} = 30$, respectively. The average number of verbal feedback counts for all the test persons is also shown in the figure as the horizontal dotted line, with a value of 24.5 (All test persons gave the verbal feedback more than 20 times).

While T_{noise} is low (noise threshold: 3), the slope on each point in the graph is steep. Then, as T_{noise} becomes larger (noise threshold: 18 or 30), the graph reaches an almost flat shape between respective thresholds around T_{dur} 14 and 23. Since the number of obtained AG events on the flat area is close to the average number of verbal feedback counts, the T_{dur} values in that range are considered as candidates for the optimal T_{dur} with respect to the ground truth.

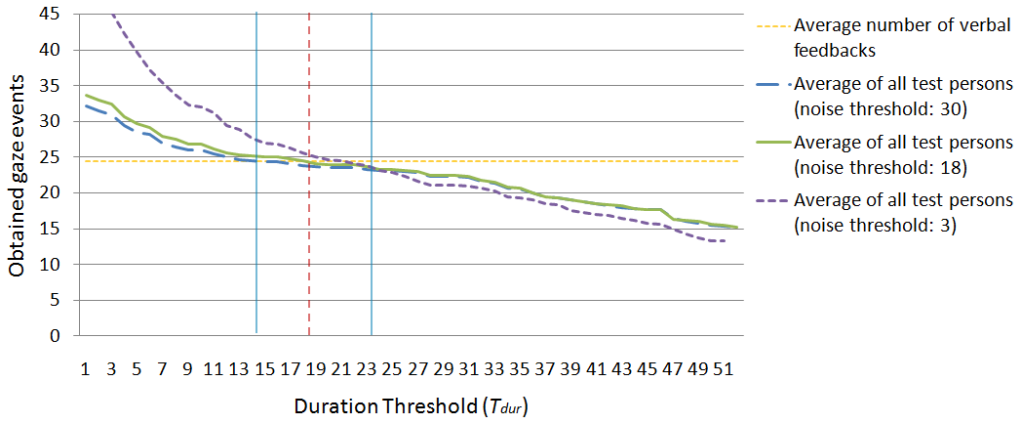


Figure 4.18: Number of AG events obtained by the algorithm with changing T_{dur} (frames). The dotted vertical line is drawn on duration threshold 18. In the area between two vertical solid lines (on 14 and 23, respectively), the graphs (noise threshold: 18 and 30) reach almost flat shapes.

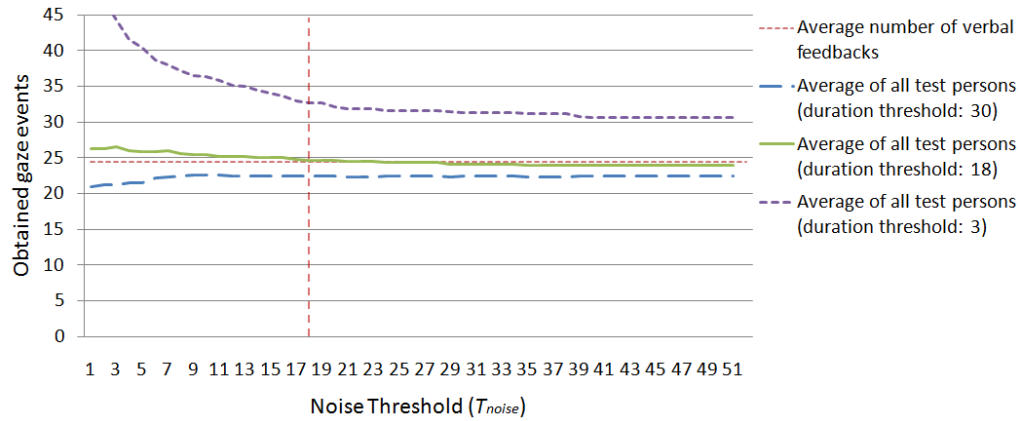


Figure 4.19: Number of AG events obtained by the algorithm with changing T_{noise} (frames). The dotted vertical line is drawn on duration threshold 18. All graphs converge as T_{noise} increases.

Figure 4.19 shows the average number of obtained AG events with changing T_{noise} for $T_{dur} = 3$, $T_{dur} = 18$ and $T_{dur} = 30$, respectively. All the graphs converge as T_{noise} increases. From these graphs, we can infer that 18 is a reasonable value for T_{noise} as the number of obtained gaze events remain constant from this point on.

From these observations, we can conclude that the algorithm reliably obtains quite similar results to the spoken ground truth with the proper setting of threshold values. We select 18 (approx. 0.7 sec) as the optimal threshold values for both noise and duration in a general case because this combination reflected verbal feedback well within this experimental framework.

4.1.5.3 Evaluation of Methods for Detection of AG

In the previous subsection, we confirmed that the attentional gaze-based ground truth obtained by our algorithm reasonably reflects the verbally expressed consciousness. By using the ground truth obtained by this algorithm, we evaluated the methods for detecting of AG (*plain method*, *accumulation of last* n *frames method*, and *pseudo ground truth generative method*, refer back to Section 4.1.4.2) using *SIFT weighting* and *non-weighting* fixation guided object recognition method.

Experimental Setup This experiment is processed off-line so the processing time is not considered critical here. We use the same data as the experiment of fixation guided object recognition, i.e., ten video files with gaze data. By applying the ground truth processing algorithm to the data recorded in Section 4.1.5.1, 183 ground truth AG events were generated in total. We compared the obtained ground truth and computationally detected AG by the system on particular objects as described in Section 4.1.4.2. To evaluate the methods, we use recall R and precision P again. Since each AG has a label (the identity of the object being gazed at), evaluation is done on a per class basis and then averaged over all classes.

Results and Evaluations Figure 4.20 shows the Precision-Recall curves of the *plain method* which directly outputs the results from the fixation guided object recognition process unless they are "undefined". These graphs are drawn by changing the threshold for the output from

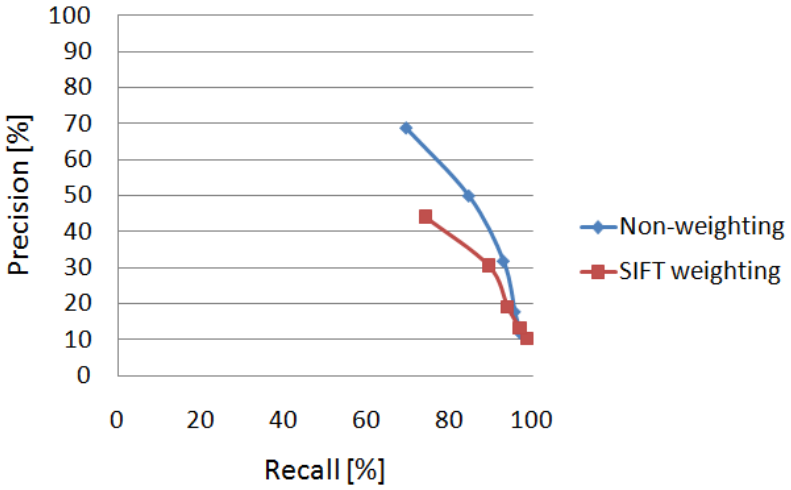


Figure 4.20: Results of the plain method. For the same thresholds, the recall of the SIFT weighting method was higher while the precision was lower than the non-weighting method.

0.5 to 0.9. As one can see in this figure, though the plain method could score high recall, the precision scores were low. In this method, the *non-weighting* method outperformed the *SIFT weighting* method.

Second, the results of the *accumulation of last n frames method* are shown in Figure 4.21 with the threshold value for the output changing from 0.5 to 0.9. Here, we compared $n = 18$ and $n = 30$ with non-weighting, and $n = 18$ with SIFT weighting. The reason the method

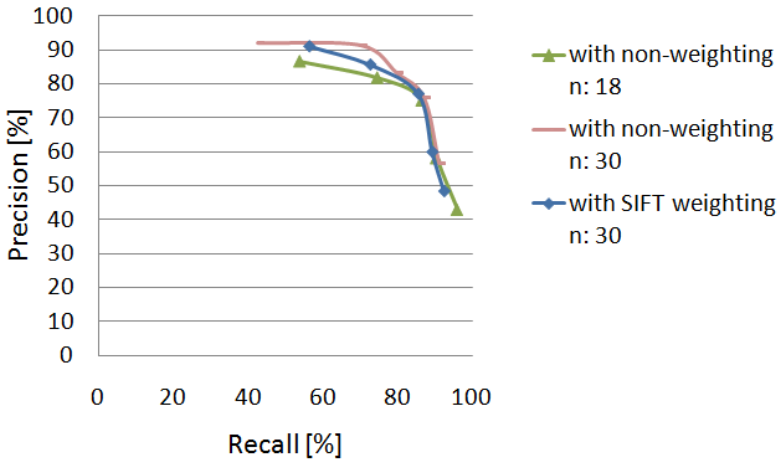


Figure 4.21: Results of the accumulation method. The performance with each parameter was similar.

using $n = 30$ with SIFT-weighting is not shown here is that the method could not outperform others. Generally, as n increases, it becomes harder for each identity of object to obtain the value that exceeds the threshold value. This is the reason the method with $n = 30$ scored higher precision than the method with $n = 18$ on the same threshold value. However, the overall performances of them were similar.

Third, Figure 4.22 shows the results of *pseudo ground truth generative method*. This method requires the threshold parameters for T_{dur} and T_{noise} individually. The graphs with $T_{dur} = 18$ and values 10 and 18 for T_{noise} and with $T_{dur} = 22$ and $T_{noise} = 14$ are shown in this figure. Only the results for the SIFT weighting method are shown here because this method worked better than the other one in this experiment. The reason of this is that this

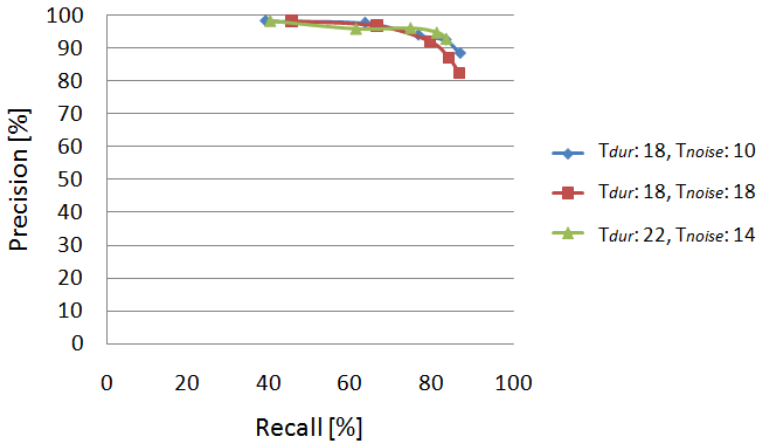


Figure 4.22: Results of the pseudo method, with three different combinations of T_{dur} and T_{noise}.

pseudo method requires higher number of possible answers rather than their preciseness, i.e. as a raw output of fixation guided object recognition, recall is more important.

Even with the same T_{dur}, the results differ when T_{noise} changes. When T_{noise} is large, the incorrect result from the fixation guided object recognition process remains as AG that is possibly detected for a long time. Thus, when T_{noise} is large, precision decreases. Similarly, a larger T_{dur} makes the recognition results more precise.

Finally, we compare all the methods in Figure 4.23. One can observe that the two proposed sophisticated methods outperformed the plain method. The accumulation method worked well compared to the plain method, however it was inferior to the pseudo method. The reason for this was that this method highly depended on the features from each frame. For example, even if the number of frame that captured object X was only one and the other frames in n frames had fewer features than it, the features from one frame (features from X) affected the entire recognition process and therefore this method returned X as an AG event even though this was not considered as AG.

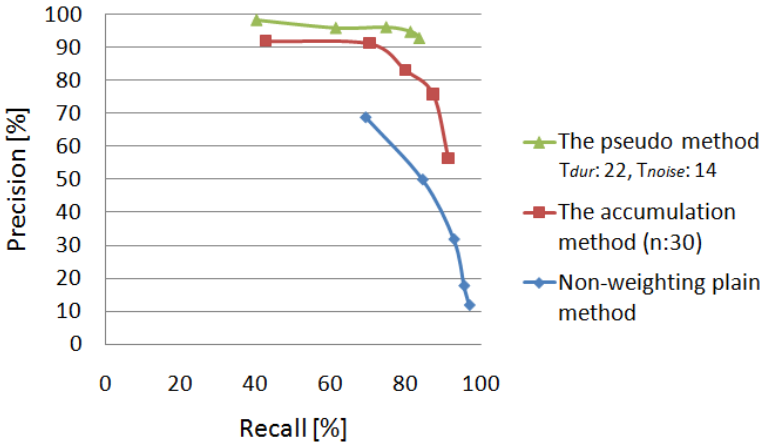


Figure 4.23: The best results of each AG detection method. The pseudo method outperformed other two methods.

Based on these experiments, we selected *pseudo ground truth generative method* and *SIFT weighting method* as our attentional gaze-based object recognition system for Museum Guide 2.0. And the parameters were set to $T_{dur} = 22$ and $T_{noise} = 14$ (for the recognition method).

4.1.5.4 Evaluation of the approach for real-time processing

Next, we evaluate our *compensation approach for real-time processing* introduced in Section 4.1.4.2. In this experiment, we used the same video and gaze data as in the previous experiments but sending 25 frames per second (the same frame rate as the scene camera of the eye tracker) to the attentional gaze-based object recognition system optimized in the previous subsection to simulate a real-time environment.

Figure 4.24 shows the results obtained by the method with and without the compensation approach and the result from the off-line experiment (presented in the previous section). The threshold values (T_{dur} and T_{noise}) for AG detection method with the compensation approach

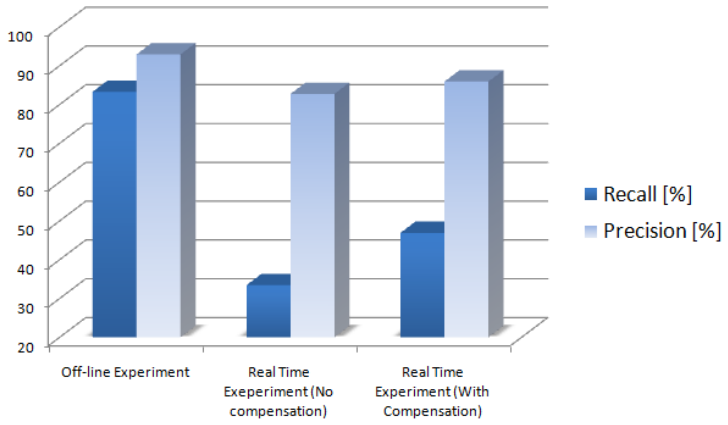


Figure 4.24: Results of real-time simulation. Although recall drops significantly, precision still remains at an acceptable level.

were same as in the off-line experiment. Thus, we needed to use different values for the no compensation approach. Due to its long processing time for object recognition, the method optimized in the last subsection could not detect any AG in the no compensation approach. As shown in Figure 4.10b, the optimized T_{dur} or T_{noise} for the off-line system were too large as the recognition system cannot catch up the real-time speed. Therefore, we dropped $T_{dur} = 6$ and $T_{noise} = 4$ for the no compensation approach to obtain a similar precision score to the other approach. One can observe an enormous drop of recall as compared to the results in the off-line experiments. However, compared to the no compensation approach, the method with compensation approach worked significantly better.

4.1.6 Conclusion

In this section, I presented a method for recognition of user-attended object and experimental results, focusing on a practical real world scenario called Museum Guide 2.0. First, it was shown that object recognition can successfully be improved by using fixation points, when we have the user's egocentric view. Then, by detecting users' AG on particular objects, the system could reasonably infer whether the user certainly attends to the objects. Furthermore, it was also shown that the compensation approach could improve the recognition performance when a real-time processing is required. In Chapter 8, I present a user study of Museum Guide 2.0, questioning participants the benefits of Museum Guide 2.0.

One of the drawbacks of this system is that it can only recognize known objects which are registered in the database previously. Therefore, the system cannot recognize the user attentional gaze if the user attends to unknown objects. In the next section, I discuss how one can detect attentional gaze on an unknown (arbitrary) objects.

4.2 Attention Detection on Arbitrary Objects

In this section, I present an attention detection method when objects are unknown to the system. Using an IMU, we compute the user's gaze direction in the 3D space. Consequently, we generate a so-called *heat-map* which depicts intensively attended regions in the environment. Such regions show that the user attends to particular subsets of content in a scene.

4.2.1 Introduction

In the previous section, we learned how the user's AG on a specific object can be detected in order to provide an automatic audio guidance of the exhibits in a museum. The application showed potential of gaze-based human-computer interaction, particularly by inferring user attention from gaze.

However, the system presented previously relies on image processing-based object recognition mechanisms in order to detect to which object in the scene the user is attending. These types of object recognition-based approaches hold two crucial drawbacks. First, a known set of objects is always required to recognize the objects. Most of object recognition-based systems need a pre-defined database in order to match local features extracted from an image (such as SIFT or SURF). Thus, these systems cannot deal with unknown objects which would be present more frequently than known one in everyday environment. Secondly, even though advances of the recent hardware relax the restriction of computational expense, image processing still requires high computational cost, particularly for object recognition with a large database. Thus, it has high latency in a real-time scenario when we have a huge number of object variations.

This section presents a method to capture the user's AG on particular objects or regions without using object recognition methods. Instead, we analyze the number of fixations in a particular region in a scene, by combining eye tracking data with head motion data extracted from physical sensors such as an accelerometer, a gyroscope or a magnetic compass. Normally, a mobile eye tracker provides with the gaze position as a coordinate in an individual scene image. In this work, motion sensor data is used to calculate relative gaze positions between consecutive video frames. By computing the spatial orientation of each gaze sample and aggregating this data in a *global gaze heat-map*, we find intensively fixated regions, which are considered as AG on particular (an) object(s). This way the method can detect AG on arbitrary objects without recognizing the individual objects.

Additionally, the motion data is also used to recognize the user's physical activity. This enables the system to treat the gaze mapping differently according to the activity. Here, we only focus on two types of activities: *walking* and *stationary*[6]. A *global gaze heat-map* is

[6] *stationary* includes the activities that the person is sitting and standing still.

created only when the user is in a *stationary* state because accurate relative gaze positions are hardly available when the user is walking.

In the experiment, we compare this method with the method proposed in the previous section to show this method can reasonably capture AG compared to the object recognition based method while it is also applicable to other arbitrary objects. This method cannot name "what" is being attended by the user; however, it can be used to infer to which type of daily content the user attends. We also show that we may use an online image retrieval engine in order to name the content automatically. Such a system can used to log everyday attentional data as *visual diary*.

4.2.2 Apparatus

As mentioned in the introduction, we compute the user's gaze in the 3D space by combining eye tracking data with physical motion data such as acceleration and rotation. In order to capture the motion of the user, we use an IMU. Figure 4.25 shows pictures of the ETG with a 9 degrees of freedom (9DoF) IMU. We attached the IMU on one side of ETG frame using

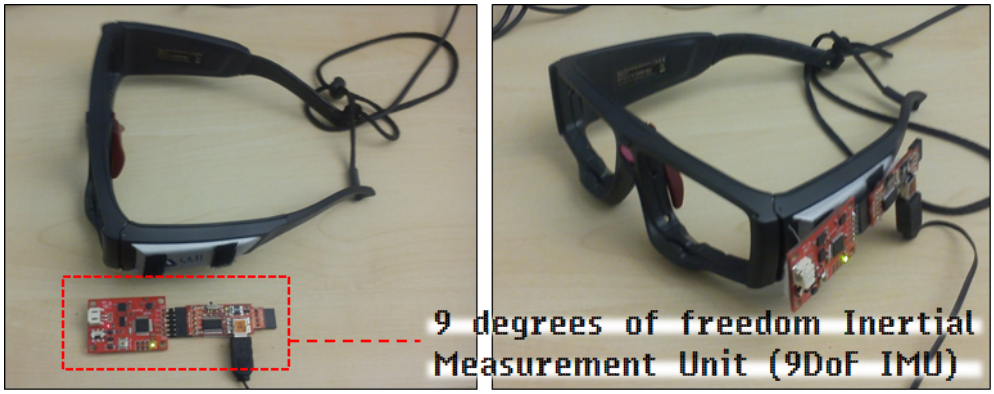


Figure 4.25: ETG with a 9 DoF IMU. The IMU is attached to the frame by Velcro fasteners.

Velcro fasteners as shown in the figure. From a 9DoF IMU, we obtain 9 independent motion values, i.e., three dimensional acceleration values, three dimensional gyroscopic values, and three dimensional magnetic values. These values are continuously sent to the computer in real-time with a USB cable. Then, raw values are processed in the computer. Since we also obtain data from the eye tracker simultaneously in real-time, we can easily synchronize the data.

Since the IMU is light-weight, it can be fixed to the ETG with the Velcro fasteners very firmly. Thus, the motion data (especially accelerations) can be captured reliably[7].

4.2.3 Method

For detection of AG on arbitrary objects, we combine motion features from the IMU and gaze data form the eye tracker. A *global gaze heat-map* is generated using the data. The

[7] A low-pass filter can be applied to remove noisy (pulse) data.

heat-map is used to predict the regions or the objects in a scene that are intensively attended by the user.

4.2.3.1 Head Movement Tracking Using an IMU

Using the IMU, we extract the acceleration and orientation vector of head movements. The raw data is preprocessed in order to capture the acceleration vector $\boldsymbol{A} = (A_x, A_y, A_z)$ and orientation vector $\boldsymbol{O} = (O_{Pitch}, O_{Roll}, O_{Azimuth})$ of current motion. The acceleration vector consists of three values: A_x, A_y, and A_Z, where each of them corresponds to the acceleration value of axis x, y and z, respectively. Also, the orientation vector consists of three values: O_{Pitch}, O_{Roll}, and $O_{Azimuth}$, where each corresponds to Azimuth, Pitch and Roll as shown in Figure 4.26. These two vectors show at which direction the user's head

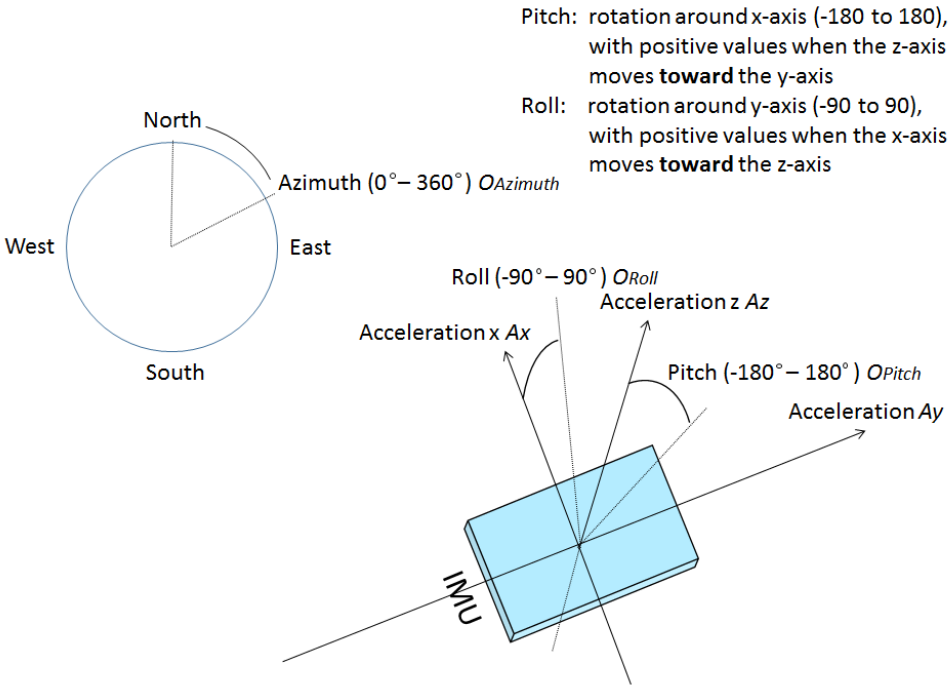


Figure 4.26: Motion values measured by the IMU.

aims and how strongly the movement is accelerated. If the acceleration vector is perfectly accurate and data sampling rate is fast enough, we might as well map the absolute spatial position of the user in the environment so that the system can reconstruct the 3D map of the user position and his or her eye movements. However, the technology available today is not advanced enough to reconstruct such a map only by using accelerometers and gyroscopes. Thus, instead, we adopt an alternative approach based on the following observations.

- Eye movement patterns change when the user switches his or her activity state from stationary to walking.
- If the user position remains in the same, the direction of the user's head measured by the orientation vector has the same origin in the real space.

We analyze the user gaze data in the 3D space, only when the relative direction of gaze is reliably available. Thus, when "walking" action is detected, the system resets the map. Only when the user's activity state is "stationary", it aggregates all gaze samples in a map to generate a *global gaze map* as shown in Figure 4.27 (The map generation method is presented in Section 4.2.3.3). We analyze the user gaze and attention using this map. There is also another insight here; people often stop and stay stationary when they want to attend to something.

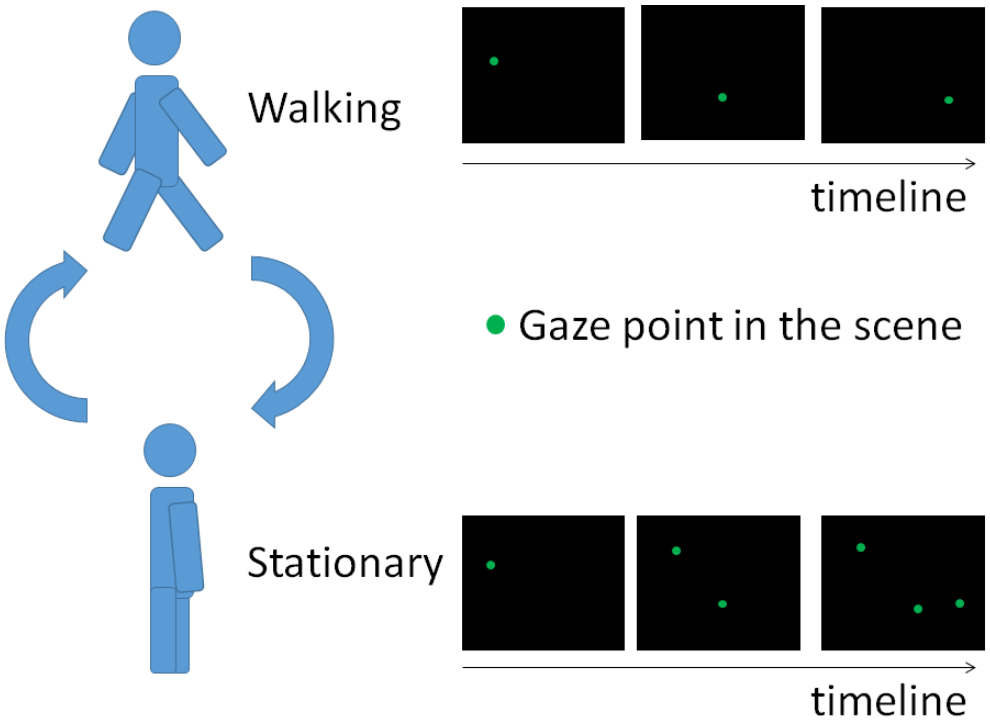


Figure 4.27: State transition between two activities. When walking activity is detected, the system resets all gaze position in the scene. Otherwise, it aggregates all gaze samples in a single *global map*.

4.2.3.2 Activity Recognition

As stated in the previous subsection, this method distinguishes a walking and stationary state in order to switch the processing mechanism of gaze samples. To recognize these two different states, we use the acceleration vectors obtained from the IMU. Although a number of approaches have been proposed for activity recognition to date [BI04], these approaches were intended to be applied to recognition of multiple activities (typically more than 8 different activities). Therefore, they adopt relatively complex features and classifiers. Since our intended activities are only two simple ones, we only use the mean acceleration value of the signals over a period, which is also used as one of the features in [BI04]. By thresholding the mean value, we classify the user activities into walking or stationary. The threshold value is obtained from the mean value of the average value of the training samples from each activity.

4.2.3.3 Gaze Vector Computation in a Scene

While a *stationary* activity state is being recognized, the system maps all the gaze samples from the eye tracker onto a two-dimensional plane. The plane represents a *global gaze map* where the user's head is centered as shown in Figure 4.28. The two axes are introduced as

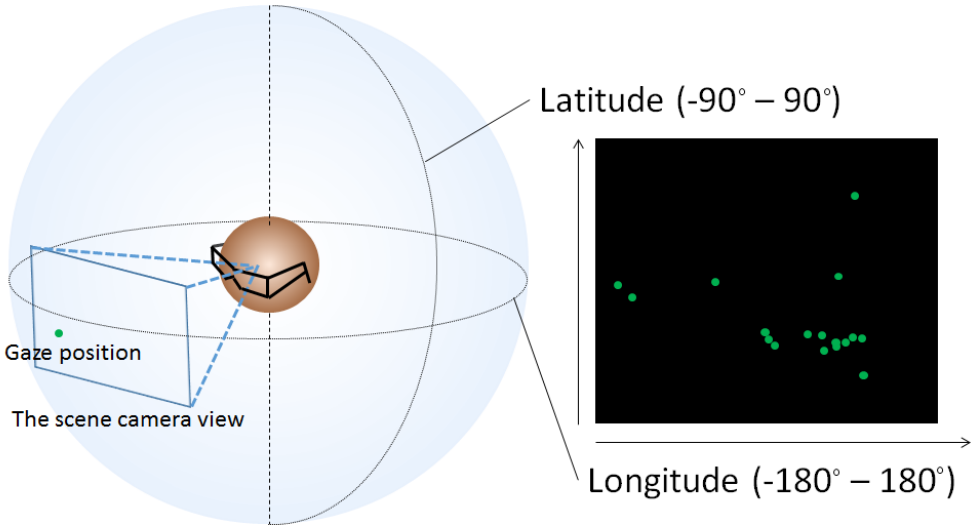


Figure 4.28: Global gaze map. The globe represents a sphere where the user's head positions the center. Each green dot represent each sample of gaze.

longitude and latitude, which range from $-180°$ to $180°$ and from $-90°$ to $90°$, respectively. In order to compute a gaze position on this map, the gaze vector and the head orientation vector are added as shown in Figure 4.29. The gaze coordinate in a scene image $\boldsymbol{g} = (g_x, g_y)$

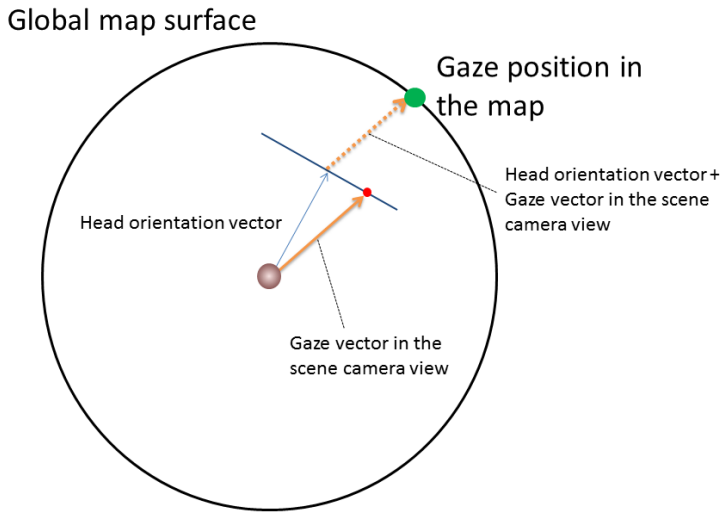


Figure 4.29: Gaze vector in the 3D space is computed as the sum of two vectors.

is transformed into the 3D space $\boldsymbol{G} = (G_x, G_y, G_z)$, assuming that the vector has a unit length and given the field of view angle of the scene camera ($60°$ horizontal and $40°$ vertical, refer back to Section 3.1.5). Hence, a gaze position on a global map $\boldsymbol{g_M}$ is given by,

$$\boldsymbol{g_M} = (G_x + O_{Azimuth}, G_y + O_{Pitch}).$$

Here, we ignore O_{Roll}, which corresponds to a movement orientation that changes when the user tilts his or her head. In this work, we assume that the user keeps his or her head straight up for simplicity. All gaze samples in a standing activity state are aggregated in one map to observe how the user views the scene, concerning the user's head direction. We analyze this map with respect to user attention by generating a *heat-map*.

4.2.3.4 Heat-Map Generation and Attentional Region Detection

We detect user attention by counting the number of gaze samples located nearby in the global map. For detection of such attentional gaze (AG) regions, each axis of the global map is divided into n_{lat} and n_{lon} respectively as shown in Figure 4.30. When a cell region

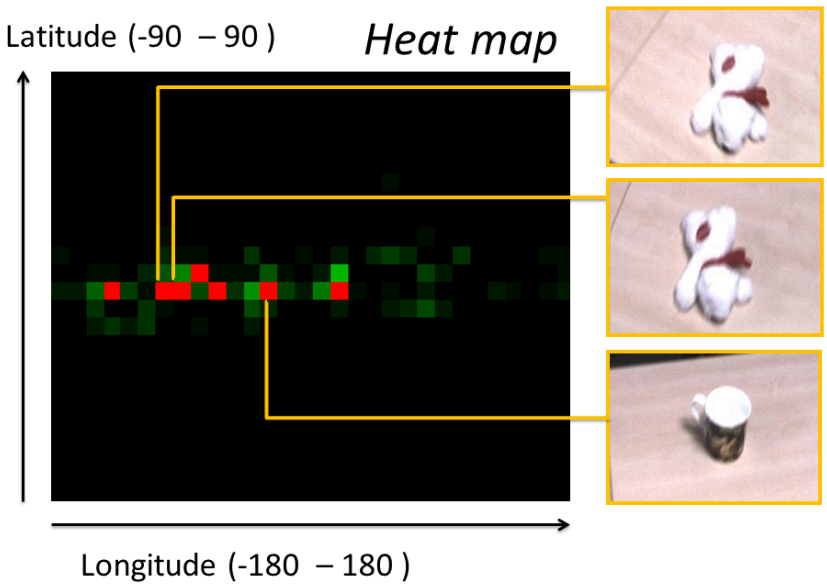


Figure 4.30: Heat-map of user gaze and images from respective cells. Each axis is divided into n_{lat} and n_{lon}. Therefore, the map has $n_{lat} \times n_{lon}$ cells. In this example, $n_{lat} = 25$ and $n_{lon} = 30$. The contiguous cells capture the same object (a toy bear). Red cells are detected AG regions.

$R(n_{lat}, n_{lon})$ has more than T_N samples, the system outputs the region as an AG region. The color intensity of a cell represents the number of gaze samples located in the cell; the greener the cell is, the more gaze samples are located in the cell. If AG is detected, the color of the cell turns into red. Also, if the detected AG cell is contiguous to any cell in which already AG exists, these cells are considered to be covering the same object (e.g., a toy bear in the figure). Therefore, if that is the case, the system assumes two contiguous AG cells as one AG event.

Since the proposed method employs a simple and computationally light-weight algorithm, the entire process can be done within approx. $70 \sim 80$ msec, while object recognition-based methods may miss important gaze information due to complicated processes. In the experiment, we evaluate the performance by comparing the proposed method with an object recognition-based method.

4.2.3.5 Image Labeling

Once the AG on a particular region is detected, the system saves the local image of the AG region. It crops the local region of the image centered around the detected AG position. The saved images contain an object or content that drew the user attention. Labeling of the image can be done either manually by a user or automatically by an image retrieval engine. Recent online image search engines such as CamFind[8] provide with a very reliable *reverse image search* function which returns a label for a query image. When one needs to get a label of an AG image, he or she could also use these online APIs for image search. Note that we only use the image retrieval systems in order to get the label of the image but the detection of attention is done without using any image retrieval (object recognition) methods.

Another advantage of motion tracking is that we could also predict ego-motion blur in an image. It is important to assess a quality of an image when we apply image search or object recognition since we may not get good results from poorly captured images. Based on acceleration values, we can filter out images that may have severe motion blur, which often degrades image qualities.

In Chapter 8, I present an application called *Visual Diary*, which uses this arbitrary AG detection method for collecting daily attentional events. In the application, we combine own object recognition framework presented in Section 4.1 and one of the reverse image search engines for acquiring labels for attended content.

4.2.4 Experiments and Evaluation

In order to evaluate how well the proposed method detects the user's AG, we compare the proposed AG detection method with the object recognition-based attention detection method presented in Section 4.1. We use the same experimental procedure used in the previous section (Section 4.1.5.3).

The point of this experiment is a comparison of AG detection methods. To compare two AG detection methods, we have to define a benchmarking framework. Since the conventional AG detection method only provides with an identity of object as a result, we also have to identify the label for the proposed method to compare on the same basis. Thus, we apply object recognition for the AG regions detected by the proposed method to get object labels. Then we compute the recall and precision rates for both conventional and proposed methods similar to Section 4.1.5.3. Note that even though we apply the same object recognition technique in order to compare the outputs from both methods, object recognition performance itself is not the focus of this experiment.

[8] http://camfindapp.com/

4.2.4.1 Data Acquisition

First, we put ten different objects well spaced-out on a table similar to the setup in Figure 4.12. Then, we asked ten test persons to wear the eye tracker and to browse the objects naturally. In other words, we asked them to view objects with a certain attention if any object is interesting or otherwise just to give a glance. After labeling the recorded video frames as the identity of the object being indicated by the user gaze, the ground truth of the user gaze are obtained by using the same method as described in Section 4.1.4.2. The ground truth data consists of the frame number of the beginning of AG, the frame number of ending of AG and the label of an object. The evaluation is done by checking whether the system can detect the AG on the labeled object during the period indicated by the beginning and the ending. The total AG events obtained in this experiment were 72.

Similar to Section 4.1.5.4, we simulate a real-time environment. All the experiments are done by sending video frames with the same speed of the scene camera sampling rate to the AG detection system. The sample rate of the scene camera of the eye tracker and the eye tracking was 25 fps. The IMU provides data immediately when a motion is detected. All the experiments were done on an Intel Core i5 M560 2.67GHz CPU with 8GB RAM.

4.2.4.2 The Conventional Method

The conventional method is the method presented in Section 4.1. The conventional method applies an object recognition process to the image when it has a gaze position. Since the eye tracker does not always provide the gaze position due to several reasons, such as blink or a failure of the image processing, only when the gaze position is available, object recognition is executed. Furthermore, if the system is run in a real-time environment, it misses some gaze positions during the processing. Thus, not all frames are necessarily processed even if the frame have the gaze position.

This method counts the number of frames that have the same label of object recognition result. When the number of such frames reaches a threshold value while accepting a certain number of noise frames, the system outputs as the result that the user is attending to the object.

4.2.4.3 Results

In Figure 4.31 and Figure 4.32, I show the precision and recall graphs of the proposed AG detection method. In the figures, the results from $T_N = 6, 8, 10, 12, 14$ with $n_{lon} = 20, n_{lat} = 15$, $n_{lon} = 30, n_{lat} = 25$, and $n_{lon} = 40, n_{lat} = 30$ are shown.

First, Figure 4.31 shows the system recall rate which indicate how well the system can detect the manually labeled ground truths. In these graphs, the results for different combinations of n_{lon} and n_{lat} are shown respectively. The horizontal axes represent T_N. As shown in this graph, as T_N increases, the recall gradually drops. The exceptions were $n_{lon} = 20, n_{lat} = 15$ where $T_N = 6$ and $n_{lon} = 40, n_{lat} = 30$ where $T_N = 10$, that the recall rate was lower than others. There are two possible explanations for that. First, when T_N is small, the system outputs more regions as attentionally gazed regions. Therefore, since the method treats contiguous cells as an identical AG region, if one cell is recognized as AG and the recognition fails (or the result is rejected), all the contiguous cells cannot be detected as AG even if that is actually AG. Secondly, the larger a region is covered by

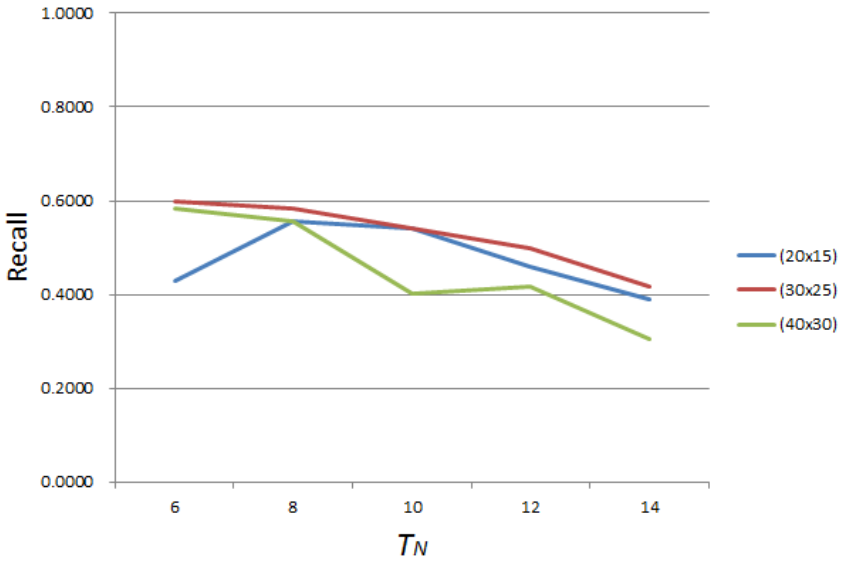


Figure 4.31: Recall rate for the proposed method for different cell sizes. Each curve represents a particular $(n_{lon} \times n_{lat})$ pairs. The values mostly decrease as T_N increases.

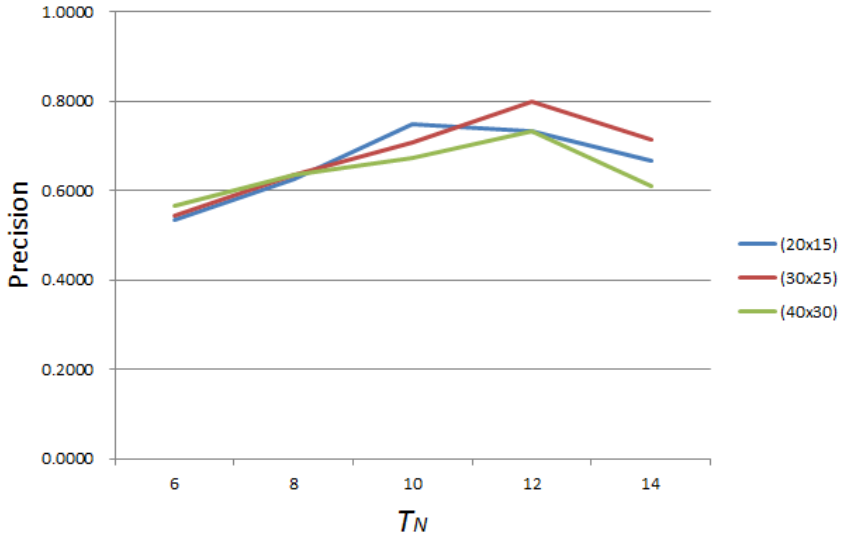


Figure 4.32: Precision rate for the proposed method. The graphs peak where $T_N = 10$ and $T_N = 12$.

one cell (the smaller n_{lon} and n_{lat} are), the more AG events are aggregated as one event. Hence, when a cell size is too large, two distinct events are not distinguishable.

Next, the system precision rate is shown in Figure 4.32. The rate indicates how precise the detection is performed. The results show that when T_N is too large, the precision decreases. This is mainly because of failures of object recognition which are caused by the selection of different frames from the video. The system waits until the number of gaze samples mapped in a particular cell reaches T_N. Then, it picks up a frame for a query

image. Therefore, sometimes the recognition fails even if the same object appears in frames when the images are distorted by several factors such as image blur, which frequently occurs when the user's head moves.

Finally, we compare the proposed method with the conventional method. The best precision and recall of the conventional method for the recorded test data were both 0.61. This is slightly better than the result from the proposed method, whose recall was 0.58 and precision was 0.64 when $T_N = 8$ and $n_{lon} = 30, n_{lat} = 25$. However, the results show the proposed method is still competitive even in the limited scenario that set of objects are all-known. More importantly, the processing time required for AG detection by the conventional method was 180 msec on average, whereas it was 82 msec by the proposed method.

In summary, the experimental results show that the proposed AG detection method can actually infer occurrences of AG events as well as the object recognition-based method. As discussed in Section 4.1.5.2, the AG ground truths are considered as the user's attention towards objects which are differently treated from short glance without attention. Even if we cannot name what draws the user's attention, detection of this type of attention can also be used in various applications. In Chapter 8, an example of such applications is presented.

4.2.5 Conclusion

This section presented a method to detect the user AG on objects without using object recognition-based approaches so that the method is adaptable for a wide variety of applications without the restriction of an object image database. The experimental results clearly show that the proposed method is competitive to the conventional method and faster. Using this method, we are able to detect a user's attention to arbitrary objects. A prototypical application which shows the benefit of such an AG detection framework is presented in Chapter 8.

4.3 Gaze-Guided Face Learning and Recognition

Similar to the gaze-guided object recognition method presented in the previous sections, I also discuss a gaze-guided image analysis method for human faces. This section presents a system that recognizes the person's face to which the wearer of the eye tracker is attending. In addition to recognition, the system also learns new faces by combining multiple input modalities: eye gaze and speech. For information presentation and user interaction, we use an HMD in combination with the wearable eye tracker (eye-trackable see-through HMD: see Section 3.3.3).

4.3.1 Introduction

Augmented human is a novel conceptual term of human-computer interaction where a human is supported by augmentative computational systems. This type of system can sense the physical world and provide the user with supportive information regarding individual problems. A typical example of such system is memory augmentation where a computer logs data of user's daily activities and use the log to help him or her to recall prior events or things that he or she does not remember. In several everyday scenarios, especially in a professional scene, these kinds of augmentative supports benefit many people. The medical scene is one of such fields where augmentative professional work support has great potential. A face-to-face examination process between the doctor and the patient can be improved using a system that augments the doctor's memory [ST13b; Son+13].

In this section, I present a face recognition system that recognizes a face of person which is attended by the user and acts as an "external brain" of the user. The system identifies the face the user is looking at by using his or her eye gaze data. Furthermore, the system can learn new faces online (in real-time), combining input from eye gaze and speech. The user speaks to the system: *"Now I'm looking at Mr. Toyama, please learn a new face."* to command the system to learn a new face that he or she is gazing at at the moment as shown in Figure 4.33. Here, eye gaze is effectively used to identify the attended face in the scene, which is not a trivial task when multiple faces are in presence in the scene.

The proposed gaze-guided face learning and recognition system have potential for different types of applications. Not only the professional medical domain described above, but also ordinary people also benefit from such a supportive system. Facial memory augmentation is also an application where we also directly benefit in our everyday life [Iwa+14]. Such a system is extremely effective for those who suffer from memory related illnesses such as *vascular dementia* or *Alzheimer's disease*.

Also, it is important to mention that eye movements are influenced by a viewer's conversational state [Ken67]. Eye gaze during mutualistic social interaction, especially during having a conversation, has been studied a lot. For instance, the study by Novick [Nov+96] showed that eyes have an important role for conversational turn-taking. Another study

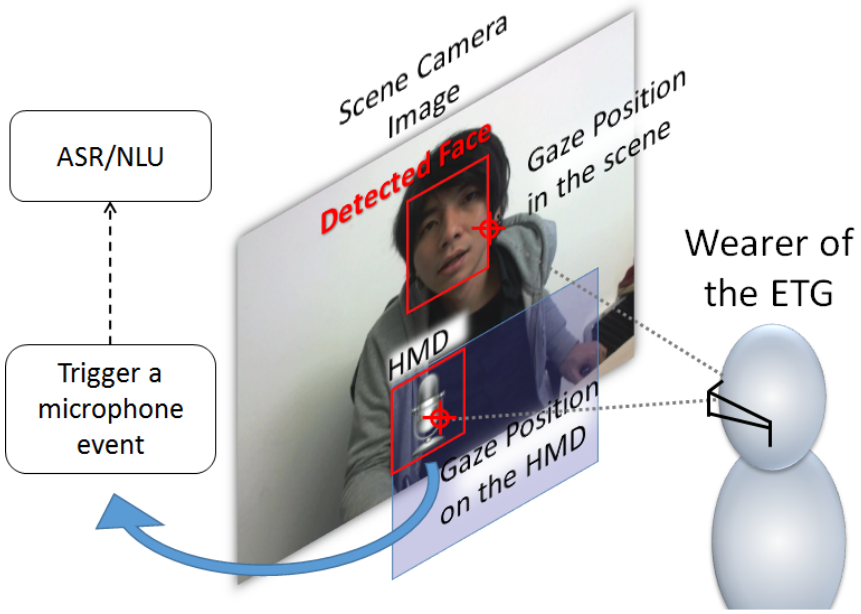


Figure 4.33: Face recognition and learning.

showed that eye gaze does not necessarily fixate on the eyes or face of the other person, because it may threaten the other [Maz+80]. Based on the findings in those prior studies, we can learn that eye gaze has different roles depending on the cognitive states. It is more dominant during face recognitions and communications. Thus, eye movement patterns may be different between cognitive states (e.g., recognition vs. communication) even if the viewer's eyes are fixated on human faces all the time. In our scenario, we do not deeply analyze which conversational state the user is involved in. Instead, we recognize the user attention to the other's face by analyzing the attentional gaze on a consistent face in a scene.

In this face learning scenario, we use a see-through HMD for an active facial image learning system. As previously discussed, several studies show that people often need to keep their attention towards the other person in face-to-face conversation. Augmented reality techniques using a see-through display, which overlay virtual machine-generated images onto the physical world can facilitate seamless information presentation without disturbing the users' face-to-face conversation. We may also need to avoid auditory feedback sometimes, because it can disturb natural conversation.

In the following subsection, I introduce related work regarding face recognition technologies and applications. Then, I present the proposed system architecture and a preliminary recognition test.

4.3.2 Overview of Related Work

4.3.2.1 Face Recognition

In the field of computer vision (CV), face recognition has been a main challenge for decades. An early method for facial image recognition was proposed by Bledsoe [Ble64], presenting the

challenges of recognition of facial images such as tilt, lighting condition, angle expression, aging, etc. These challenges are still fundamental issues for face recognition, though recent approaches show drastic improvements from a performance point of view. Above all, the method proposed by Turk [TP91] showed outstanding recognition performance in that time. Using eigenvectors of facial images, the method can rapidly and correctly detect and identify human faces.

Many recent face recognition systems employ a three-step approach [Zha+03]. First, a system detects facial regions in a given image and then extracts image features in the second step. Finally, it identifies individuals using the extracted features. For face detection, the method proposed by Viola and Jones [VJ04] showed the robustness to challenging datasets and it is nowadays widely used in many applications. In their method, Haar-like features are extracted from a bounding box. By sliding a window of the bounding box and classifying the area as a facial image or a non-facial image using so-called a boosted cascade, they localize a face in an image.

In addition to face detection, there are plenty of methods for face recognition, i.e., individual facial image identification. In face detection, a computer is queried "Where is a face?". But in face recognition, the query is "Who is this person?". Although features for face detection can be applied to face recognition problems (such as Histograms of Oriented Gradients (HOGs) [Dén+11]), researchers often use different features. Local Binary Patterns (LBP) are popularly used features for face recognition [Aho+06]. Moreover, V1-like features [Pin+09] also show the robustness for face recognition.

In this thesis, I adopt a three-step approach, using Haar-like features with a boosted cascade for face detection and LBP features with the Nearest Neighbour (NN) method for face recognition.

4.3.2.2 Eye Movements Analysis on Facial Image Perception

A study by Henderson et al. [Hen+05] showed human eye movements are functional during face learning and also during recognition. Human eyes fixate on particular areas of a face such as eyes and a nose for a certain duration, in order to gather visual information necessary for learning and recognition. Furthermore, Peterson and Eckstein also conducted an empirical study [PE12] where they showed eyes are looking at below the other's eyes during face recognition. These types of eye movement patterns during face recognition could be predicted using a computational sequence pattern recognition method such as Hidden Markov Models (HMMs) [Chu+14].

The abovementioned studies revealed characteristics of eye movements during face perception and suggest some important factors when we develop a gaze guided face recognition framework. However, we should keep in mind that the eye movements during such empirical conditions do not behave exactly same as those in a dynamic face-to-face conversation scenario. As previously mentioned, eyes may have different roles than recognition or learning, such as turn-taking [Nov+96]. Because the proposed system for gaze guided face learning and recognition is still a relatively new topic, I simply focus on attention duration on faces rather than a complicated eye gaze perception analysis such as one using gaze scan-paths.

4.3.3 Method

I present the method used in the proposed gaze-guided face recognition and learning system. First, I describe how the entire recognition process works and then describe individual components such as feature extraction.

4.3.3.1 Process Flow

Figure 4.34 is an illustration of the process flow of the proposed recognition system. Eye tracking data and scene images are captured by the eye tracker and are sent to the face detection module. Face detection is applied to the received image. The face closest to the gaze position is selected as the target face. If the distance from the gaze position to the closest bounding box d is more than the threshold value T_d, we discard the detected faces. Which means that if there are multiple facial regions are detected in one image, only the closest facial region is selected as the target.

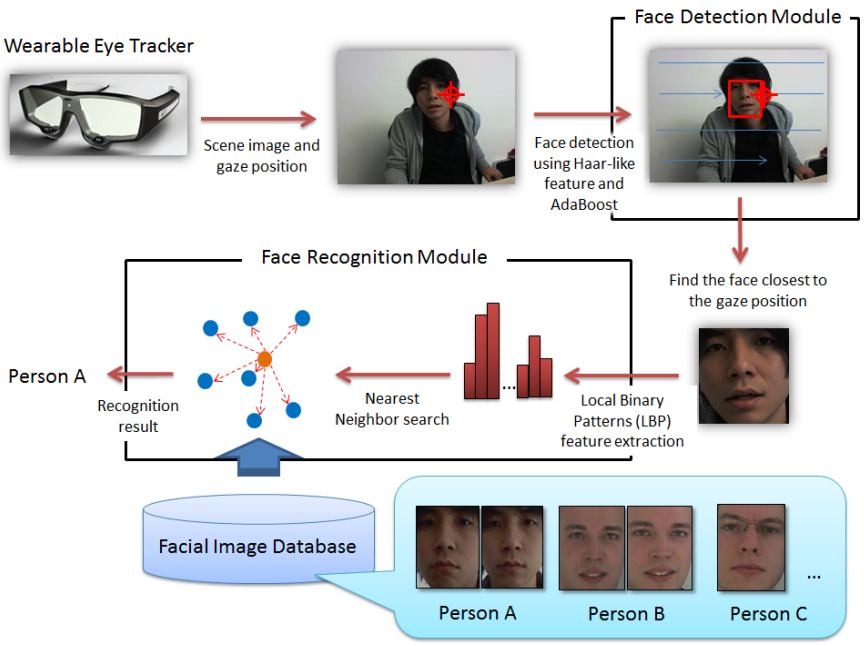


Figure 4.34: Process flow of the proposed face recognition system.

The selected target facial image is sent to the face recognition module. The module extracts image features (LBP) from the given image and identify the person in the pre-built facial image database. As the result, the identity of the facial image is output from the recognition module.

4.3.3.2 Face Detection

We use Haar-like features and cascaded AdaBoost proposed by Viola and Jones [VJ04] for face detection. Here detection is done by sliding a sub-window as shown in Figure 4.35 and continuously classifying the region surrounded by the sub-window as a face or not. From a sole sub-window, many Haar-like features can be computed. A Haar-like feature is

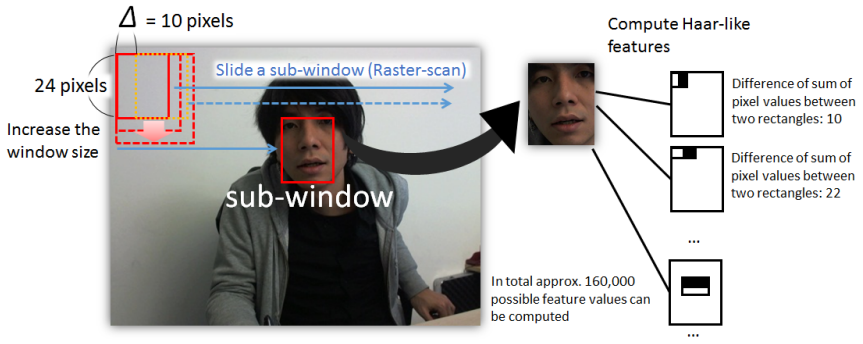


Figure 4.35: Face detection using Haar-like features.

a difference of sum of pixel values between two (three, or four) adjacent rectangle areas. The number of Possible Haar-like feature values within a sub-window with a size of 24×24 pixels becomes approx. $160,000$. In order to select more efficient features for face detection, AdaBoost weights a weak classifier which employs a single Haar-like feature as a feature value for classification. By aggregating classification results with weights from all weak classifiers, a strong classifier of AdaBoost classifies a given sub-window image as a face or not.

To speed up the classification process, the method employs a cascade approach where multiple strong classifiers are prepared and filter out less likely facial regions in an early stage of the entire classification process. By speeding up an individual classification process, we can detect facial regions in real-time. A sub-window is slid by Δ pixels and the size of a sub-window is also increased to detect facial regions with different scales. Classification is done for all the sub-windows, i.e., if we have 320×240 pixels in a query image, 3 scales of a sub-window (24×24, 32×32, and 64×64), and $\Delta = 10$, then classification is done $(320-24)/10 \times (240-24)/10 + (320-32)/10 \times (240-32)/10 + (320-64)/10 \times (240-64)/10 \approx 1700$ times. In the actual recognition system, Δ value is set smaller than this example (1 or 2), thus number of classifications becomes larger. Here the speed up technique by the cascade benefits considerably.

We use a Haar-like features-based cascaded AdaBoost object detector implemented in the OpenCV C++ library [Pro]. We also adopt the trained model for frontal face detection provided by the Open CV library.

4.3.3.3 Face Recognition

After detecting the facial image closest to the eye gaze, we recognize the detected face. We compute a vector of LBP features [Aho+06] and retrieve the nearest vector from the facial image database. First, I explain how to extract LBP features from a facial image.

For an LBP feature, a histogram of LBP values for $K \times K$ blocks from an image is computed. First, we divide an image into $K \times K$ blocks. Then for each pixel in each block, we compute an LBP value which encodes the number of neighbouring pixels that have a greater value than the pixel value of itself as shown in Figure 4.36. After computing LBP values of all pixels within a block, a histogram of LBP values of a block is computed. An LBP feature vector of an image is obtained by concatenating histograms of all blocks.

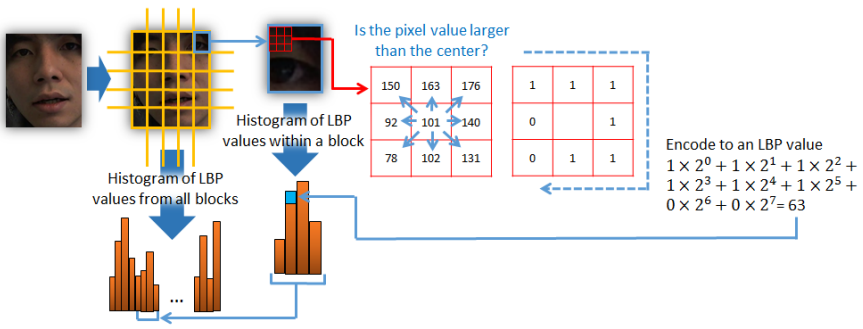


Figure 4.36: LBP feature vector (histogram) computation. The image is divided into 5×5 blocks ($K = 5$).

In advance of recognition, we need to create a database of facial image features. In order to cover various facial images of each individual, N images are captured from a single person with changing lighting conditions and facial expressions. Thus, if we have M individuals in the database, the number of LBP feature vectors is $N \times M$. Using a simple NN method, if a query facial image is given, we search for the nearest LBP vector from the database. If the Euclidean distance to the nearest vector is less than a threshold value T_f, the identity of the nearest vector is returned as the result. Otherwise, the system does not return any result (recognition is rejected).

In our user attention-oriented scenario, we also distinguish whether the user is attentionally gazing at the other face or not. Therefore, similar to the approach presented in Section 4.1, the consistence of recognized identity label is sequentially examined. If the identity label X is returned more than T_{dur} times before $\overline{X}$ is returned T_{noise} times, we regard the user is attentionally gazing at the face (thus, AG is detected), otherwise we regard face recognition results as non-attentional. The following face learning and face recognition result presentation are activated when such attentional gaze on a face is detected.

4.3.3.4 Gaze Guided Online Learning and Recognition Result Presentation

In addition to the recognition system, I also present a method for online face learning. In this online face learning scenario, we utilize an HMD in order to interact with the user. The HMD can guide the user visually. For example, as shown in Figure 4.37, it can prompt when the system is ready for learning so that the user starts gazing at the other's face and also when the learning is completed. In the proposed system, we use an off-the-shelf automatic speech recognition (ASR) engine to recognize the user's speech input to activate the face learning module. A microphone icon is in the field of view in the HMD all the time. Whenever the user attentionally gazes at the icon (Thus, AG on the icon is detected), it opens the ASR input. Then, the user can activate the online learning by giving speech: e.g., "Learn a new person, Takumi Toyama". As soon as the online face learning module is activated, the HMD guides the user to look at the corresponding face. If any facial image is detected near the gaze position (within T_d), the system saves the facial image as well as the identity label ("Takumi Toyama"). In order to have various facial images from one individual, the system has a time interval for detecting new facial images for L seconds. If N images are saved in the database, the system terminates the learning module and prompts in the HMD that the leaning is completed.

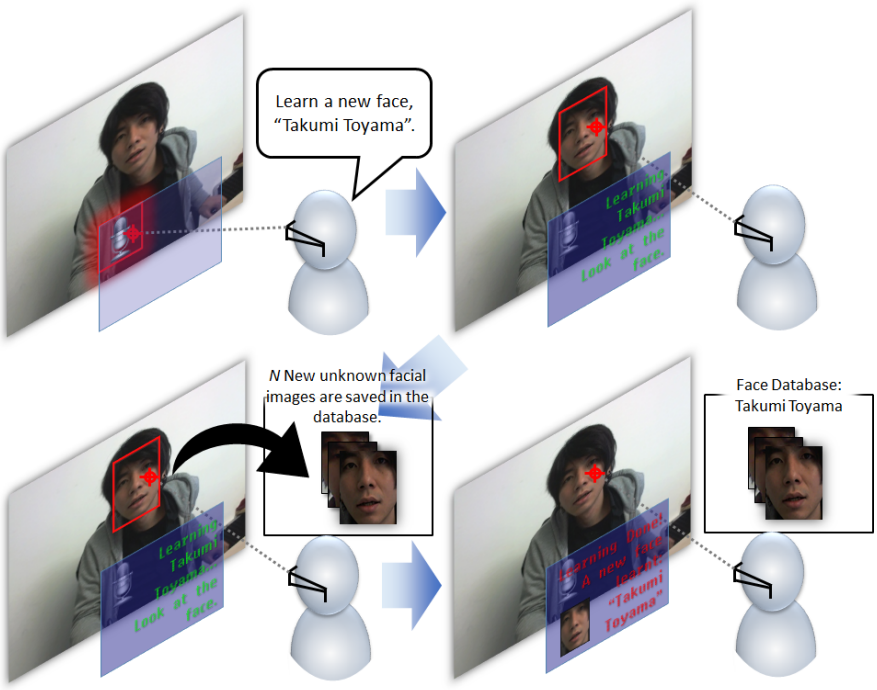


Figure 4.37: Proposed online face learning procedure. Top-left) The user activates the ASR and commands to start face learning by speech input. Top-right) The guidance text is presented in the HMD "Look at the face". Bottom-left) If any unknown face is detected, the system saves the facial image. Bottom-right) When N images are saved, it prompts that the learning is completed in the HMD.

In addition to the learning interaction, the HMD is also used to present a recognition result. When AG on a face is detected by the system, it presents the name of the recognized person and the reference facial image. Thus, the proposed system can be used to aid the user's memory for people's faces. As discussed in the introduction, the system can be effectively used in professional work such as a medical scene, where doctors need to examine many patients in a day and they sometimes need to examine a patient who was examined by another doctor. Using the proposed system, they can also collaboratively share information of a patient and examine processes can be sped up.

4.3.4 Preliminary Evaluation of the Learning System

We conducted a preliminary test for (online) face recognition. Thereby, we focused on the general applicability of the technical methods for a mobile active face recognition environment. In this preliminary test, six test users wore the eye tracker and we recorded two video files with eye-tracking data from each test person where they viewed eight individual's faces ($M = 8$) in the same condition. The recordings were done in an office room with a natural lighting condition. Since we focus on a personal memory scenario, we do not necessarily compete with a large scale of facial image database, where typically more than hundreds of individuals are included. Thus, we test with a somewhat small database size. The first video is used to extract training face images (5 face images, i.e., $N = 5$) from each person

and the second is used for testing. We then checked in how far the faces from the test video can be recognized while using the face recognition method. Figure 4.38 shows the precision-recall graph of the overall recognition result with the threshold value T_f from 0 to 25. In this figure, the recall and precision were averaged among the individuals. As one can see in the figure, the face recognition system achieved 0.97 precision with 0.81 recall with the threshold value $T_f = 17$. This result indicates that if the training is done in the same condition as the test case, this online face recognition method can perform reasonably well. We found out why the recall rate is lower: when a facial image is too dark, the face detection cannot be performed successfully. However, when we are applying to a particular scenario such as a medical examination scenario, we can provide for suitable conditions for face recognition (good lighting to reveal many individual face textures), thus this problem could be solved in a real examination environment.

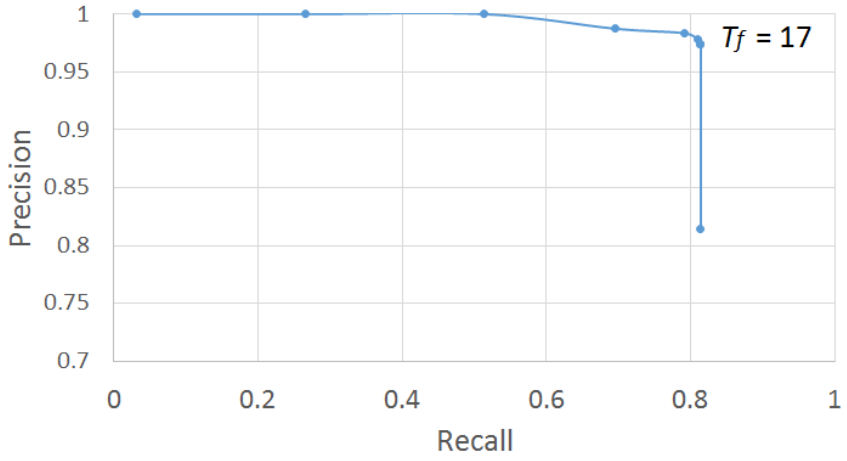


Figure 4.38: Precision-recall graph for the face recognition test. The graph was drawn by changing the threshold value T_f from 0 to 25. When $T_f = 17$, the precision and recall were 0.97 and 0.81, respectively.

4.3.5 Discussion

An advantage of the proposed system is that we can guide the user to perform proper actions for online learning using a see-through HMD and eye gaze input, maintaining seamless interactions. The user does not have to go back to a desk to look at computer screens to check the speech input and learning is properly processed. Furthermore, a computer can also be guided by eye gaze input in order to interpret the user intention that he or she needs to save new facial data. If there is no face near the gaze position, the system can infer that the online face learning activation might be a failure.

Another feature of the system is that it demonstrates a good example of fusion of multimodal inputs. Input by speech and eye gaze is effectively collaborated in order to realize a natural human-computer interaction. The icon of microphone can visually appeal to the user and he or she can intuitively predict what would happen when the icon is selected. The loop of interaction could be closed between the HMD and the user; thus, user does not have to take care of other interfaces.

In chapter 8, I present how the proposed face recognition or learning system could be used in a professional work scenario.

4.3.6 Conclusion

This section presented a method for online face recognition and learning which effectively employs eye gaze input for selecting a face in focus and an interaction interface. Using eye gaze input, the system can identify to which person the user is attending. This function can be used i) to detect the face the user wants the system to learn and ii) to present the information of the appropriate target person in which the user is interested. Furthermore, this type of learning and recognition system can also benefit from utilization of a see-through HMD, which can interactively guide the user to perform a proper action.

4.4 Summary

This chapter presented user-attended VCA approaches for non-textual content. First, I focused on a question: "Can a computer recognize user attention on objects?". In order to investigate how the user attention to objects can be recognized, I designed an experimental scenario of a museum, where the visitors attentionally gaze at the exhibits or objects. The object recognition framework can be extended by using eye gaze for recognition of use-attended objects. From the experimental results, I confirmed that the proposed method can robustly recognize user-attended objects compared to the conventional methods.

Then, the second question arose: "Can a computer also detect attentional gaze on arbitrary objects without recognizing the object?". To answer this question, I proposed a method to detect AG on arbitrary objects and content. The proposed method computes a gaze vector in a 3D space by combining the eye tracker and motion sensors. Accordingly, it can identify the regions which capture the user attention in a scene. These regions are considered as areas where objects-of-interest exists. One could apply further image analysis to the detected region if we need to detect which content the region contains.

Another typical case of non-textual visual content analysis is face recognition. Similar to attention to objects, human eye gaze may fixate on particular regions, but the order of which region to fixate first is not always the same. Thus, I employ a duration-based attentional gaze detection for gaze-guided face recognition as well. By using eye gaze input, one can develop an interactive system which can effectively drive human-computer interaction for online face learning and recognition. Also, utilization of a see-through HMD can benefit the user to see the other's face without being disturbed by information presentation.

Chapter 5

User-Attended Visual Content Analysis: Scene Text and Documents

In this chapter, I discuss user-attended VCA approaches for textual content. We analyze visual content in a scene when a user reads text. The main technical difference of this chapter from the prior one is that we have a prominent cognitive task (reading text) of the user. When text is present in the user's field of view and his or her eyes move alongside the text, one can intuitively infer that the user is likely *reading* the text. Such characteristic eye behaviours for text are more apparent than for objects and faces, since cognitive tasks that users might perform for faces and objects are usually more complex (recognizing an object, appreciating an artistic painting, recalling the name of content, etc.). Therefore, we must take into account these characteristics for textual visual content analysis. This chapter presents approaches for recognizing the text (words) to which the user is attending in a natural scene and document printouts reading scenario.

First, I discuss a method for user-attended text analysis in natural scene text using OCR (Section 5.1). Similar to the approach used in the VCA method for objects, we can speed up the recognition process for OCR utilizing eye gaze. I propose a prototypical application for real-time scene text translation that makes use of user-attended VCA on text. The application presents translations of words which are written in a foreign language, as soon as the user attends to the text in a scene.

Then, I present a method for analyzing user attention when he or she is reading text with a document printout (Section 5.2). Especially, I propose a method to identify gaze position on a document using an image-based document retrieval method. Using the proposed eye tracking method for document reading, one can analyze the reader's eye gaze in natural reading scenarios. As a result, the reader can interact with the document paper using his or her eye gaze.

Each section in this chapter is based on the work presented in [Kob+12; Toy+14a] and in [Toy+13c; Toy+13b], respectively.

5.1 Eye Gaze on Natural Scene Text

First, I discuss a method for the recognition of scene text that is attended or read by the user. User eye gaze can be used to identify to which word the user is currently attending. Consequently, we are able to boost the recognition performance compared to a normal OCR system. I present a method for the recognition of user-attended text when the user is naturally viewing text. Additionally, I also investigate an approach for scene text recognition and results presentation activation where explicit user eye gaze gestures are utilized. Since text is normally aligned horizontally (or vertically), a gaze gesture which mimics natural gaze behaviours during reading can be performed effortlessly by the user. I compare two types of gaze gestures for triggering OCR, which are particularly designed for interaction with textual content.

5.1.1 Introduction

A camera-based character recognition system has many possibilities to help our daily life [SX05; Wat+98]. One good example is a so-called translation camera system. The system recognizes a text in scenes and provides the user with translated words only by taking a picture of the words. Such a type of application is quite helpful especially when the user is in a foreign country and surrounded by a huge number of unknown words. One of the existing methods which can be used in this type of application was proposed by Iwamura et al. [Iwa+10]. The method recognizes words in the query image with high accuracy in real-time. Besides, it provides information about the recognized words to the user with multiple forms such as translated meaning, an image related to the word, and so on. However, this system requires the user to hold the camera and direct the lens toward the words he or she is interested in. This constraint limits the usability of the application.

One solution is to use a head-mounted camera. A character recognition system that uses a head-mounted camera to capture images was proposed by Merino-Gracia et al. [MG+12]. With this system, the user can access additional information of the interested word by directing the lens of the head-mounted camera to the word. Since this system does not require the user to hold a camera, the constraints is more relaxed than using a hand-held camera. However, this system has a problem that there is often a gap between the gaze point of the user and the direction of the user's head. Therefore, when the user likes to get the information about a certain word, he or she has to direct his or her head toward it. This might bother the user. In order to obtain the gaze position of the user, eye-tracking technology was developed. This thesis has already shown examples of applications that utilize eye tracking for object and face recognition (Section 4.1 and 4.3). According to the experimental results in these previous chapters, the recognition accuracy is improved when we use gaze information. Besides, there are two advantages when we use gaze information. First, it is useful to realize intuitive applications. People usually move their eyes instead

of moving their head when they find interesting content. Second, we can also reduce the computational cost of the recognition system by using gaze information. Since we can obtain the gaze point, we can apply the recognition process to the neighbor region of the point.

We first evaluate the effectiveness of using an eye-tracking system for word recognition in scenes with a view to realize a translation camera system. Since eye-tracking technology has not been utilized in word recognition to the best of my knowledge, investigating how effectively the eye-tracker works on the task is important. In a character recognition process, we used a method proposed by Iwamura et al. [Iwa+11]. Their method recognizes characters by using SIFT. We propose a word recognition method based on their character recognition method. In order to evaluate the word recognition method, we conduct two experiments. One is to optimize the parameters of the system. The other is to evaluate the recognition accuracy and the computational time of the method. In this section, I use Japanese as a query language to realize a translation camera system from Japanese to other languages.

Next, we also investigate suitable gaze gestures for triggering text recognition and results presentation. With a naïve method, the system attempts to recognize words and present the results every time it succeeds recognition. However, as introduced in the previous chapter, the *Midas touch* problem may occur when the system does not concern actual user intention. One solution for this problem in the previous chapter was to detect AG. That is, only when the user attentionally gazes on a word, the system presents the results. In this section, we also discuss another approach for solving this problem. Gaze gestures are broadly employed to obtain explicit user input [DS07; Wob+08; Roz+11; HR12]. We can also employ gaze gestures to trigger text recognition and results presentation. Based on prior study, we already know normal eye movement patterns for reading [Ray78], i.e., the user reads from left to right (in English, but also for some types of Japanese text). We can define a gaze gesture which is similar to such a natural eye movement pattern which is considered to be more acceptable and intuitive for users. In this thesis, I propose two different types of gaze gestures for a translation system and investigate the effectiveness of them.

In the following subsections, I present a method for gaze-guided OCR and gaze gesture recognition. Furthermore, I present experimental results that evaluate the system from several perspectives.

5.1.2 Proposed Approach

Figure 5.1 shows a sample scenario to use the translation camera system in natural scenes.
Similar to the previous chapter, we use a combination of a wearable eye tracker (SMI ETG) and a see-through HMD (Brother AirScouter) (see Section 3.3.3). We use the eye tracker to capture the user's gaze position when he or she is looking at text in a scene. Then, similar to other VCA approaches, cropped region-of-interest (ROI) images are used to recognize the characters and the words that the user is attending to. We propose two different interaction strategies. One is the AG-triggered OCR method where AG is detected to trigger the presentation and the other is the gesture-triggered method where user gaze gestures trigger the recognition and presentation.

Figure 5.1: Proposed translation system. In this example, English translations of the Japanese words the user is looking at in the real scene are presented in a see-through HMD and with Text-to-Speech (TTS).

5.1.2.1 Process Flow

Figure 5.2 shows the process flow of the proposed system. We crop an image region-of-interest (ROI) first. The cropping methods for the AG-triggered and gesture-triggered systems are presented in Section 5.1.2.2 and 5.1.2.3, respectively. A cropped image is then used to extract local features. We extend the character recognition method proposed by Iwamura et al. [Iwa+11]. Their method uses local features to recognize characters. These local features are extracted by using SIFT. In the proposed method, we adopt the affine-invariant version of SIFT (ASIFT) [MY09] to extract features also robust to perspective transformation. Using extracted ASIFT features, characters in the image are recognized. Furthermore, the system also identifies the word the user is looking at using recognized characters. It concatenates the recognized characters and retrieves the most probable word from a pre-created dictionary. If any probable word is retrieved from the dictionary, the system presents the result and translation in the HMD or plays back TTS audio data.

Differences between AG-triggered and gesture-triggered methods are: i) image ROIs are determined differently. The AG method employs a fixed size of local window whereas the gesture-triggered method determines the ROI size according to user's gesture input. ii) The AG-triggered method only recognizes and returns one word the user is temporarily attending to whereas the gesture-triggered method recognizes all the words included in the image ROI.

OCR for a high resolution image is still computationally costly. When we apply the OCR recognition algorithm (in combination with the text window detection) to the entire

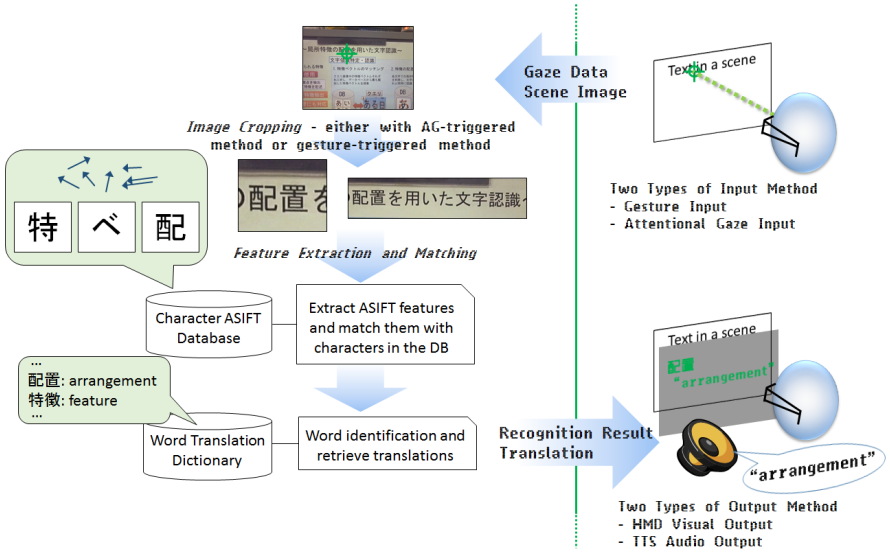


Figure 5.2: Process flow of the proposed translation system. ROI Images are cropped according to an input method (either AG-triggered or gesture-triggered). ASIFT features are extracted and matched with the features in the database. Then the word and the translation are retrieved from the dictionary.

image obtained from the scene camera, it cannot be processed in real-time. By focusing on a particular region in a scene, we can reduce the computational cost required in the recognition process.

5.1.2.2 AG-triggered OCR Method

Image Cropping and Feature Extraction With the AG-triggered method, we crop a fixed size of image ROI. Thus, given a gaze position $p_g = (x_g, y_g)$ and a scene image, a rectangular area $R = (x_{TL}, y_{TL}, x_{BR}, y_{BR})$, where (x_{TL}, y_{TL}) and (x_{BR}, y_{BR}) are the top-left and bottom-right corner of the image, is determined as follows: $R = (x_g - w_{local}/2, y_g - h_{local}, x_g + w_{local}/2, y_g + h_{local})$, where w_{local} and h_{local} are the width and the height of a local image region, respectively. In the experiment, we test different values for w_{local} and h_{local}. To recognize characters, we extract local features in the local image ROI.

When characters in an image are too small, one cannot extract sufficient local features. To solve this problem, we magnify an image ROI to increase the image size. However, if the image ROI size is increased, the time required for feature extraction increases too. Thus, we test different magnification ratio M_{ratio} values to find a good trade-off in the experiment.

Character Recognition Method First, the proposed method extracts local features from the local ROI of a query image. Then each feature is matched to the most similar feature extracted from reference character images (stored in a character database) as shown in Figure 5.3. In order to reduce the computational time, we use an approximate nearest neighbor search method proposed by Iwamura et al. [Iwa+13]. If only one character is in the query image, it can be recognized by using a simple voting method. A vote is cast for each reference character whose local feature is matched to the local feature from the query

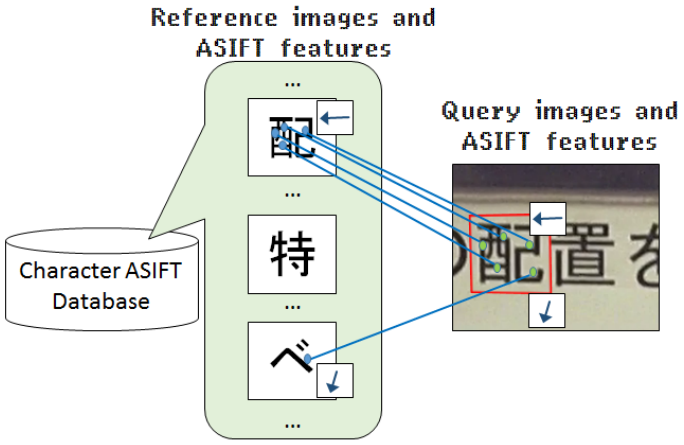


Figure 5.3: Character recognition method. Each feature in the query image is matched to the most similar feature extracted from the reference character images.

image. Then, the reference character which has the largest number of votes is returned as the recognition result. However, a query image usually has many characters. In order to recognize multiple characters at the same time, we use arrangements of local features extracted from each character to estimate the region of each character in the query image. Specifically, three pairs of matched feature points are used to calculate an affine matrix to project the character region upon the query image. Each character region is marked with a bounding box. The bounding boxes are projected according to the estimated affine transformation matrix. After all character regions are estimated, we can apply the simple voting method to each character region. A score for each character is given by

$$\text{score} = \frac{m_p}{\sqrt{r_p}},$$

where m_p is the number of feature points matched to the recognized character inside the character region and r_p is the number of feature points extracted from the reference image of the recognized character. $\sqrt{r_p}$ is used to normalize the difference between the number of feature points extracted from each reference character [Kob+13]. Since the projected character region sometimes largely overlap with each other, we group such characters. Overlapped character regions are grouped if they satisfy the inequality given by

$$\text{dist} < \text{mean\_length}/2,$$

where dist is the distance between the center of two character regions and mean length is the average length of each side of the two bounding boxes. After the process, the recognized character with the highest score among them is treated as the recognition result in the group. Generally, the character recognition process finished in less than one second with Intel Core i5 2.53GHz CPU.

Character Concatenation Recognized characters in the query image are then concatenated with their adjacent characters to obtain words. Certain two characters are concatenated if

they satisfy the inequality given by

$$\text{dist} < \text{mean\_length} \times 1.2,$$

where the meaning of each word is the same as before. Based on the prior experiment by Kobayashi et al. [Kob+13], it was shown that 1.2 is an optimal value for relaxing the distance constraint.

Word Identification and Translation Our Japanese-English translation system provides the user with a translated Japanese text snippet as a direct translation from the Japanese text snippet. For the test system, we implement our own translation function. First, we create a Japanese-English dictionary using $10,000$ Japanese words. We selected $10,000$ common Japanese words from a common Japanese dictionary service and translated them into English using the Microsoft Bing Translator API[1]. The simple translation process works as follows: 1) The recognized OCR text is preprocessed by a very shallow text processing pipeline for Japanese tokenization. This method is specialized to Japanese characters; 2) The individual Kanji[2] tokens (possibly compounds) are matched against an approximate Kanji index that uses a Levenshtein distance metric; 3) The token-by-token translation of the nearest Kanji compounds in the dictionary according to the distance metric are presented; and 4) If the Levenshtein distance exceeds a threshold, no translation is returned (we assume a bad OCR result). It is to be noted that complex linguistic-based translation pipelines can easily be integrated into the architecture. However, we relied on a fast on-board solution for the translation integration test. There are also several online translation services available, but they are dependent on the speed of the wireless network and are blackbox processes. Their utility evaluation is beyond the reach of evaluating the real-time feature of the head-mounted text translation system using eye gaze input.

5.1.2.3 Gesture-triggered OCR Method

An intelligent head-mounted interface should filter out the irrelevant information and provide the information only when it is needed and about what is needed in the context of the text reading and translation scenario. The first method presented above employs attentional gaze (AG) to trigger recognition and presentation. Another method for interaction is to use gestures where the user can explicitly show his or her intention to a computer. In this thesis, I present two types of gaze gestures for text recognition and translation system.

Text-based Gaze Gesture We propose two gaze gestures for an OCR text reading and translation scenario and investigate which type of gesture suits our OCR scenario best. Essentially, the first looks at the beginning and the end of the text line alternately and repeatedly (gaze repetitive leap), and the other is to move gaze from the beginning to the end gradually (gaze scan). Figure 5.4 shows these two gestures. These two types of gaze gestures can be divided into two groups based on the deliberateness and the complexity of the gesture. The gaze scan gesture is less complex and can occur less deliberately, whereas gaze repetitive leap is more complex and hardly occurs without intention. It can be that the less complex the gesture is, the more false recognition results (false positives) occur.

[1] http://www.bing.com/translator

[2] Actually, it contains Kanji, Katakana, and Hiragana Japanese characters mixtures. For the sake of simplicity, we say Kanji.

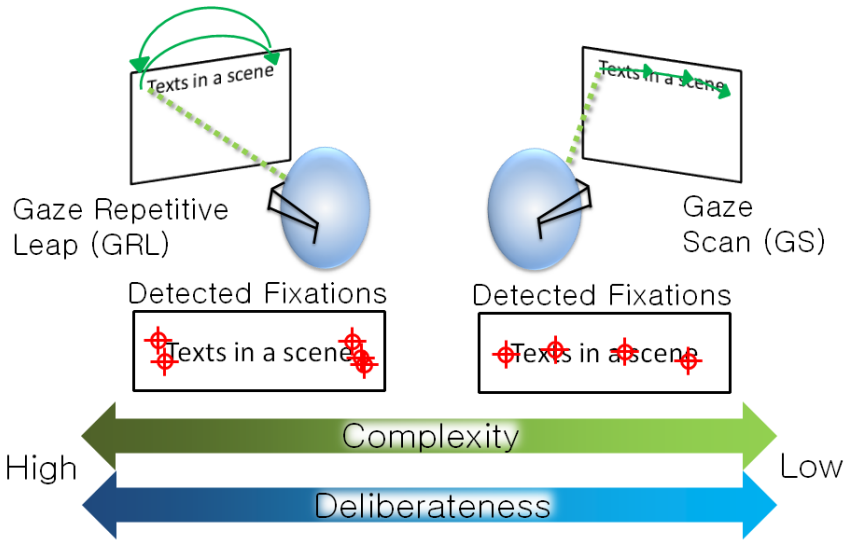


Figure 5.4: Two proposed gestures. *Gaze Scan* occurs more often and undeliberately in our daily life.

However, if the gesture is too complex, it is also quite demanding for the user. Thus, we need to explore and evaluate the trade-off of the deliberateness and complexity of the text-based gaze gesture.

The recognition algorithm of each gaze gesture is as follows:

Gaze Repetitive Leap (GRL): If a fixation that lasts one second is detected, it activates the recognition process and set the fixation as the start point. If the next detected fixation is right from the start fixation within $\pm 30^{\circ}$, which is a normally acceptable range for users as right direction, it is set as the end point fixation. If the third fixation is within d pixels from the start point, it continues the recognition; otherwise the recognition process is discarded. If the following fixations are within d from either the end point or the start point (switches alternately) and n of such fixations are detected in total, the gesture is recognized. By changing n in the GRL gesture, we can increase the complexity of the gesture. If n is small, the gesture can also occur less deliberately. d is a parameter for determining how many pixels the gaze position may be deviated from either the end or start point[3].

Gaze Scan (GS): If a fixation that lasts one second is detected, it activates the recognition process and set the fixation as the start point. If the next detected fixation is right from the start fixation within $\pm 30^{\circ}$, it continues the recognition; otherwise the recognition process is discarded. If such a fixation (within $\pm 30^{\circ}$ to the right) is detected continuously more than three times, it is recognized as a gesture. The end point is determined when the fixation point swerved from $\pm 30^{\circ}$.

[3]These parameters (d and n) may be changed depending on recognition scenario. However, for the simplicity we fix these values in the experiment in Section 5.1.3.2.

Image Cropping and OCR When the end of the gesture is recognized, the system crops the text region indicated by the start fixation point and the end fixation point as shown in Figure 5.5. As a background process, scene image tracking using the Lucas-Kanade method [LK81] is running, which tracks reference points in scenes. Based on image tracking, we can compute the relative distances between the scene image frames. Thus, we can estimate the position of the start point even at a later frame[4]. The cropping rectangle area is determined as follows: Left: 50 pixels left from the start point, Top: 50 pixels above from the higher point of either of the start point or the end point, Right: 50 pixels right from the end point, and Bottom: 50 pixels under from the lower point of either of the start point or the end point. From normal distances when people view posters (1.0 - 2.0 m), the rectangle size can reasonably contain the whole characters. In the experiment (Section 5.1.3.1), we check the size of characters from different viewing distances. The cropped text image is sent to the character recognition module similar to the AG-triggered method.

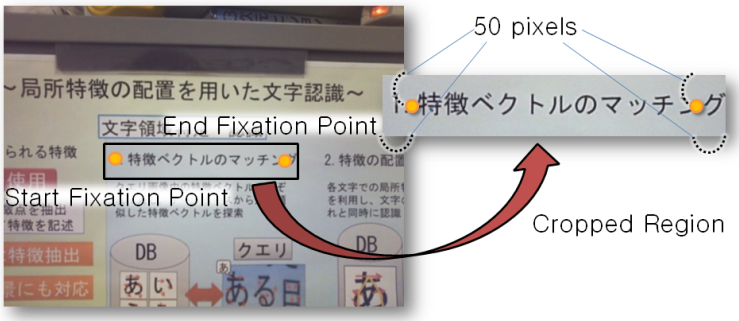


Figure 5.5: Text region cropping guided by a gaze gesture.

Visual Feedback in HMD Screen Once the start of the gaze gesture is recognized, a navigation image is presented in the HMD screen as shown in Figure 5.6. In this way, the user can ensure that the start of the gesture command is correctly recognized by the system. When the end of the gesture is recognized, the OCR module is triggered and returns the result of the text recognition of the given textual image. The resulting selected text snippet which also includes the cropped image is visualized in the HMD screen, so that the user can check if the correct region-of-interest is recognized; the visual feedback is given instantly in the virtual screen. If no Japanese (Kanji) character is recognized in the given region, no visual feedback is presented in the screen. Hence, false gesture recognition can be rejected based on the character recognition result. If gaze gesture is recognized mistakenly (false positive) and actually no text is read, it will be classified as gaze gesture detection error (false positive).

By taking the gaze position in the HMD into account, we can present the navigation near the gaze position as shown in Figure 5.6. Dynamic text management and intuitive positioning of the augmented translations in the users field of view to migrate user-centric text content is important because if the visualization is far from the user's focus, it cannot be perceived easily; and therefore, the see-through feature of the HMD plays an important

[4]To obtain the entire image from the beginning, one sometimes has to apply mosaicing; however, it is not integrated in this work because we focus rather short texts and they can be captured by one frame.

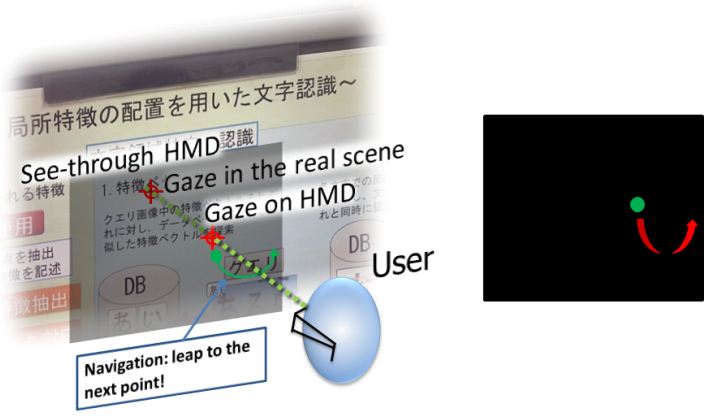


Figure 5.6: The interaction navigation is presented nearby the gaze position. In this example, GRL navigation is shown in the HMD.

role in this system. Another consideration is the position of the translated text throughout the user's mobile environment [Orl+13a].

5.1.3 Experiments and Studies

5.1.3.1 AG-triggered OCR

In order to evaluate the effectiveness to use eye-tracker for word recognition, we had two experiments. The first one was to optimize three parameters, the size of cropped image w_{loca} and h_{local} and the magnification ratio M_{ratio}. The other one was to evaluate the recognition accuracy and computational time of the proposed system. In the experiments, we employed 71 categories of Hiragana, 71 categories of Katakana (see Appendix B) and $1,945$ categories of Kanji (Chinese character) tokens in MS Gothic font for reference characters. These Kanji categories are so-called "Joyo-Kanji", which stands for a collection of Kanji tokens used in everyday use cases. All experiments were performed on a computer with Intel Core i5 2.53GHz CPU and 6GB memory.

Parameter Optimization To optimize our word recognition system, we compared the performance of the system with changing two parameters related to the size of a query image. The first two parameters were the size of an image cropped from a captured scene image (w_{local} and h_{local}) and the other parameter was the magnification ratio of a cropped image M_{ratio}. Since there is a trade-off between the accuracy and computational time, we need to select a good parameter combination.

First, we select a well-balanced magnification ratio. Before we started the experiment, we investigated the relationship between the size of a character and typical distances from a user to the characters to estimate how far a user looks at characters from. We asked five participants to look at words on a wall from the distance they think natural to look at them. We prepared six words including 20 characters in total and the length of each side of the bounding box for each character was six centimeters. As a result, the range of the distance was approximately between 1.0 and 2.0 meters. Thus, we investigated the accuracy of character recognition when the characters were captured from 1.0, 1.5 and 2.0 meters

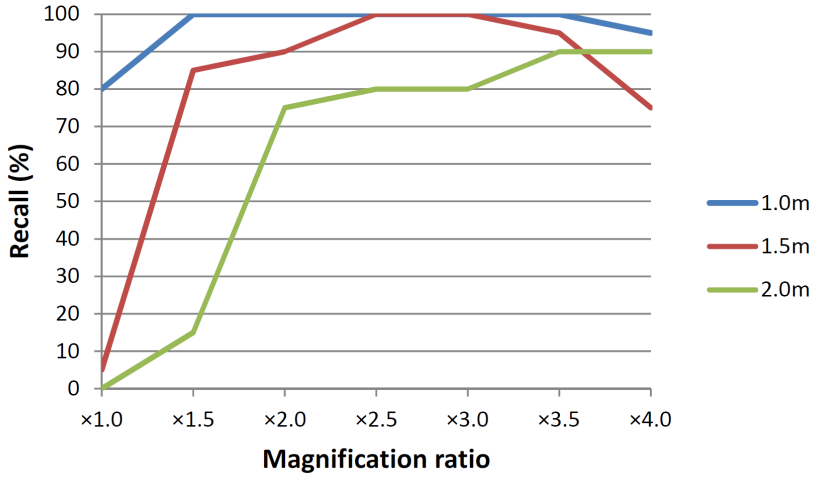


Figure 5.7: Relationship between magnification ratio and recall of character recognition.

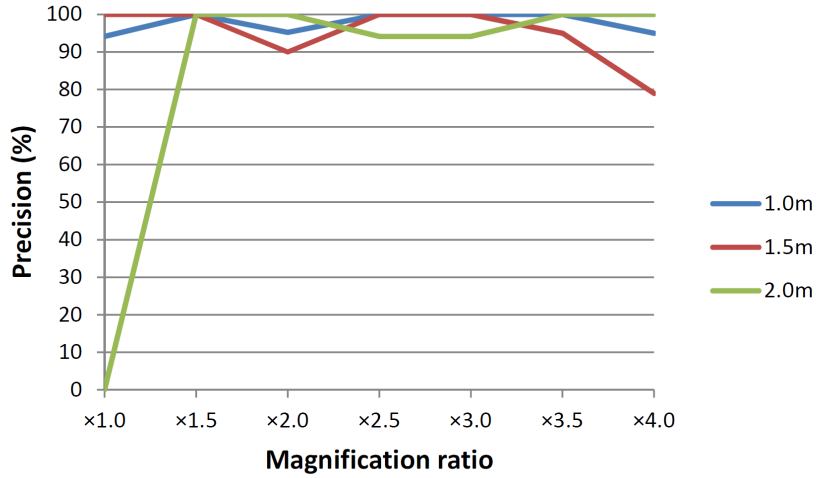


Figure 5.8: Relationship between magnification ratio and precision of character recognition.

distance, respectively. Figures 5.7 and 5.8 show the relationships between the magnification ratio values M_{ratio} and the recall or precision of character recognition for each distance. Recall and precision are calculated by

$$\text{recall} = \frac{c_r}{n_c}, \text{precision} = \frac{c_r}{n_r},$$

where c_r is the number of correctly recognized characters, n_c is the number of characters on the wall and n_r is the number of recognized characters including correct and incorrect recognition. For each distance, recognition accuracy increased as the images were digitally magnified. However, the recognition accuracy of 1.0 and 1.5 meters decreases when the magnification ratio reaches 4.0. This is because when an image is magnified too large, the image is blurred and the stability of local features declines. Table 5.1 shows the relationship

between the distances from a user to the characters and the length on each side of a bounding box for a captured character. By investigating how the length of each side of a

Table 5.1: Relationship between distance from a participant to captured characters and length on each side of a bounding square of a captured character.

distance	1.0 m	1.5 m	2.0 m
length (pixel)	45	30	25

character and the magnification ratio affected the recognition accuracy, we found out that the length should be more than 60 pixels to achieve over 80% recall rates. For example, when the distance was 2.0 meters, we need to magnify the image 2.5 times to exceed the length of 60 pixels. For further analysis, we focus on magnification ratios of 2.5, 3.0, and 3.5, which yielded high recall and precision.

In order to find a good combination of the magnification ratio and the size of a cropped image, we conducted another experiment. We investigated how size of a cropped image and magnification ratio affect computational time. Table 5.2 shows the relationship between them. Computational time shown in the table was measured as the time needed to recognize characters in an image. From this result and Figure 5.7 and 5.8, one might think 200x200

Table 5.2: Relationship among size of a cropped image ROI w_{local}xh_{local} and magnification ratio M_{ratio} and computational time (millisecond) to recognize characters in an image.

	Size of a cropped image (w_{local}xh_{local})		
	200x200	250x250	300x300
$M_{ratio} = 2.5$	813.2 msec	855.0 msec	998.2 msec
$M_{ratio} = 3.0$	874.8 msec	1118.0 msec	1424.3 msec
$M_{ratio} = 3.5$	1073.7 msec	1473.8 msec	1790.8 msec

seems better with respect to the computational time. However, the size was sometimes too small to contain all characters in a word when images were captured from 1.0 meter distance as shown in Figure 5.9. Therefore, we selected the combination of parameters

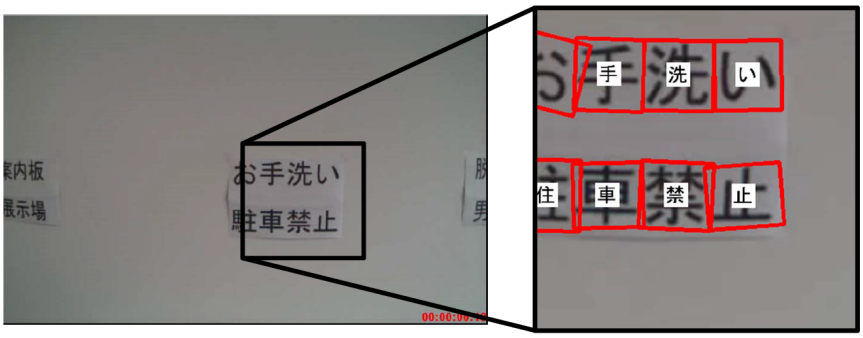
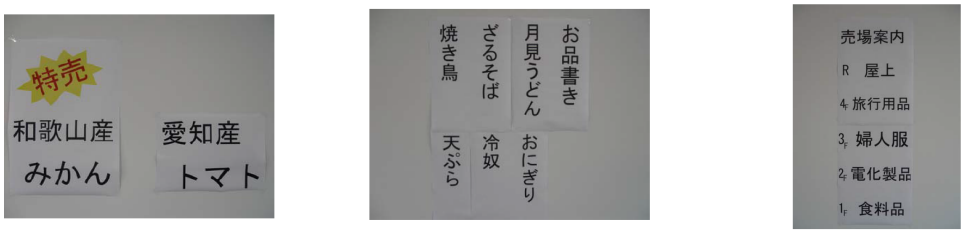


Figure 5.9: Failure case of recognition result. Because the size of the cropped image was small, some characters were not contained in the image completely.

that the size of a cropped image was 250x250 pixels and the magnification ratio was 2.5 since the computational time did not reach one second. We used these parameters in the following experiment.

Evaluation of the Word Recognition System Next, we conducted another experiment to evaluate our word recognition system. We asked 13 persons to look at words on a wall as they usually do so. We set the distance between the wall and the persons as 1.5 meters and they looked at the words from two viewpoints, straight in front of the wall $0°$and $30°$left from that point. Figure 5.10 shows three different scenarios we prepared to simulate the real daily scenes in Japan. They contained 18 Japanese words and 60 characters in total



(a) Offer in a supermarket (b) Menu in a restaurant (c) Floor guide in a department store.

Figure 5.10: Three different daily shopping scenarios in Japan. These scenarios simulate real daily scenes.

and the length of each side of a bounding box for a character ranged from 5.5 to 7.0 centimeters. First, we calibrated the eye-tracker by asking the user to look at five points on the wall. Then, we asked each of them to gaze at each word for several seconds. We recorded the video files for every word and then applied the character and word recognition process only to the fixated frames by the eye-tracker. Ten frames were used for each word and we calculated the average of the recognition results. In the experiment, we treated only one recognized word which was closest to the gaze point as the recognition result. Table 5.3 shows the recall and precision of character and word recognition calculated from the whole recognition results. We achieved a high recall rate for character recognition with

Table 5.3: Overall recognition accuracy of the character and word recognition system.

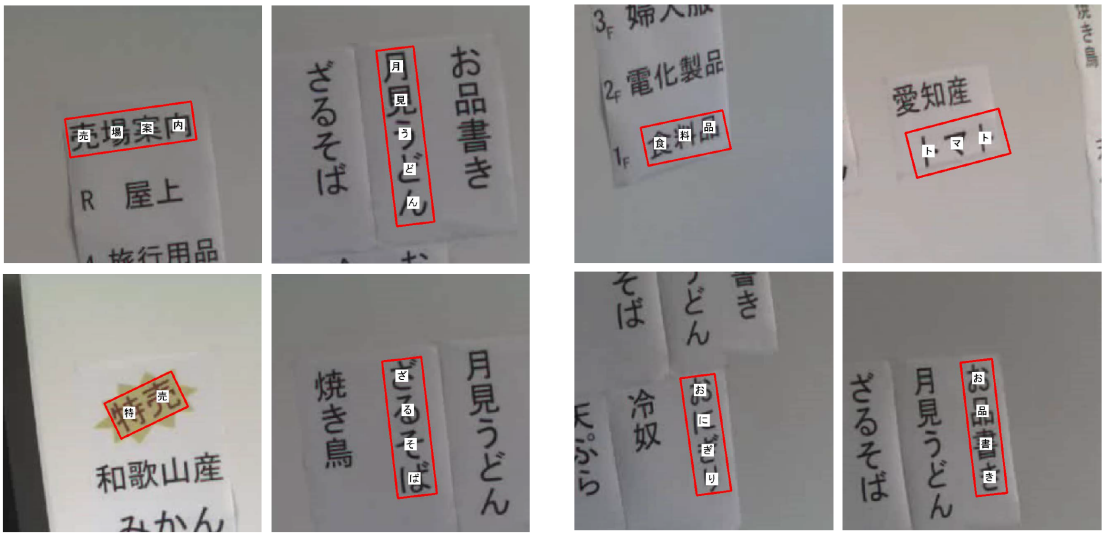
	angle	
	$0°$	$30°$
recall [%]	88.1	69.3
precision [%]	94.3	90.4
recall (word) [%]	69.7	42.8

the angle of $0°$. Although the recall decreased when the angle was $30°$, the precision for both angles was over 90%. The drop of the recall was caused by changing a parameter of ASIFT description. We can choose which to prefer, a robustness to perspective distortion or a reduction of computational time by changing the parameter. Since we selected the latter in the experiment, the recall decreased when the angle was $30°$. The recorded gaze data showed that almost all gaze positions were on the correct query word. Only when the

user gazed at words which were much lower than their eyes, the gaze positions sometimes pointed to the wrong word.

The average computational time required for recognition was 917.0 msec with a cropped image and 3101.2 msec with an entire scene image. When we use the gaze-guided cropping method, the computational time was three times faster than the time without it. When we did not use the gaze information, we used the entire image without cropping since there was no information which region to crop. Regarding the size of a cropped image, only when the user gazed at the edge of a long word, the system failed to contain the whole word region into a cropped image. This problem can be solved by accumulating the information of recognized characters through several frames as we discuss later in this section. From these results, we confirmed that we can improve the performance by using gaze information.

Next, we consider the word recognition accuracy. Figure 5.11 shows examples of correct word recognition result. A red bounding box is the region of the word and recognized



(a) 0° (b) 30°

Figure 5.11: Examples of correct word recognition results.

characters are put on the center of each character region. For both angles, the recognition accuracy decreased compared with the results of character recognition (see Table 5.3). There are two reasons of this result.

First, in order to detect a word region, we connected adjacent characters. Thus, when the method fails to detect a character in the middle of a word, it cannot connect the separated parts of a word. To solve this problem, it would be effective to use a word segmentation approach. Maximally stable external regions (MSERs) can be used to detect word region as used in [MG+12]. By combining MSERs with gaze information, we might reduce the computational time. Besides, in order to improve the recognition accuracy we consider accumulating feature points and recognition results through several frames when we realize a translation camera system. If the user gazes at any interesting word for several seconds, the system can accumulate the recognition results through the several frames. This method can recognize long words even if they are not contained in a cropped image

completely.

The second problem was that we recognized only one character per a character region in our method. If regions of recognized characters overlap with other ones, we treated the character which has the highest score among them as the recognition result. However, since many Japanese characters have similar shape, they were often mis-recognized as other similar characters. Thus, we sometimes must consider the rest of detected characters in a character region. A simple way is to create a candidate character lattice from the detected characters in a word. We can find the best combination of characters to be a proper word by considering the scores or by comparing with the list of words in a dictionary.

5.1.3.2 Gesture-triggered OCR

Gesture Recognition We also provide an initial evaluation of the proposed gesture recognition approach; we sought to determine whether the two proposed eye gestures are adequate for triggering the translation function while reading the text.

To test this, a user evaluation under realistic circumstances has been conducted including the following steps: First, we tested the gesture recognition without using any textual real world image or document in order to evaluate the acceptability of the proposed gestures for the users, i.e., how well people can perform those gestures and how much the system can recognize them in general. The experiment included ten participants, ranging from age 22 to 56, with an approximately even number of males and females. The participants were asked to perform the two gestures in order to trigger the translation function. Two types of gesture navigation sheets (as shown in Figure 5.12) were presented to the user and we asked them to perform each gesture (GRL and GS) on each sheet. In addition to the gesture types,

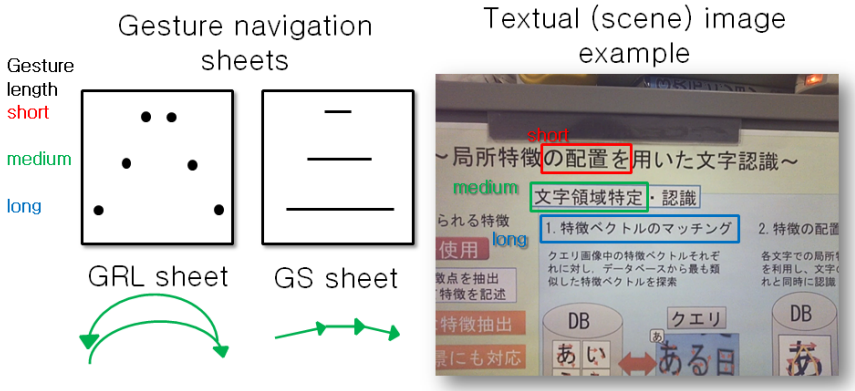


Figure 5.12: Gesture navigation sheets and an example of actual scene image. The scene image contains a complex background.

three types of gesture length are prepared here (short, medium and long), to compare the recognition performance of each gesture with a different length. The summary of accuracy of all gestures is shown in Table 5.4. For GRL, n was set to 4 because four time repetitions of this gesture do not often occur in natural behaviours but they are still acceptable to users. Additionally, d was set to 50. Even if the user tries to gaze on the same point, the gaze usually drifts few pixels. The setting of $d = 50$ is a suitable value for allowing such drifting gaze. In general, the results show that the longer the gesture length is, the less accurate

Table 5.4: Gesture recognition accuracy.

	GRL gaze gesture		GS gaze gesture	
length	sheet	scene image	sheet	scene image
short	80%	83%	67%	77%
medium	60%	60%	80%	80%
long	60%	33%	87%	83%

the GRL gesture becomes; contrariwise, the longer the GS gesture is, the more accurate the recognition is. We found that this was because the user could not find the end point easily when the distance of two reference points was large; contrariwise, using the GS gesture it is easier to track a path (line) by eye gaze when it is longer. In a second evaluation step, we tested the recognition on a textual (scene) image example of proper Japanese text of different lengths. An example image that we used is shown in Figure 5.12 (right), and the result is also shown in Table 5.4. Here, we asked the users to perform the gesture on proper text of length: short, medium, or long. Compared to the non-textual image, it can be said that it becomes even harder to find the end point with proper texts. The GS gesture slightly gets better with texts. Some users mentioned that to move eyes along a text line is easier than to move it along a normal straight line. Since these experiments were conducted without a training phase (for the users, to try to learn these gestures), it would be interesting to see how the performance to use those two proposed gestures improves over time. Though three users had difficulties because they had to concentrate more than usual and felt stressed, these gestures were rather easy and intuitive to perform for the majority of users. We might switch the gesture depending on the length of text, since GRL suits short text length, while GS suits long text length. Furthermore, the visual feedback could reasonably navigate the user for the correct gesture throughout the experiment. Many users failed the gesture without the visual feedback in the HMD. This result shows the benefit of a see-through AR screen for visual feedback in the proposed system though the experiment we conducted is only adequate to show the basic effectiveness of our algorithm.

Gesture-based Cropping and OCR Performance Next, we tested the gesture-based image cropping and OCR performance. One of the results of image cropping is shown in Figure 5.13. Similar to this image, most of the texts were cropped correctly if the gestures

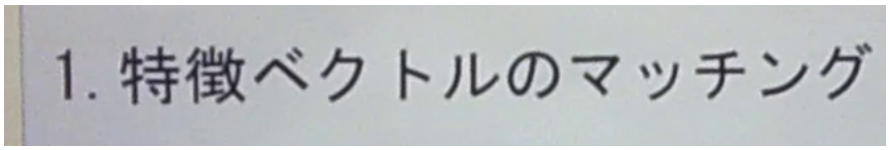


Figure 5.13: Example of cropped image by gaze gesture.

were correctly recognized. However, sometimes it contained more (or less) characters than the appropriate output if the end or the start point is not accurately detected. This was because of an eye tracking error or because the user could not gaze on one point properly. The users had difficulties to gaze on a consistent point if there is no characteristic (reference point). In addition, since we adopted a fixed length for image cropping region, it contains more characters than appropriate when the distance to the text becomes further. One could

solve this problem by applying a text detection approach to this small size (cropped) image, which does not require computational cost so much.

Furthermore, OCR is applied to the cropped images. We compared the processing time required for the recognition of a cropped image and a whole image. For a whole image, it required 4.77 sec (on average) to process the recognition. On the other hand, it required only 0.19 sec. (on average) when we crop the text region from the whole image. This result shows that we can effectively reduce the computational cost by using the gaze gesture guided image cropping. Since the OCR method is specialized to Japanese characters, we used Japanese text in this experiment. However, the OCR engine used in this work can be easily replaceable with another engine.

5.1.4 Conclusion

This section presented an OCR system integrating eye gaze in order to recognize the text the user attends to. In the experiment of both the AG-triggered one and the gesture-triggered one, we showed that the computational time can be improved keeping the recognition accuracy high. Consequently, the system can present a translation of the word that the user wants to know by analyzing the user's eye gaze. The experimental results of the gesture-triggered one also showed the acceptability and the feasibility of the gaze gesture recognition, and the efficiency of gaze gesture as an indicator of an interest text region. One good implication of these methods is that a computer can recognize characters and text as humans do. When we read text, we have a particular order of language processing. By mimicking such a behaviour, a computer can infer which word the user is interested in and which word is more important for the user. Future work is to investigate the capability of the gesture thoroughly and to conduct further user studies for a comprehensive evaluation of the approach.

5.2 Eye Gaze on Documents

Recently, researchers proposed to use eye tracking to analyze user attention during reading a document and for interaction [Bus+12; Bie+10]. However, such research is usually restricted to a computer display scenario where the users have to sit at a desk because they use a remote eye tracker. In this section, I present a method that can remove such a restriction using a wearable eye tracker and image analysis. By analyzing a scene image and applying a document image retrieval engine, we can also identify which document and where in the document the user is reading, even with a printed physical document paper.

5.2.1 Introduction

The advance of recent computer technologies evolves people's reading life. Today, we have many choices of document forms; we have not only a traditional paper-based form, but also a digitized form. These types of digital forms can be read with a computer display, a tablet PC, or other devices. Readers can enjoy various types of digital and physical document forms depending on different reading scenarios (read digital scientific papers on a computer screen, read physical books on a sofa, etc.). It can be said that the powerful features of digital form of document are the reusability and linkability. We can easily copy and paste text or access relevant documents from embedded links. The fact that the usefulness of these various functionalities of digital documents is recognized by many people shows the potential of various types of interaction with documents. Therefore, to explore the potential of new types of document interaction is profoundly important in the context of document analysis. Various types of new interactive documents make use of AR [Ero+08; HB11; QC12]. We propose a new human-document interaction AR system by combining an eye tracking, document retrieval, and see-through HMD technology. The proposed system monitors real-time gaze data of the reader and presents augmentative information of the document in the HMD with respect to his or her attention. That is, when the reader reaches a particular part (word) in a document, the system detects where it is and presents supportive information such as a translation, a glossary, a picture of the article, etc. The proposed system is a new fusion of a document retrieval system with a wearable computing system in an everyday natural reading context.

As discussed in Chapter 2, human attention on document reading has been a main topic in the eye tracking research field over several years. Following the studies in this domain, a number of gaze interaction applications have been developed. For example, in [Bie+10], Biedert et al. present a framework for developing such a gaze-based interaction application for documents on a computer screen. These traditional gaze-based interaction systems typically rely on a stationary (desk mounted) eye tracker. On the other hand, eye tracking devices available today which have become small, light-weight and wearable open up opportunities to extend the scenario to more ubiquitous scenes. Such a development of

eye tracking devices is very fruitful since a lot of people still read a paper type document. Indeed, there are some advantages of a paper form when we consider various types of document-related scenes [Jab13]. I propose a new gaze-based interactive document system that is based on the previous systems for a computer screen but extends them to apply to various forms of document, not only a digital one but also a printed one by using an image analysis approach.

In the following sections, I present the system overview and the individual components for document image retrieval, eye gaze analysis, and HMD-camera calibration. Then, I will discuss the experiments. I conduct a couple of experiments that test the accuracy of document image retrieval with a reading scenario, gaze position mapping on a document, and attended word identification. Furthermore, I also conducted a user study for evaluation of the translation system. In this study, I compare the proposed system with a traditional online translation interface.

5.2.2 Proposed System

An image of the proposed system is shown in Figure 5.14. The core process of the proposed system is comprised by gaze analysis and image-based document retrieval. Using image-based document retrieval, we are able to apply the system both to a printed paper document and to a digital one on a computer screen with the same manner. As a result of retrieval,

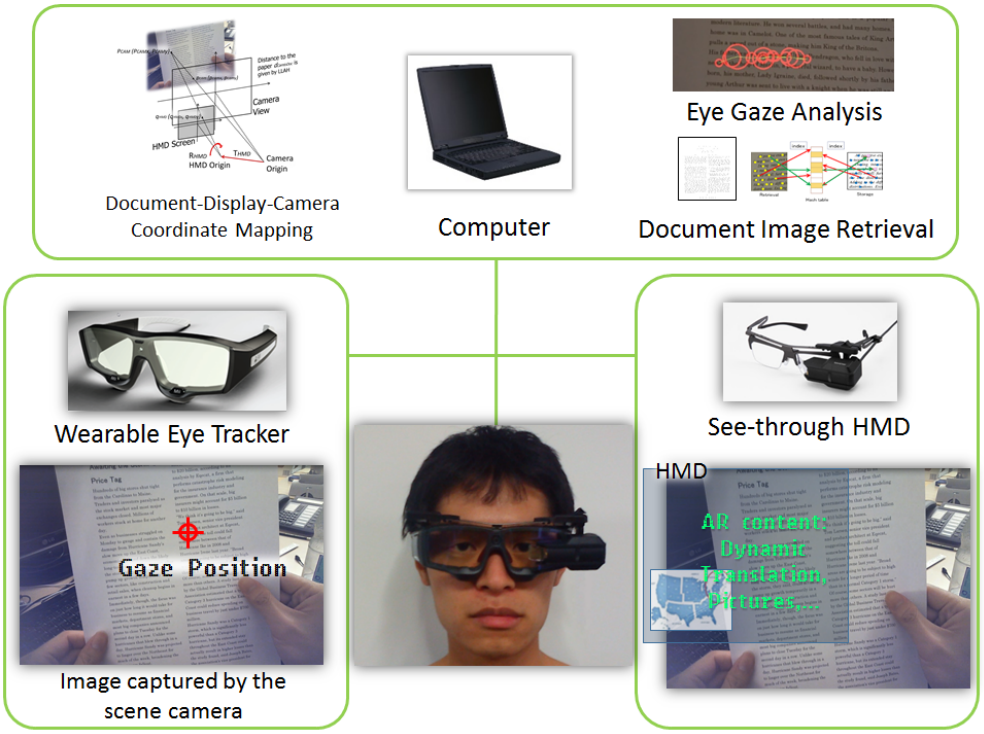


Figure 5.14: Proposed eye gaze-based interactive document system. The reader can get supportive information of the reading document using eye gaze. For the document-display-camera coordinate mapping image (top-left), a large image is presented in Figure 5.21

an identity of the reading document image (page) and its perspective transformation matrix are returned. Once we calculate the geometrical relations between the camera, the HMD and the document, we can map the coordinate of a point from one plane to another. Thus, for example, we can calculate in which position in the HMD the word the user is attending to is located. Based on this coordinate mapping, the system can present information of the attended word (e.g., translation of the word) as an overlay virtual image onto the exact position of the physical document.

Using the proposed system, one can enjoy the benefits of attention analysis during document reading. I present two examples of reading assistance function. One is an annotation presentation system where the user can get the annotations of the last attended key-word. This system analyzes user eye gaze and logs the data of which annotated key-word the user has read. When the user wants to refer to the system for further information of the word he or she read, he or she can get the information by looking at the HMD. The system presents the annotations in the display. The other is a dynamic translation system where the user can look up a translation of word in the document. By calculating the geometrical relations, the system presents a translation on the exact location of the word in the HMD, when the user attentionally gazes on the word.

A workflow of the system is shown in Figure 5.15. The eye tracking server streams online scene video images and gaze data. The document retrieval module returns an ID of the document present in the scene image. A document database must be created in advance for

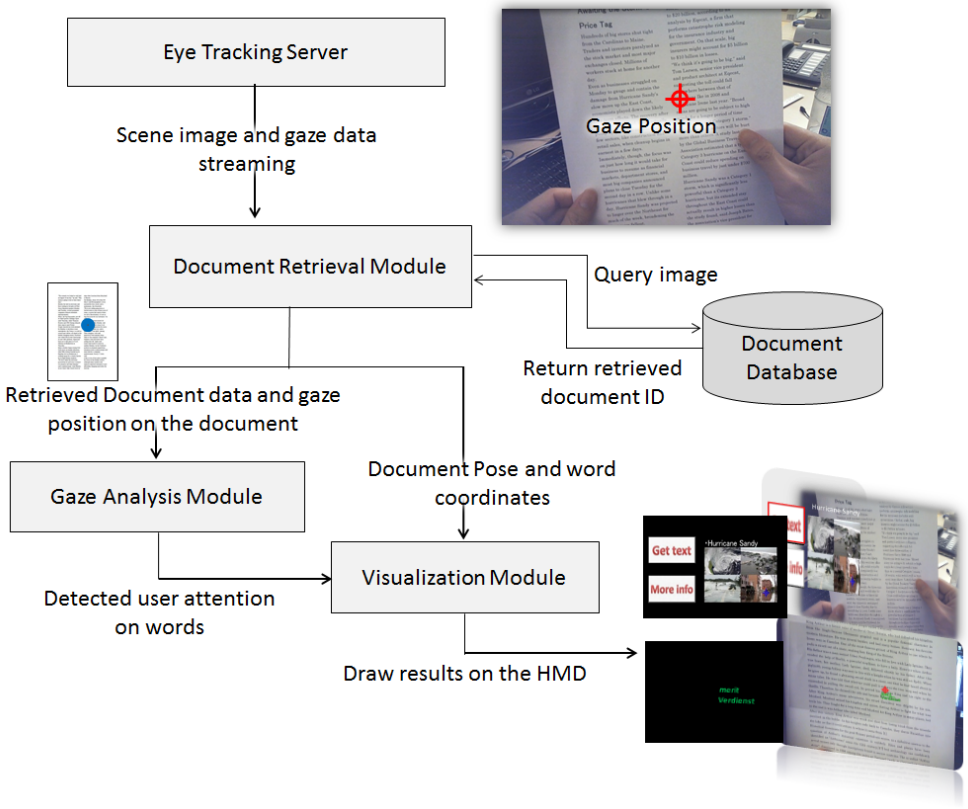


Figure 5.15: Workflow of the eye gaze-based interactive document system.

the retrieval module. From the retrieval result, we can also estimate the document pose in a given image query. Using the homography between the query image and retrieved reference image, we calculate the gaze position on the retrieved document page. Consequently, we extract the word which is the closest to the gaze position. In the gaze analysis module, user attention to words is analyzed, i.e., it checks whether he or she attentionally gazes on a word, reads text naturally, or looks at the HMD (or an icon in the HMD). The visualization module presents the result in the HMD, if any information is requested to present. The following sections describe the process of each module and method.

5.2.2.1 Document Retrieval

We adopt an image based document retrieval method proposed in [Nak+06]. This method, called Locally Likely Arrangement Hashing (LLAH) is robust to perspective distortion of an image and scale-invariant. An overview of the document retrieval method is shown in Figure 5.16. When a scene image is given from the camera, the image is blurred by

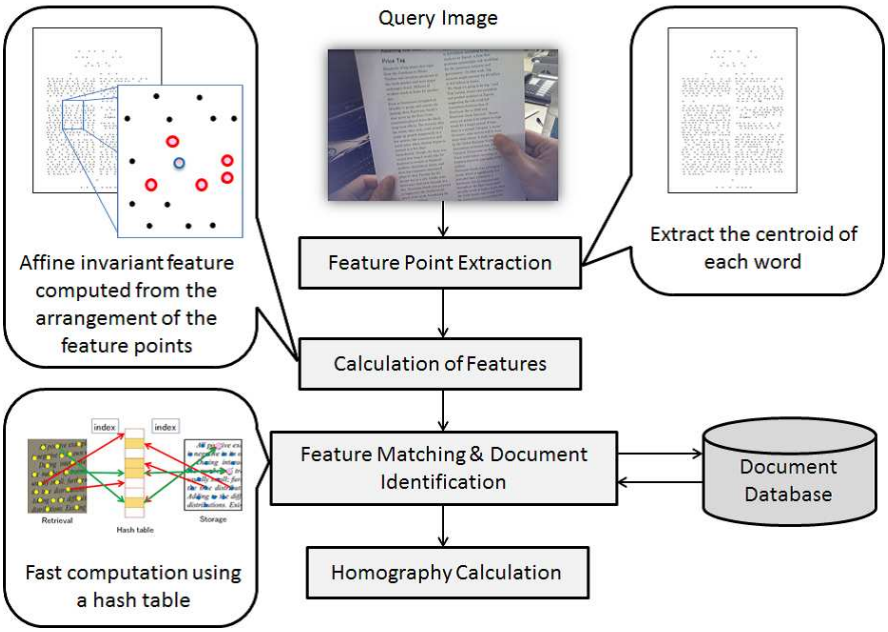


Figure 5.16: Overview of the document retrieval (LLAH) process.

a Gaussian kernel and thresholded adaptively into a binary image in order to detect the centroid of each word region. By changing the size of the Gaussian kernel, we can adjust the blur of image for document retrieval. In the experiment, we seek to find a good kernel size for our scenario where the user has typical distances from his or her eyes to the reading document. From the arrangements of the detected centroids, affine invariant feature vectors are calculated. The recognition process is done by matching the extracted features to the ones from the reference images stored in the database. A hashing technique is used for fast computing. In [Tak+11], it is shown that this method can be extended to deal with no less than 10 million pages.

By matching the features between the scene image and the retrieved database image,

we also calculate the homography between them. Based on this homography, the pose of the document in the scene image is estimated. Furthermore, from the matches between two images, we can also calculate the rotation and transformation matrix of the document image to the scene image, which can be used later for the calibration of the HMD and word position projection. Suppose **H** as the homography of the document image and the scene camera image calculated by LLAH. The given point p_s in a scene camera image is projected to the point on the retrieved document p_d by,

$$p_d = \mathbf{H} \cdot p_s.$$

Thus, the gaze position p_g in a scene image can be projected to the gaze position p_G in the reference document image by $\mathbf{H} \cdot p_g$.

Since we apply an image-based method, we can deal with almost any type of document forms, not only paper document forms but also digital document forms which are typically displayed in a computer screen. This powerful feature enables us to use the same setting without concerning the different types of forms.

5.2.2.2 Gaze Analysis and Attended Word Identification

The gaze analysis module receives the gaze data from the eye tracking server and the result from the document retrieval module. We calculate the gaze position on a document and detect user attention to a particular word.

By projecting the obtained gaze position in the scene image to the retrieved document image, on which word the user is currently fixating is recognized. This temporal fixation on each word is a good indication for the inference of where the user is currently reading. However, as I have already addressed in the previous chapter, if a computer reacts against every single action the user makes, the information overload sometimes would not be acceptable to the user (the *Midas touch* problem). It is very likely that when the user reads the document fluently (without being stuck with any word), the user does not need any supportive information. Hence, instead of providing the information for every temporal fixation on a word, we detect the user attention on a specific word on which the user really gazes, and then use it as a trigger for information provision. Moreover, there is another reason why we adopt this approach for information provision. Even if the calibration of the eye tracker is done properly, human gaze cannot stay at a very static position. Therefore, gaze is always fluttering around the fixation point and an erratic noisy fixation sometimes (or often) occurs, even unconsciously to the user.

We extend the AG detection method presented in Section 4.1. Since the temporal fixation on each word is detected quite robustly, we can simply replace the art objects used there to temporally fixated words in this system. Our attention detection method works as follows: Suppose we have video frames and each video frame F at time t has an identity label $F_L(t) = X$, given by the result of each temporal word identity label X at time t. Starting from $t = 0$, if the given label is X (i.e., $F_L(0) = X$), let $D_X = 1$ and $N_X = 0$, where D_X is the number of frames that have a label X, N_X is the number of frames that have any other label but X. Hence, at time $t = 1$, if $F_L(1) = X$, then $D_X = 2$ and $N_X = 0$. Otherwise, $D_X = 1$ and $N_X = 1$. If D_X reaches to the threshold value T_D, it is detected as an attention to X and N_X is reset to 0, or if N_X reaches to the threshold value T_N before D_X reaches to T_D, it is not detected as an attention, and D_X is reset

to 0. Once an attention to X is detected, the algorithm does not increment D_X but only increments N_X (however it is reset to 0 every time it receives label X). As different point from the previous method in Section 4.1.4.2, if N_X reaches to N_T, it is considered as the end of AG. With this algorithm, by tuning the parameters T_D and T_N, the user could also adjust preferable threshold values for AG detection (e.g., if it is too quick, he or she can set T_D longer).

To cope with a problem of inaccurate eye tracking, we propose a compensatory approach. The proposed attention detection algorithm works reasonably well for words. However, when the eye tracking accuracy is not good enough, it still produces false outputs. It is quite often that the gaze position has a certain offset, even if the calibration process is completed successfully. Sometimes it occurs because the eye tracker is moved accidentally. With a compensatory approach, we regard the other words located nearby the detected word as candidates for the attended word, assuming that an eye tracker might have an error in a certain range. This *word candidates* method returns all words within R pixels from the detected attended word X, which will be entirely presented to the user.

5.2.2.3 Visualization Module

In the visualization module (see the workflow in Figure 5.15), we control which information to be presented in the HMD. As previously mentioned, I propose two different types of user assistance function. The first function is called presentation of attended key-word annotation and the other is dynamic gaze-driven translation. First, I present the process of presentation of attended key-word annotation.

Presentation of Attended Key-word Annotation This function provides the user with annotations of attended key-words. The system analyzes the reader's eye gaze during reading and logs the read words. From the gaze position in the scene image, the system infers whether the reader is reading the document or looking at the HMD (described later). When the reader looks at the HMD, it presents the annotation of the last attended key-word in the HMD.

Annotations can be created using an annotation tool as shown in Figure 5.17. We can annotate words in a document with pictures, text, and videos. Furthermore, we can also add interactive gaze buttons to the annotation. When an annotation is presented to the user, he or she can request for further information using these interactive gaze buttons. In this system, *key-words* refer to the words that have annotations.

Since the HMD is mounted on either side of user perspective as previously shown (Figure 5.14), the user can simply focus on the display spontaneously in order to refer to the system for the annotation of a key-word. Though the display can be peripherally seen during reading, it is not obtrusive to the user because of the transparency. The system detects when the user activity transits from reading the document to looking at the display, by checking if the user gaze is on the display or not. For this detection, the user needs to calibrate the display position with respect to a scene camera image. Four dots are presented in the HMD and the user clicks a mouse button where the position of each dot matches in a calibration window, as shown in Figure 5.18. Knowing the display position in a scene image, the system can infer whether the user is looking at the display or not from the gaze position. Thus, if the user's AG is detected within the display area, it infers that the user is looking at the display. This approach works well because the display can be positioned in a peripheral region of

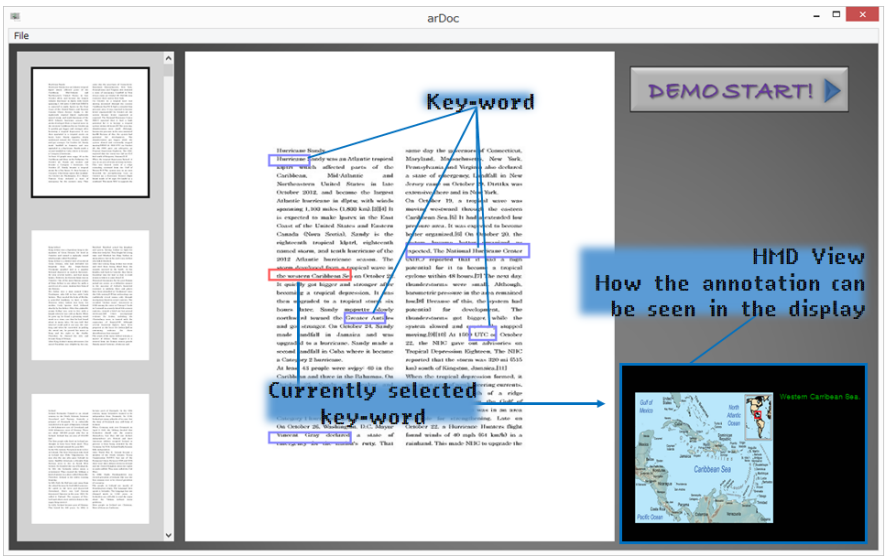


Figure 5.17: Screen shot of the annotation tool. Users can annotate each part (word) of documents with pictures, text, and videos.

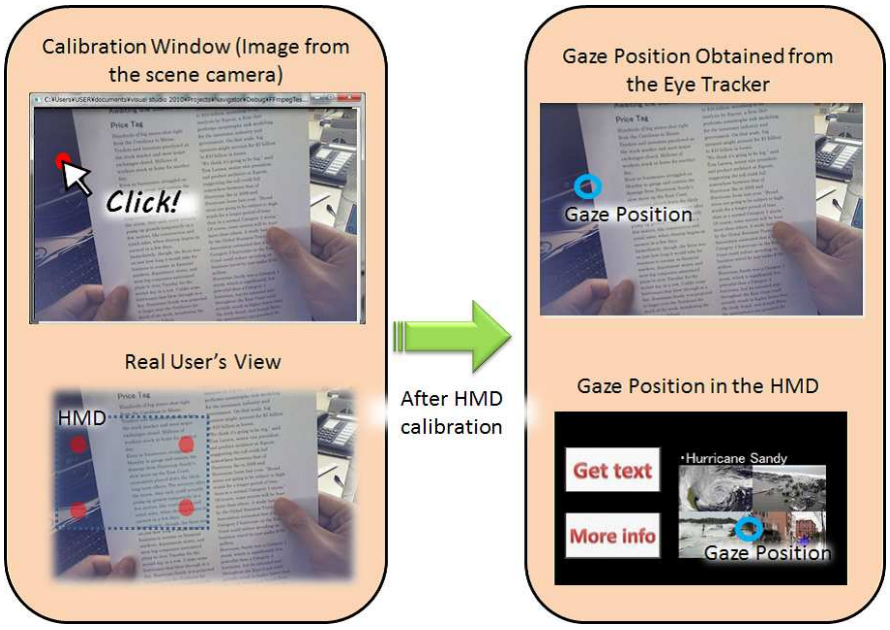


Figure 5.18: Calibration of HMD. The user has to click a computer mouse on four points in the window corresponding to the dots in the HMD view.

the user's view (During reading, it is rare that the user's eye gaze locates in a peripheral region).

When the user's AG on the HMD is detected, the gaze analysis module sends the annotation data of the latest attended key-word to the display visualization module. Then, the annotation is presented to the user using the HMD immediately. A sample view of

Figure 5.19: Sample view of annotation presentation. The red bounding rectangle indicates the user is currently selecting the "Get Text" button.

annotation presentation is shown in Figure 5.19[5]. Interactive gaze buttons can also be selected by AG detection. When a button is selected, it presents further annotations which were previously created by the tool.

Dynamic Translation Another user assistance function is dynamic gaze-driven translation. The system calculates the pose of the document and presents a translation of word in the HMD on the exact position of physical document dynamically. As a preprocess of this function, the system needs to calibrate the HMD-camera positions (which is different from the previous HMD-calibration in a scene camera image). After calibration, one can project a point on the scene camera to a point on the HMD.

The calibration is done by using a calibration paper as shown in Figure 5.20. The calibration rectangle (the green rectangle in the figure) is presented in the HMD and the wearer of the system moves the paper so that the green cross-hair on the paper and the green rectangle lines in the HMD exactly overlay. By this means, we assume that the origin of the HMD is placed vertical to the paper with distance $d_{HMDtoDoc}$, as shown in the figure. We include the calibration paper in the document retrieval database. Thus, the paper can also be retrieved. Based on feature matching using LLAH, the extrinsic parameters (the rotation matrix R_{doc} and the transformation matrix T_{doc}) between the reference document image and the scene camera image and the homography $\mathbf{H}$ are calculated. Hence, the world coordinate of a point of the document paper P_W (the origin is the camera origin) is given by:

$$P_W = \mathrm{R}_{doc} \cdot \mathrm{T}_{doc} \cdot \alpha \begin{pmatrix} p_{camx} \\ p_{camy} \\ 0 \\ 1 \end{pmatrix}, \tag{5.1}$$

where p_{camx} and p_{camy} are respectively the x and y coordinate of point in the camera (see Figure 5.21) and α is a scale factor for a scene camera image. Likewise, P_W is transferred

[5]In order to provide the user with a option, on which eye front the HMD is mounted (left or right), the visualization can be flipped depending on the HMD position.

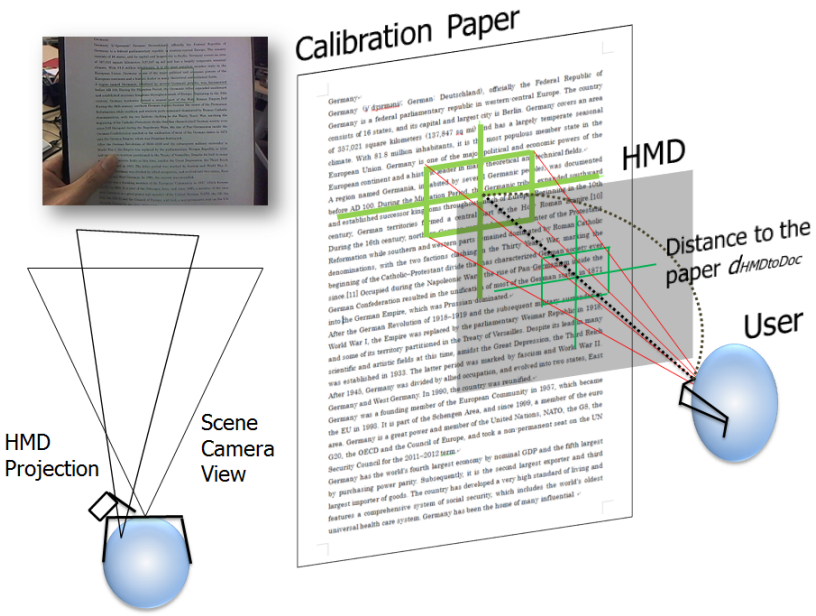


Figure 5.20: Calibration paper and HMD calibration process. The user moves the head or paper so that the rectangle image overlays the physical view. The HMD screen must be parallel to the calibration paper.

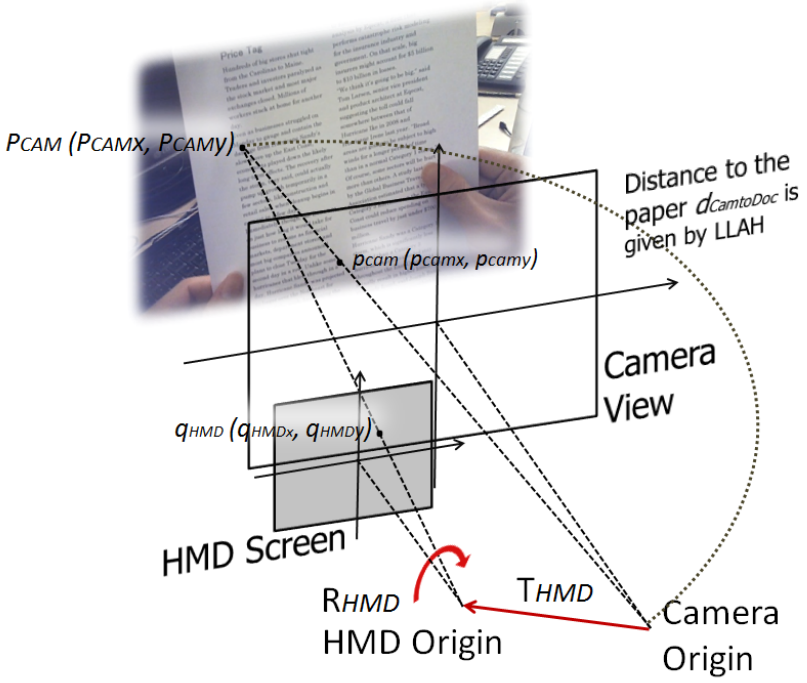


Figure 5.21: Point projection to the HMD screen.

to the HMD space by:

$$P_{HMD} = \mathrm{R}_{HMD} \cdot \mathrm{T}_{HMD} \cdot P_W. \tag{5.2}$$

In the calibration process, we have to calculate R_{HMD} and T_{HMD}. Since the calibration paper is parallel to the HMD plane, the rotation of the HMD R_{HMD} is the reverse of the document rotation R_{doc}, thus $\mathrm{R}_{HMD} = \mathrm{R}_{doc}^T$. Given the correspondence between the world coordinate of the cross-hair point $P_{CAM} = (P_{CAMx}, P_{CAMy}, P_{CAMz})$ (By LLAH, the point of the crosshair on the calibration paper is known) and the cross-hair point in the HMD $Q_{HMD} = (0, 0, d_{HMDtoDoc})$ (the crosshair is the center in the HMD screen), the position of the HMD from the camera $D = (D_x, D_y, D_z)$ is derived as follows:

$$D = \begin{pmatrix} D_x \\ D_y \\ D_z \\ 1 \end{pmatrix} = -\mathrm{R}_{doc} \cdot \begin{pmatrix} 0 \\ 0 \\ d_{HMDtoDoc} \\ 1 \end{pmatrix} + \begin{pmatrix} P_{CAMx} \\ P_{CAMy} \\ P_{CAMz} \\ 1 \end{pmatrix}. \tag{5.3}$$

Thus, the extrinsic parameters of HMD R_{HMD} and T_{HMD} are calculated (T_{HMD} is given by $\mathrm{T}_{HMD} = (ID)$, where I is the identity matrix of size 3.) and a point of the document is projected to the point of the HMD $q_{HMD} = (q_{HMDx}, q_{HMDy})$ by equation 5.2 as shown in Figure 5.21.

Because the HMD position needs to be adjusted for an individual user for visibility, the calibration (the calculation of the extrinsic parameters R_{HMD} and T_{HMD}) must be done individually. Therefore, the calibration is required for each user before using the system. Our approach, however, only requires the user to move the calibration rectangle in the HMD to the proper position and press a key (to start the calculation). Thus, the calibration can be done quickly and easily.

When we project all words in a document as well as translations, the user sees the full information overlay as shown in Figure 5.22 (rightmost) and 5.23. Here, we show the example of German-to-English translation. As we can see in those images, the information would be too obtrusive to read text for those who do not want all translations. Therefore, full information overlay might be unacceptable to many users. In the experiment, we compare four different visualization modes to test usability of each of them. With this comparison, we evaluate the benefit of gaze-driven translation system which only presents the result when the user attends to a word. If we only display one word translation, the HMD view looks like the leftmost image. As we described in the previous subsection, the user can tune the threshold parameters in order to get the information at a preferable moment. If the system detects the user's attention to a word, the translation is immediately presented. The second-left image shows the HMD view with the word candidates approach (refer back to Section 5.2.2.2). When the calibration is not accurate enough to obtain the correct attended word, this option provides with more possible words, presumably including the correct attended word. For comparison, we also propose another visualization mode which does not overlay the information onto the document paper. Instead, the information is displayed on the side of the HMD screen, as shown in Figure 5.22 (second-right). With this option, when the attention to a word is detected, the translation is presented in the middle of either side (left or right, depending on the HMD position). When a new attended word is presented, old ones in the HMD are shifted up, thus a log of previously attended words are also kept visible.

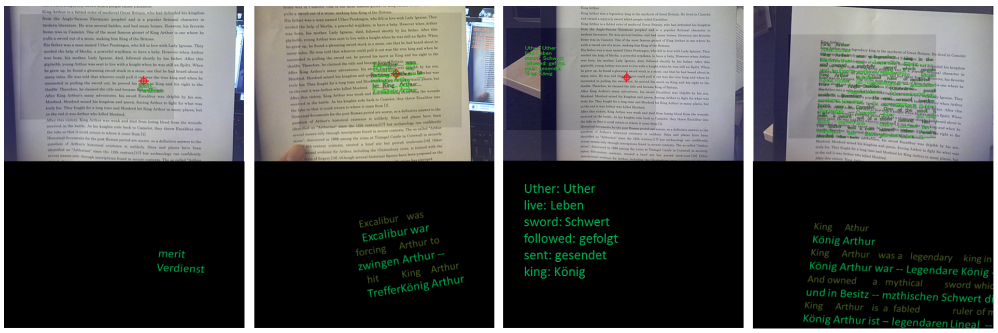


Figure 5.22: Example images of respective visualization modes (The bottom part of each image shows an image in the HMD). Left to right: Single word visualization, Word candidates visualization, No dynamic overlay, and visualization of all words. Note that the font size is not the actual size in the HMD. The text is shown with a larger size here. German translations are presented as supportive information in this example.

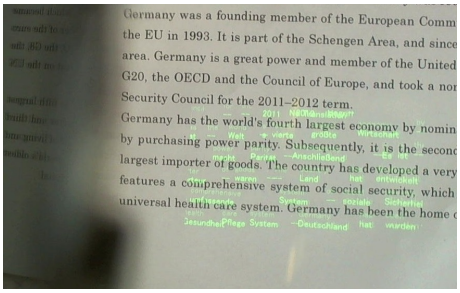


Figure 5.23: Sample image of HMD view in real. Full translations are shown in the display.

5.2.3 Experiments

5.2.3.1 Document Retrieval Performance Using a Wearable Camera

First, we evaluate the performance of the document retrieval method, testing the system with a natural reading scenario. In this experiment, we show how well the LLAH-based document retrieval can perform with different reading conditions.

Table 5.5 and 5.6 show the accuracies of document retrieval using the scene camera of ETG when the distance and the angle to the document (A4 printout, single column) are changed as shown in Figure 5.24, respectively. In Table 5.5, the accuracies when the size of Gaussian kernel is changed are also shown. We built a database of 1010 document images[6] (pages). The accuracy is calculated as the ratio of the number of correctly retrieved document images to the number of retrieval processes for 30 seconds (one document retrieval process takes less than 40 msec, i.e. faster than the scene camera capturing speed with 25 fps). From the results, we can conclude that this method works quite well when the distance from the document to the camera is ranged from 15 cm to 40 cm and the angle

[6]This method is however able to extend the database size even larger than 10 million images without significant performance loss [Tak+11].

Table 5.5: Document retrieval accuracy for each distance of the camera to the printout.

Distance [cm]	15.0	20.0	25.0	30.0
Accuracy (Gaussian kernel size 3×3) [%]	99.41	100.0	100.0	100.0
Accuracy (Gaussian kernel size 7×7) [%]	100.0	100.0	100.0	99.59

Distance [cm]	35.0	40.0	45.0	50.0
Accuracy (Gaussian kernel size 3×3) [%]	100.0	98.78	74.37	23.36
Accuracy (Gaussian kernel size 7×7) [%]	47.65	0.0	0.0	0.0

Table 5.6: Document retrieval accuracy for each angle of the camera to the printout.

Angle [°]	45	60	75	90
Accuracy [%]	18.44	100.0	100.0	100.0

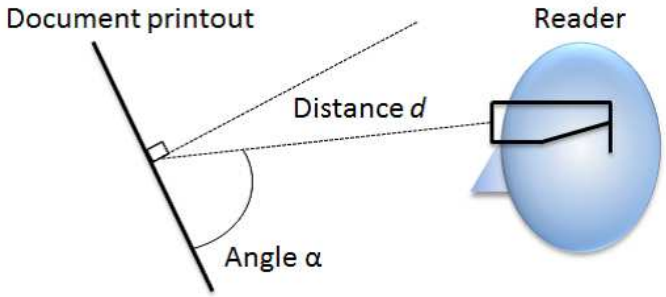


Figure 5.24: Distance d and angle α to the document.

is ranged from $60°$ to $120°$. The performance drops significantly when the distance or the angle is not in these ranges. The results imply that the system allows the user to move the head position quite freely.

We then also asked 10 persons to wear the eye tracker and read one page of printed document naturally, i.e., they read the document paper as they usually do. The retrieval accuracy and natural angle and distance for each participant are summarized in Table 5.7 (with Gaussian kernel size 7×7). The system worked perfectly for almost all persons. We also asked the participants to read documents (PDFs) displayed on a computer screen (Samsung SyncMaster 24 inches) with the same size as the document printout. As shown in the table, this method can also deal with digital documents on a computer screen. In addition to those experiments, we also confirmed the accuracy changes only very little with a different brightness value of the screen. The results showed the feasibility of the state-of-the-art document retrieval method (LLAH) in a reading context using a wearable camera for a handheld document printout as well as a digital document displayed on a computer screen.

Table 5.7: Document retrieval accuracy for different person with different distances, angles, and document forms.

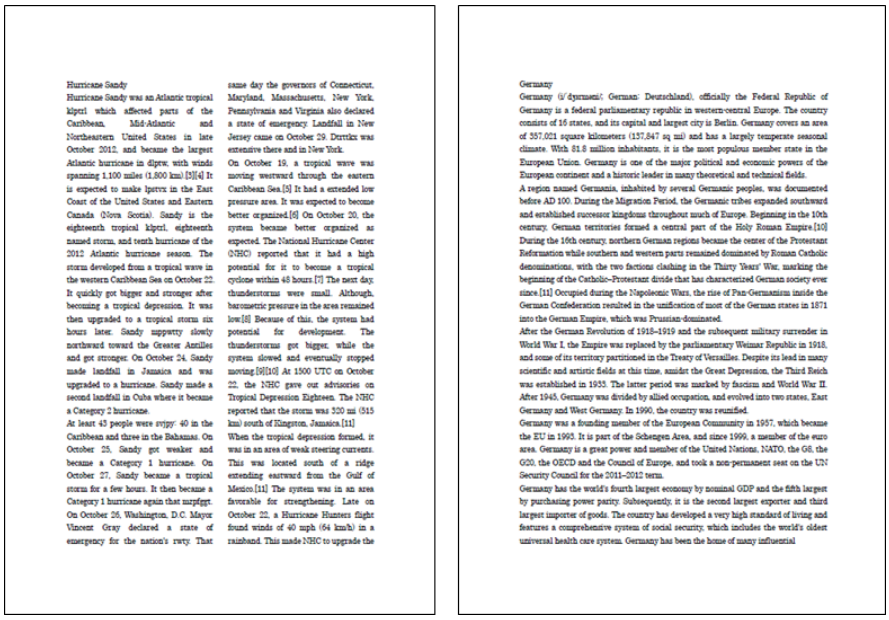
Test person	A	B	C	D	E
Natural distance [cm]	35.0	40.0	40.0	30.0	30.0
Natural angle [°]	60	60	60	65	65
Accuracy (with a handheld printout) [%]	100.0	100.0	100.0	100.0	100.0
Accuracy (with a computer screen) [%]	100.0	100.0	100.0	100.0	100.0

Test person	F	G	H	I	J
Natural distance [cm]	35.0	30.0	35.0	40.0	35.0
Natural angle [°]	50	80	50	65	60
Accuracy (with a handheld printout) [%]	97.31	100.0	100.0	99.80	100.0
Accuracy (with a computer screen) [%]	100.0	100.0	99.87	100.0	100.0

5.2.3.2 Eye Tracking and Attended Word Identification

Next, we evaluate the eye tracking performance on reading document and attention detection method.

Eye Tracking on a Document First, we investigate how accurately the proposed method can track the gaze position on a document during reading. We asked 13 users to read a document *Germany* (see Figure 5.25) and recorded the gaze data. Then we checked how


Hurricane Sandy

Hurricane Sandy was an Atlantic tropical kiprtl which affected parts of the Caribbean, Mid-Atlantic and Northeastern United States in late October 2012, and became the largest Atlantic hurricane in dipew, with winds spanning 1,100 miles (1,800 km).[3][4] It is expected to make lporvx in the East Coast of the United States and Eastern Canada (Nova Scotia). Sandy is the eighteenth tropical kiprtl, eighteenth named storm, and tenth hurricane of the 2012 Atlantic hurricane season. The storm developed from a tropical wave in the western Caribbean Sea on October 22. It quickly got bigger and stronger after becoming a tropical depression. It was then upgraded to a tropical storm six hours later. Sandy supwerty slowly northward toward the Greater Antilles and got stronger. On October 24, Sandy made landfall in Jamaica and was upgraded to a hurricane. Sandy made a second landfall in Cuba where it became a Category 2 hurricane.

At least 43 people were svjpy: 40 in the Caribbean and three in the Bahamas. On October 25, Sandy got weaker and became a Category 1 hurricane. On October 27, Sandy became a tropical storm for a few hours. It then became a Category 1 hurricane again that nzpfgpr. On October 26, Washington, D.C. Mayor Vincent Gray declared a state of emergency for the nation's rwty. That same day the governors of Connecticut, Maryland, Massachusetts, New York, Pennsylvania and Virginia also declared a state of emergency. Landfall in New Jersey came on October 29. Dtrrtkx was extensive there and in New York.

On October 19, a tropical wave was moving westward through the eastern Caribbean Sea.[5] It had a extended low pressure area. It was expected to become better organized.[6] On October 20, the system became better organized as expected. The National Hurricane Center (NHC) reported that it had a high potential for it to become a tropical cyclone within 48 hours.[7] The next day, thunderstorms were small. Although, barometric pressure in the area remained low.[8] Because of this, the system had potential for development. The thunderstorms got bigger, while the system slowed and eventually stopped moving.[9][10] At 1500 UTC on October 22, the NHC gave out advisories on Tropical Depression Eighteen. The NHC reported that the storm was 320 mi (515 km) south of Kingston, Jamaica.[11]

When the tropical depression formed, it was in an area of weak steering currents. This was located south of a ridge extending eastward from the Gulf of Mexico.[11] The system was in an area favorable for strengthening. Late on October 22, a Hurricane Hunters flight found winds of 40 mph (64 km/h) in a rainband. This made NHC to upgrade the

Germany

Germany (/ˈdʒɜrməni/; German: Deutschland), officially the Federal Republic of Germany is a federal parliamentary republic in western-central Europe. The country consists of 16 states, and its capital and largest city is Berlin. Germany covers an area of 357,021 square kilometers (137,847 sq mil) and has a largely temperate seasonal climate. With 81.8 million inhabitants, it is the most populous member state in the European Union. Germany is one of the major political and economic powers of the European continent and a historic leader in many theoretical and technical fields.

A region named Germania, inhabited by several Germanic peoples, was documented before AD 100. During the Migration Period, the Germanic tribes expanded southward and established successor kingdoms throughout much of Europe. Beginning in the 10th century, German territories formed a central part of the Holy Roman Empire.[10] During the 16th century, northern German regions became the center of the Protestant Reformation while southern and western parts remained dominated by Roman Catholic denominations, with the two factions clashing in the Thirty Years' War, marking the beginning of the Catholic–Protestant divide that has characterized German society ever since.[11] Occupied during the Napoleonic Wars, the rise of Pan-Germanism inside the German Confederation resulted in the unification of most of the German states in 1871 into the German Empire, which was Prussian-dominated.

After the German Revolution of 1918–1919 and the subsequent military surrender in World War I, the Empire was replaced by the parliamentary Weimar Republic in 1918, and some of its territory partitioned in the Treaty of Versailles. Despite its lead in many scientific and artistic fields at this time, amidst the Great Depression, the Third Reich was established in 1933. The latter period was marked by fascism and World War II. After 1945, Germany was divided by allied occupation, and evolved into two states, East Germany and West Germany. In 1990, the country was reunified.

Germany was a founding member of the European Community in 1957, which became the EU in 1993. It is part of the Schengen Area, and since 1999, a member of the euro area. Germany is a great power and member of the United Nations, NATO, the G8, the G20, the OECD and the Council of Europe, and took a non-permanent seat on the UN Security Council for the 2011–2012 term.

Germany has the world's fourth largest economy by nominal GDP and the fifth largest by purchasing power parity. Subsequently, it is the second largest exporter and third largest importer of goods. The country has developed a very high standard of living and features a comprehensive system of social security, which includes the world's oldest universal health care system. Germany has been the home of many influential

Figure 5.25: Samples of documents we used in the experiment. Left: two-column *Hurricane Sandy* and right: single-column *Germany*. *Germany* is also used as the calibration paper.

large the offset of a projected gaze position becomes during the reading. Before recording, the eye tracker is calibrated, thus we assume that the gaze position is correct in the beginning of each recording. Figure 5.26 shows that the offset becomes larger as the user reads the document. There are two possible reasons. One is that the eye tracker moved during the

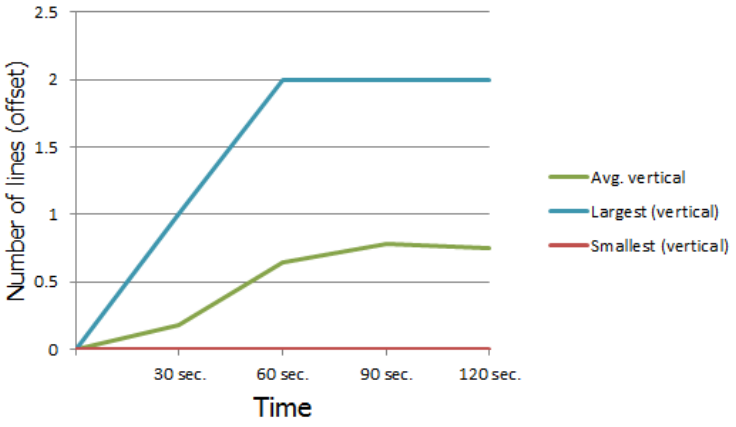


Figure 5.26: The offset of gaze position becomes larger during reading in many users. The graph shows the result from who had the smallest offset (no offset, i.e. 0 line), the largest offset (two lines offset in the end), and the average over 13 persons, respectively. Only vertical offset is shown here but horizontal offset was similar to these results.

reading, i.e., the calibrated parameters for eye tracking that had been set in the beginning became invalid. The other possible reason is that the calibration was not properly done, even if the system succeeded the parameter calculation. It is often the case that it produces a certain offset when the wearer looks at the edges of scene camera image or farther or closer spaces than the calibration plane. Especially in our scenario, an error of 1.0° (SMI ETG's accuracy is 0.5°) makes a big difference. When we have an error of 1.0°, there is one line difference on a document paper with 30 cm distance. Overall, we concluded that state-of-the-art eye tracking devices have a very good performance for some users, but not all. Therefore, we need to consider a compensatory solution for those who the eye tracker does not work perfectly, in order to develop a practical application. In the following experiment, we investigate how well the proposed attended word detection (identification) method performs.

Attended Key-Word Detection Next, we investigated the accuracy of the attended key-word detection, which is used in presentation of attended key-word annotation. In this experiment, we asked 13 persons to participate. They were given one page document (A4 printout with two columns) which was generated from the text of The New York Times online article on October 30, 2012, titled "Awaiting the Storm's Price Tag". We selected seven key-words: *the Carolinas and Maine, Hurricane Sandy, the Northeast, Eqecat, Hurricane Ike, Hurricane Irene*, and *Category*, where annotations were also created. This function logs the key-words that are attended by the reader. Hence, we tested the detection performance on those key-words in this experiment. We asked the participants to read aloud the text and checked if the spoken word matches the detected one for each key-word. The histograms of recall and precision rates of entire test persons are shown in

Figure 5.27. When the test persons had problems in the calibration of the eye tracker,

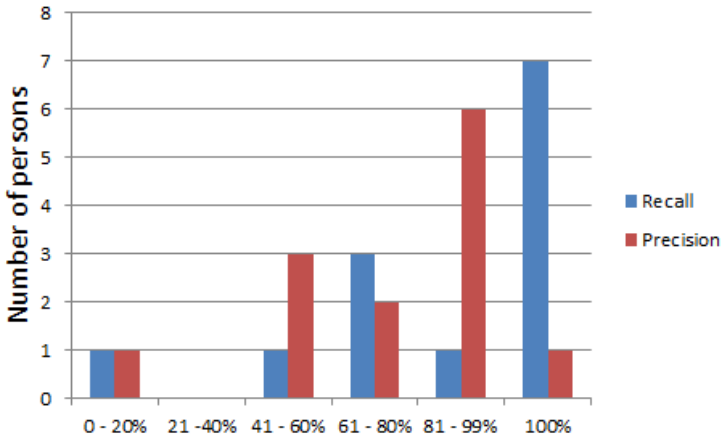


Figure 5.27: Histograms of recall and precision rates of attended word detection for entire test persons.

the performance dropped drastically (0 - 20%). However, for a couple of users the system worked quite well (more than 80% precision and 100% recall rate). For many participants, the precision rates were worse than the recall rates. Typical false positives occurred when two key-words located nearby (in two consecutive lines) as shown in Figure 5.28 and the other key-word is presented mistakenly. However, if two key-words have more than one line

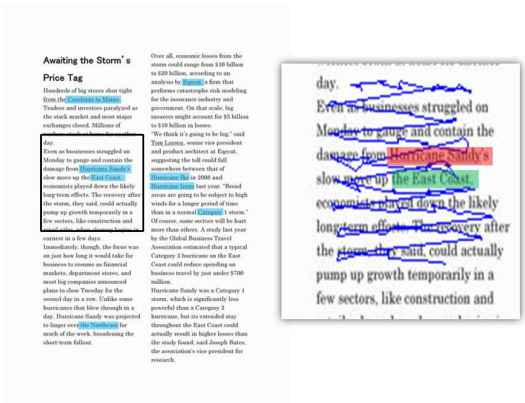


Figure 5.28: Left: The page of document used in the study. Blue rectangles show the positions of key-words in this document. Right: A gaze path image obtained from one of the users. As shown in this image, two words closely located are too hard to distinguish to which word the user attended.

in between, the detection mostly succeeded, i.e., the system can detect attended key-words with approx. 0.4 cm (one line) accuracy, which actually corresponds to the accuracy of ETG (0.4 cm is approx. $40(\text{cm}) \times tan(0.5°)$). From this result, we can observe performance gaps among individuals. However, we could infer that the attended key-word detection approach reasonably performs well although it may depend on the eye tracking calibration. As long as the eye tracking calibration is done properly, the attended key-word can be detected correctly.

Attention Detection and Visualization We also tested the proposed attention detection method for the dynamic translation function. Having an inherent eye tracking error as shown in the prior experiment, we thought it would be meaningful to compare several visualization modes. Therefore, we tested the single word visualization mode and the word candidates mode. The user may be able to find the target word with the candidates mode with a certain case where he or she may not be able to find with the single word mode.

This test consisted of two phases. The first was the tuning phase, where the two attention detection parameters T_D and T_N were adjusted, so that individual user satisfied with the trade-off between the response time latency and false positive results. If T_D increases, the user receives less false positives, though it takes longer for the system to detect it as attention. If T_N increases, the presentation period of the translation is longer, thus the user can view the augmented information longer but it might remain there even if the user quit to view that information. By testing several parameters with some words and asking the user if the trade-off was acceptable for him or her, we tuned these parameters in this phase.

After the tuning, we tested the attention detection and the visualization. In the test, the user was given one page document (*Germany*). We specified eight words that the user had to attend to for triggering the translation presentation. Only when the user reached to the word that we specified, he or she should attend to that. Otherwise, he or she must read the document normally (even if he or she found any unknown word, he should not stop reading). We defined a correct result as a translation presentation that the user could recognize as the correct translation. Thus, the user had to find the right word out of several candidates with the word candidates visualization mode. If the system had false presentation, it was regarded as 'false positive', and if it did not return the result on the eight specified words, it was regarded as 'false negative'. When the presentation was incorrect, the user might try to shift the gaze position slightly in order to catch the correct output (the user could recognize if the output is correct or not since the original text was also shown), which decreases 'false negative' but increases 'false positive', otherwise the user might give up the word and continue reading (the user did not get 'true positive').

The results of the averages of scores are shown in Figure 5.29. A precision score P and a recall score R were calculated as follows respectively: $P = tp/(tp + fp), R = tp/(tp + fn)$, where tp is the number of 'true positive', fp is the number of 'false positive', and fn is the number of 'false negative'. F measure F is calculated as $F = 2 \times P \cdot R/(P + R)$. 12 persons tested the system in this experiment. These results obviously indicate that many users could benefit from the word candidates modes. They sometimes did not find the correct outputs with the single word visualization mode; however, they found ones that they looked for with the word candidates mode. However, for some users, the single word visualization was sufficient to catch a correct output, i.e., the recall and precision scores were 100% for these users.

5.2.3.3 HMD Calibration

Next, we evaluated how easily the HMD calibration method we proposed in Section 5.2.2.3 can be carried out by users. The biggest problem for evaluating HMD calibration is that no other person but the user him or herself can see the screen, thus we, as the third person, cannot judge if the calibration is correct or not. Furthermore, to process the calibration, the user has to confirm that the calibration rectangle in the HMD accurately overlaps the

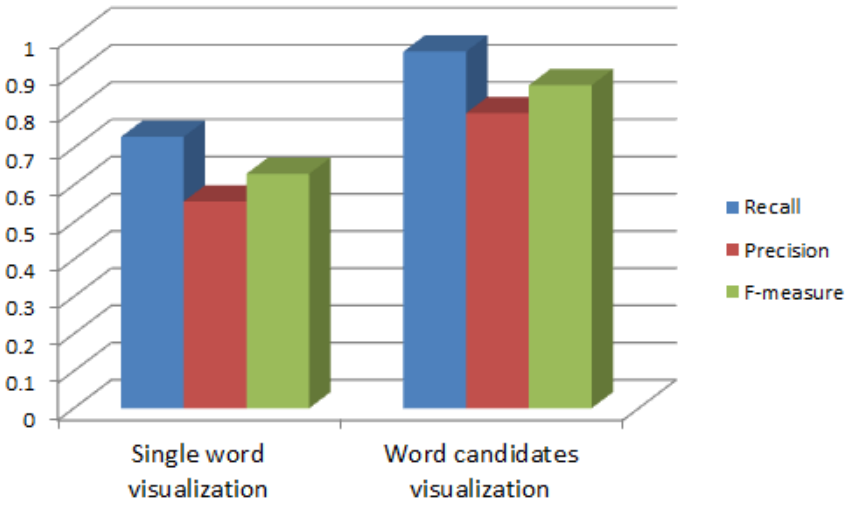


Figure 5.29: Results of the attention detection and the visualization experiment.

one on the document printout. Therefore, it must be noted that the accuracy of the HMD calibration totally depends on the user and the acceptance also differs from one user to another. Although we asked the users to put the HMD screen straight for the calibration, sometimes it was hard due to some reasons; for example, some users could not see the full screen in certain position so they had to move it. In addition, to perceive the straight line as straight was sometimes hard for some users. In order to compensate the calibration error, we asked the user to move the calibration rectangle and recalibrate it, until it fitted to the real view.

Here, we asked to 13 participants to try the HMD calibration and counted how many times they had to repeat the calibration process until the overlay fitted the physical text from the user's view. In Figure 5.30, the histogram of the number of users for each number of calibration processes required is shown. From this figure, we can infer that most of

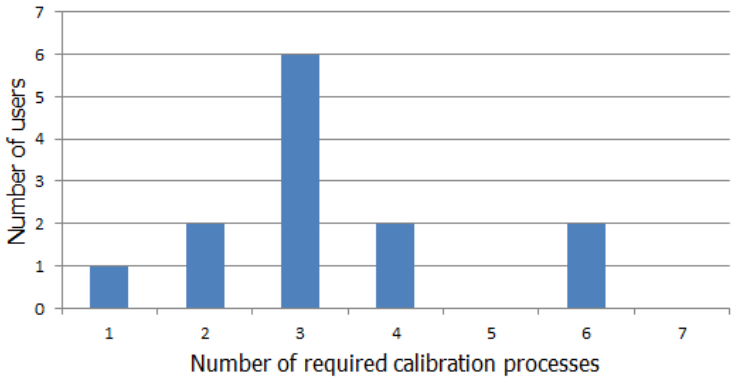


Figure 5.30: Histogram of the number of users for each number of calibration processes required.

the participants (11 of 13) succeeded the calibration within four times of repetition. Each calibration process requires the user to move the calibration paper to the center of the HMD and press a computer keyboard, which takes only few seconds. Thus, an entire calibration process could be completed within one minute for most of the users. Note that all the users were new to the system when they did the experiment, so they were not used to do the calibration process. Some users, who used the system again after this experiment could complete the calibration much less repetition and faster. The result shows that by using the proposed method, we can complete the HMD-camera calibration very quickly and easily.

5.2.3.4 User Study for Dynamic Translation Function

Finally, we conducted a user study to evaluate the benefit of the proposed system in a reading context. In this user study, we focus on the dynamic translation and compare the proposed system with a baseline approach as a translation system.

Set-up As stated in the introduction, this work focuses on a translation use case, where the user benefits from the system when he or she wants to look up a translation of an unknown word. We prepared two types of one page document (A4 and single column) printout written in English. The text was generated from two Wikipedia pages[7]; The title of each document page is *Hurricane Sandy* and *King Arthur* (see Figure 5.25). Translation is done by querying each word to Microsoft Bing Translator. The mother tongues of the participants of these experiments were either Japanese or German; therefore the original text was translated into these two languages. The height of each line was 7 mm and one page contains 36 lines.

Participants of this study were given two documents ("Hurricane Sandy" and "King Arthur") and had to read them and understand the written content. For assisting the reading, they were given two translation systems. As a baseline system, we provided a web-browser-based dictionary (with an online dictionary) to look up unknown words. As the other system, we provided the proposed wearable gaze-driven dynamic translation system.

The task was as follows:

1. The participant was given one page document (either "Hurricane Sandy" or "King Arthur") and a translation system (either the proposed system or the online dictionary).
2. He or she read the paper from the beginning.
3. If he or she found any unknown word, he or she had to check the meaning (translation) by using the given translation system. He or she could not skip it even if the meaning was inferable from the context (As an exception, he or she could skip it when the translation was not found.)
4. The time limit was four minutes. The participant must stop reading even if he or she could not reach the end of the text.
5. The participant answered four questions regarding the given text with a questionnaire sheet. Additionally, we made an interview session to testify if he or she really understood the content.

[7] http://simple.wikipedia.org/, http://en.wikipedia.org/

6. He or she switched the translation system and the document and repeated the process from step 2.

The language skill differs from one participant to another. Thus, in order to let every participant look up the meaning of words several times, we replaced eight words in each text into placeholder words which consisted of random consonants (e.g. "prwpwr", "hrtyttkp", etc.). With these replacements, we simulated a situation where the user does not know the meanings of words. The participant must look up the meanings of these eight placeholder words in any case. We prepared a special dictionary for these words which cannot be found in an ordinary dictionary. Note that they could be found easily since these placeholder words consist of only consonants (a web-browser interface has a tool for search word). The document and the translation combination was swapped during the study (If the first participant used our system for "Hurricane Sandy", then the next one should have use our system for "King Arthur").

Due to a time issue, we could not test the all visualization modes for each participant. Therefore, we asked participants to select one option out of four (*gaze-based single word*, *gaze-based word candidates*, *full translation* and *no dynamic overlay*) visualization modes. Before the study, all visualization modes were presented to the participant and the participant was allowed to select one that worked best (or favorite one). Note that even though this selection was done by first impression of the participant, this was good indication for evaluation of the appreciation of each visualization mode.

Result We asked 13 participants to do the task. Results of different analytic dimensions from all participants are summarized in Figure 5.31. First of all, one can observe there

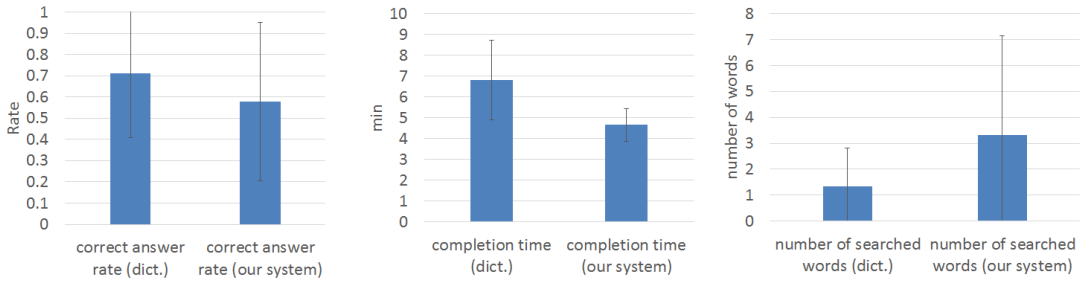


Figure 5.31: User study results of all participant. The error bars represent for the standard deviations. Left: correct answer rate of four questions for two systems (dictionary vs. our system). Middle: Required time for reading completion. Right: number of searched words except for the placeholder words.

was a significant difference between the online dictionary system and the proposed system regarding the time required for completion of reading (the middle graph). Since we set a time limit (four minutes), the required time of those who did not complete reading was estimated by the ratio of the read text length to the entire text length. In this study, the users needed longer time with the ordinary dictionary system because they had to type a word with a keyboard and had to search for the last read line after looking up the translation. Here was a huge advantage of the proposed system; since with the proposed system, the user did not have to switch the reading activity and could keep reading after looking up the translation.

However, if one looks at the left graph, the weakness of our system is somewhat shown. The correct answer rate for the questions (which were asked after each reading task) was slightly lower than the results from the ordinary dictionary system. There were two reasons. One was that when the user could not find the translation using the proposed system (because of calibration errors), i.e., when the system recall was low, the user had no option to look up the translation of the word. One participant could not answer the questions correctly at all with the proposed system because he could hardly get the translation results. A workaround of this problem would be to provide the user with the option to switch the visualization mode, so that the user can find the translation with full translation visualization mode. The other, which was possibly based on a mental reason, was that the users tended to forget the translation they looked up with the proposed system in the end of the reading. It was a quite interesting result, since we could somehow witness the fact that the knowledge acquired quickly can be lost quickly. Although some users could not answer the question correctly with the proposed system, many participants performed better with our system. They could answer the questions correctly with the proposed system because they could reach to the end of the text, while they could not do that with the ordinary dictionary. It is also important to mention that the variances of the correct answer rate were very huge. Thus, no significant difference was observed between two systems regarding the correct answer rate. Since we only asked four questions for each document, it might have not been enough to evaluate the full comprehension of each participant. Nevertheless, the experiment result shows some drawbacks of the proposed system, not only the benefits.

Furthermore, the result shown in the right graph also depicts an interesting fact. Although we asked the participants to look up every word that they do not know, they tended to look up more words with the proposed system. This result indicates that the proposed system is more convenient for looking up the translation during reading. Many participants commented that *"The online dictionaries sometimes kept me from a search because it was very cumbersome"*.

Last but not least, the participant selections of visualization modes were *single word*: 6, *word candidates*: 5, *no dynamic overlay*: 2, and *full presentation*: 0. From this selection result, we can infer that it would be true that if we overlay all information onto the HMD, it is not acceptable for many users. Gaze is one of the options for selecting a region of interest. Thus, it showed high potential in this human-computer interaction context.

5.2.4 Subjective Feedback from Participants

We asked the participants (13 persons) to fill out a questionnaire to collect subjective feedback. The questionnaire and the answers are summarized as follows:

Q1. How do you judge the feasibility of the overall system?

	Number of people
very good	0
good	3
acceptable	7
not good	3
dislike	0
unanswered	2

Q2. Ignoring the hardware: Would you appreciate additional information while reading?

	Number of people
strongly agree	2
agree	8
neither agree nor disagree	3
disagree	0
strongly disagree	0
unanswered	2

Q3. Do you think this system, which does not require things like typing a word, is convenient when reading a document?

	Number of people
strongly agree	4
agree	5
neither agree nor disagree	4
disagree	0
strongly disagree	0
unanswered	2

In summary, they had positive impressions about the proposed system, although they were somewhat pessimistic for the feasibility. For example, regarding the question about the benefit of this type of interaction system: "Would you appreciate additional information when you read a document?", more than 77% of participants agreed. Although we ignored the hardware constraint in the questionnaire to investigate the potential of the system, four participants commented that they disliked the hardware constraints that they had to wear two glasses (the HMD and the eye tracker, even more when they had their own optical glasses). Furthermore, they also reported they sometimes felt stress during the calibration, especially when they had to repeat it. We still need to tackle with these challenges in order to realize a more useful application.

In the user study, we discussed the benefits of the proposed system, as well as some drawbacks. Overall, the participants sometimes had a problem to understand the meaning of a whole sentence even though they got the translation of each word. Most of current online dictionaries also offer the user to input a sentence (however, no participant attempted this in the study when they used the online dictionary, it may be because they knew there was a time limit). It would be another powerful feature if we could recognize whether the user wants to query a word or a sentence to the translation system.

As many participants mentioned, the advantage of the proposed system that they do not have to switch and can keep their reading activity showed the irreplaceable benefit of the combination of AR and the eye tracking system. The use of human gaze as an input may be compared to other input modalities such as speech commands or hand gesture commands. We focus these comparisons as future work.

5.2.5 Conclusion

We presented a system that assists people's reading activity by combining a wearable eye tracker, see-through HMD and image-based document retrieval engine. Furthermore, we showed the feasibility of the state-of-the-art image-based document retrieval method using a wearable camera and proposed a method for detecting an attended word in a document

during reading. The results from the experiments and study showed the real potential of the future of this assistance system in a natural reading context. In the future, we would like to improve the calibration performance and increase the resolution of attended word detection, not only with key-words, but with arbitrary words. In addition, a thorough study for the usability of the system and interface design (HMD) is required.

5.3 Summary

This chapter presented several methods for user-attended VCA for textual content. First, we focused on natural scene text. Two important advantages are shown for a gaze-guided system. First, similar to the object recognition, we can improve the processing time by limiting the image for character recognition. Second, we can infer when the user needs supportive information by analyzing either attentional gaze on text or gaze patters for gesture commands. The experimental results also showed that users can easily get information of natural scene text using AG or gaze gestures.

Then, I focused on eye gaze during reading documents. I presented a method for analyzing eye gaze on document paper using a document retrieval engine. Furthermore, several functions for a reading assistance system are presented. Detecting attentional gaze on particular annotated key-words in a document, the proposed system presents annotations in an HMD. Also, when the reader is stuck in with a word, the system can present virtual text of the translation dynamically at the exact position of the document using an HMD. The user study showed that such dynamic assistance is helpful during reading a document.

In this chapter, I showed that eye gaze is a useful interface for interacting with textual visual content. The systems presented in this chapter demonstrated the feasibilities and benefits of user-attended VCA for scene text and documents.

Chapter 6

Eye Gaze with a See-Through HMD

In the prior chapters, we have seen several methods for user-attended VCA. In the proposed methods, I utilized a see-through HMD for information presentation and gaze-based interactions. This chapter further explores the user interaction with a see-through HMD using eye gaze.

The advantages of a see-through HMD has already been mentioned previously. The transparency of screen allows the wearer to see the physical environment through the display. Therefore, the user can have the physical world and virtual view at the same time. In terms of pervasive computing, this is a strong feature, since we can embed AR content into the physical environments seamlessly.

In the first section (Section 6.1), I present a method for analysis of user attention engagement with a see-through display. Using an eye tracker, one can estimate on which reality plane (virtual or physical) the user is focusing. With this estimation, we infer whether the user's attention is engaged with the display or not. Furthermore, I present methods for analysis of the user's cognitive activities during interaction with the HMD. By combining cognitive activity analysis and attention engagement analysis, I implement proactive user assistance functions for a see-through display.

In Section 6.2, I present an interface for a multi-focal display. We build a prototypical interface for a multi-plane HMD where we assemble three AirScouter displays in an array. Thus, the proposed interface consists of three different display planes. The user can see the virtual environment with multiple focal planes, which in turn, a semi-volumetric display can be realized. In this section, I investigate the feasibility of eye gaze interaction using gaze depth for such a semi-volumetric display.

Each section in this chapter is based on the work presented in [Toy+15; V¨+14] and in [Toy+14b], respectively.

6.1 Attention-Driven HMD Interaction

An issue of a wearable see-through HMD is that images in the virtual screen may obstruct the physical images. When information is presented as a virtual image in the HMD, the background in the physical world becomes invisible. To overcome such a problem, I propose an interface that takes the user's eye gaze into account. By analyzing the user's eye gaze, one can detect the user's attentional engagement with a display and infer which type of cognitive task (e.g., reading text) he or she is involved in. As a result, I implement several novel proactive display functions which support more flexible and seamless human-computer interaction.

6.1.1 Introduction

Recent developments in optical see-through HMD technology have enabled a particular type of AR where virtual information is present in the same line of sight with the physical environment, as shown in Figure 6.1. In this type of AR system, virtual images are superimposed onto the environment directly in a user's field of view, which presents various challenges. Here I address two challenges. The first is that traditional direction-based eye gaze input approaches cannot identify on which reality space (virtual or physical) the user is focused and the second is that the user may be involved in different cognitive tasks when he or she is interacting with augmented content, such as reading text, viewing pictures, or other activities [CM01; CK14; Hen+13].

Though several eye gaze-based interactive AR systems have already been proposed such as [Lee+11], there are still only few interactions available through gaze, and many of them

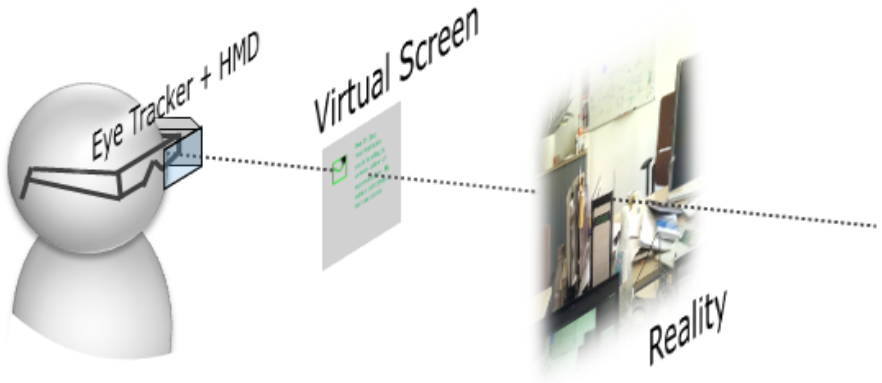


Figure 6.1: AR system with a see-through HMD. Wearing a see-through HMD, the user sees the reality and virtual content on the screen in the same line of sight.

are limited to line of sight (concerning gaze direction only). Usually, line of sight-based techniques have an inevitable drawback when used with optical see-through displays in that they cannot distinguish on which interactive space (virtual or physical) the user is focused. Thus, the system may mistakenly activate AR functionality in the HMD even if the user is not attending to the virtual space, which is often obstructive for the user. Furthermore, it was shown that recognition of user's cognitive state, such as reading [Pre+09; Bie+09], can effectively facilitate proactive assist functions [Sal+04; Bie+10] with ordinary computer screen interactions but still not used in AR wearable displays. I also investigate advantages of cognitive state analysis for interactions with an AR see-through HMD.

In this section, I have two primary goals. The first is to determine if we can extract a user's cognitive state and attention engagement based on eye-gaze alone. Secondly, once we have determined a cognitive state and attention engagement, we want to use this knowledge to facilitate interactive functions which would normally require manual or time consuming interaction to execute. For example, when a user is reading text on a display, he or she should not have to manually scroll down to continue reading. Moreover, if he or she wants to glance at an environmental object, perhaps when at a crosswalk, virtual text should be dimmed for the duration of the glance, and immediately become visible again when the user looks back at the screen. Additionally, the user should be able to continue reading at the exact spot where he or she left off, without losing his or her place.

Proactive user assistance that uses attention engagement and cognitive state analysis include the following practical examples. I focus on the implementation of these proactive system functions which intelligently control HMD screen interfaces using eye gaze input:

Automatic Dim Automatic dim enables the system to control the brightness of the screen intelligently by estimating the user's attention engagement with the display. When the user is attending to the virtual screen, the system increases the brightness and vice versa when not attending. For attention engagement estimation, we propose two methods: gaze depth-based and vestibulo-ocular reflex (VOR)-based (see Section 6.1.2.2).

Eye-con Eye-cons are gaze-selectable icons in the HMD that are activated by attentional user gaze. The previous chapter (see Section 4.1) showed that activation of commands using AG-based approaches are useful, and can even detect fixations with noisy data. We implement this function by detecting attention to icons on the display (Section 6.1.2.2).

Automatic Scroll Due to a narrow field of view, optical see-through displays typically cannot show full texts when the document is large. Therefore, only a portion of the text is present at once. The proposed function can scroll text automatically when the user is reading the text in the display. This way, we can more effectively use display space and the user can read text seamlessly. In Section 6.1.2.3, we propose a method to detect a user's reading state in the display.

Last Read Word Highlighter (Identifier) The user will also frequently need to return his or her focus to reality, to shift attention to an emergency or other distractions for example. It is cumbersome for the user to come back to the place where he or she left off the screen, especially when he or she was reading text. To address this, the system highlights the last read word when the user comes back to the virtual screen, using

the last word read prior to shifting gaze away from the display. A similar function has been implemented with a stationary setup by Biedert et al. [Bie+10]. We extend a framework like this to a mobile scenario. In Section 6.1.2.3, I present tan algorithm for identification of the last word read.

Attentive Notification When the user is engaged with a cognitively intensive task such as reading, notifications by the system such as email alert may be distracting. Based on active/passive cognitive state classification (Section 6.1.2.3), the system predicts an appropriate moment for notification. The system then notifies the user attentively by presenting information only when the user is a passive cognitive state. We propose a method for prediction of a busy cognitive state during display interaction.

The contribution in this section is to propose an intelligent interaction framework for augmented reality, taking the user's attention and cognitive states into account. In experiments and user studies, I investigate robustness of the proposed attention engagement estimation and cognitive state classification approaches, and I evaluate the usability of the proposed proactive system functions.

6.1.2 Approach

Figure 6.2 illustrates a flowchart of the proposed architecture. The entire system is driven by eye gaze-based attention engagement and cognitive state analysis. Consequently, the proposed system interacts with the user intelligently and attentively. In the following subsection, I present methods for eye gaze-based attention analysis (blue boxes in the figure) and proactive user assist functions (red boxes).

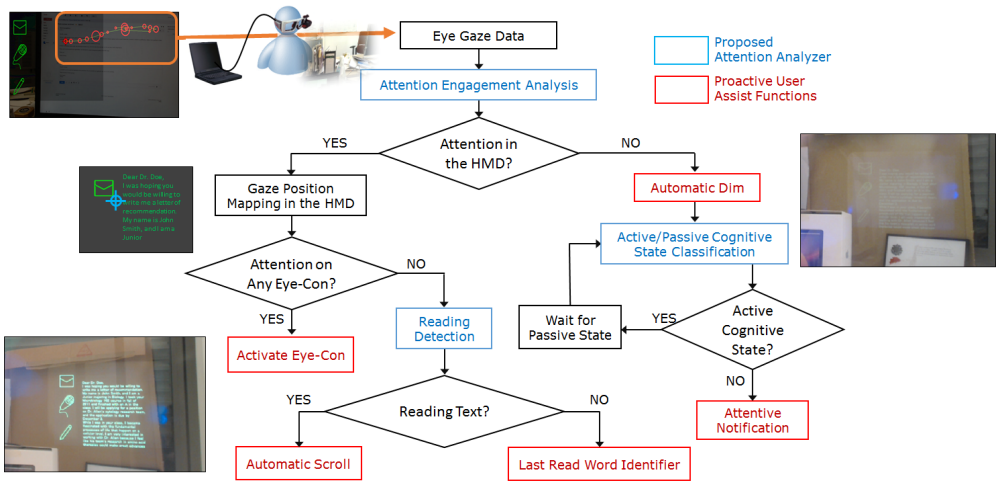


Figure 6.2: Flowchart of the proposed system. Blue boxes are the proposed attention analyzers and red boxes are the proposed proactive functions. From a user's eye, the HMD can be seen as the pictures (left-bottom: bright screen and right: dark screen (dimmed)).

6.1.2.1 Hardware Setup

Similar to the previous chapters, I use the setup (eye-trackable see-through HMD) presented in Section 3.3.3. From the user's view, the HMD virtual screen appears as shown in the bottom-left of Figure 6.2. Typical optical see-through HMDs that are currently available, including the Brother AirScouter, have a quite limited range of field-of-view (22.4° diagonal). To develop an effective interface within a limited HMD screen is a challenging topic for augmented reality.

6.1.2.2 Attention Engagement Analysis

The first challenge is to estimate whether the user's attention is located in a virtual screen or in the physical reality space. We propose two approaches for attention engagement estimation: gaze depth-based method and VOR-based method.

Gaze Depth-based Method When the user's attention is engaged in a virtual display space, the 3D focal depth of eye gaze must be located near the virtual display plane in the space. Based on this observation, we estimate whether the user's attention is engaged with a virtual environment or the physical environment. We calculate the user's gaze depth in the environment using online gaze data from the eye tracker. Since the eye tracker very reliably provides the 2D gaze coordinate in a scene image, we only implement the approach for gaze depth calculation.

First, we extract gaze vectors of both eyes $\mathbf{g}_L$ and $\mathbf{g}_R$ from the each eye camera. Using this data, we get the point of convergence of the two gaze vectors in space as shown in Figure 6.3. Based on this convergent point, we estimate the focal depth value d of the

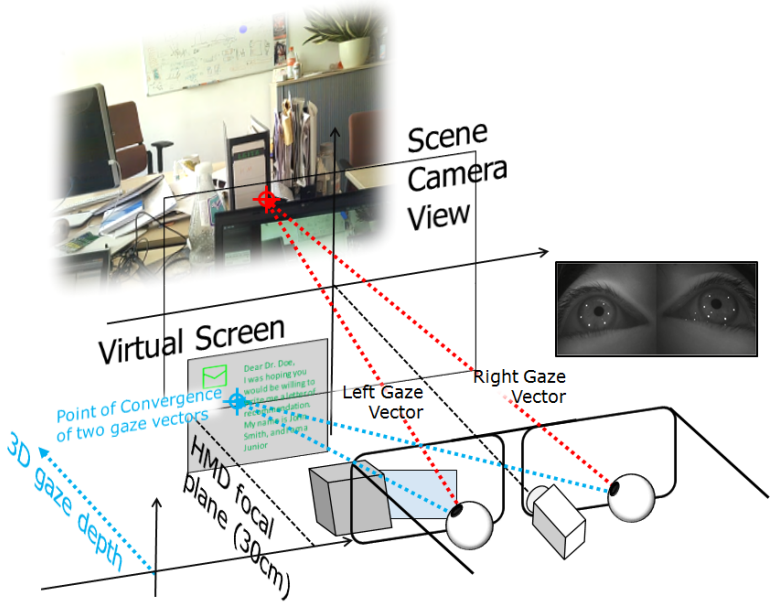


Figure 6.3: Setup of the virtual screen and the physical environment. We detect the user's attention engagement if the gaze depth is near the plane.

user's eye gaze as follows:

$$d = p_z,$$

where p_z is the z-axis value of the intersection point of two gaze vectors $\mathbf{g}_L$ and $\mathbf{g}_R$.

Using the depth value d, we find out whether user attention is located in the HMD or in reality. In Figure 6.3, one can also see an illustration of our HMD and eye tracker setup. Although the focal depth of AirScouter may be adjustable from 0.3 to 10.0 m, in this section, the focal plane of the see-through HMD is fixed to a near distance from the user's eye (30 cm). Hence, if the user's gaze focal depth is near the fixed virtual plane, we can infer that the user is focusing on the display. We estimate the user A is engaged with the virtual screen of the HMD if the calculated depth value d is:

$$-k\sigma_A + d_{AHMD} \leq d \leq k\sigma_A + d_{AHMD},$$

where k is a scale factor, d_{AHMD} is the average depth value when the user A is focusing on the HMD and σ_A is the standard deviation of such depth values. Because scales of vectors from the eye tracker are different for individual users, the resulting depth value is relative for each user. Thus, we need to tune parameters d_{AHMD} and σ_A for individual users. We first collect gaze depth values when a user (in this case, A) is focusing on the HMD. After collecting the depth values, we calculate the average depth d_{AHMD} and the standard deviation σ_A. In the experiment (Section 6.1.3.1), we find an optimal k for the attention estimation (separating virtual and physical).

A disadvantage of this gaze depth-based approach is that if the user focuses on physical objects near the virtual plane, the system cannot tell whether gaze is on the virtual screen or in the physical world. In such a case, we can use the following VOR-based approach.

VOR-Based Approach When a human moves his or her head while fixating on an object in the environment, the eyeballs move opposite direction to the head movement. This type of behavior is called a VOR, which indicates that the human is actually attending to the object in the reality. Recent work by Vidal et al. [Vid+14] showed potential of attention engagement estimation using VOR. When a head movement occurs, eye gaze moves either to the same direction or to another direction, depending on the user's attention engagement with an HMD.

In this work, inspired by [Vid+14], we implement an attention engagement estimation technique using head and eye gaze motion tracking. Vidal et al. used an IMU for head motion tracking. Instead, we use optical-flow, i.e., the pattern of motion of objects and edges between consecutive image frames [BB95], of scene images for head motion tracking to have a more simple setup. We calculate optical flow between two consecutive scene images as shown in Figure 6.4. First, we extract local interest points using the Shi-Tomasi method [ST94]. Then, optical flow of each interest point is calculated by the Lucas-Kanade method [LK81]. We use the methods implemented in the OpenCV C++ library. Finally, we calculate the mean 2D vector S_m of optical flows from all the interest points. The mean vector S_m shows motion of the scene camera (i.e., head motion).

Similarly, we also extract the movement of gaze positions between two consecutive frames. By subtracting the previous gaze coordinate g_p from the current coordinate g_c, we obtain the gaze motion vector g_m. Hence, the difference between the head and gaze motion d_{abs} (which corresponds to the absolute gaze motion in space) is given by,

$$d_{abs} = \mathrm{abs}(S_m - g_m).$$

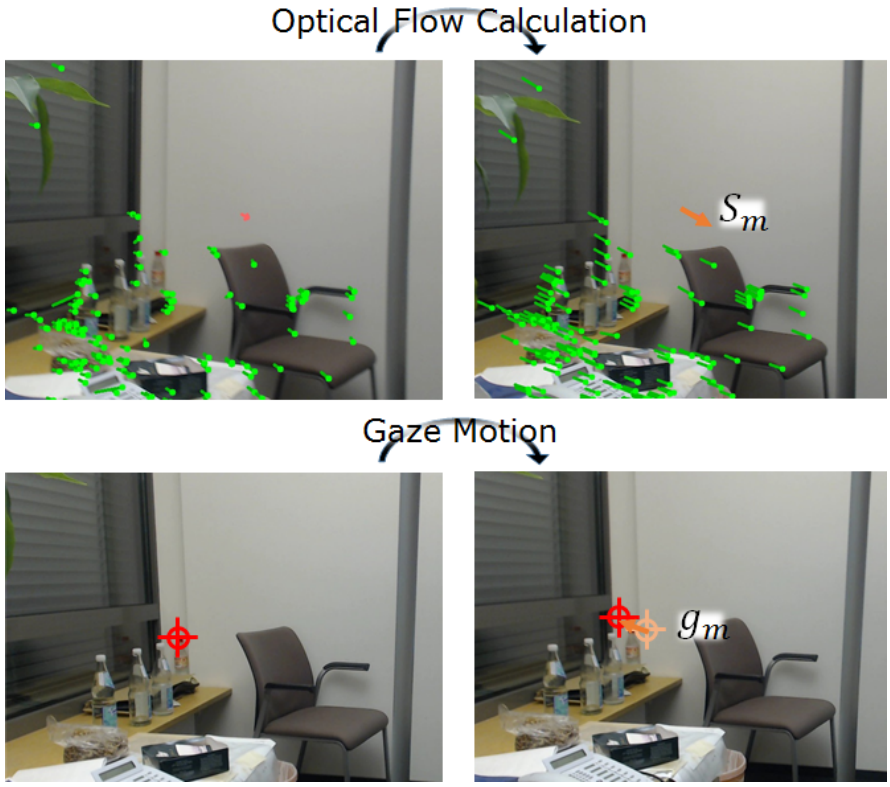


Figure 6.4: Optical flow and gaze motion vector from two consecutive video frames. When the user fixates on an object (a bottle cap), the gaze moves opposite direction to the head motion (the mean optical flow vector).

If d_{abs} is smaller than the threshold value T_d, we assume that the user is focusing on the display, and vice versa. In the experiment, we find an optimal value for T_d.

Using the abovementioned attention engagement approaches (depth-based and VOR-based), we implement an automatic dim function. To allow detection errors of attention engagement estimation which may occur sporadically, the system activates this function if a new engagement state (either virtual or physical) is detected more than N_a times. Thus, if the "attention to display" state lasts more than N_a times, the system increases the display brightness and vice versa for dimming the display.

Mapping Gaze Position in an HMD Using the gaze coordinate in the scene image, we can also calculate the gaze position on the HMD when the user is attending to the virtual space. In this section, we employ a simple mapping method as shown in Figure 6.5. Using the HMD calibration tool introduced in Section 5.2.2.3, the user can calibrate the HMD position with respect to a scene camera image. Because the HMD is positioned in parallel to the scene image, the gaze coordinate in the image is linearly mapped to the position in the HMD ($(x_{hmd}, y_{hmd}) = (ax + \alpha, by + \beta)$, where (x, y) is the gaze coordinate in the scene image, (x_{hmd}, y_{hmd}) is the gaze position in the HMD, and a, b, α, and β are the calibrated values for mapping). Once we get a gaze position in the HMD, we can activate interactive commands using user's eye input in the display. We implement a gaze-selectable user interface, called the *eye-con*, in the HMD. If AG (see Section 4.1) on the eye-con is

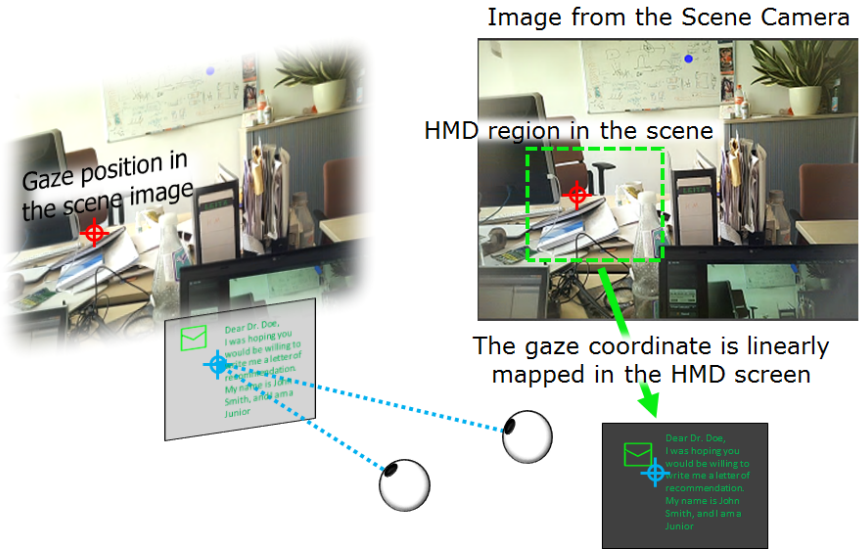


Figure 6.5: Gaze position mapping on an HMD screen. The HMD screen positions in the same line of sight of the green rectangle in the right image. The HMD location must be calibrated in advance, by adjusting mapping parameters manually.

detected, the system activates the respective function (for example, it opens a new email.)

6.1.2.3 Cognitive State Analysis

In addition to the attention engagement analysis, we propose several cognitive state analysis methods which control the graphical interfaces and other proactive assistance functions.

Reading State Detection For reading detection, we extract fixations and saccades from gaze data using a dispersion approach [Bus+08]. Based on the fixations and saccades from a certain time window W, we compute dominant saccadic angular values in W. Suppose

$$S_a = \{s_{a1}, s_{a2}, ..., s_{ai}, ..., s_{an}\}\ (0 < s_{ai} \leq 2\pi)$$

is a set of angular values of the extracted saccades in W. Each angular value s_{ai} is quantized into one of 18 values (i.e., the range of each bin is $\pi/9$; 0 and π is the horizontal angle). Consequently, we generate a histogram of angular values, which has 18 elements. The largest element of the histogram is the most dominant saccadic angle A_1 in W (1st-Sac) and the second largest one is the second most dominant angle A_2 (2nd-Sac). In this chapter, we only focus on English text. Thus, we analyze whether saccadic angular values are horizontal or not. Thus, a reading state of the user R is detected by,

$$R = \begin{cases} 1 & ((-\pi/9 < A_1 < \pi/9 \vee 8\pi/9 < A_1 < 10\pi/9) \\ & \wedge(-\pi/9 < A_2 < \pi/9 \vee 8\pi/9 < A_2 < 10\pi/9) \\ & \wedge(n > min_s)) \\ 0 & (otherwise), \end{cases}$$

where n is the number of extracted saccades and min_s is the minimum number of saccades for a reading state. In this thesis, I set $min_s = 5$ and $W = 2.0$ (sec), which provide a good balance between latency and accuracy.

Using this reading state detection, we propose an automatic text scroll function. While a reading state is being detected, it keeps scrolling the text with a velocity of v_s (pixel per sec). One may consider to change the velocity depending on the reading speed. However, in this thesis, v_s is fixed to 20, which is accepted to several test users in a preliminary test before the experiment conducted in Section 6.1.3.2.

Last Read Word Identifier In the proposed system, we aim for seamless interaction between the virtual environment and the physical environment. Thus, it may often occur that the user suddenly switches his or her focus to the physical environment while reading text in the HMD. Using the automatic dim function, the screen is automatically dimmed when he or she leaves it off. In order to remind the user afterwards where he or she was reading when he or she comes back to the text, the system needs to identify the last word he or she read before leaving off the screen. As shown in Algorithm 1, we propose an algorithm for *last word identification*.

Algorithm 1 Last read word identification algorithm

procedure LASTWORDIDENTIFIER($GazeSequence$)
  $i \leftarrow 0$, $PrevGaze \leftarrow (0, 0)$, $readEnd \leftarrow 0$
  while $CurGaze \leftarrow GazeSequence[i]$ **do**
    if $CurGaze$ is in the Text Area **and**
$PrevGaze.x - CurGaze.x > ReadForwardTh$ **and**
$\mathrm{abs}(PrevGaze.x - CurGaze.x) > LineTh$ **then**
      if $readEnd \neq i - 1$ **then**
        $LastReadGaze \leftarrow CurGaze$
      $readEnd \leftarrow i$
    else if $readEnd \neq i - 1$ **then**
      $LastReadGaze \leftarrow CurGaze$
    $i \leftarrow i + 1$, $PrevGaze \leftarrow CurGaze$
  return $GetNearestWord(LastReadGaze)$

$ReadForwardTh$ and $LineTh$ are threshold values for the horizontal axis and the vertical axis, respectively. The proposed algorithm identifies the latest gaze sample located in the text area during reading. The system stores the identity of the identified last read word when the user leaves off the screen. When the user attends to the virtual screen again, it highlights the word by presenting an underline.

Similar to the reading detection, we validated the threshold parameters in a preliminary experiment: $ReadForwardTh = 50$ and $LineTh = 40$.

Classification of Active/Passive Cognitive State During visual perception, eye movements may involve several cognitive tasks. There are several types of cognitive tasks such as scene memorization, visual search, etc. Visual cognitive tasks often require high mental workload and concentration [Bar+11]. In this chapter, we classify the user cognitive state either passive or active in order to predict appropriate times to interact with the user, focusing on a system notification function. That is, the system proactively interacts with the user when he or she is in a cognitively relaxed state, i.e., a passive cognitive state.

In our preliminary experiment based on a classic study by Gaarder [Gaa66], we found that when people are involved in a non-visual task, such as listening to music or thinking

about something, saccades occur less frequently. Thus, we classify the active/passive state using the frequency of saccades within W (the same as the reading detection). If the frequency of saccades f_s is less than T_f and the state is not *reading*, we classify the state as cognitively passive (We set $T_f = 10$).

As a prototypical application, we implement an interactive function that notifies the user when the user is in a passive state. Using this interaction technique, we present virtual information to the user at unobtrusive times. Note that this function may also be used while the user is interacting with an HMD, though in the experiment (Section 6.1.3.4) we test the proposed method with a task only in the physical world.

6.1.3 Experiments

For evaluation of the proposed attention engagement and cognitive analysis methods and proactive assistance functions, we conducted three different experiments. A total of 13 participants participated in the experiments. First, we conducted an experiment for evaluation of attention engagement estimation accuracy and the ability of users to focus on different planes (virtual and physical). Second, we evaluated accuracy of reading detection and identification of the last read word with reading tasks in the HMD using actual emails. Finally, we conducted a user study for active/passive cognitive state classification with respect to attentive notification functions. In this experiment, we asked the participants to evaluate whether such cognitive state classification based attentive notification is actually beneficial or not.

6.1.3.1 Attention Engagement Analysis

In this experiment, we investigated the following three aspects:

1. attention engagement estimation accuracy of the proposed methods (both depth-based and VOR-based methods),
2. ability of users to switch focus between different focal planes (virtual and physical, and different target distances in the reality), and
3. ability of users to switch focus with varying display brightness and UI settings.

As stated above, we also examined whether users were able to focus on different planes when conditions were changed. In particular, we tested the user's ability to focus on the virtual plane when the screen was dark (such as shown in Figure 6.2), since this is a crucial point for automatic dim function and the user must eventually refocus on the darkened screen.

Depth-based Method: Setup First, we evaluated the depth-based method. As shown in Figure 6.6, the participant sat on a chair and wore the eye-tracking HMD. Facing the participant, two pieces of paper (focus targets) were hung at 1 meter and 3 meter distances from the eyes. We then recorded gaze data when he or she was focusing on the physical (paper) or the virtual (HMD) target with multiple settings. We had three display settings – A: bright text and icon, B: dark text and icon, and C: icon only. For each setting, we had two distance options – a: physical far target (3 m) and b: physical near target (1 m). We repeated the following session three times for each distance option (a or b) and each

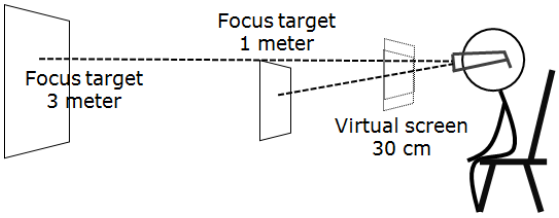


Figure 6.6: Experimental setup in the gaze depth-based attention estimation experiment.

display setting (A, B, or C). *The participant focused on the virtual screen for 5 sec. (one recording).* $\rightarrow$ *After 5 sec., we requested him or her to switch the focus to the physical target.* $\rightarrow$ *He or she focused on the physical target for 5 sec.* $\rightarrow$ *We again requested him or her to switch focus back to the virtual screen.*

When the participant finished display setting A (A-a $\times$ 3 and A-b $\times$ 3), we dimmed the display (change to B). After completing the same steps with the B setting (B-a and B-b), we also tested with icon only UI setting (C-a and C-b), where only the *email* icon was shown in the HMD (see Figure 6.2). During the experiment, the participant kept his or her head fixed and the HMD position was also kept fixed in the same line of sight as the physical target from the eye. We recorded eye gaze data during the experiment. Consequently, we had gaze recordings from 3 different display settings (A, B, and C) $\times$ 2 distance options (a and b) $\times$ 3 sessions of *focus virtual* and *focus physical* (total $3 \times 2 \times 3 \times 2 = 36$ recordings) from one participant. We compared the icon only setting with the text and icon setting, to see whether users have difficulty for focusing on the display when multiple items are present in the HMD.

Depth-based method: Results First, we calculated the depth value for each frame using the recorded data. On average, we had approximately 200 depth values for each recording (5 sec). In Figure 6.7 we show the averages and standard deviations (SDs) of estimated depth values on two representative participants for each condition (display setting, distance option, and virtual (HMD) or physical (1m/3m)). In this figure, we show the case with most easily separable values (P1: participant 1) and the case that was the most difficult to distinguish (P2: participant 2). With the easily separable values, we can clearly see that the averages are similar within each plane (HMD, 1 m, and 3 m). The focal plane is easily separable between HMD, 1 m, and 3 m in this case. However, with the data from P2, it is hard to separate between 1m and HMD (b), though it seems possible between {HMD, 1 m} vs. 3 m. From this analysis, we found three things. First, the granularity of gaze depth estimation is dependent on the user. For some users, we can separate between 1m and HMD but for some other users we cannot. Second, separation between far (3 m) and near (approx. until 1 m) was still possible for all participants. Third, the value may not match to the actual distance. It depends on the user. Therefore, we need tuning of d_{AHMD} and σ_A for each user.

Subsequently, we tested the proposed gaze depth-based attention engagement method using the recorded data. First, we conducted the following test using all recorded data. We divided one recording into two parts. One was used for parameter validation (to calculate d_{HMD} and σ), whereas the other part was used as test data, i.e., if we had 200 depth values, 100 values were used for validation and 100 values were used for test. We computed the accuracy of each engagement estimation: test with A-a (3 m vs. HMD), test with A-b (1 m

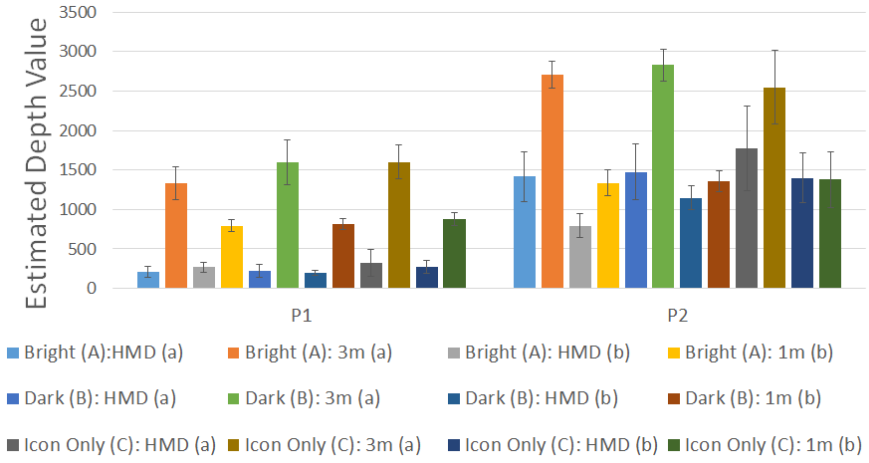


Figure 6.7: Averages and SDs (represented by an error bar) of estimated depth values for each recording from two representative participants (easy separable and hard to separate). Bars of odd columns are in virtual (HMD) whereas even ones are in the reality (3m and 1m). We can see that the estimated depths from P1 are easily separable between virtual and physical. Note that the unit is supposed to be centimeters but actual calculated values do not necessarily match the units.

vs. HMD), and so on, checking whether the system can correctly predict physical or virtual. From several initial tests with other data, we found that $k = 1$ (for $k\sigma$) yields the best accuracy for estimation. For most participants (9 of 13), the average accuracy of attention engagement estimation exceeds 90% on this overall test. However, for some participants, it was difficult to estimate attention engagement (two of them had only 70%) because of noisy gaze data. A possible reason for this is that the eye tracker accidentally moved or the participant had a problem focusing on the virtual screen. Two of the participants wore normal eyeglasses (with corrective lenses) and both of them had low accuracy. With our eye tracker (SMI ETG), such errors are inevitable since the accuracy with corrective lenses is not guaranteed. However, for most of the participants, it achieved high accuracies. For the automatic dim function, 90% is quite good since sporadic errors can be ignored because it needs N_a consecutive frames for the activation.

To analyze the difference in different display settings, we compare the estimation accuracy between A, B, and C. In Figure 6.8 (left), we compare the average and SD of the accuracies. To see if the accuracies have significant differences, we also performed analysis of variance (ANOVA) test [RG89]. The figure shows that the average accuracy decreases slightly when we use dark text and icon or an icon only setting. However, the result from a one-way ANOVA test showed the p-value of these sets is 0.100, which shows that there may be a slight difference, but we cannot conclude that there is a statistically significant difference. Also, the ANOVA on different distance options (3m-HMD (a) vs. 1m-HMD (b)) showed a large p-value (middle in Figure 6.8 $p = 0.66$). A large p-value does not necessarily mean that there is no impact. However, from these results one can infer that the three different display settings (bright, dark, and icon only) and two different distances may not influence the accuracy of attention engagement estimation so much. Therefore, we consider that users can focus on a dark screen as less effort as a bright screen, which is important for

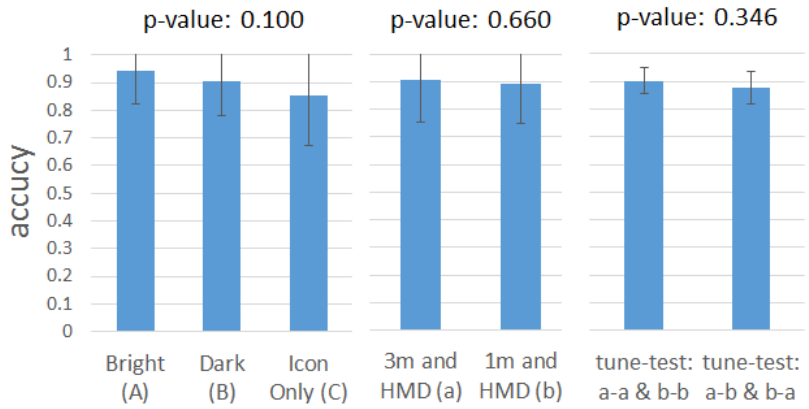


Figure 6.8: Comparison of estimation accuracy between different screen settings (left): A, B, and C, different distance options of physical target (middle): a and b, and using different data for parameter tuning (right): tuned by a and b, and then vice versa. There is no significant difference between the middle and right. In the left, there are slight differences but they are not statistically significant (p-value is higher than 0.05).

the automatic dim function where the user has to refocus on a dark screen. Furthermore, to investigate whether parameter tuning (validation) is feasible regardless of physical target distance, we compare the following. On the right of the figure, we show the comparison of recordings from different distance options (a or b) for the validation (tuning) and test. The left bar is the baseline, where the same distance option was used for tuning and testing. The right shows the average accuracy when we use a for tuning and b for testing and vice versa. They show that the accuracies were similar. Thus, one can infer that the parameters are properly calculable whichever distance we have for the physical background.

Evaluation of VOR-Based Approach We recorded data while the participant moved his or her head with focusing on either the HMD or the 1 m target, similar to the previous experiment. We again recorded 5 sec of gaze data for both virtual and physical and repeat this session three times. Then, based on gaze motion g_m and the calculated optical flow S_m, we computed d_{abs} for each frame. Using a separated validation data, we optimized $T_d = 25$ (pixel) (see Section 6.1.2.2).

For the test of the VOR method, we calculated the accuracy of attention engagement estimation. If $d_{abs} < T_d$ and the participant was focusing on the HMD, the estimation was regarded as correct. For the overall average of all participants, we achieved an accuracy of 89.8% ($SD = 8.0\%$). Compared to the depth-based method, this method did not have such large differences between participants. This result shows the potential of the proposed method, though it can only be used when the user's head moves. However, we may combine the VOR-based approach with the depth-based approach for further improvement of the attention engagement analysis.

6.1.3.2 Reading Detection and Last Read Word Identification

In the second experiment, we evaluated the accuracy of the reading state detection and the last read word identification method. We recorded gaze data while the participants

read two emails using the HMD and applied the reading detection and the last read word identification.

Setup The participant first put on the eye-tracking HMD. We next asked him or her to read emails in the HMD. He or she was told to open an email by gazing at the email icon (using eye-con) and read to the end of the email. We had two emails: *A* and *B*. Since the entire emails could not be displayed on the HMD screen, they were separated into several pages. *A* contains 5 pages (277 words) whereas *B* contains 4 pages (245 words). To go to the next or previous page, the user gazed at the forward or backward eye-con. For this experiment, we only used static (non-scrolling) text.

Before starting the experiments, participants were given several minutes for testing and interacting with the interfaces, including eye-cons. All participants were able to grasp the functionality and could use them appropriately within approximately one minute of first use. The HMD position was calibrated accurately in advance to allow the eye-cons to work properly. For activating the eye-con, the user needed to gaze at the icon approx. for 3 seconds[1].

We recorded the gaze data during reading of the email and tested the reading detection and last read word identification. For the reading detection test, we had two types of gaze tasks: reading text or gazing at eye-cons. The proposed method must correctly identify when the user reads text in the HMD. For last word identification, the user verbally signaled when he or she had reached the last word in the email. We then checked if the gaze position identified as the last reading gaze was actually near the last word.

Reading Detection Result First, we discuss the accuracy of reading detection. After recording the email reading data, we labeled the gaze classes in the recordings for test. When the participant was reading text, *read text* was used, and when gazing at eye-cons, *gaze at eye-cons* was used. The first second in each class was dismissed since there is inherent latency during a transition of gaze state. On average, we had approximately 646 gaze samples for *read text* and 252 samples for *gaze at eye-cons* from each participant. We calculated the accuracy as the ratio of samples correctly classified to total samples.

On average over all participants, the accuracy of reading detection was 78.6% (the SD was 12.9%). This result is not perfect; however, as one can see in the high SD, there was again a large gap between the participants. For three participants, the accuracy was more than 90%, while one participant only had 51%. Similar to the depth analysis, this gap is mainly because of eye tracking instability. One can improve the accuracy if we increase the window size W more than 2.0. However, in order to keep a quick response of the proactive function, this value provides a good balance.

Last Read Word Identifier Result Next, we evaluated the last read word identification method. We calculated the distance from the last read word predicted by the system to the actual last word in the email. The distance was 42.11 pixels on average (the SD was 32.60), which is fairly good if we consider the HMD resolution (800x600). In this experiment, the text area was 450x500. Thus, if the area has 10 text lines, 42.11 approximately corresponds to one line. Also, the field-of-view angle of the HMD is 22.4° (diagonal). Therefore, the resulting offset in angle is only approximately 1.0°. It is actually inevitable to have certain

[1]We set the dwell-time parameter to a slightly longer duration in order to avoid false positives as much as possible.

offset distance because eye tracking is also not perfectly accurate (accuracy is 0.5° with ETG). We found that if we manually correct for eye tracking offset, the predicted last word mostly matches to the actual last word. Thus, the offset was mainly because of the eye tracking.

Figure 6.9 shows an example of the results where the system successfully identifies the last read word even though the last gaze sample in the recording sequence was moved away from the last word. The white dot represents the detected last gaze position in reading state, whereas the pink dot represents the actual last gaze in the recorded sequence. The figure also shows the gaze positions (red dots) when the participant gazed at the eye-con. The offset is not large. Thus, one can infer that the eye-con activations performed reasonably well.

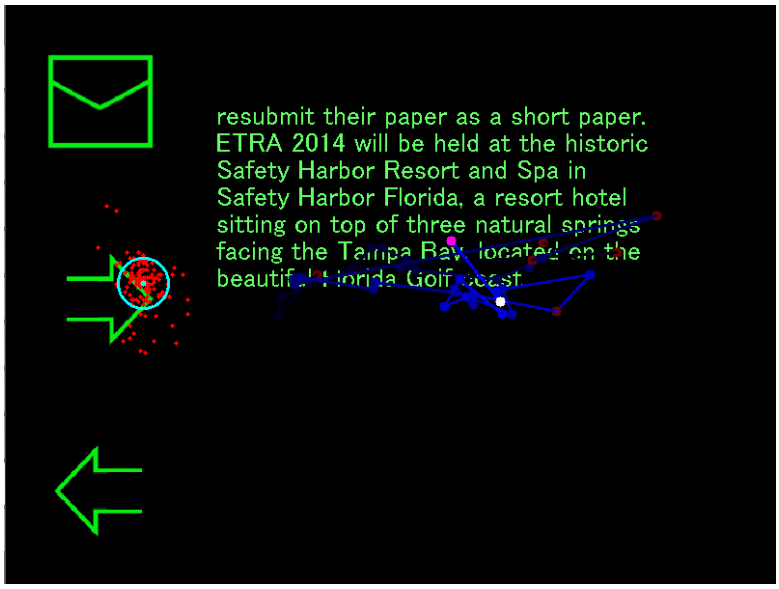


Figure 6.9: Example of last read word identification: The last gaze position (pink) and the detected last read position (white). Red dots are the gaze locations when the user gazed on the eye-con (next page) and blue dots (connected by blue lines) are the ones during reading. The gaze position moved after reading, (pink) but the system successfully detected a point near the last read word (white).

6.1.3.3 Evaluation of Automatic Scrolling Function

Next, we compared the proposed proactive scrolling function with static text from a reading efficiency point of view. The same participants from the previous experiment also read two other emails using the auto scroll function. Using this function, the user did not need to gaze at eye-cons to switch the pages. Thus, he or she could keep reading text. Figure 6.10 shows the averages of time required for reading completion in each email. The email *A* and *B* are from the previous experiment and *C* and *D* are from this experiment (the numbers of words were *C*: 268 and *D*: 239). Having dwell-time for activation of each eye-con, the two interfaces (auto scroll and static text) had similar reading time. It becomes shorter when we subtract the time needed for eye-con activation. One of the reasons that auto-scroll required more time for reading is that the velocity of the scroll was fixed in this experiment. Each

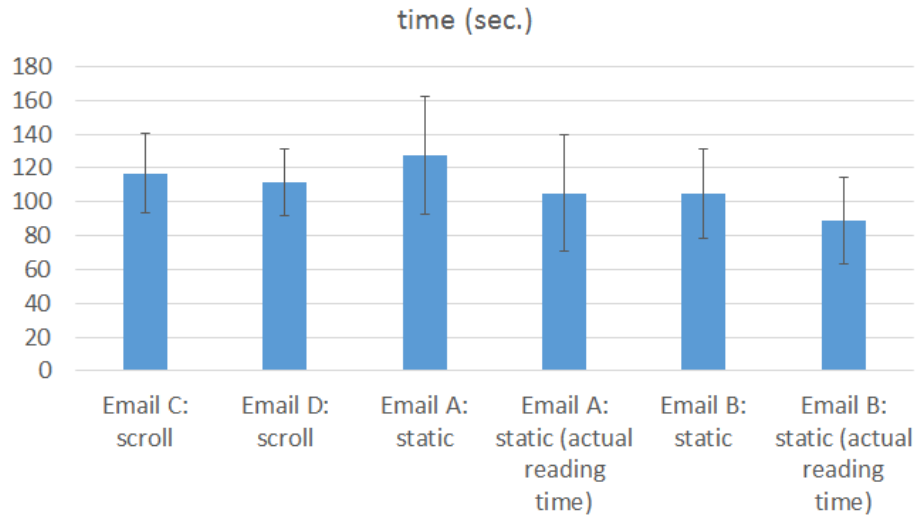


Figure 6.10: Average time required for complete reading text in each email. "Actual reading time" was calculated by subtracting the time for activating eye-cons from the time required in the interface B.

user should have different reading speed. Some participants mentioned that the scroll speed was too fast whereas some other participants mentioned that it was too slow. Therefore, by adapting the scroll speed for each user, the reading process should become much more efficient.

Also, we have asked about preference regarding these two types of interfaces. A majority of the participants (8 of 12) commented that they preferred the automatic scroll to the eye-con page switch because they could read text continuously without having to pause. When users fixate on eye-cons, the process of reading and understanding text is interrupted. Because of this, users often feel stress. However, on the other hand, some participants disliked the automatic scroll function because it was hard for them to read moving text.

Overall, the scroll did not show the efficiency in reading completion time. However, many participants appreciated the function.

6.1.3.4 Passive/Active Cognitive State Classification

In the third experiment, we conducted a user study to evaluate passive/active cognitive state classification with respect to the proposed attentive notification function. The proposed method infers an appropriate moment for a notification by analyzing eye gaze movements. The system notifies the user only when he or she is not involved in active cognitive tasks (i.e., passive cognitive state).

We run the system when the participant was performing office tasks (read an email with a normal computer screen and reply it). After this, we asked the participants for subjective feedback. As mentioned previously, the tasks here were done on a physical device (a laptop) that is not in the virtual display, but the participant still received notifications in the HMD (virtual)[2].

[2]Although we selected an office scenario in this experiment, the proposed system can be applied to other

Setup The participant sat at a desk wearing the apparatus. He or she read an email with a web browser on a normal computer screen (of a laptop on the desk) and replied the email. During the task, four notifications (Figure 6.11 shows an example) were presented in the HMD. The participant performed the task two times with different notification timing inference methods: 1) predefined timing 2) attentive notification (notifies when the cognitive state is passive).

Figure 6.11: Alert in the attentive notification experiment.

We did not explain the methods to the participants in advance. Instead, we only told them that we would test two different inference techniques. After the task with two techniques, we asked the participants if they noticed the difference and which they found more distractive.

Result 12 participants took part in this experiment. For half of the participants, the attentive method presented notifications after reading the emails and while thinking and typing a reply email. Three participants never received notifications because the system could not detect any passive cognitive state. The remaining three participants received notifications while reading an email. The method with predefined timing presented notifications both during reading and typing.

Those who did not receive notifications during reading noticed the difference between the two techniques. Overall, seven participants preferred the attentive notifications, four participants had no preference, and one participant preferred the predefined one (who received the notification during reading). In summary, many participants found that receiving notifications while reading an email was very distracting. Two participants who never received the notifications also commented that it would have been distracting if they had received notifications when reading.

Although this study shows the potential of the proposed notification technique, when we design an actual system, we should take into account how urgent notification is for inferring the right timing for notification. Furthermore, the proposed method has the risk that a user may never receive any notification. However, the results from this study suggest that to notify the user when he or she is not in a cognitively busy state is beneficial.

mobile scenarios. Therefore, the alert was presented in the HMD not in the laptop screen.

6.1.4 Discussion

The depth-based approach showed its robustness for attention engagement estimation. One drawback is as previously mentioned, this approach cannot distinguish if it is physical or not when the user focuses on objects near the virtual plane. Ideas to overcome this problem are to combine the depth-based approach with the VOR-based method or to analyze gaze positions with respect to visual content in the reality and in the virtual space. For example, if the user attentionally gazes at a particular location where an icon exists in the virtual screen, he or she is more likely engaged with the virtual space. Similar to the depth-based one, the VOR-based approach also showed its feasibility when user's head movements are detected. Our experimental results also showed that without using any motion measurement devices such as an IMU, we can track VOR based on optical flow calculation. Another important finding is that the users can actually focus on a dark screen successfully, even though it is less visible than the bright one. Some experiment participants mentioned that they found the dark screen was more practical when they needed to focus on the physical environment, which shows a dimmed screen is actually useful for scene perception when wearing a see-through display.

Overall, the participants commented that they had very positive impressions regarding the system, including the proposed proactive user assistance functions. We asked the impressions on *automatic scroll*, *eye-cons*, and *attentive notifications*. Although adaptation of reading speed for each user is strongly required, many users enjoyed reading text with the automatic scrolling function. The biggest benefit for users is that they can seamlessly continue reading. This technique may also be used for other mixed reality scenarios where text is overlaid onto physical object planes such as walls or desktop surfaces. In order to maximize reading capabilities in a limited space, detection of reading state and proactive assistance such as scroll can be used effectively.

Participants also mentioned that it took a long time to activate eye-cons, but this was likely because we set a long duration for activation time in order to avoid false positives of eye-con activation in the experiment. We plan another experiment to evaluate accuracy and optimal duration for eye-con activation, and will take trade-offs between false-alarm rate and usability into account. HMD position can also be automatically calibrated by detecting the user's static fixations on a particular position, which often indicates focus on eye-con.

In the user study for active/passive cognitive state classification, we tested the proposed *attentive notification* function in an office work scenario. The principle idea of attentive notification is that a human might wait for the other person when he or she is involved in a busy task, such as reading an email. The proposed system can mimic such human behavior, by analyzing cognitive states of the user. In the future, we will conduct further analysis regarding different cognitive states that users may have during interaction with various AR systems. According to a state of the user, more proactive assistance functions could be added. For instance, we can guide users in a visual search task [Los+14], if we detect a user's visual search activity.

6.1.5 Conclusion

I presented a novel eye gaze-based interactive system which analyzes user's attention engagement and cognitive states and accordingly control interfaces in an HMD. Using such eye gaze analysis, the proposed system provides the user with various proactive assist func-

tions such as automatic dim, automatic text scroll, eye-cons, last read word highlighter, and attentive notification.

Experimental results showed that attention engagement can be estimated well, and a reading state in the HMD can also be detected, keeping a quick response. Lastly, the screen UI setting did not influence the user ability to focus on different planes in virtual and physical. The feedback from participants showed the benefits of the proposed functions.

The future of mixed and augmented reality must be well-blended with human attentional behaviors. Computer systems should consequently utilize attention-awareness to assist the users more attentively and intelligently.

6.2 Gaze Depth for a Multi-focal Plane HMD

In the previous section, I presented a method for interaction with a see-through HMD separating the user attention engagement between physical and virtual. Particularly, gaze vectors were utilized to calculate the location of the user's attention – whether it is located on the virtual plane or in the physical environment. In this section, I extend this attention location estimation. I explore the potential of further interaction techniques for multiple focal planes using multiple HMDs. Calculating the gaze depth of the user, we can infer which virtual plane the user is interacting with. This section presents a novel interface using gaze depth-based multi-focal plane HMD interaction.

6.2.1 Introduction

Safely interacting with wearable displays has been a rising concern since lighter, more portable form factors have become available [Hor11]. Information can now be constantly displayed in a user's field of view, which can cause distractions and can require users to interact in ways that may be tiring or unnatural [Orl+13b; Woo+12]. For example, text displayed on a sidewalk in front of a user should only be visible when the user wants to read it. Examples of text overlaid onto potentially dangerous locations are shown in Figure 6.12. If the user glances down at his or her watch, or looks out for oncoming traffic, virtual text should be removed from his or her field of view as quickly as possible to prevent interference. In common approaches, users typically need to press a button or perform some sort of physical action on the device in order to close or manipulate content. Not only does this

Figure 6.12: Images taken through an optical see-through HMD showing virtual content overlaid onto oncoming traffic (left) and onto other pedestrians in a user's path (right).

take time, but it may be distracting and dangerous, especially in mobile situations. We seek to reduce or completely eliminate this manual interaction in order to improve mobility and safety.

As a solution, we propose a combination of eye tracking with a multi-focal plane HMD. Currently, there is a lack of methods developed to improve the safety of monoscopic HMDs. By taking advantage of users' natural tendencies to focus on objects of attention at different depths, we can reduce the need for physical button presses or other manual interaction. Though eye gaze has often been proposed as a form of interaction, most gaze based methods only show the direction a user is looking, but not whether the user is focusing on a more distant object in the same line of sight, for example a real car versus a virtual e-mail. This is where focal depth becomes very useful, since we can calculate whether the user is looking at content on the display or at a hazardous object in his or her environment. This depth can then be used to automatically dim or close distracting content. In addition to automatic content dimming or closing, once a user looks back into the display, he or she should be able to quickly continue viewing content uninhibited. Instead of having to find and press a button or remove a touch-screen device from a pocket, focus can again be used to re-engage an active window. Users then have a more intuitive and robust interface for interacting with virtual content. Recent developments in display technology show that multi-focal plane displays will soon be commercially available, making this interface relevant to both current and future HMD systems [Ure+11].

To some extent, this sort of interaction has been previously tested with stereoscopic 3D displays that sit in a static position away from the user's face [Kim+11; Kwo+06]. To expand this research into the mobile domain, we use a glasses-type eye tracking interface, and combine it with a prototype HMD containing focal planes in the near, mid and far field (approximately 30 cm, 1 m, and 2 m+, respectively). Using this setup, we asked 14 individuals to participate in an experiment testing focus based interaction, and measured the variance of their eye convergence at each focal depth. Our experiments (Section 6.2.4) show that convergence for nearly all users can be used to accurately select objects on any of the 3 focal planes, and that certain depth cues such as blur do not have an effect on the physical focal tendencies of the eye.

6.2.2 Prior Work

State of the art research can be subdivided into three related areas: 1) the construction and prototyping of multi-focal plane displays, 2) the study of depth perception and depth cues related to virtual objects, and 3) the study of eye gaze for interaction with virtual objects. Our prototype, interaction framework, and experiments draw from elements of each of these areas of study.

6.2.2.1 Multi-focal Plane Displays

Since the advent of the HMD, accurate image reproduction has been a goal of HMD research. Focus in particular is difficult to reproduce since the focal depth of an HMD must be at a variable distance depending on the eye's focal point. Though not a wearable device, one of the first attempts to solve this problem was by producing a volumetric display with 20 focal planes in 2003 [Sul04]. Akeley et al., produced a similar display with 3 focal planes, and conducted a study on user perceptions of objects with different depth cues [Ake+04].

Another display by Schowengerdt and Seibel was designed to allow for dynamic shifts in both accommodation and vergence [SS04]. In 2008, a similar prototype with 4 different focal depths was developed by Kim et al., and accommodation results were measured using an artificial eye composed of a pin-hole and multiple lenses [Kim+08]. A more recent HMD type prototype display was developed using liquid lenses, providing addressable focal planes from as close as 8 diopters[3] to infinity as well as variable focal depth. One of the most recent attempts at creating a mutli-focal plane display was by Maimone et al., but displaying content correctly in different planes is computationally intensive [MF13]. For reference, Figure 6.13 shows views through our HMD, the display designed by Kim et al., and a simulation of accommodation effects for the multi-focal plane display design proposed by Liu et al.

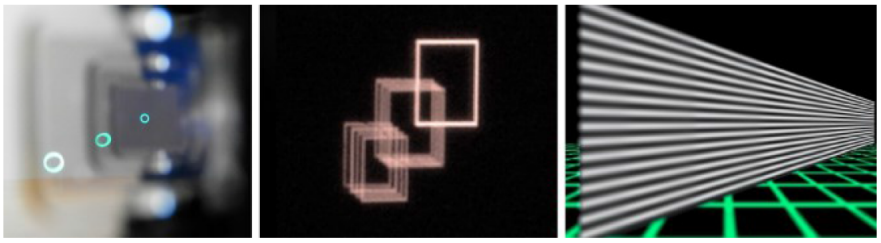


Figure 6.13: View through our multi-focal plane HMD showing circular icons (left), through Kim et al.s slanted light source display showing rectangular icons (center), and a simulation of accommodation effects by Liu et al. based on their guidelines for designing depth-fused displays (right) [Kim+11; Liu+10].

Though eye tracking is not utilized in most of these studies, the results of experiments involving depth judgment suggest that multiple focal planes can potentially be utilized for interaction in future research [LH10]. With improved methods for reproduction and perception of images, the opportunity has arisen for focus to be used as a means to automatically manage content on an HMD screen.

6.2.2.2 Studies on Depth Perception

Another closely related field of study is that of depth perception of virtual objects displayed in the real world. Initial work in this field sought to influence human perceptions of virtual objects in a single focal plane in order to more accurately reproduce digital content. One study by Uratani et al. attempted to use depth cues in a video see-through display to alleviate the depth ambiguity problem [Ura+05]. A similar study by Swan et al. investigated depth judgments in an optical see-through display, further emphasizing the importance of depth cues for depth judgments. One study of a static 3D display evaluated perception of accommodation, finding that focus is a viable depth cue for 3D displays in the monocular case [Kim+08]. More recently, a number of studies on depth were conducted, the first of which tested a wide field-of-view HMD and measured user perceptions of Landolt C-rings in retro-reflective screens. Results showed that perceptual accuracy diminishes with distance, especially when only presented with a monoscopic image [Mas+12]. Lang et al. studied the relationship between depth and visual saliency, showing that measuring depth can be

[3] A diopter is a unit of measurement of the optical power of a lens

used to improve saliency models. Though this was not a study on user perception, it further motivates the use of depth for improved interaction. Several other studies exist that evaluate perception and outline new display designs [Cho+12; LH10].

Based on the prior work in depth perception, we came to the conclusion that since depth perception can be influenced by adding appropriate depth cues, it is very possible that physical convergence of the eyes is also affected by the same cues. Since depth perception and eye vergence are controlled by different regions of the brain, we also included a variety of logically selected depth cues in our experiments to observe how closely these psychophysical functions are linked.

6.2.2.3 Gaze-Based Interaction in Augmented Reality

Gaze has long been studied as a method for interaction, but only due to the recent developments in display and eye tracking technology have 3D displays, gaze depth, and vergence been considered for interaction [BG10; Ure+11]. One of the first attempts at using gaze and depth for interaction in a static 3D display was conducted by Kwon et al. in 2006 [Kwo+06]. They set up a system using a parallax barrier type stereo display positioned 84 centimeters away from the user, and were able to estimate depth from a user's gaze toward virtual darts in one of 16 different regions of the screen. Another application by Lee et al. [Lee+11] used gaze and blink interaction for annotation tasks in a marker based AR workspace, though the display only utilized a single focal plane and focal depth was not considered. Most recently, 3D gaze has been proposed as a method for monitoring human attention [Ki+07]. Paletta et al. [Pal+13] also developed a system to recover a user's gaze onto objects in the real world using an RGB-D sensor to recover depth information. Though focal point is not directly calculated from the eyes, the gaze point in 3D can be recovered by incorporating reconstructed environment data.

As previously discussed in Section 6.1, one can use gaze focus for interaction. However, research up until now lacks interaction methods for multi-focal plane HMDs. Despite the appearance of several multi-focal or vari-focal HMDs, studies with those displays are limited to depth perception and have yet to take advantage of focal depth via eye tracking. This section describes the study that 1) measures accuracy and variance of focus in a monoscopic, multi-focal HMD, and that 2) tests the feasibility of natural methods for interaction in this kind of hardware system.

6.2.3 System Design and Framework

6.2.3.1 Overview

Taking the previously mentioned challenges into account, we set out to build an interactive prototype, test its potential for focus based interaction, and develop a framework to facilitate both automated and manual interaction methods. We first construct a 3D gaze tracking system combined with a multi-focal plane HMD that does not require the use of external tracking or projection hardware. Next, we develop a framework to facilitate more natural interaction with elements at varying focal distances and propose various methods for automating display of virtual content. We then conduct a series of tests on focal accuracy and depth cues in the prototyped display to determine the viability of the proposed methods, the results of which are discussed in the experiments section.

6.2.3.2 Multi-focal Plane HMD Prototype

Since most 3D display prototypes are static and cannot be used for mobile AR, we selected the following HMD setup for our prototype. It consists of an array of three 800x600 pixel AirScouter displays, each with its own digital input and depth control. For each plane, the focal depth can be set from 30 centimeters (cm) to 10 meters (m). The three displays were lined up so that three images could be viewed simultaneously during the experiment. The number of planes and their corresponding distances were selected via a pilot experiment. Furthermore, previous research has been conducted on information presentation in the near, mid, or far visual fields [Ura+05], which motivated to use these values. Also, the larger the number of focal planes, the harder it is for users to distinguish between them. The focal distances were set at 30 cm for near-field, approximately 1 m for mid-field, and at 10 m for far field using the manual depth controls on each display. These distances are similar to several other static displays [Kim+10; LH10].

Furthermore, we needed an apparatus for eye and vergence tracking that could be used simultaneously with an HMD placed near the user's eye. In order for focal depth to be measured appropriately, a user's eye convergence must be consistent and eye tracking hardware must provide enough accuracy to correctly select a target icon in the proper focal plane. In addition, we needed a way to make sure that the tracker and HMD would remain at the same distance to each other for all.

To ensure these conditions, we used a pair of SMI Eye Tracking Goggles, and created a 3D printed fastener that fixed the distance between the prototype HMD and the eye tracker as shown in Figure 6.14 B. Though the system still needed to be adjusted slightly for height and width of each participant's eyes, the distance between tracker and HMD remained constant. Using our setup, the images of the displays can be seen in a single eye (either left or right one).

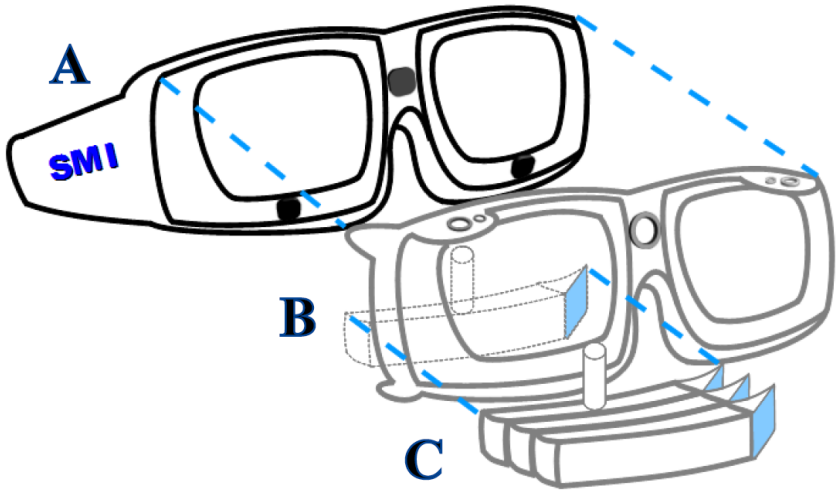


Figure 6.14: Our hybrid eye-tracker / HMD system showing A) SMI Eye Tracking Goggles, B) 3D printed attachment to affix tracker to HMD and C) custom multi-focal plane HMD.

6.2.3.3 Interaction Methodology

Though our prototype can be used for a number of purposes, we first sought to design two natural modes of interaction for a multi-focal see-through display scenario similar to the user assistance AR functions presented in Section 6.1. The first method was created to intelligently dim or close virtual content when a user changes his or her gaze to an environmental object (automatic dim for multi-focal display). The second method is designed to allow a user to manually re-open or interact with virtual content in a manner that is natural and accurate, and that incurs minimal distraction (eye-cons for multi-focal display).

6.2.3.4 Calculating Depth from Gaze

Up to now, a number of different models for calculating gaze depth have been proposed, though few have been designed for multi-focal plane HMD systems [WH12]. The method presented in the previous section (Section 6.1.2.2) is able to estimate whether the user is focusing on the HMD or not. However, for interactions with multi-focal planes, we need more fine-grained estimation. i.e., we need to calculate which HMD plane the user is focusing on. We propose two methods for robust focal plane estimation using a Support Vector Regression (SVR) and Support Vector Machine (SVM) (described later).

3D Gaze Estimation Using Vector Intersection From the eye tracker, we first extract a 3D vector of the direction of each eye, represented by $\mathbf{G}_R = (g_{rx}, g_{ry}, g_{rz})$ (right) and $\mathbf{G}_L = (g_{lx}, g_{ly}, g_{lz})$ (left) in Figure 6.15 (A). Using this data, the first type of tested estimation was based on linear gaze depth, which is the intersection of the two gaze vectors in 3D space. Unfortunately, these vectors rarely exactly intersect at a single point due to imperfections in the muscles and nerves that govern human eye movement (as I discussed in Section 6.1). During several informal experiments testing gaze estimation models, it was sometimes impossible to calculate the point on which the eyes converged in the far-field, since the angle between the vectors was obtuse, as shown in Figure 6.15 (B) and (C).

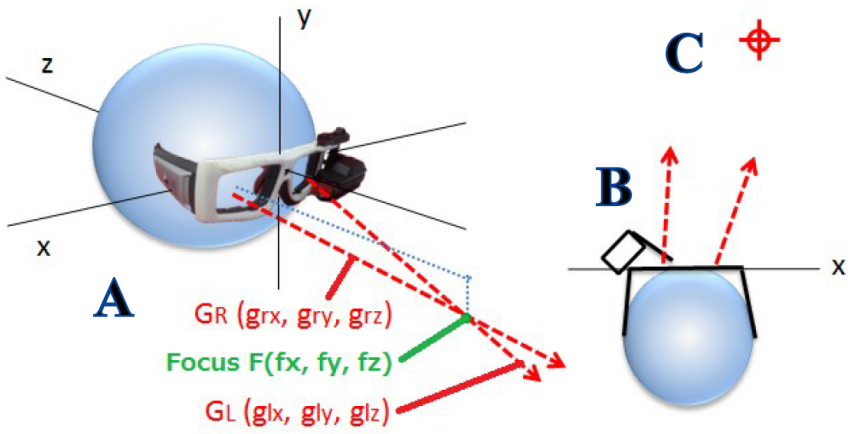


Figure 6.15: A) Visual representation of depth calculation using intersecting vectors, B) an example of far-field gaze vectors that do not intersect, and C) distant focal point.

Furthermore, this method often produces a large error even with a small difference in angle values when the focal point is in the far visual field.

SVR Model Therefore, instead of calculating depth using the intersection, we train a regression model of focal depths based on the x-value (g_{lx}, g_{rx}) of two gaze vectors. In short, by comparing the current gaze vector to a number of previously saved gaze vectors for each focal plane, we can achieve a more accurate depth estimate. Since the y axis in Figure 6.15 (A) is perpendicular to the line of sight, we can safely assume that the g_{rx} and g_{ry} are always the same. Suppose we have training data represented by

$$(\mathbf{x}_1, y_1), ..., (\mathbf{x}_n, y_n),$$

where $\mathbf{x}_i$ is the ith element from the training data. Vector $\mathbf{x}_i$ is represented by x-values of both gaze vectors, i.e.,

$$\mathbf{x}_i = (g_{lxi}, g_{rxi}),$$

and y_i is the depth value for ith training data. We train an SVR for our model, which computes a gaze depth value according to given gaze vectors. After training the SVR, we obtain the depth estimation function for a vector given by

$$f(\mathbf{x}) = \sum_{i=1}^{n} (\alpha_i - \alpha_i{}^*) k(\mathbf{x}_i, \mathbf{x}) + b,$$

where α_i and $\alpha_i{}^*$ are the Lagrange multipliers for the ith sample, and $k(\mathbf{x}_i, \mathbf{x})$ is the kernel function. For the kernel function, we use the radial basis function (RBF):

$$k(\mathbf{x}_i, \mathbf{x}) = e^{-\frac{\|\mathbf{x}_i - \mathbf{x}\|^2}{2\sigma^2}},$$

where $\|\mathbf{x}_i - \mathbf{x}\|^2$ is the squared Euclidean distance between two vectors $\mathbf{x}_i$ and $\mathbf{x}$, and σ is a free parameter. In the training of the SVR, the Lagrange multipliers α_i and $\alpha_i{}^*$ are optimized for the given training data $(\mathbf{x}_n, y_n)$.

SVM Model Although an SVR can calculate the depth value for any given gaze data, the estimation may have a large error. If the task is only to discretize the focused plane the user is currently looking at, we can consider this a multiclass classification problem. When we classify user gaze depth into one of multiple focal planes in our prototype, SVMs are applied. The advantage of this method is that we are able to distinguish focal planes even if we cannot calculate a precise focal distance. In other words, even if we have noisy gaze data or if calculated depth varies between users, we still know the plane in our prototype on which a user is focused. On the other hand, we apply SVR when we need a linear gaze depth value. The same formulas as the SVR are used for the SVM. However, instead of calculating the optimal Lagrange parameters for regression, the SVM calculates the parameters which are optimal for separating individual classes.

6.2.4 Experiments

In designing experiments, our goals included 1) testing the resolution at which focal depth could be tracked and 2) testing the ability of users to focus on interactive elements to determine if our interaction framework was feasible.

6.2.4.1 Pilot Study

Here we briefly describe the results of a pilot experiment with 4 users testing focus on physical objects at different distances. The resulting data allowed us to determine appropriate distances between focal planes in our prototype.

Setup To get some idea of how accurately we could calculate gaze depth, we gave participants the task of focusing on a plus symbol on a small sheet of paper and rotating their heads horizontally from left to right in a 180 degree arc. Participants sat with their eyes at the level of the plus as shown in Figure 6.16. The eye-tracking apparatus was affixed to

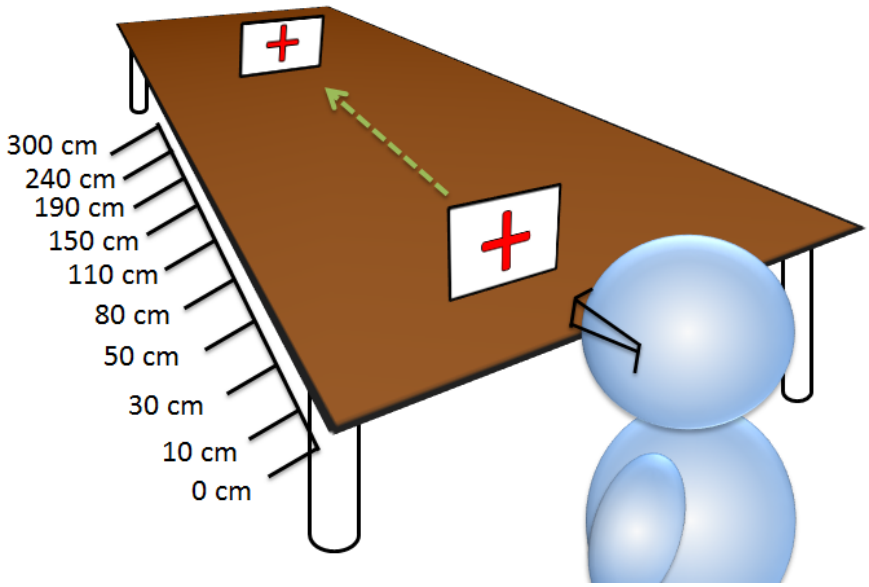


Figure 6.16: Pilot experiment setup showing gaze targets and depths at which measurements were taken.

the participant's head, and the task began. The participant was asked to focus on the plus at 10 cm and rotate, and approximately 10 seconds of gaze data (at least 200 samples per user) was recorded. This process was repeated at 30 cm, 50 cm, 80 cm, 110 cm, 150 cm, 190 cm, 240 cm and 300 cm, and then repeated for each participant. For SVR, recorded gaze data was separated for testing and training. A 10-fold cross validation was processed for evaluation. For the kernel function, we used the RBF from LibSVM [CL11].

Results Resulting rotation data for one user is plotted in Figure 6.17, which shows gaze samples at each distance and trend lines representing feasible, relative depth estimations up to 190 cm as a function of head movement. Despite rotation, depth estimates remained relatively constant in the near and mid viewing fields. Humans typically turn their heads if gaze angle exceeds 30 degrees, making data outside this range less relevant.

More importantly, accurate separation of focal planes becomes difficult after 110 cm. A plot showing depth estimation for each participant at each focal distance is shown in Figure 6.18. Each cluster of four points shows both the estimated distance and variance at that depth for a single user. From this data, we needed to select focal distances that

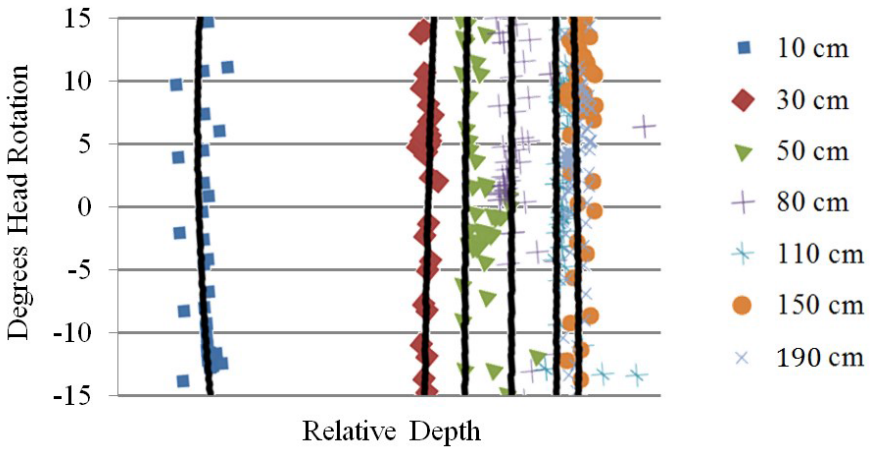


Figure 6.17: Gaze depth estimates for approximately 30 degrees of head rotation, where y axis is rotation and x axis is relative depth. Trend lines are shown for reference.

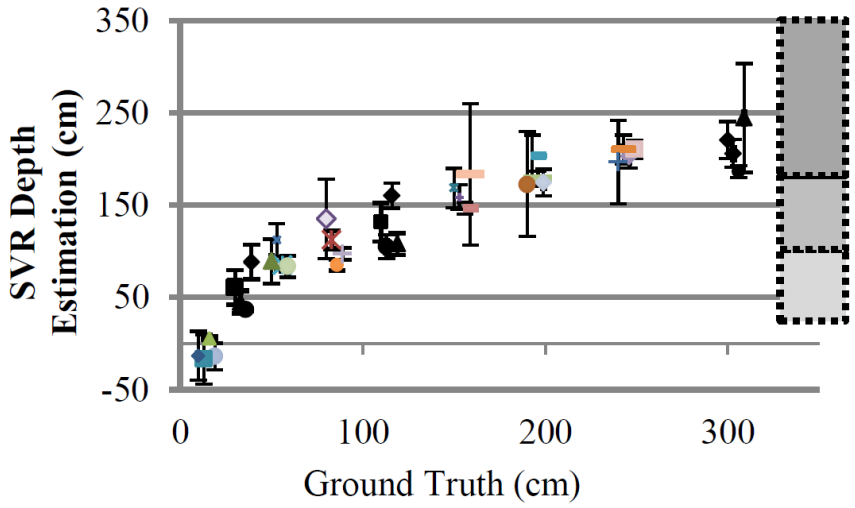


Figure 6.18: Gaze depth estimates with SVR for all four users and respective standard deviation. Resulting focal plane discretizations are delimited by the dotted boxes on the right.

had the smallest overlap, which would consequently maximize the chance a focal plane in our prototype would be correctly selected by a user, even with noisy data. Minimal overlap occurred at the three depth ranges outlined by the gray dotted boxes. We decided to utilize a near, mid and far plane, with distances at approximately 30 cm, 1 m, and anything further than 2 m, respectively. As expected, gaze data differs between subjects, making precise depth estimation difficult for practical use, however; our method of discretizing focal planes provides significant robustness to variable data. As evident in the next experiments testing depth based selection, these distances were appropriate choices.

6.2.4.2 Study on Icon Selection and Depth Cues

In order to determine whether our automated dim/close and manual interaction methods would be feasible with a multi-focal HMD, we conducted a second, larger experiment. Users were tasked with viewing a number of different icons through the HMD at different focal depths. We also included various colors and depth cues such as blur and texture to see if there was any effect on physical eye convergence. Since prior research mostly focuses on perception, we wanted to go a step further and learn about the physical behavior of the eye, especially in this type of monoscopic display.

Setup A total of 14 users, 9 male and 5 female, participated in the experiment. Using the hybrid eye-tracker/HMD prototype discussed previously, we presented 9 sets of 3 different icons to each user (one icon for each display). For each set, we had different color and depth cue setting as shown in Table 6.1. A view through the HMD of one set of icons without any

Table 6.1: 9 sets of different color and depth cue setting.

set	color	depth cue
A	green	no cue
B	green	size
C	green	gradient
D	green	blur
E	green	all cues (size, gradient, and blur)
F	white	no cue
G	white	all cues (size, gradient, and blur)
H	fuchsia	no cue
I	fuchsia	all cues (size, gradient, and blur)

simulated depth cues is shown on the left of Figure 6.13. Though a number of different icon shapes could have been tested, we chose a circle since Landolt rings and circular icons are often used in visual aptitude tests [Mas+12]. Secondly, depth cues such as texture can be displayed symmetrically on all axes, eliminating additional variables. Simulated monocular depth cues were held as variables, and included relative size, texture gradient, defocus blur, and a combination of all three. Figure 6.19, shows a simulated 3D view of a single set of icons including all depth cues, where the effect of each cue progressively increases as plane distance increases. Note that the blue lines were not visible in the experiment. Cues inherent to the display which were held constant throughout the experiment included accommodation due to focal depth and elevation, since more distant icons were presented at higher vertical locations. Three different colors were presented (bright green, white, and fuchsia) to test whether certain colors are better focal targets at different depths.

Each participant was first asked to put on the prototype and adjust the HMD until he or she could see all three icons clearly. Next, the user was asked to gaze at one of the three icons, and we recorded 10 seconds of gaze data for that icon. The process was repeated for each icon in the set, then the next set of icons was displayed, and the task was repeated for all 9 sets of icons. The order of the 9 sets of icons was randomized between participants to prevent any learning effects. All trials were conducted in a room with constant lighting conditions, and all participants were instructed to face a diffuse, uniform wall throughout

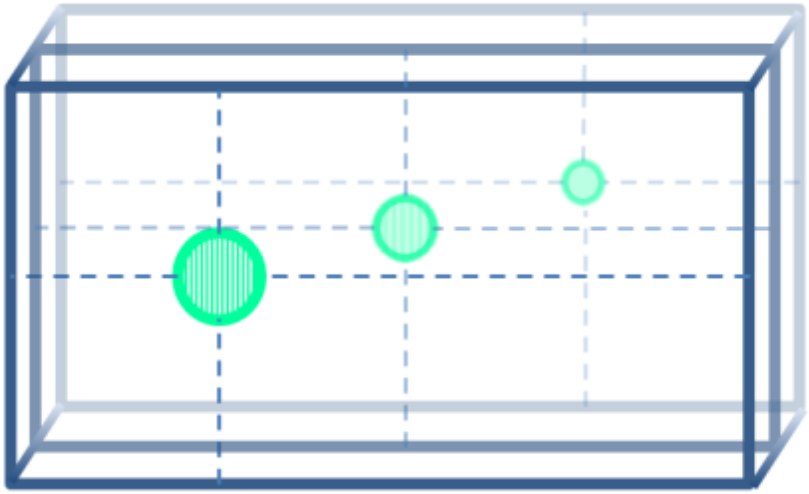


Figure 6.19: Simulated 3D view of a set of icons including relative size, texture gradient, and defocus blur.

the experiment.

Results The most important result from this set of experiments was that we achieved a high degree of accuracy for focal plane identification. Out of all samples taken for all users, 98.63% of points were classified into the correct focal plane. Even in the worst case scenario, the sampling data of which is shown in Figure 6.20 (B), we achieved 85.6% accuracy.

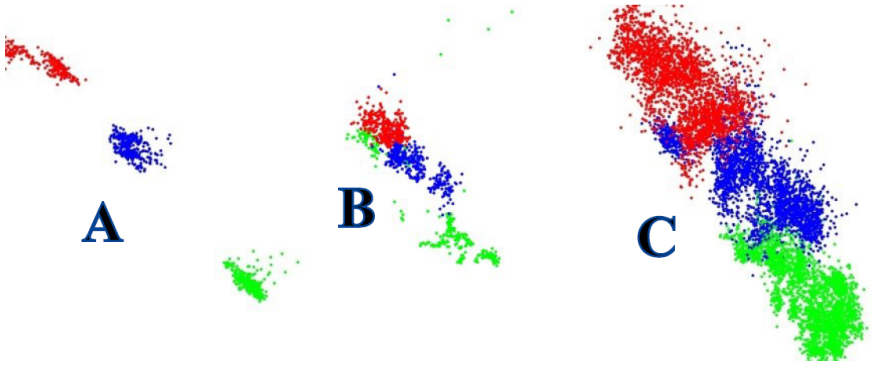


Figure 6.20: Gaze data for A) a trial with high plane classification accuracy ($98\%+$), B) the trial with the worst accuracy (85%) and C) all trials for a user with average accuracy. Focal plane classifications are represented by green (30 cm), blue (1 m), and red ($2+$ m). Each dot represents an individual gaze sample.

Since this data is per sample, this means that we can correctly classify a focal plane more accurately using a running average of several samples. For instance, 99.98% accuracy can be achieved using a running average of 8 samples. Thus, it requires 500 msec to identify a correct plane for any user, and less than 100 ms for a vast majority of cases. For real

time everyday use, this low latency is essential for quickly removing distracting information. Also, based on the sample data for all trials shown in Figure 6.20 (C), it would become increasingly difficult to accurately determine which sample belongs to which plane as the number of planes increases. Hence, 3 focal planes is likely an excellent choice to ensure robustness for correct identification. The second and perhaps more interesting result from our experiment was that the simulated depth cues and colors had no statistically significant effect on physical eye convergence.

For each comparison, we conducted an ANOVA test to determine if any effect existed, as outlined in Table 6.2. Looking at the fourth row of Table 6.2, where all trials were conducted

Table 6.2: ANOVA of gaze accuracy with both colors and sets of cues held as constants and variables.

constants	variables	p-value
green	no cues, all cues	$0.83 > .05$
white	no cues, all cues	$0.70 > .05$
fuchsia	no cues, all cues	$0.81 > .05$
green	blur, gradient, size	$0.95 > .05$
no cues	green, white, fuchsia	$0.38 > .05$
all cues	green, white, fuchsia	$0.95 > .05$

with green icons, but blur, gradient, and size were varied between trails, p-values show no effect. Though we expected at least some depth cues to influence classification accuracy, the highest p-value observed was only 0.38. This result suggests that when designing icons for selection or interaction, blur, gradient, and small deviations in size are likely to have little effect.

6.2.5 Discussion

One useful finding from our experiments is that eye convergence occurs even when an image is presented only to a single eye in a monocular display with multiple focal planes. Even without presentation of a stereoscopic image, the brain directs both eyes to converge on a point of interest. It is very likely that this effect is tied to the same mechanism that controls blinking and pupil dilation, considering both eyes will respond when presented with blink stimuli or light to only one side [Dra12]. We can safely assume that eye tracking for depth calculations can be accomplished in a monocular HMD with multiple focal planes.

Another unusual finding is that the depth cues in our experiment and variance in physical eye convergence were largely unrelated, despite a strong demonstrable relationship between depth cues and depth perception in other research. This evidence suggests that accommodation, the constant monoscopic depth cue in our experiment, may be the stronger factor when the human eyes try to converge at a certain depth in 3D space. Though only three highly visible colors were selected, color mixes also deserve consideration.

6.2.5.1 Implications for Interaction

Most importantly, we find that users can consistently focus on icons in different focal planes, and that this can be tracked accurately for use with visual interfaces. In most cases,

gaze depth could be calculated with only a few samples, which means that the automated dim/close method can be executed almost instantly. This speed is essential for times when a user may have to react to an oncoming car or hazardous object.

Eye-based interaction using dwell or focus changes are often thought to be tricky and require fast reaction times on the part of both the user and HMD. However, in our interface, there is an innate difference between the automated dimming and manual select methods. Automatic dimming can be done very quickly through the HMD hardware since any deviation from the current focal plane for a short period of time can be considered a change in focus. Hence, the automated method, which we can think of as a more natural interface, does not suffer from some of these timing and dwell problems.

6.2.5.2 Future Work

As future work, we plan to conduct additional experiments testing the use of icons in a user's peripheral vision. This will allow us to test distraction and other methods for interaction in more detail. Additionally, we intend to utilize the outward facing camera on the eye-tracking interface to determine appropriate locations for icons in a user's field of view or in the environment such as [Orl+13a]. Lastly, thorough testing of the interaction methods in outdoor AR environments is necessary to determine usability in dynamic environments. Recordings of user eye movements coupled with the outward facing camera will likely provide further insights into the behavior of the eye.

6.2.6 Conclusion

In this section, I developed a novel interface for interaction with semi-volumetric displays. To do so, I prototype a hybrid multi-focal plane eye-tracker/HMD, which facilitates methods to automatically close, remove, or dim distracting content from a user's field of view. I then conduct several experiments to test the effectiveness of our prototype and viability of methods for interaction and find that users can accurately focus on virtual content on different focal planes in our prototype. More importantly, this focus can be tracked robustly for use with other similar HMD interfaces, and distracting text can be quickly removed in situations that require immediate attention.

6.3 Summary

In this chapter, I discussed user interaction with a see-through HMD using eye gaze. In the first section, I investigated the user attention engagement estimation methods for a see-through HMD. The experimental results indicated that both proposed methods (gaze depth-based and VOR-based) can reasonably estimate the user attention engagement separating virtual and physical. Furthermore, I proposed several proactive user assistance functions for a see-through HMD that utilize the engagement estimation and the cognitive state analysis. Users can benefit from such functions when interacting with a see-through HMD.

In the second section, I proposed a gaze-based interface for a multi-focal display. Similar to the attention engagement estimation method in the first section, the interacting focal plane could also be estimated using user's gaze focal depth. The experiments showed that three different focal planes can be robustly separable using the SVM classifier with the state-of-the-art eye tracking device.

The presented systems and interaction frameworks in this chapter showed the potential of eye gaze interaction in combination with a see-through HMD. The users can intuitively interact with an AR system using eye gaze.

Chapter 7

Comprehensive Framework for User-Attended Visual Content Analysis in a Complex Scene

Chapters 4 and 5 presented several methods for recognition of user-attended content. These methods can recognize to which visual content the user is attending, especially in individual everyday scenarios. However, we always have a precondition for previous VCA methods that the type of visual content is determined by the application. For example, we use OCR to recognize text for the translation system. In this chapter, I discuss extension of VCA for a more uncontrolled everyday scenario where multiple types of visual content are present in the same scene. In particular, I propose a system that can recognize the visual content regardless of the content type the user is attending to. When various types of visual content are present in a scene image, we need to run an appropriate image analyzer. Here I propose a method that recognizes the user's cognitive state to select a proper image analysis module. With a text example, by recognizing his or her reading state from eye movements, we can select gaze-guided OCR as the image analyzer. In the experiment (Section 7.2.3), I compare the performance of the two different processing strategies for VCA for multiple types of content. One is a strategy which uses cognitive analysis for image analysis module selection, whereas the other runs all image analysis modules simultaneously.

Figure 7.1 shows an exemplary scene with objects, text, as well as a face, all of which may be valuable resources for understanding a task context. Our aim is to observe users in uncontrolled environments, recognize their visual attention and employ the data to run several modules for image analysis, each specialized to a different resource type. The objective of the proposed system is to provide a comprehensive framework that is general to cope with different types of visual resource by using different types of image recognition modules.

Studies on human eye movements have shown evidences that especially fixations and eye movements are closely related to tasks or actions a subject is performing [LH01; Cas+09; Ray95]. Employing these insights, several approaches have been proposed for recognizing user activities, cognitive states, reading tasks, or viewing-tasks [Bul+11a; Hen+13; Bie+12; CK14]. Especially, cognitive states are a valuable source of knowledge. Once the user's cognitive state is identified, it generates expectations towards the content, i.e., it reduces the number of possible classes for content recognition, but also helps to guess a user's intention

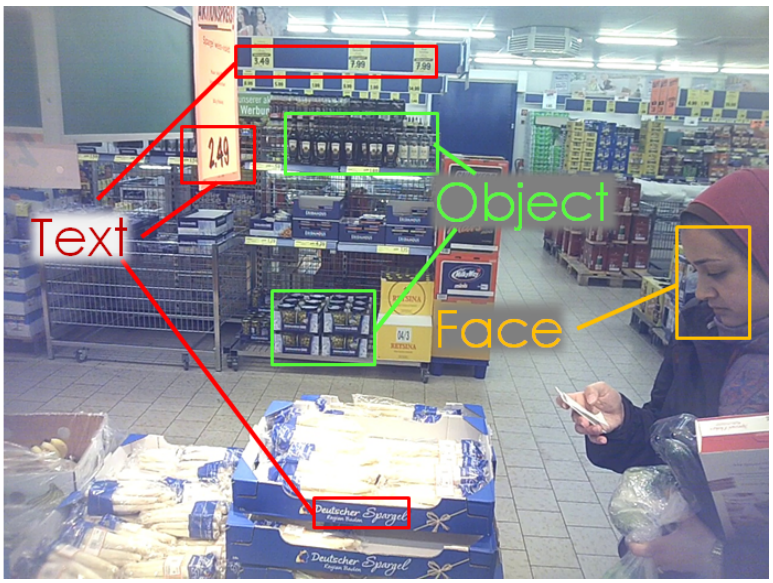


Figure 7.1: Daily scenes often consist of a complex structure. Usually, various types of background objects and foreground objects are present in a scene. In such a complex visual context, to ascertain which content the user is paying attention to is a key issue. The proposed framework aims to understand the user context by recognizing the content conceptually in presence of the user's attention.

or situational demand. The proposed attention-driven image analysis (AIA) method is a processing strategy which incorporates the cognitive state recognition framework for selection of VCA, i.e, image analysis modules. In contrast, the continuous image analysis (CIA) method runs all image analysis modules simultaneously and selects the most probable visual content that the user is attending to in a later step of the recognition process.

Similar to the prior chapters, the eye tracking components are used for analyzing a user's visual attention and the HMD is used for presenting visual information to the user. This system is intended to be applied to several scenarios in complex everyday environments, where multiple contextual content is present. The various resource types are processed by different image analysis modules and combined to a single result via a so-called *cognitive merger* (Section 7.1.6).

The main contributions of this chapter are i) to present a comprehensive framework for recognition of user-attended content in complex environments and ii) to evaluate the framework of two different processing strategies, i.e., with and without integrating information of the user's cognitive state for user-attended content recognition. In order to demonstrate the system in different environments, three scenarios are selected. One is a "Poster Browse" scenario, where many images and text are present. The second scenario is "Factory Work", where 3D objects, written instructions, and colleagues' faces are present. The third scenario is "Meeting", where faces and text are present. We conducted experiments for evaluation of the proposed system in these three scenarios.

7.1 User-Attended VCA for Multiple Types of Visual Content

7.1.1 Architecture

The architecture of the proposed system is illustrated in Figure 7.2. As an input interface, we use the wearable eye tracker showed in Section 3.3.3. Initially, a scene image and two (left and right) eye images are sent from the eye tracking glasses to the eye tracking server.

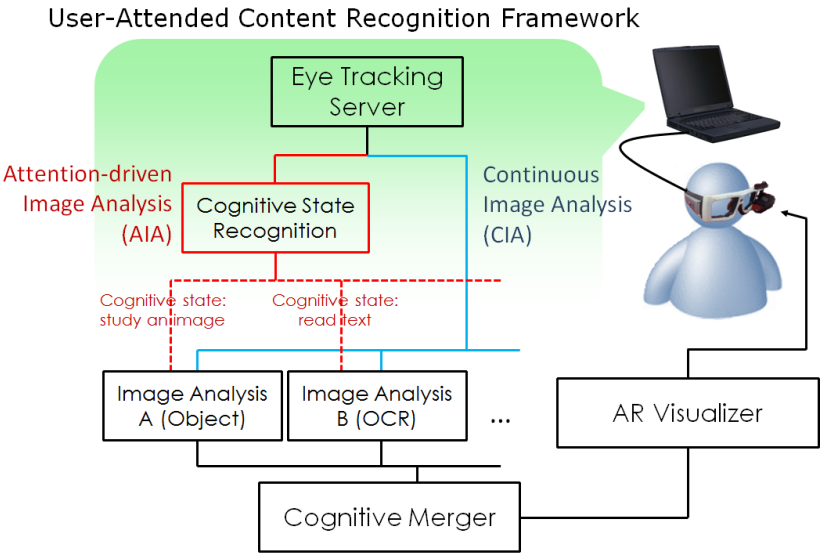


Figure 7.2: Proposed comprehensive user-attended visual content recognition system architecture. We propose two different processing strategies. With the AIA strategy, a corresponding image analysis module is selected according to the user's cognitive state. On the other hand, cropped scene images are directly piped to all image analysis modules with the CIA strategy. The final recognition result is generated in the *cognitive merger* and the supportive information about the user-attended content is presented in the HMD by the AR visualizer.

For recognition of user-attended content, we compare two different processing strategies. One is called **Attention-driven Image Analysis (AIA)** and the other is **Continuous Image Analysis (CIA)**. With the AIA, the cognitive state of the user is identified by eye gaze movement analysis in order to select a respective image analysis module. Suppose the user reads text. The cognitive state recognition module analyzes his or her eye movements and outputs a *reading state*. Since a *reading state* is detected, OCR is selected as the image analysis module for current images.

On the other hand, images from the scene camera are continuously piped to all image analysis modules with CIA, whatever the user's cognitive state is. Since this strategy processes all images anytime, CIA can be regarded as a brute-force approach. For both strategies, a scene image is cropped using the user gaze position in order to analyze the visual content in the user's focus (For detail, see 7.1.5). Hence, both approaches benefit from the user's eye gaze to recognize the user-attended visual content. To investigate the potential of cognitive state analysis in terms of user-attended VCA, we compare these two approaches in the experiment section (Section 7.2.3).

From each image analysis module, the identity of the image content is returned (e.g., if no text is present in the image, it returns "no text"). The cognitive merger unit then collects the recognition results from all image analysis modules and generates the most probable result, i.e., what content the user is attending to. Once user-attended content is recognized, the database manager selects relevant information from the database and sends it to the AR visualizer, which displays the meta-information in the HMD. For the AR visualizer, we can also use the gaze-based interactive functions presented in Section 6.1. Especially, the automatic dim function is effectively used in order to present the AR only when the user attends to the HMD virtual screen.

7.1.2 Type of Visual Content

The content comprises different resource types, i.e., objects, text, faces, etc., which may be conceptually recognized by human in a given environment. In order to understand attention, we first need to unitize different image analysis approaches in order to recognize the considered resource types and classify the content respectively. In the previous chapters (Chapters 4 and 5), I showed the robustness of the state-of-the-art image recognition approaches. For instance, matching of local image features such as SIFT can robustly recognize a rigid (2D and 3D) object (Section 4.1). Also, there are robust approaches for face recognition (Section 4.3) and character recognition (Section 5.1). By either properly selecting the right image analysis module (AIA) or merging the individual recognition results (CIA), recognition of visual content may become more accurate and appropriate.

For a demonstration, I present a framework that allows for an integration of image analyze modules for any type of resource. However, in this thesis, I first focus on application-relevant objects, related text and the faces of involved people for a proof of concept.

7.1.3 Processing Strategies

As mentioned previously, we compare two different processing strategies:

1. **Attention-driven Image Analysis (AIA)**: An image analysis module is selected according to gaze data and the derived cognitive state. Each cognitive state is a precondition of each image recognition module as shown in Figure 7.2.

2. **Continuous Image Analysis (CIA)**: The scene image and the corresponding gaze position are continuously sent to all the image analysis modules (a brute-force approach). As shown in Figure 7.2, all image analysis modules run regardless of the cognitive state.

The difference of the two strategies is illustrated in Figure 7.3. Both strategies have individual weaknesses. While CIA is computational resource-consuming and therefore may cause a serious overhead, AIA may miss some important information because of a misinterpretation of a cognitive state.

7.1.4 Cognitive State Recognition

The gaze analysis step comprises a cognitive state recognition module, which follows approaches of previous work [Hen+13; CK14; Bul+11a]. The module continuously tracks the

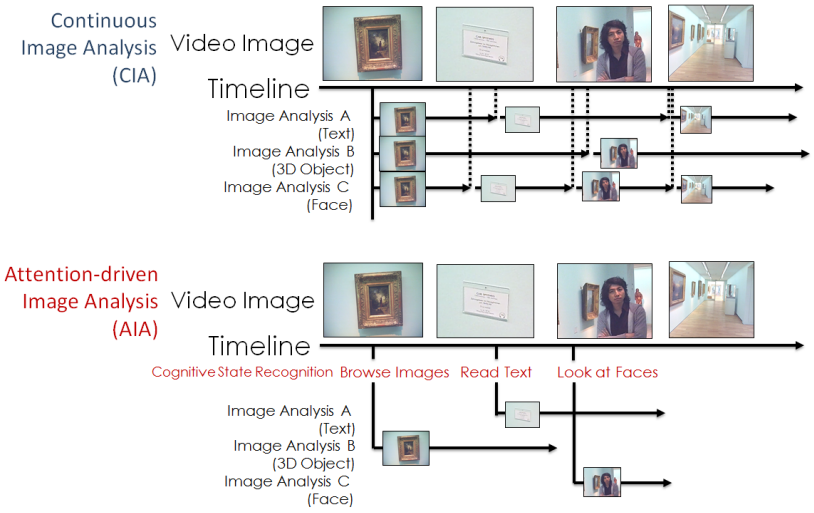


Figure 7.3: Difference between AIA and CIA[1]. Top: CIA strategy. The system continuously runs the threads for all recognition modules in parallel (a brute-force approach). As soon as each recognition process is done, a new image is sent from the eye tracker. Bottom: AIA strategy. On the other hand, an image analysis module is selected according to the cognitive state recognition result.

eye gaze for extracting features that are used in the AIA strategy.

I extend the feature extraction method proposed in Section 6.1.2.3. The gaze features are measured within a certain local time window W, that spans from the current time frame t to the preceding time frame $t - W$ as shown in Figure 7.4. The gaze data is taken

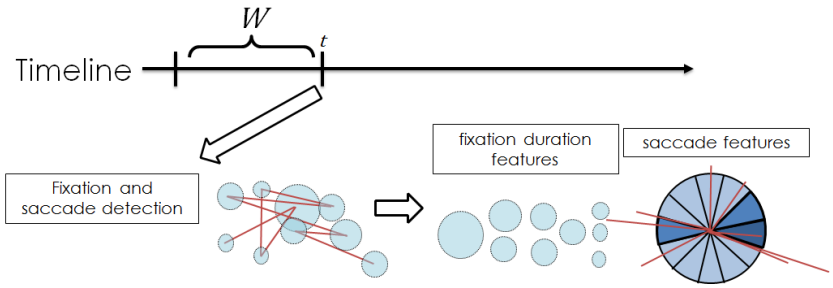


Figure 7.4: Gaze features for cognitive state recognition. For time t, fixations and saccades are detected from a local window W. Then features are extracted from fixation duration and saccade orientations.

to extract fixations and saccades using a dispersion approach [Bus+08]. For detection of fixations, we have two threshold parameters: the fixation duration threshold T_{dur} and the fixation dispersion threshold T_{dis}, which respectively determine the temporal and spatial size of a fixation. Using the extracted fixations and saccades, we compute statistical features of eye movements as follows (also see Figure 7.4):

- Fixation duration average: the average of overall fixation duration.

[1]Images were taken in Museum Pfalzgalerie Kaiserslautern (mpk), Carl Spitzweg, Zeitungsleser im Hausgrtchen, 1845/48, Öl/Holz, 21,5 x 15,5 cm, Inv Nr. BST 81.

- Fixation duration deviation: the deviation of overall fixation duration.
- The most dominant saccade orientation (O_1): the bin index of the most dominant saccade orientation.
- The second most dominant saccade orientation (O_2): the bin index of the second most dominant saccade orientation.
- The dominancy of O_1: the dominancy ratio of O_1.
- The dominancy of O_2: the dominancy ratio of O_2.

The orientation angular value of saccades is divided into 18 bins and each saccade falls into one of the bins. From all saccade eye movements, a histogram of orientation is created as shown in Figure 7.4. The most dominant orientation is the largest population of the bin and the ratio is the population ratio of the number of saccades.

The features are used to train a multi-class SVM classifier for cognitive state classification. Although there may be other classes of user cognitive states (e.g., discrimination between visual search and memorization), in this work I focus on the following three classes: *image study*, *text read*, and *non-visual task*. I discard other cognitive states to keep the mechanism simple as it is still a prototype.

Note that each cognitive state implies a respective image analysis module in the AIA framework. If the cognitive state is *image study*, the image is sent to the object recognition module and if it is *text read*, the image is sent to the text recognition module respectively. Though the cognitive state *non-visual task* merely indicates that neither active visual recognition nor visual attention is required. However attention may be motivated by other reasons, e.g., by listening to someone. Such an attentive activity may be important in the setting we focus on. Moreover, listening to someone's speech could also be inferred from eye movements, as shown by [Gop73]. If a respective cognitive state is recognized, we apply the face recognition module that searches for known faces in the user's field of vision[2].

7.1.5 Image Analysis

As previously mentioned, I focus on application-relevant objects, related text and the faces of involved people in this work. To recognize visual content in both CIA and AIA strategies, we use three different recognition modules as shown in Figure 7.5: object recognition, text recognition, and face recognition modules.

7.1.5.1 Image Crop

From the eye tracker, we receive a gaze position, which indicates the user's focal point (point of attention). This allows us to limit the region of interest (ROI) for each image analysis module as shown in Figure 7.6. As a consequence, image analysis processes are sped up since the image sizes become smaller than the original scene images. In both AIA and CIA strategies, images are cropped according to the current gaze position. The size of local image region is determined according to the type of image analysis module. That is, we change the cropping size depending on the algorithms used in the image recognition

[2]We discard other conversational states such as that the subject speaks, etc. since these types of activities are rather complicated to integrate in the framework.

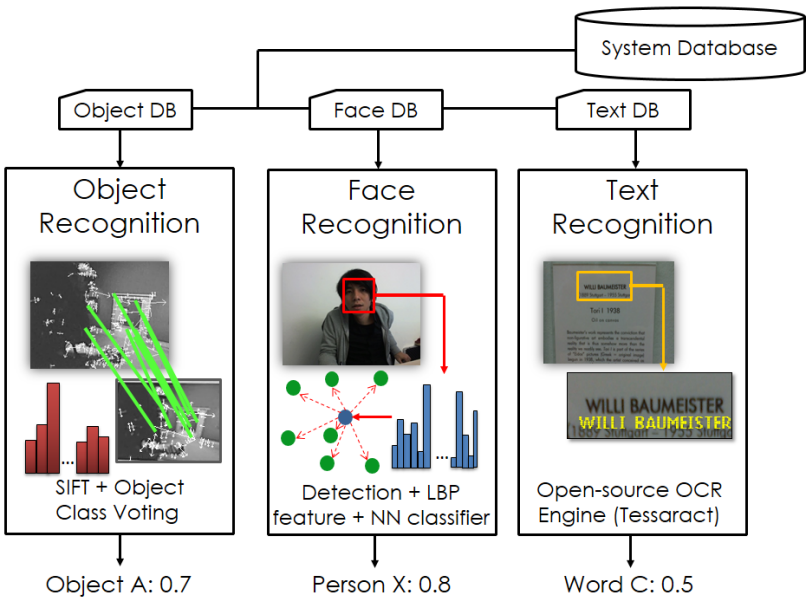


Figure 7.5: Image analysis modules. Each module loads a respective database from the system database. After each recognition process is completed, it outputs the result label with a confidence score (e.g. Object A: 0.7).

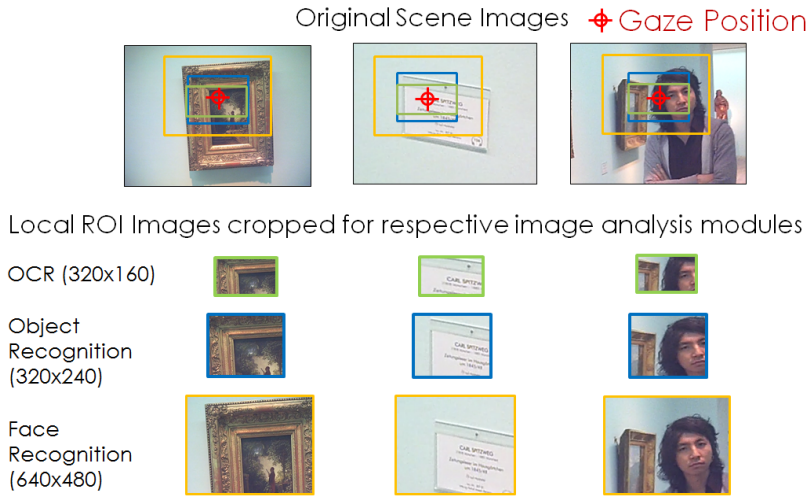


Figure 7.6: Local ROI image cropping. The size of a local image is determined by each image analysis module. The cropped image is used to recognize the visual content the user is fixating at in both AIA and CIA. The face detection would fail if the size is too small (in 320x160 or 320x240).

modules. As a result of recognition, each image analysis module returns the identity of the content and the confidence score as shown in Figure 7.5. The confidence score is later used by the *cognitive merger* in order to identify the most confident recognition result at the moment. In the following, I present each image analysis module and confidence score calculation.

7.1.5.2 Object Recognition

Similar to the approach used in Section 4.1, all relevant objects in each intended scenario are described by SIFT features learned on a training images of each object from different perspectives. This way, we collect thousands of local features for each object class, which are stored in a database. The recognition is done by calculating the majority vote of SIFT features as described in the previous section. If the majority ratio is less than the threshold value T_{obj}, the recognition result is discarded, i.e., it does not send any recognition result. The confidence score for object recognition result C_{Object} is computed as the normalized value of the majority ratio. For object recognition, the local image size is set to 320x240, which is sufficient to extract SIFT features from the user's focus (see also Section 4.1.5.1).

7.1.5.3 Face Recognition

Face recognition is applied for a limited set of faces belonging to persons who are relevant to the application, e.g., colleagues who are collaborators in a factory scenario. Accordingly, the faces are learnt and stored in a database. For their recognition, the face recognition module presented in Section 4.3 is used. Because we use the NN method, the nearest facial feature vector in the database in terms of the Euclidean distance is returned as the recognition result. A respecting confidence score for face recognition result is calculated using the Euclidean distance of the query LBP vector to the recognition result vector as follows:

$$C_{Face} = \begin{cases} 1 & (d_{withinclass} > d) \\ \frac{d_{otherclass} - d}{d_{otherclass} - d_{withinclass}} & (d_{otherclass} > d > d_{withinclass}) \\ 0 & (d > d_{otherclass}) \end{cases},$$

where d is the calculated distance, $d_{withinclass}$ is the closest distance within the class, and $d_{otherclass}$ is the average of the distances in the other classes.

We set the local size for face recognition relatively wide (640x480) because the face detection would fail if a whole facial image is not present (e.g., see also the images for OCR and object recognition in Figure 7.6.) and is quite fast (compared to the other image analysis modules).

7.1.5.4 Text Recognition

In this chapter, I focus on English text because I need to collect the gaze data when the experimental participants read text for cognitive state analysis and English is the only language which can be read by all participants. Thus, we need an OCR engine for English text. Since the OCR method presented in Section 5.1 is specialized to Japanese text, we use another OCR engine called *Tesseract*[3] in this chapter. Once the text recognition module receives a local image, the image is directly forwarded to the Tesseract recognizer. The Tesseract recognizer returns recognized text strings for a given image. Similar to the face recognition, we take the string that is the closest to the center (the gaze position). To recognize words, the resulting string is verified via a word dictionary, capturing relevant terms, phrases and sentences of the application. For finding the relevant dictionary entry, we use the *Levenshtein Distance* d_{Leven}, an edit distance measure [Lev66]. As result, the

[3] https://code.google.com/p/tesseract-ocr/

closest match plus a confidence score is returned (e.g., if the OCR result is "differelle", the closest word in the dictionary: "difference" is returned). The confidence score is calculated by the following formula:

$$C_{Text} = \frac{max(l_{ori}, l_{dict}) - d_{Leven}}{max(l_{ori}, l_{dict})} \cdot C_{Tesseract},$$

where l_{ori} is the length of the recognized word string, l_{dict} is the length of the closest word string from the dictionary, and $C_{Tesseract}$ is the confidence value provided by Tesseract.

Compared to the other recognition modules, the local size of text recognition is narrow (320x160). With the camera we use, 320 pixels correspond to 15 degrees in the angle of view, which is sufficient enough to capture the word that the viewer is attending to (with normal visual acuity).

7.1.6 Cognitive Merger

As an outcome of image analysis, the identity of the current gazed content is provided. Recognition results from all the image analysis modules are continuously gathered in the cognitive merger unit as already shown in Figure 7.2 (both in AIA and CIA strategies) in order to identify a single content that the user is most likely currently attending to. The confidence scores provided by the image analysis modules are used to identify the most probable user-attended content. In order to make a balance between the image analysis modules, each confidence scores is scaled using a respective heuristic scaling factor: object recognition s_o, face recognition s_f, and text recognition s_t. These scaling factors are tuned in the experiment. They provide final confidence values for individual recognition results: $C_{oFinal} = C_{Object} \cdot s_o$, $C_{fFinal} = C_{Face} \cdot s_f$, and $C_{tFinal} = C_{Text} \cdot s_t$, respectively. The recognition result that has the highest final confidence value at time t is determined as the current user-attended content. By adapting the threshold value for confidence score, the cognitive merger can also remove less precise erratic outputs.

7.2 Experiments and Evaluations

In this subsection, I describe the set of experiments for evaluating the system and show some exemplary results. First, preliminary experiments are carried out to tune the parameters for each image analysis module and to investigate individual accuracy of image analysis in everyday environments. Second, the accuracy of cognitive state recognition is evaluated. The accuracy of cognitive state recognition is a key factor for the accuracy of the AIA strategy since the selection of image analysis module is driven by the cognitive state recognition. Finally, evaluations of the whole user-attended content recognition framework with different practical scenarios are presented. Here, we compare two processing strategies (AIA vs. CIA) from various perspectives such as recognition accuracy and computational efficiency. As practical scenarios, a "Poster Browse", "Factory Work", and "Meeting" scenario are chosen for evaluation.

7.2.1 Preliminary Experiments

In order to investigate the accuracy of each image analysis module individually and to tune the parameters, we carried out preliminary experiments. These experiments would provide

us with assumptions as to what degree individual visual content could be recognized in different conditions of natural environments using the image analysis modules. Note that this experiment was done only using images. Eye gaze was not involved at all in this experiment.

7.2.1.1 Experimental Settings

For evaluating the robustness of each image recognition module (OCR, object recognition, and face recognition) in different conditions, we created two different test datasets for individual modules. The *basic* sets contain images which are visually intuitive and distinct, i.e., no noise, easy foreground-background separation, similar lighting conditions, and similar distances to the targets (compared to the training datasets). The second *complex* sets comprise examples with multi-resource settings or disturbed images, different lighting conditions, and different distances to the targets, which simulate more realistic environments. Example images of each test and train dataset and a summary of the experiment settings are shown in Figure 7.7 and Table 7.1, respectively. In order to test the recognition with everyday environments, we took pictures of faces of people in an office, objects in a museum, vocabulary cards on a table.

Figure 7.7: Examples of test images used in the preliminary experiments[4]. Images in the two left columns are examples from the *basic* sets while the images of the right two columns are from the *complex* sets.

By changing the threshold values for the confidence scores of image analysis modules, we can control the trade-off between precision and recall of recognition. Precision is the ratio of true positives within the whole outputs, whereas recall is the ratio of true positives within the whole test samples. If a confidence score is under the threshold value, there is no output. Thus, high threshold values could increase precision but decrease recall and vice versa.

[4]Museum exhibits: Arnold Böcklin, Nessus und Deianeira, 1898, Öl/Lw, 103,5 x 150 cm, Inv. Nr. LG 66/15, Kunstbesitz des Bundes, Leihgabe im Museum Pfalzgalerie Kaiserslautern (mpk); Niclaus Gerhaert van Leyden, Trauernder Engel, um 1462, Nussbaumholz, geschnitzt, gefasst, vergoldet, 33 x 54 x 30 cm, Inv. Nr. K 1497, Museum Pfalzgalerie Kaiserslautern (mpk).

Table 7.1: Settings for preliminary experiments.

image analysis	training set	*basic* test set	*complex* test set
OCR	- built-in database of Tesseract (no additional training image was taken) - $10,000$ English words in the dictionary	- images were taken under bright lighting condition and a distance to the vocabulary cards of approx. 30 cm - 133 test images	- either the light was dark, the distance to the vocabulary cards was 50 cm, or 70 cm - 133 test images
object recognition	- 18 objects (2D and 3D), images were taken from different perspectives - $143,945$ SIFT features were extracted in total	- same or similar light conditions to the training set - 100 test images	- different light conditions[5] and perspectives to the training set - 100 test images
face recognition	- 10 individuals (aging from 20 to 50) - 116 facial images (without expression) in total	- images were taken under the same condition as the ones in the training set - 225 test images	- different light conditions and facial expressions - 287 test images

7.2.1.2 Results

After the experiments, we calculated F1 scores ($2 \cdot \frac{precision \cdot recall}{precision + recall}$) for individual recognition modules on each test set. The highest F1 scores for the tests are summarized in Table 7.2. On the basic test sets, all recognition modules could achieve relatively high scores, whereas

Table 7.2: Highest F1 scores for individual image analysis module tests.

	basic test set	complex test set
OCR [%]	79.7	22.7
object recognition [%]	97.8	91.8
face recognition [%]	88.6	27.0

only object recognition could achieve competitive results on the complex test sets.

To summarize the results of the preliminary experiments, the object recognition module showed robustness to different environmental conditions unlike the other two modules. Although the complex test set for object recognition might be relatively easier than the other two hard sets, it is also true that the object recognition module holds an advantage due to the robustness of local features [TM07]. On the other hand, other recognition modules showed sensitivity to environmental changes. Based on an empirical comparison of the confidence scores between image analysis modules, the scaling factors of each image analysis

[5] Since we took pictures in a museum, the lighting could not be changed much.

modules in the cognitive merger are fixed as follows: $s_o = 0.8$, $s_f = 1.0$, and $s_t = 1.3$ (refer back to Section 7.1.6).

7.2.2 Cognitive State Recognition

Next, the accuracy of the proposed cognitive state recognition method is evaluated. In this experiment, we asked participants to perform three types of cognitive tasks, which correspond to the cognitive states in our AIA framework, i.e., *image study*, *read text*, and *non-visual state*.

7.2.2.1 Experimental Settings

The participants performed the following tasks: 1) Read a one page document (X1) printout. 2) Read a one page document (X2) printout. 3) Study four different pictures. 4) Study another set of four different pictures. 5) Listen to a song without closing eyes. 6) Ponder something without closing eyes. For task 1 and 2, we used the single-column and double-column document also used in the previous chapter (see Figure 5.25). For task 5 and 6, although we told them not to close eyes but they might have corneal reflexes (involuntary blinking). Task 1 and 2 correspond to *read text*, task 3 and 4 correspond to *image study*, and task 5 and 6 correspond to *non-visual state*. Sample images of a subject performing the tasks are shown in Figure 7.8. After each task, the participants had to write down the

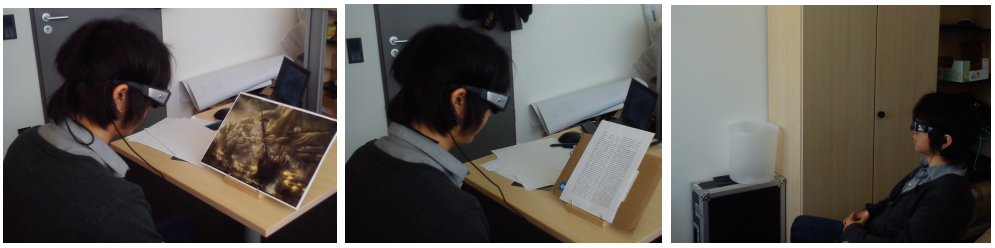


Figure 7.8: Sample images of a subject performing each cognitive task. Left: *image study*, middle: *read text*, and right: *non-visual state*.

summary of each task, for example, a summary of the document content or impressions of pictures. Therefore, the participants must try to understand the content rather than just browsing it without any intention. Note that this task is not a memorization task, since the participants only needed to describe their impressions freely. Four minutes were given for each task. The users were allowed to stop in the reading task only, if they reached the end of the page. The participants were told not to close their eyes during 5 and 6, and were also told to concentrate on the tasks.

Seven participants performed the aforementioned tasks and we recorded the scene image video and the gaze data. Using the recorded data, we performed two different types of cognitive state recognition experiments.

- Experiment I: Use 1, 3, and 5 for training and 2, 4, and 6 for testing.
- Experiment II: Use the first half of each recording for training and the rest for testing.

If the results of the Exp. I are comparable to those of Exp. II, it could be inferred that eye gaze movement patterns in those tasks are dependent on the cognitive processes, rather

than the instances of the stimuli. That is, the cognitive states could be identified correctly even if the instance of the target object changes (e.g., another unknown picture or text). Furthermore, if that is the case, it would show that listening to a song and ponder something can be categorized as one non-visual state.

The sampling rates of the eye tracker and the scene camera were both 24 fps. In both experiments, we extracted gaze features from the training data by sliding the local time window every $n = f_{total}/200$ frames, where f_{total} is the total number of frames in the video. Consequently, we extracted 200 feature vectors from each video. After training a multi-class SVM, classification tests were performed for every frame on the test data (by sliding a local window every single frame).

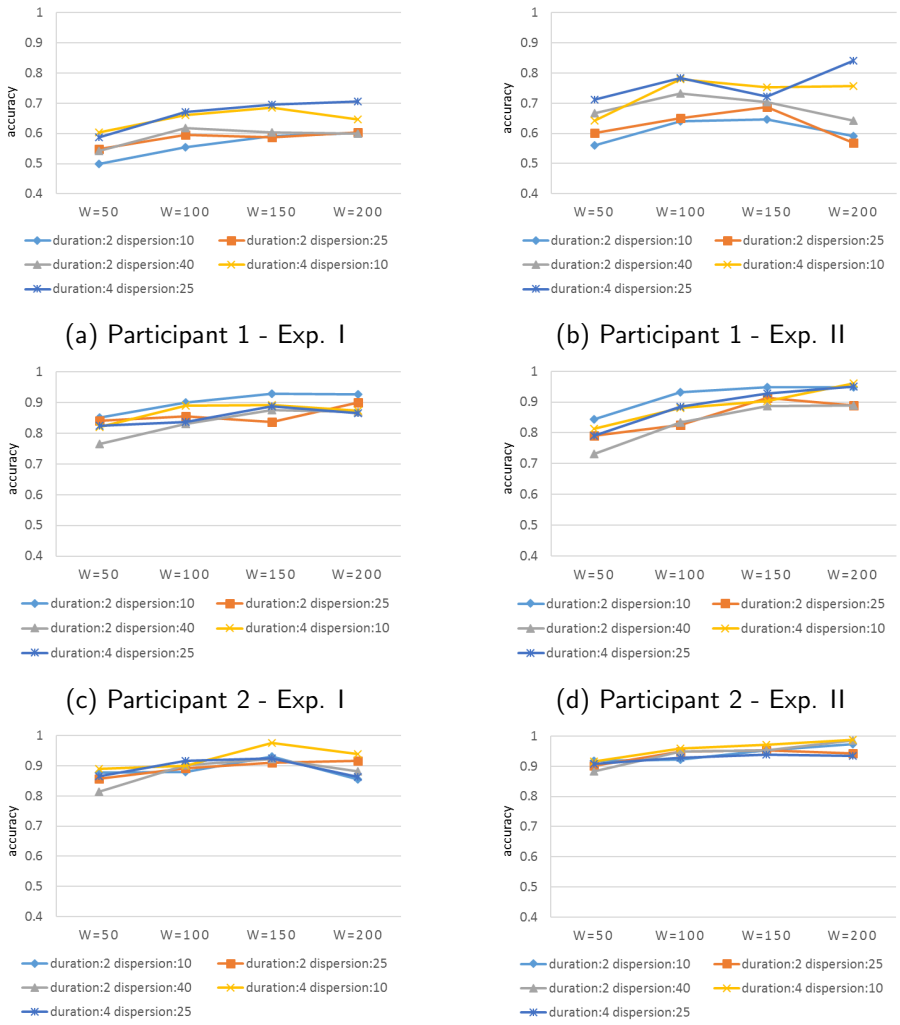
7.2.2.2 Results

The classification accuracy results from three representative participants are shown in Figure 7.9. In the graphs, I show the accuracy of different combinations of fixation duration T_{dur} and dispersion T_{dis} threshold parameters; T_{dur} is 2 or 4 (frame) and T_{dis} is 10, 25 or 40 (pixel) for different sizes of a local window W (from 50 to 200). The results from the other three participants are not shown here since they were similar to these results. It is observed that the longer the window is, the higher the accuracy of recognition is. This is very straightforward because in general, when the length of a window is long, one could obtain a more 'stable' feature vector. However, when concerning a real-time application, it is more preferred that the cognitive states can be classified with a shorter window similar to the interaction with the HMD in Section 6.1. Otherwise, the outputs from the system may have serious latency and the system cannot catch up with rapid transitions of cognitive states. The optimal parameter settings are different between the individuals (Later we test the performance when we do not optimize the parameters for individuals). There are two reasons for that. One is that the movement patterns during each of the cognitive tasks are slightly different. For example, some individuals read text very quickly, whereas the others read slowly. The other reason is that because of the instability of eye tracking, some users have noisy gaze outputs. When such noisy gaze outputs are obtained, the gaze from the eye tracker jumps from position one to another quickly and stable fixations cannot be not detected.

Overall, the recognition accuracy was fairly good. Even in the worst case (participant 1) within all seven participants, the accuracy was more than 0.7, which is far better than random classification. The results from this experiment suggest that the AIA strategy may be useful for user-attended content recognition.

For comparison between Exp. I and II, the results of each experiment with the parameter $W = 100$, $T_{dur} = 2$, and $T_{dis} = 10$ are shown in Figure 7.10. The results show that there are only small differences between the two experiments (I vs. II) for most of the participants. Thus, we may infer that the proposed gaze features are effective for classification of the cognitive states regardless of the stimuli or the instance of the target object (different documents or images) in each task. More surprisingly, "listen to a song" and "ponder something" can be classified successfully as a non-visual cognitive state, which supports the hypothesis that the eye gaze features extracted during these two tasks are similar.

Next, we also performed an experiment for evaluating cognitive state recognition using a generic classifier trained by other participants. Here, we investigated whether a generic classifier trained by a group of users could be applied to another unknown user. We performed



(a) Participant 1 - Exp. I

(b) Participant 1 - Exp. II

(c) Participant 2 - Exp. I

(d) Participant 2 - Exp. II

(e) Participant 3 - Exp. I

(f) Participant 3 - Exp. II

Figure 7.9: Experimental results of cognitive state recognition (Exp. I and II) from participant 1 to 3 (duration - fixation duration threshold T_{dur} (frame), dispersion - fixation dispersion threshold T_{dis} (pixel), and local window size W (frame)). In general, a longer local window can achieve a better result.

a classification test of each participant using a classifier trained by all the other participants (thus, leave-one-out). Figure 7.11 shows the results with the window size $W = 100$ and different combinations of T_{dur} and T_{dis}. We can see that the accuracy dropped when we use a generic classifier. However, they are still compatible for many participants. We may conclude that the eye movement patterns for these cognitive tasks are similar between users to some extent. This is a promising result since it shows that a trained classifier by a group of users may be applicable to an unknown new user.

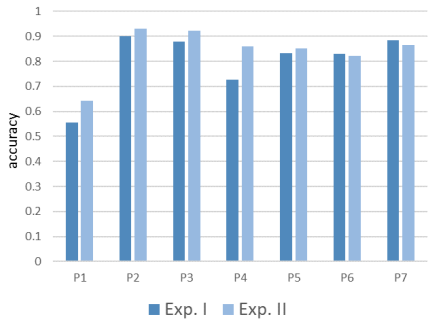


Figure 7.10: Comparison of Exp. I. and Exp. II. in cognitive recognition experiments (Participant 1 – 7). We can see there are only small differences between Exp. I and Exp. II.

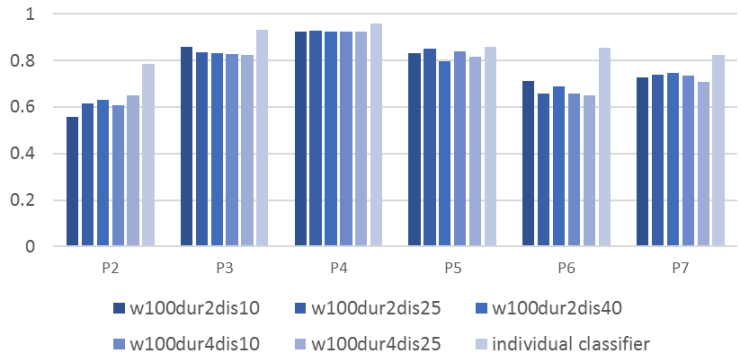


Figure 7.11: A generic classifier vs. individual classifiers in cognitive recognition experiments (Participant 1 – 7, w: W, dur: T_{dur}, and dis: T_{dis}). The right-most bars are the results from individual classifiers. A generic classifier can achieve comparable results to individual classifiers.

7.2.3 Recognition of User-Attended Content

In this subsection, the entire recognition framework of user-attended content is evaluated as a whole. The main challenge of the proposed system is to identify the content the user is attending to in a complex scene. For the evaluation of the proposed framework, we compare the two processing strategies (AIA and CIA). We selected three different scenarios for this evaluation.

- **Poster Browse**: Subjects read text and look at figures and images in a poster.
- **Factory Work**: Subjects read written instructions, look at objects of a factory system, and look at colleagues' faces.
- **Meeting**: Subjects look at attendees and listen to their speech. Subjects also read text on and look at a slide.

In these scenarios, several types of visual content are present in the same field of view of a scene camera. Thus, in order to understand the user attention more appropriately, a proper image analysis module must be executed on a proper local ROI image. These scenarios show

the significance of the proposed framework in such a complex daily scene for understanding user attention.

7.2.3.1 Poster Browse

The first scenario is "Poster Browse". In this experiment, we test the system when the user browses a poster, which contains text and images. Thus, text recognition and object recognition (2D image recognition) would be required in this experiment.

Experimental Settings We prepared a poster that explains local tourist attractions of a city and hung it on a wall as shown in Figure 7.12 (left). Figure 7.12 (right) shows the text areas and the image areas in the poster. It has nine text areas and nine image areas in total.

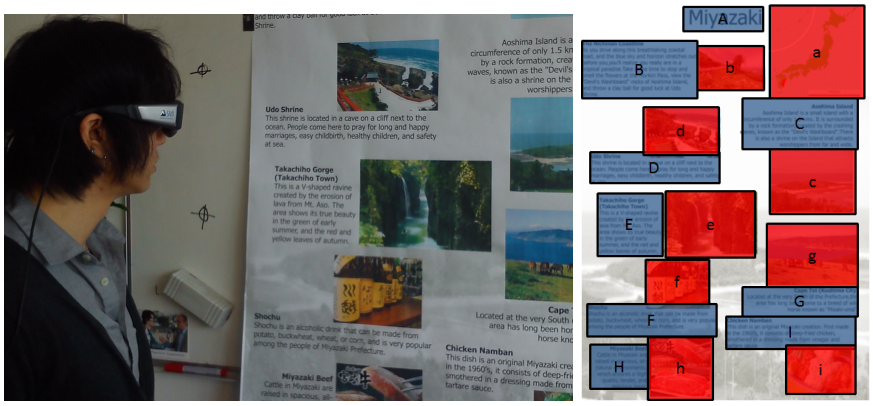


Figure 7.12: Poster Browse scenario. Left: a subject browsing the poster. Right: the poster used in the experiment. Blue rectangles depict the text regions and red rectangles depict the image regions.

As a task of this experiment, the participants were told the following:

- *You are going to a local city for a vacation and stay there for some days.*
- *You have to make a travel plan using the information presented in the poster.*
- *It's up to you how many days you stay there.*
- *You have four minutes to browse this poster. You cannot quit browsing it before four minutes.*
- *After four minutes, you have to write down your travel plan.*

We recorded the data of six participants and labeled them manually as shown at the bottom of Figure 7.13. Labeling was done by checking at which area of content identity the user looked at in each frame. For example, if the user gaze of frame number i is on the area of the image identity X, the label of i is X. If the user gaze is on the text identity Y, the label is Y. The label of the text is based on an area, not based on a word neither a character, since it is quite difficult to find out on which word or character the user gaze

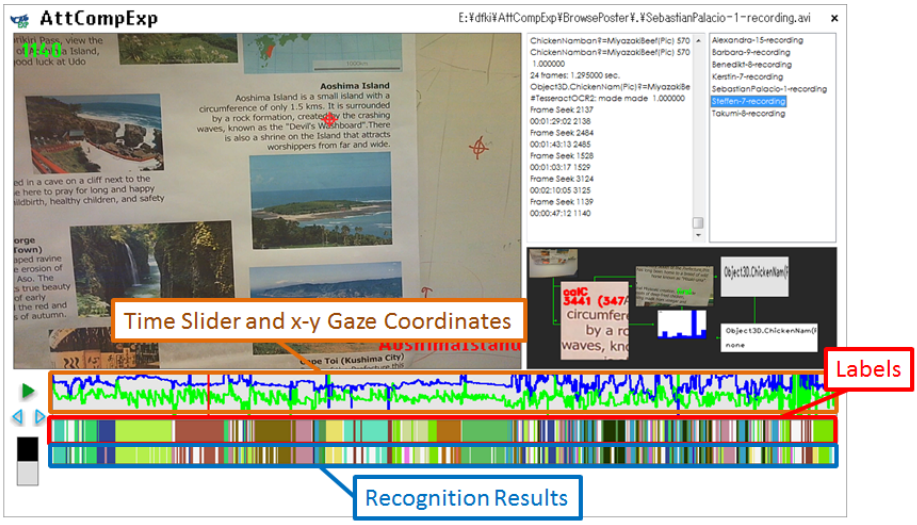


Figure 7.13: Screen shot of the experiment software. In the main window, the scene image and the gaze position (a red crosshair) are shown. Below the image, content labels for the whole video and the recognition results are represented by different color bars.

is located. In Figure 7.13, the label of this example video frame is text "C", since the user gaze is on text identity C (see also Figure 7.12).

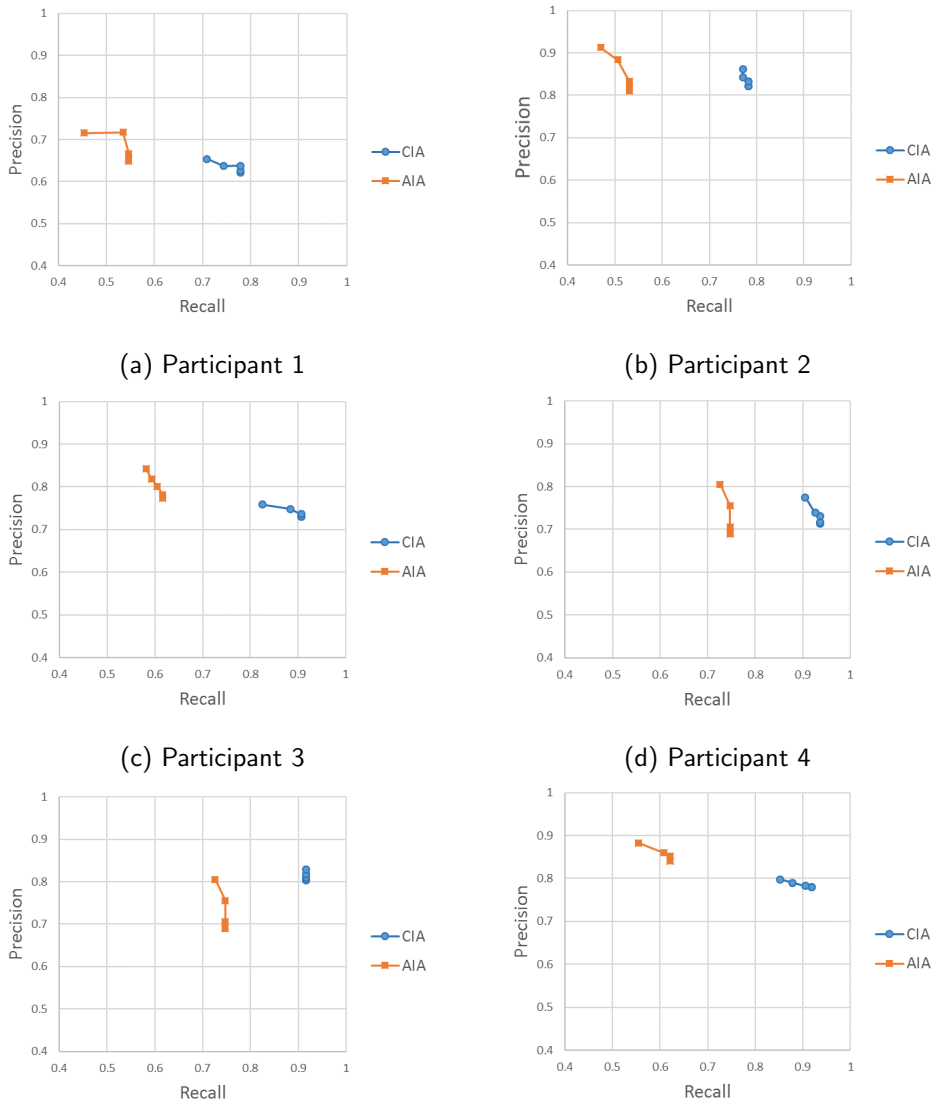
After labeling all the frames of recorded data, we generated label sequences L_S in order to obtain the user-attended content ground truths. In this evaluation, we ignored the labels that do not repeat more than nine frames, in order to remove noise labels that are not regarded as user-attended content. Thus, we grouped the continuous frames that have the same labels into one label sequence. We evaluate the system based on the label sequences $L_S = \{s_i, ..., s_n\}$. The recall score R and precision P is computed as follows:

$$R = \frac{R_{correct}}{L_{total}}, P = \frac{O_{correct}}{O_{total}},$$

where $R_{correct}$ is the number of label sequences correctly retrieved by the system, L_{total} is the total number of label sequences, $O_{correct}$ is the number of sequences correctly output by the system, and O_{total} is the total number of output sequences. When an output sequence is overlapped with a label sequence and the labels are the same, that is regarded as correct recognition. That is, if the beginning frame number of the ith output sequence o_i is between the beginning frame number and the end frame number of jth label sequence s_j and $s_j = o_i$, the o_i is a correct output and s_j is recognized correctly. For OCR output evaluation, if a text region contains the output word, it is regarded as correct. Note that the labels only contain user-attended content and do not contain cognitive states since it cannot straightforwardly be determined which cognitive state the user is engaged in from the recorded gaze data alone. In this experiment, we only check whether the content the user attends to is correctly recognized.

Before the experiments, we also recorded gaze data while the participants were performing each cognitive task (*image study*, *text read*, and *non-cognitive state*) with other data in order to train the cognitive state classifier for each participant. Thus, in this experiment, cognitive state recognition in AIA is done by the individual classifier for each participant.

Results Figure 7.14 shows the precision-recall graphs for six participants in the poster browse scenario. The graphs are drawn by changing the threshold value for the cognitive merger



(a) Participant 1

(b) Participant 2

(c) Participant 3

(d) Participant 4

(e) Participant 5

(f) Participant 6

Figure 7.14: Recognition results of user-attended content recognition in the Poster Browse scenario.

from 0.5 to 0.9. From these results, we can see that higher recall is expected with CIA, whereas higher precision is expected with AIA in general. Strictly speaking, they are not comparable since CIA cannot reach to the same precision rate as that of AIA even if we increase the threshold value. However, from the results, it is inferred that the precision upper-bound of AIA would be higher than CIA, whereas the recall upper-bound of CIA would be higher than the former. Because the recognition with AIA checks what activity (cognitive state) the user is doing first, it may sometimes incorrectly recognize content when the system misclassificates the cognitive state. On the other hand, CIA sometimes incorrectly

recognizes when the gaze position is close to the border of two different areas, such as the border between the text area D and the image area d (see Figure 7.12). When the cognitive state is recognized correctly, this type of misrecognition does not occur since an appropriate image analysis module can be selected. Not surprisingly, non-visual cognitive state was also recognized frequently during the task. However, it did not cause any misrecognition since facial images were not present during the entire recordings.

Apart from the evaluation based on precision and recall, AIA has an advantage from a computational cost point of view. Since the CIA runs all the image recognition modules all the time in parallel, the computational cost is quite high; thus, it sometimes causes CPU overheads. The laptop[6] we used in this experiment has a CPU with dual-core (Intel Core i7-2640M 2.80GHz). On overall average, the processing time with the CIA was 73 msec per frame, whereas it was 53 msec with the AIA. This result implies that if we extend the CIA system by adding another image analysis module, the processing units would suffer from too much consumption of processing power. In this experiment, we had three processes running in parallel with the CIA. The longer processing time with the CIA was because of the computational overheads (three parallel processes on a dual-core processor). If we use a processor with higher performance, this problem may be solved. However, the AIA still has an advantage in a mobile scenario where we cannot always rely on high performance machines. Overall, the recognition accuracy of CIA was better than the AIA. However, the performance concerning both recall and precision of the AIA is still comparable to the CIA. Furthermore, it keeps lower computational cost.

The evaluation above does not concern the recall of individual words. Next, we investigate how many words could be retrieved while the participants read text. Reading speed of a user is usually faster than the OCR process. Therefore, it is unavoidable that the system misses some words. Figure 7.15 shows the recall rate of individual word recognition and examples of retrieved text when the threshold value of the cognitive merger is 0.5 (which results the highest recall). The results show that 20% - 45% of the text are

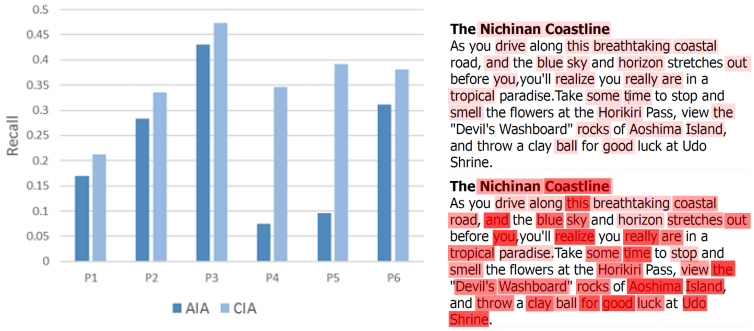


Figure 7.15: Evaluation of recall in text recognition regarding words. Left: the recall rate of word-level recognition. Right: examples of recognized word in the text region C. The highlighted words are the retrieved words. The top is from a single participant, whereas the bottom is the aggregation of the entire participants.

correctly retrieved by the system. AIA was slightly worse than CIA with P1, P2, P3, and P6. However, it was drastically worse than the other with P4 and P5. The reason of that was that the gaze patterns of these participants were relatively noisy and the cognitive states

[6]We used a mobile laptop for the experiment because we consider a mobile scenario.

were not classified correctly, especially during reading text. Thus, many words remained unrecognized. Also, on the right side of Figure 7.15, an example of the recognition results for the text area C is shown. The top is a result from a single participant and the bottom is the aggregation of all participants. One can qualitatively see in this result (from a single participant) that approx. 30% of the words were recognized, which might be sufficient for identifying the context. In other words, even if so-called *stop words* (e.g., "and", "you", "the", etc.) cannot be recognized, identification of other specific words in the context (e.g. the word "Nichinan" in Figure 7.15) would be useful enough for identification of the semantics of the content.

One approach to increase recall rate would be to send an image of whole text region to the OCR module, instead of sending an image containing only a region of the fixated word. For such extension, we could consider to use the cognitive state recognition. For example, if a reading state is identified in the beginning of reading paragraph, the system can recognize the entire paragraph in advance before the user reaches to the end of the paragraph.

7.2.3.2 Factory Work

The second scenario is "Factory Work". In this scenario, 3D objects, written instructions, and colleagues' faces would attend in users' views as shown in Figure 7.16. We selected system maintenance as a task for this experiment. In a factory, workers are often required to maintain systems under guidance of colleagues or written manuals. In this experiment, we asked seven participants to complete a system maintenance task for a liquid bottling station (see Figure 7.16 (middle)).

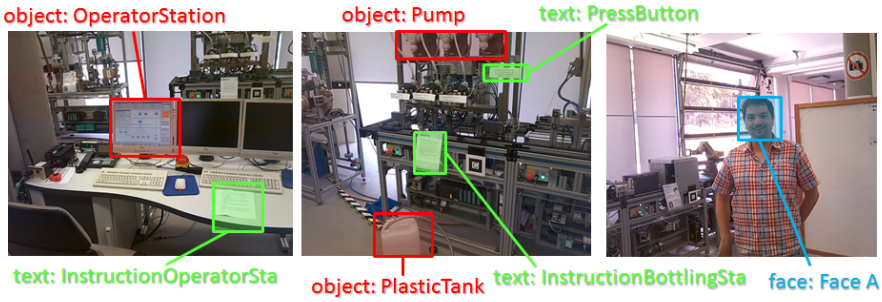


Figure 7.16: The setting of the "Factory Work" scenario. Left: The operator station desk. Middle: Bottling station. Right: A colleague standing by.

Experimental Settings The maintenance process was as follows:

1. The participant was given an instruction sheet ("text: InstructionOperatorSta", referred to in Figure 7.16), which describes the process to be completed at the operator desk.
2. He or she followed the written instruction and completed the tasks indicated there.
3. After that, he or she walked to the bottling station and followed the instruction sheet attached there ("text: InstructionBottlingSta").
4. Repeated the process again for another bottling module.

In addition to the instruction sheet handed to the participant, additional instructions were also attached to individual components of the system in order to guide him or her. Furthermore, when the participant got into trouble and could not solve the problem by himself or herself, he or she might look at the face of his or her colleague, who was standing by near the participant in order to require assistance.

For each participant, five textual instructions, five 3D objects, and one colleague were involved in the maintenance process. Note that a colleague in the experiment was alternated for each participant. In total, seven persons took part in as colleagues in the factory. In this experiment, we used a generic cognitive state classifier trained in the previous experiment (see Section 7.2.2).

Results In Figure 7.17, we show the best and the worst cases (participants) of each AIA and CIA result. The evaluation was done as the same criteria used in the previous experiment and the graphs are also drawn similarly. Similar to the previous experiment (Poster Browse), the CIA achieved higher recall compared to the one with the AIA but the AIA can still hold a comparable performance. As one can see in the results, there was a large gap between the best and the worst cases. Again, this was mostly because of failures of the eye tracking. When gaze positions calculated by the eye tracker were not stable, the recognition results (both in cognitive states and visual content) were also unstable.

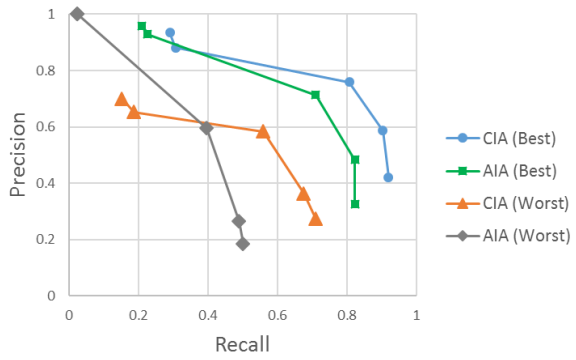


Figure 7.17: Recognition results in the Factory Work scenario. The best and worst cases (participants) are shown. There is a gap between the best and the worst participant.

Next, we discuss the number of occurrences of each content in the label (ground truth) data and the number of true positives retrieved by AIA/CIA. In this experiment, the number of occurrences of individual visual content differed because the participants needed to attend to some instructions or objects several times. For instance, OperatorStation was attended more frequently than the other 3D objects by many participants. In order to investigate the recall rate for individual visual content, we summarize the occurrences and true positives in Figure 7.18. The bars of AIA/CIA represent for the number of true positives retrieved by each recognition strategy. It can be said that the recall rates of AIA were slightly better with objects but not with text and faces. The results show that cognitive state recognition in AIA could not successfully identify the participants' reading and listening behaviours (During assistance from a colleague, the participant had to listen to him). In this experiment, we used s generic cognitive state classifier with local window size $W = 100$. This means that if the cognitive states transit within 4 second (100 frames), the state would not likely be identified.

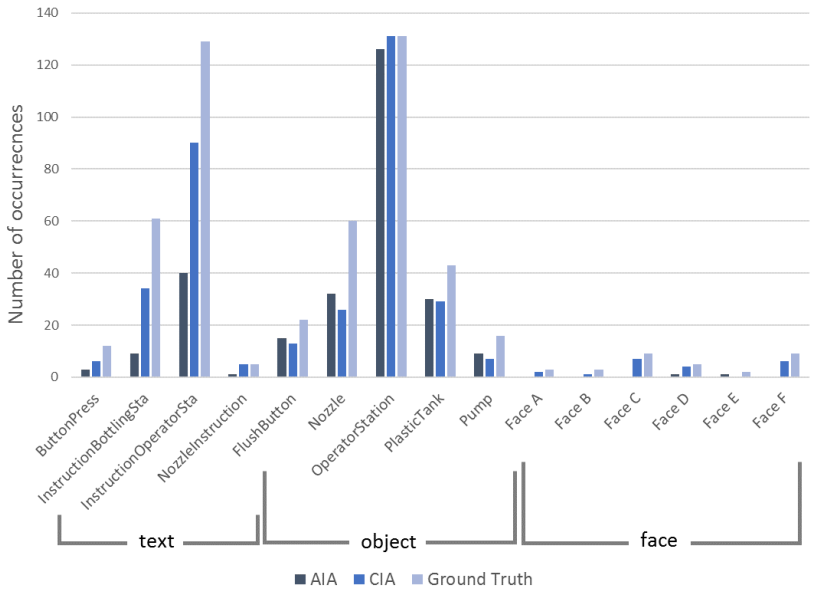


Figure 7.18: Number of occurrences in ground truth and the number of true positives retrieved by each strategy in the Factory Work scenario. The AIA is better only with objects. The CIA outperformed the AIA with text and faces.

We observed that some actions were completed within short periods in the recordings, which in turn, the recognition framework missed some attended instances. Especially, when the participants needed instructions from colleagues, they just glanced the faces rather than looked at the faces to listen to the guidance. Thus, faces were not successfully recognized by the AIA.

7.2.3.3 Meeting

In the "Meeting" scenario, we recorded test data when people have a meeting. In this scenario, the participants were attending to *text* on slides and *faces* of speakers. For the test data of meeting scenario, we arranged one hour meeting, where people discussed "how to write an abstract in a scientific paper". Ten participants were present in the meeting and the gaze data was recorded from four different participants. The length of each recording was from three to four minutes. Four recordings were used for this evaluation. Again, we used a generic classifier from the previous experiment (Section 7.2.2) for cognitive state recognition in AIA. The same evaluation criteria as the other use-attended recognition experiments was applied.

Figure 7.19 shows the overall precision and recall results of this experiment. The recall and precision scores are the averages of all four participants. Compared to the "Poster Browse" scenario, the recall rate was almost half, which was mostly because of the inefficiency of face recognition in a difficult case (as observed in Section 7.2.1). In this experiment, profile faces could not be detected or recognized since they were not included in the training dataset. Also, facial expressions in a meeting usually change dynamically, which also makes the recognition tasks difficult. The accuracy of the OCR was also poor when the viewing angle was not perpendicular as shown in Figure 7.20.

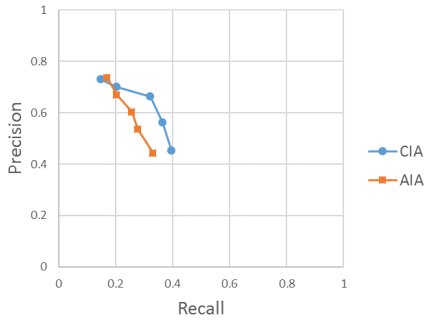


Figure 7.19: Recognition results in the Meeting scenario. The overall results were not as good as the other scenarios (such as the poster browse or the factory work). Especially, the recall was very low in both AIA and CIA.

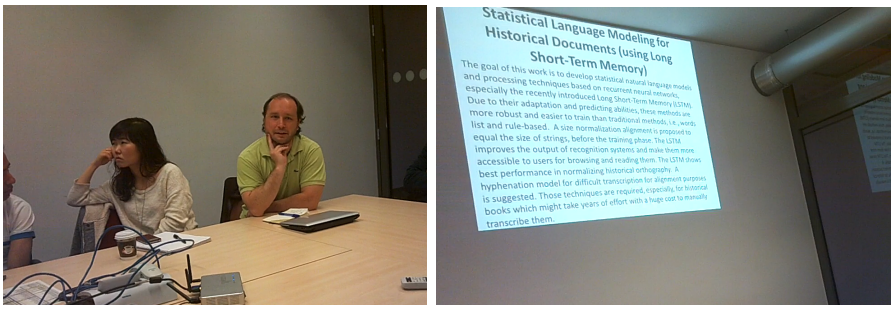


Figure 7.20: Example images of the Meeting scenario. The viewing angle was not perpendicular in many cases.

Unlike the "Factory Work" scenario, it was observed that many participants stably fixate on faces when they listen to the others (The gaze positions located on face regions for a certain amount of time). Such activities were classified as a non-visual cognitive state and successfully triggered the face recognition module. In this experiment, we chose to focus on the gaze and attention of a listener than that of a presenter. It is also inferred that the gaze patterns of a presenter would be different from those of a listener.

7.3 Discussion and Outlook

7.3.1 Recognition Performance of the System

The results of preliminary experiments showed the gap in reliability between object recognition and other recognition engines. Although the gap may be bridged by adapting scaling factors in the cognitive merger, employment of robust recognition engines is of importance for building a more robust recognition system. Since the methods we adopted in this system are primitive ones, the performance of the whole architecture would be boosted by using other state-of-the-art approaches.

In terms of wearable computing, it is also significant to reduce the computational cost. By recognizing a state of the user from multiple perspectives, for example recognition of the mental cognitive state of the user, it can offload the computation of the whole user-attended

content recognition process, as shown in the experiments.

In the experiment, though the AIA could not outperform the CIA, it showed an advantage on computational cost and speed. For possible extension, one might try to use the result of cognitive state recognition to weight the confidence scores in the cognitive merger instead of selection the image analysis module.

7.3.2 Cognitive States and Image Analysis

Since gaze movement patterns are closely related to visual structures of things, interpretation of gaze movement patterns as a cognitive state supports recognition of visual content, as we presented in the experiment. A typical example is a *read text* state, which frequently implies presence of textual content in the user's view. However, it is also not always the case that a cognitive state can directly be connected to a visual content, especially when the user is performing a non-visual cognitive task. In our framework, face recognition is applied in order to recognize faces of speakers. The experimental result from the meeting scenario showed that when a person is listening to a speaker, it can be classified as a non-visual cognitive state.

Although reading activities could effectively imply presence of text, text may also be present even when the user gaze is not drawing obvious reading patterns. A typical example is text on signs. Such types of text usually do not have a typical structure but rather they are randomly positioned in a scene. For recognition of user attention on such text, the proposed system needs to be extended.

There are more cognitive states than we handled in this work. Particularly, the states in this work could be said rather passive and the subjects did not behave actively (such as speak, write, and others). The eye movements during such activities are very different from the ones we used. Thus, we need to treat those states differently when we also handle a more pervasive daily scenario.

As previously stated, the proposed system could be extended by integrating other image analysis modules and cognitive states. One challenge would be an integration of video retrieval. Although studies show the difficulty of identification a user state of watching a movie [Dor+10], a specific type of movie could draw a similar gaze features. Furthermore, a specific natural scene such as sports could also be identified using eye movements.

7.4 Conclusion

In this chapter, I presented a method for recognition of user-attended visual content when multiple types of visual content are present in the scene. In everyday environments, objects, text and other various types of visual content are present. Therefore, the recognition of user-attended visual content is complicated and difficult. By taking user eye gaze into account, a computer system can infer important content for the user at the moment. The recognition of user-attended content provides a powerful clue for inference of user context.

The experimental results showed that different types of visual content can be recognized using the proposed comprehensive framework. Although user-attended content can be recognized without concerning what the user cognitive state is, it needs to analyze all the possible content classes by applying multiple image analysis modules simultaneously, which consume a lot of CPU power. By selecting an image analysis module based on the

cognitive state recognition result, one can reduce such computational cost with keeping comparable accuracy. In the experiment, we have seen three types of everyday scenarios where each scenario has a different difficulty level for VCA. It is still challenging to apply the system to completely uncontrolled environments as shown in the "Meeting" scenario, though it showed the feasibility in some scenarios such as "Poster Browse" and "Factory Work".

Recognition of user-attended content has potential for several applications. In a user assistance domain, a computer can provide the augmentative information of the content the user is interested in at the moment by combining information visualization devices such as wearable displays. Another scenario is a memory aid system that recalls the user with specific information which has previously been attended to by him or her.

Recent evolution of wearable sensing devices facilitate the applications of technologies in everyday life and daily activities. In the following chapter, I present a couple of applications where such user-attended content recognition plays important roles.

Chapter 8

Application Areas and Scenarios

In the previous chapters, we have seen several methods for analyzing user visual attention in everyday environments, especially methods for recognizing the visual content that the user attends to. This chapter summarizes various types of applications which utilize the proposed use-attended content analysis.

8.1 Museum Guide 2.0 and Talking Places

One can consider many scenarios where proactive information presentation is beneficial for users. Museum or city visits are typical cases of such scenarios. A human guide is needed for a visitor of museum or city so that he or she can enjoy the attractions entirely. Without guidance, the visitor may miss many attractions.

The principle idea of *Museum Guide 2.0* and *Talking Places* is to develop an intelligent computer guide system that can mimic a human guide. A good human guide would carefully monitor the visitor's attention to objects or attractions and attentively assist him or her by explaining them. The proposed machine guide monitors the visitor's eye movements and detect AG on objects. Accordingly, the system presents additional information of the object.

Images of Museum Guide 2.0 and Talking Places are shown in Figure 8.1. The information can be presented via a headphone or in the HMD. In the content DB (see Section 3.4), the images of objects, image features, and object meta-information are registered. When AG to any registered object is detected, the system presents the meta-information to the user. If the meta-information contains pictures or videos, they are presented in the HMD. An advantage of these systems is that information to be presented is filtered according to the user attention, unlike ordinary AR applications. Therefore, the system can present information obtrusively. Furthermore, only relevant information to the current user context is presented.

To evaluate the usability and the potential of this type of AR application, we conducted a user study in a museum scenario. In the following subsection, I summarize the user study.

8.1.1 User Study

23 users were asked to stroll in our experimental museum (mentioned in Section 4.1.5) with two different guide systems. One was Museum Guide 2.0 and the other was an audio player based traditional guide system. Audio player based museum guides are currently

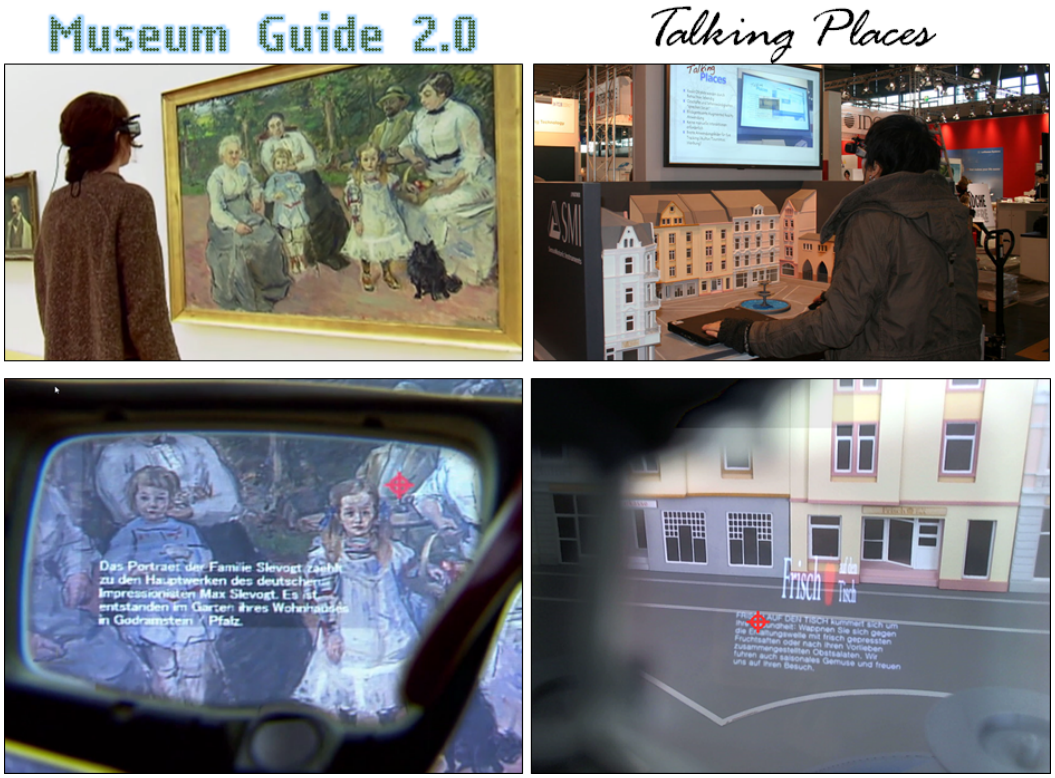


Figure 8.1: Example of AR visualization in Museum Guide 2.0 (left) and Talking Places with a miniature of city model (right). In the bottom images, views from the wearer's perspectives are shown. In the see-through HMD, information about the shop or the painting is presented.

used in many museums and therefore provide a good basis of comparison with existing technology. Usually exhibits have a tag number in front of them and the users have to select the corresponding audio track in the audio player to get the information about that exhibit. The same twelve objects from the previous experiment (in Section 4.1.5) were used and tags were put in front of them. The users freely walked in the museum and got information about the object with the audio player whenever they were interested in.

After finishing the first round with the audio guide, they were introduced to the eye tracker and the eye tracker was calibrated for each user. Then, the users were asked to take another round in the museum wearing the eye tracker. Whenever the users gazed at an exhibit and AG on exhibits was detected, Museum Guide 2.0 played a pre-recorded audio file to provide more information about the gazed exhibit. 11 users tested the audio guide in the first round, whereas the rest of 12 users tested Museum Guide 2.0 first.

When the users finished their round with both systems, they were given a questionnaire to assess different aspects of the system. First, a summary of user responses to the questions comparing the gaze based interface with the traditional audio player interface is shown in Figures 8.2 and 8.3 (The results of the rest questions are summarized later). Since the eye tracker used in the study has several hardware constraints (such as uncomfortable helmet, chin rest, etc.), we did not refer as "Museum Guide 2.0" in the questionnaire to judge the real potential of gaze based information provision. The results show that most of the users

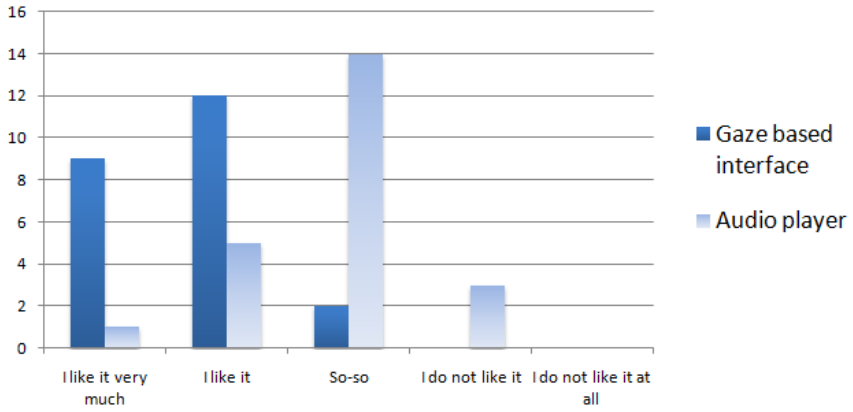


Figure 8.2: Responses in the user study for the question: How much do you like a gaze based interface (or a traditional audio player) for getting information? The vertical axis represents the number of users.

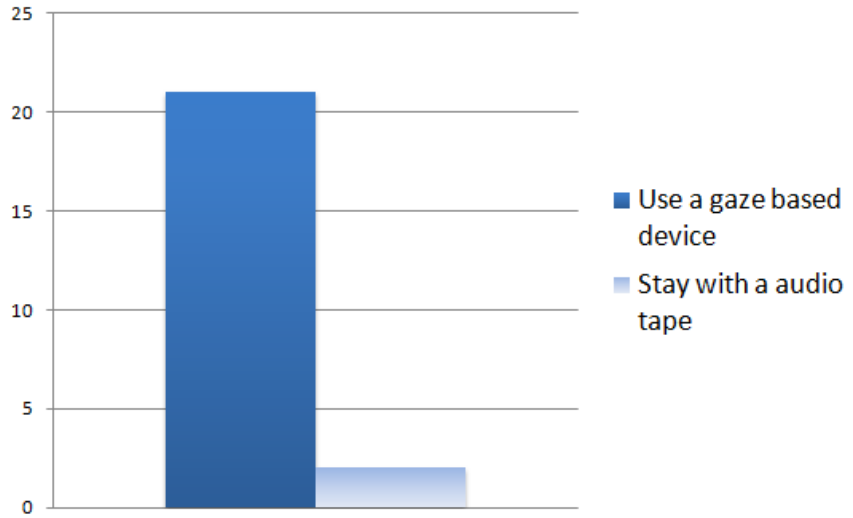


Figure 8.3: Responses in the user study for the question: What would you like to use when you go to a museum (Ignoring the hardware constraints)?

would prefer to use a gaze-based device as compared to an audio player when they go to a museum. Another interesting result was that although many users were satisfied with the traditional audio player, the *mean opinion score* (MOS) for Museum Guide 2.0 was 4.3 as compared to 3.2 for an audio player. In the rest of the questions, we also asked a couple of other questions for evaluating the system compared to the traditional one. The responses on other questions were as follows:

Q1. How much was the calibration process acceptable for you?

	Number of people
Totally acceptable	3
Acceptable	12
Neither acceptable nor unacceptable	4
Unacceptable	4
Totally unacceptable	0

Q2. Did you get the information against the object you want to know?

	Number of people
Perfectly	10
Mostly	11
Partly	1
Only Sometimes	1
Never	0

Q3. How often did you get the information against the object which you are NOT interested in?

	Number of people
Never	11
Only Sometimes	10
Sometimes	2
Frequently	0
Always	0

We can see that for most of the users (15 of 23) the calibration process was acceptable (Q1). The results of Q2 and Q3 show that recall and precision of the system was sufficient for most of the users (21 of 23 graded higher than 3 for both questions, i.e., 'Perfectly' or 'Mostly' in Q2 and 'Never' or 'Only Sometimes' in Q3).

Overall, the user study showed the promise of the proposed user attention-based machine guide systems.

8.1.2 Possible Extensions and Other Scenarios

Though the aforementioned museum and city guide systems only focus on objects, they can be extended to recognize attention to text, faces, and other visual content as I discussed in Chapter 7. For example, we often see many signs in a city and many textual explanations in a museum. By recognizing the text that the user reads, the systems can guide him or her more appropriately. Especially in an uncontrolled environment, such as city, where plenty of information resources (visual content) exist, extracting information from reading text is really beneficial because one cannot always rely on object recognition where a large database would be required.

Another extension is to generate meta-information automatically. Currently, the meta-information must be prepared manually in advance. However, it is not realistic to create information of all objects and shops in a city manually. One can extend the system by integrating meta-information generation. For example, information of objects and buildings

can be extracted from online information databases such as DBpedia[1].

The proposed system can also be used in other scenarios. Many users would benefit from this system in various types of professional work scenarios where special support would be helpful. As we discussed in Section 7.2.3.2, a novice factory worker can get assistance when he or she attends to a factory machine or a manual. It is useful when a novice worker wants to learn how to operate machines. For instance, the system can present a video manual in the HMD when he or she is stuck with a particular machine part.

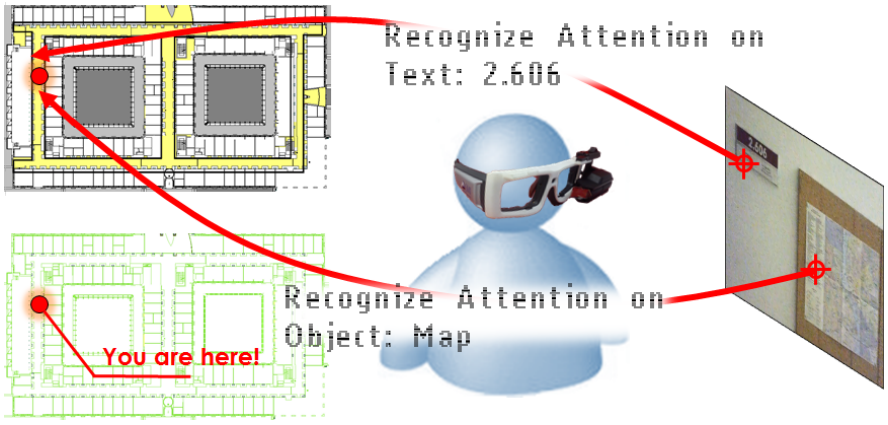
8.2 Location-Awareness Using User-Attended Content: AR Navigation

When the user's attention to the object *X* is detected, he or she must exist in a position from which the object *X* can be seen. This insight implies that one can use the user attentional information to localize his or her position. In [ST13a] and [Orl+14a], we presented applications that utilize the user's AG to infer the user position. In [ST13a], we focused on a hospital premise where floor structures are often complicated. There we discussed that the user position can be localized using signs and objects in a hospital. Similarly, in [Orl+14a], we focused on an evacuation scenario, where we tested the gaze-guided OCR in a simulated unusual case such as low-lighting or smoke (in case of fire) conditions. The gaze-guided OCR did not perform well in a low-lighting condition because of low quality of images, whereas it performed relatively good in a smoke condition. Although these tests were done in a simulated case, it showed the potential of wearable attention-based navigation system. Figure 8.4 illustrates the proposed localization system used both scenarios. Using the user-attended content recognition modules, we can recognize the text or the object the user is looking at. For the user position localization using object or text recognition, we need a priori knowledge of object or text locations. I developed a tool for managing annotations of floor maps as shown in Figure 8.4b. Using this tool, floor managers can easily annotate objects and text (such as letters on a doorplate) with positional data. Once such locational annotations are created, the localization system can automatically estimate the user position from recognized attended content.

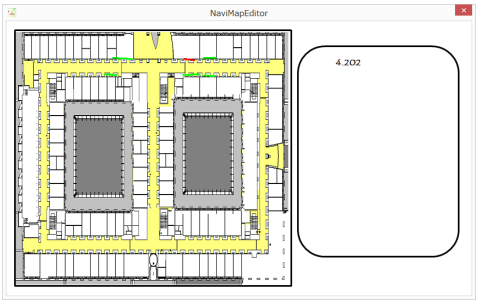
There are two granularity levels of localization. With the coarse level (used in [ST13a]), one can estimate a *fuzzy* user position. Suppose a user is in a hospital. When his or her attention to a $sonograpyDevice$ is detected, it can be inferred that he or she is in a $examinationRoom$. One can also fuse results of attention recognition for navigational reasoning: $sonographyDevice + doctorFace \Rightarrow examinationProcess$. This way the inference of the user location can be done by applying predefined rules.

On the other hand, we also consider the fine granularity level where the system estimates the user's position as coordinates in a map (used in [Orl+14a]). As I discussed in Section 4.1.4.2 and 5.2.2.3, one can estimate a pose of an object in a scene image of the eye tracker, which also inversely determines the user position in the space. When we use explicit markers such as AR markers as shown in Figure 8.5, we can expect relatively high accuracy for user localization compared to SIFT-based object recognition (right). However, in terms of user attention, users might not attend to markers unless they have a specific reason for that. Importantly, we need to consider which object or text we use to localize

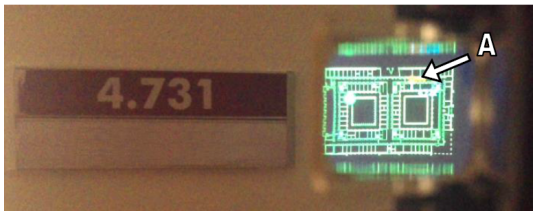
[1] http://dbpedia.org/



(a) Proposed localization system.



(b) Tool for managing locational annotations.



(c) Navigation in the HMD. 'A' is the current user position.

Figure 8.4: User position localization system using an eye tracker and use-attended content recognition.

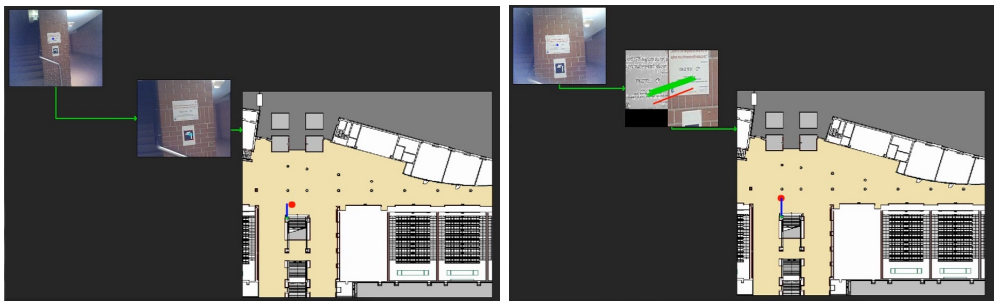


Figure 8.5: Localization using an AR marker (left) and an object (sign) (right). Red dots represent the user positions in the maps.

the user position. We have to select objects or text which more likely draw user attention. Navigational signs such as door plates may be good in the context of localization because the user reads written text especially when he or she is looking for a room or place.

8.3 ERMed – Erweiterte Realität in der Medizin

As I discussed in previous chapters, professional work benefits from attention-aware work-support systems. A medical scene is one of such areas. We developed a system called *ERMed* (*Erweiterte Realität in der Medizin* in German) which augments medical work using multi-modal computational devices including a wearable eye tracker [Web+13; Son+13]. Figure 8.6 shows an overview of ERMed. A doctor wears our eye-trackable HMD. During an examination process, he or she gazes at the patient's face. The system can act as an external brain that recognizes the patient's face and retrieves the name and the information of the patient from the previous records. The information is presented in the HMD, which supports a smooth examination process.

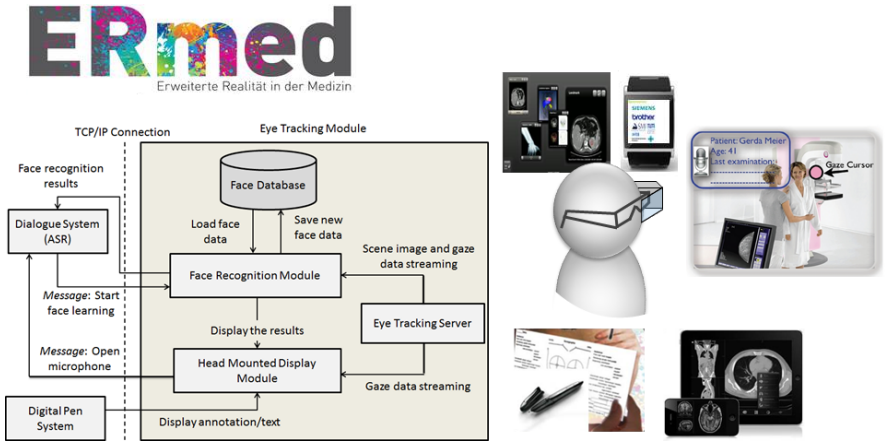


Figure 8.6: ERMed architecture. A doctor wears the eye tracking spectacle during an examination process. Multi-modal input devices such as a digital pen are integrated for various supports.

Using a digital pen, one can easily digitize the patient's record written by the doctor [Web+13]. The input data is seamlessly visualized in the HMD so that the doctor can confirm that the data is properly input. We also have a microphone eye-con (for eye-con, refer back to Section 6.1) in the HMD. When this eye-con is activated, the system opens an automatic speech recognition (ASR) engine [Sch+13]. The doctor can give speech for several commands: *"Learn a new patient, Gerda Meier."*, *"Show the axial video."*,...etc.

The ERMed showed the potential of gaze-based interaction in combination with multi-modal input devices. It effectively leverages the doctor's capabilities of recalling the specific context by virtual augmentation.

8.4 Attention-Driven Augmented Document

Eye tracking is also widely used for enhancing reading experiences [Bie+09]. Using the method proposed in this thesis, we can also augment a document paper according to reader's eye gaze behaviours. For example, when the reader wants to refer to a glossary, he or she can look at the HMD (Section 5.2.2.3). With another visualization method, the reader can see a translation which is dynamically presented in the HMD. A great advantage of eye tracking is that the computer can infer his or her reading state. When the reader is

stuck with understanding the meaning of word, a long fixation can be observed nearby the word [Ray98]. Thus, triggering translation provision by attentional gaze on a particular word can be considered as a reasonable function.

In the user study (Section 5.2.3.4), we saw that many participants had appreciated the proposed translation system. Normally, we have to search for the word from a dictionary when we want to look up the meaning of a word. Unlike such a traditional method, users can keep reading without being disturbed by translation presentation. However, we also found that the users may easily forget what they learn if they use the proposed system. This result teaches us that one has to consider how to design a system in order to adapt the technology in learning scenarios.

Recently, several techniques are developed for eye tracking on tablet devices [Han+12]. Since eye tracking becomes more easily available also on handheld devices, the potential of market for reading assistance applications also grows. In these growing fields of eye tracking devices, we should consider what we can deliver to the public for enhancing people's life. For example, we can integrate a framework of reader's comprehension inference into reading assistance systems in future [Kun+13b].

8.5 Gaze-Triggered Scene Text Translator

I presented a translation system also for natural scene text (Section 5.1). Similar to the document one, the user can see the translation dynamically in the HMD. Unlike other ordinary handheld translation applications on a tablet or a mobile phone, the user does not need to hold the device or tap the display to get translations.

As a user study, we compared the proposed gaze-triggered translation system with an ordinary mobile phone translator application where the user has to tap a screen to trigger translations for a particular text region. We asked 10 users to test two translation systems with some scene text. Most of the users could succeed recognition process faster than the mobile phone one using the proposed gaze-based system. The result showed that when the user wears the device, the action for triggering translation functions can be done quite fast. Furthermore, gaze gestures can pinpoint the region of interest faster than hand gestures on a mobile phone screen. However, some users also reported that it was sometimes demanding to perform gaze gestures. Although the setting of this study was rather small, this comparison showed the promise of the gaze-triggered translation system.

8.6 Attentional Life Event Logger

Gaze direction displays the user's visual attention, which in turn shows an important region in a scene for the user. Thus, detection of AG on particular content tells us what is meaningful content for the user. We can log meaningful images or events in his or her life using an eye tracker.

8.6.1 Visual Diary – A Prototypical Application

By detecting AG on arbitrary objects using the method presented in Section 4.2, we build a new application prototype. When an AG event is detected, we store the scene image cropped by the eye gaze data. I show a screenshot of the *visual diary system* which provides

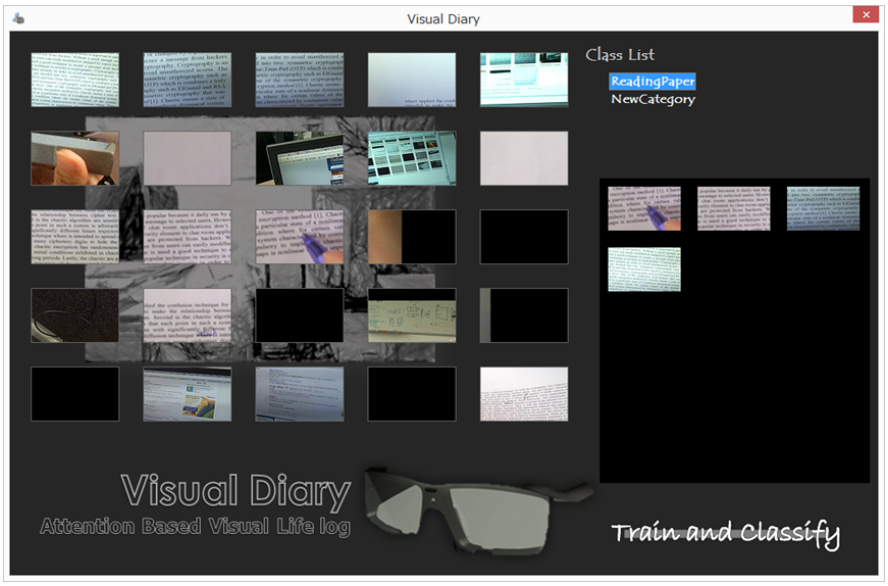


Figure 8.7: Screen shot of the visual diary prototype. The left section shows a collection of images containing user-attended content. On the right section, images of each class are shown.

the user with a collection of images of objects which drew the user attention in Figure 8.7. Images in the left section of the window are captured when AG events are detected. The user can edit classes (the right section) and add images to the class by dragging an image to the right section. Once several images are added to the class, the user can train a classifier (the right-bottom button). In this application, a graph-based semi-supervised classification technique [ZW11] is used to gather relevant images from the raw collection. Therefore, the system can automatically collect relevant images from the detected image collection by adding only a couple of representative images into each class.

Using this application, the user can log images which contain attended content in daily life. He or she can use this application as a diary for everyday visual scenes. Life-event logging is a trendy topic in the field of pervasive computing [O'H+08]. For logging events in daily activities, recognition of the content attended by the user would play an important role.

8.6.2 Possible Extension: Episodic Memory Management System

Where the person fixates shows what is important for him or her to accomplish the current task [Ray95]. If one analyzes sequentially on which thing in the scene the user is fixating, the inference of contextual event could be made. For example, if the user's sequential fixations on words are detected, one can infer that the user is reading text. As possible extension of attentional life event logging, I propose to classify user attention sequences on particular stimuli as a particular *episodic event*. In Figure 8.8, I present a layered model for constructing episodic events from observed eye gaze information in everyday environments. In this model, I construct layers of episodic memories starting from raw gaze data. In the first layer, a very primitive event such as *focus on a word* is generated. Then, those primitive events are assembled to encode a higher level of episodic event (*read text*).

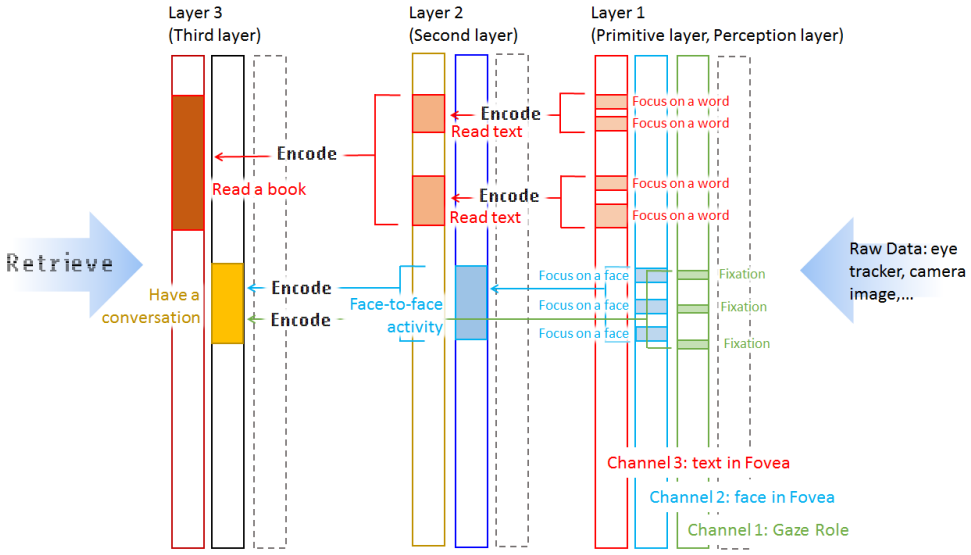


Figure 8.8: Layered model for episodic memory construction. We can build a layer of episodic events from low abstraction levels (right) to high levels (right). E.g., we encode a set of *focus on a word* events as a *read text* event.

Based on this model, we generate the episodic memory database of a user, which can be used to augment his or her memory. Such a log system could be used especially for those who suffer from dementia or Alzheimer's disease [LD08; Orl+14b]. The user can *retrieve* an episodic event from the database when he or she wants to recall previous events.

8.7 Summary

This chapter presented various types of applications which effectively use the proposed user-attended visual content analysis. Museum Guide 2.0 and Talking Places are typical examples where attended object recognition plays an important role. Furthermore, these applications can be extended to recognize text and may be adapted to other scenarios, such as factory work, where intensive supports are helpful.

User-attended content recognition is also a powerful feature for user position localization. Knowing that the user is currently looking at a doorplate, a map, etc., one can estimate where he or she is with two different granularity levels (fine or coarse).

Additionally, using the proposed attention analysis on reading, we can augment document papers in the HMD. Similarly, translations of natural scene text can be presented in the HMD according to the user attention. In both applications, gaze analysis on text reading is effectively used, especially to infer the target textual information that the user is interested in and a proper timing for presentation.

Attention analysis in daily scenes can also be used to capture important images that contain everyday meaningful content that draws user's attention. A *visual diary* shows a collection of images for user-attended content in a daily life. Furthermore, by analyzing fixation sequences, we can also construct the episodic memory database of a user which can be used to augment his or her personal memory.

Chapter 9

Discussion and Conclusion

In this thesis, I proposed attention-aware systems for everyday environments which recognize user-attended visual content in natural scenes and present the augmentative information of recognized content in a see-through virtual display. To recognize visual content that the user attends to, I also proposed several gaze-guided image analysis methods. Furthermore, I proposed gaze-based interaction methods for a see-through HMD. The proposed applications showed the benefits of attention-aware systems in practical everyday scenarios. In this final chapter, I summarize the proposed attention-aware systems and discuss the findings and future work.

9.1 Discussion

9.1.1 User-Attended Visual Content Analysis

In this thesis, methods for user-attended visual content analysis are proposed. Several experiments were carried out to evaluate the proposed methods. We discovered several findings from the experiments.

One of the findings is that we can improve the processing speed by limiting the image region. In general, when the image resolution is high, more processing time is required to process the image. Using eye gaze, we can reduce such computational cost. Limiting the image region also gives us another advantage. It is often a problem for object recognition that images contain cluttered backgrounds. When an image contains many irrelevant objects, object recognition is quite difficult. Since eye gaze typically shows where the region-of-interest in the image is, to focus on the object that draws the user's attention improves the recognition accuracy.

Another finding is that we sometimes need to filter out some noisy gaze positions in terms of attended content analysis. A fixation also sometimes locates on irrelevant regions shortly at which the viewer does not intend to look. Just because a fixation locates a certain region, it does not mean that the viewer attends to that. As presented in 4.1, we can ignore such irrelevant fixations by detecting attentional gaze (AG) on particular content in a scene. However, we still have another challenge. It is also true that long fixations does not necessarily imply that the viewer attends to that content. To differentiate a long fixation as attentional or as inattentional, we need further improvements on the proposed methods. We might need to seek an option to use other sensing devices such as an EEG to

track the user's mental state more deeply.

Furthermore, we have two open questions regarding improvements of user-attended VCA methods. In Chapter 7, it was discussed how eye movement-based cognitive state analysis can be combined to realize a comprehensive attended content recognition framework. In this chapter, eye movement features are used to classify the user's cognitive state. However, these features are not used in the image analysis processes. Previous studies showed that eye movements have a particular role for object recognition [NH10; KT06; Hen+03; Sch+09; Ray+09]. If so, we can hypothesize that such eye movement features may be used to improve the object recognition processes. For that, we need more fine-grained eye gaze localization on content; to analyze eye movements on an object, we have to map the scene eye coordinate to the object coordinate as I did in Section 5.2. Another open issue is that we need to deal with the image cropping size. Currently, the size is always fixed for each content type (object, text, and face) except for the gesture-triggered OCR (Section 5.1). However, as we saw in the experiments, this approach sometimes misses some important image features. We should develop a more flexible image cropping method to improve the recognition accuracy.

Last but not least, we should also consider the deliberateness of fixations to detect user attention as I discussed in Section 5.1. If the user knows how the system reacts depending on his or her eye movements, he or she may deliberately move the eyes so that the system can recognize that. Gaze gestures are one extreme case of such deliberate (explicit) gaze input. We should design a gaze-based system, with an insight of natural user behaviours, since an explicit manner could be cumbersome for the user.

9.1.2 Eye Gaze with a See-Through HMD

As an information presentation tool, I used an optical see-through HMD. Using an HMD, we are able to apply the proposed system to mobile scenarios. Furthermore, a see-through display allows the user to view the physical environment through the display. This see-through feature is very effective for mixed or augmented reality applications. In Chapter 6, it was shown that the user attention engagement with the display can be inferred using eye gaze analysis. Such inference is only feasible when we use an optical see-through display since the eyes always focus on the display when we use a video see-through one.

In the user study in Section 5.2, many participants commented that they appreciated dynamic translation presentation in the HMD. Furthermore, they completed reading tasks faster than the traditional one with the proposed system. Using a see-through HMD, information can be visualized directly near the eye gaze. The result from the study showed that a good mixture of virtual and physical environments can support reading tasks efficiently. A limitation with the current HMD device is that the field-of-view is quite narrow. We still need some more experiments with other HMDs which have wider field-of-view displays such as Epson Moverio.

I presented several interaction functions for see-through type HMDs using eye gaze (Section 6.1). Analyzing eye gaze in the HMD, we can proactively control display functions such as eye-con and automatic text scroll. I foresee that implementation of gaze-based interaction with a wearable display would play an important role in wearable computing technologies as the needs for the wearable devices such as Google Glass or Oculus Rift grow.

Estimating the user's eye gaze location in the 3D space, we can separate his or her

attention engagement in virtual or physical. Although accurate gaze depth value is hard to calculate (Section 6.2), the attention can be robustly estimated with three different focal planes. The experiment in Section 6.1 also showed that many people can effortlessly switch their focal plane from virtual to physical and vice versa. With the Current HMD, we cannot control the focal length programmatically; thus, the focal length of the HMD is always fixed. In the future, we want to use an HMD that allows adapting focal length to the user's gaze depth. However, if we adapt the focal length of the HMD, estimation of user's attention engagement using gaze depth becomes very difficult. We need to develop another approach for attention engagement estimation when we use such types of focal length adaptive HMDs.

9.1.3 Cognitive State Analysis and the Comprehensive Framework

I proposed a method for recognition of the user's cognitive state using eye movement features (Chapter 7). Although the classes are only limited to three (*read text*, *image study*, and *non-visual state*), the experimental result showed the potential of the proposed method. It is very hard to categorize such cognitive states in general, since they are always implicit. In everyday life, we are not necessarily aware of the cognitive state that we are in. Interestingly, the experimental result also showed that *listening to music* and *ponder something* can be classified as one *non-visual state* using eye gaze features. However, we need a more thorough study on a categorization methodology for such cognitive states. We can also include other classes such as *visual search* and *memorization* [Cas+09] for further analysis. Furthermore, we could also use these gaze features for classification of physical activities such as *writing* [Shi+14].

The proposed comprehensive user-attended content analysis framework showed the feasibility. Combining cognitive state classification results with image analysis modules, it can reasonably recognize the visual content that the user attends to. Although the attention-driven image analysis method did not outperform the brute-force continuous image analysis method in accuracy, it effectively reduced computational cost. This result is very important for further extension; we want to integrate other image analysis modules to recognize more various visual information resources. We can connect other modules easily using the proposed framework as long as the user cognitive state can be associated with the content. For example, we can integrate a video retrieval module for recognition of video content in which activity for watching can also be inferred from eye gaze [Shi+14].

We have another advantage of recognition of the user's cognitive state as discussed in Section 6.1. If the system knows the user's cognitive busyness, it can arrange a proper moment to present the information in the HMD. This way the cognitive state analysis can be used to present augmentative information more attentively to the user. We can also manage the information to present according to the cognitive state. For example, if the user is *searching* for something, we can present guidance information in the HMD or auditory such as in [Los+14].

As previously mentioned, analysis of cognitive states or eye movements can also be used to improve the image analysis. For example, one can introduce a feedback and feed-forward mechanism between image analysis and cognitive state analysis to collaborate the recognition task. If the system detects an explicit gaze pattern of *reading*, it can feed back the result to image analysis module to amend the recognition and vice versa.

9.2 Conclusion

The nature of human visual attention mechanism is still an unsolved problem for researchers, although we unconsciously control our eyes to direct visual attention in everyday life. However, that is why many researchers have been attracted for studies on eye movements and human scene perception. A great outcome of these studies is that we can benefit from eye gaze analysis in several computer applications.

In this thesis, I proposed attention-aware systems which analyze human eye movements and infer to which visual content the user is attending in several everyday environments. The experimental results showed the plausibility of two hypotheses introduced in Introduction.

- By combining image analysis technologies with eye gaze analysis technologies, a computer can recognize the visual content the user is attending to.

- Analysis of user eye gaze is useful to present information of attended content in an adequate way.

Developments of wearable eye trackers open up the opportunities to apply attention analysis approaches in ubiquitous everyday environments. Additionally, recent wearable see-through type displays can be used to present digital information in the user's field-of-view immediately and seamlessly. Combining these state-of-the-art technologies, the proposed user-attended content analysis framework can effectively recognize what the user attends to and present augmentative information in an adequate way. The experiments showed the feasibility of the proposed systems. Furthermore, the user studies showed the benefits of the proposed applications in practical everyday scenarios.

I believe that technologies become more pervasive and easily available in everyday environments in the near future. For next generation of such pervasive CHI, attention-aware systems will play a very important role.

Appendices

Appendix A

Epipolar Geometry

Figure A.1 shows epipolar geometry of two camera view case. A point X in the real space can be seen as a point X' in the camera view C_1. The point X' in C_1 corresponds to a line-of-sight to the target point X from the camera origin (the point-of-view). This line-of-sight from C_1 corresponds to the epipolar line X'' in the camera view C_2. The point X exists on the epipolar line X'' in the camera view C_2.

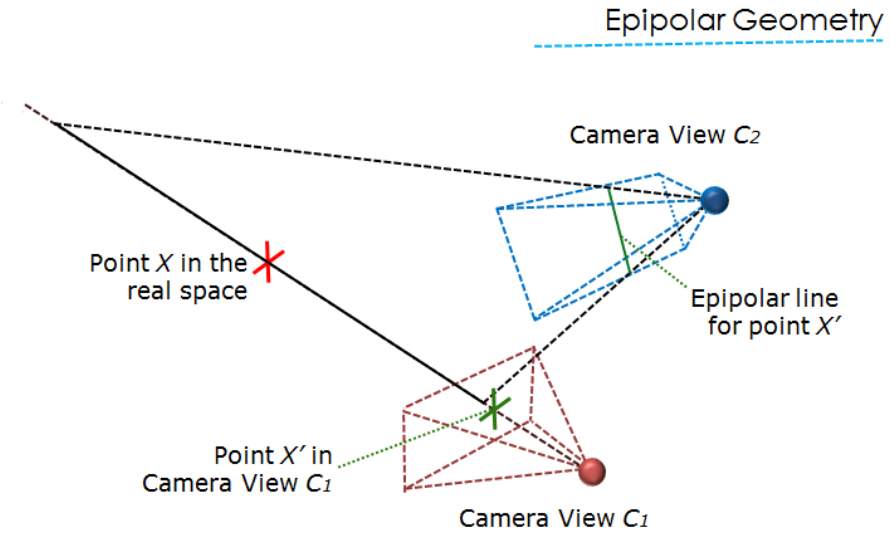


Figure A.1: Epipolar geometry.

Appendix B

Japanese Characters: Katakana and Hiragana

In Figure B.1 and B.2, I show Japanese katakana and hiragana tokens.

ワ	ラ	ヤ	マ	ハ	ナ	タ	サ	カ	ア
	リ		ミ	ヒ	ニ	チ	シ	キ	イ
ヲ	ル	ユ	ム	フ	ヌ	ツ	ス	ク	ウ
	レ		メ	ヘ	ネ	テ	セ	ケ	エ
ン	ロ	ヨ	モ	ホ	ノ	ト	ソ	コ	オ
			パ	バ		ダ	ザ	ガ	
			ピ	ビ		ヂ	ジ	ギ	
			プ	ブ		ヅ	ズ	グ	
			ペ	ベ		デ	ゼ	ゲ	
			ポ	ボ		ド	ゾ	ゴ	

Figure B.1: Japanese katakana table. This table shows 71 different tokens.

わ	ら	や	ま	は	な	た	さ	か	あ
	り		み	ひ	に	ち	し	き	い
を	る	ゆ	む	ふ	ぬ	つ	す	く	う
	れ		め	へ	ね	て	せ	け	え
ん	ろ	よ	も	ほ	の	と	そ	こ	お
			ぱ	ば		だ	ざ	が	
			ぴ	び		ぢ	じ	ぎ	
			ぷ	ぶ		づ	ず	ぐ	
			ぺ	べ		で	ぜ	げ	
			ぽ	ぼ		ど	ぞ	ご	

Figure B.2: Japanese hiragana table. This table shows 71 different tokens.

List of Terms and Abbreviations

AG Attentional gaze. The type of eye gaze process that the viewer attentionally gazes on (visually attends to) something.

AIA Attention-driven image analysis.

AirScouter An optical see-through HMD glass produced by Brother.

ANN Approximate nearest neighbour.

ANOVA Analysis of variance.

AR Augmented reality.

ASR Automatic speech recognition.

attention-aware system The type of computer system that can understand the user attention in the environment.

CHI Computer-human interaction.

CIA Continuous image analysis.

congitive state The state of a person's cognitive processes. e.g., reading text.

DoF Degree of freedom.

DoG Difference of Gaussian.

EOG Electrooculography.

ERMed Erweiterte Realität in der Medizin (German). In English, augmented reality in the medicine. This system can support the doctor's professional work using AR.

ETG Eye tracking glasses.

Eye-con The type of icon in a display that can be activated by eye gaze.

gaze-guided image analysis The image analysis method guided by the user's eye gaze. This method allows a system to limit a scene image to analyze usually by cropping a local image region.

GRL Gaze repetitive leap. One of the proposed gaze gestures.

ground truth The true data label for classification/retrieval/detection test.

GS Gaze scan gesture. One of the proposed gaze gestures.

HMD Head-mounted display.

IMU Inertial measurement unit.

LBP Local binary patterns.

LLAH Locally likely arrangement hashing.

m meter.

mean opinion score The score to provides a numerical indication of the perceived quality of the system which ranges from 1 to 5, where 1 is the lowest quality and 5 is the highest quality measurement.

MR Mixed reality.

msec millisecond.

Museum Guide 2.0 One of the proposed applications in this thesis. The application analyzes the eye movements of the user and presents augmentative information of objects in a museum.

NN Nearest neighbour.

OCR Optical character recognition.

RBF Radial basis function.

SIFT Scale-invariant features transform.

SVM Support vector machine.

SVR Support vector regression.

Talking Places One of the proposed applications in this thesis. The application analyzes the eye movements of the user and presents augmentative information of objects, shops, and signs in a city.

TTS Text-to-speech.

user-attended VCA Analysis on visual content that is attended by the user.

VCA Visual content analysis. The vision-based approach for content analysis in a scene.

Visual Diary One of the proposed applications in this thesis. The application stores a collection of images that attract the user attention in his or her everyday life.

VOR Vestibulo-ocular reflex.

VR Virtual reality.

Own Publications

[Kob+12] Takuya Kobayashi, Takumi Toyama, Faisal Shafait, Masakazu Iwamura, Koichi Kise, and Andreas Dengel. "Recognizing words in scenes with a head-mounted eye-tracker". In: *The Proceedings of the 10th International Workshop on Document Analysis Systems (DAS2012)*. Gold Coast, Queensland, Australia: IEEE, 2012, pp. 333–338.

[Orl+14a] Jason Orlosky, Takumi Toyama, Daniel Sonntag, András Sárkány, and András Lincz. "On-body multi-input indoor localization for dynamic emergency scenarios: fusion of magnetic tracking and optical character recognition with mixed-reality display". In: *Proceedings of the 2014 International Conference on Pervasive Computing and Communications Workshops (PerNem2014)*. Budapest, Hungary: IEEE, 2014, pp. 320–325.

[Orl+14b] Jason Orlosky, Takumi Toyama, Daniel Sonntag, and Kiyoshi Kiyokawa. "Using Eye-Gaze and Visualization to Augment Memory". In: *Proceedings of the 16th International Conference on Human-Computer Interaction* (2014), pp. 282–291.

[Sch+13] Christian Schulz, Daniel Sonntag, Markus Weber, and Takumi Toyama. "Multimodal interaction strategies in a multi-device environment around natural speech". In: *Proceedings of the companion publication of the 2013 international conference on Intelligent user interfaces companion - IUI '13 Companion*. 2013, p. 97.

[Shi+14] Yuki Shiga, Takumi Toyama, Andreas Dengel, Koichi Kise, and Yuzuko Utsumi. "Daily activity recognition combining gaze motion and visual features". In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication - UbiComp '14 Adjunct*. New York, New York, USA: ACM Press, 2014, pp. 1103–1111.

[ST13a] Daniel Sonntag and Takumi Toyama. "Location-Awareness of Vision-Based Augmented Reality Applications". In: *Proceedings of the 3rd Workshop on Location Awareness for Mixed and Dual Reality (LAMDa' 13)*. Santa Monica, CA, USA, 2013.

[ST13b] Daniel Sonntag and Takumi Toyama. "On-body IE : A Head-Mounted Multimodal Augmented Reality System for Learning and Recalling Faces". In: *Proceedings of the 9th International Conference on Intelligent Environments (IE2013)*. Atgens, Greece: IEEE, 2013, pp. 151–156.

[Son+13] Daniel Sonntag, Sonja Zillner, Christian Schulz, Markus Weber, and Takumi Toyama. "Towards medical cyber-physical systems: Multimodal augmented reality for doctors and knowledge discovery about patients". In: *Design, User Experience, and Usability. User Experience in Novel Technological Environments Lecture Notes in Computer Science* 8014 (2013), pp. 401–410.

[Toy11] Takumi Toyama. "Object recognition system guided by gaze of the user with a wearable eye tracker". In: *Proceedings of the 33rd Symposium of the German Association for Pattern Recognition (DAGM2011)* (2011), pp. 444–449.

[Toy+11] Takumi Toyama, Thomas Kieninger, Faisal Shafait, and Andreas Dengel. "Museum guide 2.0 – an eye-tracking based personal assistant for museums and exhibits". In: *Re-Thinking Technology in Museums 2011: Emerging Experiences*. Limerick, Ireland, 2011.

[Toy+12a] Takumi Toyama, Thomas Kieninger, Faisal Shafait, and Andreas Dengel. "Gaze guided object recognition using a head-mounted eye tracker". In: *Proceedings of the Symposium on Eye Tracking Research and Applications - ETRA '12*. 2012, pp. 91–98.

[Toy+12b] Takumi Toyama, Thomas Kieninger, Faisal Shafait, and Andreas Dengel. "User Gaze Detection on Arbitrary Objects Using Physical Sensors and an Eye Tracker in a Real Environment". In: *Proceedings of the 10th Asia Pacific Conference on Computer Human Interaction (APCHI)*. Matsue, Japan, 2012, pp. 421–426.

[Toy+13a] Takumi Toyama, Daniel Sonntag, Markus Weber, and Christian Schulz. "Gaze-based Online Face Learning and Recognition in Augmented Reality". In: *Proceedings of the IUI 2013 Workshop on Interactive Machine Learning*. Santa Monica, CA, USA: ACM, 2013.

[Toy+13b] Takumi Toyama, Wakana Suzuki, Andreas Dengel, and Koichi Kise. "User attention oriented augmented reality on documents with document dependent dynamic overlay". In: *Proceedings of the 2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)* (2013), pp. 299–300.

[Toy+13c] Takumi Toyama, Wakana Suzuki, Andreas Dengel, and Koichi Kise. "Wearable Reading Assist System : Augmented Reality Document Combining Document Retrieval and Eye Tracking". In: *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR)* (2013), pp. 30–34.

[Toy+14a] Takumi Toyama, Daniel Sonntag, Andreas Dengel, Takahiro Matsuda, Masakazu Iwamura, and Koichi Kise. "A mixed reality head-mounted text translation system using eye gaze input". In: *Proceedings of the 19th international conference on Intelligent User Interfaces - IUI '14* (2014), pp. 329–334.

[Toy+14b] Takumi Toyama, Jason Orlosky, Daniel Sonntag, and Kiyoshi Kiyokawa. "A natural interface for multi-focal plane head mounted displays using 3D gaze". In: *Proceedings of 12th International Working Conference on Advanced Visual Interfaces (AVI 2014)*. Como, Italy, 2014, pp. 25–32.

[Toy+15] Takumi Toyama, Jason Orlosky, Daniel Sonntag, and Kiyoshi Kiyokawa. "Attention Engagement and Cognitive State Analysis for Augmented Reality Text Display Functions". In: *Proceedings of the 20th international conference on Intelligent User Interfaces - IUI '15*. 2015.

[V¨+14] Gyula Vörös, Anita Verő, Balázs Pintér, Brigitta Miksztai-Réthey, Takumi Toyama, András Lőincz, and Daniel Sonntag. "Towards a Smart Wearable Tool to Enable People with SSPI to Communicate by Sentence Fragments". In: *Proceedings of the 4th International Symposium on Pervasive Computing Paradigms for Mental Health (MINDCARE2014)* (2014), pp. 1–10.

[Web+13] Markus Weber, Christian H Schulz, Daniel Sonntag, and Takumi Toyama. "Digital Pens as Smart Objects in Multimodal Medical Application Frameworks". In: *Proceedings of the IUI 2013 Workshop on Interacting with Smart Objects* (2013), pp. 5–8.

Bibliography

[Aho+06] Timo. Ahonen, Abdenour. Hadid, and Matti. Pietikainen. "Face Description with Local Binary Patterns: Application to Face Recognition". In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 28 (2006), pp. 2037–2041.

[Aja+11] Antti Ajanki et al. "An augmented reality interface to contextual information". In: *Virtual Reality* 15 (2011), pp. 161–173.

[Ake+04] Kurt Akeley, Simon J Watt, and Martin S Banks. "A Stereo Display Prototype with Multiple Focal Distances". In: *ACM transactions on graphics (TOG)* 1.212 (2004), pp. 804–813.

[Anc+12] Massimo Ancona, Betty Bronzini, Davide Conte, and Gianluca Quercini. "Developing Attention-Aware and Context-Aware User Interfaces on Handheld Devices". In: *Interactive Multimedia*. Ed. by Ioannis Deliyannis. InTech, Mar. 2012.

[Arb+11] Pablo Arbeláez, Michael Maire, Charless Fowlkes, and Jitendra Malik. "Contour detection and hierarchical image segmentation". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33 (2011), pp. 898–916.

[Ast+09] Stylianos Asteriadis, Paraskevi Tzouveli, Kostas Karpouzis, and Stefanos Kollias. "Estimation of behavioral user state based on eye gaze and head pose-application in an e-learning environment". In: *Multimedia Tools and Applications* 41 (2009), pp. 469–493.

[BK06] Brian P. Bailey and Joseph A. Konstan. "On the need for attention-aware systems: Measuring effects of interruption on task performance, error rate, and affective state". In: *Computers in Human Behavior*. Vol. 22. 2006, pp. 685–708.

[Ban+04] Sang Woo Ban, Minho Lee, and Hyun Seung Yang. "A face detection using biologically motivated bottom-up saliency map model and top-down perception model". In: *Neurocomputing* 56 (2004), pp. 475–480.

[BI04] Ling Bao and Stephen S. Intille. *Pervasive Computing*. Ed. by Alois Ferscha and Friedemann Mattern. Vol. 3001. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 1 –17.

[Bar+11] Florin Barbuceanu, Csaba Antonya, Mihai Duguleana, and Zoltan Rusak. "Attentive User Interface for Interaction within Virtual Reality Environments Based on Gaze Analysis". In: *Human-Computer Interaction. Interaction Techniques and Environments*. Ed. by Julie A. Jacko. Vol. 6762. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 204–213.

[Bar+08] Christian Bartolein, Achim Wagner, Meike Jipp, and Essam Badreddin. "Easing Wheelchair Control by Gaze-based Estimation of Intended Motion". In: *Proceedings of the 17th IFAC World Congress, 2008* (2008), pp. 9162–9167.

[Bay+06] Herbert Bay, Tinne Tuytelaars, and Luc Van Gool. "SURF: Speeded up robust features". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 3951 LNCS. 2006, pp. 404–417.

[BB95] S. S. Beauchemin and J. L. Barron. *The computation of optical flow*. 1995.

[Bee+10a] Nikolaus Bee, Johannes Wagner, Elisabeth André, Thurid Vogt, Fred Charles, David Pizzi, and Marc Cavazza. "Gaze Behavior during Interaction with a Virtual Character in Interactive Storytelling". In: *International Workshop on Interacting with ECAs as Virtual Characters*. 2010, p. 6.

[Bee+10b] Nikolaus Bee, Johannes Wagner, Elisabeth André, Fred Charles, David Pizzi, and Marc Cavazza. "Interacting with a gaze-aware virtual character". In: *Workshop on Eye Gaze in Intelligent Human Machine Interaction*. 2010, pp. 71–77.

[Beu+12] Stijn De Beugher, Younes Ichiche, Geert Brône, and Toon Goedemé. "Automatic analysis of eye-tracking data using object detection algorithms". In: *Proceedings of the 2012 ACM Conference on Ubiquitous Computing - UbiComp '12* (2012), p. 677.

[Beu+14] Stijn De Beugher, Geert Brône, and Toon Goedemé. "Automatic analysis of in-the-wild mobile eye-tracking experiments using object, face and person detection". In: *Proceedings of the international conference on computer vision theory and applications (VISIGRAPP 2014)*. VISIGRAPP, 2014, pp. 625–633.

[Bie+09] Ralf Biedert, Georg Buscher, and Andreas Dengel. "The eyeBook Using Eye Tracking to Enhance the Reading Experience". In: *Informatik-Spektrum* 33.3 (Sept. 2009), pp. 272–281.

[Bie+10] Ralf Biedert, Georg Buscher, Sven Schwarz, Jörn Hees, and Andreas Dengel. "Text 2.0". In: *Proceedings of the 28th of the international conference extended abstracts on Human factors in computing systems - CHI EA '10* (2010), p. 4003.

[Bie+12] Ralf Biedert, Jörn Hees, Andreas Dengel, and Georg Buscher. "A Robust Realtime Reading-Skimming Classifier". In: *Proceedings of the Symposium on Eye Tracking Research and Applications ETRA 12* 1 (2012), pp. 123–130.

[Ble64] Woodrow W. Bledsoe. "The model method in facial recognition". In: *Technical Report PRI 15*. Panoramic Research, Inc., 1964.

[Bol82] Richard A. Bolt. *Eyes at the interface*. 1982.

[Bon+09] D. Bonino, E. Castellina, F. Corno, a. Gale, a. Garbo, K. Purdy, and F. Shi. "A blueprint for integrated eye-controlled environments". In: *Universal Access in the Information Society* 8.4 (Mar. 2009), pp. 311–321.

[Bro+06] P. Brooks, K. Y. Phang, R. Bradley, D. Oard, R. White, and F. Guimbretière. "Measuring the Utility of Gaze Detection for Task Modeling: A Preliminary Study". In: *Workshop on Intelligent User Interfaces for Intelligence Analysis (part of IUI 2006)*. 2006.

[BG10] Andreas Bulling and Hans Gellersen. "Toward Mobile Eye-Based Human-Computer Interaction". In: *Technology* (2010), pp. 8–12.

[Bul+08] Andreas Bulling, Jamie A. Ward, Hans Gellersen, and Gerhard Tröster. "Robust Recognition of Reading Activity in Transit Using Wearable Electrooculography". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. 2008, pp. 19–37.

[Bul+09] Andreas Bulling, Daniel Roggen, and Gerhard Tröster. "Wearable EOG goggles: Seamless sensing and context-awareness in everyday environments". In: *Journal of Ambient Intelligence and Smart Environments* 1 (2009), pp. 157–171.

[Bul+11a] Andreas Bulling, Jamie A. Ward, Hans Gellersen, and Gerhard Tröster. "Eye movement analysis for activity recognition using electrooculography." In: *IEEE transactions on pattern analysis and machine intelligence* 33.4 (Apr. 2011), pp. 741–53.

[Bul+11b] Andreas Bulling, Daniel Roggen, and Gerhard Troester. "What's in the Eyes for Context-Awareness?" In: *IEEE Pervasive Computing* 4 (2011), pp. 48–57.

[Bus+08] Georg Buscher, Andreas Dengel, and Ludger van Elst. "Eye movements as implicit relevance feedback". In: *Proceeding of the twenty-sixth annual CHI conference extended abstracts on Human factors in computing systems - CHI '08* (2008), p. 2991.

[Bus+12] Georg Buscher, Andreas Dengel, Ralf Biedert, and Ludger V Elst. "Attentive documents: Eye tracking as implicit feedback for information retrieval and beyond". In: *Information Retrieval* 1 (2012), pp. 1–30.

[Bus35] Thomas G. Buswell. *How people look at pictures: a study of the psychology and perception in art.* 1935, p. 198.

[But+08] Nicholas J. Butko, Lingyun Zhang, Garrison W. Cottrell, and Javier R. Movellan. "Visual saliency model for robot cameras". In: *Proceedings - IEEE International Conference on Robotics and Automation*. 2008, pp. 2398–2403.

[Cal+13] Marco Caligari, Marco Godi, Simone Guglielmetti, Franco Franchignoni, and Antonio Nardone. "Eye tracking communication devices in amyotrophic lateral sclerosis: Impact on disability and quality of life." In: *Amyotrophic lateral sclerosis & frontotemporal degeneration* 14 (2013), pp. 546–52.

[CM01] Christopher S Campbell and Paul P Maglio. "A robust algorithm for reading detection". In: *Proceedings of the 2001 workshop on Percetive user interfaces PUI 01* ACM Intern (2001), pp. 1–7.

[Cas+09] Monica S. Castelhano, Michael L. Mack, and John M. Henderson. "Viewing task influences eye movement control during active scene perception." In: *Journal of vision* 9 (2009), pp. 6.1–15.

[CL11] Chih-Chung Chang and Chih-Jen Lin. "LIBSVM: A Library for Support Vector Machines". In: *ACM Transactions on Intelligent Systems and Technology* 2 (2011), 27:1–27:27.

[Cho+12] Isaac Cho, Wenwen Dou, Zachary Wartell, William Ribarsky, and Xiaoyu Wang. "Evaluating depth perception of volumetric data in semi-immersive VR". In: *Proceedings - IEEE Virtual Reality*. 2012, pp. 95–96.

[Chu+14] Tim Chuk, Alvin C W Ng, Emanuele Coviello, Antoni B Chan, and Janet H Hsiao. "Understanding eye movements in face recognition with hidden Markov model". In: *Journal of vision* 14 (2014), pp. 328–333.

[Cir+12] Dan Ciresan, Ueli Meier, and Juergen Schmidhuber. "Multi-column Deep Neural Networks for Image Classification". In: *2012 IEEE Conference on Computer Vision and Pattern Recognition* (2012), pp. 3642–3649.

[CK14] Moreno I. Coco and Frank Keller. "Classification of visual and linguistic tasks using eye-movement features". In: *Journal of Vision* 14 (2014), pp. 1–18.

[Cou+11] François Courtemanche, Esma Aïmeur, Aude Dufresne, Mehdi Najjar, and Franck Mpondo. "Activity recognition using eye-gaze movements and traditional interactions". In: *Interacting with Computers* 23.3 (May 2011), pp. 202–213.

[CM06] F. L. Coutinho and C. H. Morimoto. "Free head motion eye gaze tracking using a single camera and multiple light sources". In: *2006 19th Brazilian Symposium on Computer Graphics and Image Processing*. 2006, pp. 171–178.

[Csu+04] Gabriella Csurka, Christopher R. Dance, Lixin Fan, Jutta Willamowski, and Cédric Bray. "Visual categorization with bags of keypoints". In: *Proceedings of the ECCV International Workshop on Statistical Learning in Computer Vision*. 2004, pp. 59–74.

[DP04] Elizabeth T. Davis and John Palmer. "Visual search and attention: an overview." In: *Spatial vision* 17 (2004), pp. 249–255.

[Dén+11] O. Déniz, G. Bueno, J. Salido, and F. De La Torre. "Face recognition using Histograms of Oriented Gradients". In: *Pattern Recognition Letters* 32 (2011), pp. 1598–1603.

[Dod08] R. Dodge. *The Psychology and Pedagogy of Reading, with a review of the history of reading and writing, and of methods, texts, and hygiene in reading*. 1908.

[Dor+10] Michael Dorr, Thomas Martinetz, Karl R Gegenfurtner, and Erhardt Barth. "Variability of eye movements when viewing dynamic natural scenes." In: *Journal of vision* 10 (2010), p. 28.

[Dra12] Valentin Dragoi. "Ocular Motor System (Section 3, Chapter 7)". In: *Neuroscience Online: An Electronic Textbook for the Neurosciences* (2012).

[DS07] Heiko Drewes and Albrecht Schmidt. "Interacting with the computer using gaze gestures". In: *Proc. of the Int. Conf. on Human-computer interaction'07*. 2007, pp. 475–488.

[Duc02] Andrew T. Duchowski. "A breadth-first survey of eye-tracking applications." In: *Behavior research methods, instruments, & computers : a journal of the Psychonomic Society, Inc* 34 (2002), pp. 455–470.

[Duc+00] Andrew T. Duchowski, Vinay Shivashankaraiah, Tim Rawls, Anand K. Gramopadhye, Brian J. Melloy, and Barbara Kanki. "Binocular eye tracking in virtual reality for inspection training". In: *Proceedings of the 2000 Symposium on Eye Tracking Research & Applications* (2000), pp. 89–96.

[Eck11] M. P. Eckstein. *Visual search: A retrospective*. 2011.

[Ehr+07] Howard Ehrlichman, Dragana Micic, Amber Sousa, and John Zhu. "Looking for answers: Eye movements in non-visual cognitive tasks". In: *Brain and Cognition* 64 (2007), pp. 7–20.

[Ein+08a] Wolfgang Einhäuser, Merrielle Spain, and Pietro Perona. "Objects predict fixations better than early saliency." In: *Journal of vision* 8 (2008), pp. 18.1–26.

[Ein+08b] Wolfgang Einhäuser, Ueli Rutishauser, and Christof Koch. "Task-demands can immediately reverse the effects of sensory-driven saliency in complex visual stimuli." In: *Journal of vision* 8 (2008), pp. 2.1–19.

[Ero+08] Berna Erol, Emilio Antúnez, and JJ Hull. "HOTPAPER: multimedia interaction with paper using mobile phones". In: *Proceedings of the 16th ACM international conference on Multimedia, MM' 08* (2008), pp. 399–408.

[Eva+12] Karen M. Evans, Robert a Jacobs, John A. Tarduno, and Jeff B. Pelz. "Collecting and Analyzing Eye-tracking Data in Outdoor Environments". In: *Journal of Eye Movement Research* 5 (2012), pp. 1–19.

[Fat+12] Alireza Fathi, Yin Li, and James M. Rehg. "Learning to recognize daily actions using gaze". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 7572 LNCS. 2012, pp. 314–327.

[Fin97] John M. Findlay. "Saccade target selection during visual search". In: *Vision Research* 37 (1997), pp. 617–631.

[FU08] Tom Foulsham and Geoffrey Underwood. "What can saliency models predict about eye movements? Spatial and sequential aspects of fixations during encoding and recognition." In: *Journal of vision* 8 (2008), pp. 6.1–17.

[Gaa66] Kenneth Gaarder. "Fine eye movements during inattention". In: *Nature* (1966), pp. 83–84.

[Gid+13] Kerstin Gidlöf, Annika Wallin, Richard Dewhurst, and Kenneth Holmqvist. "Using eye tracking to trace a cognitive process: Gaze behaviour during decision making in a natural environment". In: *Journal of Eye Movement Research* 6.1 (2013), pp. 1–14.

[Gop73] Daniel Gopher. *Eye-movement patterns in selective listening tasks of focused attention*. 1973.

[Gre+12] Michelle R. Greene, Tommy Liu, and Jeremy M. Wolfe. "Reconsidering Yarbus: A failure to predict observers' task from eye movement patterns". In: *Vision Research* 62 (2012), pp. 1–8.

[HB09] Adrian Haffegee and Russell Barrow. "Eye tracking and gaze based interaction within immersive virtual environments". In: *ICCS 2009 Proceedings of the 9th International Conference on Computational Science* (2009), pp. 729–736.

[Han+12] Seongwon Han, Sungwon Yang, Jihyoung Kim, and Mario Gerla. "EyeGuardian: a framework of eye tracking and blink detection for mobile device users". In: *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications - HotMobile '12*. New York, New York, USA: ACM Press, 2012.

[HJ10] Dan Witzner Hansen and Qiang Ji. "In the eye of the beholder: a survey of models for eyes and gaze." In: *IEEE transactions on pattern analysis and machine intelligence* 32.3 (2010), pp. 478–500.

[Han+04] J. P. Hansen, K. Tø rning, A. S. Johansen, K. Itoh, and H. Aoki. "Gaze typing compared with input by head and hand". In: *Proc. of the Eye tracking research and applications symposium on Eye tracking research and applications - ETRA 2004* 1 (2004), pp. 131–138.

[Har+73] Robert M. Haralick, K. Shanmugam, and Its'Hak Dinstein. "Textural Features for Image Classification". In: *IEEE Transactions on Systems, Man, and Cybernetics* 3 (1973).

[HR12] H Heikkilä and KJ Räihä. "Simple gaze gestures and the closure of the eyes as an interaction technique". In: *Proceedings of the Symposium on Eye Tracking Research and Applications - ETRA '12* 1 (2012), pp. 147–154.

[HE08] MR Heinen and PM Engel. "Visual selective attention model for robot vision". In: *2008 IEEE Latin American Robotic Symposium* (Oct. 2008), pp. 29–34.

[Hen03] John M. Henderson. "Human gaze control during real-world scene perception". In: *Trends in Cognitive Sciences* 7.11 (Nov. 2003), pp. 498–504.

[Hen+03] John M. Henderson, Carrick C. Williams, Monica S. Castelhano, and Richard J. Falk. "Eye movements and picture processing during recognition." In: *Perception & psychophysics* 65 (2003), pp. 725–734.

[Hen+05] John M. Henderson, Carrick C. Williams, and Richard J. Falk. "Eye movements are functional during face learning." In: *Memory & cognition* 33 (2005), pp. 98–106.

[Hen+13] John M. Henderson, Svetlana V. Shinkareva, Jing Wang, Steven G. Luke, and Jenn Olejarczyk. "Predicting Cognitive State from Eye Movements". In: *PLoS ONE* 8 (2013).

[HB11] Niels Henze and Susanne Boll. "Who's that girl? Handheld augmented reality for printed photo books". In: *Proceedings of the 13th IFIP TC 13 international conference on Human-computer interaction - Volume Part III, INTERACT11*. Vol. 6948 LNCS. 2011, pp. 134–151.

[HMC12] Irwandi Hipiny and Walterio Mayol-Cuevas. "Recognising Egocentric Activities from Gaze Regions with Multiple-Voting Bag of Words". In: *CSTR-12-003* (2012), pp. 1–15.

[Hof79] James E. Hoffman. "A two-stage model of visual search." In: *Perception & psychophysics* 25.4 (Apr. 1979), pp. 319–27.

[Hor11] Tim Horberry. "Safe design of mobile equipment traffic management systems". In: *International Journal of Industrial Ergonomics* 41.5 (2011), pp. 551–560.

[Hor+06] Junichi Hori, Koji Sakano, and Yoshiaki Saitoh. "Development of a communication support device controlled by eye movements and voluntary eye blink". In: *IEICE Transactions on Information and Systems* E89-D (2006), pp. 1790–1797.

[IM98] Piotr Indyk and Rajeev Motwani. "Approximate nearest neighbors: towards removing the curse of dimensionality". In: *Proceedings of the thirtieth annual ACM symposium on Theory of computing* 126 (1998), pp. 604–613.

[IB04] Shamsi T. Iqbal and Brian P. Bailey. "Using Eye Gaze Patterns to Identify User Tasks". In: *The Grace Hopper Celebration of Women in Computing*. 2004, p. 6.

[IJ11] Yoshio Ishiguro and Rekimoto Jun. "Peripheral Vision Annotation : Noninterference Information Presentation Method for Mobile Augmented Reality". In: *AH 11 Proceedings of the 2nd Augmented Human International Conference*. 2011, pp. 1–4.

[Ish+10] Yoshio Ishiguro, Adiyan Mujibiya, Takashi Miyaki, and Jun Rekimoto. "Aided eyes: eye activity sensing for daily life". In: *The 1st Augmented Human International Conference (AH2010)*. 2010, pp. 1–7.

[Itt+98] Laurent Itti, Christof Koch, and Ernst Niebur. "A model of saliency-based visual attention for rapid scene analysis". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (1998), pp. 1254–1259.

[Iwa+10] Masakazu Iwamura, Tomohiko Tsuji, and Koichi Kise. "Memory-based recognition of camera-captured characters". In: *Proceedings of the 8th IAPR International Workshop on Document Analysis Systems - DAS '10* (2010), pp. 89–96.

[Iwa+11] Masakazu Iwamura, Takuya Kobayashi, and Koichi Kise. "Recognition of multiple characters in a scene image using arrangement of local features". In: *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*. 2011, pp. 1409–1413.

[Iwa+13] Masakazu Iwamura, Tomokazu Sato, and Koichi Kise. "What is the Most EfficientWay to Select Nearest Neighbor Candidates for Fast Approximate Nearest Neighbor Search?" In: *Computer Vision (ICCV), 2013 IEEE International Conference on*. 2013, pp. 3535–3542.

[Iwa+14] Masakazu Iwamura, Kai Kunze, Yuya Kato, Yuzuko Utsumi, and Koichi Kise. "Haven't we met before? A Realistic Memory Assistance System to Remind You of The Person in Front of You". In: *Proceedings of the 5th Augmented Human International Conference (AH2014)* (2014).

[Jab13] Ferris Jabr. "The Reading Brain in the Digital Age: The Science of Paper versus Screens". In: *Scientific American* (2013).

[Jac90] Robert J. K. Jacob. "What you look at is what you get: eye movement-based interaction techniques". In: *Proceedings of the SIGCHI conference on Human factors in computing systems Empowering people - CHI '90*. 1990, pp. 11–18.

[JK03] Robert J. K. Jacob and Keith S. Karn. "Eye tracking in human-computer interaction and usability research: Ready to deliver the promises". In: *The Mind Eye: Cognitive and Applied Aspects of Eye Movement Research*. Vol. 2. 2003, pp. 573–605.

[JH07] Jannick Rolland and Hong Hua. "Head-Mounted Display Systems". In: *Encyclopedia of Optical Engineering* (2007), pp. 1–14.

[Jim+08] Jorge Jimenez, Diego Gutierrez, and Pedro Latorre. "Gaze-based Interaction for Virtual Environments". In: *Journal Of Universal Computer Science* 14 (2008), pp. 3085–3098.

[JC76] Marcel Adam Just and Patricia A Carpenter. *Eye fixations and cognitive processes*. 1976.

[Kea+03] Foo Tun Keat Foo Tun Keat, S. Ranganath, and Y.V. Venkatesh. "Eye gaze based reading detection". In: *TENCON 2003. Conference on Convergent Technologies for Asia-Pacific Region* 2 (2003).

[Ken67] A. Kendon. "Some functions of gaze-direction in social interaction." In: *Acta psychologica* 26 (1967), pp. 22–63.

[Ki+07] Jeongseok Ki, Yong M. Kwon, and Kwanghoon Sohn. "3D gaze tracking and analysis for attentive human computer interaction". In: *Proceedings of the Frontiers in the Convergence of Bioscience and Information Technologies, FBIT 2007*. 2007, pp. 617–621.

[Kim+10] Kiyoung Kim, Vincent Lepetit, and Woontack Woo. "Scalable real-time planar targets tracking for digilog books". In: *The Visual Computer* 26.6-8 (Apr. 2010), pp. 1145–1154.

[Kim+08] Sung-Kyu Kim, Dong-Wook Kim, Yong Moo Kwon, and Jung-Young Son. "Evaluation of the monocular depth cue in 3D displays." In: *Optics express* 16 (2008), pp. 21415–21422.

[Kim+11] Sung-Kyu Kim, Eun-Hee Kim, and Dong-Wook Kim. "Full parallax multifocus three-dimensional display using a slanted light source array". In: *Optical Engineering* 50.11 (Nov. 2011).

[Kim+13] Takashi Kimura, Rong Huang, Seiichi Uchida, Masakazu Iwamura, Shinichiro Omachi, and Koichi Kise. "The Reading-Life Log–Technologies to Recognize Texts That We Read". In: *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR)* (2013), pp. 91–95.

[KT06] Holle Kirchner and Simon J. Thorpe. "Ultra-rapid object detection with saccadic eye movements: Visual processing speed revisited". In: *Vision Research* 46 (2006), pp. 1762–1776.

[Kis+10] Koichi Kise, Megumi Chikano, Kazumasa Iwata, Masakazu Iwamura, Seiichi Uchida, and Shinichiro Omachi. "Expansion of queries and databases for improving the retrieval accuracy of document portions: an application to a camera-pen system". In: *Proceedings of the 9th IAPR International Workshop on Document Analysis Systems (DAS '10)*. 2010, pp. 309–316.

[Kla+08] Arto Klami, Craig Saunders, Teófilo E. de Campos, and Samuel Kaski. "Can Relevance of Images Be Inferred from Eye Movements ? Categories and Subject Descriptors". In: *Proceedings of the 1st ACM international conference on Multimedia information retrieval MIR '08*. 2008, pp. 134–140.

[Kob+13] Takuya Kobayashi, Masakazu Iwamura, Takahiro Matsuda, and Koichi Kise. "An anytime algorithm for camera-based character recognition". In: *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR 2013* (2013), pp. 1140–1144.

[Koh+08] Stefan Kohlbecher, Stanislavs Bardins, Klaus Bartl, Erich Schneider, Tony Poitschke, and Markus Ablassmeier. "Calibration-free eye tracking by reconstruction of the pupil ellipse in 3D space". In: *ETRA '08 Proceedings of the 2008 symposium on Eye tracking research & applications* 1 (2008), pp. 135–138.

[Kun+13a] Kai Kunze, Hitoshi Kawaichi, Kazuyo Yoshimura, and Koichi Kise. "The Wordometer – Estimating the Number of Words Read Using Document Image Retrieval and Mobile Eye Tracking". In: *Proceedings of the 12th International Conference on Document Analysis and Recognition (ICDAR)* (2013).

[Kun+13b] Kai Kunze, Hitoshi Kawaichi, Kazuyo Yoshimura, and Koichi Kise. "Towards inferring language expertise using eye tracking". In: *CHI '13 Extended Abstracts on Human Factors in Computing Systems on - CHI EA '13* (2013), p. 217.

[Kwa+10] Jennifer R. Kwapisz, Gary M. Weiss, and Samuel A. Moore. "Activity Recognition using Cell Phone Accelerometers". In: *Human Factors* (2010).

[Kwo+06] Yong-moo Kwon, Kyeong-won Jeon, Jeongseok Ki, Qonita M Shahab, Sangwoo Jo, and Sung-kyu Kim. "3D Gaze Estimation and Interaction to Stereo Display". In: 5.3 (2006), pp. 41–45.

[LH01] Michael F. Land and Mary Hayhoe. "In what ways do eye movements contribute to everyday activities?" In: *Vision Research*. Vol. 41. 2001, pp. 3559–3565.

[Lee+11] Jae-Young Lee, Hyung-Min Park, Seok-Han Lee, Soon-Ho Shin, Tae-Eun Kim, and Jong-Soo Choi. "Design and implementation of an augmented reality system using gaze interaction". In: *Multimedia Tools and Applications* (Dec. 2011).

[LD08] Matthew L. Lee and Anind K. Dey. "Lifelogging memory appliance for people with episodic memory impairment". In: *Proceedings of the 10th international conference on Ubiquitous computing - UbiComp '08* 344 (2008), p. 44.

[LC12] Young-Seol Lee and Sung-Bae Cho. "Recognizing multi-modal sensor signals using evolutionary learning of dynamic Bayesian networks". In: *Pattern Analysis and Applications* (Sept. 2012).

[Lev66] Vladimir I. Levenshtein. "Binary codes capable of correcting deletions, insertions, and reversals". In: *Soviet Physics Doklady* 10.8 (1966), pp. 707–710.

[Li+13] Cheng-Yuan Li, Yen-Chang Chen, Wei-Ju Chen, Polly Huang, and Hao-hua Chu. "Sensor-embedded teeth for oral activity recognition". In: *Proceedings of the 17th annual international symposium on International symposium on wearable computers - ISWC '13* (2013), p. 41.

[LA08] Jing Li and Nigel M. Allinson. "A comprehensive review of current local features for computer vision". In: *Neurocomputing* 71 (2008), pp. 1771–1787.

[Li+05] Xiaowei Li, Yue Liu, Yongtian Wang, and Dayuan Yan. "Computing homography with RANSAC algorithm: a novel method of registration". In: *Proc. SPIE 5637, Electronic Imaging and Multimedia Technology IV* 5637 (Feb. 2005). Ed. by Chung-Sheng Li and Minerva M. Yeung, pp. 109–112.

[LH10] Sheng Liu and Hong Hua. "A systematic method for designing depth-fused multi-focal plane three-dimensional displays." In: *Optics express* 18 (2010), pp. 11562–11573.

[Liu+10] Sheng Liu, Hong Hua, and Dewen Cheng. "A novel prototype for an optical see-through head-mounted display with addressable focus cues". In: *IEEE Transactions on Visualization and Computer Graphics* 16 (2010), pp. 381–393.

[Los+14] Viktor Losing, Thies Pfeiffer, Lukas Rottkamp, and Michael Zeunert. "Guiding visual search tasks using gaze-contingent auditory feedback". In: *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing Adjunct Publication - UbiComp '14 Adjunct*. New York, New York, USA: ACM Press, 2014, pp. 1093–1102.

[Lot+07] F. Lotte, M. Congedo, A. Lécuyer, F. Lamarche, and B. Arnaldi. "A review of classification algorithms for EEG-based brain-computer interfaces." In: *Journal of neural engineering* 4 (2007), R1–R13.

[Low99] David G. Lowe. "Object recognition from local scale-invariant features". In: *Proceedings of the Seventh IEEE International Conference on Computer Vision* 2 (1999).

[Low04] David G. Lowe. "Distinctive image features from scale-invariant keypoints". In: *International Journal of Computer Vision* 60 (2004), pp. 91–110.

[LK81] Bruce D. Lucas and Takeo Kanade. "An iterative image registration technique with an application to stereo vision". In: *Proceedings of the 7th international joint conference on Artificial intelligence (IJCAI'81)* Volume 2 (1981), pp. 674–679.

[Mae+10] Takuya Maekawa, Yutaka Yanagisawa, Yasue Kishino, Katsuhiko Ishiguro, Koji Kamei, Yasushi Sakurai, and Takeshi Okadome. "Object-based activity recognition with heterogeneous sensors on wrist". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 6030 LNCS. 2010, pp. 246–264.

[MF13] Andrew Maimone and Henry Fuchs. "Computational augmented reality eyeglasses". In: *2013 IEEE International Symposium on Mixed and Augmented Reality, ISMAR 2013*. 2013, pp. 29–38.

[MR02] Päivi Majaranta and Kari-Jouko Räihä. "Twenty Years of Eye Typing: Systems and Design Issues". In: *Proceedings of the 2002 symposium on Eye Tracking Research & Applications (ETRA)*. 2002, pp. 15–22.

[Mak+09] Jarmo Makkonen, Ivan Avdouevski, Riitta Kerminen, and Ari Visa. "Context Awareness in Human-Computer Interaction". In: *Human-Computer Interaction* (Dec. 2009).

[Man+12] Radosw Mantiuk, Micha Kowalik, Adam Nowosielski, and Bartosz Bazyluk. "Do-it-yourself eye tracker: Low-cost pupil-based eye tracker for computer graphics applications". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 7131 LNCS. 2012, pp. 115–125.

[MGA14] Pascual Martínez-Gómez and Akiko Aizawa. "Recognition of understanding level and language skill using measurements of reading behavior". In: *Proceedings of the 19th international conference on Intelligent User Interfaces - IUI '14*. New York, New York, USA: ACM Press, 2014, pp. 95–104.

[Mas+12] Tomohiro Mashita, Kiyoshi Kiyokawa, and Haruo Takemura. "Subjective evaluations on perceptual depth of stereo image and effective field of view of a wide-view head mounted projective display with a semi-transparent retroreflective screen". In: *2012 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, Nov. 2012, pp. 327–328.

[Mat+01] Yoshio Matsumoto, Tomoyuki Ino, and Tsukasa Ogasawara. "Development of intelligent wheelchair system with face and gaze based interface". In: *Proceedings - IEEE International Workshop on Robot and Human Interactive Communication*. 2001, pp. 262–267.

[Maz+80] A. Mazur, E. Rosa, M. Faupel, J. Heller, R. Leen, and B. Thurman. "Physiological aspects of communication via mutual gaze." In: *AJS; American journal of sociology* 86 (1980), pp. 50–74.

[MG+12] Carlos Merino-Gracia, Karel Lenc, and Majid Mirmehdi. "A head-mounted device for recognizing text in natural scenes". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 7139 LNCS. 2012, pp. 29–41.

[Mog+14] Mohammad Moghimi, Pablo Azagra, Luis Montesano, Ana C. Murillo, and Belongie Serge. "Experiments on an RGB-D Wearable Vision System for Egocentric Activity Recognition". In: *CVPR Workshop on Egocentric (First-person) Vision*. 2014.

[MY09] Jean-Michel Morel and Guoshen Yu. *ASIFT: A New Framework for Fully Affine Invariant Image Comparison*. 2009.

[MM05] Carlos H. Morimoto and Marcio R.M. Mimica. "Eye gaze tracking techniques for interactive applications". In: *Computer Vision and Image Understanding* 98 (2005), pp. 4–24.

[MR81] R. E. Morrison and K. Rayner. "Saccade size in reading depends upon character spaces and not visual angle." In: *Perception & psychophysics* 30 (1981), pp. 395–396.

[MN95] Hiroshi Murase and Shree K. Nayar. "Visual learning and recognition of 3-d objects from appearance". In: *International Journal of Computer Vision* 14 (1995), pp. 5–24.

[MN13] George Wamamu Musumba and Henry O. Nyongesa. "Context awareness in mobile computing: A review". In: *International Journal of Machine Learning and Applications* 2.1 (May 2013), pp. 1–10.

[Nag+09] Takashi Nagamatsu, Junzo Kamahara, and Naoki Tanaka. "Calibration-free gaze tracking using a binocular 3D eye model". In: *Proceedings of the 27th international conference extended abstracts on Human factors in computing systems - CHI EA '09* (2009), p. 3613.

[NG05] Jiri Najemnik and Wilson S Geisler. "Optimal eye movement strategies in visual search." In: *Nature* 434 (2005), pp. 387–391.

[Nak+06] Tomohiro Nakai, Koichi Kise, and Masakazu Iwamura. "Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 3872 LNCS. 2006, pp. 541–552.

[Nei64] Ulric Neisser. *Visual Search*. 1964.

[Nib+93] Wayne Niblack, Ron Barber, William Equitz, Myron Flickner, Eduardo H. Glasman, Dragutin Petkovic, Peter Yanker, Christos Faloutsos, and Gabriel Taubin. "QBIC project: querying images by content, using color, texture, and shape". In: *Storage and Retrieval for Image and Video Databases (SPIE)* 1908 (1993), pp. 173–187.

[Nil+09] Susanna Nilsson, Torbjörn Gustafsson, and Per Carleberg. "Hands Free Interaction with Virtual Information in a Real Environment : Eye Gaze as an Interaction Tool in an Augmented Reality System". In: *Psychology Journal* 7.2 (2009), pp. 175–196.

[NN06] Ko Nishino and Shree K. Nayar. "Corneal imaging system: Environment from eyes". In: *International Journal of Computer Vision* 70 (2006), pp. 23–40.

[NS92] C. L. Novak and S. A. Shafer. "Anatomy of a color histogram". In: *Proceedings 1992 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1992).

[Nov+96] D. G. Novick, B. Hansen, and K. Ward. "Coordinating turn-taking with gaze". In: *Proceeding of Fourth International Conference on Spoken Language Processing. ICSLP '96* 3 (1996).

[NH10] Antje Nuthmann and John M. Henderson. "Object-based attentional selection in scene viewing." In: *Journal of vision* 10.8 (Jan. 2010), p. 20.

[O'H+08] Kieron O'Hara, Mischa M. Tuffield, and Nigel Shadbolt. "Lifelogging : Privacy and Empowerment with Memories for Life". In: *Identity in the Information Society* 1.2 (2008), pp. 155–172.

[Oga+12] Keisuke Ogaki, Kris M. Kitani, Yusuke Sugano, and Yoichi Sato. "Coupling eye-motion and ego-motion features for first-person activity recognition". In: *2012 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops* (June 2012), pp. 1–7.

[Orl+13a] Jason Orlosky, Kiyoshi Kiyokawa, and Haruo Takemura. "Dynamic text management for see-through wearable and heads-up display systems". In: *Proceedings of the 2013 International Conference on Intelligent User Interfaces* (2013), pp. 363–370.

[Orl+13b] Jason Orlosky, Kiyoshi Kiyokawa, and Haruo Takemura. "Management and manipulation of text in dynamic mixed reality workspaces". In: *2013 IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*. IEEE, Oct. 2013, pp. 1–4.

[Pal+13] Lucas Paletta, Katrin Santner, and Gerald Fritz. "An Integrated System for 3D Gaze Recovery and Semantic Analysis of Human Attention". In: *arXiv preprint arXiv:1307.7848* (2013).

[Par78] R. E. Parker. "Picture processing during recognition." In: *Journal of experimental psychology. Human perception and performance* 4 (1978), pp. 284–293.

[PE12] Matthew F. Peterson and Miguel P. Eckstein. "Looking just below the eyes is optimal across face recognition tasks." In: *Proceedings of the National Academy of Sciences of the United States of America* 109 (2012), E3314–23.

[PH10] Panagiotis C. Petrantonakis and Leontios J. Hadjileontiadis. "Emotion recognition from EEG using higher order crossings". In: *IEEE Transactions on Information Technology in Biomedicine* 14 (2010), pp. 186–197.

[Pfe08] Thies Pfeiffer. "Towards Gaze Interaction in Immersive Virtual Reality : Evaluation of a Monocular Eye Tracking Set-Up". In: *Virtuelle und Erweiterte RealitatFunfter Workshop der GIFachgruppe VRAR*. 2008, pp. 81–92.

[Pfe+08] Thies Pfeiffer, Marc E. Latoschik, Ipke Wachsmuth, and J. Herder. "Evaluation of binocular eye trackers and algorithms for 3d gaze interaction in virtual reality environments". In: *Journal of Virtual Reality and Broadcasting* 5 (2008), p. 16.

[Pin+09] Nicolas Pinto, James J. DiCarlo, and David D. Cox. "How far can you get with a modern face recognition test set using only simple features?" In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPR Workshops 2009*. 2009, pp. 2591–2598.

[Pon+06] Jean Ponce, Martial Hebert, Cordelia Schmid, and Andrew Zisserman. *Toward Category-Level Object Recognition*. Vol. 4170. 2006.

[PV98] M. Pontil and A. Verri. "Support vector machines for 3D object recognition". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20 (1998).

[Pos80] Michael I. Posner. *Orienting of attention*. 1980.

[Pre+09] Helmut Prendinger, Aulikki Hyrskykari, Minoru Nakayama, Howell Istance, Nikolaus Bee, and Yosiyuki Takahasi. "Attentive interfaces for users with disabilities: eye gaze for intention and uncertainty estimation". In: *Universal Access in the Information Society* 8.4 (Mar. 2009), pp. 339–354.

[Pro] Open Source Project. *OpenCV (Open Source Computer Vision)*.

[QC12] Liu Qiong and Liao Chunyuan. "PaperUI". In: *Proceedings of the 4th international conference on Camera-Based Document Analysis and Recognition*. Ed. by Masakazu Iwamura and Faisal Shafait. Vol. 7139. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012.

[QZ05] Pernilla Qvarfordt and Shumin Zhai. "Conversing with the user based on eye-gaze patterns". In: *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '05* (2005), p. 221.

[Ras+10] B. Rasolzadeh, M. Bjorkman, K. Huebner, and D. Kragic. "An Active Vision System for Detecting, Fixating and Manipulating Objects in the Real World". In: *The International Journal of Robotics Research* 29 (2010), pp. 133–154.

[Ray78] Keith Rayner. "Eye movements in reading and information processing." In: *Psychological bulletin* 85 (1978), pp. 618–660.

[Ray95] Keith Rayner. "Eye movements and cognitive processes in reading, visual search, and scene perception". In: *Studies in Visual Information Processing* 6 (1995), pp. 3–22.

[Ray98] Keith Rayner. "Eye movements in reading and information processing: 20 years of research." In: *Psychological bulletin* 124.3 (Nov. 1998), pp. 372–422.

[Ray09] Keith Rayner. *Eye movements and attention in reading, scene perception, and visual search.* 2009.

[Ray+09] Keith Rayner, Tim J. Smith, George L. Malcolm, and John M. Henderson. "Eye movements and visual encoding during scene perception". In: *Psychological Science* 20 (2009), pp. 6–10.

[RG89] W R. Rice and S. D. Gaines. "One-way analysis of variance with unequal variances." In: *Proceedings of the National Academy of Sciences of the United States of America* 86.21 (1989), pp. 8183–8184.

[RT06] Claudia Roda and Julie Thomas. "Attention aware systems: Theories, applications, and research agenda". In: *Computers in Human Behavior*. Vol. 22. 2006, pp. 557–587.

[RW08] Peter M. Roth and Martin Winter. "Survey of Appearance-Based Methods for Object Recognition". In: *Technical Report ICG-TR-01/08* (2008).

[Roz+11] David Rozado, Francisco B. Rodriguez, and Pablo Varona. "Gaze gesture recognition with hierarchical temporal memory networks". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 6691 LNCS. 2011, pp. 1–8.

[Sal+04] Jarkko Saloj, Kai Puolam, and Samuel Kaski. "Relevance feedback from eye movements for proactive information retrieval". In: *Proceedings of the Workshop on Processing Sensory Information for Proactive Systems (PSIPS 2004)*. 2004, pp. 37–42.

[Sal+03] J Salojärvi, I Kojo, J Simola, and S Kaski. "Can relevance be inferred from eye movements in information retrieval". In: *Proceedings of the 4th Workshop on Self-Organizing Maps (WSOM 2003)*. 2003, pp. 261–266.

[Sch+99] Albrecht Schmidt, Michael Beigl, and Hans W. Gellersen. "There is more to context than location". In: *Computers and Graphics (Pergamon)* 23 (1999), pp. 893–901.

[SS04] Brian T. Schowengerdt and Eric J. Seibel. "True three-dimensional displays that allow viewers to dynamically shift accommodation, bringing objects displayed at different viewing distances into and out of focus." In: *Cyberpsychology & behavior : the impact of the Internet, multimedia and virtual reality on behavior and society* 7 (2004), pp. 610–620.

[Sch+09] Alexander C. Schütz, Doris I. Braun, and Karl R. Gegenfurtner. "Object recognition during foveating eye movements." In: *Vision research* 49.18 (Sept. 2009), pp. 2241–53.

[Sch+11] Alexander C. Schütz, Doris I. Braun, and Karl R. Gegenfurtner. "Eye movements and perception: A selective review." In: *Journal of vision* 11 (2011).

[Sha+12] Asif Shahab, Faisal Shafait, Andreas Dengel, and Seiichi Uchida. "How salient is scene text?" In: *Proceedings of the 10th IAPR International Workshop on Document Analysis Systems (DAS)* (Mar. 2012), pp. 317–321.

[Shi+07] Fangmin Shi, Alastair Gale, and Kevin Purdy. *A New Gaze-Based Interface for Environmental Control*. 2007.

[ST94] Jianbo Shi Jianbo Shi and C. Tomasi. "Good features to track". In: *Computer Vision and Pattern Recognition, 1994. Proceedings CVPR '94., 1994 IEEE Computer Society Conference on* (1994), pp. 593–600.

[SX05] Xi Shi and Yangsheng Xu. "A wearable translation robot". In: *Proceedings - IEEE International Conference on Robotics and Automation*. Vol. 2005. 2005, pp. 4400–4405.

[Spe+00] Charles Spence, Jane Ranson, and Jon Driver. "Cross-modal selective attention: on the difficulty of ignoring sounds at the locus of visual attention." In: *Perception & psychophysics* 62 (2000), pp. 410–424.

[SB90] India Starker and Richard A. Bolt. "A gaze-responsive self-disclosing display". In: *Proceedings of the SIGCHI conference on Human factors in computing systems*. 1990, pp. 3–10.

[Sul04] Alan Sullivan. "DepthCube solid-state 3D volumetric display". In: *Virtual Reality* 5291 (2004), pp. 279–284.

[Sut68] Ivan E. Sutherland. "A head-mounted three dimensional display". In: *Proceedings of the December 9-11, 1968, fall joint computer conference, part I on - AFIPS '68 (Fall, part I)*. 1968, p. 757.

[Tak+11] Kazutaka Takeda, Koichi Kise, and Masakazu Iwamura. "Real-time document image retrieval for a 10 million pages database with a memory efficient and stability improved LLAH". In: *Proceedings of the International Conference on Document Analysis and Recognition, ICDAR*. 2011, pp. 1054–1058.

[Tam+78] Hideyuki Tamura, Shunji Mori, and Takashi Yamawaki. "Textural Features Corresponding to Visual Perception". In: *IEEE Transactions on Systems, Man, and Cybernetics* 8 (1978).

[TJ00] Vildan Tanriverdi and Robert J. K. Jacob. "Interacting with eye movements in virtual environments". In: *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '00* (2000), pp. 265–272.

[Tre01] S. Treue. "Neural correlates of attention in primate visual cortex." In: *Trends in neurosciences* 24 (2001), pp. 295–300.

[TN00] M. Tuceryan and N. Navab. "Single point active alignment method (SPAAM) for optical see-through HMD calibration for AR". In: *Proceedings IEEE and ACM International Symposium on Augmented Reality (ISAR 2000)* (2000).

[TP91] M. A. Turk and A. P. Pentland. "Face recognition using eigenfaces". In: *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1991).

[TM07] Tinne Tuytelaars and Krystian Mikolajczyk. "Local Invariant Feature Detectors: A Survey". In: *Foundations and Trends in Computer Graphics and Vision* 3.3 (2007), pp. 177–280.

[Ura+05] K. Uratani, T. Machida, K. Kiyokawa, and H. Takemura. "A study of depth visualization techniques for virtual annotations in augmented reality". In: *IEEE Proceedings. VR 2005. Virtual Reality, 2005.* (2005).

[Ure+11] Hakan Urey, Kishore V. Chellappan, Erdem Erden, and Phil Surman. "State of the art in stereoscopic and autostereoscopic displays". In: *Proceedings of the IEEE*. Vol. 99. 2011, pp. 540–555.

[Vat+05] Radu Daniel Vatavu, efan-gheorghe Pentiuc, and Christophe Chaillou. "On Natural Gestures for Interacting with Virtual Environments". In: *Advances in Electrical and Computer Engineering* 24 (2005).

[Ver03] Roel Vertegaal. "Attentive User Interfaces". In: *Communications of the ACM* 46 (2003), pp. 30–33.

[Vid+14] Mélodie Vidal, David H. Nguyen, and Kent Lyons. "Looking At or Through? Using Eye Tracking to Infer Attention Location for Wearable Transparent Displays". In: *Proceedings of the 2014 ACM International Symposium on Wearable Computers*. 2014, pp. 87–90.

[VJ04] P. Viola and M. Jones. "Robust Real-Time Face Detection". In: *International Journal of Computer Vision* 57 (2004), pp. 137–154.

[Wat+98] Y. Watanabe, Y. Okada, Yeun-Bae Kim Yeun-Bae Kim, and T. Takeda. "Translation camera". In: *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170)* 1 (1998).

[Web+00] M. Weber, M. Welling, and P. Perona. "Unsupervised Learning of Models for Recognition". In: *Computer Vision - ECCV 2000*. Vol. 1842. 2000, pp. 18–32.

[WH12] Sunu Wibirama and Kazuhiko Hamamoto. "A Geometric Model for Measuring Depth Perception in Immersive Virtual Environment". In: *Proceedings of the 10th Asia Pacific Conference on Computer Human Interaction (APCHI)* 1 (2012), pp. 325–330.

[Wob+08] Jacob O. Wobbrock, James Rubinstein, Michael W. Sawyer, and Andrew T. Duchowski. "Longitudinal evaluation of discrete consecutive gaze gestures for text entry". In: *Proceedings of the 2008 symposium on Eye tracking research & applications - ETRA '08* (2008), p. 11.

[Woo+12] RL Woods, Ivonne Fetchenheuer, Fernando Vargas-Martin, and Eli Peli. "The impact of non-immersive head-mounted displays (HMDs) on the visual field". In: *Journal of the Sociey for Information Display* 11 (2012), pp. 191–198.

[Yan+12] J. Yang, Y. Tian, L. Y. Duan, T. Huang, and W. Gao. "Group-sensitive multiple kernel learning for object recognition". In: *IEEE Transactions on Image Processing* 21 (2012), pp. 2838–2852.

[Yar67] Alfred Yarbus. "Eye movements and vision". In: *Plenum Press* (1967).

[Yos13] Atsuo Yoshitaka. "Image/Video Indexing, Retrieval and Summarization Based on Eye Movement". In: *Proceesings of the 4th International Conference on Computing and Informatics, ICOCI 2013*. 2013, pp. 15–21.

[Zel+05] Robert C. Zeleznik, Andrew S. Forsberg, and Jurgen P. Schulze. "Look-That-There: Exploiting Gaze in Virtual Reality Interactions". In: *Technical Report CS-05-04* March (2005).

[ZW11] Changshui Zhang and Fei Wang. "Graph-based semi-supervised learning". In: *Frontiers of Electrical and Electronic Engineering in China* 6 (2011), pp. 17–26.

[Zha+08] Hui Zhang, Jason E. Fritts, and Sally A. Goldman. "Image segmentation evaluation: A survey of unsupervised methods". In: *Computer Vision and Image Understanding* 110 (2008), pp. 260–280.

[Zha+06] J. Zhang, M. Marszak, S. Lazebnik, and C. Schmid. "Local Features and Kernels for Classification of Texture and Object Categories: A Comprehensive Study". In: *International Journal of Computer Vision* 73.2 (Sept. 2006), pp. 213–238.

[Zha+03] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. "Face recognition: A literature survey". In: *Acm Computing Surveys* 35 (2003), pp. 399–458.

[Zho+08] Feng Zhou, Henry Been Lirn Dun, and Mark Billinghurst. "Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR". In: *Proceedings - 7th IEEE International Symposium on Mixed and Augmented Reality 2008, ISMAR 2008*. 2008, pp. 193–202.

Curriculum Vitae

Takumi Toyama

Education

Mar.2004	Graduation, Miyazaki-Ohmiya High School, Miyazaki, Japan
Mar.2009	Bachelor of Engineering, Department of Computer and Systems Sciences, Graduate School of Engneering, Osaka Prefecture University, Japan
Mar.2011	Master of Engineering, Department of Computer Science and Intelligent Systems, Graduate School of Engneering, Osaka Prefecture University, Japan

Work Experiences

Jun.2010 - Feb.2011	Internship as an exchange student at DFKI GmbH
Jun.2011 - Sep.2014	Junior Researcher at DFKI GmbH
Oct.2014 - Jul.2015	Researcher at DFKI GmbH
Aug.2015 - present	Application Developer at SMI GmbH