FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Visual Attention in Computer Graphics

DISSERTATION

zur Erlangung des akademischen Grades

Doktor der technischen Wissenschaften

eingereicht von

Matthias Bernhard
Matrikelnummer 0427314

an der
Fakultät für Informatik der Technischen Universität Wien

Betreuung: Associate Prof. Dr. techn. Dipl.-Ing. Dipl.-Ing. Michael Wimmer

Diese Dissertation haben begutachtet:

(Associate Prof. Dr. techn. Dipl.-Ing. Dipl.-Ing. Michael Wimmer)	(Prof. Dr.-habil. inż Karol Myszkowski)

Wien, 15.09.2014

(Matthias Bernhard)

FAKULTÄT
FÜR !NFORMATIK
Faculty of Informatics

Visual Attention in Computer Graphics

DISSERTATION

submitted in partial fulfillment of the requirements for the degree of

Doktor der technischen Wissenschaften

by

Matthias Bernhard
Registration Number 0427314

to the Faculty of Informatics
at the Vienna University of Technology

Advisor: Associate Prof. Dr. techn. Dipl.-Ing. Dipl.-Ing. Michael Wimmer

The dissertation has been reviewed by:

(Associate Prof. Dr. techn. Dipl.-Ing. Dipl.-Ing. Michael Wimmer)	(Prof. Dr.-habil. inż Karol Myszkowski)

Wien, 15.09.2014

(Matthias Bernhard)

Erklärung zur Verfassung der Arbeit

Matthias Bernhard
Kröllgasse 18, 1150 Wien

Hiermit erkläre ich, dass ich diese Arbeit selbständig verfasst habe, dass ich die verwendeten Quellen und Hilfsmittel vollständig angegeben habe und dass ich die Stellen der Arbeit - einschließlich Tabellen, Karten und Abbildungen -, die anderen Werken oder dem Internet im Wortlaut oder dem Sinn nach entnommen sind, auf jeden Fall unter Angabe der Quelle als Entlehnung kenntlich gemacht habe.

(Ort, Datum) (Unterschrift Verfasser)

Acknowledgements

First and foremost, I would like to thank my supervisor Michael Wimmer for his consistently very constructive suggestions guiding me through the course of this thesis. I am deeply grateful that I had the opportunity to learn so many extremely useful skills from him. Next, I would like to express my deep gratitude to my close collaborator Efstathios Stavrakis. This thesis would not have been possible without his unlimited help and indispensable moral support.

Great thanks go to my appreciated colleagues Michael Hecher, Ralf Habel, Martin Knecht, Daniel Scherzer, Manuela Waldner and Oliver Mattausch, for the great collaboration, the inspiring discussions, and the endless number of interesting and useful things I could learn from them and extend my knowledge so profoundly. Also, I greatly appreciate the great assistance of my Master student Camillo Dell'mour, who provided with his accurate working style a significant contribution to one chapter of this thesis. Further, I would like to thank my former students Karl Grosse, Le Zhang and Kartik Asooja, for the great cooperation and assistance, and the CG-Club for providing the PentaG game engine, which we used in one of our eye-tracking experiments. Besides, special thanks go to my international colleagues with whom I had the pleasure to cooperate in the Crossmod project, particularly Veronika Sundstedt, Erik Reinhard, Oliver Warusfel, Khoa-Van Nguyen, Michael Schwarz and George Drettakis.

My special thanks are extended to all my colleagues of the rendering and visualization group, including the masterminds Meister Eduard Gröller, Michael Wimmer, Ivan Viola and Werner Purgathofer, for their lovely craziness making our institute to one of the most enjoyable environments. With a smile in my heart and a tang of nostalgia I will fondly recall the unique sense of humor, the inspiring mood, the lightening irony and the incredible amusement I so appreciate to have experienced in this stimulating, creative, open-minded and extremely funny culture of our institute.

I owe sincere and earnest thankfulness to the unlimited support provided by all our secretaries and technicians. Especially, I wish to thank Anita Mayerhofer, who is probably the fastest and most reliable secretary in the world. I would also like to express my very great appreciation to our enormously well organized, unbureaucratic and exceedingly helpful "I-find-always-time-to-help-you-instantly"-technicians Stephan Plepelits and Andreas Weiner.

Many thanks I owe to my beloved parents for their support, especially in the most difficult moments. Very personal thanks go to my great brothers Daniel with Carolin and Dominik with Maria Theresia, and to all my friends for the great time I could spend with them and nicely distract myself from this work.

Abstract

This thesis is concerned with gaze analysis methods to study visual attention in interactive 3D computer-graphics applications, such as virtual environments or computer games. Under this scope, research has been carried out in two directions: On the one hand, it was investigated how gaze analysis in three-dimensional virtual environments can be advanced. On the other hand, approaches were explored which improve three-dimensional graphics by taking into account visual attention of a user.

To advance gaze analysis in 3D computer graphics applications, two challenges have been addressed: First, inferring the object of attention at a certain point in time from the current output of an eye tracker – a technique which we denote as gaze-to-object mapping –, and second, deriving a statistical model for visual attention - a data structure we denote as importance map - from sequences of gaze samples recorded from many users. While addressing these challenges is a crucial step towards advancing gaze analysis and research on visual attention which employs modern computer graphics, the results may also be used in applications which attempt to perceptually optimize rendering. Thus, the third challenge addressed in this thesis was to explore an example application for attention-aware rendering techniques, where gaze-to-object mapping or importance maps can be employed to determine or predict the object of attention at run time. Thus, this thesis concludes with a pilot study on an application that dynamically adjusts the configuration of a stereo 3D display such that the object being attended by the user can be seen most comfortably.

Kurzfassung

Diese Arbeit beschäftigt sich mit Eyetracking-basierten Methoden zur Messung, Analyse und Vorhersage visueller Aufmerksamkeit in Echtzeitgrafikanwendungen, wie Computerspiele oder Virtual Environments. An dieser Schnittstelle zwischen Computergrafik und Psychologie wurde in zwei Richtungen geforscht: zum einen wurde untersucht, wie man Computergrafik zusammen mit Eyetracking als methodisches Werzeug zur experimentellen Analyse der Aufmerksamkeit einsetzen kann. Zum anderen wurde versucht, 3D Graphik mithilfe von aufmerksamkeitsgesteuerten Methoden perzeptuell zu verbessern.

Um die Blickpunktanalyse in Grafikanwendungen weiterzuentwickeln, stellten sich in dieser Arbeit folgende zwei Herausforderungen: erstens, die direkte Bestimmung der Objekte auf die ein Benutzer seine Aufmerksamkeit richtet – eine Technik die wir "Gaze-to-Object Mapping" nennen –, und zweitens, die Herleitung eines statistischen Models für die visuelle Aufmerksamkeit - eine Datenstruktur die wir "Importance Map" nennen - aus den Blickpunktdatensequenzen mehrer Benutzer. Während die Bewältigung dieser Herausforderungen ein fundamentaler Schritt zur Weiterentwicklung von Methoden zur Blickpunktanalyse in 3D-Computergrafikanwendungen ist, sollen die Ergebnisse dieser Arbeit auch zur perzeptuellen Optimierung in Computergrafikanwendungen zum Einsatz kommen. Daher war die dritte Herausforderung für diese Arbeit die Erkundung einer Beispielanwendung für ein aufmerksamkeitsgesteuertes Bildsyntheseverfahren. Im dritten Forschungskapitel dieser Arbeit findet sich eine Pilotstudie zu einer Applikation in Stereo 3D Displays. Diese Anwendung steuert die Konfiguration des Displays dynamisch, sodass das Objekt auf welches ein Benutzer seine Aufmerksamkeit richtet mit hohem visuellem Komfort wahrgenommen werden kann.

Contents

CHAPTER 1

Introduction

In case of doubt, make it sound convincing

Murphy's Laws

Computer graphics has developed quickly in the recent decades. Besides the great advances in graphics hardware, the application of basic and new knowledge about perception could be one important factor of the continuation of this progress, particularly towards the ambitious goal of providing a close-to-natural experience with virtual reality simulators.

By taking the functionality and limitations of the human visual system into account, algorithms and hardware can be effectively optimized to render and display computer-generated images of high realism in real time. For instance, when rendering complex virtual environments, there is often a trade-off between computation speed, or electric energy consumption, and display quality. Perceptual metrics that estimate visual sensitivity or attention can be utilized to allocate computational resources more appropriately in computationally expensive calculations.

While there are several kinds of perceptual models that can be used to improve realism and computational efficiency of computer graphics, the focus of this thesis is on *visual attention*, which is an important aspect of visual perception as it determines what we are seeing in the scene we are looking at. With computational attention predictors or eye tracking, an application can be informed about the focus of a user's attention in order to choose, for instance, an appropriate level of detail for scene geometry, or adjust image resolution to the visual accuracy of the human retina. Apart from rendering, attention-aware approaches can be also used, for example, to specify the minimum required accuracy in physical simulations, or to enable the animation of huge crowds of characters. Besides saving computational workload and electric energy, attention-aware techniques could be one of the keys to make simulations of effects that are inherent results of the physiology of the visual system, such as lens accomodation or binocular vision, applicable without disturbing a user's experience due to a perceptually misaligned configuration of the simulation of these effects. This applies, for instance, to depth-of-field simulations, where it should be avoided that a user focuses into the blurred image regions. Another

important application are stereo 3D displays. An ideal stereo display should be responsive to a user's attention in order to minimize visual discomfort, which results from perceptual conflicts introduced by the simulation of stereoscopic viewing on graphics displays. Other applications where visual attention tracking is useful include attention-aware design techniques to visually facilitate (or deliberately impede) interaction in virtual reality simulators and serious gaming environments.

Visual attention is one of the most important properties of visual perception. Attention is the basic control mechanism determining which features of a (virtual) environment we are actually aware of. Visual attention and all other sorts of attention arise from the nature of every perceptual system. Perception can process only a fraction of the enormously rich multimodal stimulation of a vivid socially, intellectually and perceptually interactive environment a sensing person lives in. Attention is the functionality which selects *what* in the information-overloaded perceptual stream is worth to be perceived and cognized. There is no doubt that attention is an obvious phenomenon and we are all aware of it. We experience attention anytime we need to concentrate on a task, to search for a lost item, to follow the story of a movie or when we focus on safely driving a car.

One of the most important behaviors which are related to visual attention are eye movements. Due to the physiology of the retina, there is a tight coupling between gaze behavior and visual attention. The eyes sense visual information in high acuity only in a small region, the fovea, spanning an angle of only 1-2°. This corresponds to the size of a thumbnail held in the distance of an outstretched arm. Outside the fovea, visual acuity drops tremendously with eccentricity. With this perceptual "limitation", the retina embodies a visual filter. Since only a small fraction of the stimulus, e.g., a single object, can be perceived in full detail at a time, the visual system has to collect visual information in a serial fashion, i.e., to move gaze from one interesting location in the stimulus to the next. To optimize this process, visual attention guides gaze to the most meaningful locations, features or objects. Thereby, the field of view, which comprises a tremendous amount of information, is decomposed into a sequence of perceptual samples. These samples help the visual system to update, i.e., to falsify or verify, the predictions of a speculative mental model of the structure, content and meaning in the external scene. Attention-guided control of eye movements is a crucial functionality of this sampling process which constitutes vision. Hence, gaze can be considered as the external manifestation of visual attention. With *eye tracking* we have a technology to sense where gaze is directed to and thus a powerful tool to observe visual attention.

However, an eye-tracker outputs *where* a user is looking at, while visual attention is mainly directed to the content, like an object in the scene. Hence, it would be useful to have a method which allows us to infer *what* a user is attending from the spatial information provided by eye tracking. This imposes the challenge of processing and analyzing gaze data in a way that we can correlate attention to properties of a visual stimulus, like for instance the semantics of objects. In this work, we propose to study visual attention with virtual-environment applications, which are tools gaining increasing acceptance for experimental psychology. We propose that by means of using interactive three-dimensional virtual environments in combination with new methods to gather and analyze gaze data, we can better infer what a user is attending. Using virtual environments as stimulus in attention research has two great advantages: (a) it allows logging

the states of the application in order to fully *reconstruct* the stimulus at the time we analyze the experimental data, and (b), virtual scenes are usually encoded in data structures which cluster scene geometry data in object-based representations. Having access to the internal representation of virtual environments allows us to segment objects in the stimulus. Having a representation of the stimulus which tells us which pixel belongs to which object, we can map gaze to objects. The objects can then be further abstracted by their semantics in order to correlate gaze with meaning.

One of the goals of this thesis is to establish a pipeline for studying visual attention in three-dimensional virtual environments in order to derive *empirical* models for visual attention from gaze recordings which also can account for semantics. The desired result of this process is an importance map which quantifies the likelihood that a user pays attention to a certain semantic category. While the importance map can be used for analysis purposes, it also provides us a function we may use to *predict* visual attention.

Besides advancing tools to study visual attention with virtual environments, this thesis also explores *attention-aware* algorithms for rendering computer graphics images. The idea behind this concept is to improve a display by interactively adapting the rendering configuration to the attention of a user. Knowing a user's focus of attention allows, for instance, to control 3D stereo rendering such that a user experiences less visual discomfort.

These goals impose several challenges, which are described in following section.

1.1 Overview and the Bigger Picture

The research presented in this thesis is structured according to three major challenges, which are illustrated in the overview shown in Figure 1.1.

The first challenge, which is addressed in **Chapter III**, is to *infer* visual attention from gaze data. The output of an eye-tracker provides us only the spatial coordinates of the center of both eyes' receptive field. But since attention is mostly directed to visual information rather than spatial coordinates, the challenge is to determine *what* a user is attending to, given the output of an eye-tracker, which provides only information about *where* gaze is directed to. We propose to perform attention inference in object-space and denote this approach as *gaze-to-object mapping*. The desired result is a probability density function $P(\mathbf{O}|\mathbf{g})$, which predicts for each object an "attention probability" as a posterior of the gaze position $\mathbf{g}$, which is observed with the eye-tracker. The box in the top-left corner of Figure 1.1 shows an example where this probability is visualized as an intensity.

Attention inference from gaze can be useful in real-time applications. For instance, an attention-aware display, as proposed in Chapter 5, can utilize this information to optimize rendering selectively for attended objects.

Besides the use in real-time applications, gaze-to-object mapping is an important component which is required for processing gaze data recorded with virtual environment applications. In the research presented in **Chapter IV** (see bottom-left box in Figure 1.1), we addressed the challenge of analyzing gaze data recorded from many users navigating a large virtual environment. So we take one step further from the inference of attention in a single frame to the derivation of attention models using data of many frames and users. We propose that attention models should be defined by abstracting objects by their properties, such as semantics. Hence, the goal is to

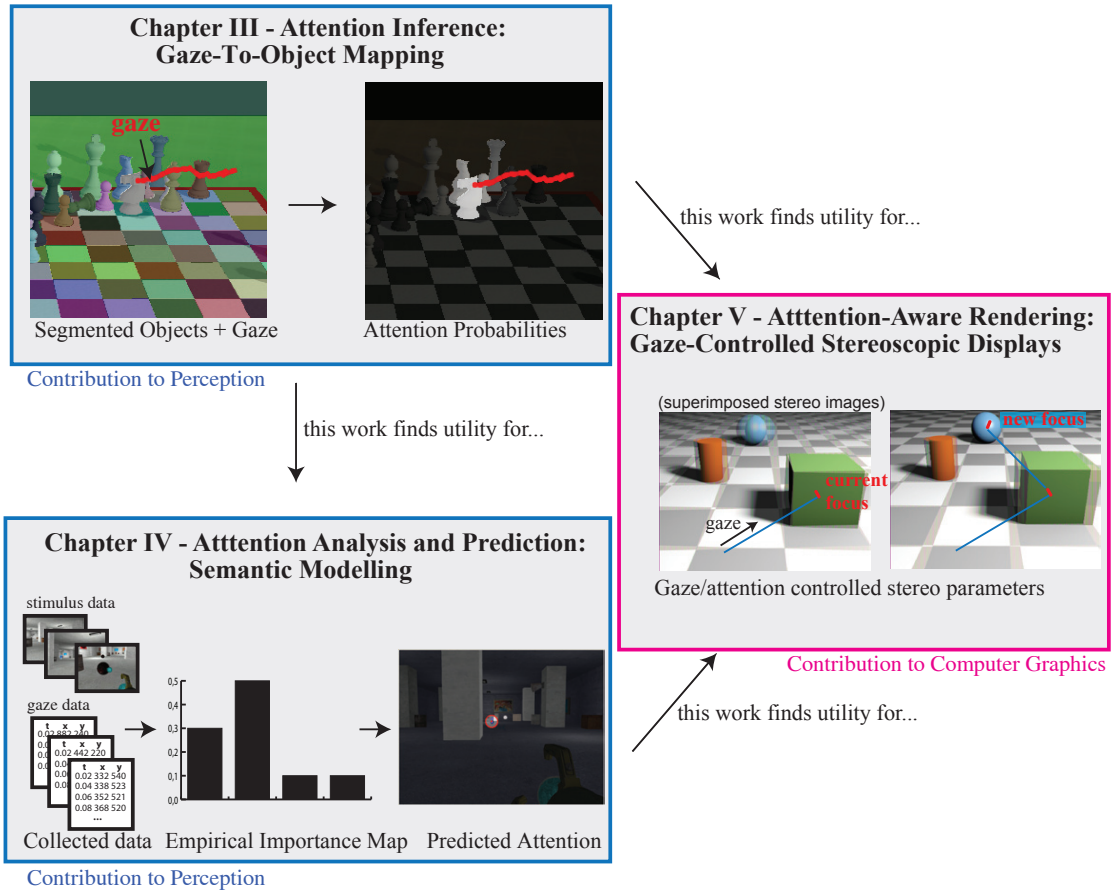


Figure 1.1: Overview

derive an importance map which is defined for object properties rather than individual objects. Ideally, such an importance map can be used to predict attention by estimating the probability $P(\mathbf{O}|\mathbf{x})$ that object $\mathbf{O}$ is attended, given the set of properties $\mathbf{x}$ we use to abstract $\mathbf{O}$. To approach these goals, we devised a technical pipeline that allows to generate empirical importance maps. With the importance maps generated with this pipeline we then attempted to predict visual attention (without eye-tracking).

While Chapters III and IV address the challenges of measuring, analyzing and predicting visual attention with gaze data, we explore in **Chapter V** an example application for attention-aware rendering. In particular, we investigate whether an application which is aware of the objects attended by the user (i.e., we have either $P(\mathbf{O}|\mathbf{g})$ or $P(\mathbf{O}|\mathbf{x})$) can use this knowledge to improve rendering of 3D stereo images. As illustrated in the right box in Figure 1.1, the basic idea is to use an eye tracker (or gaze predictor) to determine a user's focus and use this information to select an optimal configuration of the stereo display. Users of 3D displays often report discomfort caused by perceptual conflicts arising under artificial stereoscopic viewing conditions. In our work, we investigated how discomfort can be reduced by selecting a stereo

rendering configuration which minimizes conflicts in the depth range of the object currently attended.

1.2 Challenges and Research Questions

Challenge I: Inferring Attention from Gaze (Chapter III)

In this work we contribute an important step to facilitate eye-tracking with highly dynamic three dimensional virtual environments. A straight-forward way to analyze gaze recordings obtained from dynamic stimuli is to superimpose visualizations of gaze information (e.g., heatmaps) on synchronized video captures. With this approach, conclusions can be drawn from rather anecdotic evidence. By means of subjective inspection an experimenter can identify behavior patterns that occur under particular circumstances. To establish more objective methodology, it is necessary to perform gaze analysis in a more systemic way by means of building statistics. For instance, counting how often a particular category of objects was attended. To this end, commercial analysis tools offer facilities which allow to specify regions of interest (ROI). However, the ROI have to be specified manually. While this strategy is sufficient for static stimuli, like web pages, dynamic stimuli require tedious editing efforts to place ROI on objects which may change their position in every frame.

With the work presented in Chapters 3 and 4, this thesis proposes a pipeline which accesses the internal structure of virtual scene content rendered to the display and helps to *automate* this process. An important question which arises when we want to infer attention from gaze, is what the units of attention are. Attention can be directed to locations, features and objects (see Chapter 2 Section 2.3). This thesis makes the reasonable assumption that attention is usually *object based*. While the existence of object-based attention is well-supported by a huge body of experimental evidence, an object-based approach is also practically well suited when attention is studied in interactive virtual environments or computer games. These applications usually represent the visual content with a data structure commonly referred to as *scene graph*. A scene graph provides an object-based ontology that allows segmenting objects in the stimulus straightforwardly without object recognition algorithms.

Assuming the availability of an object-based representation of the stimulus, the first important research question which is addressed by this work is:

- Which are the best methods to infer the object of attention from a gaze location ?

A straight-forward way to infer the object of attention is to determine the object underneath the current gaze location. However, the concrete challenge is that an eye tracker does not give an exact 2D position of the user gaze. Instead, because of eye-tracker inaccuracies and small eye movements, the result is only a set of 2D points within a “region of uncertainty”. Moreover, interactive 3D environments are highly dynamic, and scene objects which are arranged in three dimensions are projected onto a 2D screen where they vary considerably in size. For instance, large objects have a higher likelihood to spatially coincide with gaze compared to small objects, though a user does not necessarily attend more often to large objects. Another problem is object density, which is usually avoided in eye-tracking experiments or gaze-controlled games.

To find the most accurate technique to infer object-based attention from spatial gaze data, we need a methodology to evaluate the performance of a gaze-to-object mapping method. Thus, we first require an evaluation data set where the ground truth object of attention is known. So the second research question we had to address was:

- How can we obtain a ground truth data set to evaluate gaze-to-object mapping methods ?

Obtaining the ground truth attention target is not trivial, as eye tracking is usually used to determines this. Therefore, the challenge was to design an experiment which yields a sequence of gaze samples, which is the input of the gaze-to-object mapping technique, together with a ground truth attention target, which is the desired output of an optimal method.

Challenge II: Attention Analysis and Prediction (Chapter IV)

Having solved the problem of finding a good method to infer attended objects from gaze in a single frame, the next step is to develop strategies to study attention by using gaze data of many frames and users. The challenge is to infer models for visual attention empirically through statistical analysis of gaze data. Data collected while eye-tracking a user of a virtual environment is a temporal sequence of gaze points from which we can infer the object of attention in each frame. In every frame, the object of attention is the response, i.e., the *dependent variable*, on a complex stimulus, which is the *independent variable* of the eye-tracking experiment. A representation of the data in terms of independent and dependent variables provides us the formalism which allows to infer a model, i.e., a function which predicts the dependent variable from the independent variables. However, one problem is to derive such a representation of the stimulus. Potentially, a user's attention is affected by all visual properties of the stimulus, as well as the task carried out by the user. We address this problem by defining *abstractions* of object and user properties. This challenge refers to the research question:

- How can we abstract the visual content in a virtual environment application for gaze analysis ?

Thanks to using synthetic stimuli rendered from content which is represented as a scene graph, where geometries are clustered into scene entities, we can encode the visual information in object space. This is a representation which fits well with the way a user forms a mental model of an external environment. Despite the fact that this representation provides a discrete encoding of the visual information as a set of unique objects, we are faced with the curse of dimensionality. A large environment contains many objects and gaze is distributed rather sparsely over individual object instances. Thus, we propose to group individual objects by their semantics. However, grouping objects by their semantics requires knowledge. We address this problem by a system design which allows a user of the analysis tool (i.e., the experimenter) to specify how scene objects are abstracted to infer gaze statistics. To this end, it was necessary to design a suitable interface to support this interaction.

Assuming we have represented the stimulus with a set of carefully chosen semantic properties, we can use this representation to build gaze statistics by counting how often a certain property was attended by the user. However, when a user freely navigates a three-dimensional

environment, the field of view may change in every frame and not object is equally often visible. Thus, another challenge addressed in this work was:

- How to deal with changing visibility while collecting gaze statistics ?

To address this challenge, we needed to specify a normalization strategy which is suitable to compensate for varying visibility of different objects.

Challenge III : Attention-Aware Rendering (Chapter V)

Another challenge of this thesis was to explore the utility of attention-aware approaches in computer graphics. To this end, we selected 3D stereo rendering as example application. In particular, we investigated whether an eye-tracker can be used to improve viewing comfort for stereoscopic display. Nowadays, stereoscopy gains rapid foothold in interactive applications like computer games, virtual reality installations, customer presentations, simulation, planning etc. With the wide availability of 3D-capable TVs and Autostereo Displays, stereoscopy is becoming a commodity. However, current stereoscopic technology is tiring for users, and often reported as uncomfortable after even short times of use. The reason for this is the so-called *vergence/accomodation* conflict, i.e., a mismatch between the actual focusing point of the eyes (the display) and the virtual focusing point in the scene. The goal was to investigate attention-aware approaches to reduce this conflict. Equipping a stereo display with an eye-tracking device, we can infer the object of attention in real time and use this information to dynamically adjust the configuration of the stereo display in a way that discomfort is minimized. We performed a pilot study to answer the question:

- Can a gaze-responsive dynamic stereo configuration increase viewing comfort significantly compared to static settings ?

To answer this question required us to devise an evaluation methodology. The challenge was to design an experiment to measure the stereo viewing comfort experienced by a user in order to evaluate the benefit of gaze-controlled stereoscopic applications in comparison to a conventional solution without attention control. User experience with stereo displays is usually studied by asking users in a questionnaire to rate the subjectively perceived quality of experience on a Likert scale (e.g., by asking: Was the experience very comfortable, comfortable, ..., or painful ?). However, subjective rating scores are rather vague, biased towards the desired result of the experimenter and provide less insights about when, where and why stereo-viewing discomfort occurs during application time. Thus, we attempted in this work to address the challenge:

- How can visual discomfort in stereoscopic applications be evaluated objectively ?

With the joint use of psychophysics and eye tracking, this work performs initial steps towards a methodology to objectively measure discomfort during the use of an stereoscopic application. While psychophysics is used as measurement tool, we propose to use eye-tracking in order to control *where* in a three-dimensional scene and *when* comfort measurements are taken.

Addressing these three major challenges, this thesis made several contributions to scientific literature with important research steps that advance the state-of-the-art in attention analysis and potential applications for attention-aware approaches.

1.3 Scientific Contributions

The following contributions to scientific literature were made with the publications presented in Chapters III – V of this thesis:

Attention Inference – Gaze-to-Object Mapping (Chapter III)

In this work, we devised an experimental methodology to assess methods inferring object-based attention from gaze data recorded in highly dynamic 3D virtual environments.

This work was accepted to the ACM Symposium on Applied Perception (SAP 2014), where it has been selected to appear in the SAP special issue of ACM Transactions on Applied Perception and to be presented as a poster at SIGGRAPH 2014:

- Matthias Bernhard, Efstathios Stavrakis, Michael Hecher and Michael Wimmer. **Gaze-to-Object Mapping During Visual Search in 3D Virtual Environments**. In *ACM Transactions on Applied Perception*, Symposium on Applied Perception Special Issue, August 2014

Collaboration Statement: In this work, I designed, prepared and conducted the eye-tracking study. Moreover, I implemented major components of the software infrastructure we had to establish to analyze and visualize the results. This work was supervised by Michael Wimmer. Efstathios Stavrakis was an important contributor to this work. Besides his assistance in software development, he created the accompanying video and was intensively involved in paper writing. Michael Hecher assisted with feedback on formal issues and helped with the illustrations in the paper. Further acknowledgments we owe to my former intern Kartik Asooja, who helped during an early stage of the experimental design with content selection for the virtual environment scenes.

Attention Analysis and Prediction – Empirical Modeling (Chapter IV)

This work proposes a novel pipeline for gaze analysis which we developed with the vision to derive empirical attention models from gaze data which can be used to predict visual attention.

This chapter is a revised and merged version of the following two publications:

- Matthias Bernhard, Efstathios Stavrakis and Michael Wimmer. **An Empirical Pipeline to Derive Gaze Prediction Heuristics for 3D Action Games**. In *ACM Transactions on Applied Perception*, 8(1), October 2010

- Veronica Sundstedt, Matthias Bernhard, Efstathios Stavrakis, Erik Reinhard and Michael Wimmer. **Visual Attention and Gaze Behaviour in Games: An Object-Based Approach**. In *Game Analytics: Maximizing the Value of Player Data*, pages 543 – 583, April 2013

Collaboration Statement: In the work published in the TAP article, I was responsible for the design, preparation and execution of the eye-tracking study. Moreover, I designed and implemented crucial parts of the pipeline, the algorithms and gaze visualizations. This work was supervised by Michael Wimmer, who provided a strong contribution in paper writing. Efstathios Stavrakis implemented the GUI of our analysis software (Lyzer) and assisted with paper writing and scientific feedback.

Some passages and illustrations were taken from this book chapter. In particular, I merged sections which I have originally written for the book chapter into the introductory parts of the thesis chapter. Partially, the reused text was proof-read and polished by the other authors of the book chapter.

Attention-Aware Rendering – A Pilot-Study on Gaze-Controlled Stereoscopic Applications (Chapter V)

This chapter presents an experiment to perceptually evaluate gaze-controlled stereoscopic applications. This work was published in:

- Matthias Bernhard, Camillo Dell'mour, Michael Hecher, Efstathios Stavrakis and Michael Wimmer. **The Effects of Fast Disparity Adjustments in Gaze-Controlled Stereoscopic Applications.**. In *Proceedings of the Symposium on Eye Tracking Research and Applications* , March 2014.

Collaboration Statement: The work is also part of Camillo Dell'mour's Master Thesis which I have supervised. I initiated this research, designed the experiment and took most responsibilities in the analysis of the data, the interpretation of the results and paper writing. The experimental test application was programmed and configured by Camillo Dell'mour, who also conducted the study and provided us the data in the form we requested for analysis. Michael Wimmer supervised this work. Michael Hecher assisted in the experiment design, data analysis and editing of the result plots. Efstathios Stavrakis was a strong contributor in paper writing.

1.4 Potential Applications

The research presented in this thesis has a variety of potential applications in different fields. So this chapter is concluded with some speculative examples how this work could be used:

Gaze-Controlled User Interaction

While eye-tracking devices have been expensive lab equipment in the past, this technology is on the verge of becoming an affordable consumer hardware, and future desktop computers, game consoles and mobile devices may use this technique as standard accessory. This work investigates how attention to scene objects can be inferred from gaze data. While this is an important component for gaze analysis and attention-aware real-time rendering, it may also find

utility for applications, like gaze-controlled computer games, which allow users selecting and manipulating objects in a three-dimensional virtual environment via gaze.

Game Analytics

For an optimal game design, the designers need a maximum understanding of the player's attention. This knowledge can be applied at the design stage of computer-game development, or at run time when the application is used. During game design, knowledge about what a user is most often attending to can be, for instance, utilized to identify scenarios where a user's attention should be better guided. For instance, appropriate game design (e.g., salient colors) could be used to guide attention in order to facilitate gaming tasks and make the game more enjoyable. Furthermore, knowing how user attention is distributed over different scene objects would allow to selectively allocate more efforts (e.g., working time of content artists) into the creation of frequently attended objects or animations. At application run-time, attention models or predictors can be potentially employed to inform the game's artificial intelligence to behave responsively to the attention of a user.

Psychophysical Instruments

Apart from using attention analysis for game analytics, the methods proposed in this thesis may assist psychologists in behavioral studies carried out with interactive virtual environments or virtual-reality setups. The great benefits of virtual-environment applications for perceptual and cognitive experiments can be summarized with four basic advantages [56]: First, stimuli generated with virtual environments are *flexible* because they are programmable. Second, thanks to modern 3D rendering technology, they can convey the realism and complexity to approach a high degree of *ecological validity*. Third, physically correct simulation and interactive behavior give a rich and plausible *sensorial feedback* to the user. And fourth, due to their virtual nature, they allow an accurate *performance recording* by tracking exactly all states (including user responses) of an interactive application. This allows tracking all variables in complex scenarios, which is not possible with pen-pencil based observational methods.

Attention-Aware Rendering

With attention-aware approaches, one can optimize computationally intensive simulations required to achieve high realism in high-end virtual-reality applications. We can selectively maximize fidelity of computationally expensive simulations, such as physically based lighting, complex and detailed animations (e.g., huge crowds of characters), or physically correct interaction of materials (e.g., behavior of fluids or clothes), only where a user is attending to, while computational resources can be saved elsewhere by using low-quality solutions. Besides making computationally expensive effects more applicable in real-time graphics, this approach could also be used to reduce energy consumption of graphics hardware in mobile devices.

Apart from using high-fidelity graphics, realism can be also increased by taking into account the physiology of the human eye. While traditional computer-graphics applications are based on the pinhole camera model, there are several properties of natural vision which are not captured

by this model: stereoscopic vision, lens accommodation and luminance adaptation. The first arises from the fact that humans have two eyes and leads to depth perception. The second stems from the finite aperture of the lens and leads to blurring of out-of-focus regions, an effect denoted as depth-of-field. The third property permits visual perception under very dark and bright light conditions and changes the way how we perceive a certain range of colors. We hypothesize that with attention-aware approaches, we can increase plausibility and viewing comfort in the perception of these effects by dynamically adjusting the configuration (e.g., focus distance) taking into account the actual focus of a user.

Thesis Overview

In the following Chapter, this thesis will be continued with an introduction to the interdisciplinary background of this work. This includes a summary on basic knowledge about visual attention, a selective review on eye-tracking technology and applications related to computer graphics, and a state-of-the-art report on computer-graphics applications that make use of attention-aware techniques.

As mentioned above, Chapters III to V present the original research carried out for this thesis. The scientific contributions made with these chapters will be also discussed in the conclusion (Chapter VI) with respect to the main findings and new questions arising from this research.

CHAPTER 2

Background: Visual Attention, Eye Tracking and Computer Graphics

> Never trust a citation from the internet
>
> Leonardo da Vinci

This chapter covers the related work and interdisciplinary background of this thesis in a broader context, while previous work that is specifically related to one of the three research topics presented in Chapters III to IV will be reviewed there.

With Sections 2.1 to 2.4 this chapter gives a brief overview about what is known about visual attention. Section 2.5 continues with a brief introduction to eye-tracking technology, including a brief history of this technology and a review on related work on eye-tracking applications concerned with computer graphics. Section 2.6 will review the state-of-the-art on attention-aware computer graphic applications and Section 2.7 will conclude this chapter with a discussion of the potential value of computer graphics for research on visual attention.

2.1 Introduction: What is Attention?

Without doubt, attention has been one of the most intensively studied subjects in science in the past centuries. Starting from philosophic considerations, being later one of the main topics in psychology, and becoming recently investigated with novel methods from neuro-physiology that attempt to "take a look" inside the brain, attention has been a subject considered from quite different scientific perspectives and ways of thinking.

While attention is a well-known term that is often used in daily language, there is no clear answer what attention actually is. It has been related to eye movements, to the ability to focus on one's voice in a mixture of distracting voices, the capability to stay concentrated in cognitive tasks, or seen as the gate to consciousness. Attention appears in all sensory modalities, as well

as in cognitive processes, such as memorization, remembering, or reasoning. To understand attention, several metaphors have been used [52]. Prominent metaphors for attention are: The information filter metaphor, which explains attention as a bottleneck to the input of a limited capacity processor (e.g., [21]); the spotlight metaphor, which considers attention as an enhancer that mentally highlights an attended spatial region (e.g., [151]); the pre-motor control theory, which views attention as mechanism that directs motor actions, such as grasping or eye movements (e.g., [144]); and the boolean map metaphor, which proposes to represent the content of attention as a binary mask data structure encoding a set of attended locations in the visual array (Huang and Pashler [73]).

While metaphors are illustrative, they also guide researchers in building models that give rise to predictions which can be falsified by experimental investigation. This is often done by drawing analogies between the metaphors source domain (e.g., spotlight in a theater) and the target domain where the metaphor is applied as explanatory instrument (e.g., spatial region mentally highlighted by attention). But, attention did neither evolve analogously to the metaphors that we use to understand it, nor is it an anatomically separate module which operates independent of data processing, as once suggested by Posner and Petersen [137]. Attention is rather a category representing a set of important functions which are crucial to coordinate perception, cognition, eye movements and motor actions.

The following three sections give a selective overview on the most prominent ideas, metaphors, rules, theories and models which evolved with research on visual attention. The descriptions given in this manuscript are intended to sketch an intuitive way of understanding, which may not provide a rigorous explanation of all relevant details. This summary will be structured according to the three questions:

- *Why* do we need attention ? (Section 2.2)
- To *what* is attention directed ? (Section 2.3)
- *How* attention is directed ? (Section 2.4)

2.2 The Purpose: Why Do We Need Attention?

2.2.1 Information Filtering

Filtering of irrelevant or less important information is one of the major purposes which have been attributed to attention. Attention can be seen as a prioritization strategy that balances computational workload in a cognitive system according to the presumed relevance of the information to be processed. The basic assumption behind this view is that perception and cognition are information processing problems which are solved in a nervous system that implements an information processor. From using the information processing metaphor, Broadbent [21] concluded that perception and cognition must be subject to computational capacity limitations and proposed that attention controls which information passes the bottleneck between sensory devices and a limited-capacity processor.

Taking also the physiology of the eyes into account, the two moving eyes can be seen as information filter in action. Prior to processing at an internal stage of the perceptual pathway,

where sensory information is filtered after being sensed, visual information is already filtered in the eyes. This type of filter mechanism is actually *embodied* in the physiology of the retina, the eyes' optical properties, gaze behavior and the peculiarities of binocular vision. One factor is the strong fall-off of visual acuity from the center of the fovea to the periphery. Hence, our eyes selectively sense most information at the point we are currently looking at. Besides this, the vergence and accommodation systems, which can bring only a limited depth range simultaneously into focus, can also be considered as a filtering mechanism, but in depth direction. When we look at an object with both eyes, we can only fuse a certain range of disparities between corresponding features in the images projected onto left and right eye retina. This range is denoted as *Panum's fusional area* and separates fusible objects from other objects seen with retinal disparities outside this range, which are perceived with double vision. Though less pronounced than in stereoscopic vision, a focus and out-of-focus separation may also arise from the natural depth-of-field of the eye lenses. Hence, physiological limits of the eyes require to perform a selection in a three-dimensional volume defined by the cone of foveal vision and the limits of stereoscopic fusion and the eye lenses' depth-of-field. With every binocular shift of gaze and lens focus, the visual system performs a spatial selection. Another example where gaze behavior determines filtering of visual information are smooth pursuits, which occur when a user's eyes track a moving object by compensatory eye movements. During smooth pursuits, objects that move in a similar trajectory as gaze are seen in more detail, while other objects (also stationary ones) are sensed with motion blur, because of their motion relative to the retina.

2.2.2 Motor Preparation and Control of Eye Movements

Besides perception, attention has been also related to action. For instance, the *Premotor Theory of Attention* (PTA) of Rizolatti et al. [144] considers attention as the mechanism which prepares motor actions, such as grasping a target with our arms. Most obvious is, however, the relation of attention and eye movements. According to the PTA, there is a strong link between shifts of attention and shifts of gaze. However, gaze does not necessarily correspond to attention [181]. Since we are able to focus mentally on objects without looking at them, one commonly distinguishes two types of attention: one that moves the focus internally, which is denoted as *covert* attention, and an external manifestation through a shift of gaze, which is referred to as *overt* attention. The pre-motor theory of attention states that covert and overt attention are tightly coupled, and a shift of covert attention is most likely followed by a corresponding shift of overt attention. In other words, an covert shift of attention programs subsequent eye movements.

2.3 The Units: To What is Attention Directed ?

Attention can be directed to locations, to features and to objects. Figure 2.1 shows gaze patterns that are probably resulting from these three types of attention.

In Figure 2.1(a), we see an example for gaze behavior when attention is directed to a spatial location. In this scene, participants were searching for a particular car. Since new cars appeared usually from behind the wall at the corner of the street (left), the heatmap shows that participants deployed gaze often to this location while anticipating the next car to show up there.

(a) Gaze Directed to Space (b) Gaze Directed to Features (c) Gaze Directed to an Object

Figure 2.1: Different types of attention observed in gaze movement patterns.

Figure 2.1(b), shows a gaze path sampling particular features, such as the top edges of the green vase, the spout of the tea pot and the corner of the table lamp.

The heatmap shown in Figure 2.1(c) was recorded while a user focused on an object (blue Stanford bunny). In this condition, gaze is mostly directed to the center of the object, while gaze is located less often on details such as the bunny's ears.

The theoretic foundations and experimental evidence which lead to the differentiation of these types of attentional selection will be reviewed briefly in following:

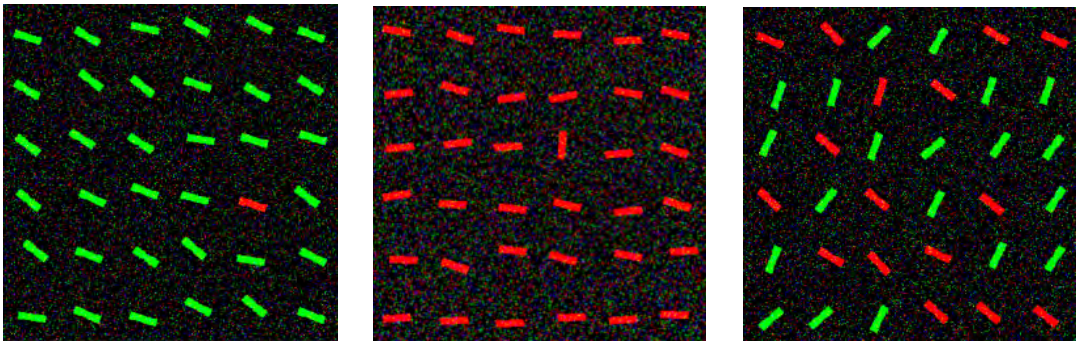
2.3.1 Space-Based Attention

When an attentional selection encodes a region of space, we speak of *space-based* attention. Evidence for spatial attention was given by experiments with the *cueing paradigm* [138]. In such an experiment, the task is, for instance, to detect an object which appears suddenly. If the appearance location is cued with a previously shown marker, response times are faster than in trials with no or misleading cues. From these results it was concluded that a space-based type of attention must exist which facilitates the response to events which happen in attended regions, like for instance the sudden appearance of a target object. The most prominent has become the spotlight metaphor [138], which compares space-based visual attention to a mental highlight that selectively facilitates perception in a circle-shaped region.

2.3.2 Feature-Based Attention

When we attend a feature, such as a color or an edge, we speak of *feature-based* attention [39]. For instance, during a search task, attention can be oriented to features that help to identify the target. Feature-based is also the most established model for visual attention – the Feature Integration Theory (FIT) of Treisman [178].

According to this theory, the visual stream is encoded along a small number of feature dimensions, most notably color, orientation, and luminance. These features are extracted in parallel to generate a set of feature maps. Since feature maps are built independently from each other, features of different dimensions that share a common location (e.g., a red horizontal bar) cannot



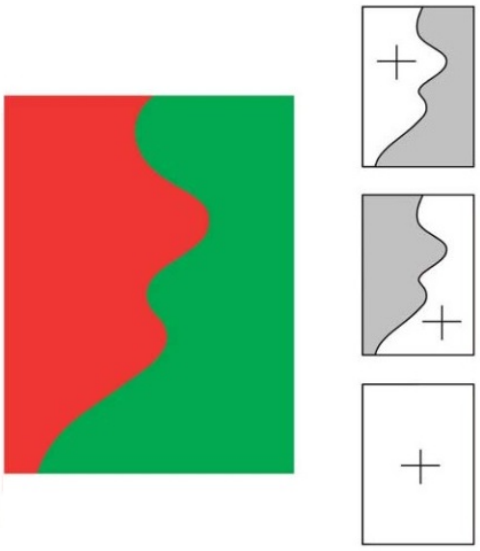
(a) Color Popout (b) Orientation Popout (c) Conjunction Target

Figure 2.2: Stimuli which are used to study visual attention with visual search experiments. These images were taken from [76].

be efficiently identified to correspond to one and the same object (e.g., a red vertically oriented bar). Hence, recognition of conjunctions of two or more features of different dimensions requires a procedure which is denoted as *feature binding* or *feature integration*. The FIT proposes that attention determines which features are going to be bound and that the binding occurs serially, that is to say, only one feature conjunction can be identified at a time. As a consequence, vision is considered as a process with two stages: a pre-attentive stage, where feature maps are generated by efficient parallel processes, and a much less efficient attentive stage, which integrates for a few selected locations spatially correlated features of different dimensions. The FIT is mainly based on experiments which used the visual search paradigm to study visual attention. In visual search experiments, a participant looks at a display which may, or may not, contain a previously specified target and has to respond whether the feature is present. Task performance is usually measured with reaction time. The main finding which led to the FIT was that the presence of a target (e.g, a red vertical bar) can be detected instantly if targets and distractors can be distinguished within a single feature dimension. In this case, feature singletons become visually salient and pop out due to a one-dimensional contrast to all other features. An example for the pop-out in color is shown in Figure 2.2(a) and another for orientation in Figure 2.2(b). On the other hand, Figure 2.2(c) gives an example where the pop-out does not occur, as two features have to be considered in combination to identify the singleton (the red vertical bar) in this image. FIT explains the tremendously increased difficulty to find the target with the fact that search for conjunctive features requires feature binding, which can only be carried out serially by deploying attention to one object after another. Since in this case, attention has to scan on average half of the items until arriving at the target, reaction time increases linearly with the number of elements in the display.

2.3.3 Feature-Location-Based Attention

A question that arises with feature-based attention models is how features are outlined in space. A feature can occur in a region, with a certain scale or size, and in one or many locations. While



(a) Feature-Location Maps (Huang et al.)

Figure 2.3: This figure (taken from [73]) illustrates three alternative feature-location maps for an object. Each data structure maps a unique feature value (e.g., low illuminance) to a set of occurrence locations.

the FIT stays rather vague about how feature values and their occurrence locations are related, the Boolean Map Theory (BMT) of Huang and Pashler [73], a more recent and promising theory for visual attention, makes the clear claim that features and locations are inseparable. To represent an attentional selection, the BTM proposes a hybrid approach with a data structure denoted as *Boolean map*. This data structure maps one feature value (e.g., green) to a set of locations $l_1, \ldots, l_i$ which bear this value.

Figure 2.3(a) shows a stimulus which gives rise to three alternative feature-location maps: one map for the color "green" , a second for the color "red", and a third that separates the square from the white background in feature dimension illuminance (i.e., "dark") .

2.3.4 Object-Based Attention

Object-based attention gives rise to the coherent perception of an ensemble of features that constitute an object. This is achieved by a mechanism which groups multiple features within the outline of an object through a comparison with internal representations that have been learned by experience [4]. First evidence for object-based attention was given with a phenomenon called *same-object advantage* (SOA). SOA is observed when two features that belong to the same object have to be discriminated simultaneously and was discovered by Duncan [48] in an experiment using the objects shown in Figure 2.4. Participants were exposed to one of these objects and had to discriminate features of several dimensions: the tilts of the line (clockwise vs counter-clockwise), the texture of the line (dashed vs dotted), the side of the gap in the box (left vs right)

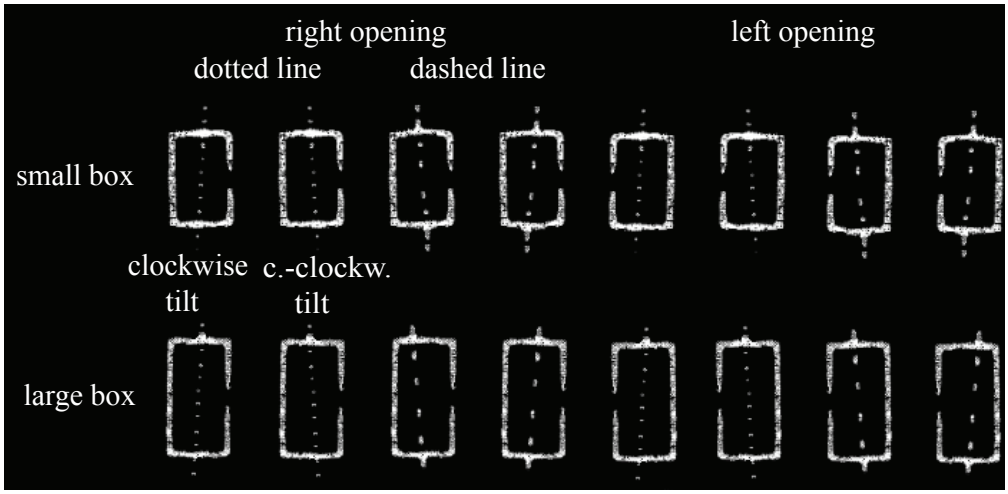


Figure 2.4: The stimuli used in Duncan's experiment where it was observed that discrimination of feature values (e.g., dotted versus dashed texture of the line) is more efficient when the feature discrimination is performed only on a single object (e.g., in every trial on features of the line). This illustration was adapted from [48].

or the size of the box (small vs big).

Participants were divided into a test and a control group. Members of the test group had to discriminate a pair of features of two different dimensions which belonged to the same object. For example, the texture and size of the line or the size or side of the opening of the box. The feature pair was fixed throughout the experiment for one participant. In the control group, participants had to discriminate two features each belonging to a different object. For instance, a participant had to discriminate the tilt of a line and the size of a box. The result was that the test group, which was judging features associated with the same object, had a higher rate of correct responses compared to the control group. The SOA gave rise to the hypothesis that attention is primarily object-based, and further evidence supporting this hypothesis was found in numerous similar experiments which followed [29].

In a later extension of the BMT, Huang [72] also proposed an object-based account. While the original version of the BMT assumed a feature-based data structure which maps one selected feature value to a set of locations, the object-based variant proposes that a scene is structured into a set of indivisible units corresponding to objects, and one object is selected at a time. An object is defined as a continuous region which may contain heterogeneous parts, but those have to be consistently separable from a background which builds a sufficient contrast to constitute the object's contours. While Huang [72] proposes that attentional selection can be object-based, access to properties of an object is strictly feature-based and limited to one feature label per dimension. This means that once a spatial region corresponding to the outline of an object is selected, the features of that object can be accessed simultaneously if, and only if, they belong to different dimensions. Thus, the SOA occurs under this condition, while perception of more than one feature value of one dimension (e.g., green and red) belonging to the same object is not subject to this advantage. It has to be noted, however, that the number of different feature

values which can be accessed at a particular moment in time does not prevent the integrated perception of multiple feature values of an object. Using memory and experience, the visual system can collect and combine visual information which has been accessed serially over time. The limitation of access to one feature becomes thus only apparent when a stimulus is exposed only for a very short time (e.g. 100ms).

Recent theories state that these different forms of selection are not mutually exclusive and appear depending on task and stimuli [177, 180]. Probably, all types of attention can also occur simultaneously, and different types of attention selections interact and compete [95].

In this thesis, it is consistently assumed that attention is object-based. While such an approach fits well to the object-based representations of scene graphs in computer graphics applications, we also believe that it is one of the most suited approaches in computer games and virtual environments, since objects are usually the basic units being sought for manipulation and interaction. In the spirit of the BMT [73], we assume that an object is a holistic feature that is related to a spatial region where the object occurs in the visual array. Thus, we use "object-location maps", which we represent in a data structure denoted as *item buffer*. The item buffer is a map which assigns to each pixel a key color corresponding to the object which is rendered to this fragment. This data structure is formally equivalent to a set of boolean maps corresponding to each visible object o_i in a scene (i.e., object-location maps of the form $olm(i, \{l_1, \ldots, l_n\})$).

2.4 The Control: How Attention is Directed ?

Beginning with first attempts on understanding attention [81], the intuition that there are at least two different mechanisms influencing what is attended holds up to nowadays theories [23]. On the one hand, attention can be volitional and what is, and will be, attended is controlled by intentions of the observer. On the other hand, attention can be driven by salient visual features. In the latter case, external factors determine what we are attending to.

In the left illustration in Figure 2.5 an example is shown where attention arises from the intention of a person ("I want to sit") and determines what attention selects to be perceived ("a chair"). Some theories suggest [190] that during visual search this selection is guided by a mental visual representation of the target object that predicts features which can be used to search for a previously unseen target. The example shown here should give an intuitive illustration of a causal chain which descends from higher-level cognitive functions to early loci in the perceptual pipeline where incoming sensory information is selected to be attended. Due to the causal direction from high-level to low-level processing stages, we call this type of attention *top-down* or goal directed attention.

On the other hand, the causal chain illustrated on the right-hand example goes in the opposite direction. In this case, an object ("red apple") in the stimulus becomes salient due to a high feature contrast (red on green background) and attracts attention. After the salient object is recognized, it will be associated with meaning and may influence thoughts and generate intentions ("I want to eat an apple"). Thus, this type of attention is denoted as *bottom-up* attention.

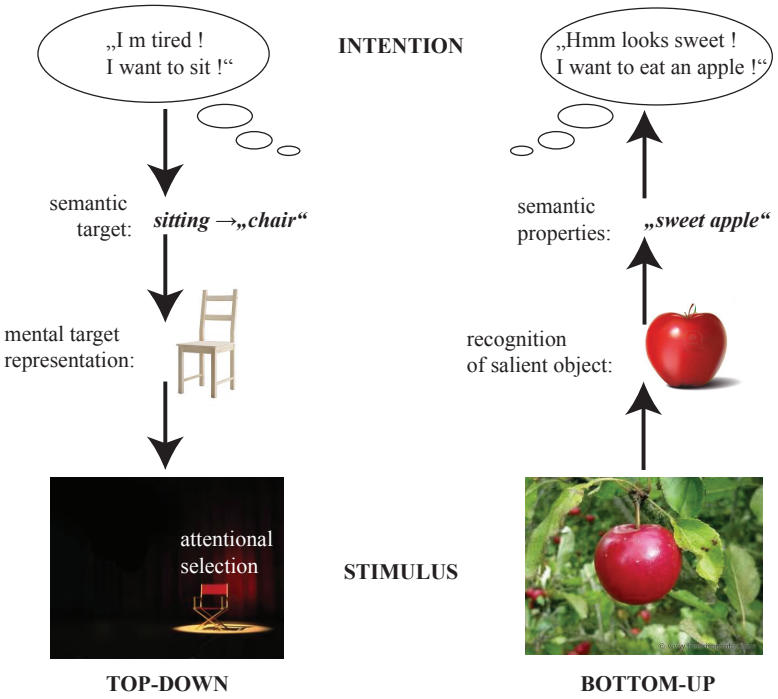


Figure 2.5: Intention driven versus stimulus driven attention.

2.4.1 Bottom-Up/Stimulus-Driven Attention

Bottom-up driven attention is not under volitional control, reactive to the properties of the stimulus, and memory-free. Bottom-up selection occurs at early stages of visual perception and thus is often attributed to *pre-attentive* stages of vision. Bottom-up attention acts all the time on all features in the stimulus and it is transient, i.e., bottom-up attention attraction persists only in a short time window. When the stimulus drives attention, we are attracted to certain features in the field-of-view which appear to visually pop out, that is to say, those features become *salient*. Salient are those image features which appear rarely, have undergone sudden changes, are surprising or novel. For instance, a right-tilted bar which is surrounded by a context with many left tilted bars becomes salient due to being unique (as shown previously in Figure 2.2(a)). An efficient way to score the rareness of a visual feature is to compute its contrast to the surrounding context.

This strategy applied in alignment with the model provided by Treisman's FIT gave rise to computational models that generate saliency maps as predictors for bottom-up attention.

Computational Models for Bottom-Up Attention

Koch and Ullman [91] provided the foundation for this computational model by providing a detailed neurally inspired mathematical framework, which specifies how features are extracted and processed to derive a salience map that topographically encodes the likelihood that attention

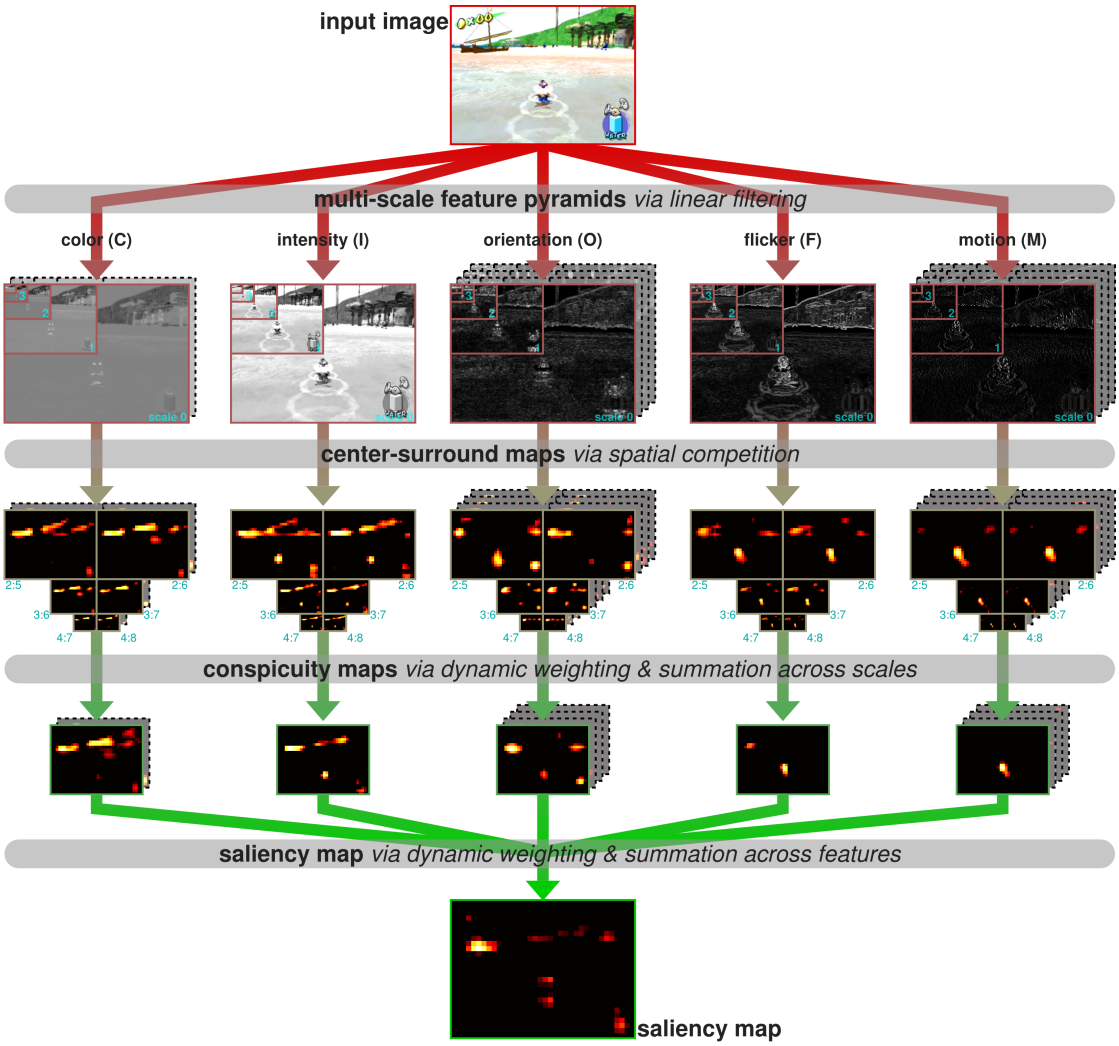


Figure 2.6: Saliency maps are state-of-the-art computational tools to predict image regions attracting the viewer's visual attention. This illustration, which was taken from [133], shows one of the most recent implementations that has been used to predict/model attention in a computer game.

is attracted to a certain location. This model was later advanced by Itti and Koch [79] to an algorithm which has been used in many fields of computer science, including computer graphics, computer vision and human-computer interaction.

A more recent variant of this algorithm [133] applied to rendered images of a video game is illustrated in Figure 2.6. This algorithm computes salience maps in four steps: first, topographic features maps are computed for each channel (e.g., color, orientation, flicker or motion); second, a *center-surround contrast* map is computed. Due to the fact that the visual system perceives at different levels of detail, both the feature maps and the center-surround contrast maps are generated on multiple scales. This means that the size of the kernels used for feature extraction and the size of context which is considered in the contrast computation varies with the respective scale. To this end, the input images are hierarchically decomposed into a pyramid containing in each level an image which is a down-sampled low-pass filtered version of the image from the preceding lower level. In the third step, center-surround maps from different scales are then combined by a weighted linear summation into one *conspicuity map* for each feature dimension. Finally, the fourth step integrates conspicuity maps of multiple feature channels into one *master map* by linear summation, which is the salience map output.

Overall, computing feature contrast at several scales is the main mechanism which determines saliency.

Saliency of Temporal Features and Change Blindness

Besides static features with a high contrast, it turned out that temporal changes are strong attention attracting features. Since temporal features, such as flicker or motion, are also processed by computing center-surround contrasts, salience maps predict the phenomenon *change blindness* [153]. Change blindness is the inability to notice changes when the stream of visual information is temporally disrupted, either due to a fast shift of gaze or due to a change in the entire stimulus (e.g., cut in a movie).

Temporal features become salient if and only if they occur in only few positions and thus produce a high center-surround contrast. A disruption of the visual stream means that there is a sudden change in every pixel of the input image. This raises the intensity of change features in all locations and thus avoids peaks in the center-surround map, which results in a flat distribution of salience magnitudes in the entire image. The flicker paradigm used in experiments to observe change blindness, exploits this behavior. With this paradigm, change blindness is provoked by interupting a video stream through briefly flashing a flicker image (e.g., a blank image). This produces a global change of all image features and effectively avoids that other temporal changes that occur at the interuption time in the original image stream become salient. Thus, after a flicker, an observer is barely able to notice changes in the visual field, even if those seem to be obvious, like a change of a person's clothes or skin color. This underlines the high importance of stimulus-driven saliency in a dynamic environment, which enables our visual system to efficiently process temporal changes. A fast perception of changes is crucial to react quickly to potentially important, or even dangerous, events.

Salient Features of Higher Complexity

In addition to conspicuous low-level features, attention can also be driven by faces or face-similar objects [99], which requires processing more complex features, or feature compositions, already at early stages of perception. Moreover, Walther and Koch [184] suggested that bottom-up saliency often correlates with feature ensembles which potentially constitute objects. They speculated that saliency provides an early object hypothesis – a "proto-object" – for items seen with peripheral vision. This hypothesis is then validated after an overt shift of attention when visual details can be perceived through foveal vision and sustained attention.

Multi-Modal Saliency

Recently, there is also a trend towards applying the concept of saliency to predict stimulus-driven attention in other sensory modalities. Algorithms have been proposed for saliency computation in the auditory domain, for example. To this end, features such as spectral properties, intensity and temporal changes (e.g., silent space of time) are detected to estimate the salient events and features in sound signals [85]. As perception interacts across multiple senses, this idea has been further advanced with multimodal models for bottom-up attention (e.g., [145]). Attention can also emerge "crossmodally" [40] when a feature in one perceptual modality (e.g., sound of ringing telephone) raises saliency of a congruent feature perceived in another modality (e.g., visually recognized telephone).

In summary, stimulus-driven attention is mainly feature-based, as it is a result of low-level processing taking place at early stages of vision. Stimulus-driven attention makes certain locations appear salient as a result of which they attract attention. Features become salient when they are rare (e.g., high contrast to surrounding) or presumably important (e.g., faces). This type of attention is particularly important to cope with novel and dynamic stimuli. In order to respond fast to relevant events in a dynamic environment, attention is most attracted by temporal features. Through a strong sensitivity to temporal features, such as motion and flicker, stimulus-driven attention provides an efficient strategy which minimizes both, the amount of information which has to be processed and the number of gaze movements that are required, in order to quickly update an observer's mental model of the external environment.

Stimulus-driven attention follows the same principles also in other modalities, such as auditory perception. Moreover, attention is even driven stronger by perceptual events or objects which are sensed congruently in several sensory modalities.

2.4.2 Top-down/Goal-Directed Attention

The first and most prominent example demonstrating that attention is also driven by intentions was given by Yarbu's famous eye tracking experiment [195], where participants were visually inspecting Repin's painting "The Unexpected Visitor" were given the instruction to answer a certain question about the depicted scene. Since each question elicited a different task, this resulted in distinct gaze movement paths (Figure 2.7). Under this condition, eye movements are less driven by the stimulus, which was constant in all conditions, than by the task of the observer (e.g., answering a question about scene properties).

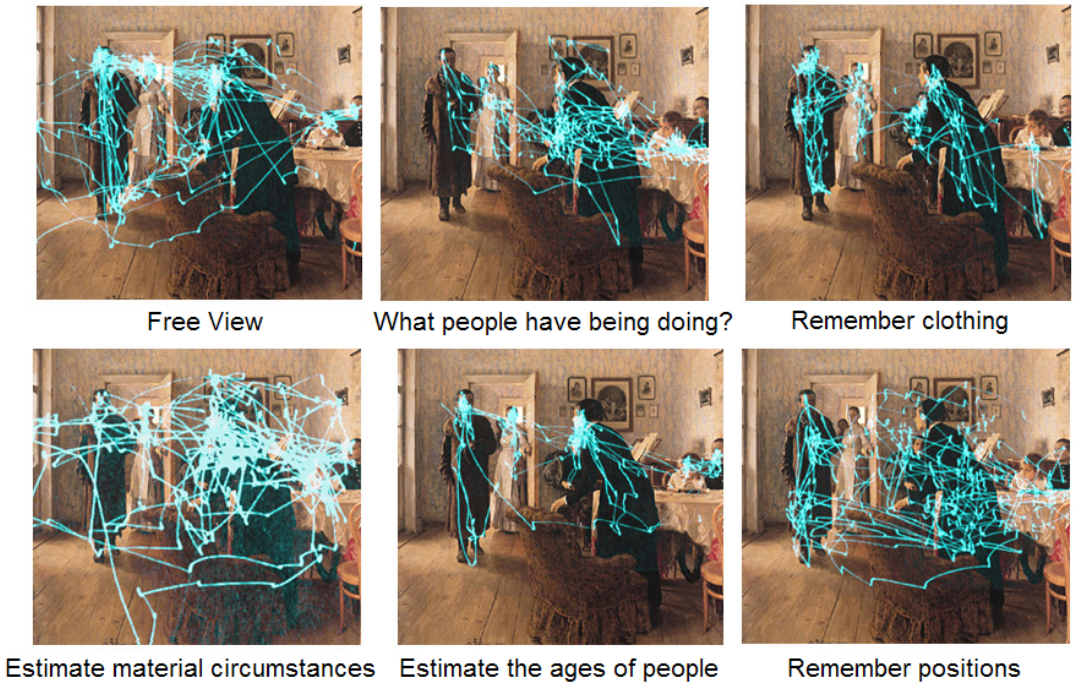


Figure 2.7: Yarbus [195] (1967) showed Repin's painting "The Unexpected Visitor" and recorded eye movements. The observer was asked different questions to investigate how visual scan paths differ according to the observer's task. This image was adapted from [195].

Sustained Attention

As opposed to bottom-up attention, top-down attention is persistent over an intentionally controlled period of time. Therefore, goal-directed attention enables a sustained perception of a particular object or feature. Moreover, it has been suggested that voluntary attention preserves continuity over rapid changes in the stimuli (e.g., change of camera perspective in a movie) and eye movements [68], though this appears somewhat controversial with the change-blindness phenomenon. However, top-down attention also involves memory. Thus, it may preserve a stable internal representation of one or a few attended objects or features in the external scene. Since continuity is only preserved for one or a few task-relevant elements, changes of other details are not noticeable, and thus an observer is still blind to changes on the majority of objects which remain unattended.

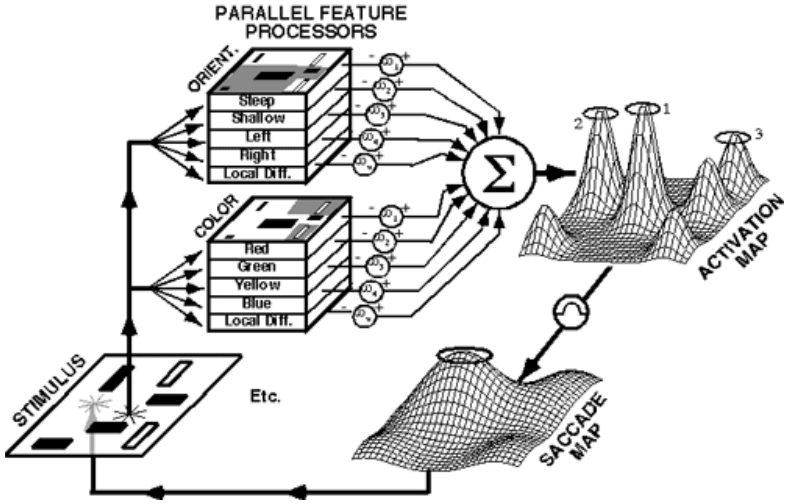


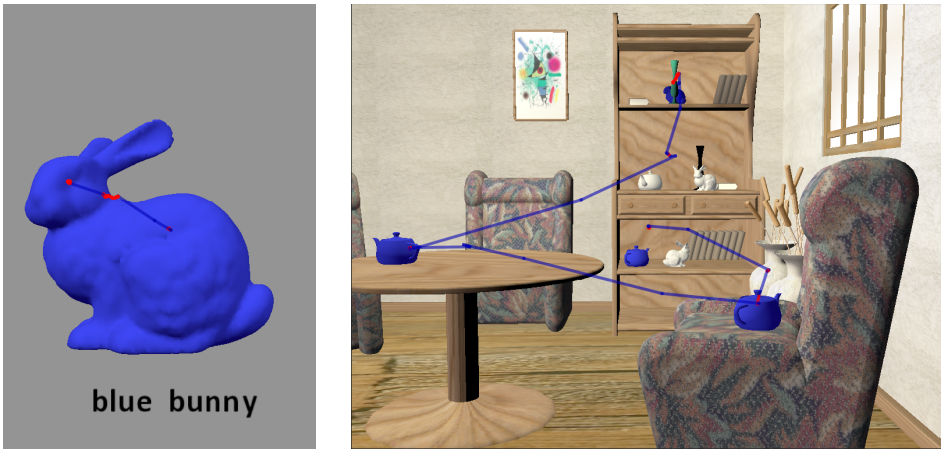
Figure 2.8: Wolfe's Guided Search model [190]. This illustration was taken from [76].

Inattentional Blindness

Top-down attention is actually highly selective and neglects most visual information that is not relevant for an observer's goals or task. The high selectivity of visual attention was underlined with the metaphor *inattentional blindness* [107]. A famous example was given with the "invisible gorilla"- experiment [152]. Participants saw a video clip of a team playing basketball and were instructed to count the number of passes between members of one of two teams. Due to being focused on the counting task, the majority of participants did not notice a person dressed as gorilla running through the scene, even though the gorilla crossed several times the foveal field of view of the observers.

Attempts to Model Top-down Attention

While it is difficult to model top-down attention as it evolves cognitively, there are some attempts to cover certain aspects of top-down attention. One of the most profound attempts to integrate top-down factors into (existing) models for visual attention was provided by Wolfe with Guided Search (GS) [190]. GS extends the FIT with a top-down component which bias the salience of features which are sought, as illustrated schematically in Figure 2.8. According to this model, attention spatially guides search with a master activation map that is derived by a linear summation of each feature dimension's conspicuity map. Top-down influence is modeled as a feature-specific weighting coefficient in the linear summation. For instance, during search for a blue object, the weighting coefficient for the blue color is raised in order to make blue objects appear to be salient. Such a case is shown in Figure 2.9 with gaze data we recorded from a user performing a visual search for a blue-colored bunny. We see that gaze directed to several locations which contain blue color until finding the target object. More complex than the search for a specified feature, such as a certain color, becomes search when only the object category is known. This requires recalling an internal representation of the target category in order to guide



(a) Target (b) Visual Search Path

Figure 2.9: Gaze path in a visual search task. Image (a) shows the target (a blue bunny) a participant was instructed to search. In image (b), we see the gaze path which shows that several distractors (other blue objects) were visited before the target could be found.

the search for features which are most characteristic for this category of objects.

GS provided the first theoretical foundation for computational models that attempt to include top-down control. GS is limited to modeling attention during visual search, which is, however, the most fundamental task of vision. Thus, computational models based on GS need to encode each higher-level task into a sequence of visual search tasks, as proposed by Navalpakkam et al. [123]. Their approach first determines the task relevance of each entity in the visual field with an ontological model of the scene and task. From an internal visual representation of these objects, visual features are then predicted and passed as input to an algorithm which implements GS. The biological plausibility of this computational model was validated using simple tasks and a rather small set of natural images and video clips.

Despite the complex nature of top-down attention, recently there have been some promising advances towards computational models that predict top-down attention [16–18]. Predicting top-down attention becomes possible in very constrained situations, such as when a user plays a computer game and has some clear goals and tasks. This thesis further extends on methods to predict top-down attention in the related work section of Chapter 4.

2.4.3 Interaction Between Top-Down and Bottom-Up Attention

There is an ongoing discussion about whether top-down and bottom-up attention interact or are independent. Recent work tends to deny that there is interaction between top-down and bottom-up attention and states that both processes are independent of each other [136].

Chapter 4 of this thesis ventures a novel approach on the study of top-down attention. To this end, we proposed a framework allowing to shape an ontology which is required to determine

task relevance dynamically according to the behavior of a user. With this approach, we attempted include those variables which allow us to infer high-level behavior. Using, for instance, semantic descriptors and user inputs should provide us the information to infer the current task and the respective target objects during a user's interaction with the virtual environment application.

2.5 Eye Tracking

Eye tracking is the technology of choice to study visual attention. This section begins with a brief introduction to how eye tracking evolved as a technology. Then previous work on applications making joint use of eye tracking and computer graphics, particularly in virtual environment applications and computer games, will be reviewed.

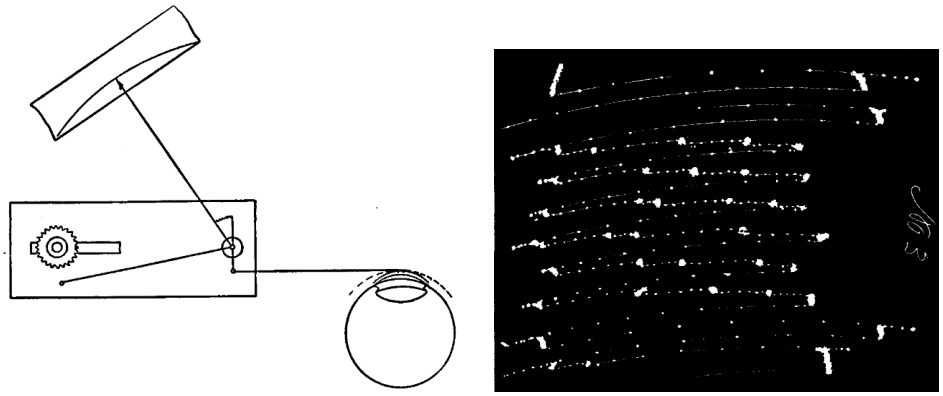
2.5.1 Brief History of Technology

Mechanical Recording

As many other methods of investigation in psychology, the first steps of understanding gaze behavior and visual attention were done by the method of introspection. The first objective attempts to observe eye movements are reported from the 19th century, when techniques using pinholes, telescopes or mirrors were developed. Thereby it was early understood that eye movements are jerky and alternate between *saccades* and *fixations*. Studying eye movements with a mirror during reading tasks, Javal et al. [82] were able to provide a first objective evidences for this behavior. While motion properties could be observed already well with early eye tracking techniques, it was more difficult to accurately measure *where* gaze is directed to. Early techniques for measuring fixation locations more accurately were using intrusive mechanical devices. For instance, Huey [74] attached a contact lens to the cornea which transmitted eye-movement forces over an aluminum pointer to a kymograph, which recorded the movement trace on a rotating drum cylinder (Figure 2.10). Since this could be painful, inconvenience was reduced by putting small doses of anesthetics on the cornea. However, since invasive techniques which attached lenses to the cornea were the most accurate option, they had been used in many experiments until the mid 1960s. For instance, by Yarbus [195] to obtain the recordings shown in Figure 2.7. But with the emergence of better alternatives, these techniques have become less popular and nowadays they are only used for experiments with animal subjects [143].

Electro-Occulography

A less inconvenient alternative to lens-based eye tracking is Electro Oculography [49] (EOG). This technique measures electric potentials with electrodes attached to the skin near the eyes. From this signal, the magnitude of muscle contractions involved in vertical and horizontal eye movements can be estimated. Since EOG measures eye movements only relative to the head, additional head tracking or maintaining a fixed head position (e.g., with a chin-rest) is necessary to determine the point of regard [42]. An advantage of EOG is that eye movements can be robustly measured, even if eyes are closed, which allows, for instance, to sense eye movements during sleep (e.g., to identify REM sleep intervals). Moreover, since saccadic movements generate



(a) Huey's Eye Tracker (b) Gaze Recordings

Figure 2.10: (a) Setup used by Huey [74] (1898) to observe eye movements. In this apparatus forces resulting from eye movements are transmitted by via an aluminum pointer attached to the cornea to a rotating cylinder. The recorded gaze path of a subject performing a reading task are shown in (b). This illustration was taken from [74]

spikes in the EOG which are well-detectable, EOG is suited for a temporally accurate saccade identification, even for ultra-short saccades [87] (e.g., micro saccades). However, EOG is prone to errors caused by muscle artifacts and other sorts of electromagnetic interference resulting from neural activities near and in the eyes eyelid [198]. EOG has found wide application during the mid 1970s [198], but in the long run, optical eye-movement tracking, which is described next, has become the most accepted technology.

Optical Eyetracking

The great advantage of optical gaze sensing is that it can be done non-intrusively and hence avoids distractions caused by uncomfortable sensations in or near the eyes. First attempts at optical tracking of eye movements can be found in the work of Dodge and Cline [37], who recorded corneal reflections on an optical plate to determine the horizontal angle of the eyes. The corneal reflection is the white specular highlight which occurs close to the pupil on the cornea. Tracking the corneal reflection can be used to infer gaze direction, since its location is more or less invariant to small head movements while it moves together with the eyes' line of sight. A famous example can be found in Mackworth and Mackworth's work [108], where the corneal reflection was used for directly visualizing an approximate gaze position by superimposing images of the corneal reflection spot on the video captures seen by the eye-tracked participant. The superimposition was calibrated by maximizing the alignment of the reflection spot with the actual gaze position. Further accuracy can be gained by extracting the position of the pupil. Determining the spatial relation of the pupil relative to the corneal reflection spot (Figure 2.11) has proven to be one of the most accurate and robust techniques, and found wide application in commercial high-quality eye trackers [42]. A problem was that the contrast between the cornea and the pupil spot is low under natural lighting conditions. Thus, modern eye trackers project an infrared or

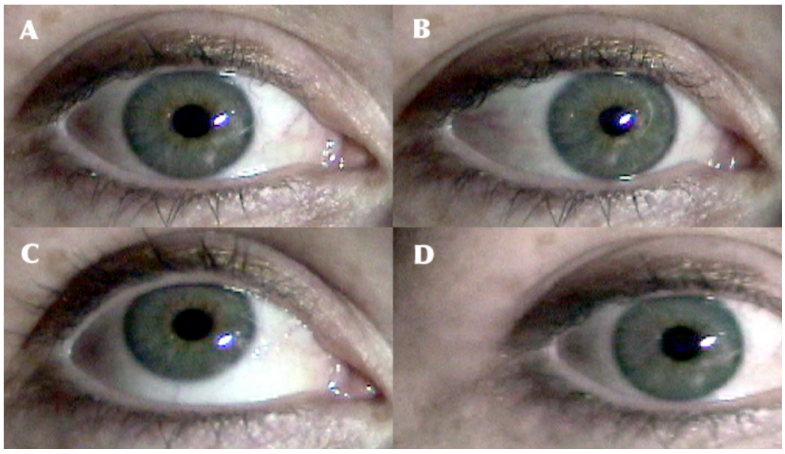


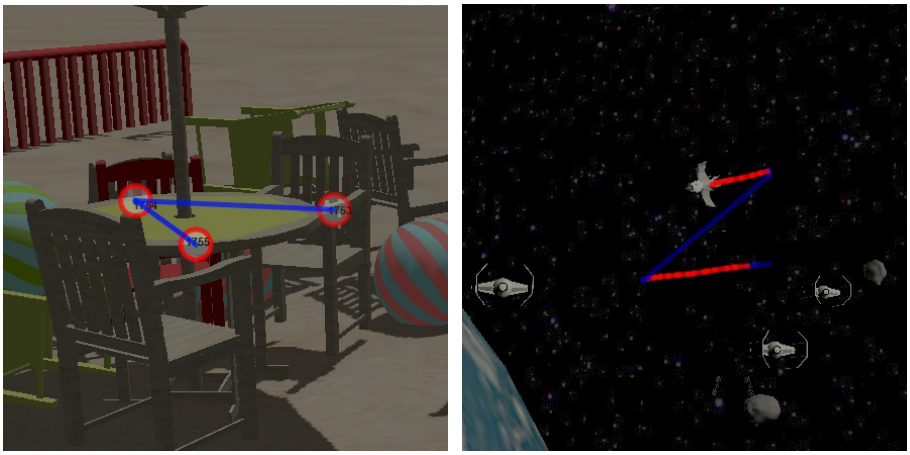
Figure 2.11: Corneal reflection on the eye occurs as a white dot close to the pupil (A). While the rotation of the eye in vertical (B), horizontal (C) changes the relation between pupil center and the reflection spot, this offset is invariant to head translations given the gaze direction is stable. This image and the description were taken from [143].

near-infrared light on the eyes which is captured by a high-resolution infrared camera. With this technique, images with a high contrast between the pupil and iris can be obtained. There are two variants of infrared-based eye tracking techniques: *bright-pupil* tracking and *dark-pupil* tracking. Bright-pupil tracking techniques project the infrared light from a position which is close to the optical axis such that light entering the interior of the eyes is reflected directly back into the camera. This technique produces a red-eye effect intentionally and the pupil position can be detected by segmenting bright image regions. For dark-eye pupil tracking, the infrared light source is placed off the optical axis. In this case, infrared light is not reflected from the interiors of the eye balls and the pupil appears as a dark spot.

The advantage of bright-pupil tracking is that it creates a higher contrast, allowing a more robust detection of the pupil. However, since bright-pupil tracking requires to discriminate illuminated regions in the infrared spectrum, it is less suited for outdoor use where sunlight which contains a high amount of infrared light may interfere. Bright-pupil technique is the best method for indoor use with people having a light brown, red and blue iris color (e.g., Hispanics and Caucasians), while the dark-eye pupil tracking has proven to work better for subjects with very dark or black eyes (e.g., Asians). To benefit from the advantages of both techniques, more recent eye tracker releases (e.g., Tobii Eye Trackers of the T/X Series) use bright and dark pupil tracking in conjunction and switch dynamically to the technique which is more accurate [1].

2.5.2 Processing of Raw Gaze Data

When an eye tracker is used to sense gaze while a user is looking at a display where the stimulus is presented, the output is an array of raw gaze points, which are defined by a timestamp and a 2D position on the screen. From this data, the actual foci of user attention are determined. This requires to pre-process gaze data to identify *when* a user is perceiving, because the visual system



(a) Saccades and Fixations (b) Smooth Pursuits and Saccades

Figure 2.12: A visualization of processed gaze data. We used a fixation filter which separates saccades (blue) and fixations (red). Between saccades, two types of gaze movements can occur: either stationary fixations (circles in left image) or smooth pursuits (red gaze trace in right image), which occur when a moving object (e.g., flying space ship) is tracked.

can only process visual information when the image projected onto the retina is (relatively) stable.

Most Important Gaze Patterns

A stable retinal image is maintained during *fixations* or *smooth pursuits*. If gaze is stationary for a sufficiently long time (e.g., 100ms), we speak of a fixation. Fixations usually occur on objects which are not moving. To enable the perception of objects in motion, the oculomotoric system usually attempts to move gaze in the same direction and velocity as the attended object in motion. This tracking behavior is denoted as smooth pursuit and preserves a stable retinal image of the moving object. Between fixations or smooth pursuits, gaze moves rapidly in *saccades*. Saccades are fast movements of gaze which allow the eyes to quickly jump from one interesting point to another. Fixations, smooth pursuits and saccades are the three most prominent patterns we can use to abstract gaze behaviors, which can be seen in the gaze paths visualized in Figure 2.12.

For simplicity reasons, we will not continue to literally distinguish smooth pursuits and fixations. Instead, we consider a smooth pursuit as a fixation in motion. In contrast to these two types of fixations, saccades disrupt the stream of visual processing [128]. This means that a user's overt attention correlates with fixation times only [42]. Thus, prior to further analysis of gaze data, it is useful to process the raw gaze signal to separate fixation times from temporal intervals where saccades are performs.

Fixation Identification Methods

Several fixation-identification techniques have been proposed (see [146] and [42] for a review). According to the taxonomy proposed by Salvucci and Goldberg [146], fixation identification algorithms can be categorized based on their spatial and temporal characteristics. Spatial fixation filters may take into account the velocity of gaze samples (velocity-based), the spread distance of fixation points (dispersion-based) or they may use an area of interest (AOI) to identify gaze samples belonging to a fixation (area-based). Temporal filters may be locally adaptive by exploiting the temporal coherence of gaze, or use a time duration threshold to account for the fact that fixations occur after 100 ms.

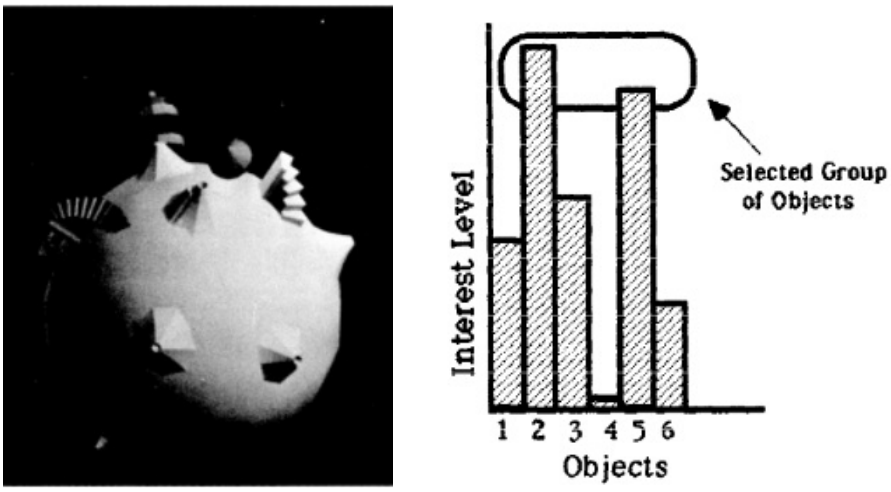
Salvucci et al. [146] analyzed the performance of various fixation-detection methods and concluded that dispersion-threshold filters are most accurate. However, Salvucci et al.'s study was performed using static stimuli. In interactive 3D graphic applications, such as paced video games, one has to assume a high optical flow due to rapid object or viewpoint motion. Under these conditions, the eyes attempt stabilizing the retinal image by tracking moving features. In these applications, dispersion filters are less appropriate. As dispersion increases proportional to drift velocity, dispersion filter may artificially separate a smooth-pursuit into a series of short fixations. Therefore, for highly dynamic stimuli, velocity-based fixation-detection methods are more appropriate, as they cluster together all gaze points which belong to an uninterrupted smooth pursuit. Since unprocessed gaze data has a considerable amount of noise, it is further also useful to low-pass filter the gaze signal before using velocity-based fixation-detection methods.

2.5.3 Eye-Tracking Applications Related to Computer Graphics

Eye-tracking technology is currently evolving from a costly scientific laboratory equipment to an affordable consumer-hardware accessory. Thus, it is possible that future commercial products, such as game consoles or tablets, will be equipped with eye trackers. There are many applications fields where eye tracking has been used, ranging from industrial engineering, psychology, neuroscience, human factors and marketing/advertising to computer science (a broad survey can be found in [41]). However, the focus of this section will be on the use of eye trackers in real-time computer-graphics applications (e.g., games and virtual environments). To this end, this section will review several previous-work examples where eye tracking was used with stimuli generated by real-time computer graphics. The section will conclude with previous work on gaze processing with 3D stereoscopic displays, which has recently become an interesting field in eye-tracking research.

Attention-Guided Narrative Speech Synthesis

A famous experimental application of eye tracking in computer-graphics displays has been proposed in 1990 by Starker et al. [156]. They used gaze information recorded in realtime to guide synthesis of speech in a way that narration refers to the current object a user is attending to. The scene they used in their example application was a planet with several objects, such as volcanoes or flowers, which were placed on the planet's surface. Gaze was mapped to the object located



(a) Planet (b) Interest Levels

Figure 2.13: Starker and Bolt [156] used an eye tracker to narrate information according to an observer's gaze behavior. While the user was looking at the objects located on the planet shown in **(a)**, gaze data was collected to compute a gaze statistic in object space. These statistics were then used to derive interest levels, which are depicted in **(b)**. The system performs a selection of those objects with the currently highest interest levels in order to determine the subject being narrated. These illustrations were adapted from [156].

on the planet to compute an *interest level* (Figure 2.13). The objects with the currently highest interest levels were then chosen to select a related topic to be narrated with synthesized speech. As a user might switch attention between several objects, the narrated information was chosen according to a common semantic category shared by objects currently attended.

Gaze-Controlled Collaborative Virtual Environments

Gaze behavior also plays an important role in face-to-face communication as it enhances our awareness of the attention of others. Moreover, seeing another persons gaze helps us to determine to where, or which objects, another person is pointing to, which may considerably improve communication and collaboration in a task [119]. To mimic this behavior, one can animate the eyes of virtual agents in collaborative virtual environments. Using an eye tracker to determine the current orientation of a user's gaze, Steptoe et al [162] animated the gaze of an avatar such that it appears to react responsively to the gaze behavior of the user (e.g., by mimicking some of the user's gaze patterns).

Studying Attention with Virtual Environments

Eye tracking has been often used in real environments to analyze gaze behavior in natural tasks. Famous studies were conducted with easy tasks ranging from hand washing to sandwich making [22, 60, 130]. However, carrying out an eye tracking study in a real-world setup is challenging,

as it requires a mobile eye tracker and a registered camera to record what a participant has seen throughout an experiment. Since virtual environments and computer games provide a test bed to study eye movements that is more convenient to control, these applications are on the verge of becoming important as tools to generate (interactive) stimuli for eye tracking experiments.

Virtual environments found their application also in the field of vision science. For instance, Hayoe et al. [60] used a virtual-environment setup to carry out a series of experiments to study the gaze behavior in a navigation task. A comparison between gaze data and predictions of a saliency map revealed a rather weak correlation. Nevertheless, they found significant similarities in the distributions of gaze points recorded for different subjects. This result supports our hypothesis (Chapter 4) that gaze recorded for one person predicts gaze of another subject. In addition to that, they found that a user's current task can be inferred from the observed gaze (which should then also work in the reverse direction).

Studying Attention in Video Games

An early attempt at using eye tracking for a commercial computer game was carried out in Kenny et al.'s work [86]. They recorded gaze in a first-person shooter game and analyzed the 2D spatial pattern in the gaze distribution. The main finding was that in this type of game, the majority of gaze points is located close to the center of the screen.

The spatial extent of a video game player's visual attention was also investigated by Yokoi et al [197]. They used a circular gaze-contingent window that was dynamically moved with a user's gaze. The gaze-contingent window allowed a user to see the game only in a fraction of the display in normal colors, while the rest of the screen was considerably darkened. They studied how the radius of the gaze-contingent window affects gaming performance and found that a window size below 20° significantly reduces gaming performance in a ski-racing game or in a rhythm action game, while no effect of window size was found for a puzzle game.

Another early study making use of video games was done by El-Nasr et al. [51], who concluded that both, top-down task relevance and bottom-up salience, influence gaze behavior in computer games. Since they used a commercial game, they had to capture the sequence of rendered images as video recording. At the analysis stage, this required them to perform a tedious manual preparation by tagging regions of interest manually (frame by frame!). However, having access to the internals of a computer game permits to automate this process. A first step into this direction was proposed by Sundstedt et al. [170] and furthered by the work presented in Chapter 4 (Gaze Analysis). Since then, several alternative approaches to support gaze analysis in studies carried out with virtual environments and games emerged [121, 122, 147–149], which will be reviewed in the related work section of Chapter 4.

Gaze-Controlled Video Games

Besides studying gaze behavior, eye trackers can be also used to control the difficulty of a computer game. For instance, Jie et al. [83] developed a shooter game connected to an eye tracker informing the application about the current gaze location and movement of a user. This information was processed by the game engine to decide where game elements (e.g., enemies) should be placed in the environment. With this design, they constructed two difficulty levels: a hard level,

where enemies spawn at positions raising a high attentional cost (e.g., in the opposite direction of a smooth pursuit gaze movement), and an easy level, which makes enemies appear where a user can attend to more easily (e.g., in the direction of a smooth pursuit).

An application which is increasingly attracting interest is user interaction via gaze control in games. Since good eye trackers are still expensive, these types of gaze-controlled video games are in their infancy. Yet it is an open question whether gaze-controlled user interaction will find broad application in commercial applications of the near future. Isokoski et al. [75] reviewed the early state-of-the-art of gaze-controlled games and discussed the potentials and limitations of this application field. One of the main problems is the low accuracy of gaze pointing compared to mouse control. Another problem can be that gaze is not under full volitional control and shifts of gaze are crucial to orient visual attention in scene viewing. One issue which is difficult to avoid is the Midas Touch problem, which means that commands are inadvertently issued every time a user looks at a task-relevant object [80]. Thus, the application of gaze-control for games which require continuous position control, or a dissociation between focus of attention and control, is very challenging, or even inappropriate. Less suited are also games where a large number of commands are used for interaction. Isokoski et al. [75] found that gaze control is perceived positive by most users, but stated that this could be primarily due to the fascination for the technology and the novelty of experience. Longitudinal studies to investigate the potentials and user acceptance in long-term use are, however, pending. At least in games which were not designed for the use of eye tracking, a player using gaze control is most likely outperformed by users of traditional input devices. This may cause frustration in multi-player games with competitors who do not use gaze-control. Undoubtedly, gaze control is of great utility for physically handicapped people. There are also some particular game genres which could benefit from this new type of user input, but whether equipping game consoles or gaming PCs with eye trackers is in general worth while for gaze controlled gaming cannot be predicted yet.

A more detailed review on this topic can found in Sundstedt's book "Gazing at Games" [165], which provides a broader introduction to recent work on using eye trackers to study gaze behavior and improve user interaction in computer-game applications.

Eye Tracking in Stereo 3D Applications

Another interesting field is the use of eye trackers with stimuli shown in 3D stereo. A very interesting problem which emerges with stereoscopic stimuli is how to determine a gaze point in three dimensions from a 2D gaze signal. This requires to obtain an additional depth coordinate, which is denotes as *gaze depth* [47]. Methods to obtain a 3D gaze point can be divided into two categories: *geometry-based* and *binocular* techniques.

Geometry-Based Gaze-Depth Estimation: *Geometry-based* approaches map a (monocular) 2D gaze position back to the 3D geometry or objects of a scene. This approach relies on the assumption that the gaze depth, which is the depth where both eyes' lines of sight intersect, corresponds to the depth of the geometry's visible surface underneath the gaze position. This can be, for instance, implemented on graphics hardware by rendering a depth-buffer image from the scene geometry and selecting the depth of the pixel(s) near the current gaze position. A problem arising with this approach is that a user's gaze may not point accurately on the actually

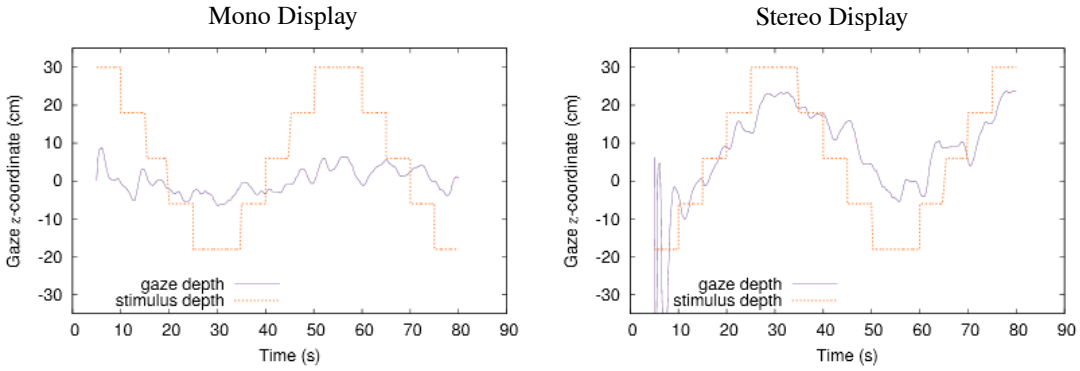


Figure 2.14: Measuring gaze depth with an eye tracker: the left image shows how gaze and stimulus depth are relatively independent if the display showed a scene in mono, while a coarse correlation between gaze and stimulus depth can be observed with stereoscopic displays. The images were adapted from [47].

focused pixels or scene geometry. Instead gaze is more distributed in a region of uncertainty which may contain many different depths. Thus, instead of using plain geometry information, such as triangles or pixels in a depth buffer, to infer gaze depth, one can also assume that attention is directed to entire objects. This requires to first map gaze to objects before determining the depth which is closest to the gaze position. This is an approach which is favored in this thesis (Chapter 3 and 5), where we propose an probabilistic approach on object-based attention. This has the advantage that gaze depth can be better disambiguated as there are far less objects than pixels or triangles in a scene. With an "attention probability" for each object, we may also obtain a probability which quantifies the uncertainty of a certain gaze depth. So rather than assuming a single 3D gaze point, we can define a volume of uncertainty. Moreover, for gaze controlled applications, the object-based approach further relates gaze to the stimulus a user is actually seeing.

Binocular Gaze-Depth Estimation: No geometry information is required for *binocular* methods, which estimate the 3D position from an observer's binocular gaze information only. When recording binocular gaze, a 3D gaze point can be obtained by triangulation (e.g., [46, 173]). This method is particularly useful for real world stimuli where no geometric model is available (e.g., [134]). Since the gaze direction measured by an eye tracker for both eyes does not necessarily converge in the plane of zero parallax, Duchovski et al. [47] proposed to perform a depth calibration. They used for this procedure a scene of cubes arranged in different depth layers. During the eye tracking experiment, participants focused subsequently on one after another cube while gaze data was recorded together with the depth of the focused cube, which they denoted as *stimulus depth*. Calibration was done by minimizing the error between gaze and stimulus depth using a model with two parameters: a shift and a scale factor. With the shift, the zero point of gaze depth is calibrated to correspond to the plane of zero parallax. The scale factor is used to align the range of measured gaze depths with the scene's depth range. The

temporal graph of gaze depth recorded with this procedure is shown in Figure 2.14. The results obtained in a monocular control condition, which is shown in the left graph, are almost unaffected by the virtual depth of the attended object. But when the scene is shown in stereo 3D, a considerable variation which correlates well with the stimulus depth (Figure 2.14 right graph) can be observed. Since there is a high amount of noise in binocular gaze data, the results shown in this figure have been pre-processed with strong low-pass filtering.

A holistic approach to determine gaze depth from binocular data was proposed by Pfeiffer et al. [135]. They used a 3D gaze-calibration procedure based on a self organizing map that learns to correctly map an input of two gaze points to a gaze depth. Using the 3D gaze point computed from binocular data, they determined the closest scene objects in the vicinity of the 3D gaze point from the scene geometry. However, their investigation revealed that for a gaze-based object selection task, the use of a 3D gaze point computed from binocular gaze data increases performance only in some critical cases where objects which are located in different depth layers occlude each other. But in other cases, the accuracy was lower compared to methods inferring depth with a geometry-based approach from the 2D gaze point.

Gaze-depth estimation can be useful for applications such as gaze-controlled interaction with 3D environments, and gaze-driven depth-of-field or 3D stereo displays (see Section 2.6). Depending on the application, both approaches described above have their pros and cons. The big advantage of binocular techniques is that they are independent of the stimulus. This means that they can be used in any eye-tracking application without the necessity of having access to a digital representation of the scene geometry. However, binocular methods suffer from the strong noise in binocular gaze signal, which produces a very inaccurate and unstable gaze-depth signal (with current eye tracking hardware). Moreover, binocular methods can determine the gaze depth only after a sufficient number of gaze samples (depending on the low-pass filter size) were output from the eye tracker, while geometry-based approaches may allow to *predict* future gaze depth at saccade landing positions or future gaze locations in smooth pursuits. Future gaze locations could be, for instance, predicted during smooth pursuits from the current gaze movement with a kalman filter, or during saccades with a ballistic model which takes the gaze samples at the onset of an saccade as predictor, as proposed by Kolmogortsev et al. [93].

Probably, at least for applications which render virtual scenes, the combination of both approaches may provide the best results.

Besides the applications proposed above, gaze-depth estimation can be also useful to investigate temporal aspects of stereoscopic vision, as done in the example reviewed below:

Extracting Vergence-Response Times from Gaze Depth Signals: Recently, Templin et al. [174] utilized gaze-depth recordings to estimate the vergence response time after an abrupt change of disparities, such as elicited by video cuts in 3D stereo movies. To this end, they fitted a sigmoid to the gaze depth signal recorded during a discontinuous change from an *initial disparity* to a *target disparity*. Since a gaze depth signal obtained from one person was too noisy, they proposed to use an average gaze depth signal from many persons for the fitting (a strategy we also used in our work presented in Chapter 5). From the mathematical abstraction obtained by fitting the sigmoid function, they were able to analytically deduce the time a user needs to converge/diverge to the target disparity, which they denote as *vergence-response time*.

They determined vergence-response times for a systematically sampled bivariate grid of initial-disparity/target-disparity combinations and fitted a linear model to these results. This model can be used to predict vergence response times from the type of disparity change and could be used to avoid inappropriate response times in cuts with 3D-stereo film footage.

2.6 Attention-Aware Computer-Graphics Applications

This section reviews the state-of-the-art of applications in the field of computer graphics which make use of gaze information or attention models.

On the one hand, attention-aware techniques could be used to accelerate image generation. This means to save computational cycles by rendering unattended image regions less accurately, which we denote as *computational acceleration* (Section 2.6.1).

On the other hand, recent work also utilizes attention to control various effects resulting from the physiology of the eyes. This applies in particular for stereoscopic displays and depth-of-field simulations. Using attention predictors or eye trackers could be a key to make these effects more realistic and pleasant for the users. Since those effects are primarily useful to improve a user's perception, they will be categorized with the term *perceptual optimization* (Section 2.6.2).

Another interesting application is to guide a users attention to important parts in a scene. Therefore, this review will conclude with a paragraph on *attention guidance* in computer-graphics applications (Section 2.6.3).

2.6.1 Attention-Aware Computational Acceleration

Given a limited computational budget, attention models have been utilized to selectively allocate computational resources to parts of a scene a user pays more attention to. This is particularly useful for real-time rendering of high-fidelity graphics or to avoid that graphics processing consumes too much energy on mobile devices.

Attention-based optimization can be performed, on the one hand, at the development stage. Geometry information can be reduced by simplifying the mesh in parts which are attended less often. On the other hand, the optimization can be performed at run-time to make rendering more computationally efficient.

Mesh Simplification

One way to make computer-graphics applications more efficient is to reduce mesh complexity. Using attention-based techniques, the resulting loss of quality can be made less perceivable using an importance-driven mesh simplification tool which prioritizes simplifications for less salient parts in the geometry. To this end, Lee et al. [100] proposed a method which extends the basic algorithm for bottom-up saliency computation to geometric features of meshes. They applied center surround contrast metrics on several spatial scales on curvature features (Figure 2.15). The result is a texture which specifies a saliency score for each texel on the surface of the geometry. The saliency-map-guided geometry simplification is, e.g., useful for the generation of level-of-detail representations. As further utility, they proposed to use mesh saliency for an optimal viewpoint selection. An empirical approach to compute mesh saliency was proposed by

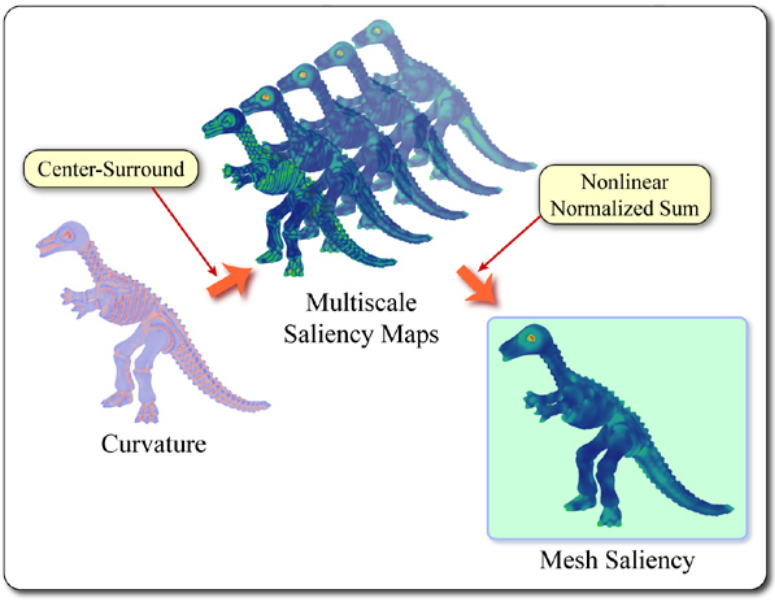


Figure 2.15: Pipeline for the computation of mesh saliency. This image was taken from [100].

Howlett et al. [71], who used eye tracking data to infer mesh saliency. Overall, saliency-based mesh simplification found most utility for the generation level-of-detail simplifications and several improvements and several alternative methods for computing saliency of mesh features have been proposed later (e.g., [113, 150, 154, 192]).

Geometric details can also be reduced in densely cluttered scenes where selective attention lowers sensitivity to geometric details. Ramanarayanan et al. [141] showed that in complex aggregates containing groups of object instances (e.g., a plant species in a vegetation scenes) the proportion of these groups can be manipulated considerably without a user noticing a difference. In a proof-of-concept evaluation, they showed that this perceptual insensitivity can be exploited to reduce the amount of polygons in the scene. This is done by increasing the number of objects with a lower polygon count while lowering the proportion of geometrically more complex instances.

Accelerating Rendering

Using perception models permits to accelerate rendering using an aggressive optimization strategy that trades rendering speed against a loss of quality. Perceptual models are used to make the reduction of image quality less perceivable.

From Visible Difference Predictors to Attention Models: First attempts in perceptual graphics used metrics which predict the perceivable differences between two images. Since the perceived difference can decrease significantly due to masking effects of high spatial frequencies [53], difference predictors are usually a function of image features such as color, luminance, spatial frequency and contrast. Difference predictors, such as the Daly's Visible Difference Pre-

(a) Reference Image (b) Aleph Map

Figure 2.16: Figure **(b)** shows a visualization of the spatio-temporal error tolerance map (Aleph Map) corresponding to the image shown in **(a)**. This image was taken from [196].

dictor [34], can be used in ray tracers to define a stopping criterion when the results converge to a point where further iterations do not provide a perceivable increase of image quality [13, 120]. They have been also applied in rasterizers to reduce level of detail of meshes until reaching a threshold where the simplifications start to become noticeable [38].

In the spirit of perceptual methods using visible difference predictors, attention predicting models have been explored as well to accelerate rendering techniques. A general algorithmic framework to employ attention metrics in order to accelerate realtime rendering was proposed in the seminal work of Funkhouser and Sequin [55]. The core module solves a cost/benefit optimization problem that trades rendering speed (benefit) against rendering errors (cost). Attention models can be introduced by weighting cost/benefit estimates with importance scores. Yee et al. [196] and Haber et al. [59] were the first to apply bottom-up saliency maps [79] to accelerate global-illumination rendering. In Yee et al.'s work [196], the attention model was combined with a visible difference predictor and a metric which predicts sensitivity to details of objects in motion. These metrics were combined into one *Aleph Map*, shown in Figure 2.16. The Aleph Map spatially encodes the predicted tolerance for rendering errors. Where the Aleph score is low, quality should be maximized, while computational resources can be saved by sampling fewer rays in regions where the error tolerance is high (e.g., the reflections in the mirror).

Using Task-Relevance Maps: Cater et al. [24] showed that quality can be optimized more selectively with a predictor that accounts for top-down attention as well. They performed an experiment with a counting task (e.g., look for pencils) in a virtual environment. With this example scenario, they constructed a case where inattentional blindness may occur due to the user narrowing the focus on the task. Thus, high-quality rendering could be applied selectively in the neighborhood of task-relevant items (i.e., targets of visual search) and near important features which were predicted with a saliency map. They called their method "selective rendering", and

(a) task-relevance map (b) task-importance map (c) saliency map (d) importance map

Figure 2.17: Illustration how importance maps for selective rendering can be encoded in image space. These illustrations were taken from [167].

their user study provides a proof-of-concept which confirms that inattentional blindness can be successfully exploited to increase the perceived quality in real-time rendering applications. Inspired by Yee et al.'s work [196], Cater et al. [25] advanced this idea in subsequent work to a more sophisticated selective-rendering framework that also made use of visible difference predictors.

Another example for using task relevance as top-down attention predictors for selective quality rendering can be found in the work of Sundstedt et al. [166, 167]. They proposed a task-importance map which combines (bottom-up) saliency with a top-down attention prediction component. Task-relevant objects were manually selected by the application developer, and at run-time, this information was encoded in image space as a binary mask defining pixels covered by task-relevant objects (Figure 2.17(a)). To account for the spatial extent of foveal vision, this binary mask was low-pass filtered with a kernel corresponding to the size of the fovea in order to obtain a task-importance map (Figure 2.17(b)). Moreover, a salience map (Figure 2.17(c)) was created to also predict bottom-up attention, which was then merged with the task-importance map to derive the master importance map (Figure 2.17(d)) used for selective rendering.

While previous selective rendering methods used high-quality rendering in a circular region around a task-relevant object and thus conservatively assumed that a user attends to everything that could be seen with foveal vision, Sundstedt et al. [169] showed that due to inattentional blindness users may even fail to notice rendering errors within foveal vision when they appear on task irrelevant objects.

Exploiting Change Blindness: Besides task relevance, also the phenomenon change blindness has been exploited for selective rendering. Cater et al. [26] performed an experiment where rendering quality was reduced during a disruption of the continuous image stream through a flicker image. This interruption provoked change blindness, and users were not able to notice the changes caused by quality reductions. To justify the applicability of the method without artificially interleaving the visual signal with flicker images, they argued that a visual disruption by flicker images mimics the effects of eye blinks and saccades. A potential application would use a low-latency and high frame-rate eye tracker to detect saccades in realtime. This approach could be particularly useful to mask popping artifacts, which frequently occur when the ren-

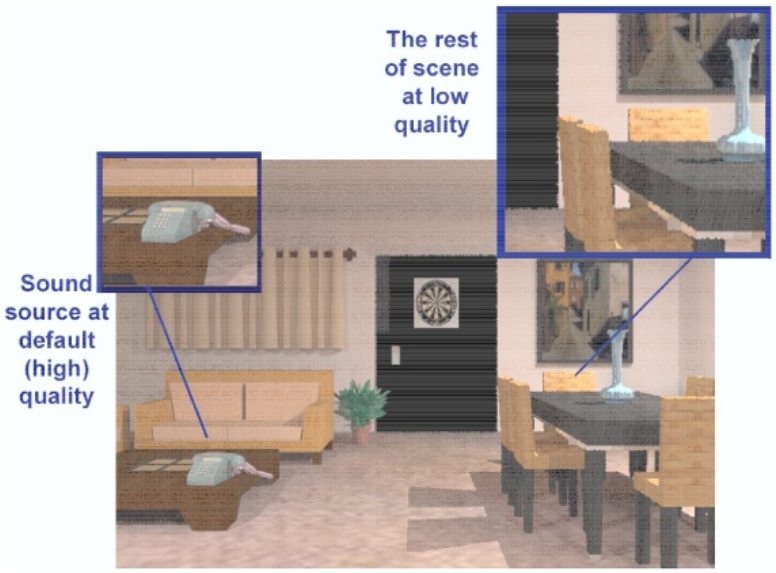


Figure 2.18: Selective rendering of an object (telephone) which attracts attention crossmodally due to semantically congruent sound events (onset of ringing tone). This image was taken from [112].

dering system has to switch between different level-of-detail representations (LOD) because the camera is moving towards (requiring switching from high to low LODs), or away from scene objects (requiring switching from low to high LODs).

Predicting Attention from Schematic Congruency: Another alternative has been proposed Zotos et al [201]. They predicted attention based on whether an object is schematically congruent (e.g., pencils on a desk top) or incongruent with the scene context (e.g., scull on an office desk). Since surprising or unexpected geometries tend to attract more attention, they can be selectively rendered in higher LOD than objects which are expected to appear within the scene context.

Using Crossmodal Attention Predictors: Apart from visual saliency, also sound events may attract visual attention. Using an audio-visual importance map, Mastoropoulou et al. [112] demonstrated that users tend to focus on objects which are semantically congruent with an accompanying sound event (e.g., a ringing telephone). In this condition attention is attracted *crossmodally* and can be selectively rendered in high quality, while a user will not notice significant quality reductions in other parts of the scene (Figure 2.18). Later, Brkic et al. [20] showed that crossmodal selective rendering even works for a combination of visual and olfactory stimuli. However, since crossmodal attraction of attention is a stimulus-driven effect, it occurs only for a short time window. Unfortunately, this makes selective rendering only applicable in a few cases when crossmodal events occur.

Miscellaneous Examples: Further interesting example applications where attention-based optimization has been applied include the acceleration of physically based rendering of participating media [58, 118], efficient graphics on mobile devices [9] and compression of videos [103, 111].

Optimizing Animations

Attention predictors or eye trackers can also be employed to reduce other costly computations, such as physical simulations or animations of huge crowds of characters. For instance, O'Sullivan et al. [126] showed that collisions can be simulated less accurately where a user is not attending to.

McDonnel et al. [114] used an eye tracker to study which parts of virtual human characters are mostly attended. They found that most attention is paid to body textures and faces and used this knowledge to disguise cloned characters and animations in huge crowds.

While there are only a few exemplary studies attempting to utilize attention to accelerate animations and physical simulations, this could be a promising avenue for future research. Besides complex animations of virtual agents (e.g., facial animation), the increasing use of physically based simulations to render effects such as floating fluids or physical interaction of a large number of objects (e.g., a stack of cans being crashed by a collision) are costly to simulate in realtime and should greatly benefit from attention-based optimization.

Gaze-Contingent Rendering

With real-time eye tracking, the focus of attention can be determined much more reliably compared to computational models. This allows rendering only one small region around the current gaze point that is resolved by foveal vision in full resolution or high quality. This approach is commonly denoted as *gaze-contingent* or *foveated* rendering and dates back to 1987 to the work of Kocian et al. [92]. Image-space techniques adjust the image resolution or the sampling rate in a ray tracer according to a model which predicts visual acuity as a function of eccentricity (e.g., [102]), while object-space approaches (e.g., [124]) use an analogous strategy for LOD management.

As improvement to simple eccentricity-based approaches, Luebke et al. [106] proposed to use a more sophisticated model for the space-dependent sensitivity of the retina that also takes contrast and spatial frequency into account. Overall, a considerable body of work has been published on gaze-contingent rendering, reviewed, for instance, by Duchovski et al. [43,44]. Among those, there have also been attempts to bring forward the gaze-contingent rendering approach to stereo displays [27]. This can be, for instance, useful to overcome bandwidth limitations in the transmission of stereoscopic images. More recently, gaze-contingent rendering has been revisited by Guenter et al. [57], who proposed an efficient implementation yielding impressive speed-ups (they reported a factor of 5-6) without users noticing any changes in rendering quality. This work demonstrated that foveated displays remain a promising solution to accelerate current and future computer-graphics. This, taken together with the fact that eye trackers are on the verge of becoming a commodity, shows that gaze-contingent rendering could provide an impor-

tant utility for energy-efficient rendering on mobile devices or to realize high-fidelity graphics on immersive high-definition displays.

2.6.2 Attention-Aware Perceptual Optimization

Nowadays, in the age of high-performance GPUs, frame rates and level-of-detail are not always the most pressing challenge, and it is also useful to focus more on optimizing a user's perception of visual information. Thus, other attention-adaptive graphics applications have been proposed that focus on a perceptual optimization, such as an appropriate depth-of-field configuration or an attention-aware optimization of disparities in stereoscopic applications.

Attention-Aware Depth-of-Field Simulation

Simulations of depth-of-field (DoF) effects can increase depth perception since they produce an effective monocular depth cue [62]. The most prominent application is to use DoF to direct the viewer's attention. While artificial DoF is certainly useful to direct a viewer's attention in passive viewing conditions, its application as additional depth cue and asset to increase visual fidelity in interactive applications requires to predict the user's focus to avoid annoyance. A first attempt to make DoF applicable for interactive use was proposed with Hillaire et al.'s work where the DoF plane was aligned with the actual or predicted focus of a user. They proposed to determine the depth of the currently attended object by real-time eye tracking [65] or attention-prediction heuristics [64]. As simple heuristic to predict attention, they proposed to use the proximity to the center of the screen and an object's task relevance. In addition to DoF simulations, Hillaire et al. [65] proposed a method which controls the movement of the camera according to the gaze behavior of the viewer. They reported results from a user study where perceived realism, fun, depth perception and the feeling of immersion were ranked significantly higher when gaze-driven DoF and camera control were applied [65]. However, in highly dynamic scenes (e.g., spaceship games) the estimation of the current depth can be highly ambiguous and result in a high temporal error rate, i.e., an unstable frequent switching between unattended and attended depth planes. Thus, Mantiuk et al. [109] recently improved gaze-controlled DoF simulations with a fixation detection method which combines gaze and object-space information to increase temporal coherence and spatial accuracy in the determination of the user's focus. Using spatial proximity and motion similarity, they used a probabilistic approach to map gaze to object features. To combine probabilities over subsequent frames, they employed a Hidden Markov Chain (HMC). The HMC was configured such that a shift of attention from one to another feature is assumed to occur less probable (5%) than continued attention (95%). With this approach, fixation filtering and gaze-depth determination was done in one step. Thanks to using object-space information, they could increase spatial accuracy beyond the limitations of the eye tracker, and due the use of Hidden Markov Chains, they could significantly reduce the number of (erroneous) switches between different depths.

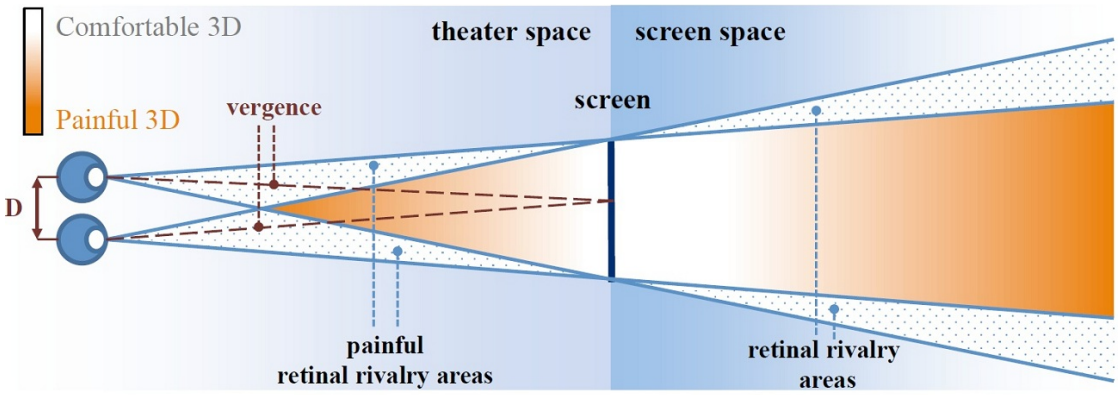


Figure 2.19: An illustration of the stereoscopic comfort zone. This image was taken from [98].

Increasing Comfort in Stereoscopic Displays

Current stereoscopic technology may be tiring for users, and is often reported as uncomfortable after even short times of use. One main reason for this is discomfort caused by the *vergence/accomodation* conflict. Comfort is usually predicted as a function of disparity, which is visualized as comfort zone in Figure 2.19, or as a bivariate function of vergence and focal distance [66, 90]. Using a comfort model, disparities can be manipulated by mapping them to the range of disparities which can be seen comfortably. However, the unwanted side effect of manipulating disparities can be a significant reduction of depth quality, for example in the form of cardboard effects. One perceptual strategy proposed by Didyk et al. is to map disparities in a way that the loss of depth quality is less noticeable. To this end, they developed a stereo quality model which is a function of disparity frequency [35] and luminance contrast [36]. This function is then used to predict the just noticeable difference in disparity manipulations in order to minimize the perceived loss of depth quality while maintaining an acceptable level of stereo viewing comfort.

Using Attention Predictors: A complementary approach to the use of disparity sensitivity functions is to apply attention predictors or eye trackers. Lang et al. [98] proposed an automated non-linear disparity remapping technique for movies. With this approach, disparity in visually unimportant regions is compressed, while correct disparity gradients are maintained in visually important regions (Figure 2.20). By using saliency maps to score the visual importance of scene features, visual artifacts of disparity range compression, such as flattening, are moved into visually unimportant regions where they are less likely to be recognized. To compute attention maps, they used simple low-level features, such as depth and edges, but propose that also top-down factors could be integrated for further improvements. A similar approach, which employs attention predicting heuristics, was proposed by Celikcan et al. [28] for interactive virtual environments.

Using Eye Tracking: With real-time eye tracking, stereo parameters can be adjusted more selectively. For instance, the plane of zero parallax can be shifted to coincide with the depth

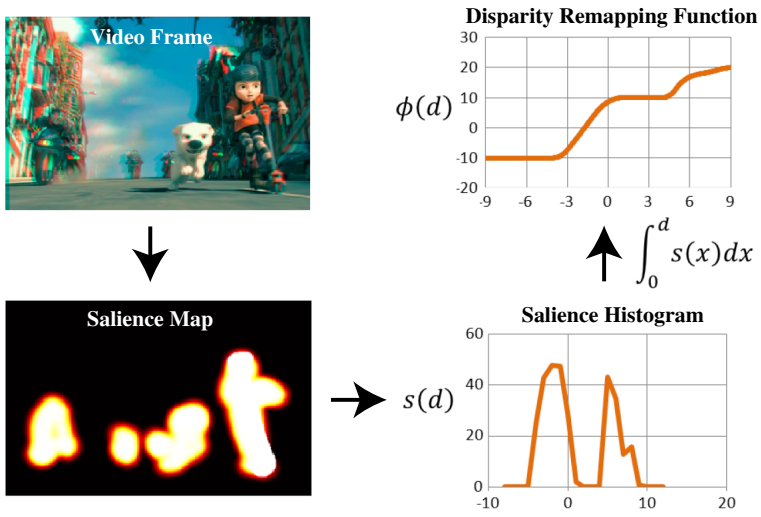


Figure 2.20: The method proposed by Lang et al. [98] to obtain a disparity remapping function $\phi(d)$ mapping a disparity d to a new disparity d': First, a saliency map is computed from the current video frame, then the total saliency $s(d)$ (y-axis) for each disparity value (x-axis) is computed, from which the disparity remapping function ϕ is derived by building an integral over $s(d)$ and normalizing. This strategy has the effect that the disparity gradient is low in disparity ranges with low saliency. This image was adapted from [98].

of the currently fixated object. This was proposed by Fisker et al. [54], who determined the focus depth from binocular gaze. In our work, which will be presented in Chapter 5, we used a geometry-based approach to determine gaze depth (e.g., using a depth buffer) and performed a formal evaluation of this strategy, where we evaluated how this affects subjectively experienced comfort and the time users need to fuse stereo image pairs, which is a good objective predictor for comfort. Besides disparity adjustment, comfort can also be increased by low-pass filtering of high-frequency content, as proposed by Perrin et al. [131], who introduced a comfort model that is a function of disparity and spatial frequency in the images. By using an eye tracker in these applications, it can be avoided that the a visual quality reduction caused by blurring is noticed by the user and instead is experienced as DoF effect [11, 12, 45, 101, 171].

Determining Optimal Cuts in Stereoscopic Movies: Visual discomfort can also occur during cuts in stereo-scopic movies, which require a user to adjust eye vergence to a new disparity. To optimize comfort, the vergence response time after a cut should be minimized. Thus, Templin et al. [174] recommended to use a model which predicts vergence response time in an abrupt transition from one disparity to the other disparity as a function of this disparity pair. To obtain this model, they proposed an eye tracking-based method, which has been already described in Section 2.7.3.

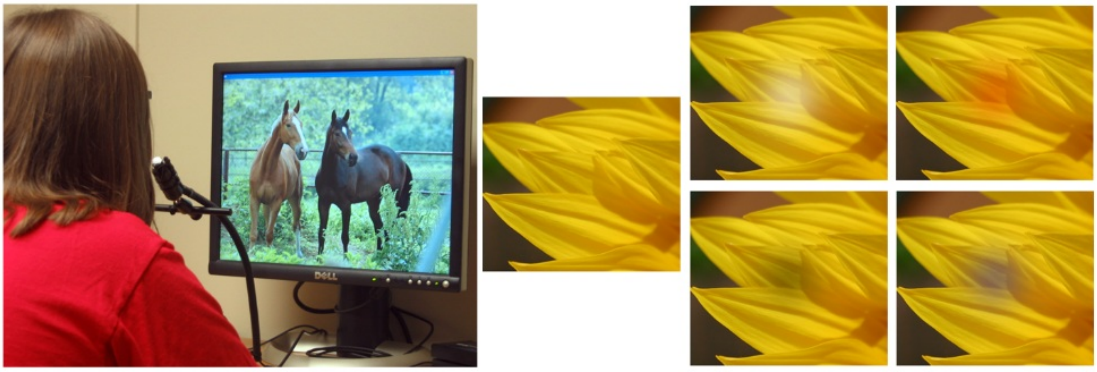


Figure 2.21: Subtle gaze direction in natural images (left). Attention is attracted to the blossoms of a flower shown in the small patch of the stimulus image (center). This can be achieved by luminance modulations (third column) or warm-cool modulations (right column). This illustration was taken from [3].

High Dynamic Range Tone Mapping

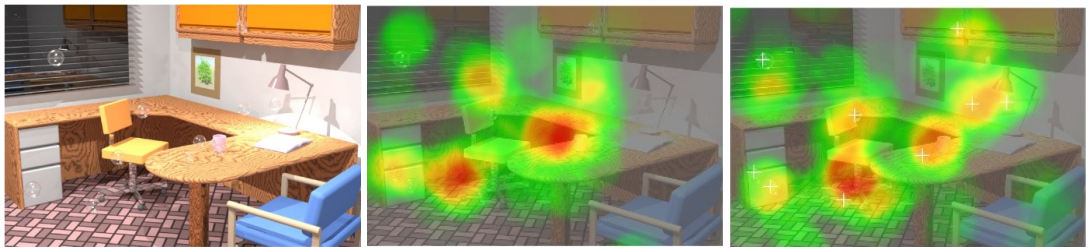
For large displays or immersive virtual environments, also high dynamic range (HDR) tone mapping can benefit from gaze-adaptive methods. HDR tone mapping is required when HDR images are shown on a low-dynamic range display. Tone-mapping techniques should ideally preserve the perceived contrast, even though the luminance range in the tone-mapped images shown on the display is considerably compressed. Since the eyes adapt most sensitively to luminance within foveal vision, an ideal tone-mapping operator should temporally adapt to the luminance in the image region near a user's point of regard. Thus, it is useful to use an eye tracker to determine which parts of an image are seen with foveal vision and adjust tone mapping accordingly. For instance, when a user is looking into the sun, the tone-mapping function should maximize contrast in the range of bright colors, as opposed to when a user is gazing at an object in a shadow, where the mapping should at best preserve contrast within a narrowed range of dark colors. First examples for gaze-adaptive tone mapping were proposed by Rahardja et al. [140], Cheng et al. [30] and Mantiuk et al. [110].

2.6.3 Attention Guidance in Virtual Environments

Another interesting application is to direct a user's attention to those parts in an image which are most important. This can be done, for instance, to facilitate search for task-relevant objects or to notify a user about important elements in a scene (e.g., maps or signs).

Guidance with Illumination

One way to guide visual attention is to use lighting. In the simplest case, this can be achieved by selectively illuminating content with a spotlight, while the rest of the displayed scene is shown in darker colors [88]. A more subtle way to direct attention by lighting design was proposed



(a) Scene with soap bubbles (b) Heat map without SGD (c) Heat map with SGD

Figure 2.22: Heat maps of gaze data recorded during a visual search task. The task was to count the number of soap bubbles in the scene **(a)**, which were located on the positions labeled with white crosses in heat map shown in **(c)**. The heat map in **(b)** shows the gaze distribution in the control condition, i.e., under normal viewing conditions, and the heat map in **(c)** illustrates the distribution directed gaze. This illustration was taken from [115].

by El-Nasr et al. [50]. They took inspiration from techniques which are employed in movies or on theater stages to guide attention of the audience. They implemented a system guiding a user's attention in virtual environments by automatically adjusting lighting conditions, including angles, positions, intensities, and colors of the light sources.

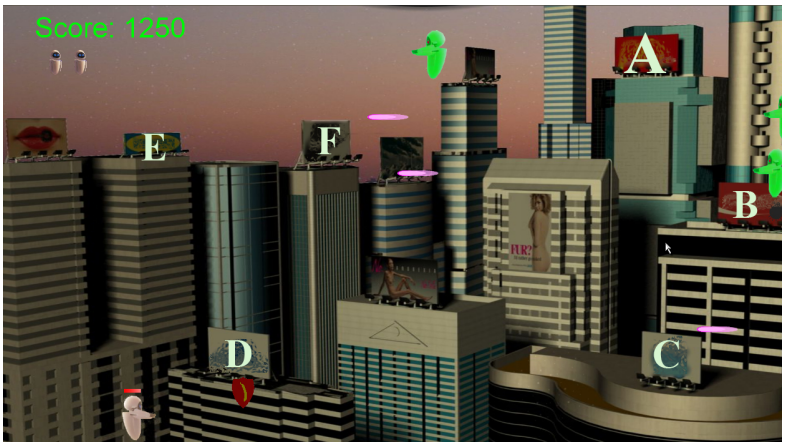
Guidance with Temporal Features

Attention is strongly attracted by temporal changes in the field of view. Thus, Bailey et al. [3] proposed to attract a user's gaze with short luminance or warm-cool modulations in image regions a user sees peripherally (Figure 2.21). The eyes are particularly sensitive to temporal changes in the periphery of the fovea, and such modulations strongly attract gaze towards the location of change. To know what a user is currently seeing with peripheral vision and to avoid conscious perception of the modulations, an eye tracker notifies the system when a saccade to the modulated regions is triggered, upon which the modulations are stopped. With this strategy, a user can barely notice the modulations attracting eye movements, and thus the method is denoted as *subtle gaze direction*.

In subsequent work, McNamara et al. [115, 116] showed that performance in a counting task with objects which are difficult to find (e.g., soap bubbles) can be significantly improved (Figure 2.22) using subtle gaze direction, even in the presence of distractor modulations. Further applications of the subtle gaze-direction method were proposed later, including narrative art for attention guidance during visual story telling [117], training of novice radiologists [155], and attention guidance in real-world environments [14].

Recently, Waldner et al. (including the author of this thesis) [183] proposed to use an intensity modulation (i.e., flicker) to guide attention in narrative visualizations of molecular reaction chains. In contrast to subtle gaze direction, we avoided that an eye tracker is required which stops the flicker modulation as soon attention is successfully attracted. Instead, we attempted to configure the flickering in a way that it minimizes the annoyance factor of this technique. To this end, we proposed a two-staged design comprising an orientation stage, which attracts atten-

(a) Game elements



(b) Action game scene (TEST condition)

Figure 2.23: The elements and a screen-shot of the game we used in our preliminary study on directing attention by visual design to advertisements billboards in the background.

tion to peripheral events with a fast modulation with a high amplitude, and an engagement stage, which guides attention (after orientation) in foveal vision with a more visually comfortable modulation. The optimal configuration of the modulation parameters was determined with a series of experiments investigating the effects of modulation amplitude and frequency on detection times and subjectively experienced annoyance.

Guidance by Feature Linking

Another way to guide attention is to increase the salience of objects to be attended by visual design. A preliminary study how this could be realized was carried by the author of this thesis [8] (in a publication which is not part of this thesis). The general challenge addressed by this work was to investigate how effectively the theory of guided search can be applied to improve the perceptual visibility of elements which are not directly relevant to the task of a video game (e.g., billboards in the background). To this end, we investigated how well users are able to remember advertising billboards located in the background of a computer game if their features are "linked" to features of task-relevant game elements (Figure 2.23(a)). We decided to use advertisements of real brands, since those have logos which are a result of a sophisticated design process. They are optimized to be aesthetically pleasant, well recognized and, most importantly, to be *remembered easily*.

To evaluate this strategy, we performed a simple study with three conditions, which were assigned to three groups of participants. In the test condition, we used a health item (health item 1, Figure 2.23(a)) which shared color and orientation with a target brand advertised in the background scene of the game (billboard A in Figure 2.23(b)). The results of the test condition were compared to two control conditions: in the first control condition (CTRL 1), we used the same health item as in the test condition, but there was no advertisement in the background sharing features with this game item (i.e., billboard A was replaced by another advertising billboard A'). In the second control condition (CTRL 2), we used the same background as in the test condition, but the health item was replaced by another health item with different features (i.e., health item 2 in Figure 2.23(a)).

Attention was measured with a subsequent memory test after a participant played the game in one of the three conditions depending on the group he or she was randomly assigned to. Our result was that recall of the target advertisement (A) reached over 90%, whereas other advertisements in the same condition, or the targeted advertisement in both control conditions (CTRL1 and CTRL2), were remembered drastically less often (0-42%). Another interesting observation was that participants of the test group also recalled other advertisements (billboards B,C,. . . ,F in Figure 2.23(b)) more often than participants of the control groups. Though our trick significantly manipulated a users attention, we did not observe any negative effects on gaming performance, indicating that game fun is not affected by this little but effective change in game design. This work could be a starting point for the exploration of new avenues for the "reverse" application of visual-attention models. For example, the proposed guiding principle should also work to avoid inattentional blindness to other targets than advertisements. The results may inspire game designers to direct attention in situations where a user does not know which object is task relevant.

2.7 Conclusion

Visual attention is a challenging research topic and the key to understanding the principles of vision. Knowledge and theory about attention can be of tremendous importance for various fields of visual computing, most notably computer vision and graphics. This chapter reviewed many interesting examples of computer-graphics applications where visual attention and gaze tracking may find a great utility. While computer science and IT applications can greatly benefit from research done on attention in psychology and neuroscience, the knowledge transfer can also go in the opposite direction. Particularly computer graphics provide a set of valuable tools which may assist research on visual attention in many other disciplines. The most important benefits of tools that state-of-the-art computer graphics can contribute to assist experimental work on visual attention can be summarized as follows:

Control of visually complex stimuli: With state-of-the-art computer-graphics, researchers can create visually complex virtual environments for their experiments. The main advantage is that artificial stimuli can be controlled with the configuration of the rendering methods used for displaying stimuli. For instance, the role of several lighting effects such as shading, shadows or ambient occlusion, can be studied in isolation.

Automation: Using synthetic stimuli to study visual attention allows automating the analysis process. In virtual-environment applications, for instance, we can log object information and every change of the scene in synchrony with eye movements and user interaction events. Having this data permits a convenient and detailed analysis of gaze behavior (and potentially other behaviors) with respect to the properties of the stimuli. For instance, it avoids tedious efforts which are necessary to annotate objects or define regions of interest in eye tracking experiments using animated or interactive stimuli.

Validation of Models and Hypotheses: The work described in Section 2.6 provides many interesting examples how knowledge about visual attention can be utilized for computer-graphics applications. Though this work was oriented towards utility, the experiments also add further pieces of evidence to the theories the applications relied on.

CHAPTER 3

Attention Inference: Gaze-to-Object Mapping

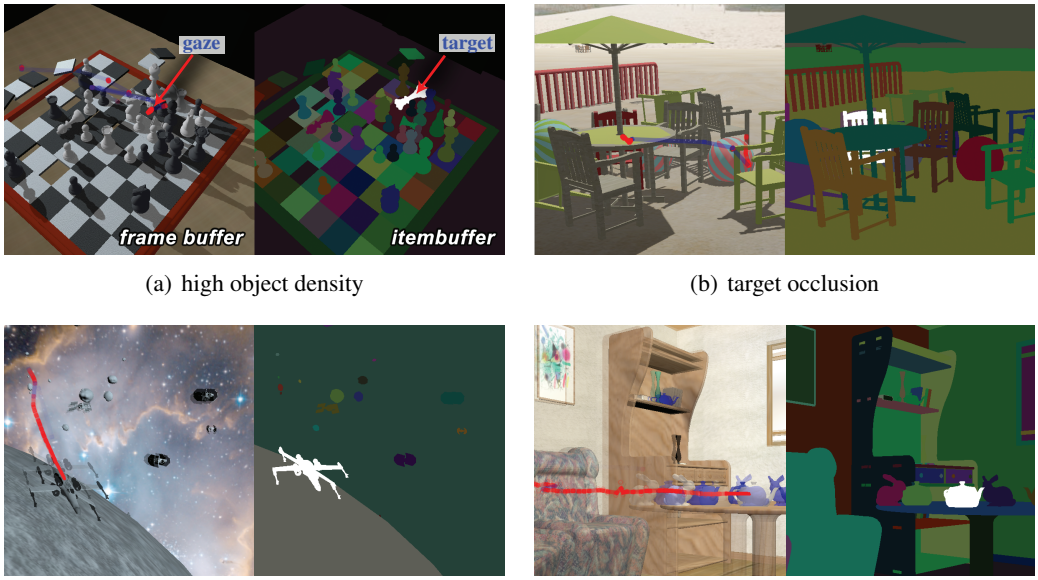
> Whenever you set out to do something, something else has to be done first
>
> Murphy's Laws

In this chapter we tackle the first challenge, which is inferring the object of attention, given the current gaze position observed with an eye-tracker. We denote this process as *gaze-to-object mapping* and devise a methodology to evaluate different gaze-to-object mapping techniques. Addressing this challenge is the basis required to make gaze analysis (Chapter V) and object-based attention-aware rendering (Chapter IV) more reliable.

The work described in this chapter has been published under the title "Gaze-To-Object Mappipng During Visual Search in 3D Virtual Environments" [6] in the SAP 2014 Special Issue of ACM Transactions on Applied Perception.

3.1 Introduction

Gaze tracking has become one of the most important tools in the study of human behavior and interaction with graphical software. Applications with 2D graphical user interfaces (GUIs) lend themselves to well-established gaze-analysis techniques, such as heat-maps or gaze-path visualizations, since the stimulus remains mostly static [42]. Recent techniques, for example based on dynamic areas of interest [7, 127], have enabled analyzing gaze behavior and visual attention in 3D graphical applications. These techniques leverage the rich information contained in the scene graph of modern video games and virtual environments in order to identify the object(s) of attention, rather than relying on recorded image sequences.



(a) high object density

(b) target occlusion

(c) object motion

(d) camera motion

Figure 3.1: Scenes exhibiting the main challenges of gaze-to-object mapping. For each scene the frame buffer (left) and the item buffer (right) are shown. In the frame buffer the red points indicate fixations, while in the item buffer the attention target is highlighted in white. Figure (a) is a scene with a high object density, while the target in (b) is occluded. In (c) objects are animated and in (d) the camera is in motion.

The mechanisms behind the deployment of gaze over visual stimuli is still an open research topic in visual perception, and a significant body of research suggests that attention arises from distributed interactions within and among different types of perceptual representations (e.g., space-based, feature-based, and object-based) [95]. In this chapter, we focus on object-based attention [29,48,72] since 3D objects are those scene units that are typically sought for manipulation, interaction and representation in 3D virtual environments. Therefore, identifying objects that are (potential) attentional targets is important for a multitude of applications.

Using an eye tracker to record gaze while viewing visual stimuli generated by a 3D application (e.g., a game) is becoming a widely popular method of studying visual perception, including attention. The process of identifying the scene object that is the target of attention is called gaze-to-object mapping (GTOM). The primary challenge these algorithms face is to accurately provide information on *what* (i.e., scene object) the user is attending to, which oftentimes does not coincide with *where* the user's gaze is deployed (i.e., positions of recorded gaze). Inferring what a user is attending to from where a user is looking at is of high importance and utility, since it forms the foundation of modeling visual attention, and potentially other cognitive processes. A small number of GTOM methods geared towards dynamic 3D scenes have already been devised [168], however these methods are still in their infancy and they have not been for-

mally evaluated. Therefore, their validity and accuracy remain uncharted territory, which has motivated the more thorough evaluation presented in this chapter.

There are various challenging cases GTOM techniques have to deal with that are common in 3D, some of which are shown in Figure 3.1. For instance, due to eye-tracking inaccuracies and rapid involuntary eye movements, gaze may be deployed over multiple objects at different depths when object density is high (see Figure 3.1(a)). Furthermore, attentional targets can become highly occluded due to dynamic scene changes (see Figure 3.1(b)), and it is very frequent that scene objects or the camera are in motion, as shown in Figures 3.1(c) and 3.1(d), respectively. These cases render commonly used GTOM techniques, such as areas of interest, difficult or impractical to use in dynamic 3D scenarios.

In this work, we adopt and further assess an established methodology for gaze analysis which has been initially devised by Sundstedt et al. [170] and later proposed as a gaze-analysis methodology for computer games [168]. This methodology assumes that the input of any GTOM method is gaze data and an item buffer [187], which is an object segmentation image in which the color of each pixel encodes the unique identity of the respective object rendered to the pixel fragment. For stimuli that are computer generated, such an image can be obtained from an object-based scene graph which is commonly used by content designers to semantically structure scene geometry. Assuming a well-structured content, the item buffer provides a useful hypothesis about the perceptual organization of the scene, i.e., it specifies a set of boolean data structures which map an object identification number to a set of coherent locations which a user might perceptually group and perceive as an object. We further the understanding of the GTOM problem by proposing, formalizing and comparing several approaches. Since it turned out that GTOM is a hard problem, we decided to start with simple, intuitive and reasonable techniques that fit to our framework, which assumes that all stimulus information is represented in a sequence of item-buffer images that is passed to a tool box for gaze analysis. We have then evaluated their accuracy on a set of 30 test scenes covering a broad range of difficulty levels. We consider the desired output of a GTOM method as a predictive probability $P(\mathbf{O}|\mathbf{g})$ for each discrete object $\mathbf{O}$ to be attended by the user, given the gaze observation $\mathbf{g}$. To derive such a probability, we propose to implement GTOM methods using Bayesian inference. As to be proposed in Chapter 4, results of the GTOM output can then be used to derive object space-fixation statistics, which can be further analyzed according to the semantics of the respective objects. The goal is to evaluate different variants of GTOM approaches against a large set of challenging cases. For this, we introduce an experimental methodology to obtain a ground-truth data set suitable for the evaluation of GTOM methods. This is not trivial, since eye-tracking results are often considered ground truth themselves when a user's attention target has to be determined.

We believe that understanding and modeling accurate GTOM in dynamic 3D virtual environments is important because GTOM can be utilized as an implicit means of forming and verifying novel hypotheses of human visual attentional mechanisms. We provide in this work the foundation toward tackling the GTOM problem by making the following contributions:

- an introduction and assessment of several GTOM approaches for dynamic 3D stimuli, formalized via a Bayesian approach.
- an experimental methodology, which uses a set of 30 different 3D scenes, to perform an

objective evaluation of GTOM methods in visually challenging cases.

3.2 Related Work

GTOM is fundamental to eye-tracking software applications. In such applications, recorded eye gaze is first processed to identify fixations, which provide on-screen locations where gaze has been deployed by the user. Fixations are subsequently collected for elements in the scene within one frame, but may also be accumulated over time, and correlated to elements in the stimuli. When studying user behavior in interactive applications, especially video games, the visual signal each user perceives emerges from the pattern of his interactions and therefore differs among users. Until recently, the primary means of studying gaze behavior in VEs involved capturing all images displayed to the user while simultaneously recording his gaze [133]. Mapping eye gaze to stimuli requires gaze and stimuli to be captured simultaneously and subsequently processed to obtain representations suitable for GTOM algorithms. A comparison of nine GTOM methods designed primarily for gaze-controlled applications is presented in [182]. Two different methods implementing dynamic areas of interest are presented in [127] and the publications Chapter 4 is based on, while 3D attentional maps [158] and 3D attention volumes [134] perform GTOM directly in 3D.

The most straightforward way is to cast a ray into the scene and determine the closest scene object intersected by this ray, a general-purpose computer graphics technique known as *Ray Casting*. An early example using this approach to determine attended objects in a three-dimensional scene can be found in Starker and Bolt's work [156], where a ray shot into the scene was intersected with the scene objects' bounding spheres. Coupled with an item buffer, this strategy can be implemented by sampling the object id from the pixel fragment corresponding to the current gaze position. This is computationally inexpensive, particularly in 3D applications where geometry information resides on the GPU, but its major drawback is that it determines an object as the attention target with a binary test (i.e., is intersected or not). The problem stems from the fact that gaze locations sensed by eye trackers have limited accuracy and, more importantly, users do not necessarily center gaze directly on pixels belonging to the attended object. Therefore, ray casting is not necessarily a reliable method to infer the attended object. To mitigate the problem of determining objects at a single position, Sundstedt et al. [170] proposed to sample the neighborhood of the gaze position in an item buffer using a Gaussian splat corresponding to the size of the fovea as a weighting function. This strategy yields an importance value which can be used as a probability prediction. An alternative to rendering an item buffer, proposed recently by Mantiuk et al. [109], is to sparsely attach target points to object surfaces and use those to compute the distance between gaze and these target points. They further proposed to utilize motion information to increase accuracy, since there is a certain amount of correlation between gaze movement trajectories and the motion of an attended target point, which can be transformed to a probability estimate. Mapping gaze to target points placed on scene geometry enabled them to determine the depth of the features currently attended in order to control the depth-of-field according to the attention of the user. Moreover, they processed this probability over time with a Hidden Markov Chain, which served as a fixation filter and guaranteed temporal coherence in the selection of the current depth-of-field distance.

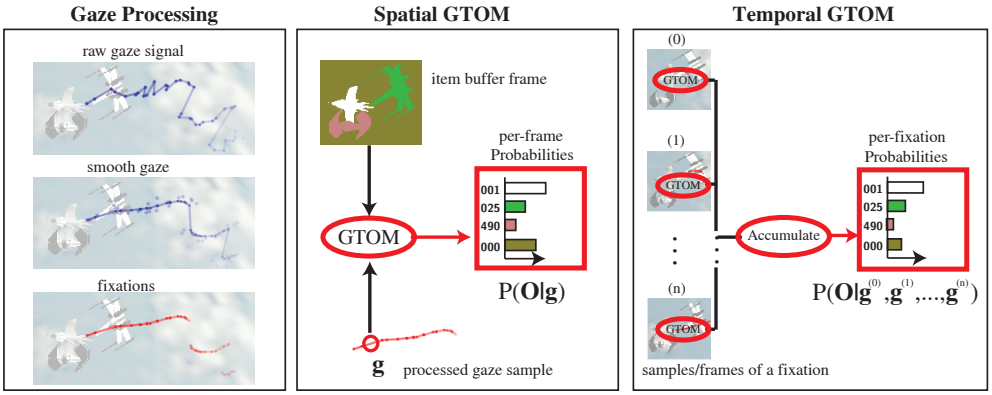


Figure 3.2: Basic steps required to infer object attention from gaze data for each fixation.

3.3 Pipeline

Mapping gaze to objects in 3D scenes can be considered as the process of computing the spatio-temporal correlation of gaze information to objects in a 3D scene. The process can be broken down into the following three steps which are illustrated in Figure 3.2:

- **Gaze and Stimulus Processing**: Raw gaze data are processed to obtain a smooth gaze signal, which is filtered to detect fixations.
- **Spatial GTOM**: Fixations are used to compute an object-space probability density distribution for each frame in the stimuli.
- **Temporal GTOM**: A probability over many frames (e.g., a time window or an entire fixation) is computed using an accumulation strategy of spatial GTOM probabilities.

3.3.1 Gaze and Stimulus Processing

GTOM methods take the following temporal signals as input: a smooth gaze position $g(t)$, a fixation signal $f(t)$, which is a sequence of bits denoting whether a user fixates at time t, and a stimulus signal $s(t)$, which contains information about the objects in the scene. To obtain these signals, we perform the following three processing steps:

Gaze Filtering (provides $g(t)$)

Our binocular eye tracker observes a discrete two-dimensional gaze signal for the left eye $g_L[n]$ and right eye $g_R[n]$ (with $n = \lfloor t/\Delta t \rfloor$). Both are sampled in time intervals of length Δt from a continuous signal that is the sum of the smooth trajectory of natural gaze $G(t)$, a high-frequency oculumotoric noise signal $T(t)$ caused by tremor and micro-saccadic movements, and the random error $\epsilon(t)$ of the eye-tracker:

$$g_{L/R}[n] = G_{L/R}(n\Delta t) + T(n\Delta t) + \epsilon(n\Delta t) \tag{3.1}$$

Instead of treating the two eyes independently, we calculate the target's position as the average of the two eyes, which provides a single gaze signal $\mathbf{g}[n]$. In case of monocular eye-blinks, we consider only the signal of the non-blinking eye.

To suppress the noise ($T(t) + \epsilon(t)$) in the gaze signal, we use a bilateral low-pass filter [176] on the discretized gaze signal. This filter computes a weighted average of the spatially and temporally k-nearest gaze samples:

$$\hat{g}[n] = \sum_{i=n-k}^{n+k} K_t((n-i)\Delta t) K_s(\|g[i] - g[n]\|) g[i] \tag{3.2}$$

We use a Gaussian filter kernel K_t as temporal weighting function and a kernel K_s as spatial weighting function. We use a standard deviation of $\sigma_t = 0.5sec$ for the temporal kernel and $\sigma_s = 0.7°$ for the spatial kernel, respectively. The advantage of bilateral filtering is that it reduces over-smoothing over the abrupt transitions between fixations and saccadic gaze movements. The noise-suppressed signal is used for two purposes: first, to increase the robustness of subsequent saccade and fixation identification, and second to estimate the fixation position, which we assume to be a function of time.

To obtain the position of gaze at a time t, we perform a linear interpolation on the discrete filtered gaze signal $\hat{g}$:

$$g(t) \quad = \quad \hat{g}[\lfloor t/\Delta t \rfloor] + \frac{(t - n\Delta t)(\hat{g}[\lfloor t/\Delta t \rfloor + 1] - \hat{g}[\lfloor t/\Delta t \rfloor])}{\Delta t} \tag{3.3}$$

Identifying Fixation Times (provides $f(t)$)

Based on the assumption that attention correlates with fixation locations only [42], we identify the fixation state (i.e., fixating or not) to partition gaze data into different fixations. We found that dispersion filters often used for fixation identification are less appropriate, as dispersion increases proportionally to drift velocity, and therefore, single moving fixations, such as those occurring when tracking a moving object, are identified as a series of shorter fixations. Instead we utilized a velocity-based fixation detection method that avoids this problem. We found an acceleration threshold-based saccade detector, as proposed by Tole and Young [175], to be most appropriate since it performs best in identifying smooth pursuits, which frequently occur in dynamic stimuli. This detector identifies saccades by their ballistic properties, which have a strong acceleration ($> 40000°/sec^2$) at the onset and a deceleration of similar magnitude at the end of a saccade. Fixations are then determined as groups of non-saccadic consecutive gaze points with a minimum duration (e.g., 3 gaze samples).

Stimulus Processing (provides $s(t)$)

We represent the stimulus of each frame t with a segmentation image that can be stored on a GPU texture. To this end, we render a so-called item buffer as proposed by Sundstedt et al. [170]. The item buffer is rendered in an additional GPU pass to an image where object colors are replaced by unique object ids. For transparent objects, we propose to use an opacity threshold to render them as opaque single objects into the item buffer. Some example item-buffer images and corresponding renderings from the frame buffer can be seen in Figure 3.1).

3.3.2 Gaze-to-Object Mapping (GTOM)

After preprocessing gaze and stimulus data, we map gaze to objects for each frame during a fixation. The desired output of a GTOM strategy is an object-space probability density function (PDF) for a discrete moment in time (t). This can be expressed as the posterior probability $P^{(t)}(\mathbf{O}|\mathbf{g})$ for object $\mathbf{O}$ being attended by the user at time t, given gaze information $\mathbf{g}$. The required information, which we use to infer the object of attention, is extracted from the stimulus signal $s(t)$.

Note that this work focuses on a thorough evaluation of *per frame* GTOM results. That is, our evaluation was aimed at the problem of spatial gaze-to-object mapping in each frame without utilizing information from results of previous frames. Hence, we followed a simple strategy to process GTOM results over time by accumulating the probabilities obtained in each frame over the time window of an entire fixation.

3.4 GTOM Approaches

In this section, we describe several methods for mapping gaze to objects in individual frames, which are all encapsulated in a formal Bayesian inference framework. We define two categories of gaze-to-object mapping techniques: *Passive Attention* GTOM (Section 3.4) and *Active Attention* GTOM (Section 3.4), illustrated in Figure 3.3 as simple graphical models. In both cases, the observed variable is the gaze position $\mathbf{g}$, while a probability is inferred for the hidden variable $\mathbf{O}$.

In *Passive* GTOM, we consider mapping gaze to features (e.g., pixels or target points). The hypothesis is that object attention follows gaze, i.e., where a user looks determines which features are seen and hence attended. A problem of this strategy is that attention is evaluated only locally near gaze positions without taking into account that other (more distant) parts of the objects, or even other cognitive processes such as memory [67], may have influenced gaze programming. Therefore, in *Passive* GTOM a bias toward selecting objects as the attentional targets that occupy larger portions of the stimuli is likely. If gaze positions are distributed randomly in space, variations in size cause a stochastic advantage for larger objects to be inferred as attention targets.

An alternative hypothesis is that object attention determines gaze ($\mathbf{O} \rightarrow \mathbf{g}$), which we refer to as *Active* GTOM. *Active* GTOM techniques specify an estimate for the posterior probability $P(\mathbf{g}|\mathbf{O})$ that gaze $\mathbf{g}$ follows given that object $\mathbf{O}$ is attended. This approach assumes that attention is equally distributed in object space, i.e., there is no stochastic advantage for large objects. Only high-level factors, such as task relevance, can be used as priors to increase the probability of particular objects being attended before gaze has been observed. Moreover, this approach accounts for situations where a user pays attention to a particular object as a whole and not to specific features. In this case, all features of the object could contribute holistically to the selection of an optimal viewing position.

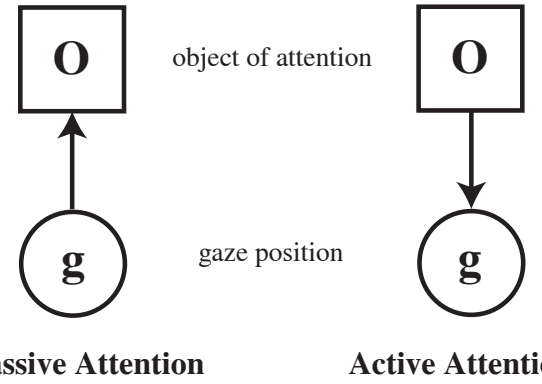


Figure 3.3: Simple graphical model to illustrate the two different approaches one can use for inferring attended objects from gaze. A square is used to illustrate that the object of attention is a discrete variable (e.g., an ID), while a circle is used to indicate that gaze is a continuous variable.

3.4.1 Passive GTOM – *Gaze Determines Attention*

In *Passive* GTOM, we assume that the gaze $\mathbf{g}$ determines what is seen with foveal vision and thus attended. To estimate a probability for an object to be attended, we assume that attention activates object features near the current gaze position more than distant features. We use the amount of activation estimated for features $\mathbf{o} \in \mathbf{O}$ being located on object $\mathbf{O}$ to derive a probability $P(\mathbf{O}|\mathbf{g})$ that object $\mathbf{O}$ is attended given that gaze $\mathbf{g}$ was observed. To avoid normalization issues, we estimate a proportional function $P(\mathbf{O}, \mathbf{g})$, from which we derive a posterior probability by normalization:

$$P(\mathbf{O}|\mathbf{g}) \propto P(\mathbf{O}, \mathbf{g}) \tag{3.4}$$

Ray Casting (RC) The simplest approach to estimate $P(\mathbf{g}, \mathbf{O})$ is to perform a binary evaluation whether a gaze position $\mathbf{g}$ is located on object $\mathbf{O}$ or not:

$$P^{RC}(\mathbf{O}, \mathbf{g}) = \begin{cases} 1 & \text{if } \mathbf{g} \in \mathbf{O} \\ 0 & \text{otherwise} \end{cases} \tag{3.5}$$

This is the most efficient GTOM strategy which requires only a single look-up in the item buffer.

Closest Feature Mapping (CFM) Another option is to assume that a user attends to the closest feature $\mathbf{o}$ of an object $\mathbf{O}$ and weight the euclidean distance with a Gaussian with a standard deviation of θ_s:

$$P^{CFM}(\mathbf{O}, \mathbf{g}) \propto \exp\left(\frac{\min\limits_{\mathbf{o}\in\mathbf{O}}\left\{\|\mathbf{o}-\mathbf{g}\|^2\right\}}{-2\theta_s^2}\right) \tag{3.6}$$

In theory, this strategy requires looking-up every pixel in the item buffer, i.e., complexity is $\mathcal{O}(N)$, with N denoting the number of pixels in the item buffer. However, since the energy of pixels exponentially drops to zero as the distance increased from the gaze location, a clipping radius (e.g., $r = 4\theta_s$) can be used to obtain an approximate solution without a significant loss of accuracy. Furthermore, in realtime applications a portion of the item buffer, which is the bounding window around the clipping circle, may be used to increase processing speed.

Fovea Splatting (FS) A second alternative is to assume that a user attends to multiple features at the same time and to infer the attention probability by accumulating all features/pixels $\mathbf{o} \in \mathbf{O}$ weighted by a Gaussian splat of the size of the fovea [170] (with standard deviation θ_s):

$$P^{FS}(\mathbf{O}, \mathbf{g}) \propto \sum_{\mathbf{o} \in \mathbf{O}} \exp\left(\frac{\|\mathbf{o} - \mathbf{g}\|^2}{-2\theta_s^2}\right) \tag{3.7}$$

The computational complexity of this method is equivalent to Closest Feature Mapping. However, instead of comparing distances while searching the closest pixel of each object we accumulate the energies.

3.4.2 Active GTOM – *Attention Determines Gaze*

In *Active* GTOM approaches, we assume that once an attention target is selected, gaze is directed towards it. This model estimates $P(\mathbf{g}|\mathbf{O})$, which is the posterior probability that $\mathbf{g}$ is observed given that $\mathbf{O}$ is attended. To infer the probability that $\mathbf{O}$ is attended from $\mathbf{g}$, we apply the Bayes Theorem, and thus we also call *Active* GTOM methods "Bayesian Inference GTOM" (BIGTOM) approaches:

$$P(\mathbf{O}|\mathbf{g}) = \frac{P(\mathbf{g}|\mathbf{O})P(\mathbf{O})}{\sum_i P(\mathbf{g}|\mathbf{O}_i)P(\mathbf{O}_i)} \tag{3.8}$$

The prior $P(\mathbf{O})$ can be used to increase the probability of task-relevant objects or to propagate previous GTOM results over time. However, our research focuses on one frame without any *a priori* knowledge, and thus we use a constant, which is theoretically $P(\mathbf{O}) = \frac{1}{N}$ (where N is the number of objects), in our evaluation.

Concerning the influence of attention on gaze position, we will investigate different ways of estimating the posterior $P(\mathbf{g}|\mathbf{O})$:

Center of Gravity Mapping (CM) A straightforward, but effective strategy, is to exploit the strong tendency of users to focus at the center of an object [63]. Similar to [193] (for mapping to words in a text documents) and [109](for mapping to small objects), we place a monovariate, but two-dimensional, Gaussian at the center of gravity of the object with a kernel size θ_s:

$$P^{CM}(\mathbf{g}|\mathbf{O}) \propto \exp\left(\frac{\|\mathbf{g} - \mathbf{c}_\mathbf{O}\|^2}{2\theta_s}\right) \tag{3.9}$$

To compute the center of gravity, we extract the set of (x, y)-coordinates of the pixels belonging to the respective object $\mathbf{O}$ and compute the average position of these samples.

In theory, this strategy requires computing the center of gravity for each object in the scene, i.e., complexity is in $\mathcal{O}(N)$). However, objects which are distant from the gaze position can be disregarded by assuming that their attention probability is zero. To this end, we can determine which objects have at least one pixel in the vicinity of the gaze position in a circular region around $\mathbf{g}$ (e.g., $r = 4\sigma_s$). The center of gravity is then determined for those objects only, while the rest have a zero probability. Another acceptable optimization is to use sparse sampling (e.g., $stepsize = 3pixels$), since the center of gravity is relatively stable against this simplification. Nevertheless, in contrast to *passive* GTOM approaches, it is recommended to generate and copy a full-screen item buffer, where resolution can be eventually down-sampled according to the step size used for sparse sampling. A coarse simplification which could be considered in applications where an item buffer is difficult to obtain, is to use the center of the object's bounding box projected to device-space coordinates. However, accurate visibility information is not available and thus should also be estimated, e.g., by using the depth of the world space bounding box. This is necessary to exclude objects which are barely visible, or to trim bounding boxes such that they exclude occluded object parts.

Normalized Closest Feature Mapping/Fovea Splatting (nCFM / nFS) Since the strategy described above assumes that a user targets gaze near the center of an object, we will also investigate strategies which better account for the object's projected shape (i.e., the pixel coverage in the item buffer). We follow the CFM proposed for *Passive*-GTOM approaches, but normalize as follows:

$$P^{nCFM}(\mathbf{g}|\mathbf{O}) = \frac{P^{CFM}(\mathbf{g}, \mathbf{O})}{\int P^{CFM}(\mathbf{x}, \mathbf{O}) d\mathbf{x}} \tag{3.10}$$

Here, we modified the closet feature-mapping method by specifying a function which sums up to 1.0 for all possible gaze positions $\mathbf{x}$. Analogously, we normalize the fovea splatting method:

$$P^{nFS}(\mathbf{g}|\mathbf{O}) = \frac{P^{FS}(\mathbf{g}, \mathbf{O})}{\int P^{FS}(\mathbf{x}, \mathbf{O}) d\mathbf{x}} \tag{3.11}$$

Both methods have in theory a computational complexity of $\mathcal{O}(N^2)$ since they correspond to a convolution. Simplifications that could be performed are: (a) evaluating only objects which are close to the gaze position, as described for CM, (b) sparse sampling of the item buffer, and (c) truncation of the splat kernel with a clipping circle.

3.4.3 Utilizing Motion Information

During smooth pursuits, we assume that the visual system attempts to minimize the velocity of an object relative to the gaze movement in order to maintain a stable retinal image. We utilize this behavior to improve the accuracy of the inference. To this end, we compute a retinal stability probability P^{RS} by using the gaze movement vector $\dot{\mathbf{g}}$ and object movement vector $\mathbf{m_O}$:

$$P^{RS}(\dot{\mathbf{g}}, \mathbf{O}) \propto \exp\left(\frac{\|\dot{\mathbf{g}} - \mathbf{m_O}\|^2}{-2\theta_r^2}\right) \tag{3.12}$$

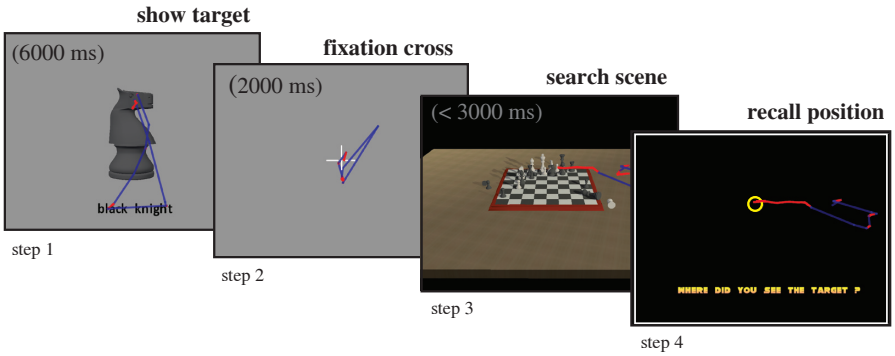


Figure 3.4: Procedure of one trial (in this case target-present). In step 1, the participant previews a target; in step 2 fixates on a cross (for validating later the eye-tracking accuracy); in step 3, he searches for the target. In step 4, the participant recalls the position of the target on a blank screen to ensure that he did not guess.

To model the standard deviation between feature and gaze motion, a retinal drift tolerance parameter θ_r is introduced. To obtain gaze movement information, we differentiate the filtered gaze position along consecutive gaze samples. To further smooth the gaze motion samples in fixations, we use a Gaussian filter kernel with a standard deviation corresponding to 2 gaze samples (sampled with $50Hz$). Object motion vectors are estimated by differentiating the temporal signal of gaze and object center positions ($\mathbf{c_O}(t)$) with a difference of Gaussian filter ($\sigma = 50ms$).

We combine this probability with the position PDF's specified above by a multiplication:

$$P(\mathbf{O}|\mathbf{g}, \dot{\mathbf{g}}) \propto P(\mathbf{O}|\mathbf{g}) P^{RS}(\mathbf{O}, \dot{\mathbf{g}}) \tag{3.13}$$

3.4.4 Relation to Previous Work

Three of the methods presented in this Section are related or equivalent to solutions proposed in previous work. These are Ray Casting (e.g., [156]), Fovea Splatting([170]) and Center of gravity Mapping (e.g., [193]). Note also that Ray Casting and Center Of Gravity Mapping could not immediately be used in our framework and had to be adjusted for the use with an item buffer. Mantiuk et al. [109] has introduced the idea of using motion information together with Gaussian distance metrics.

The other three methods (i.e., Closest Feature Mapping, normalized Closest Feature Mapping/Fovea Splatting) are, to the best of our knowledge, first proposed in this work.

3.5 Evaluation Methodology

In this section, we describe the eye-tracking experiment and the scenes used to obtain ground-truth data samples necessary for the evaluation of GTOM methods. Moreover, we will explain how the obtained data set is used to evaluate the accuracy of GTOM methods (Section 3.5).

Figure 3.5: 6 examples from the (6) static and (18) dynamic scenes used in target-present trials of the experiment overlaid with a part of the gaze path, recorded before the target was found. The red color parts of the gaze path indicate fixations. To visualize the object segmentation we overlaid the frame buffer images with the item buffer images (white color = target object). In the dynamic scenes we show the motion path (white line) of the search target or the camera, respectively. Most other "mobile" objects (i.e., space-ships, asteroids and cars) were also animated. (4 other scenes were already shown in Figure 3.1)

(a) Static (b) Moving Occluder (c) Moving Camera

Figure 3.6: Scenes where a background object (an abstract painting) had to be found. Scene (a) is static, in scene (b) an object moves in the foreground (the car) and in scene (c) the camera moves relative to the scene.

3.5.1 Background

To evaluate and compare the accuracy of GTOM methods, we need ground-truth data comprising pairs of known fixations ($\mathbf{F}_i$) and known attentional targets ($\mathbf{T}_i$). However, it is not trivial to find the actual attentional target $\mathbf{T}_i$, since the common way to do so is to use eye tracking and apply GTOM, the method we want to improve in the first place.

To solve this problem, we designed an eye-tracking experiment where we direct the user to attend a particular object (i.e., we prescribe $\mathbf{T}_i$) and use eye tracking to gather the corresponding fixation $\mathbf{F}_i$. The task we use for that purpose is a visual search task, because this does not hinder the user's ability to naturally deploy his attention in a scene, and requires active deployment of attention to locate a target. Thus, a visual search task can be used to *implicitly* instruct participants to fixate on a particular object in a scene, without visually altering or annotating the objects. According to [67], visual memory facilitates guiding attention and gaze during visual search, therefore we provide participants with a visual preview and an on-screen description of each target before each search task. To obtain an objective evaluation as possible, our experimental test bed includes an extensive set of 30 different scenarios that arise frequently in visual scenes (e.g., video games).

3.5.2 Eye-Tracking Experiment

Setup and Participants We used a Tobii x50 eye tracker (50 Hz), which was placed in front of the display of an Intel Core 2 Duo workstation with a 2.4 GHz CPU, 2GB RAM and an NVIDIA GeForce 8800 GTX graphics card. This setup was sufficiently powerful to simultaneously run the application at a frame rate greater than 50 fps, and also operate the eye tracker. The display was a commodity LCD display (IBM ThinkVision L200p with a resolution of 1600×1200 pixels at 100 dpi). 28 subjects, with ages between 20 and 55, participated in the study (12 females). All subjects had normal or corrected-to-normal vision. Participants were seated comfortably in front of the screen's center at a distance of 60 cm. No chin rest was used so as to allow for natural viewing behavior. By manually inspecting the recorded gaze data using our visualization tool, we found that only 22 gaze data sets were of reliable accuracy to be included in the evaluation

data set. The other 6 data sets were all from participants wearing contact lenses and exhibited serious systematic errors in the gaze recordings, which became most apparent during smooth-pursuit object tracking and when the fixation cross was shown.

Task and Procedure Each trial consisted of 4 steps: (i) preview a target object (6000 ms), (ii) fixate on a cross (2000 ms), (iii) search a scene for the target ($<$ 3000 ms), and (iv) recall the position of the target. This procedure is shown in Figure 3.4 and resembles experiments commonly used to verify attention models by measuring reaction times in visual search tasks (e.g., [189]). Experiment participants performed a block of 50 trials. The participants responded by pressing a key in case they located the target while in step (iii), otherwise they were instructed not to act. To motivate quick responses in all search trials, participants were told that their reaction time is measured. Among the 50 trials, 30 were target-present and 20 target-absent trials. In target-absent trials, participants had to withhold the response and wait until the application automatically started the next trial after reaching a timeout of 2000 ms in static scenes or after seeing one animation cycle in dynamic scenes. Prior to the main trials, each participant practiced the procedure on a block of another 10 trials (with different scenes). The eye tracker was calibrated between the practice and main block. Using the software provided by the manufacturer, a calibration was calculated based on 5 control points.

Obtaining the Ground Truth In order to make sure fixations really correspond to specific attended objects, we need to exclude detection methods other than eye fixation, eliminate guessing, and perform verification.

In each scene, we placed distractor objects sharing a feature (e.g., color, orientation or shape) with the target in at least one dimension. This causes targets and distractors to be indistinguishable by pre-attentive selection [178], and thus compels a user to identify targets by performing a serial search using eye fixations. From each target-present search trial, we obtain the ground-truth attention target and a respective fixation.

Through offline manual inspection, we selected these fixations using the following rules: **(a)** the fixation started briefly before a participant's response, **(b)** the fixation is reasonably close ($< 2^\circ$) to the target, and **(c)** a participant could correctly recall the target position. To check (c), participants had to place after each target-present search trial a circle on the target's last position within a void black screen (Figure 3.4).

From these strategically selected fixations, we generated an evaluation data set in which each entry consists of the target object's identification color code and the time-stamped gaze data, as well as the series of time-stamped item buffer images of the time window corresponding to this fixation.

3.5.3 Scenes

To highlight particular strengths and weaknesses of the evaluated algorithms, we designed a total of 30 different scenes featuring a variety of levels of difficulty for GTOM. We categorized 27 of these scenes into 2 groups: **(1)** 9 scenes without camera or object animation (group **Static**), shown at the top of Figure 3.5 and **(2)** 18 scenes where the target or the camera, or both, are

Method	θ_s	θ_r	max LLH
Ray **C**asting	-	-	-833
Closest **F**eature **M**apping	$0.7°$	$12°/s$	-691
Fovea **S**platting	$0.7°$	$12°/s$	-726
Center of Grav.**M**apping	$1.0°$	$12°/s$	-481
normalized **CFM**	$0.7°$	$12°/s$	-565
normalized **FS**	$0.7°$	$12°/s$	-419

Table 3.1: Optimal parameters and the maximum log-likelihood of each method.

animated (group **Dynamic**), shown at the bottom of Figure 3.5. In each group, we used different target sizes and object densities, as well as varying degrees of target occlusion. To reveal a potential bias of the method towards foreground scene objects, we constructed cases where a user has to search for a background object. Thus, the 3rd scene group, which we denote **Background**, includes 3 scenes where the target is an abstract painting and located in the background of many occluding foreground objects. One scene was static, the second was shown from a moving viewpoint and in the third, occluding foreground objects were animated.

All scenes were created and animated in Maya and exported with OgreMax to an XML file (including animation keyframes) and a set of mesh files to be loaded with the Ogre3D rendering engine that we used to build the experiment's application.

3.5.4 Performance Measurement and Parameterization

The data sets collected through the experiment presented in Section 3.5 will be used to evaluate and measure the accuracy of the GTOM methods proposed in Section 3.4. As a quantitative performance metric, we compute the log-likelihood (LLH) by summing the logarithm of the predicted probability for each evaluation data set:

$$LLH(\theta, M) = \sum_i ln \max \left\{ P(\mathbf{O} = \mathbf{T}_i | \mathbf{F}_i, \theta, M), \epsilon \right\}, \tag{3.14}$$

where $P(\mathbf{O} = \mathbf{T}_i | \mathbf{F}_i, \theta, M)$ specifies the probability predicted by a GTOM method M, configured with the parameter set θ, for the ground truth attention target $\mathbf{T}_i$. A small threshold $\epsilon = 0.01$ was used to remove outliers (i.e., zero-probability predictions, which may cause an infinite LLH, e.g. when $M =$ RC). We searched for the maximum LLH to specify the optimal parameters for each method which are listed in Table 3.1.

In addition, we use the success rate, which is a simple ordinal measure. We compute it for each fixation by the proportion of frames where a GTOM method has predicted the highest probability for the ground truth target.

3.6 Results and Discussion

An example visualization of the GTOM results for different methods is shown in Figure 3.7 and in the accompanying video. The visualizations demonstrate that depending on the method and visual scenario, the predicted PDFs can vary considerably.

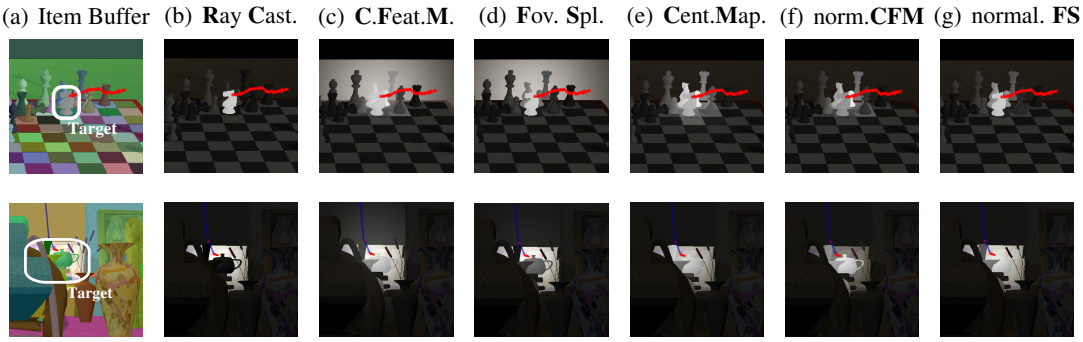


Figure 3.7: Visualization of the item buffer and the predicted PDFs for each method. In the chess scene the camera was moving in the right direction and the attention target was the knight figure. In the scene shown in the bottom row, the search target was the painting located in the background of many occluding foreground objects.

Two measures are used to estimate each algorithm's performance in predicting the object attention probabilities, as described in Section 3.5: (a) the average likelihood (normalized by the number of fixations), and (b) the success rate. In Figure 3.8 we depicted the results for the success rate and likelihoods. To illustrate the variation among subjects, we computed both measures by grouping fixations of all scenes of a category (Static, Dynamic or Background) and then ranked the results for each participant (Figure 3.8(a)). To show the variation of performance among different scenes, we grouped fixations of each participant for all scenes of a category and ranked the scores for each scene (Figure 3.8(b)). To provide a more intuitive illustration of the likelihood scores, we used in these graphs the exponential of the log-likelihood scores which were normalized by the number of fixations being grouped. We call this measure *average likelihood* and it corresponds to the expected likelihood that a certain target object and fixation pair ($\{\mathbf{T}_i, \mathbf{F}_i\}$) is observed, assuming that a particular GTOM model is correct.

In Figure 3.8(b) it can be also seen that the test scenes cover a broad range of difficulty levels, since the success rate and average likelihood drop smoothly from high to low scores for most methods. Also there is a considerable variation across different participants. An ANOVA on both measures (Model: $score \sim method + scene * partipant$) proved that there is a highly significant effect of the factors $method$ ($F_{5,2696} \geq 78.34, p < 10^{-15}$), $scene$ ($F_{29,2696} \geq 41.08, p < 10^{-15}$) and $participant$ ($F_{21,2696} \geq 3.48, p < 10^{-6}$), as well as in the interaction between $participant$ and $scene$ ($F_{488,2696} \geq 2.40, p < 10^{-15}$). Thus, we evaluated the significance of differences in a post-hoc analysis of the LLH-scores and success rates by using a generalized mixed-effect model (R-function *lmer* from package *lme4*), which is suited to fit to our repeated measurements data with the two interacting random effects $scene$ and $participant$ (Model: $score \sim method + (1|scene/partipant)$). On this model, we applied a Tukey-HSD test (using the R-function *glht* from package *multcomp*). A visualization of the HSD-test results can be found in Figure 3.9.

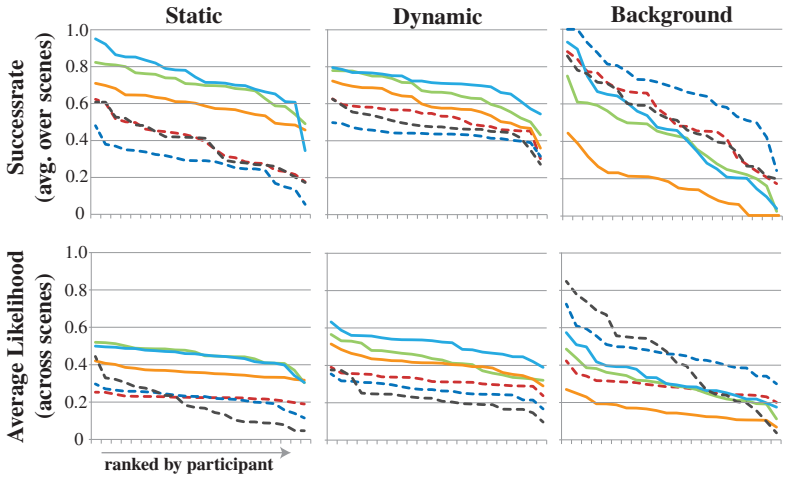
Comparing the different GTOM methods, the overall results suggest that *active* GTOM methods outperform *passive* GTOM approaches in the scenes where the target is a compact

object.

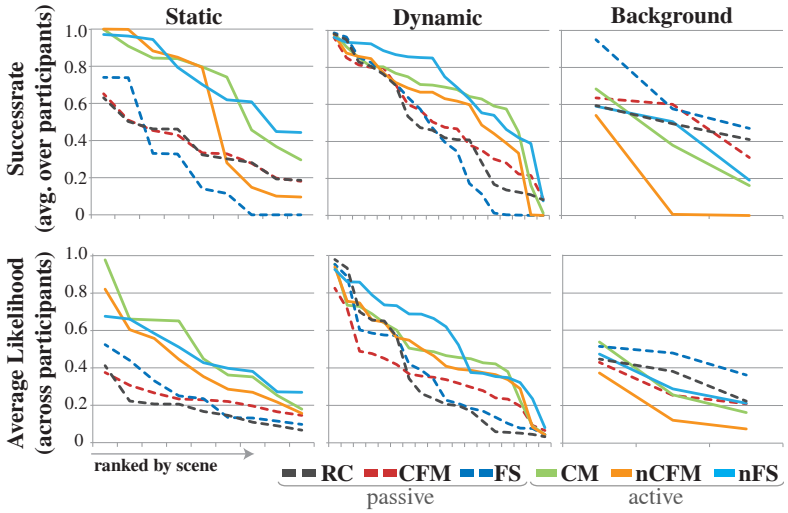
Passive GTOM approaches are more conservative as they predict attention to those features which are actually close to the center of the fovea. Moreover, they infer attention locally from object features near the current gaze position and thus are also less computationally expensive. *Passive* GTOM methods provide reliable probability predictions in cases where a user focuses on object details or on objects where features dominate (i.e., background objects). Other cases where *passive* methods provide good results are scenes with objects of similar size and few occlusions. However, since the spatial stochastic advantage of large objects is not compensated for *passive* approaches, those tend to increase importance of objects covering many pixels on the screen. This overestimation may become particularly severe with the FS approach. When compared to CFM or RC, FS performs poorly since the predicted probability increases for surrounding large objects due to integrating the energy of many pixels around the current gaze position.

Thus, when objects vary considerably in size or occlude each other, *active* methods provide better results. In contrast to *passive* methods *active* approaches treat *a priori* each object equally important, independent of the number of pixels covered on the screen. Besides compensating for the spatial stochastic advantage, *active* GTOM approaches better account for situations where a user does not focus on object details, but rather perceives the attended object as a coherent entity. The optimal position to obtain a coherent percept of an object is the center of gravity, which turned out to be one of the most effective and robust predictors of a gaze position, assuming the object of attention is known. Similar observations have also been obtained in experimental psychology research (e.g., [63]).

Overall, center-of-gravity-based methods appear to provide good results in our evaluation, particularly in cases where a target is occluded. The simplicity of CM is a great advantage for many applications and can be easily implemented by just using screen-space bounding boxes. Surprisingly, CM also performs well in scenes where a background object had to be searched. However, one reason for this could lie in our choice of objects to be compact, to some extent, in order to enable a meaningful visual search task. This is a limitation of our evaluation data set and for future work it is important to further extend our evaluation data sets with more scenarios where a user attends to background objects, such as a region in the sky or a wall. Since a user does not necessarily focus on the center of gravity of an object in this case, using the center of gravity as the only feature being evaluated introduces a strong assumption about gaze behavior which is in conflict with situations where a user focuses on object details or significant features (e.g., the handle of a teapot). nFS is a more general approach that is amongst the methods with the best scores in our evaluation and provides a significantly better probability estimate than CM in dynamic scenes. It works more consistently for a larger variety of object shapes. By integrating features around the gaze center, the circular nature of the monovariate 2D-Gaussian also yields a maximum in $P(\mathbf{g}|\mathbf{O})$ at locations near the object's center of gravity. Since this property of enhancing the probability in the center of gravity is not shared by nCFM, that method performs particularly worse.



(a) Results by Participant



(b) Results by Scene

Figure 3.8: Success rate and average Likelihood ranked by participant (top), or scene (bottom), respectively. We compare the methods Ray Casting (RC), Closest Feature Mapping (CFM), Fovea Splatting (FS), Center-of-gravity Mapping (CM), and normalized Closest Feature Mapping/Fovea Splatting (nCFM/nFS). The results were grouped into static scenes (no motion), dynamic scenes (object or camera motion) and background scenes (large object occluded by foreground objects).

(a) Success Rate (b) LLH

Figure 3.9: Post-hoc analysis for success rate and LLH scores. Algorithms are sorted by their intercept. Those which are encompassed by a rectangle are not significantly different, i.e., they are within in a 95% statistical similarity group (Tukey contrasts).

3.7 Conclusion and Future Work

In this work, we systematically investigated several alternatives to infer object-space attention from gaze in dynamic 3D environments. These methods are particularly important because the screen-space location where the gaze of a user is deployed does not always correspond spatially to attended objects in the stimuli. The Bayesian formulation proposed in this work aims at providing a framework within which gaze-to-object mapping techniques can be described, evaluated and compared. The resulting probabilities provide a confidence measure for gaze-based object selections or could be used for a probabilistic optimization of rendering methods (e.g., minimization of the expected attention error).

We performed a formal evaluation of methods based on a visual search task in a wide range of different 3D scenes, from which stimuli, gaze data and the corresponding ground-truth attention target were obtained. It turned out that it is favorable to use Active GTOM approaches that assume gaze follows attention. The main advantage of these methods is that they make the assumption that the basic units of attention are objects, and thus their likelihood to be attended is independent of the amount of features they bear or the number of pixels they cover on the screen at any time. Of course, in many cases, *passive* GTOM approaches also yield accurate results, when *Active* GTOM methods performed worse. Our intuition is that the two approaches are not mutually exclusive – instead, we believe they account for different attentional mechanisms (i.e., feature-oriented and object-oriented attention), which are both present under natural conditions. Thus, one direction to further GTOM methods could be to combine or unify *Passive* and *Active* approaches.

Another important challenge is to evaluate different approaches to combine probabilities over time. In this work we have only averaged over fixations, but accuracy could potentially be increased by propagating probabilities over subsequent frames, or fixations. This could be done, for instance, with Hidden Markov Chains (cf. [109]). However, the choice of the temporal processing method is also a matter of the application where GTOM is used. For instance, for gaze pointing tasks, non-probabilistic approaches to accumulate GTOM results over time, such as force physics [200], could be more adequate.

Moreover, it might also be interesting to find GTOMsolutions which are tailored for the use with stereo-scopic applications. For eye-tracking with stereo displays, binocular gaze information together with an item-buffer rendered in stereo should be used to maximize accuracy. The

additional use of gaze depth information (cf. [46, 173]) could help to disambiguate attention inference for objects which occlude each other.

Another important step which future work should address is the way segmentation images (i.e., the item buffer) are obtained. In this work, we rely on the assumption that the way scene geometries are grouped together (by the content artist who created the virtual environment) corresponds well to the combination of features a user's visual perception groups into objects. However, even if the scene geometry is structured well into objects, there are problematic geometries, such as the terrain of a scene, or vegetation, which are difficult to cluster into perceptual objects. Thus, it would be useful to investigate alternative item buffer generation methods which further subdivide geometry by geometric or texture feature analysis. We also believe that an optimal GTOM method should be based on a hierarchical representation, which would allow inferring object attention at different levels of this hierarchy. A user may pay attention to a group of objects, a single object, to parts of an object or to particular features. The ultimate goal is, however, to utilize GTOM approaches to non-artificial stimuli or stimuli where an object segmentation is simply not available. For these applications, gaze processing has to be combined with vision algorithms (e.g., proto-object maps [194]) that can automatically segment parts in an image which a user perceives as objects.

We assessed in this work the quantitative information value in the predicted probabilities. These attention probabilities distributed over multiple objects may find utility in algorithms that require a distribution of importances in order to optimally adjust their parameters to achieve their goals (e.g., minimization of the expected perceived error).

Improved GTOM can have a crucial impact within HCI and Computer Graphics, including attention visualization, gaze analysis, gaze-based semantic pointing, level-of-detail rendering, depth-of-field simulations, attention-aware stereo rendering, and many others.

CHAPTER 4

Attention Analysis and Prediction: Empirical Modeling

A shortcut is the longest distance between two points

Issawi's Law of the Path of Progress

With gaze-to-object mapping, the previous chapter evaluated how well object-based attention can be inferred from the gaze of one user at a particular time. This chapter goes one step further as it describes how gaze-to-object mapping results of several users and a sequence of frames can be combined in order to obtain an empirical model for object-based visual attention. The model is encoded in an *importance map* which is derived from a accumulated attention inference results. It is also important to note, however, that the results presented here were obtained a few years before the work on evaluating gaze-to-object mapping methods, which was presented in the previous chapter, has been done. Thus, at the time when the research was carried we did not know which is the best method to perform gaze-to-object mapping and used a method which was similar to Fovea Splatting. Rather, it was during the course of this work when we identified gaze-to-object mapping as an interesting problem which deserves further investigation.

This chapter is based on an article published in ACM Transactions on Applied Perception as "An Empirical Pipeline to Derive Gaze Predictors in 3D Action Games" [7] in 2010. However, the text which presented in this chapter has undergone a major revision. In the revision, some passages were shortened, while the introductory parts of this chapter were extended with new material written for our book chapter "Visual Attention and Gaze Behavior in Computer Games" [168]. This book chapter was published in "Game Telemetry and Metrics: Maximizing the Value of User Data" (Springer 2013), where this work was described in the context of methodologies to study user behavior in computer games.

4.1 Introduction

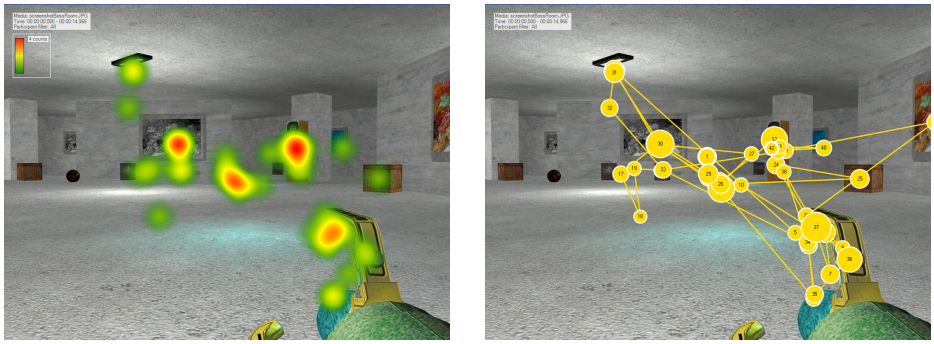
Since a user's visual attention is an important factor for the design of games and other virtual environment applications and video games, tools to measure which objects in a game scene a user is attending would be very useful to improve the design of these applications. Studying gaze behavior provides insights about visual attention of users and may assist game developers in identifying problems with the gameplay due to a misguided perception of the game environment. Moreover, knowing what a player does, or does not notice, can be used to control the difficulty of a game. For example, the designer may choose to make important task-relevant objects less apparent in the user's attentional field to increase the difficulty of the game, or accentuate them to decrease the difficulty. Other potentially useful computer graphics applications proposed so far include focus prediction for tone-mapping of high-dynamic range images [140], the selection of the optimal focal plane for depth-of-field effects [65] and the minimization of vergence/accommodation conflicts in stereo 3D to reduce visual fatigue as proposed in Chapter 5.

When an eye tracker is used to study gaze behavior in a virtual environment application or a video game, the output data is essentially a sequence of gaze points defined by a 2D position on the display screen and a timestamp. With this information, one can establish *where* gaze was deployed in screen-space over time. Analyzing gaze data for static stimuli can be time consuming, but it is even more difficult for dynamic stimuli (e.g. virtual environments such as games) [142, 159]. A useful representation of gaze data are gaze point density distributions, which can quantify the amount of attention deployed to each region in the display. When a stimulus rendered to the display is static or its changes are very limited (e.g. in web pages), it is sufficient to compute gaze point density distributions in screen space. A prominent tool to illustrate screen-space gaze density distributions is a fixation map [191] or heatmap. A heatmap visualizes gaze point densities with colors such that warm colors encode high densities and cold colors low densities. An example of a heatmap and a visualization of the corresponding gaze path can be seen in Figure 4.1(a).

For 3D computer games and interactive virtual environment applications, we have to assume a dynamic stimulus where temporal changes have a significant impact on the spatial distribution of gaze points on the screen. In this case one cannot accumulate gaze densities in screen space over long time periods because the viewpoint and the objects in the scene may considerably change their positions from one frame to the next. When spatial properties, such as the viewpoint or the object position, are changing frequently, it may not necessarily be appropriate to analyze where a user is looking. Instead, if we consider that semantic properties of scene objects are changing far less often, a more useful approach is to study what a user is looking at, especially since the meaning of game objects is supposed to have a major impact on the attention of a user.

4.1.1 Measuring "Where" but Analyzing "What" Users are Looking At

To analyze in a dynamic scene what a user is looking at, we need to record the changes in the display during the eye-tracking study. In the subsequent analysis, the recorded data is then used to reconstruct the frames depicted on the display in temporal alignment with the corresponding gaze data. Thus, gaze-analysis tools provide screen recording functions which capture the



(a) heatmap (b) gaze path

Figure 4.1: Example heatmap (a) and scanpath (b) from one participant viewing game stimulus for 15 seconds. In the heatmap, red areas indicate locations of a higher number of fixations. It has to be noted, however, that to obtain enough gaze points for this type of visualization, the gaze data visualized here had to be recorded by showing a static screen shot of the game. Thus this example is less representative for gaze behavior as it would occur during a real interaction with the game.

images rendered to the display during the experiment, which are then played back as a motion picture during the analysis stage. A synchronous visualization of the recorded gaze data superimposed on the playback of the corresponding stimulus provides an intuitive clue about the behavior of a particular participant. But with this functionality alone, one can just study the behavior of a particular subject in particular situations.

In dynamically changing games, different subjects are unlikely to be presented the same stimulus while playing a game. To complicate things further, even if the participant partakes in a number of gaming sessions, it is unlikely that the same sequence of in-game events will be triggered to generate the same stimulus. Moreover, encoding what a participant might actually be looking at mainly depends on the person who performs the analysis. For many purposes, an objective statistical evaluation, such as computing the gaze density distribution over different objects, might be preferable. Commercial gaze-analysis tools can accumulate gaze-points for manually defined regions of interest. To outline objects of interest in motion pictures, the experimenter has to define regions of interest (e.g. bounding rectangles or polygons) around the objects on a frame-by-frame basis. This can be a tedious and time consuming procedure. To some extend, tools from computer vision, such as segmentation algorithms, could assist in this process. However, translating pixel regions to semantically encoded scenes remains a difficult problem in computer vision.

4.1.2 Advantages of Rendered Stimuli

Fortunately, obtaining a semantic representation of the stimulus is significantly easier for stimuli which are rendered, such as in computer games or virtual environment applications. In these applications, information about any scene entity can be extracted from the internal represen-

tation of the rendered scene directly. 3D graphics engines usually render each image from an object-based representation of a scene, from which semantic information can be obtained to a considerable extent. Recording the graphics engine's internal representations of application states allows the conservation of object-space information of the stimulus, which is otherwise very difficult to extract when only rendered images are available. Therefore, we could use these facilities to map gaze points back to the 3D objects that were observed during a gaming session.

An important step of the proposed methodology is to map gaze data to objects. This is done by a gaze-to-object mapping algorithm (Chapter 3) determining the potential target(s) of each frame with respect to the gaze recorded at the time the frame was displayed. Gaze targets are individual objects which are represented by an identification number (ID). In some cases it might be interesting to quantify how often a particular object was attended, but for realistic game-levels we have to assume that each player navigates uniquely a spatially large environment containing many objects. Under these circumstances gaze is distributed very sparsely and is not suited for a statistical analysis. Therefore, rather than focusing on a particular instance (e.g. "AlienMonster_57") instead it would be interesting to compute gaze statistics on abstractions which represent object categories or semantics. Thus, gaze analysis tools should be implemented in a frame-work consisting of several layers of abstractions, as described next.

4.1.3 Layers of Abstraction

Overall, gaze analysis can be performed at different levels of abstraction. We distinguish four layers in which the stimulus can be represented in the analysis:

Screen space: Gaze points in 2D (e.g. position = [0.1,0.5])

Object space: Object instances (e.g. ID = 2933)

Property space: An object's category, state and behavior (e.g. category ="Alien Monster", distance = 5m, behavior ="approaching", avatar health state = 10%, etc.)

Semantics: An object's meaning to the user according to game task (e.g. "attacker","close", "dangerous", "high risk").

In Figure 4.2 we illustrated the levels of abstraction with an example of a game where a user has to move a pedestrian across the street: In the first abstraction layer (top), we see pixels as seen by the player of the game. The next abstraction is the object space, where we have particular instances, such as cars and trees, with unique IDs. In the third layer, individual objects are abstracted in terms of their properties, including the object category (e.g. "car") and spatial properties (e.g. velocity or position). In the semantic layer (bottom), the scene is abstracted according to the meaning of the objects to the user and the task at hand. In this example, the user has to move the avatar across the street, which becomes hence a pedestrian. For a pedestrian, most task relevant objects are the oncoming car and the car currently passing, whereas the car which has already passed by is not important. On the other hand, details of objects behind the street (e.g. houses, trees and sky) are of low relevance and can be abstracted as background.

To perform these abstraction in a computationally and user friendly way, we propose a pipeline of three stages, which is summarized in the following paragraph.

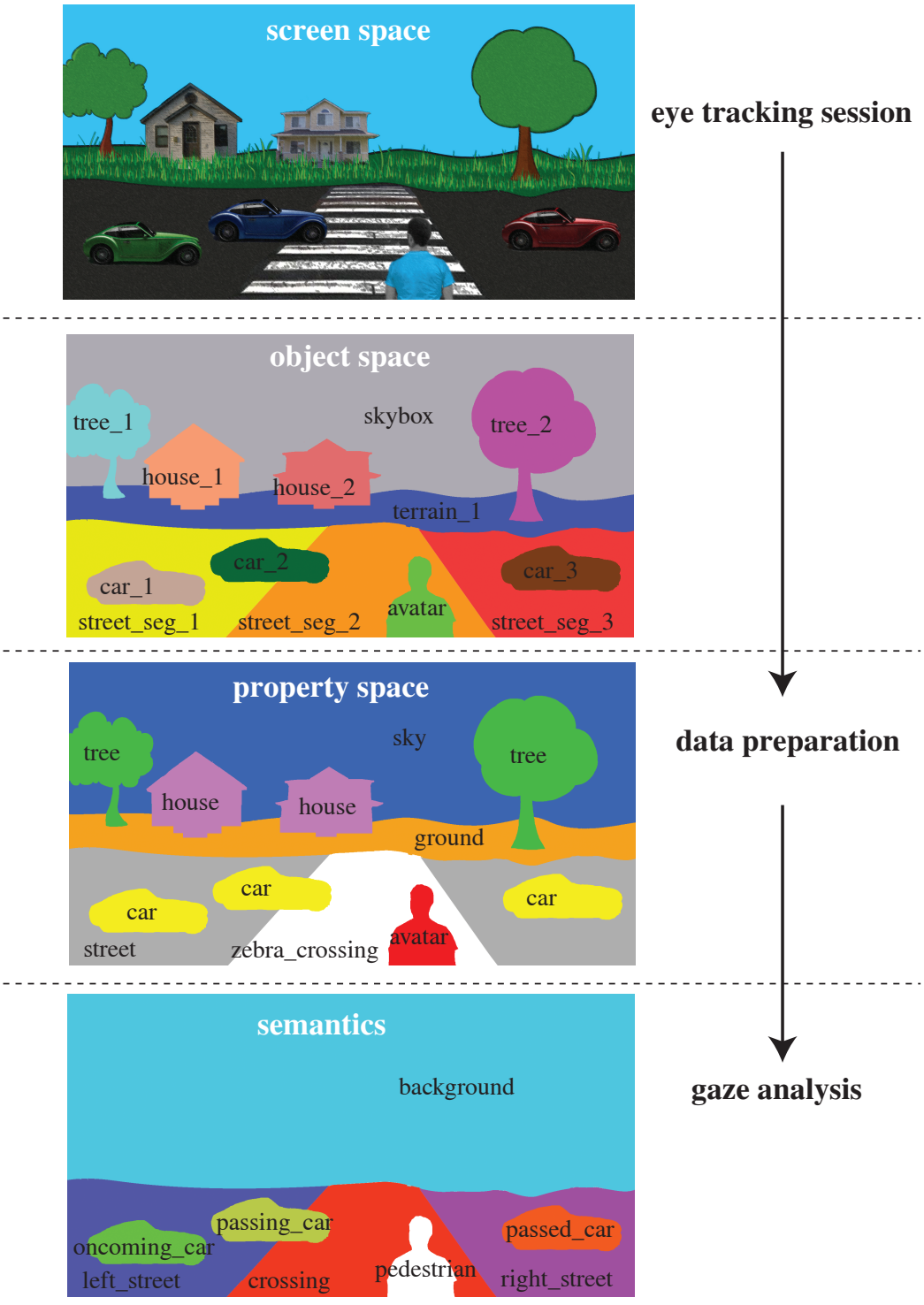


Figure 4.2: Example for layers of abstraction in a pedestrian road crossing task.

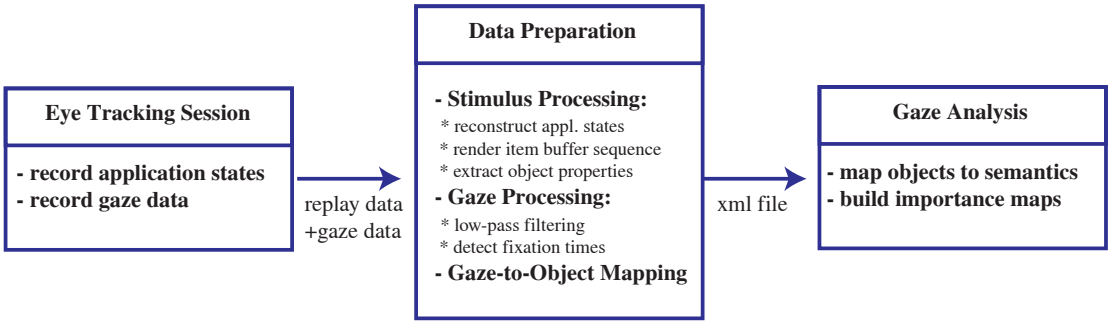


Figure 4.3: Overview of our pipeline. The eye-tracking and data preparation steps may be performed once, then the user can design and generate different gaze predictors.

4.1.4 Analysis Pipeline Overview

Figure 4.3 shows an overview of the pipeline we propose for gaze analysis. Moreover, the steps of the pipeline were added to Figure 4.2 to relate them with the corresponding layers of abstractions we use to represent the stimulus.

The pipeline starts with an **eye-tracking session** (Section 4.3), which is done to record *stimulus data* in the form of a replay-file which allows reconstructing the complete application state every frame, and *gaze data* in the form of a gaze data file recorded by the eye-tracking software.

In the next step of the pipeline, we prepare the input for the analysis. **Data preparation** (Section 4.4) requires gaze data processing by low-pass filtering and fixation identification, as described in Chapter 3. Moreover, we reconstruct the sequence of states of the application and render the item buffer images. Gaze-to-object mapping is then performed with the processed gaze signal and the item buffer sequence we reconstructed from the replay of the application states. The result of the data preparation is a sequence of frames, each consisting of a gaze-to-object mapping result (i.e., an attention probability) and the current properties of scene objects, which can be stored in an xml-file loaded by the analysis tool.

Since our goal is to build gaze statistics for object properties and not individual objects, we capture changes to the scene graph (an object might turn from a friend into an enemy) and analyze similarities between objects on a semantic level. However, in practice we can only observe combinations of properties through eye tracking, and therefore the number of properties used for deriving importance values needs to be controlled in order to obtain statistically relevant estimates.

Thus, we perform the **gaze analysis** in an additional step (Section 4.5 - 4.7), where the user of the system defines simple Boolean *high-level properties*. Using a scripting language, the experimenter specifies rules how certain properties shall be “interpreted” and selects the high-level properties to be used as the basis of the *importance map*, which is the result of the gaze analysis. In this stage (Section 4.6), the user of the system intervenes to control the analysis process, allowing him to experiment with various kinds of properties to find a mapping that best encodes the peculiarities of the application (e.g. different tasks and interaction interfaces).

The values in the importance map are derived by accumulating fixations from many frames, as described in Section 4.7.

4.1.5 Full Overview

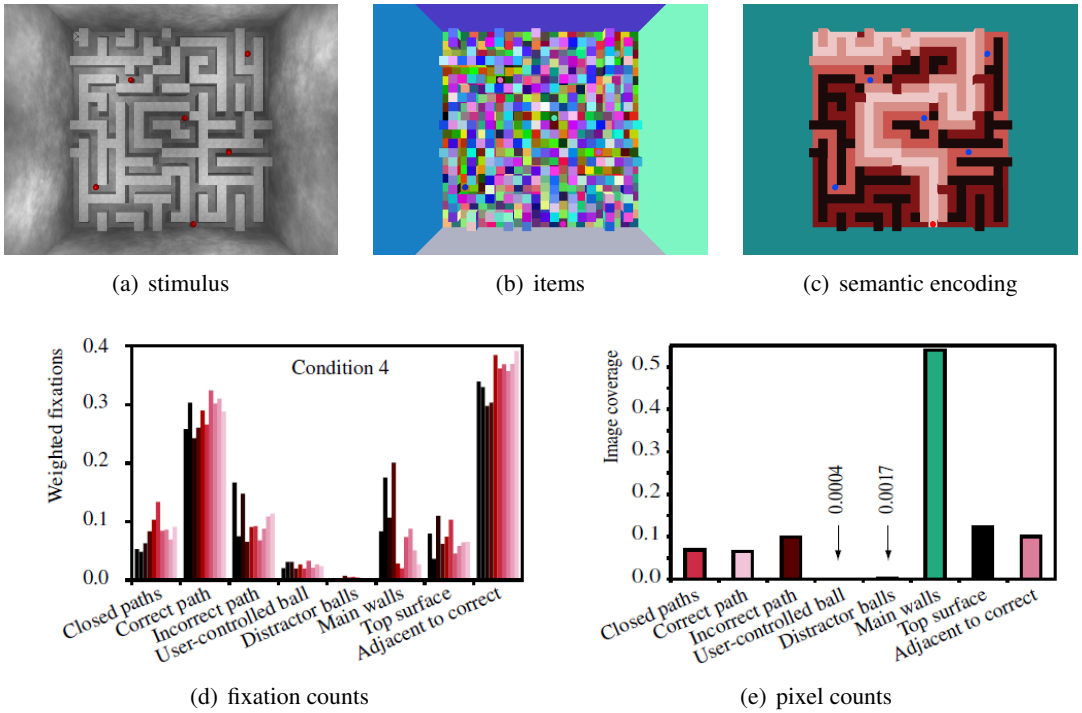
The following section we first discuss related work on gaze analysis and attention prediction in virtual environments. Then, the theoretical concepts and details of our analysis pipeline will be presented in Sections 4.3 - 4.7. Section 4.8 will describe how importance maps, which are the result of this pipeline, can be used to predict visual attention. With Sections 4.10 to 4.12, the chapter will conclude with an experimental investigation of the complete pipeline by means of an exemplary study carried out with a first person shooter 3D game.

4.2 Related Work

4.2.1 Gaze Analysis

Analyzing eye-tracking recorded in games or virtual environments allows to study which objects and events a user attends to in games [157, 164]. Gaze analysis is often performed with visualizations of gaze point density distributions, such as fixation maps [191]. This approach can be also extended to three dimensions by accumulating fixation points on the 3D surface as heat map textures [158] or by accumulating heat of 3D gaze points as particles in a volume [134]. Trends and requirements for visual gaze analysis have been reviewed by Stellmach et al. [161], who categorizes three types of visualization techniques: (1) projected, (2) object-based, and (3) surface-based attentional maps and carried out a survey with experts in the field of eye-tracking to investigate the usefulness of different gaze visualization features. A problem with projected and surface based attention maps is that in VE applications where a user can navigate through a large environment and the scene's surface shown on the display throughout interaction can be enormously large. To obtain useful insights from visualizations of gaze point densities, this poses the problem of sample size as gaze points tend to be distributed rather sparsely over the large surface. Analyzing gaze in object-space could be a promising approach to reduce the problem of sample size.

A first attempt of analyzing attention with respect to the semantics of attended objects was proposed by the work of Sunstedt et al. [170]. They studied fixation behavior in a maze game shown from a fixed birds eye viewpoint (Figure 4.4(a)). This work proposed to map gaze data to object space using an item buffer (Figure 4.4(b)). The elements of the maze were clustered according to their meaning in the maze navigation task (Figure 4.4(b)). Structuring the maze according to high-level properties such as "correct path" or "closed path" then allowed them to build fixation frequency tables with respect to meaning (Figure 4.4(d)). Since fixation frequencies counted per semantic category did correlate among participants, but not with the a category's pixel coverage (Figure 4.4(e)), their result supports the hypothesis that semantics are suited to build fixation statistics or predict attention. In this work, we consequently follow this path pioneered by Sundstedt et al. and attempt to do an important step further, which is to generalize the basic concept to games with three-dimensional navigation and more complex semantics.

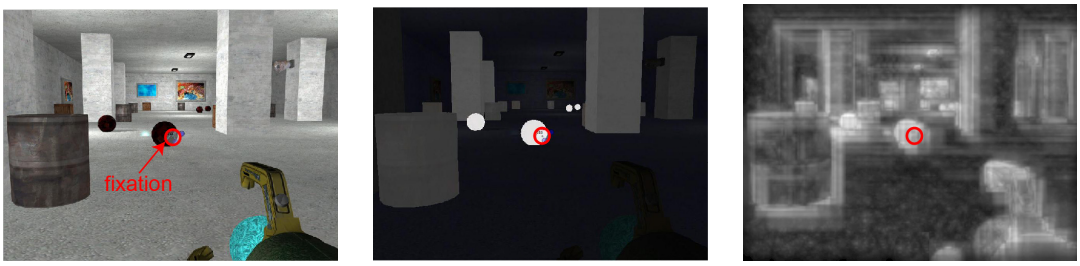


(a) stimulus (b) items (c) semantic encoding

(d) fixation counts (e) pixel counts

Figure 4.4: Results of Sunstedts et. al [170] work which was the first attempt to collect fixations according to semantics. The meaning of a particular item in this game was defined by whether it is a user controlled ball, a distractor ball, or a tile located on the correct path, an adjacent to correct path or an incorrect path.

In parallel to the work described in this thesis, similar approaches have emerged been explored in other research groups. Sennersten and Lindley [148, 149] proposed to analyze gaze in terms of Volumes of Interest (VOIs) or Objects of Interest (OOIs) in order to provide insights that are difficult to obtain with screen-space techniques only. Similar to our work, they propose to map gaze coordinates to objects and to keep track of changes in a game application with an object logging tool. Object-space gaze statistics can be also used for a visualization by assigning each object with a color value encoding its "visual attractiveness" [160]. Logging game states, as done in our gaze-analysis pipeline, has been also proposed by Nacke et al. [121] to allow a psychophysical analysis of player and gaze behavior in computer games. This approach was extended to a *logging and interaction framework* (LAIF) in [122], which enables those inexperienced with game design and programming to develop games and analyze them in a research environment. An overview on logging techniques for games can be found in [147], where the LAIF is also described in further detail.



(a) stimulus with gaze (b) semantic prediction (c) salience map

Figure 4.5: Salience maps compared to object-space predictors: Image (a) shows the image seen by the user. Visualization (b) illustrates importance values predicted for objects from gaze statistics, while image (c) shows the corresponding salience map. Since our method is sensitive to semantic properties, it can predict better the high importance of doors, while salience maps are less selective and predict the importance of pixels rather than objects.

4.2.2 Attention Prediction

The most prominent way to predict visual attention is to use *computational models* that have been established by neuro-science or cognitive psychology. In these fields, the main purpose of computational models is to run simulations that allow studying the behaviors predicted by the model and confirming, or falsifying, these predictions by comparison with experimental observations. Nevertheless, the models also found a direct application as tools to predict attention of users in computer-graphics applications or to optimize computer-vision algorithms (e.g., object tracking).

Most original computational models to predict visual attention are derivatives of the algorithmic framework to compute saliency maps, which has been originally proposed by Itti et al. [78, 79]. For non-static stimuli such as videos or computer games, a significant performance increase could be gained by introducing temporal features [77, 133], notably motion, flicker or Bayesian surprise. Evaluations on various commercial games [133] showed that even color has a non-negligible contribution, which can be explained by the fact that game designers prefer to texture task-relevant objects with salient colors. The advantage of saliency map algorithms is that they work on any sequence of digital 2D images, giving per-pixel scores of bottom-up attention. Since they are computed from low-level features only, they perform with greatest reliability in free-viewing conditions. However, due to the limitation of the model to bottom-up attention, they alone fail to predict a user's attention during the execution of a task [22,60,97,130]. Task-oriented attention, which we have to assume in interactive applications, is focused and a top-down dominated process. Thus, there are also top-down extensions of salience map algorithms. One prominent example can be found in Navalpakkam et al.'s work [123], which was inspired by Wolfe's Guided Search model [190]. Since Guided Search hypothesizes that search is guided by low-level features characterizing a search target, the application of this model for gaze prediction is based on the assumption that every task can be decomposed in to a series of visual search tasks. However, the stumbling block is how to determine the search target of a user

at any time. As a solution for this problem, Navalpakkam et al. [123] proposed an ontological model that uses a task graph to capture semantic relations between symbolic classes of objects, their super classes and the task. Using this model, a task-relevant object can be determined at each time. To predict attended locations, the weights of feature channels in the computation of the salience map are biased top-down such that the signal-to-noise ratio between the target and the distractors is maximized.

Our work is mostly related to Navalpakkams approach, since we propose a system which allows to specify an ontology in form of a script of rules which map "raw properties" to semantics. However, instead of using a computational model, we assign importance values, which we derive from gaze data, directly to semantics in object space. Thus, attention prediction requires only to perform an importance map look-up for each object in the scene, which is very efficient compared to conventional salience map algorithms, which require a lot image processing. An example for the output of our approach compared to the result of conventional salience maps is shown in Figure 4.5.

Similar to the ideas presented in this chapter, there recently emerged some other attempts in literature which attempted to derive attention predictors *empirically*, i.e., by learning from gaze data. Most examples in this direction are inspired by the seminal work of Peters et al. [132], where task-dependent influences were captured with a neural network. Trained with gaze data, the neural network learns to predict top-down salience from low-level features. With the recent work of Borji et al. [16–18], a great step forward in attention prediction frame-work has been made. Two years after the publication of the work presented in this chapter, they proposed also an empirical approach for attention prediction which accounts for goal-driven attention. They derived a probabilistic model with Bayesian inference preformed on a huge data set of eye fixations, which was recorded from users playing several types of video games. They used models that combined object-based and feature-based approaches. To correlate gaze with scene objects, they either used manual annotations or employed state-of-the-art object-detection algorithms. Moreover, feature-based classifiers, such as gist features, were used in their attention-prediction algorithm as well. In [18] and [17], they extended the probabilistic learning approach predicting visual attention by using further sources of information, including the global context, locations which were attended before and previously executed user actions. Using information about previous motor actions captured from user input devices (e.g., remote steering wheels) together with previous gaze recordings may allow predicting future gaze locations in setups using real-time eye tracking [18]. Though being solely evaluated for computer games, which have predictable tasks, stereotypical behaviors and few objects compared to real-world scenarios, the results of Borji et al. are a significant step forward compared to previous work on saliency maps, which was stuck in bottom-up attention prediction for over one decade.

Overall, there are many publications on computational models used to compute salience maps for attention prediction. A huge number of derivatives of the classic salience maps have emerged, and various alternative approaches have been proposed, reaching from pattern-classification models, such as neural networks, to probabilistic approaches, such as Bayesian or graphical models. An exhaustive and recent overview on computational modeling of visual attention can be found in the state-of-the-art report of Borji and Itti [15].

4.3 Eye Tracking Session

Since in this work we venture into the area of object-based analysis, we require access to the internals of the application. Hence, we can record the object-oriented scene graph instead of capturing the rendered frames with a screen-recording tool (as has to be done for e.g. commercial games [133]). Apart from the significantly decreased requirements for data storage, this method provides us the possibility to fully reconstruct the scene graph with all its properties, for every frame. To this end, the application has to be modified to record changes to the scene graph and the camera during runtime. To guarantee sufficient frame rates throughout the experiment, analysis is performed separately using a replay tool to reconstruct the application state at every frame. To minimize the computational overhead and storage costs, the recording tool should only track changes in the application. The application states are then reconstructed in the data preparation stage (Section 4.4) by loading the application's initial state and applying the recorded changes subsequently. Another requirement is a reliable synchronization of the recorded frames with the eye-tracking data. This can be achieved by operating the eye-tracker via the game application, by initializing and starting the eye-tracker and the replay simultaneously, so that both have the same temporal starting point.

4.4 Data Preparation

The goal of the *data preparation* step is to match stimulus data with gaze data and transform both into an abstract form that can be used by the analysis tool. This makes the analysis step (Section 4.5) independent of the application and the eye-tracking software. Furthermore, this allows rerunning the analysis with different parameters without having to redo the expensive application-specific steps.

Data preparation includes reconstruction of the stimulus presented to the player from the replay file recorded during the eye-tracking session, extraction of the scene graph's properties, and gaze-to-object mapping. While gaze-to-object mapping has been described in Chapter 3), this section focuses on how object properties can be extracted and stored.

4.4.1 Extracting Scene Properties

During the replay, we need to extract those properties from the scene graph that potentially have an impact on gaze behavior. For each frame, we extract the properties of all elements visible in the player's field-of-view. Exact visibility is determined using the item buffer. Further, we also store the properties of the player, required to infer the current task the player is performing, and general environment properties. The selection and interpretation of relevant properties is then carried out during the analysis stage and it is controlled by the user via a scripting interface, described in Section 4.6. Another purpose of this step is to normalize numerical properties (Section 4.4) for intuitive use.

The property space describes the properties of a scene or even the entire state of the current application. Ideally, such a description is generated for the entire scene, or at least for all visible objects. The reason why it is useful to consider apart from the fixated object also other objects in

the scene is that we should account for the context under which a fixation did occur. If the scene and the viewpoint changes, this is important because we need to track how many other objects could be concurring alternative targets for the fixations being issued.

There are several properties which could be of interest. Overall those can be divided in properties of objects and properties reflecting the current behavior of the user and the avatar being controlled.

Object Properties

- **Visibility:**All objects which were visible in the camera's field-of-view had a potential influence a users's behavior and could be potential fixation targets. Visibility can be determined directly from the item buffer where only visible objects may cover any pixels.
- **Object category:** The most important property is the category of an object which allows us to link an object to semantics. As we reasonably assume that the category of an object is a static property, we just need a table where each object ID is mapped to the respective category of an object.
- **Spatial properties:** Spatial properties like size, motion or position in screen space could be of some interest in the analysis too.

Player Related Properties

- **Game/Player State:** The current state of the game and the player may also affect user behavior. For instance, a low health state could cause the player to focus on searching health items.
- **Interaction State:** It might be also interesting to analyze gaze behavior with respect to the way the user is interacting with the application. Hence, it is useful to include the actions of the avatar (e.g. "running" or "shooting") or input events from mouse keyboard or joypad.

4.4.2 Logging Tool

To extract scene properties, a light-weight interface to the game engine is defined, which is used to notify a logging tool about changes of the game's internal parameters, such as the view-transformation matrix of the player camera or variables of scene entities (e.g. objects and other relevant properties of the game state). Having access to those very basic parameters, the logging tool then computes properties which might be useful for the analysis or the inference of semantics. For example, the tool infers screen space positions, bounding windows or motion vectors from camera parameters and object world-space bounding boxes.

Log Format

Though many of the properties are static throughout the game (e.g., category), we have to assume changes in many other properties (e.g., visibility or user input events), which hence have to be

```
<frame>
    <timestamp>36433.4343</timestamp>
    <object>
        <id>12423</id>
        <visibility>1.0</visibility>
        <attention_score>0.4</attention_score>
        <category>Car</category>
        <screen_bounding_window>
            <min_x> 0.1 </min_x>
            <min_y> 0.6 </min_y>
            ...
        </screen_boundin_window>
        ...
    </object>
    ...
<frame>
...
```

Figure 4.6: Example for an xml-file storing a log file used for gaze analysis

logged for each frame. If only fixations are considered in the analysis, it is useful to define a format where for each fixation the fixated object and a description for all frames between the begin and end time of that fixation are logged. For each frame description, one should log the time-stamp, the IDs of the visible objects, their dynamic properties and a description of the current game state and user input events. In Figure 4.6 we show an example how the log format for a video game could look like.

4.5 Gaze Analysis (Overview)

The input of the gaze-analysis tool is a sequence of frames containing information about the stimulus abstracted as object properties, information about the user (e.g., current task being executed) and the result of the gaze-to-object mapping pass (i.e., a probability for each object to be attended). In the analysis stage the user of the tool has to specify in which way properties of the stimulus and the player should be used or "interpreted". This means that at this stage we go from the abstraction layer *property space* to *semantics* (see Figure 4.2). Finally, the tool outputs a gaze statistic which quantifies the amount of attention an object with a particular meaning has been fixated by the user.

We denote this output as importance map $I(\mathbf{x})$. The importance map reflects how often an object holding the set of (abstract) properties $\mathbf{x}$ is attended by the user. This importance map can be either used to visualize the deployment of attention for analysis purposes, or to predict attention in an application. Figure 4.7 illustrates how we access the importance map. This is done in a two-step process: in the first step, we abstract objects by their raw properties $rp(\mathbf{O})$, i.e., we first map from object space to property space. Since there are so many properties which could be hold by an object, we perform another abstraction from the layer of raw properties ($rp(\mathbf{O})$) to the layer of higher-level properties ($\mathbf{x}(\mathbf{O})$). We denote the function which performs this mapping as *Semantic Transformation Function* ($\mathbf{x}(\mathbf{O}) = STF(rp(\mathbf{O}))$. This functions assigns to each (visible) object ($\mathbf{O}$) a set of higher-level properties ($\mathbf{x}$). The output of STF

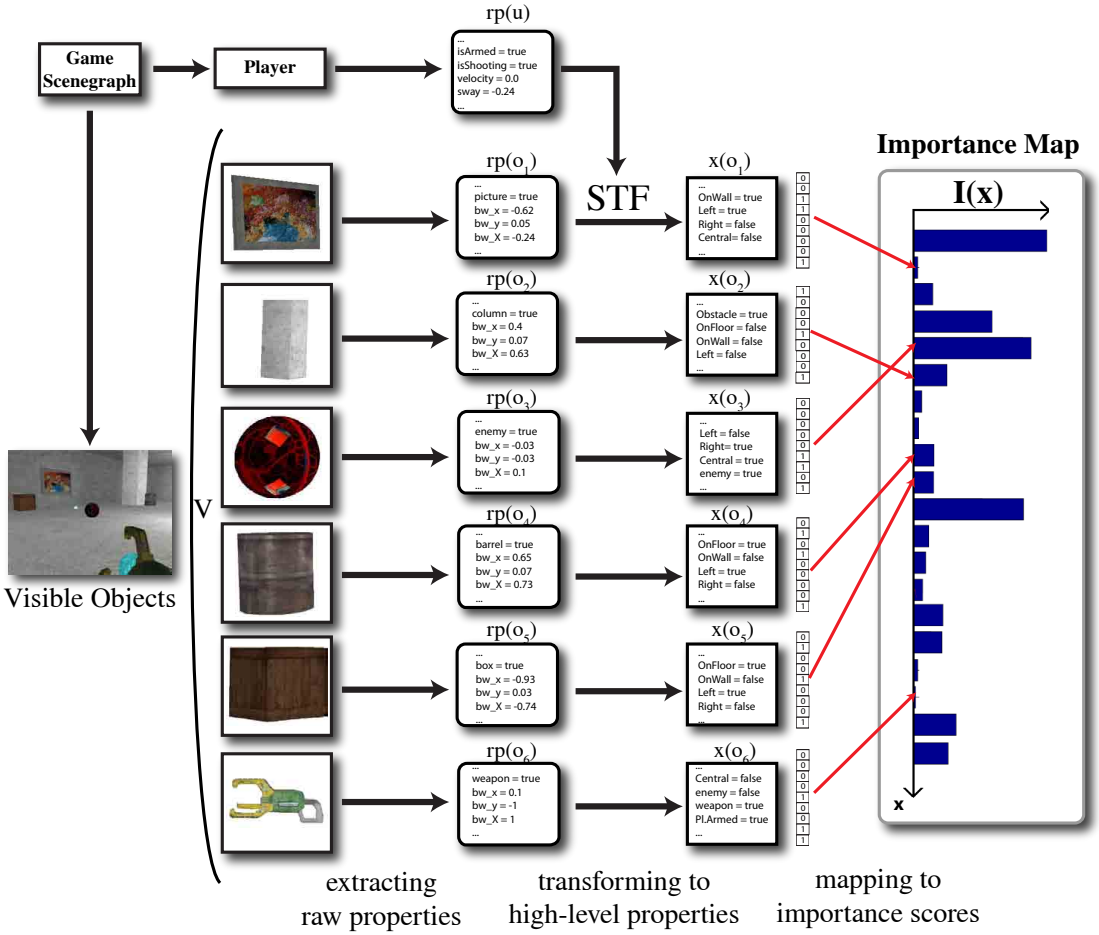


Figure 4.7: The desired result of our gaze-analysis pipeline is an *importance map* which is a function of high-level properties. High-level properties ($\mathbf{x}(\mathbf{O})$) are an abstraction of raw properties ($rp(\mathbf{O})$). The abstraction is defined by the Semantic Transfer Function STF. This illustration shows an example how the importance map is used to predict attention in one frame. The set of visible objects is extracted and mapped to a higher-level property $\mathbf{x}$ that serves as a key value to access one entry in the importance map.

is represented by $\mathbf{x}$, which is a simple data structure: a Boolean vector denoting whether a qualitative property q is held by an object ($\mathbf{x}^q = TRUE$), or not ($\mathbf{x}^q = FALSE$).

The STF is the stage where we need an *intelligent* mapping, since mapping from properties to semantics requires knowledge. We thus provide an interface to the pipeline where a human operator can specify and control how raw object properties are mapped to higher-level properties. The details how the STF can be algorithmically realized are given in Section 4.6, which provides a formal description of this function. Furthermore, Section 4.6 sketches an exemplary solution how this function can be specified by the user of the gaze-analysis pipeline (i.e., the experimenter) via a scripting interface.

Having a system which abstracts scene objects by their properties, we further need to specify a function mapping these properties to an importance. With our approach, we propose to infer importance from gaze data collected from many users while navigating one environment (e.g., a game level). The importance map is a simple structure which treats each possible $\mathbf{x}$ as a category for which an importance value is defined. The importance value of an $\mathbf{x}$ is assumed to be proportional to the probability that an object with $\mathbf{x}$ is attended. Actually, the importance values are the parameters of the statistical model which we encoded with the STF and the importance map. To obtain importance values, we propose a straight-forward solution: we count how often objects with $\mathbf{x}$ were attended. However, a problem that arises with this approach is that each $\mathbf{x}$ sample occurs with a varying frequency in the stimulus. This is because the visibility of objects changes when a user navigates an environment. Only those $\mathbf{x}$ can be attended at a time t which are associated with currently visible objects. The details on how we cope with this issue and how we generate importance maps from gaze data of many users will be given in Section 4.7.

4.6 The Semantic Transfer Function

Having a representation of the stimulus in property space, it is important to focus on those aspect of that information which might be relevant for the behavior of the user being studied and in which the user of the analysis pipeline is interested. Assuming top-down attention is mainly influenced by high-level processes, the most appropriate way to represent the stimulus is a description accounting for the meaning of objects according to the current task a user is performing. However, inferring meaning from object properties is a complex problem and requires to introduce knowledge into the analysis pipeline. To this end, we propose to use a script-based solution where the user specifies a set of rules defining how "raw" object or user properties should be translated to meaning. For the sake of clarity, we will further use the term "raw properties" (rp) to distinguish the output of the data preparation step from the result of the semantic transformation, for which use the term "high-level properties" ($\mathbf{x}$).

4.6.1 Formal Description

While the rule script is interpreted, the system creates a transformation function STF which maps at each frame the current raw properties of all visible scene objects $\{\mathbf{O}_1, \ldots, \mathbf{O}_N\}$ and the user $\mathbf{u}$ to high-level properties:

$$\{rp(\mathbf{O}_1), \ldots, rp(\mathbf{O}_N), rp(\mathbf{u})\} \xrightarrow{STF} \{\mathbf{x}(\mathbf{O}_1), \ldots, \mathbf{x}(\mathbf{O}_N)\} \tag{4.1}$$

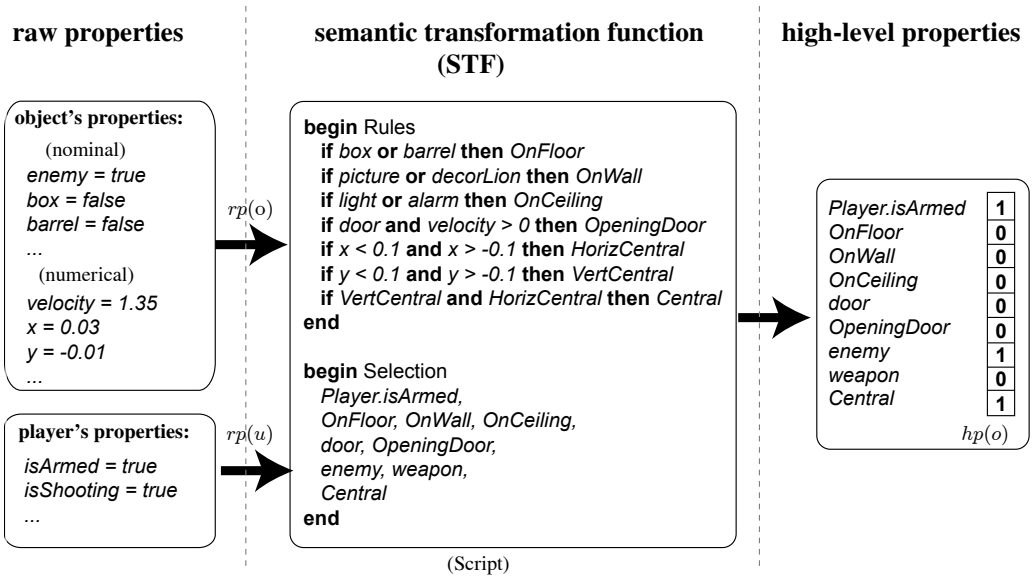


Figure 4.8: An exemplary illustration how the transformation from raw object and user properties to high-level properties could look in a computer game.

While rp represents a set of numerical (e.g., object position) and nominal attributes (e.g.,object category), $\mathbf{x}(\mathbf{O})$ is a vector of booleans which define whether object $\mathbf{O}$ belongs to a particular high-level category ($\mathbf{x} \in \{0,1\}^n$). Thus, the number of high-level categories (n) defines the size of $\mathbf{x}$.

4.6.2 Scripting the STF

In our system, the user can design and modify the transformation function via a scripting language. In Figure 4.8, we illustrated an exemplary script to show how the STF could look in a computer game. The script consists of a set of rules which define a new (high-level) property as the fulfillment of a conditional statement:

if CONDITION STATEMENT **then** PROPERTY DEFINITION

In the conditional statement, raw properties can be combined using standard operators, in particular logical operators (e.g. $\wedge$, $\vee$), arithmetical operators (e.g. $+$, $-$) and relational operators (e.g. $>$, $<$). The condition must evaluate to a Boolean value which is then assigned to a nominal category specified by the right hand side of the rule.

Since the resulting property can be used in conditional statements of subsequent rules, a user can define "intermediate" categories which are useful for defining more complex mappings in the semantic transformation function. However, finally those categories have to be selected which define the high-level property vector $\mathbf{x}$. Thus, there is a second section in the script, the "selection pass", which declares the high-level categories represented by the high-level property

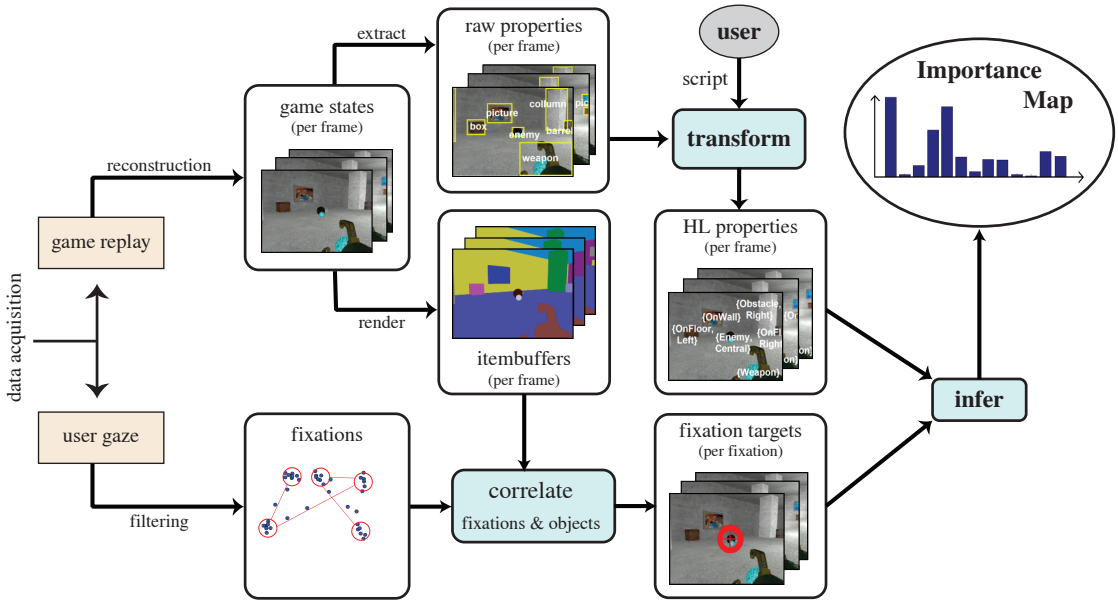


Figure 4.9: An illustration how importance maps are generated with our pipeline.

vector ($\mathbf{x}$). The high-level property vector is represented as vector of booleans each defining the membership to the corresponding high-level category.

4.6.3 Keeping Degrees of Freedom Low

In order to be able to build a reasonable gaze statistic, a SFT has to be specified with a low number of parameters. Thus, rule scripts have to be carefully written such that the number of different high-level categories is low. Due to the possibility that an object can be theoretically a member of all categories, the number of different values the high-level property vector $\mathbf{x}$ can have is, at least in this worst case assumption, exponential with the size of the vector. However, in practice one object should belong to only a few (e.g., 2) high-level categories.

4.7 Generating an Importance Map

The overall process of building an importance map $I(\mathbf{x})$ is illustrated in Figure 4.9. After mapping each object $\mathbf{O}$ to a high-level property vector $\mathbf{x} = STF(rp(\mathbf{O}))$, the next step is to derive a score for the amount of attention given to each value of $\mathbf{x}$ which occurs in the data. We denote this statistic as importance map $I(\mathbf{x})$ which ideally specifies an attention probability for each value of $\mathbf{x}$.

4.7.1 Accumulating Fixations

To derive the importance map, Sundstedt et al. [170] proposed to accumulate fixation times for each semantic property. However, in their study the viewpoint was fixed and the set of

observable objects remained constant. For the general case, we have to assume a viewpoint which is not fixed and the set of objects in the camera's field-of-view may vary considerably from one frame to another. Thus we propose a heuristic normalization strategy accounting for the different amounts of time certain objects are visible to the user. Moreover, we found that giving each fixation the same weight, irrespective to duration, provides more robust statistics with a lower variance.

We calculate $I(\mathbf{x})$ from the number $N_{\mathrm{A}}(\mathbf{x})$ of fixations that fell on objects with that $\mathbf{x}$ and normalize this statistic by the frequency $N_{\mathrm{V}}(\mathbf{x})$ that objects holding $\mathbf{x}$ were visible during a fixation (i.e., the number of fixations they were potential attention targets):

$$I(\mathbf{x}) = \frac{N_{\mathrm{A}}(\mathbf{x})}{N_{\mathrm{V}}(\mathbf{x})} \tag{4.2}$$

The normalization factor $N_{\mathrm{V}}(\mathbf{x})$ corrects for variations in the visibility of different semantic properties. If an object was visible in many frames but has been fixated in only a few of them, the importance value should be low. Conversely, if an object was fixated most of the time it has been visible, the importance value should be high. The maximum importance value is 1 and occurs if a semantic property is fixated in every frame during a fixation it is visible.

4.7.2 Visibility Normalization

Since the output of the gaze-to-object mapping is an attention probability for each visible object which is stored in the file passed to the analysis tool (see Figure 4.6), the total attention frequency $N_{\mathrm{A}}(\mathbf{x})$ can be estimated as expected value by summing attention probabilities for each fixation fix from the set of recorded fixations Fix:

$$N_{\mathrm{A}}(\mathbf{x}) = \sum_{fix_i \in Fix} \int_{begin(fix)}^{end(fix)} \sum_{\mathbf{O}_i \in V^t} \varphi(\mathbf{x}^t(\mathbf{O}_i), \mathbf{x}) P(\mathbf{O}_i|\mathbf{g}^t) dt \tag{4.3}$$

For each fixation we integrate the attention probabilities between begin and end time of the fixation. V^t defines the set of currently objects which are visible during fixation fix. The selection function φ returns 1.0 in case that $\mathbf{x}^t(\mathbf{O}_i)$, which is the temporal state of high-level properties of a particular object instance $\mathbf{O}_i$, equals to $\mathbf{x}$ and 0.0 otherwise. For the attention probability $P(\mathbf{O}_i|\mathbf{g}^t)$, we apply gaze-to-object mapping method for the gaze location $\mathbf{g}^t$ at time t.

The visibility term is computed in a similar way:

$$N_{\mathrm{V}}(\mathbf{x}) = \sum_{fix_i \in Fix} \int_{begin(fix)}^{end(fix)} max\{1.0, \sum_{\mathbf{O}_i \in V^t} \varphi(\mathbf{x}^t(\mathbf{O}_i), \mathbf{x})\} dt \tag{4.4}$$

The clamping function is used to avoid that high-level properties values which occur on more than one object in the current field of view are counted multiple times in the accumulation of the visibility time.

Limitations: It is not possible to define a normalization strategy which perfectly corrects for all latent effects resulting from the variation of the viewpoint and changes in the scene. Hence, this heuristic involves many simplifications, such as the assumption that each semantic property is perceived as one unit of attention. Defining the units of attentional selections which make up the number of alternative targets a user can fixate in a given view of the scene is a hard problem. For future work, it could be useful to investigate strategies which are inspired by models for pre-attentive object detection from vision research. Those could potentially allow to better quantify the amount of visual information a user perceives within the field-of-view.

Another important factor arising from the uniqueness of the game experience of each user, is the variation of the contexts in which a particular object may be seen, which may also influence attention. The way we count and normalize gaze statistics is agnostic to such contextual interactions. Otherwise the dimensionality of the statistic would increase to the size of all possible combinations of semantic properties, and a sufficient density of gaze samples would be impossible to acquire. However, a similar simplification has been made by statisticians who developed ranking models. These models predict a ranking of a large set of elements from a set of preference votes observed for small subsets (e.g., a pair) of all elements. This is done, for instance, when a rank order of many subjects is inferred from pairwise comparison results. To some extent, this resembles our problem, where a user sees a subset of different categories in each frame and performs a "selection" by directing gaze towards one object. Thus similar to prominent ranking models (e.g., Bradley-Terry-Luce model [19]), our approach relies on the Luce Axiom [105], which assumes that the choice probability ratio of two (or more) elements is independent of any other category in the set of currently shown elements.

4.8 Attention Prediction

After an importance map has been constructed from an eye-tracking study as described before, this map can be used at run-time to predict visual attention by generating an *importance value* for each visible object in a frame at a given time. In Figure 4.7, we have illustrated how attention is predicted at run-time by assigning importance values ($I(\mathbf{x})$) to the set of visible objects ($\mathbf{O} \in V$) in the scene. This works by mapping them to higher-level properties ($\mathbf{x}(\mathbf{O})$), which are then used as key values to access the entries of the importance map. This process first requires to determine the visibility of objects and to extract raw properties ($rp(\mathbf{O})$) from the scene graph (as previously described in Section 4.4). In this case the extraction of properties needs to be run in realtime. Thus, instead of using an item buffer in an offline stage, the set of visible objects should be computed using an efficient visibility algorithm [10]. For each object $\mathbf{O}$ that is visible at the current time, raw properties are converted into the corresponding high-level property vector ($\mathbf{x}(\mathbf{O}) = STF(rp(\mathbf{O}))$ using the transformation function (STF). Since high-level properties are encoded as a vector of bits, they can be readily converted to an integer value serving as an index into the importance map to retrieve a corresponding importance score. As discussed previously (in Section 4.6.3), the importance map may have 2^n entries, with n denoting the number of high-level properties being considered. However, there are many semantic properties which can not occur together. For instance, an object cannot be both an enemy and a friend. Hence, the STF should be designed such that only a small subset of the property combinations

defining the Boolean vector $\mathbf{x}$ can occur on the objects in the environment. So actually there are many entries in the importance map which are never accessed. Those are undefined as they do not appear in the environment or due to being nonsensical.

We estimate the final probability $P(A = \mathbf{O}|\mathbf{x}(\mathbf{O}))$ that an object $\mathbf{O}$ equals to the attention target A in a particular frame, given the object's high-level properties, with the entries in importance map I.

$$P(A = \mathbf{O}|\mathbf{x}(\mathbf{O})) = \begin{cases} \frac{1}{Z} I(\mathbf{x}(\mathbf{O})) & \text{if } \mathbf{O} \in V \\ 0, & \text{otherwise} \end{cases} \tag{4.5}$$

Here, we normalize with factor Z, which is to be the sum of all importance scores of objects being visible in the current frame. This normalization also accounts for the effects of the context: in the presence of a very important object (e.g., enemy), the importance score of objects with low importance is suppressed, while it is amplified in the absence of important objects.

4.9 Analysis Toolbox

We implemented the methods described previously in a graphical gaze-analysis tool box. This toolbox includes the following elements:

Library: a library widget designed as a front-end to gaze and stimulus data which the user can load from disk. This is useful so that the operator can select collectively which stimulus data and eye tracked subjects are relevant to his current analysis. The library holds pointers to data, but need not load the data.

Timeline: a timeline widget with different stacked tracks allows the operator to instantiate stimulus and gaze data so that they can be played back in parallel. The timeline offers typical controls of temporal data (e.g. start, stop, etc.) and allows for seeking arbitrary frames within the datasets, enabling intuitive non-sequential access to them.

Views: To visualize the data viewing widgets use, the Timeline tracks to sequentially overlay visual representations of the respective data at the time the timeline's head is positioned or a temporal window around it. For example, fixations can be easily overlaid and played back over the stimulus that produced it.

Script Editor: With the scripting editor an operator can define, execute and debug scripts which transform low level extracted game entity properties into semantic properties.

The combination of these tools into a single graphical user interface enables an operator of the analysis software to potentially gain insight and assist in scripting rules for transforming properties to semantics. This graphical user interface, shown in Figure 4.10, is effectively an Integrated Development Environment (IDE) for studying gaze behavior that not only offers the tools to setup and perform an analysis task, but also provides visual feedback and can potentially leverage the experience and intuition of the operator.

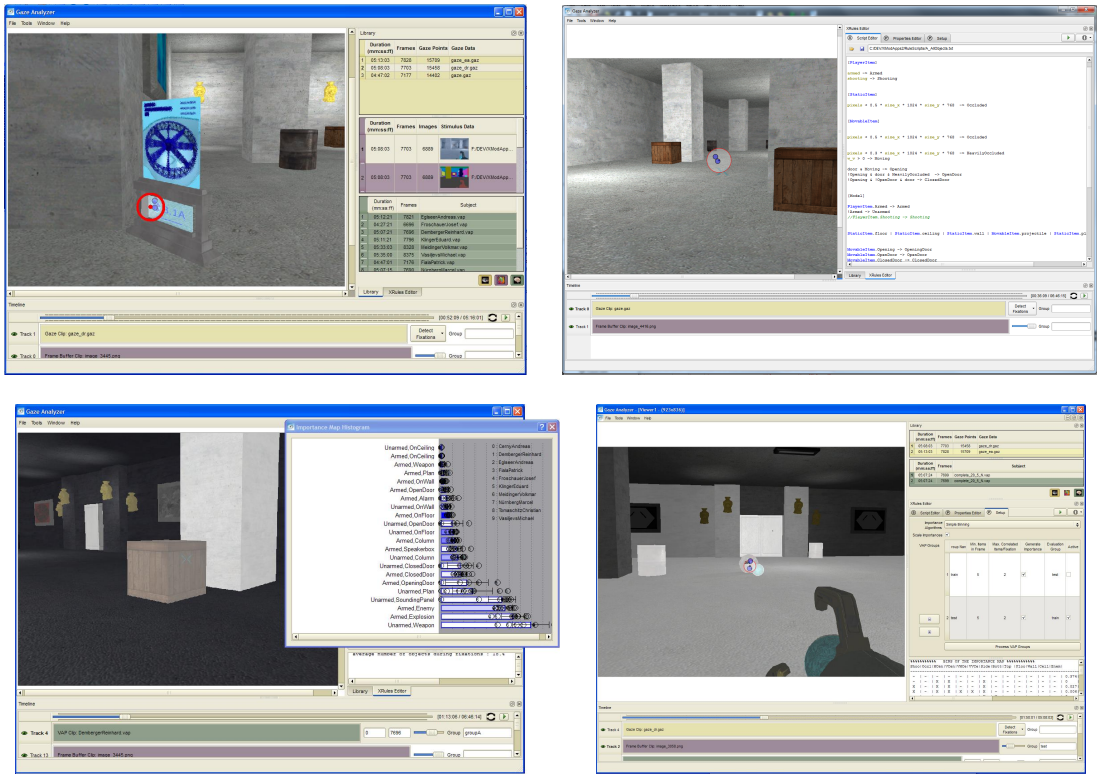


Figure 4.10: Screenshots of our visual analysis tool box *lyzer*. Figure (a) shows the tool used to replay a game for a visual inspection of recorded gaze data. In figure (b) the tool is shown while using the script editor to configure the generation of importance maps. The screen shot (c) shows a dialog visualizing the importance map being generated and in (d) the tool is used to render a visualization of attention predictions.

4.10 Experimental Investigation

In this section we describe a concrete example application of the proposed pipeline. We carried out a pilot study with a simple first-person-shooter (FPS) game developed in our lab. Shooter games are one of the most popular computer game genres, and are thus a reasonable choice for study. The following sections will describe how we designed a level, acquired the data, generated different kinds of importance maps, and evaluated the prediction performance of the corresponding gaze predictors.

4.10.1 Game Design

Using the visual editor tool shown in Figure 4.11, we created a simple level for this eye-tracking experiment.

The game was set in an indoor environment, composed with a network of rooms that contain attacking enemies. Example images of the game can be seen in Figure 4.12. The level had three

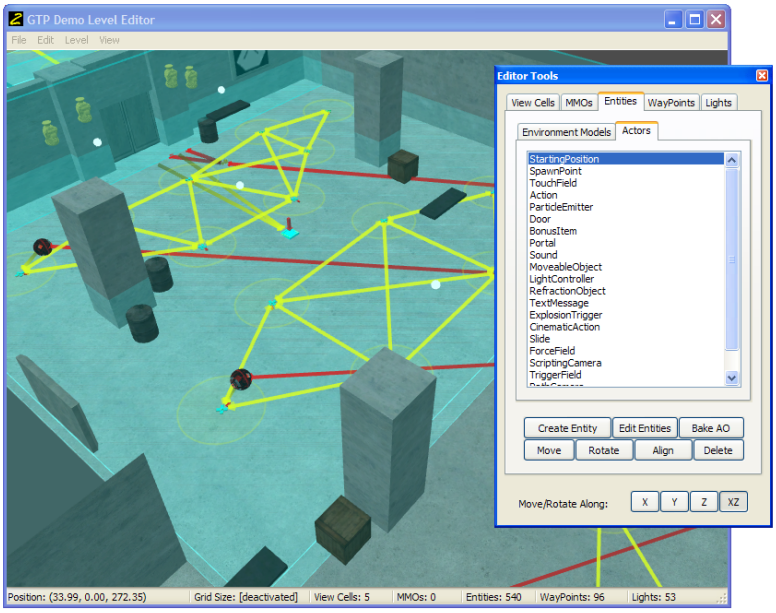


Figure 4.11: The visual editor we used to design the game level.

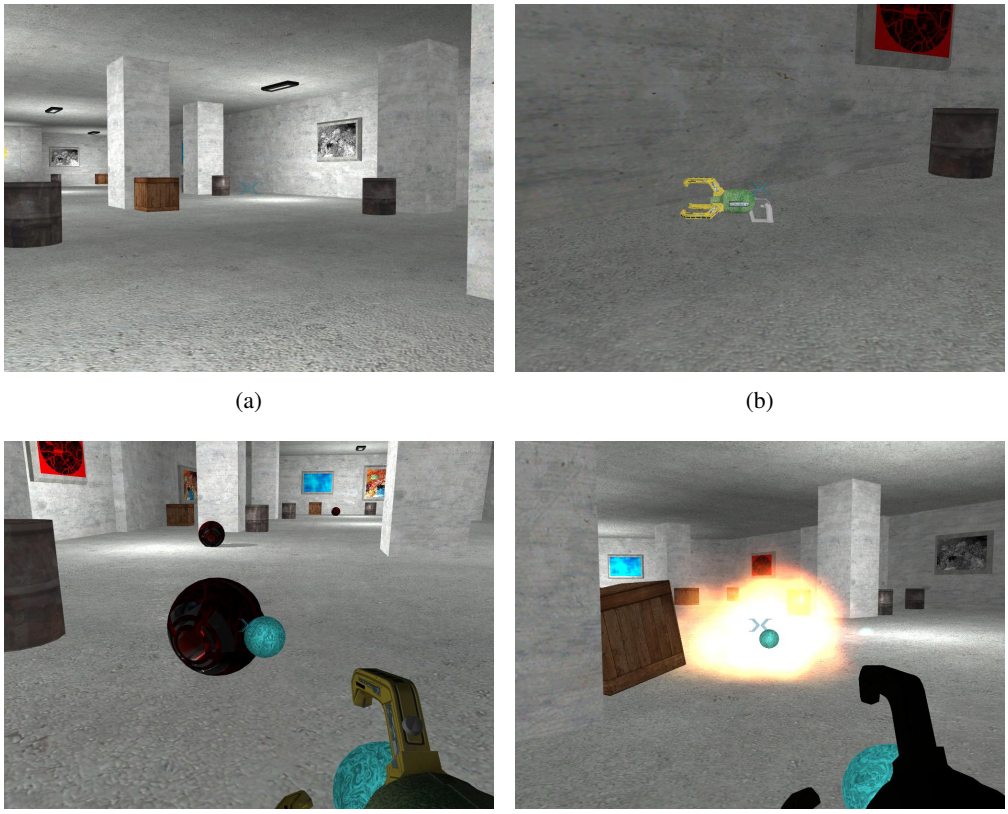
tasks: first, find a weapon (with no enemies yet), then navigate to the "boss room", eliminating any enemies along the way, and finally destroy a crowd of enemies in the boss room to win the game. After a task has been solved, the player was allowed to enter the next section with the next task.

We disabled certain attention-attracting effects and features built into the original game which could potentially bias gaze behavior. For example, our current system does not take into consideration GUI elements, therefore, we disabled all elements of the heads-up display except the cross hairs. The level designer was also instructed to avoid strong variations in the illumination of the environment, as these are not captured by our system.

4.10.2 Setup

We used a Tobii x50 eye-tracker, running at 50 Hz, which was placed in front of the display. The calibration and configuration was carried out according to the manufacturer's instructions. The eye-tracker together with the game's state recording functionality were started and stopped simultaneously, at the beginning and end of each gameplay session. For each session, two data files were stored: a file containing the gaze data provided by the eye-tracker, and another file exported by the game recording all changes of the game's scene graph.

All experiments were run on an Intel Core 2 Duo workstation, clocked at 2.4 GHz with 2GB RAM, and an NVIDIA GeForce 8800 GTX graphics card. This setup was sufficiently powerful to simultaneously run the game, including the recording functionality, at a framerate greater than 50 fps, and also operate the eye-tracker. The display was a commodity LCD display (IBM ThinkVision L200p with a resolution of 1600 x 1200 pixels at 100 dpi), set up at a resolution of 1024 $\times$ 768 pixels, displayed at a scaled down effective viewing window with dimensions



(a) (b)

(c) (d)

Figure 4.12: Screen-shots of our first person shooter game. In the first phase of the game the player explores a large environment **(a)**, and searches for the weapon **(b)**. As soon the player retrieves the weapon, enemies will spawn and the player has to defend himself and simultaneously attack the enemies **(c)**. The enemies are spherical robots that are moving in the environment and shoot at the player, and they explode after termination **(d)**.

of 34 $\times$ 27 cm. We used a smaller viewing window than possible with the monitor in order to minimize inaccuracies in eye-tracking near the outer regions of the wide screen. The setup was located in a dark room with dim lighting to avoid reflections on the screen. A commodity PC stereo sound system was used for the audio output and the loudspeakers were placed to the left and right of the screen.

4.10.3 Eye-tracking Session

Each participant played the level once while the eye-tracker to record her/his gaze. The calibration of the eye-tracker was carried out shortly before the game started and the participants were seated comfortably in the best position as recommended by the user manual of the eye-tracker. To reduce the risk of inaccuracies, the participants were instructed to attempt to hold this po-

sition as well as they could. Before the eye-tracking session, all participants played another distinct level of the game as an introduction. To avoid surprise effects, this level contained every object which appeared later in the eye-tracked level of our study. The players were informed in detail about the tasks and goals of the level before the session.

The time the participants needed to complete the level was between four and six minutes. Between one and two minutes were required to find the weapon, about two to navigate to the boss room and destroy all the ten enemies on this path, and about one and a half minutes were required to fight the final battle against twelve enemies in the boss room.

At the end of the session, a short questionnaire quantified the gaming skills of the participants. We recorded more than 20 participants. Unfortunately, long eye-tracking sessions cause an increasing drift of the eye-tracker's accuracy. Towards the end of many sessions, the precision of the gaze recordings was unacceptable. Some experiments (e.g. [170]) were conducted with a chin-rest to avoid such complications. However, eye movements of participants on a chin-rest are vastly different than when they are free to make both head and eye movements [33]. After a diligent inspection with our gaze visualization tool, it turned out that only 10 gaze datasets had a reliable accuracy. The respective participants were all male, had normal or corrected to normal vision and their FPS gaming skills were good to very good.

4.10.3 Exclusion of Background Objects

In our experiment we obtained high attention scores for the floor and the walls, but almost no attention was paid to the ceiling of the corridors. However, the background are large planes and are not really perceived as objects. Hence our gaze-analysis pipeline is not necessarily appropriate to infer gaze statistics for background objects. The main problem was that we used in this experiment a simple gaze-to-object mapping technique (Fovea Splatting) which is biased towards large objects. (Later we investigated better solutions to this problem, which are presented in Chapter 3.) With this technique, background objects are hit very often simply because of their large extent through the whole level. This introduces a bias towards background objects, which is unwanted since they rarely contain important features. While an object size weighted normalization could remove this bias, this would not work well for foreground objects.

For the results presented here, we thus excluded background objects from set of visible objects V and removed frames where background objects had the highest importance in the gaze to object mapping result (in equations 4.2 and 4.4). To this end, we used a simple rule in the scripting interface. This considerably improved the performance when gaze statistic are used as attention predictor. More appropriate treatments for background objects that could be considered in future work are a spatial subdivision of large objects and the extraction of image space features, such as corners, vanishing lines, shadows or highlights.

4.11 Results

In the following we present some illustrative examples of particular importance maps generated from the data acquired in our study. To this end, we present importance map derived using spatial object properties, such as size and eccentricity, before we continue with investigating

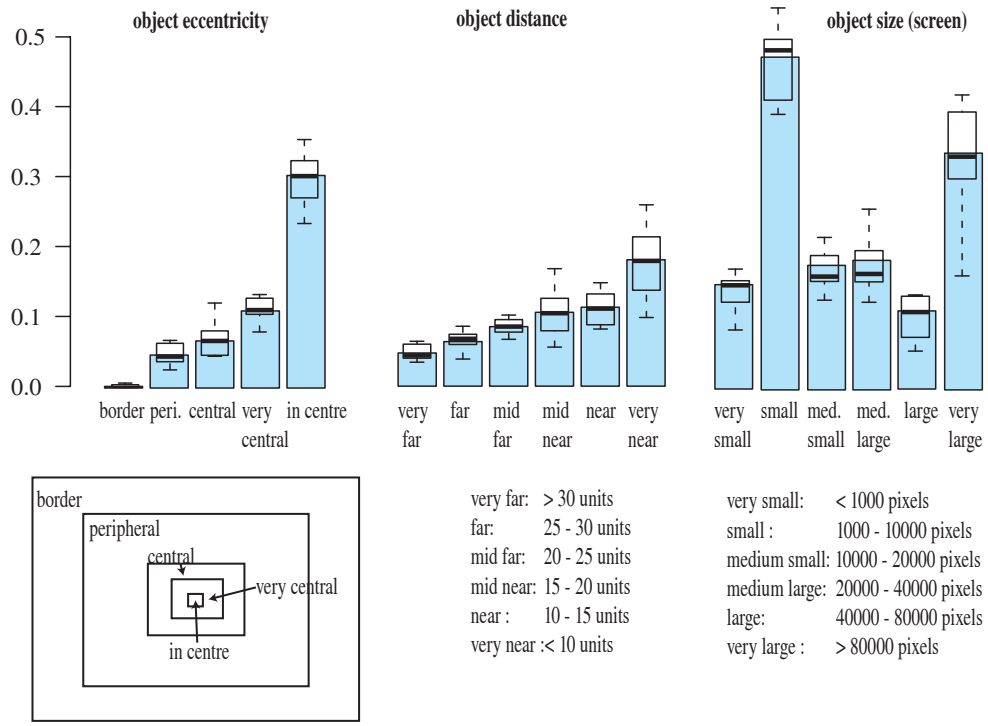


Figure 4.13: Importance maps obtained with a STF which assigns "spatial semantics".

how attention could be analyzed making use of semantic properties. Finally, we will evaluate how the importance maps obtained can be used to predict visual attention.

4.11.1. Importance Map with Spatial Properties

When we inspected the gaze behavior with our gaze visualization tool, the most obvious observation was that most fixations were close to the center of the screen. Similar results were reported by Kenny et al [86] from a study on gaze behavior in FPS games. We thus authored a script to learn an importance map which captures this behavior. Five rules were specified which intersect the object bounding boxes with 5 squares, which are illustrated at the bottom left-hand side of Figure 4.13. Each object is assigned to the lowest eccentricity level its' bounding box intersects with. With these rules, we generate five levels of object eccentricity and count fixations for each level. The resulting importance map is shown in the left plot in Figure 4.13.

The blue bars in this and other importance map visualizations illustrate the average fixation counts (normalized with the visibility term) collected from the gaze data of all participants. Beside the average importance values we were also interested in the agreement between different subjects. How well the importance values resemble for different participants can be evaluated by computing "individual" importance values from gaze data of each participant and analyze the resulting distribution which we illustrated with boxplots, which show the 5%,25%,50%,75%

and 95% percentiles of the distribution of the individual importance values.

The eccentricity importance map has a high peak for the inner eccentricity bin "in center", and rapidly decreases towards the periphery. The most probable explanations for this are that (a) the cross hair of the weapon is displayed at the center of the screen, and (b) that a user controls the camera such that objects are brought towards the center of the field-of-view.

Besides eccentricity we tested whether object size and distance can be used to predict visual attention. Similar to the script we used for object eccentricity, we defined a set of discrete bins to derive a gaze statistic for object size and distance. The results, which are shown in Figure 4.13, show that fixation counts increase with object proximity. One explanation could be that a users prefers to attend objects which are near and hence more important or sooner relevant for navigation and interaction tasks. Another possible explanation could be that objects which are closer appear with a greater size on the display due to the perspective projection. However, when considering object size alone, fixation counts do not monotonically increase with this property. While there is one peak for "very large" objects (>8000 pixels), which could be attended more often due to the stochastic advantage of objects covering a significant fraction of the screen and thus having a-priori an increased probability to co-occur at the same location as a fixation, there is no linear correlation between object size and attention for objects which are not very large. Instead, we find a second mode in the histogram for "small" objects covering between 1000-10000 pixels on the screen. One explanation for this peak could be that to a certain extent, the object size encodes the object category and the size range between 1000 and 10000 pixels corresponds to the size in which task-relevant objects (e.g. enemies) are seen most of the time.

4.11.2. Importance Map with Semantics

In order to demonstrate that our approach can generate importance maps based on more complex properties, we created a script to infer importance values according to *semantic object properties*. We combined this with one relevant property of the player, notably whether he is "armed" or "unarmed". With this combination, differences in the gaze distributions according to the current task of the player, which is to search the weapon while being "unarmed" and to shoot enemies as soon he possesses the weapon, should be captured. In Figure 4.14, we depicted the values according to the high-level properties (i.e., object category + player state) of the respective importance map.

The importance map we derived shows that only few objects are outstanding attention attractors, notably explosions, the enemy, the weapon (as long the player is searching for it), and a panel emitting sound. The weapon has high importance before the player retrieves it, but as soon the player carries the weapon, its importance drops significantly (see scores according to the states "Armed.Weapon" and "Unarmed.Weapon" in Figure 4.14). The sound-emitting panel is a conspicuous object in the scene. It has a screen displaying a rotating wheel and emits a sound which is congruent to the animation. This kind of crossmodal cue possibly causes a high attentional attraction. An interesting observation is that the importance of the overview map changes considerably between both tasks. When the player searches the weapon he seems to have time to pay more attention to maps and objects suspended on the walls, while as soon as he is armed and enemies are attacking, he has different priorities.

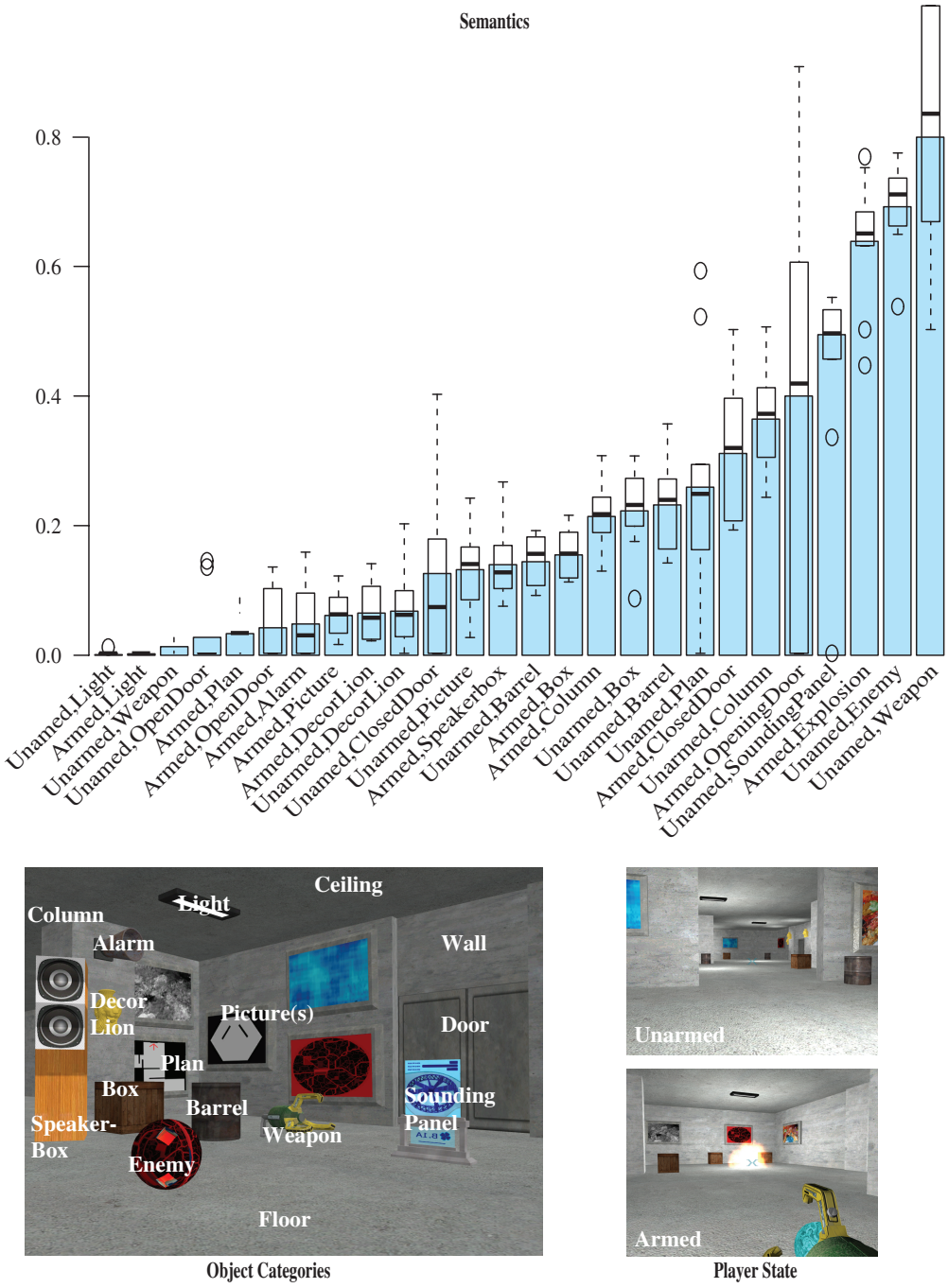


Figure 4.14: Object categories in the game level and their fixation statistics.

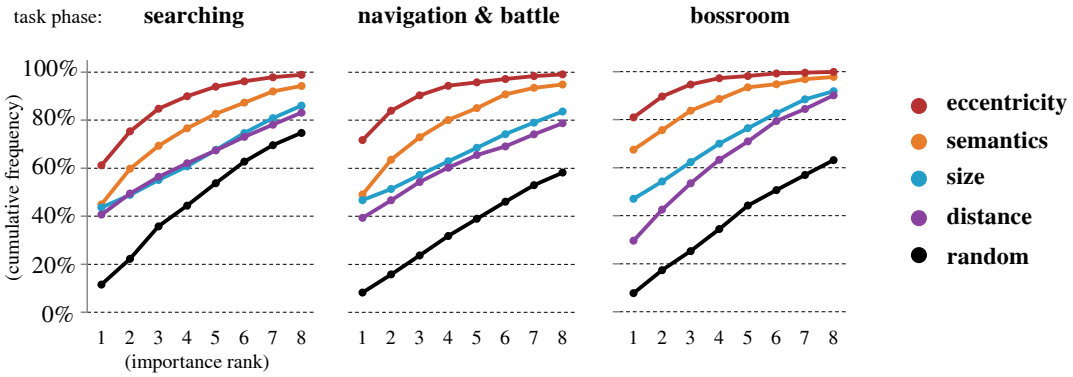


Figure 4.15: Cumulative distribution of the rank transformed probabilities predicted for fixated objects.

4.11.3. Attention Prediction

To investigate to which extent our empirical approach to visual attention allows to predict gaze of a user, we compare the predictions of an importance map with the fixated objects observed by the eye tracker. We perform this analysis by dividing the game session of each participant in three phases according to the three game tasks: *Searching* for the weapon, *navigation and battle* on the way to the boss room, and finally the final game battle in the *boss room*. The corresponding three tasks show increasing task intensity, which has an impact on the prediction performance.

To evaluate how well the importance map described previously can predict visual attention, we counted how often a fixated object was predicted to have the 1^{st}, 2^{nd}, ..., or 8^{th} highest importance among the objects visible during the fixation. From these fixation counts we build a cumulative distribution over the importance ranks, which is illustrated in Figure 4.15. To have a baseline for comparison, results derived with a random predictor were also added (black graph).

There is clear evidence for all conditions that all importance maps predict attention better than chance. Eccentricity, which predicts the fixated object as most important in more than 60% of all fixations, turns out as the most effective and reliable property to predict visual attention in a first person shooter game. A reason for this is that in first person games the user actively controls the camera. If an important object appears on screen, the player tends to orient the camera quickly towards this object and the eccentricity-based predictor will provide a good prediction. This reveals that the way a user orients the camera closely resembles to eye-movements. Thus in first person view games the center of the view frustum can used to observe attention without an eye-tracker. However, for games that do not use user controlled first-person view cameras, this trivial way to predict attention would be less accurate.

Besides eccentricity, also size and distance predict the fixated object as most important in 35%-45% of all cases. But in task-intensive situations (e.g. "bossroom"-phase), size and distance are clearly less suited to predict visual attention than predictors making use of semantic object information.

Most important and interesting are the results we obtained for predictors based on seman-

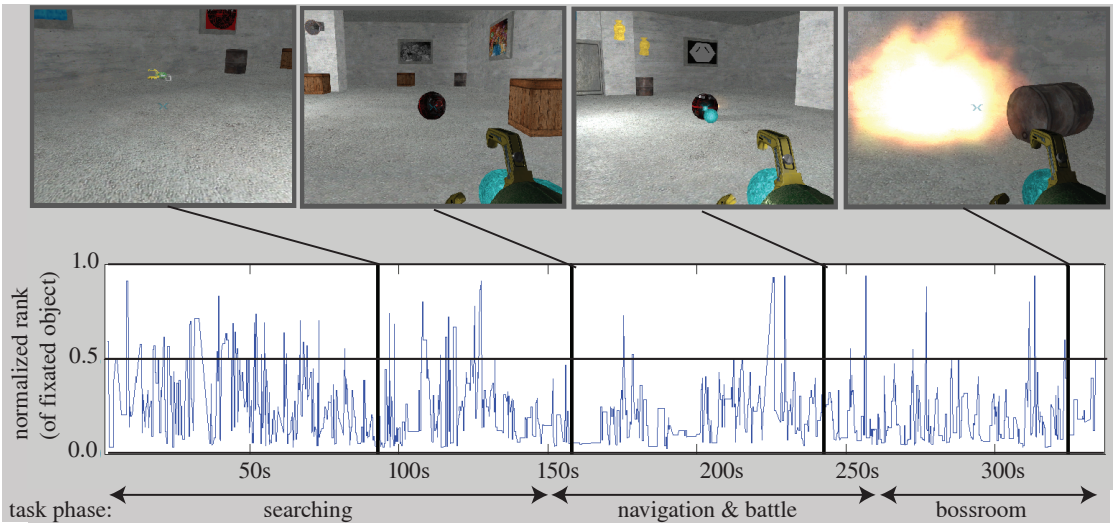


Figure 4.16: Performance of attention prediction over time (semantic). Since there is a different number of visible objects in each frame, we normalized the rank of the fixated objects by the number of (currently) visible objects. The line at 0.5 corresponds to the expected value for a random prediction.

tics. The cumulative importance rank distribution obtained using a semantic importance map as predictor is clearly better than a random predictor or predictors based on object size or distance. Even in the first phase of the game which was least task-intensive and players most time explored the level causing a free-viewing-like gaze behavior, semantics predict the fixated object as being most important in 43% of all cases. This rate increases to 70% in task intensive situation as occurring in the bossroom where the player is attacked by many enemies. To provide some insights how attention prediction performs over time, Figure 4.16 shows a plot of the importance rank of the fixated object as a function of time.

4.12 Discussion

Our results suggest that we can already predict attention well, but we need to point out that we did not address bottom-up mechanisms affecting the control of attention. We believe that the two approaches are complementary and can be combined in future work. For example, while saliency maps are good at predicting what could be the next gaze target, our approach more accounts for *sustained* attention.

Further, while image space approaches work on arbitrary images sequences, this work focused on object-based importance maps, which requires an application where the internal object structure is accessible and item buffers can be created. However, the interface to the analysis tools to be integrated is light weight and it should be easy to adapt host applications to this requirement. Also note that image space approaches require object recognition and identification.

In our experiment, we excluded background objects from the analysis and used objects with

a clear semantic category and a clear geometrical outline. However, environments in commercial games frequently comprise more difficult content, e.g. large environment models or vegetations. In future work, solutions need to be investigated which allow decomposing all elements of a scene adequately, so that gaze can be analyzed according to the key features of major impact. Particular extensions we are considering, are hierarchical decomposition of difficult objects, like for example trees or houses, and screen-space approaches to subdivide large models or background into regions in such a manner that features with a different response on visual attention can be spatially separated.

4.13 Future work

There are many ways in which the basic pipeline presented in this paper can be improved. We propose the following directions:

The first is to test the approach for more complex game settings and environments, imposing challenging tasks and missions, comparable to commercial games. Most importantly, games or virtual environments that do not use a mouse-controlled first-person camera should be investigated to better single out the potential of non-trivial attention prediction methods.

Second, investigating certain extensions to improve the performance of the predictors. This includes, on the one hand, accounting also for bottom-up mechanisms and investigating how we can effectively combine the image-space-oriented saliency maps and the object-space-oriented importance maps. On the other hand, it is also important to account for the entire multimodal stimulus presented to the player of an action game. Notably, including the impact of sound sources in 3D space and elements in the head-up-display (HUD) of games. Due to the very general design, our system is capable to be extended with items for sound sources and HUD elements.

Finally, we think that many of the tools we proposed, in particular computing gaze distributions according to object semantics in interactive 3D environments, may find great application in the study of human behavior in virtual reality.

CHAPTER 5

Attention-Aware Rendering: A Pilot Study on Gaze-Controlled Stereoscopic Applications

Every solution breeds new problems

Arthur Bloch

Most previous work on attention-aware rendering has focused on computational acceleration to make costly rendering techniques more applicable for realtime graphics. While computational acceleration is still an important issue, particularly for saving energy on mobile devices or to display complex scenes on immersive high definition displays, computational efficiency is not the only problem that can be solved with attention-aware rendering. Attention inference and prediction can be also of tremendous utility to improve depth perception and realism in computer graphics. In this chapter we present an example application for attention-aware rendering. We chose gaze-controlled 3D stereo as example and attempted to explore potential benefits and issues that may arise in such an application. While the work described in the two previous chapters may serve as basis for future applications that are designed to work with complex virtual environments, we studied in this work how gaze-controlled stereo would work in the optimal case where the attended object can be determined unambiguously in real time. Moreover, we explored in this pilot study the most straight-forward solution of configuring stereo rendering parameters via gaze-control.

The work presented in this chapter is based on the paper "The Effects of Fast Disparity Adjustments in Gaze-Controlled Stereoscopic Applications" [5], which was published in the proceedings of the ACM Symposium on Eye Tracking Research and Application 2014.

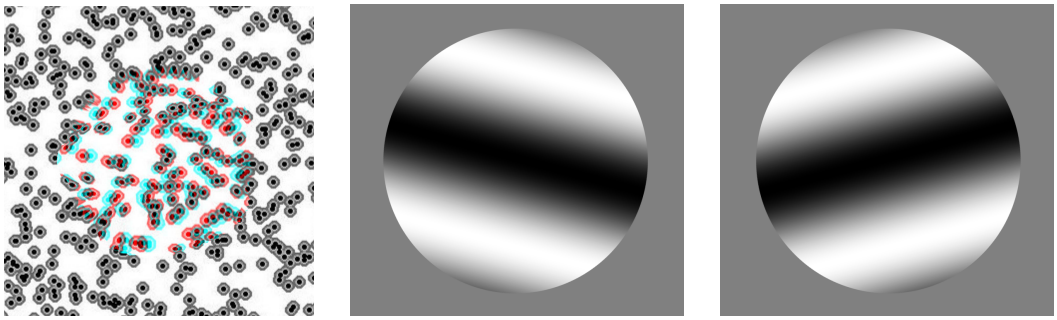
5.1 Introduction

With the emergence of affordable 3D displays, stereoscopy is becoming a commodity in the film and games industries, virtual reality, visualization, etc. However, despite the technological advancements in 3D display technologies, users often report discomfort and fatigue even after brief exposures to stereoscopic content. One of the main reasons is known to be the conflict between vergence and accommodation that is caused by 3D displays [70, 96], since the eyes naturally turn (vergence) at the distance the virtual stimuli is presented at, but the eyes' focus mechanism (accommodation) remain on the surface of the display.

The most effective way to increase stereo comfort in stereo displays is to control binocular disparities such that they fall in a disparity range where stereo viewing is comfortable. This can be achieved by either controlling stereo camera parameters (i.e., focal distance or interaxial separation) [61, 125, 179], or by manipulating binocular disparities in stereo images [98, 139, 185]. However, manipulating disparities often leads to a significant reduction of depth quality, for example in the form of depth flattening, known also as the cardboarding effect. Another way to decrease discomfort is to use depth-of-field (DOF) blurring [11, 12, 45, 101, 171], which simulates the foveal sharpness of the image at the point of focus and blurring observed in peripheral vision. However, applying DOF blurring to accurately simulate the perceived retinal image is difficult to achieve in highly dynamic applications and often causes loss of visual detail.

A promising solution is offered by the use of eye-tracking data, which is used to dynamically adjust the stereo focal plane to coincide with the user's intended focal depth. This is achieved by mapping gaze deployed on a display to the underlying geometric objects of a 3D scene [46, 173]. Fisker et al. [54] performed an informal experiment to evaluate the level of comfort these methods offer to the users. Their preliminary results indicate they can have a positive effect, but a formal experiment and validation is required.

In this work, we performed such a formal experiment in which we investigate the possibility of mitigating visual discomfort by dynamically adjusting the stereo parameters of a scene using gaze data. We use an eye tracker to determine the depth at which the user's eyes converge and manipulate the stereo parameters to bring it into focus by setting the plane of zero disparities at that depth. To measure the level of discomfort, we measured binocular fusion times, which is an *objective* metric. In particular, we measure fusion times of random dot stereograms in a scene with a three-dimensional arrangement of objects, which resembles a natural scene with multiple depth layers where a user can freely direct gaze from one object to another. We utilize a QUEST [186] procedure to determine the amount of time the user would require to fuse a random dot stereogram (RDS) showing one of two possible orientations of a wave pattern, shown in Figure 5.1. After correct responses the QUEST reduce the display time of the RDS, whereas wrong responses lead to an increase of the time the RDS is shown. Effectively, the QUEST is searching for the minimum amount of time a user requires to fuse the stereogram, which we refer to as *fusion time*. This process is illustrated in Figure 5.2 where we plotted the display times proposed by the QUEST until converging to the fusion time threshold. We found that gaze-controlled manipulation of the plane of zero parallax can lower fusion times for large disparities. In addition, a small overhead in fusion time was observed after adjusting the plane of zero parallax to coincide with disparities that initially fell within the stereo viewing comfort zone



(a) RDS (anaglyph) (b) right tilted (c) left tilted

Figure 5.1: The random dot stereograms shown by the QUESTs for a controlled time: (a) shows an anaglyph rendering. (b) and (c) show a visualization of the two different sine wave patterns participants had to distinguish.

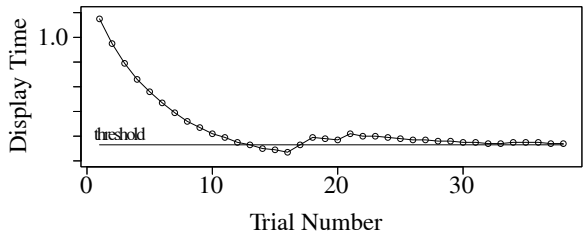


Figure 5.2: An illustration of the QUEST procedure converging towards the fusion time threshold being measured.

of subjects. Since the comfort zone differs among subjects, this means that gaze-controlled disparity adjustment should be applied in a personalized manner and ideally performed only at the extremities or outside the comfort zone of subjects.

The contributions of this work are: (a) an *objective* methodology to evaluate the benefits of visual comfort-optimizing strategies for 3D stereo displays, and (b) experimental evidence that gaze-controlled dynamic adjustment of stereo disparities improves stereo fusion of nearby objects (large disparities), while it is acceptable for objects at other distances.

5.2 Related Work

5.2.1 Stereo Viewing Comfort

The main factors [70, 96, 129, 172] causing viewing discomfort in stereoscopic displays are **(a)** *dichoptic errors*, due to stereoscopic distortions, mismatch of the stereoscopic windows or crosstalk, **(b)** *fast object motion*, due to difficulties of the human visual system (HVS) to adapt to changing viewing conditions, and **(c)** uncomfortable disparity ranges due to excessive binocular

parallax, or due to a mismatch of vergence and accommodation (V/A-conflict). While factors of category (a) are a matter of hardware, processing or bandwidth limitations, the other sources of discomfort depend on *content* and *rendering* and could possibly be mitigated by adjusting stereo configuration parameters. To find the best parameters, a function which predicts viewing comfort is required, which is obtained by taking measurements in a perceptual study. Discomfort is usually predicted as a function of disparity, or as a bivariate function of vergence and focal distance. Measurements can be taken with *objective* and *subjective* observation methods. Assuming correlations between performance in binocular perception and comfort, objective measurements are taken by determining speed or acuity in the detection of binocular corrugations (e.g. inward versus outward curvature) in random dot stereo-grams [66,89,90]. For subjective measurements, users are usually asked to rate the discomfort they experienced on a Likert scale [90]. Due to methodological limitations and simplicity, comfort-optimizing stereo displays are usually evaluated with subjective methods only. Responses can be collected either by *rating* methods, where a user issues a score on a Likert scale [11, 12, 84, 94, 104, 163], or with the method of pairwise comparison [94, 101]. An *objective* way to observe discomfort was used by [31], who counted the eye-blink rate of users.

In this work we will predominantly focus on an *objective* evaluation. Hence we will measure performance in the fusion of RDS, similar to [66, 90]. However, in contrast to previous work which measured fusion times for a stimulus showing a single disk with a RDS, we will place the RDS on objects in a scene arranged in three dimensions. In addition to that, we handed out a rating scale where participants cast a subjective judgment on stereo comfort and stereo depth.

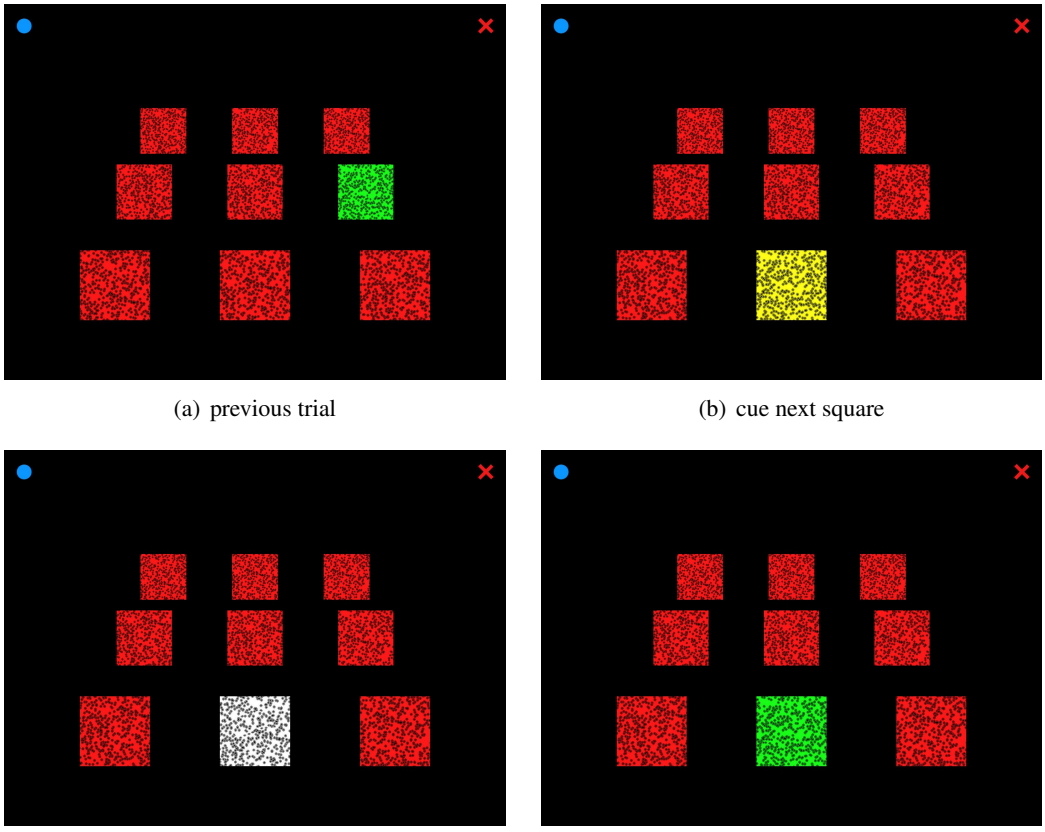
5.2.2 Eye Tracking in 3D

Hillaire et al. [64], and very recently Mantiuk et al. [109], proposed to use an eye-tracker to control DOF simulations in interactive applications. Eye tracking was also used in applications attempting to reduce stereo viewing discomfort by DOF-blurring [12, 101, 171]. In all cases, gaze data was mapped to scene content (e.g., objects [109]) in order to determine the focal distance. Alternatively, for applications which do not have access to a 3D representation of the stimulus (e.g., movies), [47] and [134] proposed to utilize binocular gaze data to determine a depth coordinate for each gaze point.

However, a reliable and accurate determination of a user's gaze depth is still an open problem. To avoid this problem in our experiment we used a simplified scene which was inspired by [47]. In this scene we placed 9 squares such that the current gaze depth can be robustly determined from the depth buffer also in the presence of eye tracker errors and a participant's inaccurate gaze orienting behavior.

5.3 Experiment

We performed an experiment to evaluate the effects of dynamic adjustment of disparities in 3D applications using eye-tracking data. In particular, we consider binocular fusion times as an objective indicator of visual comfort and asked participants to focus at objects lying at different depths while stereo viewing a 3D scene. We measured fusion times of stereoscopic stimuli,



(a) previous trial (b) cue next square

(c) show RDS (d) response

Figure 5.3: One QUEST trial: after responding in the previous trial, the next square is cued yellow, as soon the eye tracker senses a user's gaze near the cued square the RDS is shown on white for a controlled time after which the square turns green with random dots.

where binocular disparities were manipulated dynamically, using gaze data recorded in real-time. Disparity was automatically adjusted by modifying stereo parameters in order to bring the currently attended object into focus, which equates to setting zero parallax between the two stereo views at that point. We denote this condition as *dynamic stereo* (**DS**) and compare fusion times to a control condition in which predefined stereo disparities remain constant, which we refer to as *static stereo* (**SS**).

5.3.1 Participants

We recruited 38 subjects (15 female) with normal or corrected to normal vision, aged between 18 and 40 years (mean=26.3, stdev=5.4), all of which stated that they were stereoacute and not color blind. Each participant was pre-screened during a training block of trials to objectively measure their ability to achieve stereopsis. Through this procedure, we identified 2 stereo-blind participants, who were excluded.

5.3.2 Stimulus

To sample a variety of factors that might have an influence on fusion time, such as *position* or *disparity*, the 3D scene we used for the experiment consisted of 9 square planes, placed in three rows and columns (3×3) on a black background, as shown in Figure 5.3. Each row of squares was positioned at a different distance/depth, with the bottom row being the closest (**N**EAR), the middle row being at an intermediate distance (**M**ID), and the top row being the farthest (**F**AR). In the static stereo condition (**SS**), the **M**ID row was at the same distance as that of the focal plane (i.e. the plane of zero parallax). Consequently, the **F**AR row had positive disparities ($d = 0.7°$) and appeared to be $14cm$ behind the focal plane, while the **N**EAR row had negative disparities of $d = -1.2°$ and appeared $14cm$ in front of the focal plane.

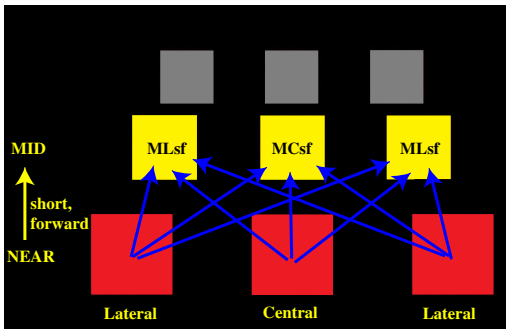
In their initial state, all squares were colored red and were textured with a pattern reminiscent of those used in random dot stereograms. However, since the same pattern was rendered for both stereo views, it provided no depth cues. The color of one of the squares was dynamically modified to attract attention during the experiment, while as soon a fixation was detected near the cued target the texture of that square was replaced by another texture containing a circular random dot stereogram (RDS) as shown in Figure 5.1. This RDS that would elicit binocular perception of a sine wave with a wavelength of $2°$ and an amplitude of $0.65°$ which was either tilted by $+15°$ or $-15°$ from the horizontal axis (see Figure 5.1). We also ensured that the rendered stereograms appeared with the same size ($radius = 1.9°$), the same resolution and the same disparity amplitude on all squares by avoiding distortions of the stereogram patterns due to perspective projection.

5.3.3 Task

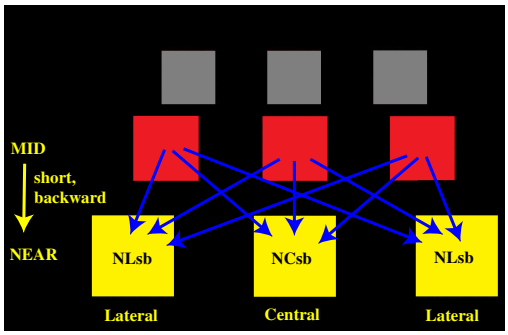
Participants were instructed to focus their gaze on one dynamically chosen square with a yellow color cue. As soon as the eye tracker could detect that a participant initiated a fixation (with two gaze point samples recorded at 50Hz) on the cued square, its color changed to white, and simultaneously the texture was replaced with a random dot stereogram. In stereo, this stereogram induced the perception of a 3D sine wave with a left- or right-winded slope on the square. The participant was exposed to the random dot stereogram (where the random dots changed in each QUEST) for a short period of time, which was controlled by a QUEST procedure, and then switched back to a different random dot pattern that induced no depth perception. Using the left and right cursor keys of a keyboard, participants had to respond whether the sine wave had left- or right-winded tilt. After the response, the next square cued the participant's attention by switching to yellow. The experiment ended when all of the running QUEST procedures were completed with a minimum of 3 turns.

5.3.4 QUEST Conditions

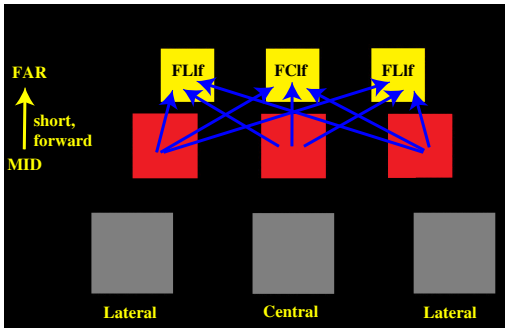
We defined 12 QUEST procedures to determine the effects of a variety of factors that may play a role in fusion times, such as position, depth, and eye movements. We illustrated the conditions of each of the 12 QUESTs in Figure 5.4. For each of the three rows (**N**, **M** and **F**) of squares, we make a distinction between squares located in the center (**C**entral) and the squares on the left- and



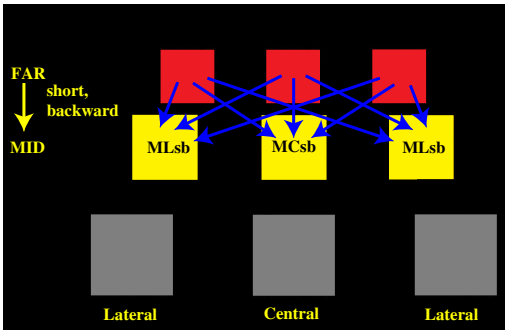
(a) NEAR to MID



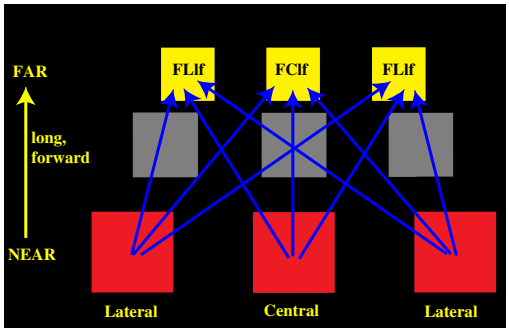
(b) MID to NEAR



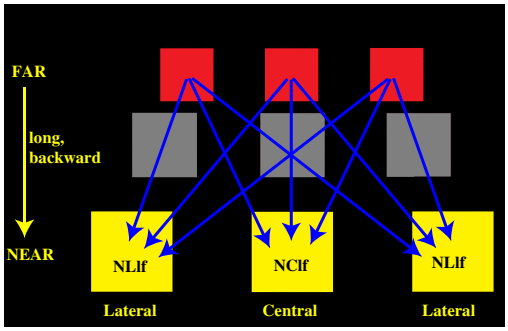
(c) MID to FAR



(d) FAR to MID



(e) NEAR to FAR



(f) FAR to NEAR

Figure 5.4: Illustration of the QUEST conditions according to the type of the preceding saccade

right-hand side of the row (**L**ateral). We differentiate eye movements between two consecutively attended squares. In particular, we distinguish **s**hort binocular saccades between two consecutive rows and **l**ong binocular saccades between the last and first row. We also distinguish **b**ackward saccades from far to near squares, and **f**orward saccades from near to far squares. For each possible combination of these categories, we use a different QUEST instance to measure the respective fusion time. Note that 50% of the 24 theoretically possible combinations are not possible in our scene, e.g., a *backward* saccade to the *near* row is an invalid case. Thus there are 12 QUESTs in total (**NCsb, NClb, NLsb, NLlb, MCsf, MCsb, MLsf, MLsb, FClf, FCsf, FLsf** and **FLlf**). All these QUESTs were run in parallel and scheduled in randomized order. In each trial, a QUEST was started after sensing two gaze points near the cued square plane. Since the mechanism to launch the QUEST was equal in both of the **SS** and **DS** conditions, the latency required to detect a fixation should not affect the measurement process itself.

We used the Matlab Psycho Toolbox to control the QUESTs. The threshold guess was set to $t_G = 1000ms$ (which is used as initial test time), the standard deviation guess to $t_{GSd} = 1000ms$, and the probability threshold to $p_T = 82\%$. The gamma parameter was set to $\gamma = 50\%$ and the delta parameter was $\delta = 0.01$. As beta parameter we used $\beta = 5.4$, which was optimized using data obtained by one of the authors performing a beta analysis over 800 trials of the experiment run in condition **SS**.

5.3.5 Procedure

Each participant was assigned to perform two blocks of multiple trials for each of the two conditions (**SS** and **DS**). The experimental condition of each block was randomly selected to counter balance potential effects of ordering. The duration of each block was 20 minutes on average since it depended on the convergence time of all QUESTs, contributing to a total experiment duration of 40 minutes in average for both blocks. Each of the 12 QUESTs converged, in average, after 65 trials summing up to approximately 800 trials in total for each block and participant. To prevent eye-strain and fatigue the procedure was paused automatically every 5 minutes for a fixed 1-minute break before continuing the block. Prior to the two main blocks, a short block of 27 trials was used as a training block for the participants to familiarize themselves with the task. During training, a participant viewed the RDS on each of the 9 squares in the scene for one second and had to identify the orientation of the sine-wave tilt. This procedure was repeated three times in the training block.

5.3.6 Subjective Measurements

Besides the fusion times measured with the QUEST procedures, we also took *subjective* measurements according to the ITU-Recommendation for subjective methods for the assessment of stereoscopic 3DTV systems [2]. This recommends a 5-level rating scale where the user scores his experience with respect to *picture quality*, *depth quality* and *visual comfort*. We told participants that the levels serve an orientation and the ratings can be done on a continuous scale. After each block, we asked participants to rate depth quality and visual comfort only, since our scene had no characteristics that could be assessed for a reliable picture quality rating.

5.3.7 Setup and Configuration

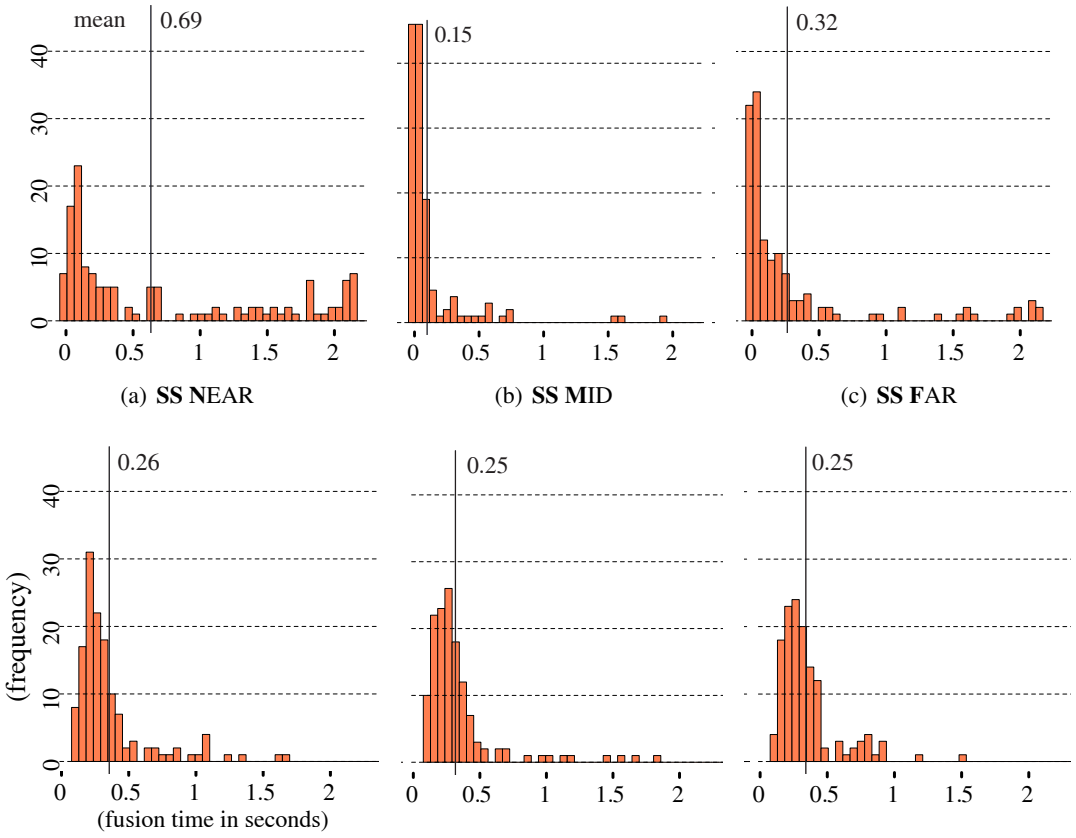
The hardware setup in use was a See Real Technology Cn 201.05 Autostereoscopic 20" monitor which uses a vertical interlace technique to display stereo image pairs. The resolution of each image was 800×1200 pixels, but due to the interlace technique they were shown with a $4:3$ ratio. To avoid reflections in the display, the experiment was carried out in a room without daylight under dim light conditions. We used a Tobii x50 eye-tracker, which was placed in front of the display. Due to the display's refresh rate of $60Hz$, the eye tracker's sampling rate of $50Hz$ and the eye tracker's latency of $35ms$, stereo disparity adjustments were performed with a delay[1] ranging, depending on the sampling phase of the eye-tracker, between a minimum of $72ms$ (best case) and a maximum of $92ms$ (worst case). Configuration and a 5-point calibration was carried out for each participant before the training block using the software provided by the manufacturer. In addition, we used a chin-rest placed 67cm away from the display's center. Besides allowing faster and more accurate gaze sampling, the chin-rest's main purpose was to prevent participants from moving their eyes away from the auto-stereo display's optimal 3D view point.

5.4 Results

5.4.1 Distribution of Fusion Times

We use descriptive statistics to analyze the main effect of the two conditions. We computed the fusion times' histograms shown in Figure 5.5. Since the effects vary considerably with the distance of the focused row of squares, we split the data for all rows. Nevertheless, the distribution of the fusion time in the **DS** condition is very similar for each row and also consistently of lower variance. To analyze the overall effect of condition and its significance, we performed a paired Wilcoxon test, a non-parametric alternative to the Student's t-test, which could not be used for our data due to violating the assumptions of normality and equality of variance. The test results are shown in Table 5.1.

Concerning the effect of condition on fusion time, we found a highly significant positive effect for **DS** only in the **N**EAR row. The overall effect for the **M**ID row was, however, a significant increase of fusion times due to dynamic switching of the focal plane. Comparing the results of both conditions in the **M**ID row, where we had in both conditions zero disparity, reveals that focal plane switching causes an overhead in fusion time which was in average $94ms$ (mean difference between **DS** and **SS**), what lies close to the average delay ($82ms$) in disparity adjustments caused by gaze sensing and the display refresh rate. To some extent, the overhead explains why the effect of **DS** was more on the negative side for the **N**EAR row, where many users had relatively short fusion times in the **SS** condition. However, the benefits of **DS** seem to vary considerably among participants, as discussed next.



(a) **SS NEAR** (b) **SS MID** (c) **SS FAR**

(d) **DS NEAR** (e) **DS MID** (f) **DS FAR**

Figure 5.5: Distribution of fusion times sorted by stereo condition (rows) and depth plane (columns). For each histogram, the abscissa denotes fusion time in seconds and the ordinate shows the occurrence frequency. Black lines mark the means.

	SS vs. **DS**	Effect
NEAR	p<.0001'***'	**DS** < **SS**
MID	p<.0001'***'	**SS** < **DS**
FAR	p=.1353	–

Table 5.1: The effect of **SS** versus **DS** for NEAR, MID, and FAR row.

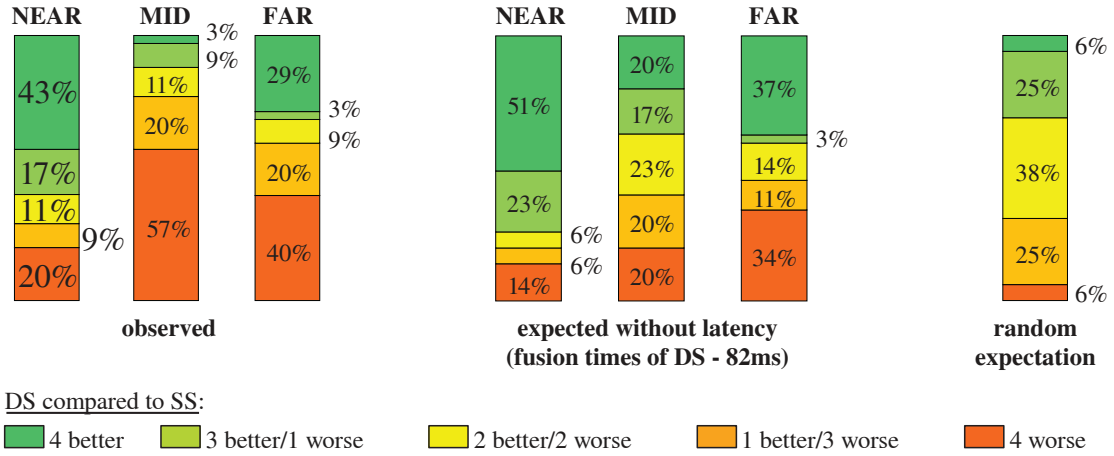


Figure 5.6: Visualization of the percentage of participants who had a benefit from dynamic stereo. We grouped users according to the number of QUESTs having a shorter fusion time in **DS** compared to **SS**. In the second group, we added results we expect from an ideal disparity adjustment without a technical latency. The last chart shows the expected result of a random process.

5.4.2 User Group Analysis

One important result of this experiment is that fusion times vary considerably with users. This suggests that stereoscopic fusion which occurs within the stereo viewing comfort zone are idiosyncratic characteristics of each participant's visual system. To investigate on how many users **DS** is beneficial, we counted for each row the amount of corresponding QUESTs which had a better fusion time in **DS** rather than in **SS**. Since each participant performed 4 QUESTs in each row, we visualized the result with pie charts with five sectors, each corresponding to one of the possible outcomes (1, 2, 3 or 4 better in **DS**, or all being worse) for each participant. We show the results in Figure 5.6 for each row of squares. In addition, we include a fourth pie chart which shows the expected result for a random process.The charts illustrate that a majority of 60% of users benefit from **DS** in the **N**EAR row ($3/4$ or $4/4$ of the fusion times better in **DS** than **SS**) which is more than double the percentage (31%) expected by a random process. For the **M**ID row, however, there is a significant majority of 77% which has a disadvantage ($0/4$ or $1/4$) from **DS** and a significantly smaller group having an advantage of **DS** than expected by chance. In the **F**AR row the results are more ambivalent. The percentage of users (32%) having an advantage with **DS** lies within the range of a random process (31%), while the fraction of those benefiting from **SS** (63%) is significantly higher than expected by chance. However, looking at the group of 29% of all participants having a maximum benefit ($4/4$), we also see a significant deviation from a random expectation (6%). Overall, these results suggest that the user's comfort zone is personalized and is related to the positive or negative impact that dynamic focal plane manipulation may have on him.

[1]best case: $35ms$(latency)$+1 \times 20ms$(sampling)$+17ms$(display)
worst case: $35ms$(latency)$+2 \times 20ms$(sampling)$+17ms$(display)

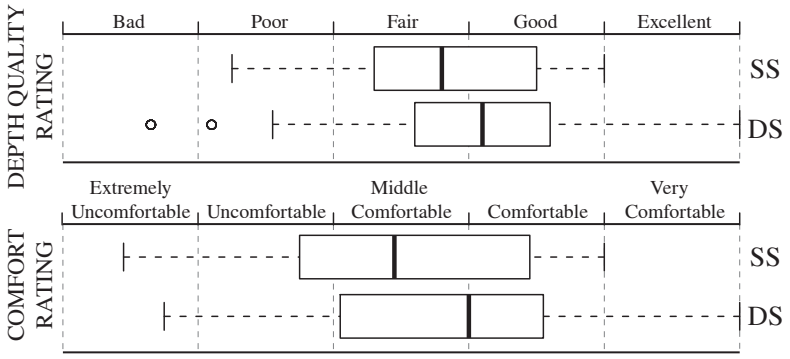


Figure 5.7: Distribution of subjective comfort and depth quality ratings visualized as Boxplots.

To gain insight on what can be expected from an optimal disparity adjustment performed with zero latency, we simulated a ideal case by subtracting the average latency of $82ms$ from the fusion times observed in the **DS** condition. However, while the amount of users having a benefit from **DS** increases considerably in all three rows, there is still a significant percentage of users who saw the maximum disadvantage from **DS** in the **M**ID and **F**AR rows. Particularly for the **F**AR row there appears to still be some overhead probably resulting from other factors other than delayed disparity adjustments.

5.4.3 Subjective Measurements

The analysis of user ratings is perfromed using descriptive statistics. Results are shown in Figure 5.7 where we visualized the distribution of the subjective comfort and quality ratings. For the comfort ratings, we obtained a positive effect of using **DS** compared to **SS**, albeit with a weak significance (paired two-tailed Wilcoxon T test, $V = 174, p = 0.09$). Overall 50% of all participants rated **DS** to be "comfortable" at least, while the median for **SS** is at the "middle comfortable" level. A slight positive shift from **SS** to **DS** was also observed in the quality ratings, but the effect is not significant (Wilcoxon T test, $V = 187.5, p = 0.16$). Since **DS** actually distorts the relation of disparities within a scene of objects, no decrease in perceived quality can be also interpreted as a result favoring the application of **DS** if viewing comfort can be increased.

5.4.4 Other Factors Influencing Fusion Times

Besides disparity, we contemplated that fusion times may also be affected by the position of the square or the type of saccade prior to the fusion process. We performed a factorial analysis to determine the effects of depth (**N**EAR vs **M**ID vs **F**AR), saccade length (**l**ong vs **s**hort) and direction (**f**orward vs **b**ackward), and azimuth (**c**entral vs **l**ateral). We first analyzed the effect of depth using a Friedman test (the non-parametric equivalent to a 2-way ANOVA) which allows replicates (i.e,4). However, the Friedman test is not applicable for a multi-factorial analysis. Assuming that depth is the most dominant factor, we split the data for each depth (**N**,**M**,**F**). Since there are only two independent variables per row which have only two categories, we could use a Wilcox test. In particular, we performed a paired Wilcox test on the single factors,

		SS	Effect	**DS**	Effect
	Depth	p=10^{-26}'***'	**M**<**F**<**N**	p=.069'.'	(**F**<**N**<**M**)
FAR	s.length	p=.0013'**'	**s**<**l**	p=.48	–
	azimuth	p = .20	–	p=.95	–
MID	s.dir.	p=.04'*'	**f**<**b**	p=.65	–
	azimuth	p=.014'*'	**c**<**l**	p=.37	–
NEAR	s.length	p=.77	–	p=.067'.'	(**s**<**l**)
	azimuth	p=.17	–	p=.056'.'	(**c**<**l**)

Table 5.2: Factorial analysis of rank-transformed fusion times.

i.e., saccade length, saccade direction and azimuth. To compensate potential effects resulting from interactions of the two factors present in each row, we sorted in the evaluation of one factor the data such that the Wilcox test always compared pairs corresponding to the same category of the other factor.

The results which are listed in Table 5.2 show that the lower variance of fusion times in the **DS** condition is due to a clear reduction of effects resulting from object distance, saccade length and direction. These factors are more pronounced in the **SS** condition. There we found that fusion times are significantly lower when a user performs a "short" saccade from the **M**ID to the **F**AR row compared to a "long" saccade from the **N**EAR to the **F**AR row, though this does not apply for saccades in the backward direction. Furthermore, we found a significant fusion time advantage at the object located in the center of the **M**ID row and for forward saccades compared to backward saccades. Overall, these results suggest that fusion times are also affected by prior gaze movement. Other effects of preceding saccades become apparent in the gaze analysis which we discuss next.

5.4.6 Gaze Analysis

By recording binocular gaze data throughout the experiment, we can also analyze the eye vergence behavior. To this end, we extracted the gaze signal of the first fixation each time the fusion task started and computed the screen distance between left and right gaze point in degrees of visual angle, which we further denote as *gaze disparity*. Since this requires an additional so-called depth calibration [47], we computed for each participant a correcting shift such that the average gaze disparity observed when a user fixates the plane of zero parallax in the (disparity manipulation free) **SS** condition fits to zero.

We analyzed this data by plotting the average gaze disparity of over 2000 fixations recorded during the experiment for each QUEST condition as a function of time. Averaging many fixations turned out to be a good remedy against the strong noise in vergence measurements taken with a conventional eye tracker (cf. [47]). We temporally aligned all gaze samples such that the fixation begin time corresponds to $t = 0$ and averaged for each sampling time interval ($\Delta t = 20ms$) the gaze disparities. The results corresponding to each QUEST condition are shown in Figure 5.8. For the **DS** condition, we split the gaze data obtained from fixations on

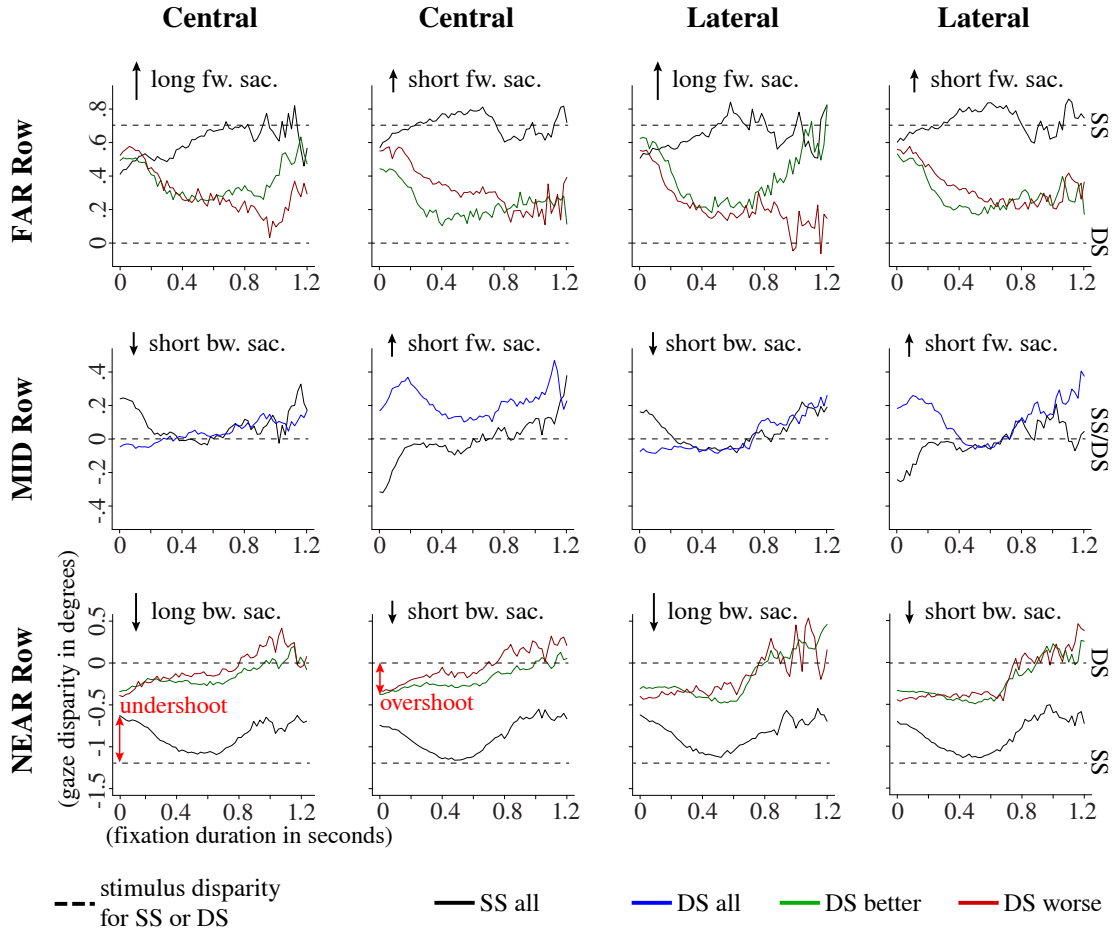


Figure 5.8: Temporal behavior of eye vergences. We derived this plots by averaging data of all fixations corresponding to the same QUEST.

the **N**EAR or **F**AR row in two groups: one group contains participants having a fusion time advantage in the **DS** condition (green graph), and the other group includes those who had a disadvantage (red graph). These groups correspond to the green/lime and red/orange fractions shown in the group analysis before, respectively (Figure 5.6). However, for the **M**ID row we have consolidated all data of the **DS** condition into a single graph (blue), since we did not observe a clear by participant clustering as for the other two rows.

In the case of **SS** condition (black), these plots make apparent that after a saccade the gaze disparity has an **undershoot** relative to the targeted stimulus disparity (dotted line). At fixation time eye rotations accommodate the stimulus disparity. In particular, saccades in forward direction are followed by a divergent eye rotation, and backward saccades by a convergent eye rotation, respectively. Moreover, we observe in the **N**EAR row after $600ms$, that gaze disparity slowly drifts away from the stimulus disparity. A reason for this could be that it is difficult to maintain the alignment between extensive stimulus disparities and the gaze disparity for a

longer time due to an **acute** emergence of visual fatigue. As a side-remark, detecting such patterns could be a new way to measure discomfort without measuring fusion time.

In the **DS** condition, the dynamic disparity modifications fundamentally change the behavior pattern observed under consistent disparity conditions. In contrast to **SS**, we observe an **overshoot** of the stimulus disparity after the saccade, which is followed by an accommodating vergence in the opposite depth-direction as the saccade. Another difference is that the stimulus disparity it not accommodated so well as observed in the **SS** condition, what is most obvious in the **F**AR row. Thus, stereo fusion in **DS** seems to often take place with higher retinal disparities requiring to fuse objects nearer to the limits of Panums fusional area (which is usually in the range of $0° \pm 1°$ of retinal disparity). A reason for this could be that a user's expectations together with monocular depth cues of the scene are in conflict with binocular depth perception. Moreover, extensive peripheral disparities, which occur in the **N**EAR row when the plane of zero parallax is shifted to the **F**AR row, might impede the accommodating vergence to disparities in the **F**AR row.

Looking at the difference between those participants having a fusion advantage in **DS** (green graphs), we can observe the most noticeable difference to those with a disadvantage (red graphs) in fixations following a short saccade to the FAR row, where the green group approaches faster the stimulus disparity. When focusing the central square this behavior is most pronounced and green vergence graphs do also have a smaller overshoot. However, the gaze disparity graphs do not explain why the green group has a fusion advantage after **l**ong saccades in the FAR row, or in all fixation on the **N**EAR row, respectively.

Overall, the gaze analysis results mainly reveal that users perform preparatory vergence movement already during a saccade, which is consistent with findings from vision literature [69, 188]. While under natural viewing conditions the target disparity is undershot, **DS** yields an overshoot through manipulating the stimulus disparity.

5.5 Discussion

We found that gaze-controlled manipulation of the plane of zero parallax can lower fusion times for large disparities. However, there is a delay in fusion time that is attributed to the way we detect *when* to adjust the plane of zero parallax. In this work we did not assume prior knowledge of the user's future gaze target intentionally. Instead we used a strategy that would be more appropriate for replicating the behavior of such a technique in a real-time gaze-contingent application. Our strategy after cuing the next square was to collect two gaze samples in proximity to the target, which constituted the initiation of a fixation. This source of fusion time delay could be technically reduced, for instance by sampling at a higher rate or by methods which can predict future gaze points.

This technical latency does not fully explain the overhead in fusion time that was observed after adjusting the plane of zero parallax. This became particularly apparent in the **F**AR plane, where a large group of users had a longer fusion time overhead than predicted by the latency alone. We believe two reasons may be responsible for this. First, the abrupt change of disparities may require a readjustment of the visual system that introduces a delay in fusion. Second, stereo fusion may be facilitated by sensory hysteresis ([69, p. 322]) or peripheral vision pro-

cessing [32]. Fast manipulation of the stimuli's disparities may be interfering or inhibiting the function of these processes. Moreover, the results of the gaze analysis suggest that the conflict between anticipated and adjusted disparities after a saccade may have a negative impact on the post-saccadic vergence accommodation. Under natural viewing conditions, vergence eye movements occur already at saccade time and usually undershoot the disparity of the stimulus at the sacced landing position. This is followed by a slow and more accurate post-saccadic vergence accommodation into the same orientation as the vergence movements occuring during a saccade [188]. Due to that, disparity manipulations cause an overshoot of the stimulus disparity. Since the vergence accommodation at fixation time has to be performed in the reverse direction as under natural conditions, we speculate that this could be another source of discomfort.

Overall, our methodology allowed an in depth investigation identifying the main problems of fast gaze-contingent disparity adjustments. The implications of our results should mainly affect the design of smart disparity adjustment strategies and the way models for stereo comfort should be designed and measured:

- Ideal comfort models should account for gaze movement from previous to the next object, as suggested by the results of the gaze analysis showing that fusion times and vergence behaviors are both affected by the type of the previous saccade.

- Apart from measuring fusion times, the additional use of gaze data to analyze temporal vergence behavior provides a powerful tool giving deeper insights how discomfort may arise. We believe that features of temporal vergence signals could be linked to discomfort and potentially allow objective comfort measurements with more natural stimuli than RDSs.

- And finally, smart disparity adjustments should be performed such that a stimulus disparity overshoot in saccadic vergence movements is avoided. Such a strategy would be, for instance, to shift the plane of zero parallax to the fixated stimulus distance biased by the expected overshoot.

5.6 Conclusion and Future Work

We consider this work as an important step towards the development of gaze-controlled disparity in stereoscopic applications. To develop such stereo controllers, it is necessary to devise a computational model that predicts the costs (i.e., expected fusion delay) and benefits (i.e., expected fusion time reduction) of a possible disparity adjustment in order to select the most appropriate configuration by means of maximizing the cost-benefit ratio. Although our results could be implemented as a preliminary instance of such a model (with 3 discrete distances), a more fine-grained model should be derived by further experiments.

For instance, our experiment may be carried out with stimuli arranged in more depth layers, which would allow obtaining samples of fusion times for more than three disparity levels. Furthermore, other disparity-adjustment strategies than fast switching need to be investigated. Such strategies may include smooth transitions and methods which can mask disparity adjustments or even avoid latency. For instance, using high frequency (e.g. $200Hz$) eye-trackers with low

latency may potentially allow to predict saccade-landing positions at the beginning of a saccade using a ballistic model [93]. Being able to estimate the future gaze position would then allow to perform the disparity adjustments within the time window of a saccade during which the stream of visual processing is disrupted [128]. However, avoiding latency is a hard problem and requires that it is being traded off against a much lower accuracy in the determination of the attended depth plane. An inaccurate identification of a future gaze location would be serious hindrance for many end-user applications. Furthermore, disparity adjustments performed during a saccade produce a conflict between the disparity a user anticipates and the disparty finally seen on the saccade landing target. This results in an unnatural experience which may cause fusion time delays and is potentially another source of discomfort.

Latency and fusion time delays, as identified in the presented experiment, are not the only problems that should be solved to advance the state-of-the-art in applications with gaze-controlled stereo disparities. For more complex scenes, that are commonplace in end-user commercial applications, it is particularly important to be able to reliably identify the feature or object currently in the user's focus, thus obtaining accurately the depth at which the user is directing his gaze. In dynamic 3D scenes with a high density of objects and motion (e.g. 3D computer games) identifying the attended depth plane can be very challenging, because visual attention does not always coincide with the exact deployment of gaze (e.g. gaze may be deployed in close proximity to the target). Robust gaze-to-object mapping methods as investigated in Chapter 3 have to be used for 3D applications.

Finally, an ideal model should account for the fact that viewing comfort zones differ among users. Ideally, future end-user applications would use calibration to generate a personalized model by fitting the model's parameters to the perceptual characteristics of each user.

CHAPTER 6

Conclusion

You will always find something in
the last place you look

Murphy's Law

6.1 Discussion of Contributions

This theses addresses the challenges that arise when processing eye-tracking data recorded in interactive virtual environment applications in order to infer attended objects in a single frame (Challenge I), or to analyze and predict visual attention from gaze data of many frames and users (Challenge II). Moreover, with a pilot study on gaze-controlled stereoscopic displays we explored the potential benefits and issues in an example application for attention-aware rendering techniques (Challenge III):

Challenge I: Inferring Attention from Gaze

With the methodology proposed Chapter 3, we studied how to correlate gaze information with object-based attention. The primary assumption made in this work was that attention is object-based and that an attentional selection can be represented with a data structure (i.e., item buffer) which encodes how objects are mapped to pixels. While this is an intuitive approach, the representation of attentional selection as pixel maps closely resembles to the data structures that were proposed with the Boolean-Map theory of visual attention [73]. Further, we assumed that attention is directed to one object at a time, and thus we proposed a probabilistic approach to quantify attention in a scene of objects. This work was published in ACM journal Transactions on Applied Perception 2014 [6].

Contribution

This work presents an initial solution on this challenge with two major contributions: First, we devised experimental methodology to obtain a ground truth data set that can be used to evaluate gaze-to-object mapping techniques. Second, we provided a formal specification of gaze-to-object mapping as Bayesian inference problem. Based on this seminal work, future work can advance gaze-to-object mapping by slowly increasing the complexity of the techniques. We believe that graphical models, such as Bayesian networks, provide an appropriate formal representation to design models for gaze-to-object mapping. We also believe that this was a fundamental step towards understanding object-based attention.

Main Findings

We learned from this work is that gaze-to-object mapping in three-dimensional and highly dynamic scenes is a challenging problem that can be easily underestimated. In fact, it turned out that it is hard to improve gaze-to-object mapping methods over trivial solutions, such as using the distance to the center of gravity. During development time, we intensively worked on finding more sophisticated solutions than the simple approaches we finally formally described and assessed. For instance, we tried to establish a Bayesian model which includes information of previous gaze samples in order to exploit the fact that gaze-pointing acuity is a function of movement distance. We also attempted to increase performance by using image features or saliency maps. Unfortunately, none of these attempts yielded improvements which justified the costs of increased model-complexity compared to straight-forward approaches we used as baseline for comparison.

The most important outcome of this work was that we understood that gaze-to-object mapping techniques can be grouped in two categories: methods from first category, which we denoted as *Active Gaze-to-Object Mapping*, are based on the assumption that visual attention is active and determines gaze. In contrast to that, approaches from second category, which we named *Passive Gaze-to-Object Mapping*, assume that visual attention is passive and follows gaze. Both approaches are reasonable, and, depending on the task and object, attention may behave in both ways, active and passive. In our experimental evaluation it turned out that gaze-to-object mapping during visual search for compact object shapes works best with active gaze-to-object mapping techniques, since these methods are independent of the size of the region an object covers on the screen.

Open Questions and Outlook

The next big challenge is to improve gaze-to-object mapping methods over straight-forward approaches. Given the limitation of our methodology, it will be necessary to increase the amount of test scenarios used to assess gaze-to-object mapping techniques. We already attempted constructing a wide range of representative test cases, but due to the curse of dimensionality in the multitude of factors involved in the perception of visually complex and animated stimuli, our evaluation data set covers only a limited fraction of scenarios where gaze-to-object mapping becomes a challenge. The evaluation data set needs to be extended with data of many further

experiments using a broader range of objects and scenes. Challenging examples that we did not use in our evaluation include scenes with virtual humans or animals, objects which are placed in vegetation scenes, or high speed animations and jerky camera movements which often occur in fast-paced computer games. Besides properties of scene content, another factorial dimension which should be considered is the task a user is performing while a gaze-to-object mapping technique is applied. Although visual search is fundamental behavior of vision, other tasks, such as visual exploration of a scene, object manipulation during interaction, or visual comparison of multiple objects, may result in different gaze behaviors which should be captured by an ideal attention-observing system as well.

Another interesting avenue for future research is to use gaze-to-object mapping techniques in the reverse direction. Instead of inferring a posterior probability for the attended object from gaze observations, we can also infer a posterior probability for gaze positions or movements, given an object-based attention prior, such as a task relevance map, statistical models derived from previously recorded gaze or simply a uniform distribution. This may find utility for increasing accuracy of low-quality eye trackers or for eye tracker calibration procedures that can be run at application time. Moreover, this could be a promising avenue to improve saliency-map algorithms. We believe that saliency maps should account for both the tendency to orient gaze towards objects and the way how a user deploys gaze on attended objects.

Challenge II: Attention Analysis and Prediction

Chapter 4 proposed a novel pipeline for gaze analysis. Our idea was to map spatial gaze data to scene objects which can be further abstracted by properties or semantics. This pipeline considerably facilitates gaze analysis since a user does not need to manually specify the spatial layout of objects of interest as necessary in commercial gaze-analysis tools.

This work was published in ACM Transactions on Applied Perception in 2010 and also contributed a signficiant part in a chapter on visual attention in computer games [168] in the book "Game Telemetry and Metrics: Maximizing the Value of User Data" (Springer 2013).

Contribution

The contributions of this work are mostly on the technical side and we made an important step to identify the problem scope that arises when eye-tracking is used to generate gaze statistics for three-dimensional computer games or interactive virtual environments. We contributed a pipeline which allows an experimenter to introduce high-level knowledge into the gaze-analysis process using a scripting language that provides a powerful interface to specify models for the statistical analysis of gaze. We tested our prototype by using the derived statistical models to predict visual attention.

Main Findings

With our example application, we demonstrated that this pipeline produces results that are consistent with our expectations (e.g., high importance for enemies or sounding objects). However, we expect that attention is way more difficult to predict intuitively in more complex interactive

virtual environments or commercial computer games, where a gaze analysis could reveal much more surprising insights.

Moreover, we attempted predicting attention using statistical models derived with the analysis pipeline. We were able to show that the gaze statistics collected for semantic categories can predict attention much better than chance, particularly in task-intensive situations of our test game, where we found that the object that is attended most likely could be predicted in about 70% of all cases. A disappointing result was, however, the observation that a trivial solution which predicts attention by the distance to the screen center performs better (up to 80% success rate) than more sophisticated models. This result is due to the nature of first-person games where the mouse is used to orient the camera. To have an optimal point of view, the user tries orienting the camera such that attended objects are brought to the center of the field of view.

However, seen in a more positive way, this results also suggests that we can easily supplement eye trackers in first-person shooter games by determining objects which are located near crosshairs. For applications such as multiplayer online games, this would be an effective low-cost solution to coarsely track a user's attention during game play.

Open Questions and Outlook

Despite our results, which indicate that visual attention is best predicted with trivial solutions, we believe that in other types of games our approach could outperform trivial solutions or heuristics. The open question is whether our approach may perform better than trivial solutions in applications where the camera is not under a user's control.

To further this research, it is also necessary to explore the utility of our tools for attention analysis in numerous applications, i.e., various game genres, or different kinds of virtual environment simulators. An interesting example where the pipeline may have a strong utility are studies on visual attention in car driving simulators. Given the scripting interface, an experimenter can define the meaning of objects in a traffic scene with respect to the current goal of a car driver. For instance, an approaching car in a street crossing may be attended in a different way than a parking car or a pedestrian that has already crossed the street.

Challenge III : Attention-Aware Rendering

The work presented in Chapter IV contributes a pilot study to explore the application of eye-tracking for increasing comfort in stereoscopic displays. At an early stage of this work, we implemented a gaze-adaptive disparity controller which we used with an interactive computer game. However, it turned out that it is very difficult to assess the perceptual benefits and drawbacks of this method. Thus, we decided to take one step back and to first establish a methodology which allows us to evaluate the application. Our goal was to design a method for an objective perceptual evaluation in order to better identify when and where a user has problems to fuse stereo image pairs.

This work was published in the proceedings of the ACM Symposium in Eye Tracking and Applications 2014 [5].

Contribution

The main contribution of this work is a study design which allows measuring fusion times for single objects which are arranged in a three-dimensional scene. With this experimental setup, we were able early on to identify issues that should be addressed when developing gaze-controlled stereoscopic applications.

We proposed to use an eye tracker and psychophysics to evaluate discomfort objectively. In particular, we evaluated visual discomfort induced by stereoscopic displays by measuring the time a user needs to fuse a pair of stereo images or by analyzing binocular gaze data. Fusion times have proven to be a good indicator for discomfort, while eye tracking can be used to (a) control the position where and the time when fusion time measurements are taken and (b) to analyze vergence behavior by processing binocular gaze data. Previous work experimental designs use RDS to measure fusion times for single objects (i.e., one RDS) shown in isolation, while under more natural conditions a focused object is located in the context of a scene of objects. We contributed an experimental design which allows to measure fusion times for particular objects which are located in the context of a three-dimensional scene. Having such a methodology, we are able to study the effects of gaze-controlled stereo on viewing comfort more accurately, since we obtain results for particular scene objects.

Main Findings

In our pilot study, we evaluated discomfort in gaze-controlled stereoscopic displays by comparing the result to a conventional stereo condition with a static configuration (i.e., the control baseline condition). We found that gaze controlled disparity adjustments are most successfully applied when a user focuses on near objects shown with high disparities, while shifting the focus to far objects in a scene should be avoided as it does not increase comfort compared to applications which keep a static focus in the center of a scene. Also, we found that the benefits of gaze-responsive stereo depend on the user. Among our participants there was a large group of users which did not have a significant benefit from gaze-controlled stereo. Most of the users do not benefit from the disparity adjustments since they have no difficulties in fusing images even if the vergence/accomodation conflict is severe. This also raises the question whether fusion-time measurement alone is sufficient to objectively measure discomfort.

Another issue which became apparent in this study is the latency in the determination of the attended object which directly translates to an equivalent increase of the fusion times. However, this is a technical problem which could be reduced or even solved by using low-latency and high frame rate eye-tracking hardware and ballistic models which can predict a saccade landing position at the time of the onset of a saccade. Thanks to change blindness during saccades, such a strategy may perform fast disparity adjustment without a user noticing the disparity manipulations. However, a really fundamental limitation identified in this work is that a user adapts vergence to an anticipated target disparity already at saccade time. When disparities are altered by a gaze-responsive stereo configuration, this behavior results in an overshoot of the manipulated disparities.

Open Questions and Outlook

An important open issue which has to be clarified is whether anticipatory vergence adaptations performed during a saccade are a significant problem for the experience of a user. While this problem could render fast disparity adjustments as not applicable at all, some hope is still left: on the one hand, a user might not experience the overshoot as discomfort or even change this behavior by learning in long term use. In personal discussions after the experiments, some users reported that they were feeling more comfortable with the gaze-controlled stereo with increasing exposure time and started to adapt to the behavior of the system. On the other hand, strategies could be developed which trade of overshoot against lower vergence/accomodation conflicts in a way that it does not affect the experience negatively. At least the results of our subjective evaluation do not indicate a negative impact on a user's subjective experience. However, the questions on experienced discomfort did not particularly address this issue. Thus, further user studies are required to answer this question.

Another critical issue that should be evaluated is how the instant alignment of the stereo focus with the depth of an attended object affects global depth perception, i.e., the perceived spatial relation between objects in the whole scene. Since sensitivity to disparity changes is highest near the plane of zero parallax, our strategy maximizes stereoscopic depth perception locally. But aligning the plane of zero parallax with the distance of the attended object comes with the cost that depth information in absolute disparities gets lost completely. Thus, the binocularly perceived relation of the scene objects' depths is severely distorted and can only be resolved from monocular depth cues. Further experiments need to be carried out to answer the question whether, or to which extent, this "distortion" of absolute disparities affects global depth perception in a negative way. To this end, it would be necessary to design experiments which assess a user's performance in tasks which require correct depth perception.

Finally, we also think that considerable work has to be done to improve methodologies to measure visual discomfort. This is, on the one hand, important to evaluate applications which try to reduce discomfort. On the other hand, this is important to build better models for visual discomfort, which should be ideally personalized and also account for the properties of the stimulus, such as the arrangement of scene objects. Measuring fusion times with the joint use of pychophysics and an eye tracker is probably not the most elegant and easiest way to do that, and the results may not necessarily generalize to applications with more complex stimuli than random-dot stereograms. With the experience of our work, we would now rather advocate to focus on binocular gaze analysis to detect patterns in the vergence signal which allow to infer fusion times. For instance, the recent work of Templin et al. [174] proposed an elegant methodology to estimate vergence response times from binocular gaze signals recorded with an eye tracker. However, still some ground work is required in order to map patterns observed in a binocular gaze analysis to viewing comfort. One way would be to perform experiments where a multitude of factors which are related to discomfort are measured simultaneously (e.g., fusion times, subjective judgments, eye blink rate, EEG etc.). Having this data, we can correlate vergence patterns with other comfort indicators in order to find ways to detect comfort-predicting features in the extremely noisy vergence signal. The great advantage of binocular comfort measurement would be that we can use stimuli of arbitrary complexity and measure discomfort every time and everywhere a user is fixating.

6.2 Ideas for Future Work

Apart from future work that advances the methods proposed in this thesis, which was discussed in the future work sections of this thesis and the discussion presented above, the following ideas could be interesting to pursue in future:

Probabilistic Attention-Aware Rendering

The approaches used to model attention in this work are in their nature *probabilistic*. This has the advantage that uncertainties are represented in a probability function that provides a confidence measure for each possible attentional selection (e.g., pixel-map of one object). Given a particular configuration of the rendering method and a perceptual model (e.g., visible difference predictor or stereo comfort model), the attention probability allows to predict the *expected value* of perceived rendering errors or the experienced discomfort. Attention-aware rendering can be defined as an optimization problem which is solved by searching for a configuration which maximizes rendering speed or comfort while the expected error a user may perceive is minimized.

Application with Non-Artificial Stimuli

The concept of using an item buffer to encode a scene as object-location map is a powerful and generic approach. The boolean representation of coherent image regions that belong to a unique object can be generalized to a "feature buffer" containing a discrete set of boolean maps representing image regions sharing a similar color, orientation, depth or motion feature value. Since such a feature buffer can be extracted with image processing tools from digital images, the gaze-analysis methodology presented here could be brought forward to find application also on real world stimuli. This could be for instance useful to generate attention predictors based on gaze statistics that were build in feature space (e.g., color importance maps).

Combining Eye-Tracking with Other Observational Devices

User experience in interactive virtual environments and games can be studied more accurately by the accompanying use of biometric devices. For instance, it could be useful to record electro encephalography (EEG) data, galvanic skin response (GSR), heart rate (EKG), blood volume pulse (BVP), and breathing [199]. With the additional use of such observational tools, further information about a user's emotional state could be obtained and linked with the visually attended scene content. Much more advanced would be the use of functional magnet resonance imaging (fMRI), which recently evolved as the most powerful tool to study brain activity in action. Linking visual attention as measured with the eye tracker with neural activation patterns observed in real-time with fMRI, could be an extremely powerful method of investigation.

Bibliography

[1] Tobii eye tracking: An introduction to eye tracking and tobii eye trackers. In *http://www.tobii.com/Global/Analysis/Training/WhitePapers*.

[2] Recommendation itu-r bt.2021, subjective methods for the assessment of stereoscopic 3dtv systems, 2012.

[3] Reynold Bailey, Ann McNamara, Nisha Sudarsanam, and Cindy Grimm. Subtle gaze direction. *ACM Trans. Graph.*, 28(4):100:1–100:14, September 2009.

[4] M. Behrmann, R.S. Zemel, and M.C. Mozer. Object-based attention and occlusion evidence from normal participants and a computational model. *Journal of Experimental Psychology: Human Perception and Performance*, 24:1011–1036, 1998.

[5] Matthias Bernhard, Camillo Dell'mour, Michael Hecher, Efstathios Stavrakis, and Michael Wimmer. The effects of fast disparity adjustment in gaze-controlled stereoscopic applications. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '14, pages 111–118, New York, NY, USA, 2014. ACM.

[6] Matthias Bernhard, Efstathios Stavrakis, Michael Hecher, and Michael Wimmer. Gaze-to-object mapping during visual search in 3d virtual environments. *ACM Trans. Appl. Percept.*, 11(3):14:1–14:17, August 2014.

[7] Matthias Bernhard, Efstathios Stavrakis, and Michael Wimmer. An empirical pipeline to derive gaze prediction heuristics for 3d action games. *ACM Trans. Appl. Percept.*, 8:4:1–4:30, November 2010.

[8] Matthias Bernhard, Le Zhang, and Michael Wimmer. Manipulating attention in computer games. In *Proceedings of the IEEE IVMSP Workshop on Perception and Visual Signal Analysis*, pages 153–158. IEEE, June 2011.

[9] Maximino Bessa, Antonio Coelho, and Alan Chalmers. Selective rendering quality for an efficient navigational aid in virtual urban environments on mobile platforms. In *Proceedings of the 4th International Conference on Mobile and Ubiquitous Multimedia*, MUM '05, pages 109–113, New York, NY, USA, 2005. ACM.

[10] Jiří Bittner, Michael Wimmer, Harald Piringer, and Werner Purgathofer. Coherent hierarchical culling: Hardware occlusion queries made useful. *Computer Graphics Forum*, 23(3):615–624, September 2004. Proceedings of Eurographics.

[11] W. Blohm, I. P. Beldie, K. Schenke, K. Fazel, and S. Pastoor. Stereoscopic image representation with synthetic depth of field. *Journal of The Society for Information Display*, 5, 1997.

[12] T. Blum, M. Wieczorek, A. Aichert, R. Tibrewal, and N. Navab. The effect of out-of-focus blur on visual discomfort when using stereo displays. In *Mixed and Augmented Reality (ISMAR), 2010 9th IEEE International Symposium on*, pages 13 –17, Oct 2010.

[13] Mark R. Bolin and Gary W. Meyer. A perceptually based adaptive sampling algorithm. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '98, pages 299–309, New York, NY, USA, 1998. ACM.

[14] Thomas Booth, Srinivas Sridharan, Ann McNamara, Cindy Grimm, and Reynold Bailey. Guiding attention in controlled real-world environments. In *Proceedings of the ACM Symposium on Applied Perception*, SAP '13, pages 75–82, New York, NY, USA, 2013. ACM.

[15] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):185–207, Jan 2013.

[16] A. Borji, D. N. Sihite, and L. Itti. An object-based bayesian framework for top-down visual attention. In *Proc. Twenty-Sixth AAAI Conference on Artificial Intelligence (AAAI-12), Toronto, Canada*, pages 1529–1535, Aug 2012.

[17] A. Borji, D. N. Sihite, and L. Itti. Probabilistic learning of task-specific visual attention. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, Rhode Island*, pages 1–8, Jun 2012.

[18] A. Borji, D. N. Sihite, and L. Itti. What/where to look next? modeling top-down visual attention in complex interactive environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part A - Systems and Humans*, pages 1–16 (in press), 2012.

[19] Ralph A. Bradley and Milton E. Terry. Rank Analysis of Incomplete Block Designs: I. The Method of Paired Comparisons. *Biometrika*, 39(3/4), 1952.

[20] Belma R. Brkic, Alan Chalmers, Kevin Boulanger, Sumanta Pattanaik, and James Covington. Cross-modal affects of smell on the real-time rendering of grass. In *Proceedings of the 25th Spring Conference on Computer Graphics*, SCCG '09, pages 161–166, New York, NY, USA, 2009. ACM.

[21] D. Broadbent. *Perception and communication*. Pergamon Press, 1958.

[22] Roxanne L. Canosa, Jeff B. Pelz, Neil R. Mennie, and Joseph Peak. High-level aspects of oculomotor control during viewing of natural-task images. In B. E. Rogowitz and T. N. Pappas, editors, *Human Vision and Electronic Imaging VIII. Edited by Rogowitz, Bernice E.; Pappas, Thrasyvoulos N. Proceedings of the SPIE*, volume 5007 of *Presented at the Society of Photo-Optical Instrumentation Engineers (SPIE) Conference*, pages 240–251, june 2003.

[23] Marisa Carrasco. Visual attention: The past 25 years. *Vision Research*, 51(13):1484 – 1525, 2011. Vision Research 50th Anniversary Issue: Part 2.

[24] K Cater, A Chalmers, and P Ledda. Selective quality rendering by exploiting human inattentional blindness: looking but not seeing. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, pages 17–24, 2002.

[25] K Cater, A Chalmers, and G Ward. Detail to attention: Exploiting visual tasks for selective rendering. In *Proc. of the 14th Eurographics Workshop on Rendering*, pages 270–280, 2003.

[26] Kirsten Cater, Alan Chalmers, and Colin Dalton. Varying rendering fidelity by exploiting human change blindness. In *Proceedings of the 1st International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, GRAPHITE '03, pages 39–46, New York, NY, USA, 2003. ACM.

[27] Arzu Çöltekin. Space-variant image coding for stereoscopic media. In *Proceedings of the 27th conference on Picture Coding Symposium*, PCS'09, pages 533–536, Piscataway, NJ, USA, 2009. IEEE Press.

[28] Ufuk Celikcan, Gokcen Cimen, E. Bengu Kevinc, and Tolga K. Çapin. Attention-aware disparity control in interactive environments. *The Visual Computer*, 29(6-8):685–694, 2013.

[29] Zhe Chen. Object-based attention: a tutorial review. *Attention, perception & psychophysics*, 74(5):784–802, July 2012.

[30] Wei-Chung Cheng and Aldo Badano. A gaze-contingent high-dynamic range display for medical imaging applications, 2010.

[31] Sang-hyun Cho and Hang-bong Kang. An assessment of visual discomfort caused by motion-in-depth in stereoscopic 3d video. In *Proceedings of the British Machine Vision Conference*, pages 65.1–65.10. BMVA Press, 2012.

[32] Patricia M. Cisarik and Ronald S. Harwerth. Stereoscopic depth magnitude estimation: effects of stimulus spatial frequency and eccentricity. *Behavioural Brain Research*, 160(1):88–98, May 2005.

[33] Han Collewijn, Robert M. Steinman, Casper J. Erkelens, Zygmunt Pizlo, and Johannes van der Steen. Effect of freeing the head on eye movement characteristics during three

dimensional shifts of gaze and tracking. In Alain Berthoz, Pierre Paul Vidal, and Werner Graf, editors, *The Head-Neck Sensory Motor System*, pages 412–418. Oxford University Press, 1992.

[34] Scott Daly. The visible differences predictor: an algorithm for the assessment of image fidelity. In Andrew B. Watson, editor, *Digital images and human vision*, pages 179–206. MIT Press, Cambridge, MA, USA, 1993.

[35] Piotr Didyk, Tobias Ritschel, Elmar Eisemann, Karol Myszkowski, and Hans-Peter Seidel. A perceptual model for disparity. *ACM Transactions on Graphics (Proceedings SIGGRAPH 2011, Vancouver)*, 30(4), 2011.

[36] Piotr Didyk, Tobias Ritschel, Elmar Eisemann, Karol Myszkowski, Hans-Peter Seidel, and Wojciech Matusik. A luminance-contrast-aware disparity model and applications. *ACM Trans. Graph.*, 31(6):184:1–184:10, November 2012.

[37] R. Dodge and Ts Cline. The angle velocity of eye movements. *Psychological Review*, 1901.

[38] George Drettakis, Nicolas Bonneel, Carsten Dachsbacher, Sylvain Lefebvre, Michael Schwarz, and Isabelle Viaud-Delmon. An interactive perceptual rendering pipeline using contrast and spatial masking. In *Rendering Techniques (Proceedings of the Eurographics Symposium on Rendering)*. Eurographics, June 2007.

[39] J. Driver and G. C. Baylis. Movement and visual attention: The spotlight metaphor breaks down. *Journal of experimental psychology: Human perception and performance*, 15(3):448–456, 1989.

[40] Jon Driver and Charles Spence. Crossmodal attention. *Current Opinion in Neurobiology*, 8(2):245 – 253, 1998.

[41] A. T. Duchowski. A breadth-first survey of eye-tracking applications. *Behavior Research Methods, Instruments & Computers*, 34(4):455–470, November 2002.

[42] A T Duchowski. *Eye tracking methodology: Theory and practice*. Springer, New York, 2003.

[43] Andrew T. Duchowski and Arzu Çöltekin. Foveated gaze-contingent displays for peripheral lod management, 3d visualization, and stereo imaging. *ACM Trans. Multimedia Comput. Commun. Appl.*, 3(4):1–18, 2007.

[44] Andrew T. Duchowski, Nathan Cournia, and Hunter Murphy. Gaze-contingent displays: A review. *CyberPsychology & Behavior*, 7(6), February 2005.

[45] Andrew T. Duchowski, Donald H. House, Jordan Gestring, Rui I. Wang, Krzysztof Krejtz, Izabela Krejtz, Radoslaw Mantiuk, and Bartosz Bazyluk. Reducing visual discomfort of 3d stereoscopic displays with gaze-contingent depth-of-field. In *Proceedings of the ACM Symposium on Applied Perception*, SAP '14, pages 39–46, New York, NY, USA, 2014. ACM.

[46] Andrew T. Duchowski, Eric Medlin, Anand Gramopadhye, Brian Melloy, and Santosh Nair. Binocular eye tracking in vr for visual inspection training. In *Proceedings of the ACM symposium on Virtual reality software and technology*, VRST '01, pages 1–8, New York, NY, USA, 2001. ACM.

[47] Andrew T. Duchowski, Brandon Pelfrey, Donald H. House, and Rui Wang. Measuring gaze depth with an eye tracker during stereoscopic display. In *Proceedings of the ACM SIGGRAPH Symposium on Applied Perception in Graphics and Visualization*, APGV '11, pages 15–22, New York, NY, USA, 2011. ACM.

[48] J. Duncan. Selective attention and the organization of visual information. *Journal of experimental psychology. General*, 113(4):501–517, December 1984.

[49] Marg E. Development of electro-oculography: Standing potential of the eye in registration of eye movement. *A.M.A. Archives of Ophthalmology*, 45(2):169–185, 1951.

[50] M. S. El-Nasr, T. Vasilakos, C. Rao, and J. Zupko. Dynamic intelligent lighting for directing visual attention in interactive 3d scenes. *IEEE Transactions on Computational Intelligence and AI in Games*, 1(2):115–134, 2009.

[51] Magy Seif El-Nasr and Su Yan. Visual attention in 3d video games. In *ACE'06: Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology*, page 22, New York, NY, USA, 2006. ACM.

[52] Diego Fernandez-Duque and Mark L. Johnson. Attention metaphors: How metaphors guide the cognitive psychology of attention. *Cognitive Science*, 23(1):83–116, 1999.

[53] James A. Ferwerda, Peter Shirley, Sumanta N. Pattanaik, and Donald P. Greenberg. A model of visual masking for computer graphics. In *SIGGRAPH*, pages 143–152, 1997.

[54] Martin Fisker, Kristoffer Gram, Kasper Kronborg Thomsen, Dimitra Vasilarou, and Martin Kraus. Automatic Convergence Adjustment for Stereoscopy using Eye Tracking. In *EG 2013 - Posters*, pages 23–24, 2013.

[55] Thomas A. Funkhouser and Carlo H. Séquin. Adaptive display algorithm for interactive frame rates during visualization of complex virtual environments. In *SIGGRAPH '93: Proceedings of the 20th annual conference on Computer graphics and interactive techniques*, pages 247–254, New York, NY, USA, 1993. ACM.

[56] A. Gaggioli. Using virtual reality in experimental psychology. *Towards CynerPsychology: Mind,Cognitions and Scociety in the Internet Age*, 2003.

[57] Brian Guenter, Mark Finch, Steven Drucker, Desney Tan, and John Snyder. Foveated 3d graphics. *ACM Transactions on Graphics*, 31(6):164:1–164:10, November 2012.

[58] Diego Gutierrez, Oscar Anson, Francisco J. Seron, Veronica Sundstedt, and Alan Chalmers. Efficient physically-based perceptual rendering of participating media. In *ACM SIGGRAPH Full Conference DVD, Los Angeles, USA*. ACM, August 2005.

[59] Jörg Haber, Karol Myszkowski, Hitoshi Yamauchi, and Hans-Peter Seidel. Perceptually guided corrective splatting. *Computer Graphics Forum*, 20(3):142–153, 2001.

[60] Mary M. Hayhoe, Anurag Shrivastava, Ryan Mruczek, and Jeff B. Pelz. Visual memory and motor planning in a natural task. *Journal of Vision*, 3(1):49–63, 2 2003.

[61] Simon Heinzle, Pierre Greisen, David Gallup, Christine Chen, Daniel Saner, Aljoscha Smolic, Andreas Burg, Wojciech Matusik, and Markus Gross. Computational stereo camera system with programmable control loop. *ACM Trans. Graph.*, 30(4):94:1–94:10, July 2011.

[62] Robert T. Held, Emily A. Cooper, James F. O'Brien, and Martin S. Banks. Using blur to affect perceived distance and size. *ACM Trans. Graph.*, 29:19:1–19:16, April 2010.

[63] J.M. Henderson. Eye movement control during visual object processing: effects of initial fixation position and semantic constraint. *Canadian Journal of Experimental Psychology*, 27(1):79–98, March 1993.

[64] Sébastien Hillaire, Anatole Lécuyer, Rémi Cozot, and Géry Casiez. Depth-of-field blur effects for first-person navigation in virtual environments. *IEEE Comput. Graph. Appl.*, 28:47–55, November 2008.

[65] Sébastien Hillaire, Anatole Lécuyer, Rémi Cozot, and Géry Casiez. Using an eye-tracking system to improve camera motions and depth-of-field blur effects in virtual environments. In *VR*, pages 47–50, 2008.

[66] D. M. Hoffman, A. R. Girshick, K. Akeley, and M. S. Banks. Vergence-accommodation conflicts hinder visual performance and cause visual fatigue. *Journal of vision*, 8(3), 2008.

[67] Andrew Hollingworth. Task specificity and the influence of memory on visual search: Comment on Võ and Wolfe (2012). *Journal of Experimental Psychology: Human Perception and Performance*, 38(6):1596–1603, 2012.

[68] Andrew Hollingworth and John M. Henderson. Accurate Visual Memory for Previously Attended Objects in Natural Scenes. *Journal of Experimental Psychology: Human Perception and Performance*, 28(1):113–136, 2002.

[69] Ian P. Howard. *Binocular Vision and Stereopsis*. Number no. 29 in Oxford psychology series. Oxford University Press, New York, 1995.

[70] Peter A Howarth. Potential hazards of viewing 3-d stereoscopic television, cinema and computer games: a review. *Ophthalmic and Physiological Optics*, 31(2):111–122, 2011.

[71] Sarah Howlett, John Hamill, and Carol O'Sullivan. Predicting and evaluating saliency for simplified polygonal models. *ACM Trans. Appl. Percept.*, 2(3):286–308, July 2005.

[72] Liqiang Huang. What is the unit of visual attention? object for selection, but boolean map for access. *J Exp Psychol Gen*, 139(1):162–79, 2010.

[73] Liqiang Huang and Harold Pashler. A boolean map theory of visual attention. *Psychological Review*, 114:599–631, 2007.

[74] E.B. Huey. *The Psychology & Pedagogy of Reading*. International Reading Association, 1968.

[75] Poika Isokoski, Markus Joos, Oleg Spakov, and Benoit Martin. Gaze controlled games. *Univers. Access Inf. Soc.*, 8(4):323–337, October 2009.

[76] L. Itti. Visual salience. In *http://www.scholarpedia.org/article/Visual_salience*, 2007.

[77] L. Itti and P. Baldi. Bayesian surprise attracts human attention. In *Advances in Neural Information Processing Systems, Vol. 19 (NIPS*2005)*, pages 547–554, Cambridge, MA, 2006. MIT Press.

[78] L. Itti and C. Koch. Computational modelling of visual attention. *Nature Reviews Neuroscience*, 2(3):194–203, Mar 2001.

[79] L Itti, C Koch, and E Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, 1998.

[80] Robert J. K. Jacob. Virtual environments and advanced interface design. chapter Eye Tracking in Advanced Interface Design, pages 258–288. Oxford University Press, Inc., New York, NY, USA, 1995.

[81] William James. *The Principles of Psychology, Vol. 1*. Dover Publications, 1890.

[82] E. Javal. Essai sur la Physiologie de la Lecture. *Annales D'Oculistique*, 1879.

[83] L Jie and J J Clark. Game design guided by visual attention. In L Ma, M Rauterberg, and R Nakatsu, editors, *Entertainment Computing, ICEC 2007*, volume 4740 of *Lecture Notes in Computer Science*, pages 345–355. Springer, 2007.

[84] Yong Ju Jung, Seong-il Lee, Hosik Sohn, Hyun Wook Park, and Yong Man Ro. Visual comfort assessment metric based on salient object motion information in stereoscopic video. *Journal of Electronic Imaging*, 21(1):011008–1–011008–16, 2012.

[85] Christoph Kayser, Christopher I. Petkov, Michael Lippert, and Nikos K. Logothetis. Mechanisms for Allocating Auditory Attention: An Auditory Saliency Map. *Current Biology*, 15(21):1943–1947, November 2005.

[86] A Kenny, H Koesling, D Delaney, S McLoone, and T Ward. A preliminary investigation into eye gaze data in a first person shooter game. In *19th European Conference on Modelling and Simulation*, pages 733–740, 2005.

[87] Alon S. Keren, Shlomit Yuval-Greenberg, and Leon Y. Deouell. Saccadic spike potentials in gamma-band eeg: Characterization, detection and suppression. *NeuroImage*, 49(3):2248 – 2263, 2010.

[88] Azam Khan, Justin Matejka, George Fitzmaurice, and Gordon Kurtenbach. Spotlight: Directing users' attention on large displays. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '05, pages 791–798, New York, NY, USA, 2005. ACM.

[89] Donghyun Kim, Sunghwan Choi, and Kwanghoon Sohn. Effect of vergence-accommodation conflict and parallax difference on binocular fusion for random dot stereogram. *IEEE Transactions on Circuits and Systems for Video Technology*, 22(5):811–816, May 2012.

[90] Joohwan Kim, David M Hoffman, and Martin S Banks. The zone of comfort : Predicting visual discomfort with stereo displays takashi shibata. *Journal of Vision*, 11(8):1–29, 2011.

[91] Christof Koch and Shimon Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 4:219–227, 1985.

[92] D. Kocian. Visual world subsystem. In *Super Cockpit Industry Days: Super Cockpit/Virtual Crew Systems*, pages 31 March–1 April 1987. Air Force Systems Command/Human Systems Division/Armstrong Aerospace Medical Research Laboratory, 1987.

[93] Oleg V. Komogortsev and Javed I. Khan. Eye movement prediction by kalman filter with integrated linear horizontal oculomotor plant mechanical model. In *Proceedings of the 2008 symposium on Eye tracking research & applications*, ETRA '08, pages 229–236, New York, NY, USA, 2008. ACM.

[94] S.J. Koppal, C.L. Zitnick, M.F. Cohen, Sing Bing Kang, B. Ressler, and A. Colburn. A viewer-centric editor for 3d movies. *Computer Graphics and Applications, IEEE*, 31(1):20–35, 2011.

[95] Dwight J Kravitz and Marlene Behrmann. Space-, object-, and feature-based attention interact to organize visual scenes. *Attention Perception Psychophysics*, 73(8):2434–47, 2011.

[96] Marc T M Lambooij, Wijnand A IJsselsteijn, and Ingrid Heynderickx. Visual discomfort in stereoscopic displays: a review. *Proceedings of SPIE*, 6490(1584), 2007.

[97] M Land, N Mennie, and J Rusted. The roles of vision and eye movements in the control of activities of daily living. *Perception*, 28(11):1311–1328, 1999.

[98] Manuel Lang, Alexander Hornung, Oliver Wang, Steven Poulakos, Aljoscha Smolic, and Markus Gross. Nonlinear disparity mapping for stereoscopic 3d. *ACM Trans. on Graphics (Proc. SIGGRAPH)*, 29(3), 2010.

[99] Stephen R.H. Langton, Anna S. Law, A. Mike Burton, and Stefan R. Schweinberger. Attention capture by faces. *Cognition*, 107(1):330 – 342, 2008.

[100] Chang Ha Lee, Amitabh Varshney, and David W. Jacobs. Mesh saliency. *ACM Trans. Graph.*, 24(3):659–666, July 2005.

[101] L. Leroy, P. Fuchs, and G. Moreau. Visual fatigue reduction for immersive stereoscopic displays by disparity, content, and focus-point adapted blur. *Industrial Electronics, IEEE Transactions on*, 59(10):3998 –4004, Oct. 2012.

[102] Marc Levoy and Ross Whitaker. Gaze-directed volume rendering. *SIGGRAPH Comput. Graph.*, 24:217–223, February 1990.

[103] Zhicheng Li and Laurent Itti. Visual attention guided video compression. *J. Vis.*, 8(6):772–772, 5 2008.

[104] Chun-Wei Liu, Tz-Huan Huang, Ming-Hsu Chang, Ken-Yi Lee, Chia-Kai Liang, and Yung-Yu Chuang. 3d cinematography principles and their applications to stereoscopic media processing. In *Proceedings of the 19th ACM international conference on Multimedia*, MM '11, pages 253–262, New York, NY, USA, 2011. ACM.

[105] R. Duncan Luce. *Individual Choice Behavior*. Wiley, New York, 1959.

[106] David Luebke, Benjamin Hallen, Dale Newfield, and Benjamin Watson. Perceptually driven simplification using gaze-directed rendering, 2000.

[107] John Mack and Irwin Rock. *Inattentional Blindness*. The MIT Press, Cambridge, MA, 1998.

[108] J.F. Mackworith and N.H. Mackworth. Eye fixations recorded on changing visual scenes by the television eye-marker. *J. Opt. Soc. Am.*, 48(7):439–444, Jul 1958.

[109] R. Mantiuk, B. Bazyluk, and R. K. Mantiuk. Gaze-driven object tracking for real time rendering. *Computer Graphics Forum*, 32:163–173, 2013.

[110] Radosław Mantiuk and Mateusz Markowski. Gaze-dependent tone mapping. In Mohamed Kamel and Aurélio Campilho, editors, *Image Analysis and Recognition*, volume 7950 of *Lecture Notes in Computer Science*, pages 426–433. Springer Berlin Heidelberg, 2013.

[111] Rafal Mantiuk, Karol Myszkowski, and Sumanta Pattanaik. Attention guided mpeg compression for computer animations. In *Proceedings of the 19th Spring Conference on Computer Graphics*, SCCG '03, pages 239–244, New York, NY, USA, 2003. ACM.

[112] Georgia Mastoropoulou, Kurt Debattista, Alan Chalmers, and Tom Troscianko. Auditory bias of visual attention for perceptually-guided selective rendering of animations. In *Proceedings of the 3rd International Conference on Computer Graphics and Interactive Techniques in Australasia and South East Asia*, GRAPHITE '05, pages 363–369, New York, NY, USA, 2005. ACM.

[113] Susana Mata, Luis Pastor, José Juan Aliaga, and Angel Rodríguez. Incorporating visual attention into mesh simplification techniques. In *Proceedings of the 4th Symposium on Applied Perception in Graphics and Visualization*, APGV '07, pages 134–134, New York, NY, USA, 2007. ACM.

[114] Rachel McDonnell, Michéal Larkin, Benjamín Hernández, Isaac Rudomin, and Carol O'Sullivan. Eye-catching crowds: saliency based selective variation. *ACM Transactions on Graphics (TOG)*, 28:55:1–55:10, July 2009.

[115] Ann McNamara, Reynold Bailey, and Cindy Grimm. Improving search task performance using subtle gaze direction. In *APGV '08: Proceedings of the 5th symposium on Applied perception in graphics and visualization*, pages 51–56, New York, NY, USA, 2008. ACM.

[116] Ann McNamara, Reynold Bailey, and Cindy Grimm. Search task performance using subtle gaze direction with the presence of distractions. *ACM Transaction on Applied Perception*, 6(3):1–19, 2009.

[117] Ann McNamara, Thomas Booth, Srinivas Sridharan, Stephen Caffey, Cindy Grimm, and Reynold Bailey. Directing gaze in narrative art. In *Proceedings of the ACM Symposium on Applied Perception*, SAP '12, pages 63–70, New York, NY, USA, 2012. ACM.

[118] H Murphy, A T Gutierrez, O Anson, F Banterle, and A G Chalmers. Perceptual rendering of participating media. *ACM Transactions on Applied Perception*, 4(3):15, 2007.

[119] Norman Murray, Dave Roberts, Anthony Steed, Paul Sharkey, Paul Dickerson, and John Rae. An assessment of eye-gaze potential within immersive virtual environments. *ACM Trans. Multimedia Comput. Commun. Appl.*, 3(4):8:1–8:17, December 2007.

[120] Karol Myszkowski. The visible differences predictor: Applications to global illumination problems. In George Drettakis and Nelson L. Max, editors, *Rendering Techniques*, pages 223–236. Springer, 1998.

[121] Lennart Nacke, Craig Lindley, and Sophie Stellmach. Log who's playing: Psychophysiological game analysis made easy through event logging. In *Fun and Games*, volume 5294 of *Lecture Notes in Computer Science*, pages 150–157. Springer Berlin / Heidelberg, 2008.

[122] Lennart E. Nacke, Sophie Stellmach, Dennis Sasse, Joerg Niesenhaus, and Raimund Dachselt. Laif: A logging and interaction framework for gaze-based interfaces in virtual entertainment environments. *Entertainment Computing*, 2(4):265 – 273, 2011.

[123] V Navalpakkam and L Itti. Modeling the influence of task on attention. *Vision Research*, 45(2):205–231, 2005.

[124] Toshikazu Ohshima, Hiroyuki Yamamoto, and Hideyuki Tamura. Gaze-directed adaptive rendering for interacting with virtual space. In *Proceedings of the 1996 Virtual Reality Annual International Symposium (VRAIS 96)*, VRAIS '96, pages 103–, Washington, DC, USA, 1996. IEEE Computer Society.

[125] Thomas Oskam, Alexander Hornung, Huw Bowles, Kenny Mitchell, and Markus Gross. Oscam - optimized stereoscopic camera control for interactive 3d. *ACM Trans. Graph.*, 30(6):189:1–189:8, December 2011.

[126] Carol O'Sullivan and John Dingliana. Collisions and perception. *ACM Trans. Graph.*, 20(3):151–168, July 2001.

[127] Frank Papenmeier and Markus Huff. DynAOI: a tool for matching eye-movement data with dynamic areas of interest in animations and movies. *Behavior Research Methods*, 42(1):179–187, February 2010.

[128] Michael A. Paradiso, Dar Meshi, Jordan Pisarcik, and Samuel Levine. Eye movements reset visual perception. *Journal of Vision*, 12(13), December 2012.

[129] Robert Patterson. Review paper: Human factors of stereo displays: An update. *Journal of the Society for Information Display*, 17(12):987, 2009.

[130] Jeff B. Pelz and Roxanne Canosa. Oculomotor behavior and perceptual strategies in complex tasks. *Vision Research*, 41(2):3587–96, 2001.

[131] Jerome Perrin, Philippe Fuchs, Corinne Roumes, and Francois Perret. Improvement of stereoscopic comfort through control of the disparity and of the spatial frequency content. In *Proc. SPIE 3387,Visual Information Processing VII,124*, pages 124–134, 1998.

[132] R J Peters and L Itti. Beyond bottom-up: Incorporating task-dependent influences into a computational model of spatial attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[133] Robert J. Peters and Laurent Itti. Applying computational tools to predict gaze direction in interactive visual environments. *ACM Transactions on Applied Perception*, 5(2):1–19, 2008.

[134] Thies Pfeiffer. Measuring and visualizing attention in space with 3d attention volumes. In *Proceedings of the Symposium on Eye Tracking Research and Applications*, ETRA '12, pages 29–36, New York, NY, USA, 2012. ACM.

[135] Thies Pfeiffer, Marc Erich Latoschik, and Ipke Wachsmuth. Evaluation of binocular eye trackers and algorithms for 3d gaze interaction in virtual reality environments. *Journal of Virtual Reality and Broadcasting*, 5(16), 2009.

[136] Yair Pinto, Andries R. van der Leij, Ilja G. Sligte, Victor A. F. Lamme, and H. Steven Scholte. Bottom-up and top-down attention are independent. *Journal of Vision*, 13(3), 2013.

[137] M. I. Posner and S. E. Petersen. The attention system of the human brain. *Annual review of neuroscience*, 13(1):25–42, 1990.

[138] M. I. Posner, R. D. Rafal, and Y. Cohen. Neural systems control of spatial orienting. *Philosophical Transactions of the Royal Society London*, B298:187–198, 1982.

[139] Y. Pritch, M. Ben-Ezra, and S. Peleg. Automatic disparity control in stereo panoramas (omnistereo). In *Proceedings of the IEEE Workshop on Omnidirectional Vision*, pages 54–61, 2000.

[140] Susanto Rahardja, Farzam Farbiz, Corey Manders, Huang Zhiyong, Jamie Ng Suat Ling, Ishtiaq Rasool Khan, Ong Ee Ping, and Song Peng. Eye HDR: Gaze-Adaptive System for Displaying High-Dynamic-Range Images. In *ACM SIGGRAPH ASIA 2009 Art Gallery & Emerging Technologies: Adaptation*, SIGGRAPH ASIA '09, pages 68–68, New York, NY, USA, 2009. ACM.

[141] Ganesh Ramanarayanan, Kavita Bala, and James A. Ferwerda. Perception of complex aggregates. *ACM Transactions on Graphics*, 27(3):60:1–60:10, August 2008.

[142] R. Ramloll, C. Trepagnier, M. Sebrechts, and J. Beedasy. Gaze data visualization tools: opportunities and challenges. In *Information Visualisation, 2004. IV 2004. Proceedings. Eighth International Conference on*, pages 173–180, july 2004.

[143] Daniel C. Richardson and Michael J. Spivey. Eye-tracking: Characteristics and methods. In G. Wnek and G. Bowlin, editors, *Encyclopedia of Biomaterials and Biomedical Engineering*, chapter 99. informaworld.com, London, 2004.

[144] G. Rizzolatti, L. Riggio, I. Dascola, and C. Umilta. Reorienting attention across the horizontal and vertical meridians — evidence in favor of a premotor theory of attention. *Neuropsychologia*, 25:31–40, 1987.

[145] Jonas Ruesch, Manuel Lopes, Alexandre Bernardino, Jonas Hörnstein, Jose Santos-Victor, and Rolf Pfeifer. Multimodal saliency-based bottom-up attention a framework for the humanoid robot icub. In *ICRA*, pages 962–967. IEEE, 2008.

[146] Dario D. Salvucci and Joseph H. Goldberg. Identifying fixations and saccades in Eye-Tracking protocols. In *Proceedings of the 2000 symposium on Eye tracking research & applications*, pages 71–78. ACM Press, 2000.

[147] D. Sasse. A framework for psychophysiological data acquisition in digital games. Master's thesis, Otto-von-Guericke-University Magdeburg, 2008.

[148] C. Sennersten and C. Lindley. Evaluation of real-time eye gaze logging by a 3d game engine. In *In 12th IMEKO TC1 & TC7 Joint Symposium on Man Science and Measurement*, pages 161–168, 2008.

[149] Charlotte Sennersten and Craig Lindley. An investigation of visual attention in fps computer gameplay. In *Conference in Games and Virtual Worlds for Serious Applications, VS-GAMES '09*, pages 68–75, 2009.

[150] Zhenfeng Shi, Hao Luo, and Xiamu Niu. Saliency-based structural degradation evaluation of 3d mesh simplification. *IEICE Electronics Express*, 8(3):161–167, 2011.

[151] G. L. Shulman, R. W. Remington, and J. P. McLean. Moving attention through visual space. *Journal of Experimental Psychology: Human Perception and Performance*, 5:522–526, 1979.

[152] Daniel J. Simons and Christopher F. Chabris. Gorillas in our midst: Sustained inattentional blindness for dynamic events. *Perception*, 28:1059–74, 1999.

[153] Daniel J. Simons and Daniel T. Levin. Change blindness. *Trends in Cognitive Sciences*, 1(7):261–267, 1997.

[154] Ran Song, Yonghuai Liu, Yitian Zhao, R. R. Martin, and P. L. Rosin. Conditional random field-based mesh saliency. In *Image Processing (ICIP), 2012 19th IEEE International Conference on*, pages 637–640. IEEE, 2012.

[155] Srinivas Sridharan, Reynold Bailey, Ann McNamara, and Cindy Grimm. Subtle gaze manipulation for improved mammography training. In *Eye Tracking Research and Applications*, page 112, March 2012.

[156] India Starker and Richard A. Bolt. A gaze-responsive self-disclosing display. In *CHI '90: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 3–10, New York, NY, USA, 1990. ACM.

[157] Sophie Stellmach. Visual analysis of gaze data in virtual environments. Master's thesis, Otto-von-Guericke-University Magdeburg, 2009.

[158] Sophie Stellmach, Lennart Nacke, and Raimund Dachselt. 3D attentional maps: aggregated gaze visualizations in three-dimensional virtual environments. In *Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI 2010)*, page 345. ACM Press, 2010.

[159] Sophie Stellmach, Lennart Nacke, and Raimund Dachselt. 3d attentional maps: aggregated gaze visualizations in three-dimensional virtual environments. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '10, pages 345–348, New York, NY, USA, 2010. ACM.

[160] Sophie Stellmach, Lennart Nacke, and Raimund Dachselt. Advanced gaze visualizations for three-dimensional virtual environments. In *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*, ETRA '10, pages 109–112, New York, NY, USA, 2010. ACM.

[161] Sophie Stellmach, Lennart E. Nacke, Raimund Dachselt, and Craig A. Lindley. Trends and techniques in visual gaze analysis. *CoRR*, abs/1004.0258, 2010.

[162] William Steptoe, Robin Wolff, Alessio Murgia, Estefania Guimaraes, John Rae, Paul Sharkey, David Roberts, and Anthony Steed. Eye-tracking for avatar eye-gaze and interactional analysis in immersive collaborative virtual environments. In *Proceedings of the 2008 ACM Conference on Computer Supported Cooperative Work*, CSCW '08, pages 197–200, New York, NY, USA, 2008. ACM.

[163] Geng Sun and Nick Holliman. Evaluating methods for controlling depth perception in stereoscopic cinematography. In *Proc. SPIE 7237, Stereoscopic Displays and Applications XX, 72370I*, pages 72370I–72370I–12, February 2009.

[164] V Sundstedt. *Rendering and Validation of High-Fidelity Graphics using Region-of-Interest*. PhD thesis, University of Bristol, 2007.

[165] V. Sundstedt. *Gazing at Games: An Introduction to Eye Tracking Control*. Synthesis digital library of engineering and computer science. Morgan & Claypool, 2012.

[166] V Sundstedt, K Debattista, and A Chalmers. Selective rendering using task-importance maps. In *Proceedings of the 1st Symposium on Applied Perception in Graphics and Visualization*, pages 175–175, 2004.

[167] V. Sundstedt, K. Debattista, P. Longhurst, A. Chalmers, and T. Troscianko. Visual attention for efficient high-fidelity graphics. In *Proceedings of the 21st Spring Conference on Computer Graphics*, SCCG '05, pages 169–175, New York, NY, USA, 2005. ACM.

[168] Veronica Sundstedt, Matthias Bernhard, Efstathios Stavrakis, Erik Reinhard, and Michael Wimmer. Visual attention and gaze behavior in games: An object-based approach. In Magy Seif El-Nasr, Anders Drachen, and Alessandro Canossa, editors, *Game Telemetry and Metrics: Maximizing the Value of User Data*, pages 543–583. Springer London, London, 2013.

[169] Veronica Sundstedt, Alan Chalmers, Kirsten Cater, and Kurt Debattista. Top-down visual attention for efficient rendering of task related scenes. In *In Vision, Modeling and Visualization*, pages 209–216, 2004.

[170] Veronica Sundstedt, Efstathios Stavrakis, Michael Wimmer, and Erik Reinhard. A psychophysical study of fixation behavior in a computer game. In Sarah Creem-Regehr and Karol Myszkowski, editors, *Proceedings of the 5th Symposium on Applied Perception in Graphics and Visualization*, pages 43–50. ACM, Aug 2008.

[171] K. Talmi and J. Liu. Eye and gaze tracking for visually controlled interactive stereoscopic displays. *Signal Processing: Image Communication*, 14(10):799–810, 1999.

[172] W J Tam, F Speranza, S Yano, K Shimono, and H Ono. Stereoscopic 3D-TV: visual comfort. *IEEE Transactions on Broadcasting*, 57(2):335–346, June 2011.

[173] 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, pages 265–272, New York, NY, USA, 2000. ACM.

[174] Krzysztof Templin, Piotr Didyk, Karol Myszkowski, Mohamed M. Hefeeda, Hans-Peter Seidel, and Wojciech Matusik. Modeling and optimizing eye vergence response to stereoscopic cuts. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 33(4), 2014.

[175] J.R. Tole and L.R. Young. *Digital Filters for Saccade and Fixation Detection*, pages 185–199. Lawrence Erlbaum, Hillsdale, NJ, 1981.

[176] C. Tomasi and R. Manduchi. Bilateral filtering for gray and color images. In *Proceedings of the Sixth International Conference on Computer Vision (ICCV'98)*, pages 839–846. IEEE Computer Society, 1998.

[177] A. M. Treisman. The perception of features and objects. *Attention: Selection, awareness, and control*, 123:5–35, 1993.

[178] Anne M. Treisman and Garry Gelade. A feature-integration theory of attention. *Cognitive Psychology*, 12(1):97–136, January 1980.

[179] Kun-Lung Tseng, Wei-Jia Huang, An-Chun Luo, Wei-Hao Huang, Yin-Chun Yeh, and Wen-Chao Chen. Automatically optimizing stereo camera system based on 3d cinematography principles. In *3DTV-Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), 2012*, pages 1–4, 2012.

[180] S. P. Vecera and M. J. Farah. Does visual attention select objects or locations? *Journal of Experimental Psychology: General*, 123:146–160, 1995.

[181] L.F. von Helmholtz. Physiological optics (1896 - 2nd german edition, translated by m. mackeben). *Vision Research*, 29(11):1631 – 1647, 1989.

[182] Oleg Špakov. Comparison of gaze-to-objects mapping algorithms. In *Proceedings of the 1st Conference on Novel Gaze-Controlled Applications*, NGCA'11, pages 6:1–6:8, New York, NY, USA, 2011. ACM.

[183] Manuela Waldner, Mathieu Le Muzic, Matthias Bernhard, Werner Purgathofer, and Ivan Viola. Attractive flicker: Guiding attention in dynamic narrative visualizations. *IEEE Transaction on Visualization and Computer Graphics (Proc. SciVis)*, To Appear, 2014.

[184] Dirk Walther and Christof Koch. Modeling attention to salient proto-objects. *Neural Networks*, 19(9):1395–1407, November 2006.

[185] Chiao Wang and Alexander A. Sawchuk. Disparity manipulation for stereo images and video. In Andrew J. Woods, Nicolas S. Holliman, and John O. Merritt, editors, *Proc. SPIE 6803, Stereoscopic Displays and Applications XIX*. SPIE, 2008.

[186] Andrew B. Watson and Denis G. Pelli. Quest: A Bayesian adaptive psychometric method. *Attention Perception & Psychophysics*, 33:113–120, 1983.

[187] Hank Weghorst, Gary Hooper, and Donald P. Greenberg. Improved computational methods for ray tracing. *ACM Transactions on Graphics*, 3(1):52–69, January 1984.

[188] G. Westheimer and D.E. Mitchell. The sensory stimulus for disjunctive eye movements. *Vision research*, 1969.

[189] Jeremy Wolfe. *Seeing*, chapter 8, pages 335–386. Handbook of Perception and Cognition. Academic Press, 2nd edition, 2000.

[190] Jeremy M. Wolfe. Guided search 2.0: A revised model of visual search. *Psychonomic Bulletin and Review*, 1(2):202–238, 1994.

[191] David S. Wooding. Fixation maps: quantifying eye-movement traces. In *Proceedings of the 2002 symposium on Eye tracking research & applications*, ETRA '02, pages 31–36, New York, NY, USA, 2002. ACM.

[192] Jinliang Wu, Xiaoyong Shen, Wei Zhu, and Ligang Liu. Mesh saliency with global rarity. *Graphical Models*, 75(5):255 – 264, 2013.

[193] Songhua Xu, Hao Jiang, and Francis C.M. Lau. Personalized online document, image and video recommendation via commodity eye-tracking. In *Proceedings of the 2008 ACM conference on Recommender systems*, RecSys '08, pages 83–90, New York, NY, USA, 2008. ACM.

[194] V. Yanulevskaya, J. R. R. Uijlings, J. M. Geusebroek, N. Sebe, and A. W. M. Smeulders. A proto-object-based computational model for visual saliency. *Journal of Vision*, 13(3), 2013.

[195] A. L. Yarbus. Eye movements during perception of complex objects. In *Eye Movements and Vision*, pages 171–196, New York, 1967. Plenum Press.

[196] Hector Yee, Sumanita Pattanaik, and Donald P. Greenberg. Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. *ACM Transactions on Graphics*, 20(1):39–65, January 2001.

[197] Kenji Yokoi, Katsumi Watanabe, and Takashi Kawai. Dynamic evaluation of distribution of visual attention during playing video game. In *Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology*, ACE '06, New York, NY, USA, 2006. ACM.

[198] LaurenceR. Young and David Sheena. Survey of eye movement recording methods. *Behavior Research Methods & Instrumentation*, 7(5):397–429, 1975.

[199] V. Zammitto, M. Seif El-Nasr, and P. Newton. Exploring quantitative methods for evaluating sports games. In *CHI 2010 Workshop on Brain, Body and Bytes: Psychophysiological User Interaction*, 2010.

[200] Xinyong Zhang, Xiangshi Ren, and Hongbin Zha. Improving eye cursor's stability for eye pointing tasks. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 525–534, New York, NY, USA, 2008. ACM.

[201] Alexandros Zotos, Katerina Mania, and Nicholaos Mourkoussis. A schema-based selective rendering framework. In *Proceedings of the 6th Symposium on Applied Perception in Graphics and Visualization*, APGV '09, pages 85–92, New York, NY, USA, 2009. ACM.