



Terms and Conditions of Use of Digitised Theses from Trinity College Library Dublin

Copyright statement

All material supplied by Trinity College Library is protected by copyright (under the Copyright and Related Rights Act, 2000 as amended) and other relevant Intellectual Property Rights. By accessing and using a Digitised Thesis from Trinity College Library you acknowledge that all Intellectual Property Rights in any Works supplied are the sole and exclusive property of the copyright and/or other IPR holder. Specific copyright holders may not be explicitly identified. Use of materials from other sources within a thesis should not be construed as a claim over them.

A non-exclusive, non-transferable licence is hereby granted to those using or reproducing, in whole or in part, the material for valid purposes, providing the copyright owners are acknowledged using the normal conventions. Where specific permission to use material is required, this is identified and such permission must be sought from the copyright holder or agency cited.

Liability statement

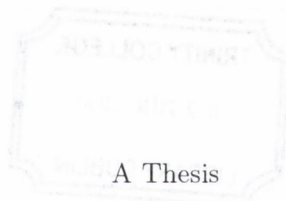
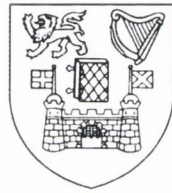
By using a Digitised Thesis, I accept that Trinity College Dublin bears no legal responsibility for the accuracy, legality or comprehensiveness of materials contained within the thesis, and that Trinity College Dublin accepts no liability for indirect, consequential, or incidental, damages or losses arising from use of the thesis for whatever reason. Information located in a thesis may be subject to specific use constraints, details of which may not be explicitly described. It is the responsibility of potential and actual users to be aware of such constraints and to abide by them. By making use of material from a digitised thesis, you accept these copyright and disclaimer provisions. Where it is brought to the attention of Trinity College Library that there may be a breach of copyright or other restraint, it is the policy to withdraw or take down access to a thesis while the issue is being resolved.

Access Agreement

By using a Digitised Thesis from Trinity College Library you are bound by the following Terms & Conditions. Please read them carefully.

I have read and I understand the following statement: All material supplied via a Digitised Thesis from Trinity College Library is protected by copyright and other intellectual property rights, and duplication or sale of all or part of any of a thesis is not permitted, except that material may be duplicated by you for your research use or for educational purposes in electronic or print form providing the copyright owners are acknowledged using the normal conventions. You must obtain permission for any other use. Electronic or print copies may not be offered, whether for sale or otherwise to anyone. This copy has been supplied on the understanding that it is copyright material and that no quotation from the thesis may be published without proper acknowledgement.

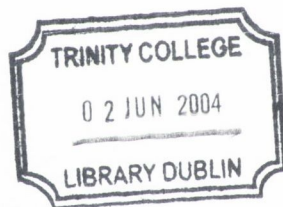
BOTTOM-UP VISUAL ATTENTION FOR
AUTONOMOUS VIRTUAL HUMAN ANIMATION



A Thesis

Submitted to the Office of Graduate Studies
of
Trinity College Dublin
in Candidacy for the Degree of
Doctor of Philosophy

Christopher Peters
March 2004

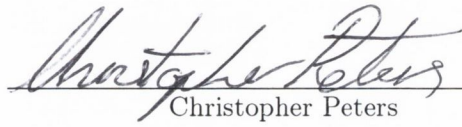


THOS
7987
7518

© Copyright by Christopher Peters 2004
All Rights Reserved

Declaration

This thesis has not been submitted as an exercise for a degree at any other University. Except where otherwise stated, the work described herein has been carried out by the author alone. This thesis may be borrowed or copied upon request with the permission of the Librarian, Trinity College, University of Dublin. The copyright belongs jointly to the University of Dublin and Christopher Peters.


Christopher Peters

Abstract

Animating autonomous virtual humans in a plausible manner is a difficult proposition. As social creatures, humans must be able to interact with each other from an early age and communication is often subject to many nuances. Humans therefore need to be relatively adept at judging the behaviour of others in order to be socially cohesive.

Appropriate gaze animations are fundamental ingredients in synthesising plausible behaviour for virtual humans. Gaze is not only a carrier of multiple types of communication signals, but is also used to directly orient our most important sense, vision, and to focus our attention towards locations of interest in our environment. It is intimately related to the way in which humans attribute intentions to the creature conducting the looking behaviour. If virtual humans are to appear plausible with respect to human viewers, then they should be seen to peruse their environment. Furthermore, if virtual humans are expected to be autonomous, then synthetic senses must be oriented in sensible manner.

We present a system for the automatic generation of gaze motions based on a stimulus oriented visual attention model for virtual humans in a virtual reality environment. Our system is composed of four interacting components: synthetic vision, memory and attention components and a gaze generator. The synthetic vision component takes a number of snapshots of the environment from the viewpoint of the virtual human and processes them in order to provide both scene and object based perception, something that is useful for internal analysis and storage of the visible environment. Perceptual information is then processed by a biologically plausible model of bottom-up visual attention, adopted from the field of cognitive engineering. The output of this model is a two dimensional map, called a *saliency map*, that encodes ‘pop-out’ over an actor’s view-field based on local feature contrast. This is used as an indicator of the regions in a scene that may elicit one’s attention during free viewing conditions when no task is at hand. A model of memory, based on the influential *stage theory* of memory from the field of psychology, provides an object based storage mechanism for recording stimuli that the character has seen and to what extent they have been attended to. Interactions between memory and attention are modelled in order to produce an *attention map*: the product of the saliency map and visual scene memory information. This map is used to generate

artificial regions of interest within the view-field that, when transformed into world coordinates, correspond to artificial fixations mimicking human scanpaths.

Artificial fixation information is utilised by a novel gaze generator algorithm that, when combined with gaze-evoked blinking, produces plausible gaze animations towards fixation targets and regions of interest.

Acknowledgments

I'd like to thank all of those, past and present, in the Image Synthesis Group, as both colleagues and friends, for their encouragement, provocative discussions and captivating experiences.

Most of all, I am deeply indebted to Carol O' Sullivan, my supervisor, for her guidance, advice and support over the past four years.

My gratitude also goes to my close family: Margaret, Diane and my cat, Socks, for their steadfast commitment and encouragement in all of my pursuits.

Contents

1	Introduction	1
1.1	Motivation	1
1.2	Overview	2
1.2.1	Problem Overview	3
1.2.2	Approach Taken	3
1.3	Scope	5
1.3.1	Our Virtual Humans	5
1.3.2	Problem Domain	7
1.4	Methodology	8
1.5	Contribution	8
1.6	Summary of Chapters	9
2	Background and Related Work	11
2.1	Character Animation	12
2.1.1	Character Representation	12
2.1.2	Skeletal Animation	13
2.2	Behaviour and Cognition for Animation	14
2.2.1	Behavioural Animation	14
2.2.2	Cognitive Modelling	15
2.3	Sensing and Perception	16
2.3.1	Animal Characters	17
2.3.2	Human Characters	18
2.4	Visual Attention	20
2.4.1	Cognitive Applications	20
2.4.2	Conversational Gaze Behaviour	21
2.4.3	General Gaze Behaviour	21

2.5	Summary	23
3	Psychological Basis	25
3.1	Visual Sensing	25
3.1.1	The Human Eye	26
3.2	Visual Attention	28
3.2.1	Bottom-Up Visual Attention	30
3.2.2	Centre-surround Processing	30
3.3	Memory	33
3.3.1	Stage Theory	33
3.4	Gaze and Blinking	36
3.4.1	Eye Movements	37
3.4.2	Gaze Shifts	38
3.4.3	Blinking	38
3.5	Summary	40
4	Synthetic Vision and Memory	41
4.1	Introduction	41
4.2	Synthetic Sensing	42
4.2.1	Synthetic Vision	44
4.2.2	Scene Perception	49
4.3	Synthetic Memory	56
4.3.1	Memory Entries and Observations	57
4.3.2	Forgetting	59
4.3.3	Stages of Memory	61
4.4	Discussion	66
5	Visual Attention	72
5.1	Introduction	72
5.2	Attention	73
5.2.1	Visual Attention	74
5.3	Modelling Visual Attention	77
5.3.1	Center-surround Processing	79
5.3.2	Inhibition of Return	87
5.4	Implementation	88

5.4.1	Dynamic Scenes	90
5.4.2	Object-Based Attention	91
5.5	Discussion	96
6	Attentive Behaviours	101
6.1	Introduction	101
6.2	Gaze	102
6.2.1	Eye and Head Movements	103
6.3	Blinking	109
6.3.1	Gaze-Evoked Blinking	111
6.4	High-Level Behaviours	113
6.4.1	Vision-based Reaching	113
6.4.2	Integration with Attention	117
6.4.3	Curiosity And Exploratory Gaze Behaviour	121
6.5	Discussion	123
7	Evaluation	126
7.1	Results	126
7.2	Speed Evaluation	135
7.3	Evaluation of Gaze Targets	144
7.3.1	Gaze Targets for Motion	146
7.3.2	Data Analysis	149
7.4	Evaluation of Gaze Animations	154
7.4.1	Evaluation	163
7.4.2	Results and Analysis	163
7.5	Discussion	167
8	Conclusions and Future Work	170
8.1	Contributions	170
8.2	Limitations and Future Work	171
8.2.1	Synthetic Vision and Perception	171
8.2.2	Synthetic Memory	172
8.2.3	Visual Attention	173
8.2.4	Attentive Behaviours	174
	Bibliography	176

List of Figures

1.1	Overview of visual attention system.	4
3.1	A close-up view of the human eye, noting the iris, pupil and sclera. . .	26
3.2	Cross section view of the human eye.	27
3.3	Colour, intensity and orientation feature dimensions.	29
3.4	On-off and off-on receptor fields of ganglion cells in the retina and LGN.	31
3.5	Edge and line receptor fields from patterns of ganglion cells.	32
3.6	Illustration of the neural pathways involved in visual processing. . .	32
3.7	Schematic of the stage theory of memory.	34
3.8	Illustration of the field-of-view of a human and a horse.	39
4.1	Diagram illustrating successive stages of filtering.	43
4.2	Synthetic vision acting as the first filter for information in the environment.	44
4.3	Depiction of object false-coloured rendering.	46
4.4	Illustration of the different snapshots taken from the viewpoint of the agent during a single update of the visual sensing module.	47
4.5	The origin and field-of-view of the agent's view frustum.	48
4.6	Depiction of proximal stimuli which are those energy patterns impinging on an observer's sensory receptors.	52
4.7	Perceptual grouping based on proximity and colour using a false-colour and full-scene rendering.	55
4.8	Successive stages of filtering. Synthetic vision and attention act as filters, selecting only certain information for further processing by the virtual human.	62

5.1	Attention acts as a filter to determine what items are entered into short-term memory.	74
5.2	Photograph of a natural, cluttered scene. Such scenes may be processed for saliency by combining feature maps for dimensions such as colour, orientation and intensity.	76
5.3	Feature dimensions for colour contrast, orientation contrast and conjunction of orientation and colour.	76
5.4	Saliency map for an input scene and saliency map superimposed on scene.	77
5.5	High-level schematic of the processing performed by the attention model.	78
5.6	An input image and its constituent channels.	81
5.7	Gaussian pyramid for an input image based on the intensity channel of the image.	82
5.8	Illustration of Gabor filters for the selection of orientations 0° , 45° , 90° and 135°	83
5.9	Gaussian and Gabor filter kernels, as used in the Gaussian and Gabor pyramids for center-surround processing.	84
5.10	Results of test runs for sample images from our implementation of the attention model.	85
5.11	Test runs of the attention model on rendered images from a virtual environment and a photograph.	86
5.12	Illustration of the normalisation process for a feature map containing multiple peaks of activity and the normalised map after suppression.	87
5.13	Illustration of the normalisation process for a feature map containing a single strong peak of activity and the normalised map after promotion.	87
5.14	Schematic of the attention model. An input image is split into consecutive channels, forming a number of dyadic image pyramids.	89
5.15	Four frames of animation from a city scene containing two moving vehicles.	92
5.16	The memory uncertainty map is constructed from the false-colour rendering and applied to the attention output to modulate the saliency map.	94

5.17	The uncertainty map provides a measure of uncertainty for a scene based on an agent's memory of its surroundings.	95
5.18	The modulated saliency map is obtained by point-by-point multiplication of the saliency map and the memory uncertainty map.	95
6.1	Virtual human configuration and the corresponding top-down view showing vectors that are used for determining gaze.	105
6.2	Calculation of extreme movement vectors when targets are outside of oculomotor range and outside of head movement range.	107
6.3	Two cases where the target is outside of the head movement range.	107
6.4	The resetting effect.	108
6.5	The final head vector is calculated as an interpolation between the extreme non-mover vector and the extreme head movement vector.	108
6.6	Illustration of midline attraction.	108
6.7	Illustration of blinking magnitudes for the virtual human.	112
6.8	Three frames of a reaching animation based on vision and memory.	114
6.9	Human fixations and scan paths from a scene in a virtual environment captured using eye tracking equipment.	118
6.10	Illustration of artificial fixation generation for a sample image.	120
6.11	Illustration of curiosity calculation for an input scene.	122
6.12	Illustration of two different curiosity thresholds for an unnormalised attention map, where scene uncertainty levels are used as height values.	122
7.1	Two screenshots from the street environment.	127
7.2	Screenshots taken from the street environment.	128
7.3	Successive frames of animation for gaze behaviours in the street scene	129
7.4	Successive frames of animation for gaze behaviours in the street scene, continued from Figure 7.3.	130
7.5	Screenshots from the pub scene.	131
7.6	Screenshots from the pub scene.	132
7.7	Successive frames of animation for gaze behaviours in the pub scene.	133
7.8	Successive frames of animation for gaze behaviours in the pub scene, continued from Figure 7.7.	134

7.9	Amount of time taken to render, read and process the three scene renderings in order to prepare them for stimulus extraction and attention processing.	136
7.10	Illustration of the amount of time taken to conduct each of the three synthetic vision renderings.	136
7.11	Graph of timing tests for uncertainty map creation.	138
7.12	Graph of timing tests for the extraction of peripheral stimuli	139
7.13	Graph of timing tests for the resolving peripheral stimuli.	141
7.14	Depiction of timings for saliency map generation in the street scene.	142
7.15	Overall timings for the synthetic vision and attention systems. . . .	143
7.16	Graph of timing tests for saliency map creation with a varying number of visible objects.	143
7.17	View of the SMI Eyelink eye-tracker head-mounted band used in the experiments.	146
7.18	The eye-tracker in operation on a test subject during a data collection.	147
7.19	Calibration phase of the eye-tracking experiment on the operators PC.	148
7.20	Images from trial a in the motion experiment.	155
7.21	Images from trial a in the motion experiment, continued.	156
7.22	Images from trial b in the motion experiment.	157
7.23	Images from trial b in the motion experiment, continued.	158
7.24	Images from trial c in the motion experiment.	159
7.25	Images from trial c in the motion experiment, continued.	160
7.26	Images from trial f in the motion experiment.	161
7.27	Images from trial f in the motion experiment, continued.	162
7.28	Results of gaze and blink animation experiments.	164

Chapter 1

Introduction

1.1 Motivation

Virtual humans are computer models of real people. There are a wide range of applications for computer models of people. They can be used as substitutes for real people in computer simulations of hazardous situations. Virtual stunt-doubles are now employed by special effects companies in Hollywood productions in order to reduce or eliminate the risks to real-life stunt-people. In this way, a stunt may also be filmed and edited a number of times without any increase in risk. Virtual humans are used in ergonomic evaluations of computer-based designs in order to save time and expense. For instance, rather than constructing a real prototype for an automobile and testing it with a real human before making design changes, a virtual human can be used with a computer-based model of the automobile, to test for the reachability and visibility of components. Another application for virtual humans is as embedded representations of ourselves into virtual environments. Finally, many recreations of the real world require or would benefit from the presence of some kind of human representation; many scenes are not complete or believable without virtual people. An architectural walk-through may utilise virtual people in order to provide a better feeling of authenticity and presence to the viewers.

The plausible synthesis of human-like characters has been a long standing goal in computer animation. It is not difficult to understand why human-like characters are an active area of research and will be for some time to come. Humans are complex, articulated objects, connected and constrained by varying joint types, self-propelled by groups of muscles and covered by a layer of skin. The number of articulations in

the body, the movement limits placed on the joints and the vigour of the muscles are a few of the many features that vary with age, sex and environment. The human adult has over two hundred bones, all affecting the mobility and style of the motion of the human.

To make matters worse, there is very little room for inconsistencies or error in human simulation. Life makes it necessary that every human becomes very capable at perceiving the realism of human motion, as it is something continually witnessed from our first days to our very last. Even the most minor flaws in an animation can leave the impression that ‘it doesn’t feel right’, even if the viewer cannot explain exactly what it was that was incorrect. Often, the subtleties in a motion are what make it believable.

In this thesis, we use the terms *virtual humans*, *human characters*, *synthetic humans* and *agents* interchangeably.

1.2 Overview

This thesis seeks to enrich the animation processes discussed previously by providing fast, automatic, low level gaze behaviours for the animation of human characters. Gaze behaviours are vital to animating characters in a convincing and realistic manner, yet many contemporary gaze animations must be either generated painstakingly by hand, or generated by automated systems that fall short of providing plausible attending behaviours. By *low level*, it is meant that these behaviours correspond to fundamental actions carried out by humans many times on a daily basis. From a behavioural animation standpoint, such actions form the basic building blocks of more complicated human motions. The advantages of such a system are clear: going back to the example of defining a goal such as “walk to the restaurant”, we wish for the character to do this in a realistic manner without having to specify every movement made. In terms of gaze, the animator would like the character to look at things that it finds interesting during the journey without having to specify exact targets, since they may not be known in advance, or may not be sensible.

Moreover, the system presented should be applicable to the animation of autonomous virtual humans, and should therefore be based on biologically plausible internal models of visual sensing, attention and memory. Autonomous actors require an intelligent automated control system for orienting their senses, since they

are dependent on their senses for informing them about their world.

The motivation and objective of this work can therefore be summarised as follows: real humans pay attention to their environment and look around; so too should virtual humans!

1.2.1 Problem Overview

This thesis tackles the problem of generating plausible gaze motions for a virtual human by dividing it into two subproblems:

1. Where to look

The first problem is the problem of where to look. Provided with an environment, what determines the locations where the virtual human will look? Why look in some locations rather than others? These locations should make some sort of sense and ideally be similar to where a real human would look under the same conditions.

2. How to look

The second problem is how the looking, or gaze motion, should take place. Given a target location in the environment and the virtual human's current configuration, how should a gaze motion be generated towards that location? This brings in the question of how much the head and eyes should contribute to the movement, as well as blink motions.

1.2.2 Approach Taken

Our approach to solving this problem is to design a system that is based on the interactions of four components: vision, memory, attention and gaze. These components approximate the visual sense and memory of an environment, process the environment in a biologically plausible manner for virtual humans and then direct gaze towards those locations deemed to be of interest (see Figure 1.1 for a schematic of the process). The vision, memory and attention components determine *where* an agent will look, while the gaze generator determines *how* the looking should be performed.

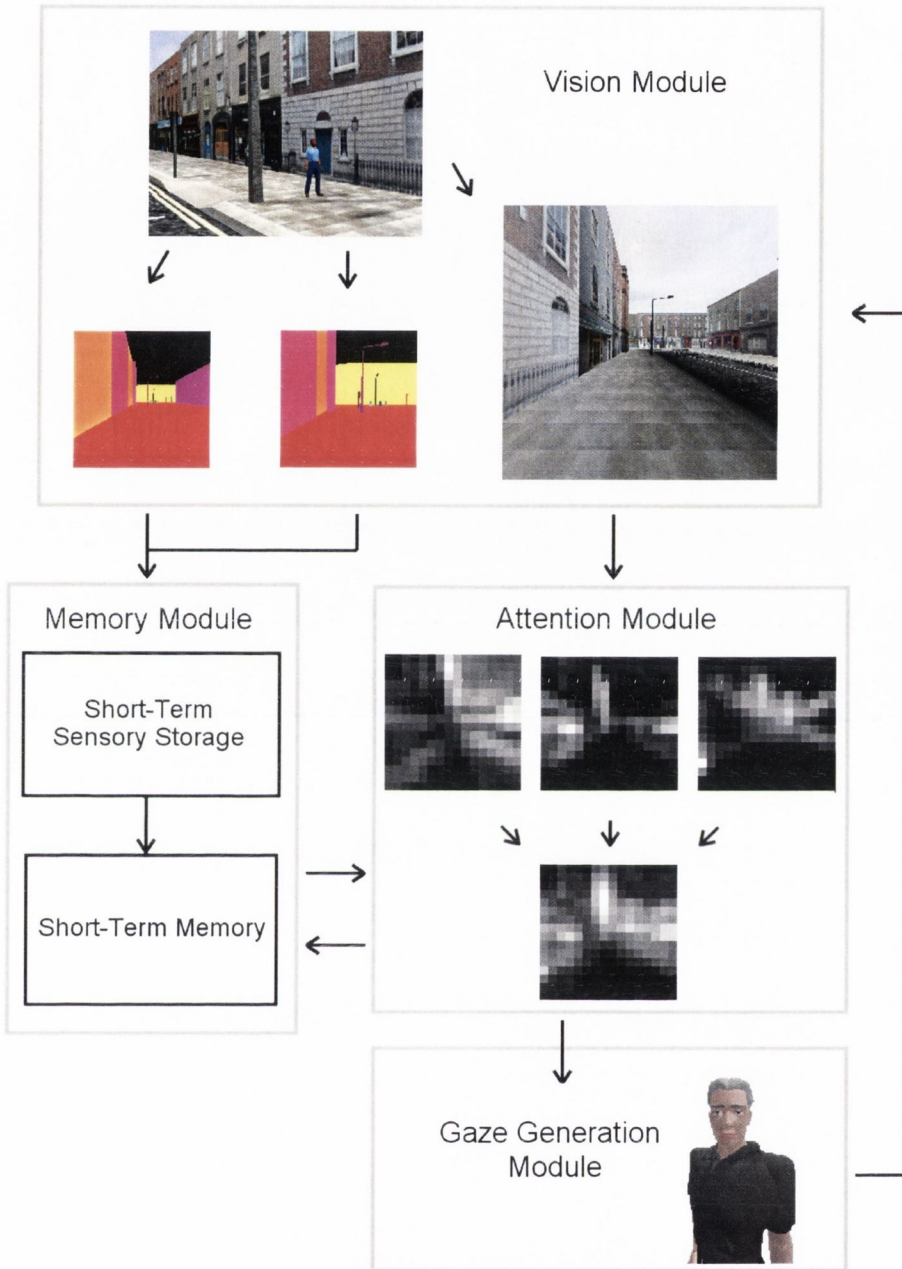


Figure 1.1: Overview of visual attention system. Scenes are processed by the synthetic vision module and passed to the memory and attention modules. After processing by the attention module and modulation by the memory component, target locations are passed to the gaze generator. The gaze generator creates eye-gaze motions towards such locations in a realistic manner.

1.3 Scope

The problem domain considered in this thesis encompasses the automatic generation of some gaze behaviours for autonomous virtual humans, also referred to as *human characters*, *synthetic humans* and *agents*, based on external stimuli in virtual reality settings corresponding to natural environments.

1.3.1 Our Virtual Humans

It is important at this stage to locate the work presented here with respect to the large amount of research already conducted on virtual humans and autonomous agents. Badler highlights five dimensions along which virtual human modelling can be characterised [Bad97]. What follows is a discussion of the importance of each of these dimensions with respect to the work presented in this thesis.

1. Appearance

In terms of appearance, virtual agents can be represented by 2D icons, cartoons, composite video, 3D shapes or full 3D bodies. The virtual humans presented in this thesis are composed of rigid-body 3D shapes. Although this thesis doesn't focus on appearance in the sense of body-modelling and deformations, 3D geometry was chosen so as to provide an indicator of the potential of the system presented here. Our two-layer virtual human definition consists of an underlying skeleton and a set of geometric primitives presenting the character's appearance. Although the result of the system presented here acts on only the skeletal layer, this system would be easily extendible to account for a more advanced skin and muscle definition.

2. Function

Virtual humans within this system are deemed to have human limitations within the problem domain. That is, the models provided here seek to impose similar limitations on virtual humans that exist with real humans. For example, in terms of sensory models, virtual humans should only be able to 'see' objects that are detected by their visual sensor. Like real humans, they should not be able to 'see' objects behind them.

3. Time

Time is an important limitation on the other dimensions. The nature of the computations involved in simulating and displaying virtual humans means that there will always be a trade-off between the time taken for computations and the fidelity of the virtual human's appearance and simulation. For movies, animations may be generated off-line, allowing months to be invested on the composition of the final motion. For real-time applications, such as video games, the task of generating plausible autonomous motions is much more demanding due to time constraints. We attempt to provide the generation of animations in a real-time manner for a single virtual human. To this end, simplified models from other fields, such as psychology, have been used.

4. Autonomy

Autonomy is another important issue with respect to the work presented here. The animation of computer characters may be done entirely by hand. Although this process is laborious, a team of professional animators will often produce excellent results. In contrast, we may wish to limit the amount of input required by the human animators. In this way, the generation of motions will be automatic. If characters are to generate all of their own motions, including high-level behaviours, then they must be equipped with some sort of intelligence. Our system allows gaze behaviours to be generated in a fully automatic manner.

5. Individuality

The final dimension, individuality, considers the reconstruction of a specific person, for example a television personality, as opposed to the generation of multiple, differing personalities, for use in crowd scenes. When generating animations for multiple characters, variation of behaviour is important to prevent the characters from appearing cloned and robotic in nature. This issue is relevant to gaze behaviours generated by our system, and indeed, behaviours generated by any computational model. We introduce variations in the head movements of different virtual humans so that gaze animations do not become monotonous.

1.3.2 Problem Domain

In relation to the computational models presented in this thesis, we aim for *plausibility* rather than *realism*, while attempting to reduce the computational expense of such models sufficiently to obtain real-time performance for the simulation and display of a virtual human. Many excellent, although for our purposes computationally prohibitive, methods exist in the domains of artificial intelligence and cognitive engineering for simulating various biological processes. To take an example, we simplify the object recognition process significantly in our system by cross-referencing with object information available in the environmental database: it is therefore presumed that virtual humans will always recognise objects in the environment. The view taken in this thesis is that, although more full-bodied implementations would undoubtedly contribute to the overall realism of the system, the results obtained in terms of animation plausibility would be overshadowed by the increase in computational expense.

The system presented here is intended to augment existing behavioural animation frameworks, providing early sensory, memory and attention capabilities that are used to drive higher level behaviours. Leading on from this, although this thesis seeks to contribute to the creation of intelligent virtual humans, this work taken in isolation results in *reactive* rather than *intelligent* behaviour. To this end, the effect of tasks and internal personality variables do not influence looking behaviours in our system. That is not to say the behaviours that are generated are not *natural*, but they are only a part of a more encompassing description of a character's full behaviour.

Although eye movements are a necessary component of gaze behaviours, we only consider eye movements as they relate to gaze motions, *saccadic gaze shifts*, and do not consider small-scale saccadic eye movement, such as microsaccades (for example, see [LBB02]). Thus, in our model, it is presumed that eye movements only take place in the context of saccadic gaze shifts triggered by shifts in attention. Such work is not mutually exclusive with respect to small scale eye movement models and would benefit from the addition of such models.

Recently, a large amount of research has been focused towards eye-gaze models for conversational agents (for example, see [PPdR00]). Unlike this research, we do not focus on eye-gaze specifically for social interactions, but for general situations. As such, we seek to solve the less-studied problem of stimulus-evoked gaze for

virtual humans in natural environments rather than gaze for virtual human social interactions.

Furthermore, we are interested only in gaze motions that take place under general viewing conditions when *no particular task is active* and *elicited by exogenous factors*. Therefore, the system presented here could not be considered as encompassing all modes of attention; only attention behaviours based on what is known as *bottom-up attention* in the psychology literature have been addressed. However, the methods presented here do provide a concrete basis for creating or augmenting such a system.

1.4 Methodology

The approach taken in this thesis has been to use research from psychology to create simplified models that are applicable to our purposes. Since the overall aim is to animate agents that act in a human-like manner, the field of psychology appears to be the perfect source of information on everything from high-level human behaviours, to memory and attention. Nonetheless, a great deal of information available in the field of psychology is too detailed in order to be directly applied to the computer graphics domain. Therefore, it has been necessary to simplify many models in order to achieve acceptable performance while still retaining the spirit of the original systems.

1.5 Contribution

This thesis presents a number of novel contributions to the domain of virtual human animation.

1. Enhance a current synthetic visual sensing technique in order to better account for visual perception while retaining computational flexibility. The visual sensor also allows for the amalgamation of object-based information, necessary for the memory component, with scene-based information from the attention component, to provide a system that supports both scene-based and object-based attention paradigms.
2. Present a synthetic memory model to filter and store visual information of relevance to the virtual human. The memory component further builds on the

concept of successive filtering of environmental information and acts to modulate the output of the attention module. The filtering in the memory module is also controlled by the attention component, both directly and indirectly, in order to create a feedback loop.

3. Introduce a biologically plausible model of bottom-up selective visual attention from the field of cognitive engineering to the domain of virtual human animation in order to act as both a control mechanism for filtering in memory and as a gaze controller for head and eye movements.
4. Present a psychologically inspired model of gaze and blink coordination for gaze animations generated by the system. Eye and head contribution to the final motions are based on head movement propensity in order to provide variation in the animations, while blinking is gaze-evoked, with eyelid closure probability and magnitude dependent on head movement amplitude.

1.6 Summary of Chapters

The remainder of this thesis is split into eight chapters. Chapter Two reviews related work in computer graphics on synthetic vision, memory and attention for the animation of computer characters. Chapter Three continues to review important background research that has provided the inspiration for the models of vision, memory, attention and gaze generation proposed in this work. Much of this work originates from the field of psychology.

Chapter Four presents our synthetic vision and memory components. These components are internal to the virtual human and provide a way of preparing, filtering and storing environmental information external to the virtual human. The synthetic vision component provides an input of the outside environment that is suitable for processing by the memory and attention modules. The memory component is used to keep track of information about remembered objects from the environment. This information is used to modulate where attention is directed to in the visual field.

The attention module is presented in Chapter Five. This module consists of a biologically inspired bottom-up model of visual attention from the field of cognitive engineering and is the main control process in the system. It processes the scene, provided by the synthetic vision component, in a bottom-up manner and outputs

a saliency map that encodes conspicuous locations. These locations are more likely to be looked at by the virtual human.

Chapter Six reviews the gaze generation module. The gaze generator consists of both the coordination of the eye and head movements in making the gaze motion and also the addition of blinking behaviours that relate to such gaze changes. It also considers how the attention module may be used to generate higher-level behaviours.

An evaluation of the proposed models is presented in Chapter Seven, followed by a discussion of limitations and future work in Chapter Eight.

Chapter 2

Background and Related Work

This chapter provides an overview of important work relating to synthetic perception, memory and visual attention for character behaviour and animation.

Computer animation is a wide and varied field. Entities within the graphical world may be animated in a number of different ways. They may be animated entirely by user input in a similar vein to traditional, hand-drawn cel animation. In this case, the user has full control over the final animation, but also has the tedious task of specifying state information for each entity over each frame of the animation. This has led to the development of computational techniques for automating some of the work that must be undertaken by the animator. These techniques range in sophistication from key-frame interpolation, used to automatically generate in-between entity states referred to as *tweens* in traditional hand animation, to physically-based simulation, requiring little input from the user and using physical laws and attributes to simulate the movement and interactions of entities in a virtual environment. The key goal here appears to be automaticity: we seek ways to automate the animation process so that the amount of user input required to adjust mundane details is minimised as much as necessary while still producing the desired results. Consider an animation of a rock-fall: using physically-based simulation methods, the animator can provide rock entities with differing attributes and starting states and then run a simulation to obtain the final animation. There are multiple benefits to using such an automatic technique. Given a large number of even simple objects such as rocks, a human animator would soon be overwhelmed by the number of possible interactions, as the rocks bounce and collide. Having to manually adjust the rotation and position of each rock in the simulation at each

frame would be extremely tedious. Furthermore, humans are not always adept at judging and recreating external physical phenomena from observation.

While such approaches are useful and often require intervention from a director or animator, unfortunately they are not always appropriate for simulating every type of entity and interaction. One particularly challenging area of research relates to the animation of what are meant to be living entities; such entities are not merely open to the effects of their environment, like rocks, but are also *reactive* and *proactive* in that they are expected to operate in a volitional and reasonable manner. In order to animate such entities, more complicated forms of control are required.

2.1 Character Animation

The animation of computer characters presents a number of challenges and problems for graphics practitioners. First of all, most characters consist of a hierarchy of linked entities and appendages. As the number of linkages increases, so too does the complexity of any simulation involving the character. Secondly, most characters are deformable in nature, forming complex, organic shapes that change depending on the configuration of the character's body. Finally, as mentioned above, characters are proactive in their environment as well as reactive. Living systems move under their own power as well as being affected by any changes in their environment. These difficulties have led to a number of methods for simulating the animation and appearance of such characters.

2.1.1 Character Representation

A layered approach to character animation helps to alleviate the first two problems of hierarchical models composed of deformable bodies. Using this approach, the animator divides the simulation model into a number of discrete layers, each with its own physical and geometric properties. The animator specifies the various constraint relationships between the layers and can then control global motions from a high level.

The simplest layered character approach consists of two layers: a *skeletal layer* and a *skin layer*. The skeletal layer defines a hierarchy of primitives and joints, each with a number of degrees of freedom. The configuration of this hierarchy determines the state or pose of the character. The skeletal layer is never displayed

directly: rather it is used as a tool to ease the control and manipulation of the figure by the animating processes. The skin layer forms a geometric representation of the character and is related to the skeletal layer. When appropriate constraints are applied between the skin layer and skeletal layer, animation only needs to take place upon the skeletal layer to affect the motion of the character; the character's appearance will be updated automatically. However, in order to achieve realistic motion and appearance, the skin layer must deform in accordance with the skeletal layer. For example, when a limb bends, the muscle on the limbs joint should bulge and the skin around the joint should crease in accordance with the real system.

2.1.2 Skeletal Animation

Given the layered approach described above, this thesis focuses primarily on the animation of the skeletal system. There are two broad approaches to animating the skeletal system. The first relies on recording motions prior to the animation using motion capture equipment and then playing the recorded animations back at runtime. Since the recordings are taken directly from the real-world counterpart of the character to be animated, in other words a real human for human character motion, resultant animations tend to be very realistic. Unfortunately, this technique also proves to be somewhat inflexible; if the location of the target or the character is not known before the animation, then one cannot pre-compute a grasping motion. This is often the situation in dynamic environments, where a general motion is needed, such as grasping or walking, but the exact situation varies from scene to scene. Motion editing has been used to alleviate this inflexibility somewhat, but it often proves to be tedious and unreliable.

The second approach requires the creation of an underlying computational model that attempts to synthesise the real-world system. Given a number of parameters, the computational model should generate the appropriate motion as output. For example, if we would like a character to be able to walk on different types of terrain with different slopes, we may construct a walk generator that accounts for the slope of the terrain as an input and outputs appropriate leg configurations. The main problem with this approach is in finding and constructing a suitable computational model: the resulting animation will always be based on the accuracy and speed of the underlying model.

Producing a good computational model for character animation is a difficult

prospect and inevitably draws from a vast amount of literature from a wide variety of sources, such as biomechanics and psychology. Consider the construction of a computational model for animating prehension, that is, reaching and grasping. Some difficult questions are raised in the construction of such a motion that span many fields. What is the anatomical structure of the arm? In what way is it constrained? When grasping a hammer, what is the most natural posture for the arm and the hand?

Although this approach often fails to provide results as good as the motion-capture method, it proves to be far more flexible for dynamic scenarios. Creating flexible, underlying models for motion generation facilitates the creation of higher levels of control that are useful for controlling and constructing autonomous characters. The creation of an internal computational model for generating gaze motions based on external stimuli is the focus of this thesis.

2.2 Behaviour and Cognition for Animation

An animation may be acceptable in terms of low level motions, but this only provides a partial solution to the animation problem. In a city scene with a number of pedestrians, although the walking animations of the pedestrians may be very plausible in synthesising natural walking motions, the animation will be ruined if the pedestrians frequently walk into lamps or under passing vehicles. The problem here is that the pedestrians are not behaving as real pedestrians would. The prehension motion discussed previously may appear to be realistic in terms of end-effector velocities and hand trajectory if we address the questions regarding the biomechanics of the arm and so forth, but if the hand is directed to a place that does not make sense, into a fire for example, then, again, the overall result will not be believable. In these cases, although the low level motions are plausible, external as well as internal considerations must be accounted for. Higher level control techniques such as behavioural animation and cognitive modelling attempt to address these problems.

2.2.1 Behavioural Animation

Behaviour is regarded as the being the actions or reactions of a person or animal to external and internal stimuli. Behavioural animation (see [BBZ91] and [BPW93]) seeks to provide high level control of entities by the specification of actions, tasks

and goals. It is based on a reactive system that takes the external environment into account. That is, a behavioural entity perceives its environment and reacts to it according to internal decisions based on rules. Because entities have internal models and make simple decisions based on external stimuli, behavioural animation is suited to the simulation of organisms and living beings. Simple actions may be combined in hierarchical manner to produce more complex actions. An example task might be defined as “take the bottle and walk to the restaurant”. The action of taking a bottle may further be split into looking at the bottle, reaching for it and lifting it up. Note that the animator does not need to specify lower control objectives, such as paths, motions, or at the lowest level, joint angles. Behavioural animation therefore provides a higher level of abstraction for character animation and interfaces with lower level control techniques, such as kinematics and dynamics, in order to produce the required motions. It should be noted that the abstraction provided by this technique is based on increased autonomy from underlying computational models: when asking the character to “walk to the restaurant”, we do not need to specify *how* it should get there. In other words, the underlying computational model must be capable of utilising or generating paths, walking motions and other natural motions in order to provide a reasonable animation. This may be achieved by endowing behavioural entities with some internal states. A simple example would be to provide a creature with a thirst variable that increased over time. A high thirst value may invoke a “search for water” behaviour, whereby the creature would attempt to walk around and look for a drink. In this way, a character gains some notion of autonomy: its actions may be driven fully by internally generated goals.

2.2.2 Cognitive Modelling

Behavioural animation considers self-animating characters that react to stimuli they perceive in their environment. Using this methodology, characters need to be intelligent in some way in order to animate their own actions. However, this can cause control problems when an animator would like a character to behave and act in a certain way. Furthermore, behaviour controllers are often hard-wired into code.

Cognitive modelling (see [FTT99] and [Fun98]) enables steps to be taken towards solving these problems by considering what a character knows, how the knowledge is acquired and how it can be used to plan actions. Cognitive modelling is divided into domain knowledge specification and character direction. Domain knowledge informs

a character about its environment and how it is liable to change. Character direction allows the definition of loose scripts, or *sketch plans*, for characters to follow. In this way, a character will behave within its world to achieve specific goals set by the animator while the other details are filled in automatically by the character.

In the same way that behavioural animation forms a higher level control technique, utilising lower level techniques such as kinematics and dynamics, cognitive modelling presents a higher level of control over behavioural animation.

As we will see in more detail in Section 2.3, a number of researchers have developed sensing and perception capabilities for virtual characters in order to provide them with internal views of their environment (for example [TT94] [Rey87] and [Blu97] provide internal systems for animal characters). These internal models attempt to approximate a type of awareness for characters and provide them with their own unique perspectives of their environment. Such internal models therefore often differ from the information contained the world database. In this way, the agent may only have knowledge about a limited number of objects in the world, i.e. those that have been sensed, and may not have perfect and complete information about such objects.

Visual attention and gaze motions have also been the focus of research for animating virtual characters in a plausible manner. An underlying attention model is usually employed that processes and prioritises those locations or objects that are deemed to be of interest to the characters; gaze and looking behaviours are the visible manifestations of these processes, so the attention model is of significance to the resulting animation of the character. Such a model may be specialised and suited to a particular role, for example, deciding where or who to look at during conversations. It may also be of a more general nature, deciding where to look at in the environment.

2.3 Sensing and Perception

This section reviews research concerning the endowment of agents with sensory and internal storage mechanisms. These mechanisms act to filter the amount of information available to the agent from the environment. There are a number of different ways of achieving this: a filtering system may be defined in such a way as to approximate specific senses, for example, visual and auditory sensing. The

purpose of limiting the information available to the target is to provide behaviours based on agent-centric, as opposed to environment-centric, views. Such behaviours may be more plausible as they parallel the real-world living systems where only an imperfect and partial view of the environment is available to an entity at any one time. For example, an agent shouldn't base its behaviour on an object that it hasn't sensed.

By implementing a memory mechanism, one may obtain a recollection of previously sensed objects. These objects need not be currently perceptible by the agent and, as such, stored information will not always be totally accurate. In this way, the agent may build an internal, if somewhat inaccurate, model of its surroundings. We will now look at how researchers have applied sensing and perception models to the creation of animal characters, such as virtual birds and fish, and how they have also been applied to the challenging domain of human characters.

2.3.1 Animal Characters

Reynolds [Rey87] presented a distributed behavioural model for flocks of birds and herds of animals. The method is based on the insight that elements of the real system (in this case, birds) do not have complete and perfect information about the world, and that these imperfections have a major impact on the final behaviour of the system. The system is based on simulated birds, or *boids*, which are similar in nature to the individual particles in a particle system. Each boid is implemented as an independent actor that navigates according to, among other things, its local perception of the dynamic environment. The boid model does not attempt to directly simulate sensing used by the real animals; rather it attempts to make the same final information available to the behavioural model that the real animal would receive as an end result of its perceptual and cognitive processing. Each boid has a spherical zone of sensitivity centred at its local origin, and the behaviours that comprise the flocking model are governed by nearby flock-mates. A key issue is raised here regarding behavioural animation: how does one analyse the success of the model? As noted in [Rey87], it is difficult to objectively measure how valid such simulations are. However, the flocks built from the model seem to correspond to the observer's intuitive notion of a *flock-like* motion. An interesting result of the experiments with the system reveal that the *flocking* behaviour that we intuitively recognise is not only improved by a limited, localized view of the world, but is dependent on it.

Tu and Terzopoulos [TT94] present a framework for animation featuring the realistic appearance, movement and behaviour of individual and schools of fish with minimal input from the animator. The fishes' repertoire of behaviours relies on their perception of the dynamic environment. Individual fish have motivations as well as simple reactive behaviour. At each time step, habit, mental state and sensory information are used to provide an intention. Behaviour routines are then executed based on this intention and motor controllers provide motions that fulfil these behaviours. Habits are represented as numerical variables for determining individual tendencies towards brightness, darkness, cold, warmth, and schooling. The individual fish also has three mental state variables for hunger, libido and fear. Behaviour patterns may be interrupted by reactions to more pressing environmental stimuli, for example, a predator. The artificial fish has two on-board sensors with which to perceive its environment - a vision sensor and a temperature sensor. The vision sensor is cyclopean covering a 300-degree spherical angle. An object is seen if any part of it enters this view volume and is not fully occluded by another object. The vision sensor has access to the geometry, material property, and illumination information that is available to the graphics pipeline and to the object database and physical simulation results for information such as identification and velocities of objects. The authors make a point of noting that, as their holistic computational model exceeds a certain level of physical, motor, perceptual and behavioural sophistication, the agent's range of functionality broadens due to emergent behaviours.

Blumberg [Blu97] presents an ethologically inspired approach to real-time obstacle avoidance and navigation. The creature renders the scene from its own viewpoint. This rendering is used to recover a gross measure of motion energy as well as other features of the environment, which are then used to guide movement. An approximate measure of motion energy is calculated for each half of the image, which is then used to provide corridor following and obstacle avoidance.

2.3.2 Human Characters

Renault et al. [RTT90] introduce a synthetic vision system for the high level animation of actors. The goal of the vision system in this case is to allow the actor to move in a corridor, avoiding objects and other synthetic actors. For the vision system, the scene is rendered from the point of view of the actor, and the output is stored in a 2D array. Objects in the scene are not rendered using their usual

colours, but are rendered using unique colours for each object. Each element in the 2D array consists of a vector containing the pixel at that point, the distance from the actor's eye to the pixel, and an object identifier of any object that is at that position. The size of the array is chosen so as to provide acceptable accuracy without consuming too much CPU time. A view resolution of 30x30 was selected for the corridor problem.

Noser et al. [NRTT95] extend the earlier model proposed in Renault et al. by adding memory and learning mechanisms. They consider the navigation problem as being comprised of two parts: global navigation and local navigation. Global navigation uses a simplified map to perform high-level path-planning. This map is somewhat simplified, however, and may not reflect recent changes. In order to deal with this, the local navigation algorithm uses direct input from the environment to reach goals and sub-goals given by the global navigation system, and to avoid unexpected obstacles. This local navigation algorithm has no model of the environment and does not know the position of the actor in the world. The scene is rendered as before, and global distances to objects are extracted for use by the navigation system. An octree data structure for the 3D environment is constructed from the 2D image and the depth information. This data structure represents an actor's long-term visual memory of the 3D environment, and can handle static and dynamic objects. Using this long-term memory representation, an actor can find 3D paths through the environment avoiding impasses.

Kuffner and Latombe [KL99] present a perception-based navigation system for animated characters. Of particular interest is an algorithm for simulating the visual perception and memory of a character. The visual system provides a feedback loop to the overall navigation strategy. The approach taken is similar to that in Noser et al. An unlit model of the scene is rendered from the character's point of view using a unique colour assigned to each object or object part. These objects and their locations are added to the character's internal model of the environment. A record of perceived objects and their locations is kept as the character explores an unknown virtual environment, providing a type of spatial memory for each character. Unlike Noser et al., this method relies on the object geometry stored in the environment and a list of objects' IDs and positions, resulting in a relatively compact and fast representation of each character's internal world. Previously unobserved objects are added to the character's list of known objects, and other visible objects are updated

with their current transformation. Objects that were previously visible but are no longer in view retain their most recent observed transformations.

Musse and Evers [EM02] have presented a system for the storage of an agent's memory. The memory model keeps a record of past experiences by using dynamic and hierarchical finite state machines to store a list of behaviours that were previously executed by the virtual human. Each task that the agent has performed is then represented as a node in a finite state machine and may be queried at a later time in the simulation.

2.4 Visual Attention

The environment contains a wealth of diverse information, too much for it all to be cognitively processed to the same degree. Visual attention refers to the way in which some information receives enhanced cognitive processing. It is an emerging area of research for the animation of virtual characters; attention models provide output that is significant when animating fundamental behaviours such as gaze and looking.

Attention models have been used to provide automatic gaze behaviours for specific situations, such as a virtual helicopter pilot in a combat environment [Hil99] or conversational agents in social situations [CV99] [PPdR00]. These models aim to provide plausible animations of agents during discourse, based on research from psychology. More general models have also been proposed that attempt to locate areas of interest in the agents environment, for example, while the agent is walking in a natural scene [CK99] [Gil01]. We will now look at some of these systems in greater detail.

2.4.1 Cognitive Applications

Hill [Hil99] provides a model of perceptual attention in order to create virtual human pilots for military simulations that are cognitively plausible. Of particular interest in this work is the approach towards modelling the pilot's perception of objects in the scene. Objects are grouped according to various criteria, such as type and proximity. The granularity of object perception is then based on the attention level and goals of the pilot. If the virtual human pays more attention to an object, then more of its properties become available for querying. The aim of this work is to provide

plausible behaviour for the helicopter; animation of head and eye movements of a humanoid model is not considered.

2.4.2 Conversational Gaze Behaviour

The automation of visual attention behaviours for virtual characters has been studied more recently. A large amount of this research has been concentrated on eye gaze for conversational and communicative behaviour.

Of particular interest is work by Cassell and Thórisson [CT99], who conducted a number of studies on user interaction with autonomous human-like agents. Importantly, they demonstrated that non-verbal communication behaviours, such as gaze, are much more important for conversational plausibility than emotional feedback, such as smiling.

Cassell and Vilhjálmsón [CV99] present the BodyChat system, which automates the animation of communicative behaviours based on context analysis and discourse theory. It allows users to communicate via a text interface while the avatars automatically animate with appropriate gaze, salutations, turn taking and facial expressions. Cassell et al. [CVB01] have also presented BEAT, a toolkit for the animation of expressive behaviours. The toolkit automatically selects appropriate gestures, facial expressions, pauses for an input text and provides synchronisation information that is necessary to animate the behaviours in conjunction with the character's speech. The toolkit's gaze generator is based on an algorithm from Cassell and Torres [CTP99] that accounts for the relationship of gaze behaviour to turn-taking and information structure.

Poggi et al. [PPdR00] generate coordinated linguistic and gaze communication for the simulation of a dialog between two agents. Gaze functions are analysed according to their functional meaning. For example, if an agent were to say "that umbrella is brown", it may accompany the utterance with a glance towards the umbrella object. Communicative acts are defined by meanings and signals; the signal is the nonverbal expression of that meaning.

2.4.3 General Gaze Behaviour

Aside from automating gaze behaviours for conversational behaviour, more recent research has been focused towards modelling more general attention behaviours for human character animation [CK99] [Gil01]. For example, while standing in a natural

environment, an agent may look at objects or locations of interest. These locations may be based on current goals or may obtain the agent's attention in other ways. Gaze behaviours generated in such a manner are integral to the attention process and may feed back into it rather than merely byproducts, as is the case with gaze generation for conversational agents.

Chopra-Khullar [CK99] presents a computational framework for generating real-time visual attention behaviour in a simulated human agent based on observations from psychology, human factors and computer vision. The AVA, or Automated Visual Attending, system is composed of a three-level attention hierarchy, allowing eye movement patterns to be combined and interact with each other and with attention capturing events. At the top of hierarchy are intentional tasks. Tasks are supplied to the system in the form of text, for example *pick up the newspaper*. The locations of objects to look at in these deliberate tasks are queued into an Intention List. This top level of the hierarchy has precedence over the lower levels. The second level of the hierarchy handles exogenous behaviours, or behaviours invoked by stimuli external to the character. Peripheral motion and flashing lights are the main stimuli considered; peripheral motion locations are entered into a peripheral motion list or Plist.

The final level, with lowest precedence, consists of idle viewing. Idle viewing mode is activated when there are no tasks at hand and attention is being captured in a bottom-up manner. In this mode, a simplified image analysis technique is used to select directions of interest. An attention arbitrator is used to select between competing behaviours in the hierarchy and direct the agent's gaze in an appropriate manner. The arbitrator works by considering the attention entries in the Intention and Periphery lists. If there are entries only in the Intention List, then they are looked at in FIFO order. If there are entries in both the Intention List and Plist, irrelevant stimuli in the Plist are looked at probabilistically. If there are no entries in either the Intention List or the PList, then idle looking commences. Within each list, competition between entries is equal; that is, behaviours of the same type have an equal probability of being attended to.

Gillies [Gil01] presents a simulation of what a character is attending to in the environment in order to generate, among other behaviours, navigation and eye movements. Attending behaviours are generated in an object-centric manner; ray-casting is used for occlusion testing and retinal images are not considered. Objects contain

basic properties that are defined as *comms* and *agents*. They also contain features that represent more abstract concepts such as beauty and interest. Each object has a variable number of features and new features can be added by the user. Every feature has a corresponding probability that determines its chance of being looked at. The system does not attempt to attribute any meaning to object properties; actors are programmed to show more interest in some properties than in others. Attention is modelled through the interactions of numerous attention agents that are controlled and arbitrated by a central attention manager. In contrast to a request queuing methodology, agents send attention requests to the attention manager. Four types of attending behaviour are implemented: immediate, monitor, search and undirected. Request types are prioritised in that order. Undirected looking has the lowest priority and is activated only if the character has nothing else to attend to. This work builds on that by Chopra-Khullar [CK99] by allowing current attention behaviours to be interrupted. For example, if the actor's attention is caught by a peripheral task, the actor will wait until it can attend to the task before performing it. In this way, gaze-task synchrony is maintained. Memory and image-based attention are not considered.

2.5 Summary

Sensing, memory and attention paradigms have been applied to the plausible animation of autonomous characters. Animal characters, such as birds and fish, have been provided with internal perceptual models for implementing flocking behaviours. Human characters have used sensing and memory models to navigate virtual environments. There has been a relatively large amount of research on implementing gaze behaviours for conversational agents. Such research has focused on where agents look in relation to turn-taking and information content and in many cases is based on some form of textual conversation input. Less research has been focused on more general attention-based gaze behaviours for natural environments. In particular, little work has acknowledged the ability of external stimuli to grab ones attention in a natural environment.

Our synthetic vision framework is inspired by the approaches taken by Noser et al. and Kuffner and Latombe. However, unlike these approaches, which only provide visual sensing, we have extended the vision framework to incorporate a

crude form of visual perception; the acuity of the human eye is approximated and objects are segmented in a different manner depending on where they are located in the visual field. Unlike other approaches, we have also implemented a multi-stage model of memory, inspired by research in psychology, that further refines the notion of visual information filtering initiated by the visual system so as to limit the amount of environmental information that must be processed by the virtual human. Our attention system differs from previous systems in that it accounts for attention that is grabbed by environmental stimuli. Attention also fulfills a dual-role in our system: not only does it link into the gaze controller in order to provide eye and head movements, but it also forms an integral part of the multi-stage memory system, deciding what visual scene information will be passed through each stage of the virtual human's memory.

Chapter 3

Psychological Basis

Our approach in this thesis is to apply simplified models inspired by research from fields such as psychology and cognitive engineering to the animation of virtual humans. As such, the models proposed here are inspired by and based on studies of vision, visual attention, memory and gaze. In this section, we present relevant background work in these areas in order to prepare the reader for the models proposed in the following chapters of this thesis.

3.1 Visual Sensing

The visual modality of sensing is of great importance to primates; almost half of the cerebral cortex in the macaque monkey is involved in visual processing of some sort [FE91]. Two main visual systems are widespread in nature. Common among arthropods, such as insects, is the multifaceted compound eye. These eyes produce a mosaic sensory image that is composed of a multitude of individual segments; each segment, or facet, spans a small field-of-view. As the number of facets in the eye increases, so too does the resolution, providing sharper images. Ants, for example, have only 50 segments, while the dragonfly possesses 30,000 [Hec98]. Because of the segmentation inherent in these visual systems, there is no real image formed on the retina; instead, synthesis is performed electrically in the nervous system. In contrast to multifaceted systems, the visual systems of humans, as well as many other vertebrates, compose a single continuous image on a light-sensitive screen or retina.

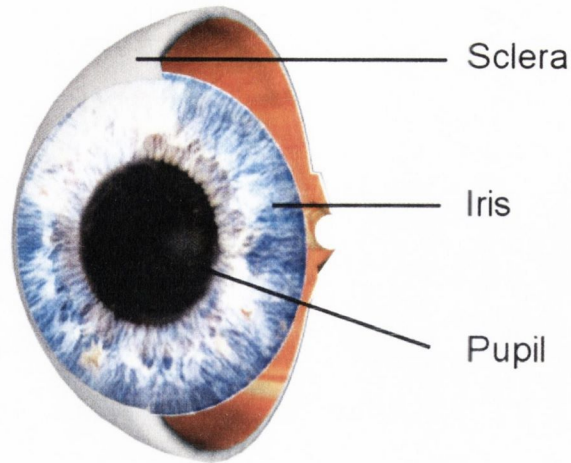


Figure 3.1: A close-up view of the human eye, noting the Iris, Pupil and Sclera.

3.1.1 The Human Eye

The eye is a spherical, jelly-like mass that is surrounded by a tough shell called the *sclera*. The *cornea* is the transparent front portion of the eye, while the rest of the sclera is white and opaque (see Figure 3.1). Light enters the eye through the *pupil*, where it is focused and inverted by the cornea and *lens*. The lens is an elastic structure that is stretched out into a disc-like shape by an array of radial fibers called *zonule fibers*. Focusing is achieved by contraction and expansion of the *ciliary muscle*, which allows the zonule fibers to slacken or tense. The amount of light entering the eye is regulated by the *iris*, which operates in a similar way to a shutter, through contractions of the *pupillary sphincter* and *pupillary dilator* muscles.

Incident light is then projected onto the back of the eye to a layered region called the *retina* (see Figure 3.2). The retina is a sheet of brain tissue covering roughly 65 percent of the eyes surface and consists of two types of photoreceptive cells, referred to as *rods* and *cones*, as well as several layers of processing cells. Rods are more numerous than cones, numbering in the region of 120 million. They are responsible for our night vision, peripheral vision and motion detection. Although rods are more sensitive than cones, they do not provide good colour sensitivity. In contrast, cones provide good colour sensitivity, but are fewer, numbering only 6 to 7 million. Both of these photoreceptor cell types pick up light energy, which is then transduced by the processing network into neural signals. The final layer in this

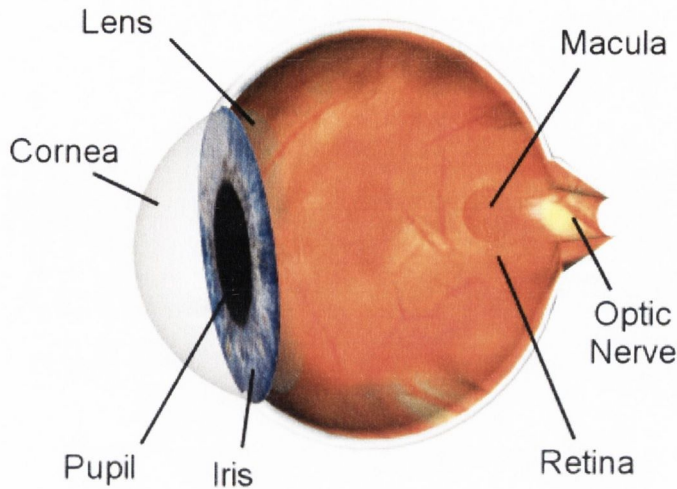


Figure 3.2: Cross section view of the human eye.

network consists of *ganglion cells*. The axons, or outputs, of these cells form the *optic nerve*, where neural signals are transmitted for higher-level processing in the brain. Although there are roughly 130 millions photoreceptor cells, there are only around 1 million retinal ganglion cells. The optic nerve leaves the eyeball at the optic disk and extends to the *optic chiasm*. Photoreceptors and processing cells need to be pushed aside at the optic disk in order to form the optic nerve, creating a blind spot in our vision.

Of particular interest is the fact that resolution across the retina is non-uniform; that is, the distribution and density of rods and cones vary considerably across the surface of the retina. Each eye covers a 130° field-of-view, with a stereo overlap of 60° . Although the eye therefore obtains light from a 200 degree field-of-view, the acuity over most of this field is poor. Many of the colour sensitive cones are concentrated in an area of the retina called the *macula*. In the centre of the macula is a small (0.3mm diameter), circular region called the *fovea centralis*. This area is rod-free, consisting exclusively of densely packed cones. This means that the fovea provides us with our sharpest and best colour vision, but only over a small field of around 15° . The remaining part of the retina affords us with peripheral vision and is roughly only 15% - 50% of the acuity of the fovea [Jac95]. Objects located in this region of vision are generally not clear enough to be seen properly.

The inherent structure of the eye is responsible for a range of higher-level behaviours; because the fovea provides the most detailed information about a scene and is rather limited in its field-of-view, the eyeball must continually move in order to ensure that light from a position or object of interest falls within it. The need to foveate regions and objects of interest would appear to be the basis for eye movements, head movements and many other higher-level movements, such as orienting, that predominate everyday human behaviour.

3.2 Visual Attention

We do not have the cognitive resources to be able to process everything in our visual field to the same degree - visual attention is a selective mechanism that enhances our processing of some sensory items. Since our field of view is limited and our most acute visual area, the fovea, is merely a sub-region of this field, some mechanism is necessary to decide what part of the environment is placed within the region of the fovea. Although visual attention can be directed to a particular region of space, feature or object to enhance processing through an eye movement, in what is referred to as *overt* attention, this does not necessarily mean that an eye movement will always follow; attention can also be allocated in a *covert* manner, so that stimuli can be attended to without eye movements. The visual attention mechanism has been defined as consisting of the following basic components [TCW⁺95]:

1. The selection of a region of interest in the visual field
2. The selection of feature dimensions and values of interest
3. The control of information flow through the network of neurons that constitutes the visual system
4. The shifting from one selected region to the next in time

There are many theories about how these functions are actually achieved, but a great deal of research literature theorises that attention consists of a hierarchical, two-stage selection mechanism. The first stage is referred to as the pre-attentive stage and refers to early processes that operate across the whole visual field. This stage precedes an attentive stage that is of limited capacity and can only process a limited number of items. When items move from the pre-attentive stage to the

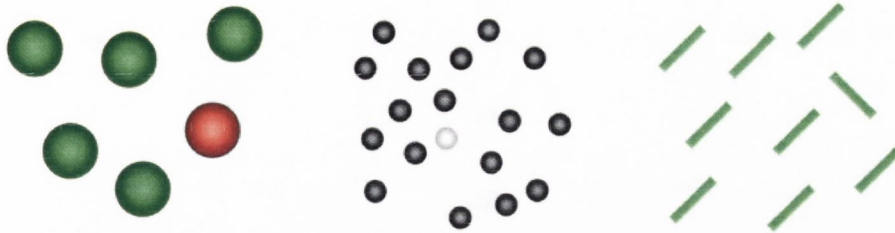


Figure 3.3: Example feature dimensions (from left to right) colour, intensity, and orientation.

attentive stage, they are considered to be selected and regarded as having entered into the consciousness of the observer and been made available for higher level cognitive processing.

Processing in the pre-attentive stage is considered to be unlimited in capacity. Furthermore, it operates in parallel across the whole visual field; that is, it operates simultaneously on a number of varying locations in the scene. Of particular interest here is a contrast effect where stimuli defined by a single feature dimension appear to *pop-out* of the display at the viewer, regardless of position. For example, a yellow dot among blue dots may tend to attract a viewers attention (see Figure 3.3 for some examples of pop-out in a single feature dimension). Parallel levels of processing seem to provide strong responses to a few parts of the scene and poor responses to everything else [Itt00]. Context is a key factor in determining the responses at this level of processing: stimuli are contrasted with their neighbours rather than being considered in isolation. Finally, pre-attentive processing appears to act, for the most part, independently of strategic control, in a *bottom-up* manner. The deployment of attention resources at this stage is therefore referred to as being exogenous or stimulus-driven. In this way, the scene elicits attention to locations or objects of possible interest for further attentive processing.

The development of computational models of attention began with *Feature Integration Theory* [TG80], which viewed the perception of objects within the framework of the two-stage process above. During the pre-attentive stage, primitive scene features such as colour, orientation, brightness and movement are processed. It is thought that these features are only resolved, or *glued*, into coherent concepts after integration by an attentive process; attention therefore not only selects an area of interest, but also elaborates on the internal representation of features within that

area, thus binding them into higher level constructs.

3.2.1 Bottom-Up Visual Attention

According to the spotlight metaphor [EY87], bottom-up attention acts in spatial terms as a spotlight. Anything within this spotlight, such as parts of an object, an object, or a group of objects, will receive the focus of attention processing. The size of the spotlight is thought to be adjustable, in a similar way to a zoom lens. A narrower spotlight will process its contents better than a broader spotlight, but will take longer to assess a full scene. Control of this spotlight can take place in a bottom-up manner; that is, low-level features, such as colour, orientation and intensity, are extracted from the scene and processed in parallel to produce a number of conspicuous locations over the entire visual scene. These locations then compete for attention and, when a winner emerges, attention focuses on it to analyse it in more detail [KU85].

3.2.2 Centre-surround Processing

The processing that occurs on low level features described above takes place in a centre-surround fashion. Centre-surround processing simulates the processing that takes place in visual receptive fields found in biological systems. For example, recalling the layering of cells in the retina, photoreceptor cells pick up light energy, which is then transduced by a processing network into neural signals. Although photoreceptors respond directly to diffuse light, ganglion cells do not. Rather, ganglion cells respond to small spots of light, small rings of light or light-dark edges. Some ganglion cells are on *centre-off* surround cells, also known as *on-off* cells. If light falls on a small region of the retina, the firing rate of these cells will increase. Light that is farther from the centre elicits no response. In contrast, there are also off *centre-on* surround, or *off-on*, ganglion cells, where light at the centre of the cell suppresses the firing rate and light in the surround increases the rate of firing (see Figure 3.4). Outputs from the on regions and off regions of the ganglion cells are summed to provide the response of the ganglion cell, so the central region must be lit and the outside dark, or vice-versa in order for the cell to provide a strong response. Uniform light across the cell will result in a weak response. In centre-surround processing, contrast is the key concept rather than absolute amplitude; colours and brightness are not judged against some fixed scale, but instead by comparison with

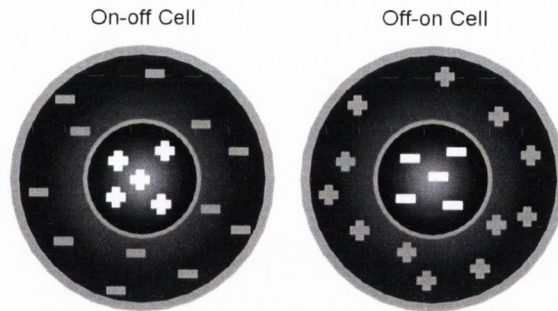


Figure 3.4: On-off and off-on receptor fields of ganglion cells in the retina and LGN.

other parts of the visual field.

The ganglion cells that predominate in the fovea are called *P cells*. They are colour sensitive cells whose small receptive fields allow them to pick up a high level of detail. *M cells*, on the other hand, are coarser in nature and have strong responses to both luminance and stimulus motion. M cells are found in the peripheral area of the retina.

Ganglion cells are present not only in the retina, but also farther along the visual pathways (see Figure 3.6). From the eye, the axons forming the optic nerve cross over at the optic chiasm to become the optic tract. The optic tract feeds into the lateral geniculate nucleus (LGN) which actually consists of two nuclei. Ganglion cells in the LGN, a part of the brain known as the *thalamus*, have their axons connected to the primary visual processing centre in the brain: the visual cortex. Because the retinal receptive fields are circular, the earliest levels of visual processing that take place in the retina provide tuning to very basic features such as spots of light. Processing that takes place further in the cortical areas is of more complex nature. For example, in contrast to the on-off cells, receptive fields may form edge detectors or bar detectors; these respond maximally to an edge of light or a bar of light that is of a certain width and orientated accordingly (see Figure 3.5 for edge and line receptor fields). These receptive fields may consist of inputs from a number of on-off or off-on cells - however, a single on-off cell will not cause a detector to respond. Thus, the detectors respond only to patterns of on-off cells.

Continuing the idea of grouping low-level features into higher level constructs, the *Gestalt Laws of perception* concern perceptual organisation in humans, such as the abilities that allow people to group similar objects into entities or groups.

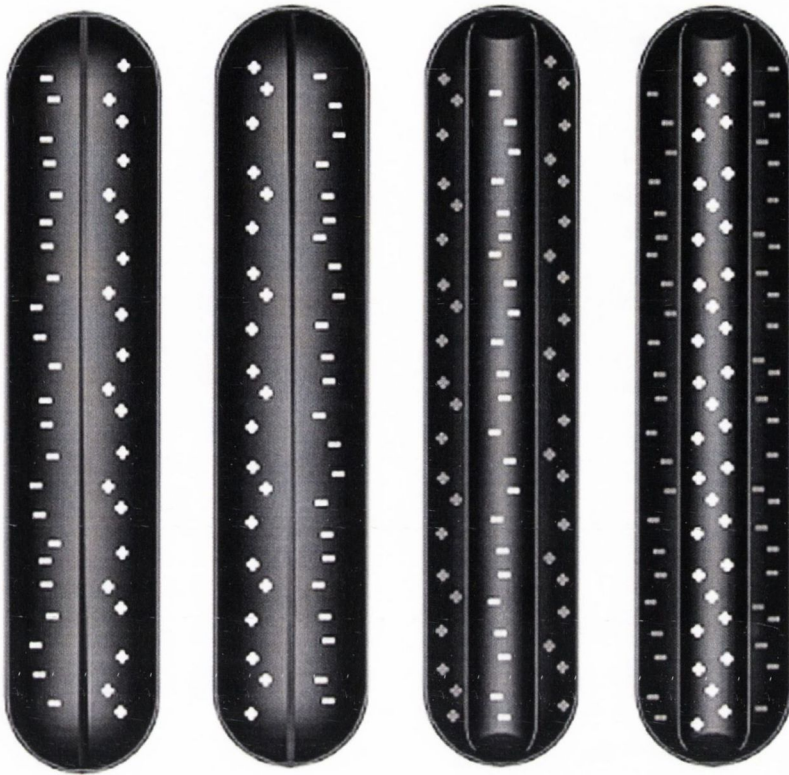


Figure 3.5: Edge and line receptor fields from patterns of ganglion cells. Fields illustrated on left side respond to edges, while the fields on the right side respond maximally to lines.

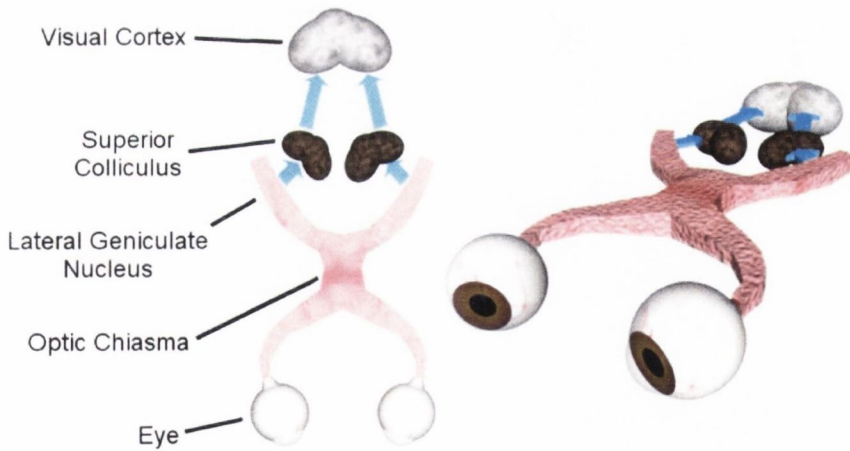


Figure 3.6: Illustration of the neural pathways involved in visual processing.

The five main factors considered in Gestalt perception are proximity, similarity, continuation, symmetry and closure. For example, scene elements that are in close proximity to each other and visually similar are likely to be grouped into a single mental construct.

To summarise, in biological vision systems, neurons involved in the earliest visual processing stages respond to simple visual attributes such as intensity contrast, colour opponency, orientation, direction and velocity of motion. As visual processing continues, neurons are tuned only to higher-level constructs and patterns such as corners and even certain views of real-world objects.

3.3 Memory

Memory is perhaps the most extensively studied area in cognitive psychology. Despite decades of intensive research, however, human memory still provides many challenges to cognitive psychologists and is an area of great debate. The 19th Century psychologist William James theorised two different forms of memory; a primary memory consisting of information currently held in the consciousness and a secondary memory allowing for the longer-term storage and retrieval of information not present in the consciousness. Over the past 40 years, a number of different structural analyses of memory have been performed on normal individuals as well as individuals suffering from brain damage and disease. One theory of memory that has been particularly influential is that referred to as stage theory, or *the modal model* [AS68]. Stage-theory, discussed in more detail in Section 3.3.1, has had an enormous influence on psychology and although more elaborate models have been proposed since, it has played a crucial role in a number of more detailed modern theories such as SAM [GS84]. Although there are many varied theories of memory, the stage theory of memory provides a very useful schema for animating virtual humans; it provides clearly defined filtering and storage mechanisms while allowing for fast implementation.

3.3.1 Stage Theory

According to stage theory of memory, memory is composed of a number of separate memory stores. Various cognitive processes act on a number of items in each stage of memory, causing them to be moved onto the next stage. For the moment,

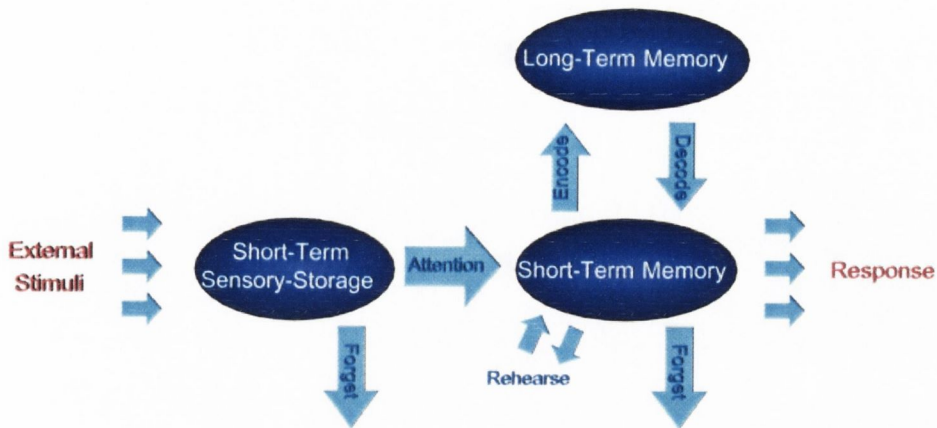


Figure 3.7: Schematic of the stage theory of memory.

we will concentrate on the structure proposed by the model; the key process that moves items between memory stages is attention, and is discussed in-depth the next chapter.

An overview of the theory is as follows (see Figure 3.7). First of all, environmental input arrives at a sensory register, or *short-term sensory storage*. A filtering process is applied to a number of items in the short-term sensory storage system causing them to be transferred into another storage area, called *short-term memory*. If a further process is applied to items in the short-term memory then they are moved into a final storage area, referred to as *long-term memory*. A key feature of the theory is that items must be present in one stage of memory before they can progress to the next stage; that is, items cannot move straight from short-term sensory storage into long-term memory.

The short-term sensory storage, often referred to as the *STSS* or sensory register, is a transient, limited capacity memory store where input arriving from the environment is converted into signals that the brain can understand. The sensory register holds information from a variety of sources of information, such as visual, auditory and tactile. Since we are interested only in the visual modality, we are particularly interested in visual sensory memory in the sensory register. Experiments by Sperling and Neisser have shown the existence of a brief visual sensory store or iconic memory that can effectively hold all of the information in the visual display [Spe60][Nei67]. While information is in the store, it can be attended to and reported; if the information is not attended to, it is quickly lost.

The second stage is referred to as short-term memory. A subset of items from the sensory register is transferred into short-term memory as dictated by the process of attention. Short-term memory (STM) relates to our thoughts at any given moment in time; that is, the contents of the STM are either items in the current focus of attention, or information from long-term memory that is currently active. Short-term memory is limited both in duration (in the order of seconds) and by the number of units of information that can be processed at any one time. Researchers suggest that the STM can process between 7 ± 2 and 5 ± 2 units or chunks of information [Mil56]. These chunks correspond to letters, numbers, and also larger units such as words and phrases. Items are removed, or forgotten, due to both trace decay and displacement due to capacity limitations. Trace decay refers to the natural dissipation of a memory trace that is not rehearsed.

The final stage of memory is long-term memory, or *LTM*. Long-term memory is perhaps the most fascinating area of memory. Studies of memories recovered through hypnosis would appear to indicate that the duration of items in long-term memory is permanent. Although this evidence is not conclusive, it can be said that some memories last a lifetime. Furthermore, the capacity of long-term memory is generally considered to be infinite. There are a number of theories regarding forgetting in long-term memory. Cue dependant forgetting assumes that memories are never forgotten; rather, our ability to retrieve them is hampered due to a lack of appropriate cues or handles. Interference theory suggests that items are forgotten when existing memories are disrupted by newer information, or when the newer information is disrupted by existing memories. Levels of processing theory suggests that, in a similar way to forgetting in the STM, trace decay may lead to forgetting.

Given the structure advanced by stage-theory, we now discuss the processes that are responsible for moving information from one stage to another. Information from the STSS can only be transferred into the STM by attending to it. As mentioned, if items in the STSS are not attended to, then they quickly decay. Items that have been attended to enter the STM and are regarded as entering the consciousness. Once items are in the STM, they can be transferred to long-term memory through the process of maintenance rehearsal. Essentially, items in the STM that are rehearsed will enter the LTM.

The original modal memory model proposed by Atkinson and Shiffrin proposes a serial arrangement of memory buffers. That is, items must first be in the STSS in

order to proceed into the STM and items must first be in the STM before they can proceed into the LTM. Contemporary views of memory acknowledge more general situations where information can flow into the STM without first residing in sensory memory and flow into the LTM without first residing in the STM (see [BB96] for an overview). As pointed out by Pashler [Pas98], the original version of the model as proposed by Atkinson and Shiffrin requires modification, primarily by abandoning the idea that information flow is strictly serial and that short-term memory is unitary and exclusively verbal. Pashler notes the attraction of this model when such modifications have been incorporated:

“...the model readily deflects most of the criticisms, and its core ideas enjoy broad and diverse empirical support ... several decades of research suggest that it has important insights at its core and may represent one of the more significant accomplishments of contemporary experimental psychology.”

As such, the modal model of memory represents an important, although simplified, attempt at investigating human memory.

3.4 Gaze and Blinking

Gaze is a salient behaviour. It has been shown to play an important role in social hierarchies [AC76]. For instance, studies of gorillas have shown that visual attention is often directed towards the more dominant members of the group, the *alpha males*, to form an attentional structure. Attention is directed upwards to more dominant animals resulting in social cohesion and a dominance hierarchy; in this way infants pay attention to their mothers, mothers to their mates and males to more dominant males [Cha67]. Such an attention structure may also apply to humans, as noted by Argyle and Cook [AC76].

Gaze also plays an important role in signalling interpersonal attitudes and emotions. It can be used as a threat signal. In experiments conducted by Exline and Yellin [EY69], rhesus monkeys that were stared at attacked in 47% of trials. Eye-spots are used by some species as a form of defence against predators; for example, butterflies will often expose their *eye-spots* when predators approach in order to deter or mislead their attack.

There is evidence that human infants pay attention to the direction of gaze of others from as young as 3 months [FJBS00] and also, that infants are faster in making saccadic eye movements to peripheral targets cued by the eye gaze of a central face [SB75]. Finally, there is an adult propensity to follow the gaze of others which seems to be related to an ability to attribute intentions to the gazer.

3.4.1 Eye Movements

Eye movements carry the fovea and visual attention to parts of the scene to be fixated upon and processed at high resolution. A number of different types of eye movements have been identified by researchers. Bruce and Green classify seven different types of eye movements [BG90].

Convergence refers to the motion of the eyes relative to each other in order to ensure that objects of varying distances from the viewer may be brought into focus. *Rolling* is the involuntary rotation of the eyes around the axis passing through the fovea and pupil. *Pursuit* is a slow eye motion that is used to keep a moving object foveated. Pursuit eye movements cannot be conducted in a voluntary manner. *Nystagmus* are smooth pursuit motions in one direction followed by fast motions in the opposite direction that occur as a response to turning of the head. In contrast, *Physiological nystagmus* are involuntary oscillations of the eye that allow an image to be perceived by continually ensuring that fresh retinal receptors process the view. *Microsaccades* are believed to very small ballistic eye movements that have a drift correcting function.

Here, we are particularly interested in very fast, jumping eye movements that are made millions of times a day. These movements, called *saccades*, enable us to direct our eyes towards different areas of our visual field for further study, usually in periods, referred to as *fixations*, of up to 600 ms. It takes approximately 100-300 ms to initiate a saccade and then up to another 120 ms to complete it. Although saccadic eye movements are generated voluntarily, they are ballistic; that is, once the eye movement is planned and initiated, it cannot be altered mid-course. While saccades are in progress, visual processing of the visual field is inhibited.

Experiments reveal that when viewing a picture or scene, the eye tends to fixate and saccade between a certain number of locations repeatedly [Yar67] [NS71]. A sequence of these locations create what is referred to as a *scan path*. Scan paths are regarded as being idiosyncratic; that is, although people may share the same

locations of interest in a scene, they will move around them in different sequences. Furthermore, the task at hand can affect the resultant eye movements, as pointed out by Yarbus [Yar67]. In this thesis, we presume that no particular task is active and we are therefore interested only in eye movements under general viewing conditions.

Of particular interest in psychology is the question of how such scan paths are constructed. One theory, *scan path theory*, suggests that a top-down, internal cognitive model of the scene drives our eye movements and fixations [NS71]. The contrasting theory is that features in the external world control eye fixations in a bottom-up manner. In essence, the scene or picture itself dictates where the eye is likely to visit. In practice, it is generally acknowledged that both bottom-up features and internal memory influence scanpaths. One possibility is that a top-down scanpath is recalled from memory to check a new scene against a memorised scene. This scanpath may then be locally adjusted according to salient image features.

3.4.2 Gaze Shifts

Looking around is an important human behaviour. As noted by Gibson [Gib79], many animals, including horses and rabbits, have eyes in the lateral areas of their heads. This means that they can see a reasonable amount of their environment without necessarily having to look around (see Figure 3.8). In contrast, human eyes are set into the front of the skull; humans must therefore move their heads in order to look around. This would seem to suggest that animals who are preyed upon have a more panoramic field-of-view, while predators, such as cats, can afford to have eyes in the front of the head [Wal42].

Gaze shifts consist of combined eye and head movements for directing attention from one point of interest in the environment to another. When looking around, humans usually use the coordinated motion of the eyes and head to direct attention from one point of interest to another. Such movements are essential for shifting gaze to targets that are located beyond the oculomotor range (50° - 55° eccentricity in human subjects) and are also common when looking at targets of smaller eccentricities.

3.4.3 Blinking

On average, a person blinks around 17,000 times per day, with a normal blink rate of 20 blinks per minute [Kar88]. Each blink lasts from 20-400 milliseconds. The key

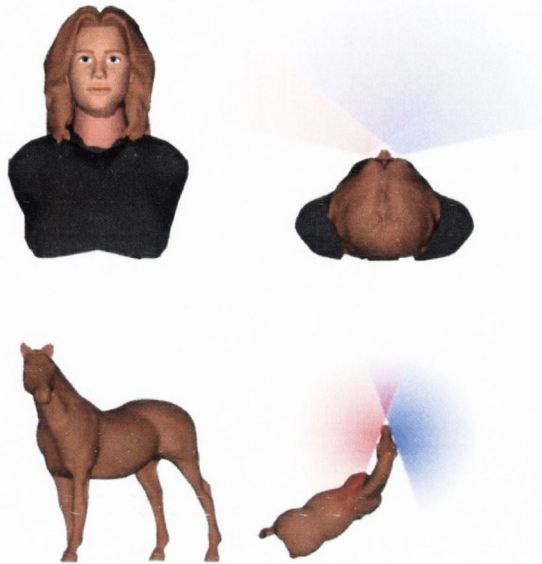


Figure 3.8: Illustration of the field-of-view of a human (top) and a horse (bottom).

purpose of blinking is to clear debris from the eye and moisten surface tissue.

A number of factors affect blink rate and can be categorised as environmental factors and internal factors. Environmental factors include alterations in temperature, lighting and airflow that have an effect on blinking, as do factors affecting the ocular surface, such as wind. Different activities may also cause changes in blink-rate, for example, engaging in conversation or concentrating on a visual task. Blink rate is also linked to internal psychological and physiological factors. Blinking increases with excitement, frustration and anxiety and decreases with guilt and often when mental load is at its lowest. Age, gender and muscular tension also appear to effect blink rate.

In this thesis, we are particularly interested in blinking motions resultant from gaze changes. *Gaze-evoked blinking* refers to blinking motions that accompany saccadic eye and head movements, as oppose to normal blinks that are elicited by external stimulation, such as a puff of wind. As noted by Evinger et al. [EMP⁺94], such blinks would appear to serve the purpose of protecting the eye during the movement, while also providing lubrication to the cornea while vision is impaired due to the saccade. These blinks often occur before initiation of the head or eye movement and occur whether the eyes are open or closed.

3.5 Summary

The biological systems involved in processes such as visual sensing, memory, attention and gaze are complex, consisting of the interacts of multitudes of small components. Although there is still debate in psychology regarding theories in each of these areas, the high-level functioning of such systems is relatively well-known. Using this research as inspiration for the construction of cruder, but computationally feasible, models applicable to virtual human animation is the primary methodology in this thesis. We use the research presented in this chapter to implement an attention system based on the integration of visual sensing, memory, attention and gaze.

Chapter 4

Synthetic Vision and Memory

4.1 Introduction

A famous poem by Edgar Allan Poe raises the query “Is all that we see or seem, but a dream within a dream?”. Since our environment is dynamic and our senses are not all encompassing, information from our environment is never totally accurate and is subject to change at any moment. As humans, we must work from an internal representation of the world that is a loose approximation of some actual reality; what we sense is only a small portion of our environment, what we perceive is open to interpretation from our senses, and what we store as our perception may vary greatly from the changing world in which we live. If we agree that human behaviour is based on an internal representation of the world as opposed to the actual world, then it becomes evident that if we wish to synthesise human behaviour for virtual humans, then we should provide them with internal models to act on. Here we consider two constructs that allow virtual humans to acquire and operate on internal representations of their environment: synthetic vision for collecting external data and synthetic memory for storing this data.

Our approach to modelling these constructs is to design simplified models inspired by studies from psychology to provide the *essence* of the actual systems while reducing the computational overhead to a minimum. A large amount of research has been conducted in the field of psychology over the course of many years and although there is often great debate about many of the finer details, remarkable progress has been made in our general understanding of the systems and processes involved. Applying models from these fields to computer animation would seem to

make good sense: after all, when a crude synthesis of human behaviour is at the heart of our goals, where better to look than to scientific disciplines that have been studying the foundations of such behaviour for centuries. Furthermore, we do not take the approach of implementing such models in a complete manner; many models are overly complicated and computationally prohibitive for our purposes. To take an example, many contemporary models are connectionist in nature, simulating calculations at the neuronal level. Such calculations are not suitable for the simulation of virtual humans on desktop personal computers and are probably not necessary for conveying the rich sense of presence that contemporary virtual humans are lacking.

A key idea towards solving such problems is that of *successive stages of filtering*. Filtering techniques can be employed to reduce the number of objects that the agent must consider, for example Bordeux et al. [BBT99]. The type of filtering technique used helps define the plausibility of the sensing system: for example, in the human being, the primary information flow direction is to the front. A filtering technique that prioritises information flow in this way would better approximate the system. It is also important to consider the balance between the realism with which the sensing takes place and the costs associated with the process. Our approach is to provide a balance between sensory honesty [Blu97] and speed. In this way we try to ensure that the information obtained by the agent is reasonably plausible while compensating for certain cognitive abilities by allowing direct queries on the scene database.

4.2 Synthetic Sensing

In order for virtual humans to interact with their environment, they must first have a means of sensing it. There are a number of options for providing agents with information about their environment. The most straightforward approach is to allow them full access to the scene database. In this way, they can read any type of object information, such as world-space object locations, directly from the database at will. The main limitations of this approach are plausibility and scalability.

In terms of plausibility, providing full access to the database is equivalent to an all-knowledgeable being. If an agent's behaviour is partly dependent on their senses and they are allowed full access to the scene database, then the resulting behaviour will appear implausible. After all, real humans do not have access to any

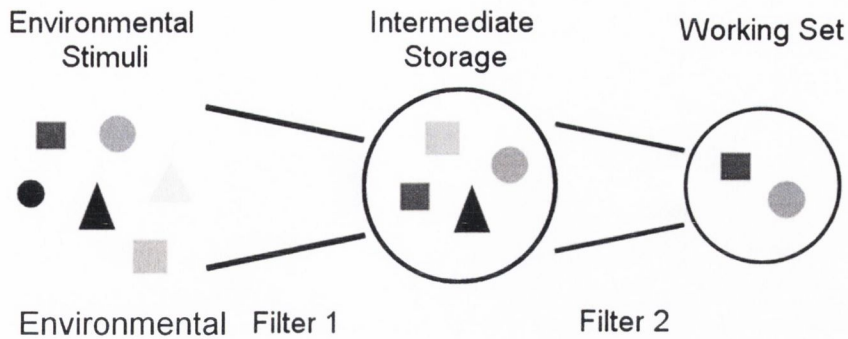


Figure 4.1: Simple diagram illustrating successive stages of filtering. Environmental stimuli are systematically filtered so that only relevant information persists for later processing.

sort of global database. As noted in [And95], stock phrases such as “I don’t have eyes in the back of my head” reflect such limitations. As humans, when we need to find something, we must search for it. These sort of low-level actions, sometimes overlooked in traditional animation, define human behaviour in the environment; the absence of such rudimentary actions are consequential in that they are perceived to be false in some way by human viewers. Plausible sensing is at the core of plausible behaviour.

With regards to scalability, virtual environments tend to contain vast amounts of information. Essentially this means that if virtual humans are to be autonomous and do their own processing, then they must consider a large amount of information in order to conduct environmental interactions. Some of this information will be of relevance to the virtual human, but a large proportion of it will likely be of little significance.

Unlike the *total access* approach mentioned above, synthetic senses provide a means for agents to perceive their environment through an indirect means. Whereas agents are normally unrestricted when performing interrogations of the environment’s state in the database, synthetic senses reduce the amount of information made available to the agent. In this way, they contribute to solving the problems of plausibility and scalability by providing filtering mechanisms that model actual senses. Indeed, real humans face the same problems when dealing with their environments. Given our limited cognitive resources, we cannot hope to process all of

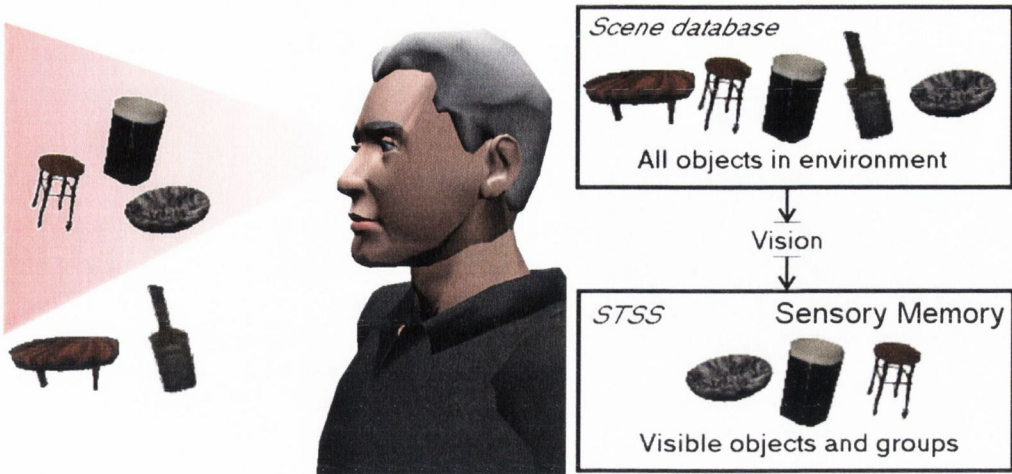


Figure 4.2: Synthetic vision acts as the first filter for information in the environment. Only items within the agents view-frustum that are not fully occluded will be retained for further processing.

this information at once: our senses provide a first step in filtering external information to manageable amounts for further processing. The concept of successive stages of filtering is at the core of our synthetic sensing and memory systems (see Figure 4.1).

4.2.1 Synthetic Vision

This thesis focuses on the visual modality of sensing. When modelling the human senses, there is good reason to start with vision; the visual modality plays a dominant role in our perceptual processing [PNK76].

It is apparent that the structure and processes of the human eye are quite complex. There are many problems in attempting to simulate such a system. First of all, the scene must be sensed in an appropriate manner. Secondly, the only information available about the environment is that queried from the vision sensor; the captured two-dimensional image must essentially be reconstructed into a meaningful three-dimensional representation. This is difficult and time-consuming, involving scene segmentation, recognition, and finally interpretation. This approach, referred to as artificial vision, is a difficult proposition and is an area of great research in many fields, particularly robotics. Fortunately, an elaborate approach such as artificial vision is not necessary. We take a faster, more appropriate approach, referred to as

synthetic vision, which bypasses many of the problems inherent in artificial vision. The purpose of the synthetic vision approach is to provide information on what objects within the agent's field-of-view are visible (see Figure 4.2). Objects in the environment may not be visible for a number of reasons: they may be outside of the agent's field-of-view; they may also be occluded by other objects in the environment; finally, the objects may be too far away for the agent to be able to see them.

Synthetic vision can be differentiated from artificial vision in a number of ways, as noted in [NRTT95]. In comparison to artificial vision, which is dependent only on the data from a real-world scene (that is, the captured two dimensional image), synthetic vision may also interrogate the three-dimensional world database. This is possible since we are dealing with simulated vision for a virtual agent as opposed to a real agent. Because the environment is virtual, all of the three-dimensional scene information is available to us through the world database. Synthetic vision thus allows us to bypass the difficult problems of segmentation, recognition and interpretation in a few quick steps. By bypassing these stages, we implicitly presume that our agents represent their environment internally at the same granularity as the world database, do not have problems recognising objects in the scene and do not make mistakes in interpreting what they sense in the scene. While real humans do make recognition mistakes from time to time, in the context of real-time virtual human animation, we view the cost of implementing such mistakes as too high to justify, given the minimal behavioural variation likely to be noticeable to an outside viewer.

Object false-colouring

Our synthetic vision module is monocular in nature and is based on the method of *object false-colouring* in order to extract meaningful information from the scene, as described by Noser et al. [NRTT95]. False-colouring proceeds as follows: every object in the environment is assigned a single, unique false-colour value. This colour value is subsequently used as an identifier for the object during processing. The rendering hardware is then used to render the scene from the perspective of the agent. However, rather than rendering the environment in the conventional manner, i.e. with textures and lighting enabled, only objects in the scene are rendered. The objects are rendered in flat-shaded mode according to the unique false colour that they have been assigned. Such renderings do not need to be particularly large or

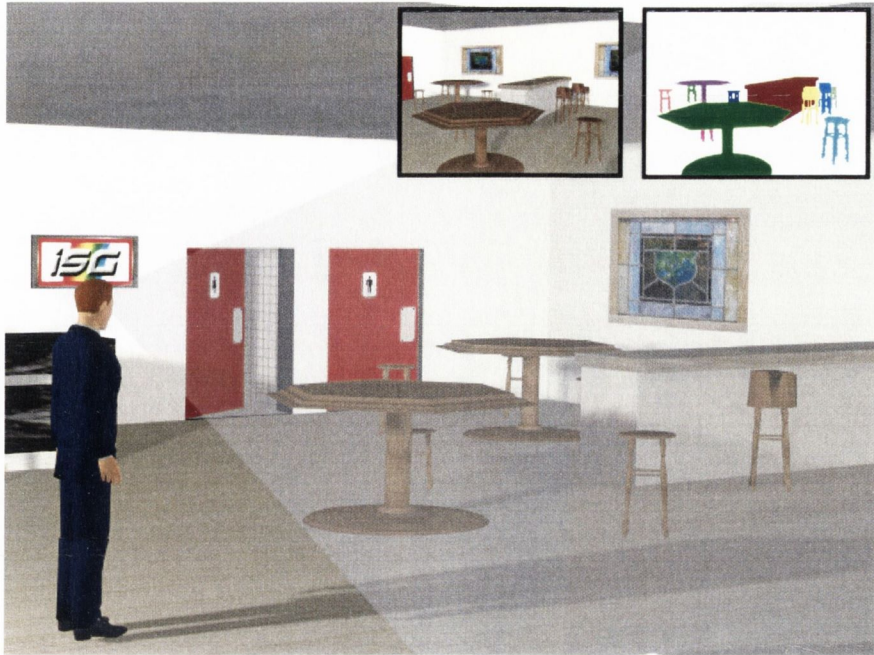


Figure 4.3: Depiction of object false-colouring. Insets show normal rendering from the perspective of the agent (*left*) and a corresponding false-coloured rendering (*right*).

detailed; the objects appear flat-shaded in their unique colours. No textures, lighting or other effects are applied to the rendering. Figure 4.3 depicts the method for an agent standing in a bar scene.

False-colours are later scanned from the viewpoint rendering in order to extract object visibility information from the scene. The false-colouring method therefore provides a method for making fast and easy visibility queries about objects in the agent's viewpoint. Moreover, the scene may also be rendered into the z-buffer in order to provide distance information for each object in the scene.

Our Approach

There are a number of requirements for our synthetic vision module. First of all, it must act as the first stage in the successive stages of filtering schema in order to reduce the environmental information reaching the agent. Secondly, it must, at a coarse level, emulate some of the basic operations that occur in early visual processing in the human system, but are too costly in terms of computation to

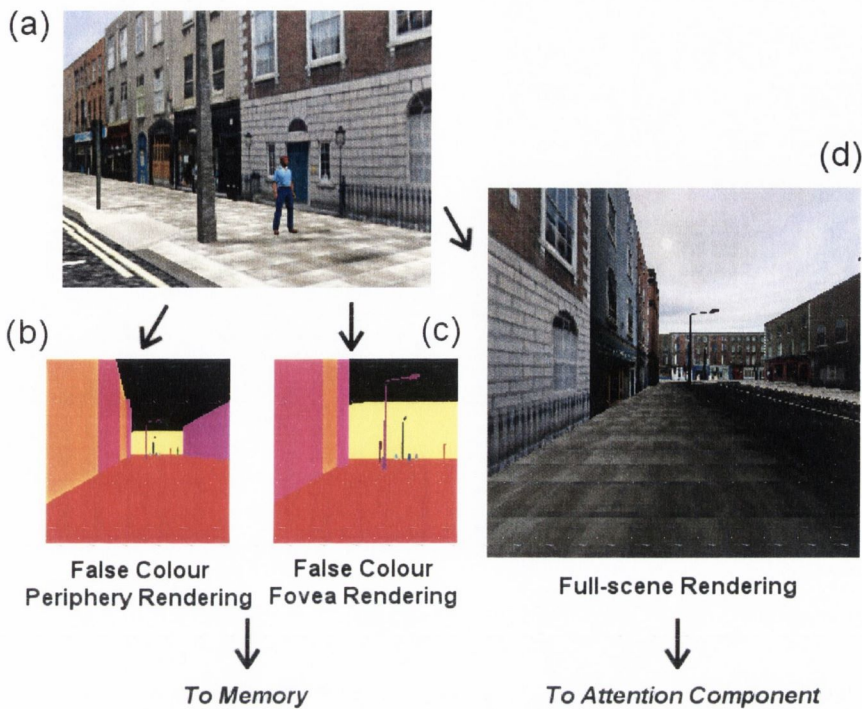


Figure 4.4: Illustration of the different snapshots taken from the viewpoint of the agent during a single update of the visual sensing module. (a) A view of the agent standing in the scene. (b) False-colour periphery and (c) fovea rendering from the point of view of the agent. Both of these renderings are sent to the memory module in order to provide storage of object visibility information. (d) A full-scene rendering is also taken from the point of view of the agent and sent to the attention module for processing.

implement. These include object determination and recognition from the basic features in the scene, for example, the Gestalt process of *glueing* edges into shapes. Finally, the vision system must also capture the scene in such a way that it can be processed later by the attention model, which operates in a scene-based manner, as opposed to an object-based manner (see Chapter 5).

Our system differs from the implementation by Noser et al. by taking three snapshots of the scene, two of which are false-coloured and one which is fully rendered. The camera position is first moved to the agent’s eye position. Because the vision system is monocular, the eye position is approximated as the midpoint between the individual eye positions in the skeletal definition of the character (see Figure 4.5). The three renderings of the environment are then taken and represent a

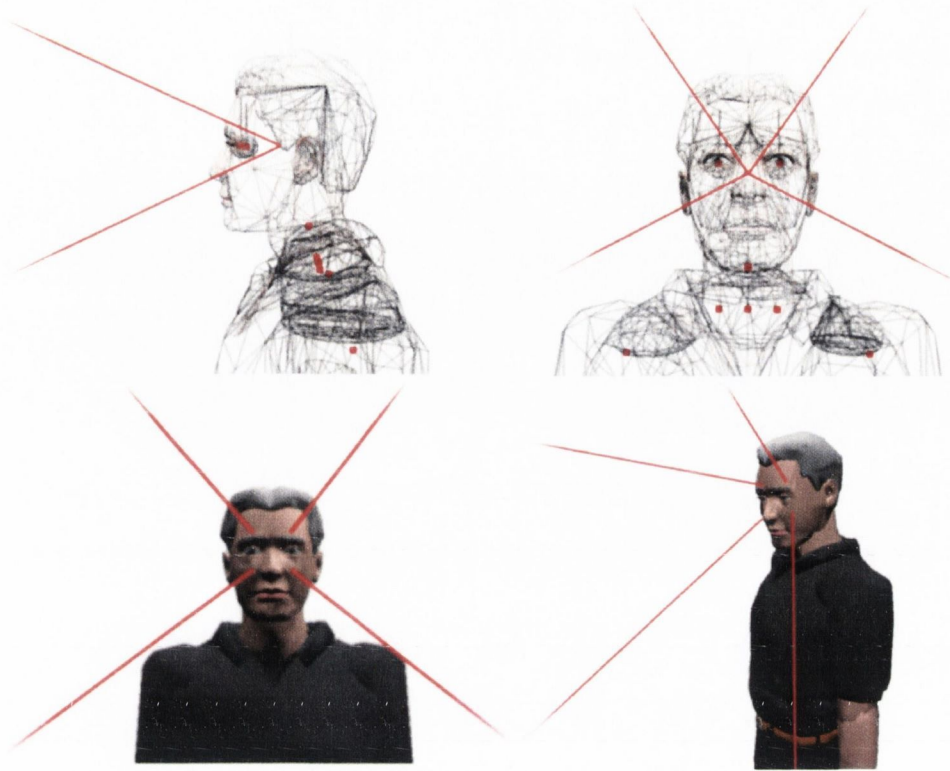


Figure 4.5: Origin and field-of-view of the agent's view frustum (depicted above by the red lines). The visual sensing system is monocular in nature.

single update of the agent's visual sensing mechanism. These renderings, referred to respectively as the *peripheral rendering*, *foveal rendering* and *full-scene rendering*, are depicted in Figure 4.4 and will now be discussed in more detail.

The foveal and peripheral renderings represent a coarse approximation of the human eye, where acuity is non-linear across the visual field. The foveal rendering is 128x128 pixels and operates over a relatively small field-of-view (half the field-of-view of the peripheral rendering). This rendering approximates the area of the eye with the highest acuity. In foveal rendering mode, objects are rendered according to their unique false-colours, with lighting and textures disabled, as previously described.

The 128x128 peripheral rendering (see Figure 4.4(b)) takes place across the full field-of-view of the agent, 100° around the y -axis of the camera, and approximates the area of lower acuity surrounding the fovea of the eye. In this mode, objects are

again assigned unique false-colours.

The higher resolution foveal rendering (see Figure 4.4(c)) plays an important functional purpose. These two renderings are relatively small in size (128x128) in order to provide an efficient and speedy solution to the vision problem. However, because the peripheral rendering takes place over a wider area, it is possible that objects that are far away from the viewer and do not occupy enough screen space to be rendered as at least a single pixel in the final rendering may be missed. The foveal rendering ensures that, at least nearer to the center of vision, objects occupying a smaller screen space will still be picked up, much in line with real visual systems. Moreover, objects in the peripheral view may be later grouped according to a perceptual grouping principle, to provide differing perceptual granularity, something that is useful for storage purposes in the agent's memory.

The third rendering is a full-scene 256x256 view (see Figure 4.4(d)). Here, the scene is rendered as normal, with textures and lighting enabled; false-colours are not used. This view is necessary for the attention component, which operates in a scene-based manner. That is, the attention component uses, among other features, colour and brightness contrast in order to process the agent's viewpoint. Because the full-scene view is not encoded in any way (as with the false-coloured information), object information is not recoverable from this rendering alone.

To summarise, none of these renderings need to be particularly large: the *peripheral* and *foveal* renderings are 128x128 pixels in size, while the *full-scene rendering* is 256x256 pixels. Both the foveal and peripheral versions are false-coloured renderings, while the full-scene version is a normal rendering with lighting and texturing enabled.

4.2.2 Scene Perception

It is important to distinguish between visual sensing and visual perception; as has been noted many times in the psychology literature, *perception is not sensation*. Internal human models of the environment consist not only of what the human senses, but also of their interpretation of what has been sensed. The image on our retina is two-dimensional, yet our impression of the world is as a three dimensional scene with depth and movement. Consider a pixelated display consisting of a number of filled locations. Although the display consists of nothing more than black and white locations, we may perceive an object. Prevailing theories of perception suggest that

perception begins with an analysis of initial sensations that become progressively more complex until an internal representation, or percept, is formed [Par00].

We use the false-colouring paradigm as a flexible basis for deriving a simplified parallel to scene perception in humans. This consists of object-centered perception, object recognition and perceptual grouping and acts as an important bridge between visual sensing and the visual attention module presented in Section 5.3.

Object-Centered Perception

Given a scene, object-centered perception consists of organising the basic features of the scene into mental constructs such as objects. In humans, this corresponds to early visual processing that starts with the processing of a two dimensional image on the retina by various configurations of ganglion cells and continues with higher-level processing by other areas in the brain, such as the visual cortex (see Section 3.2).

Our method provides a very simplified and quick parallel to some early visual processing and is as follows: after the three views, discussed in Section 4.2.1, have been rendered from the viewpoint of the virtual human, the elements of the two false-colour views (foveal rendering and peripheral rendering) are extracted into two lists of stimuli, referred to as *proximal stimuli*.

In the psychology literature, there are two types of stimuli involved in the perception of an environment (see [Gol96] for overview). A *distal stimulus* refers to what is considered to be the *actual* object in its environment. A *proximal stimulus* is defined as an energy pattern that impinges on the observer's sensory receptors, in other words, the image that is formed on the retina of the eye. Depending on the sensory modality, proximal stimuli may be associated to different degrees with distal stimuli; the important point to note is that observers depend on proximal stimuli rather than distal stimuli for perceiving their world. Figure 4.6 illustrates how a proximal stimulus is the image of a distal stimulus as it appears on the retina of the agent.

A proximal stimulus corresponds to all pixels in the scene containing an object's unique RGB reference value. Thus, each proximal stimulus corresponds to an object in the agent's eye-space, although resolution to specific objects has not yet taken place. The proximal stimulus structure is defined as consisting of a false-colour identifier for the object, screen-space bounding box coordinates, the size of the object in pixels and the average colour of the object. The average colour of the object is

determined by performing a look-up of the pixels in the full-scene rendering that correspond to the object's reference colour in the false-coloured rendering.

Proximal stimuli are important because they represent the agent's view-space perception of objects in the virtual world without doing any sort of look-up of actual object information from the scene database. They could be considered as early percepts, since they do not contain any of the information about the database objects that they refer to. Proximal stimuli are useful for representing an agent's early knowledge about its visual environment and are useful for perceptual grouping.

Proximal Stimuli Extraction

Proximal stimuli are extracted separately from the periphery and fovea false-coloured renderings. The extraction takes place for each false-coloured rendering by scanning every pixel in the respective rendering for its false-colour RGB value. The corresponding proximal stimuli list is then searched for the value: if it exists already, the information is updated to account for the new size, bounding box and average colour components. Otherwise, a new proximal stimulus is added to the list and scanning proceeds.

After proximal stimuli have been extracted from the false-coloured views, they may be later resolved to *observations* in a phase that is analogous to object recognition. In the same vein as Kuffner and Latombe [KL99], an observation structure will contain references to actual objects in the scene database. Unlike proximal stimuli, observations are also suitable for longer term storage in memory, although they require more storage space than proximal stimuli. See Section 4.3.1 and Tables 4.3 and 4.2 for a comparison of the proximal stimuli and observation structures.

The proximal stimuli list for the fovea is iterated through and matched up with database objects using the unique colour codings. Memory is then searched to see if an observation already exists for the object in question. If a memory entry already exists, then it is updated with the object's current details from the scene database, otherwise the object is entered into memory as a new entry. See Section 4.3.1 and Table 4.1 for more information on memory entries and observations.

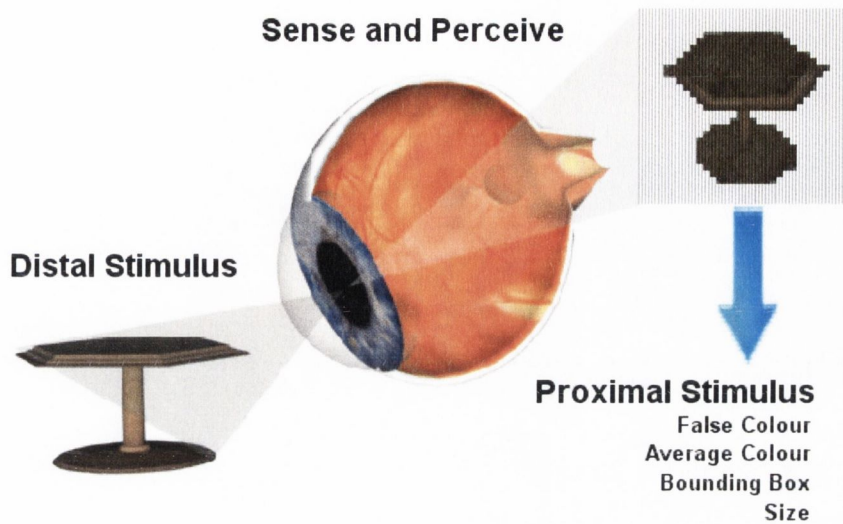


Figure 4.6: Proximal stimuli are those energy patterns impinging on an observer’s sensory receptors. They form the first stage in providing virtual humans with an internal perception of their environment.

Perceptual Grouping

Proximal stimuli may also be grouped using a perceptual grouping approach. Two methods are provided for doing this. The first method relies on objects being assigned static group identifiers manually in the scene definition file. For example, three barstools in close proximity to each other may be assigned the same group identifier, thus denoting them as a single perceptual entity. While this method is useful for static geometry, such as buildings, it has a number of disadvantages. First of all, it is a manual method; the animator is given the task of determining groups and specifying group identifiers for each object. This can be a cumbersome process, especially for scenes with large numbers of objects. Secondly, since the grouping must be done in a pre-processed manner, it is not suitable for grouping dynamic objects that may be located in different positions at different times in the simulation. Finally, this method is global as it is not based on view direction of the agent. As an agent’s view changes, different perceptual groupings may be formed. Given the barstool example mentioned above, from one angle, all three stools might be visible and therefore define a group. However, from a different angle, perhaps only a part of one of the stools might be visible, thus making it less likely that the stool would

be grouped. Such a case is not handled by a method that does not account for the agent's view in some way.

Therefore, we also provide an automatic perceptual grouping mechanism that can group stimuli at runtime using view-based information available to the agent. The computational grouping model is based on the model proposed by Thórisson for application to human-computer interaction [Thó94]. The algorithm codes object features into a multidimensional feature space in order to simulate perceptual grouping according to both proximity and similarity. The algorithm concurs with experimental evidence from brain research that suggests that features are projected into separate spaces containing the relevant feature as well as positional information.

It progresses as follows: a proximity calculation is made for every object-pair based on their two-dimensional separation distance. A perceptual grouping graph is then constructed where the vertices of the graph are objects and the graph edges represent associated proximity scores (see Algorithm 2). A similarity score is calculated for each edge in this graph along a single feature dimension, such as colour or size. Higher scores are assigned for objects that have a high similarity along a particular feature dimension. The final score is then weighted by the pair's proximity score. At the end of this process, each edge contains a score of similarity weighted by proximity for the pair that it connects. The weighted edge lists are then ordered and searched for differences between adjacent edge scores. When differences are found, edges compared so far are grouped. Groups that contain the same objects are further combined to provide larger object clusters that form the output of the algorithm (see Figure 4.7).

The original algorithm proposed by Thórisson presumes that the colour, shape and size spaces are discrete spaces. Colour is represented as a circular space containing eight colour values. Shape is represented by interpolations between square shapes and circular shapes. Object size is represented as being small, medium or large. Unlike Thórisson's algorithm, continuous spaces are used here for colour and size. Shape is not considered in this implementation due to the added overhead of determining shape similarity between arbitrary arrangements of pixels.

The synthetic vision system outlined in this chapter is useful for performing perceptual grouping of stimuli in the environment. By cross-referencing the false-coloured rendering with the full-scene rendering, view-based information about stimuli can be gathered without having to resolve stimuli into objects in the world

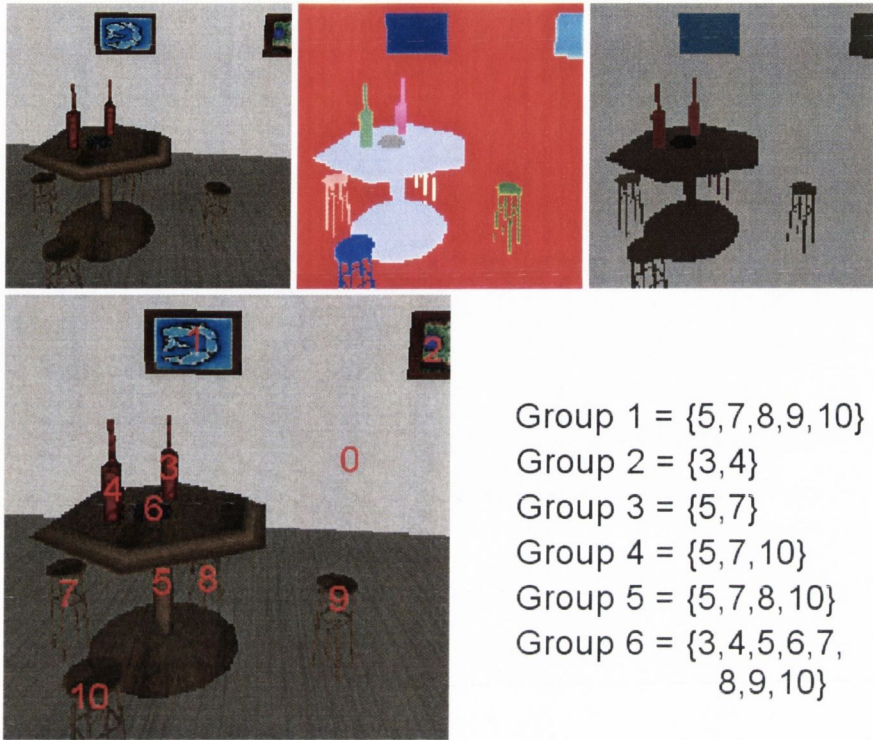
database, thus saving time on database searches. The size metric used by the grouping algorithm corresponds to the number of pixels contained in the projection of the respective object onto the agent's view-plane. Size information is accumulated as the false-coloured views are extracted into proximal stimuli representations. Colour is determined by conducting a look-up of the pixel in the full-scene rendering corresponding to the current false-colour pixel. Colours are accumulated during the extraction pass and then averaged to provide a single average colour for each stimulus. Bounding box information is also maintained during the extraction phase in order to produce extents of each stimulus; the view-space distance between stimuli is then calculated as the Euclidean distance between the centroid of the bounding box of each stimulus.

In order to create the edge lists, a measure of similarity must be made for the size and colour features. Size similarity is calculated as the difference in the normalised size scores for the objects in each respective object pair. Colour similarity, or difference, is calculated as the Euclidean distance between the respective average stimulus colours. Average colours are first converted to the CIE 1976 ($L^*u^*v^*$) colour space (see [WS82] for an overview) and then the Euclidean distance, or perceived colour difference, is calculated according to Equation 4.2.

$$\Delta E_{RGB} = \sqrt{\Delta R^2 + \Delta G^2 + \Delta B^2} \quad (4.1)$$

$$\Delta E_{uv} = \sqrt{\Delta(L^*)^2 + \Delta(u^*)^2 + \Delta(v^*)^2} \quad (4.2)$$

Colours are converted to the CIE ($L^*u^*v^*$) colour space because it attempts to approximate the way in which human observers would perceive the difference between the colours of two objects. That is, the Euclidean distance measurement in RGB colour space (Equation 4.1) does not produce a good indicator of colour differences as they would be perceived by a human, whereas the CIE ($L^*u^*v^*$) system attempts to mimic the logarithmic response of the human eye: it is nearly linear with respect to visual perception, forming a much better measurement of colour difference as perceived by a human. In order to do the conversion, colour values are first converted from RGB tri-stimulus values to CIE 1931 XYZ tri-stimulus values and from there to CIE ($L^*u^*v^*$) space. This implementation incorporates a simple conversion modification proposed by Nemcsics [Nem93] who found in experiments



Groups

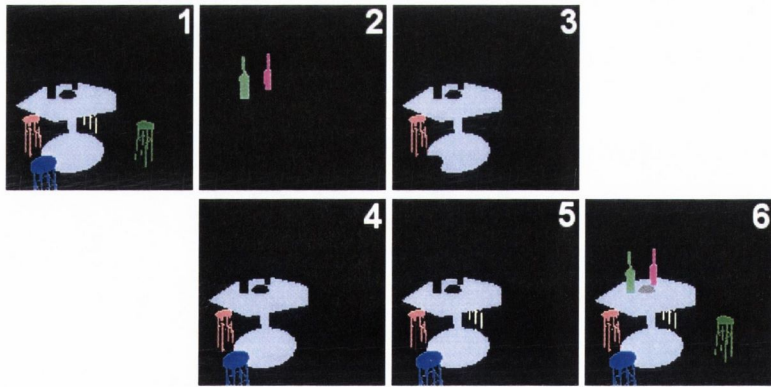


Figure 4.7: Perceptual grouping based on proximity and colour using a false-colour and full-scene rendering. The top row illustrates the input full-scene rendering (left), corresponding false-colour image (center), and averaged colour information for the scene (right). Objects are numbered from zero through to ten, and processed by the grouping algorithm to produce six groups, depicted at the bottom.

with 2500 observers that brightness, L , is better approximated for complex environments as being proportional to \sqrt{Y} , rather than $\sqrt[3]{Y}$ as proposed in the CIE ($L^*u^*v^*$) formula.

4.3 Synthetic Memory

Some form of internal storage system is crucial for agents that are expected to act in a plausible manner in their environment. If an agent has no storage mechanism, then it can never have an internal representation of its world; that is, its view of the world will always match the contents of the scene database and the database always contains perfect information about the world. Essentially, this means that it can never have a perception of its environment: it can only sense it.

Memory also plays a more direct role in influencing everyday human behaviour. Consider the example of a search for an item that is outside of the agent's view. In the case where the item has been seen previously and remembered, the search task can begin with the remembered position of the object. In contrast, if there was no memory of the position of the item, the agent would be obliged to embark on a lengthy search. As with living creatures, autonomous agents also require a storage mechanism to differentiate between what they have and have not observed. An agent that automatically knows the location of every object in the scene will destroy its plausibility with respect to a human viewer, while an agent that has no memory of its surroundings will appear to be simple-minded when conducting everyday tasks.

In a similar vein to the synthetic vision module, the implementation of the synthetic memory model uses a simplified version of the model from psychology. The system of memory proposed here is based on what is referred to as the stage theory, or *modal model*, of memory [AS68] and is used to store information relating to object's that have been sensed. Stage theory has had an enormous influence on psychology, and although more elaborate models have been proposed since, it has played a crucial role in a number of more elaborate theories such as Search of Associative Memory, or *S.A.M.* [GS84]. The central posit of the stage theory (see Section 3.3.1), is that information goes through a progressive number of memory stages and there are discreet information stores at each stage. In general, information that travels to later stages in the system tends to persist for longer. While there are

different and more specific theories of memory, the stage theory of memory provides a very useful paradigm for autonomous virtual humans. It is particularly suitable in providing us with a storage structure while also adhering to the useful concept of successive stages of filtering and elaboration of perceived stimuli (see Figure 4.8). Furthermore, stage theory provides a general, clearly defined, filtering and storage mechanism that is free from low-level complexities and the computational overhead of more elaborate models.

4.3.1 Memory Entries and Observations

The memory system presented here represents a simplification of the way in which information is thought to be stored by real humans. Although there are a number of different types of memory, such as episodic and semantic memory, this system provides agents with an object-based memory of environmental stimuli. Memories are considered to be composed of discrete constructs called *memory entries*. Memory entries contain information both about what has been remembered and other information pertinent to the memory system, allowing a recording of how recently the object was perceived and when it should be removed from memory (see Table 4.1).

Trace strength, *activation* and *decay factor* are used to determine when a memory entry should be removed or forgotten. These three attributes are used with a decay function, presented in Section 4.3.2, to provide an estimation of the endurance of a memory entry. In the model presented here, activation is the amount of time since a memory entry was last accessed and the decay factor represents how quickly a memory entry will decay once in memory; it can be thought of as the slope of memory decay. Trace strength is an indication of the strength of a memory and is regarded here as being dependent on the decay factor of the memory entry and the activation of the memory.

Uncertainty is a value between 0 and 1 that represents the completeness of the virtual human's internal representation of the stimulus in question. Essentially it is a simplified measurement representing how much is known about the stimulus, where a value of zero indicates that the agent has a full internal representation of the object. Uncertainty is used for determining when memory entries should be moved between certain memory stages and also for calculating uncertainty factors; metrics that can be used for guiding the virtual human's attention. It represents

Memory Entry Structure

Member	Meaning
<i>trace strength</i>	The strength of the memory
<i>decay factor</i>	How quickly the memory decays
<i>activation</i>	Amount of time since the memory was rehearsed
<i>rehearsal</i>	Number of times that a memory has been rehearsed
<i>uncertainty</i>	Completeness of memory representation
<i>proximal stimulus</i>	Perceived stimulus
<i>observation*</i>	Observation corresponding to perceived stimulus

Table 4.1: The memory entry structure contains information about memory contents and how to manage memories. Observations elaborate on proximal stimuli by being resolved with the scene database, but are not always stored in the memory entry.

new information that is found out about an object, rather than the amount to which current information is practiced or remembered. An agent's uncertainty about an object may be reduced by paying attention to it.

Rehearsal represents the amount that a memory has been practiced by a virtual human and is used to determine when a memory should be moved from one memory stage to the next. Unlike uncertainty, rehearsal represents how much an existing memory has been practised rather than representing the amount of knowledge available to the virtual human about that object. This means that an agent may rehearse memories without the need to be currently sensing them.

There are two types of memory entries representing different stages of elaboration of perceived stimuli. The first form of memory entries store only proximal stimulus data and are useful for earlier stages of processing where a more elaborate internal representation of the stimulus is not available (see Table 4.2). Memory entries may also contain more detailed information about stimuli. Such *observations* (Table 4.3) elaborate on proximal stimuli information by providing a snapshot of the information from the scene database about the object, to encode a more complete perception of the stimulus by the agent.

Because observations persist in memory, they may often correspond to out-of-date information that is no longer accurate. An agent therefore has a form of internal representation of the environment that, in a similar way to a real human, may be imperfect. Only by continually sensing and attending to the environment can the agent maintain a relatively up-to-date internal model.

Proximal Stimulus Structure

Member	Meaning
<i>false-colour</i>	The stimulus false colour
<i>bounding box</i>	2D Bounding box of the stimulus
<i>size</i>	Size of the stimulus in pixels
<i>average colour</i>	The average colour of the stimulus

Table 4.2: The proximal stimuli structure contains only view-based information about stimuli from the point of view of the virtual human.

Observation Structure

Member	Meaning
<i>object ID</i>	Unique identifier of the object
<i>object</i>	Pointer to the object in the scene database
<i>wsTransform</i>	world-space transformation of the object
<i>time</i>	Time at which the observation was made
<i>proximal stimulus</i>	Perceived stimulus information

Table 4.3: The observation structure forms what is considered the most complete internal representation that a virtual human may have of an object.

4.3.2 Forgetting

Forgetting is of vital importance to a computational model of memory for virtual humans. Items that are considered to be remembered by the virtual human must be stored in the physical memory of the computer system, which is finite. Yet, if the virtual human is to have a perception of its environment, it needs to store internal models of perceived stimuli that may differ from the records in the environmental database and those made by other virtual humans. One way of reducing the amount of information stored by the virtual human is to filter information that is stored, so that only information that is deemed to be important will be stored in the virtual human's memory. This can be achieved using successive stages of filtering and an attention mechanism. A second way, that is also presented here, is to provide a scheme for the removal of memory entries that are no longer of importance. This allows for the implementation of *retention* and *forgetting* processes in virtual humans.

Forgetting is an active area of research in psychology and the nature of forgetting still holds many mysteries yet to be unravelled. Unfortunately, the nature of human memory can make it difficult to provide illumination on how to construct a useful

system for virtual humans. For example, it is thought that the capacity of long term memory may be infinite; our forgetfulness may often be attributed to an inability to retrieve memories as opposed to them being no longer stored. In the current system, a simplified model of forgetting is implemented that follows the *decay hypothesis* of retention as suggested by Ebbinghaus [Ebb13] (see [EK00] for overview). It is therefore presumed that all forgetting takes place due to decay and that forgotten memories may be totally removed from the virtual human's memory system.

Studies of human memory suggest that retention and forgetting consist of rapid initial decay and slower later decay. Therefore, as time passes, losses become smaller and smaller. Power functions are generally accepted as being descriptive of human retention processes. Such functions illustrate the decline in the ability to recall information from memory as dependent on a *retention interval*, the amount of time since a memory was last entered or rehearsed, with large initial losses followed by smaller and smaller losses.

The work here makes a number of simplifications for memory recall and decay. First of all, if an item is in memory at all, then it is presumed that the probability of recalling it is always 1. That is, an item that is in memory will always be successfully recalled. In this way, trace decay is used only to determine when a memory entry should be forgotten, as oppose to how successfully information is recalled and to what extent. Memory entries are presumed to decay according to an exponential decay function, as used in the decline function in the short-term store of Atkinson and Shiffrin's buffer model [AS68]. Memory entries are removed when their trace strengths fall below a threshold trace value. The exponential function is written as,

$$Y = (A)e^{-BX}, -B < 0 \quad (4.3)$$

where Y is the recall probability, e is the base of natural logs, X is the retention interval and A and B are constants that define the shape of the curve. B represents the slope of the curve and A is an intercept specifying the value of Y when X is zero. Thus, for each trace, recall probability decreases by a constant percentage over each unit of time.

4.3.3 Stages of Memory

Each memory stage in our system is parameterised according to *capacity*, *threshold*, *duration* and *rehearsal*. By varying these parameters, the different memory stages elaborated on in the stage theory of memory may be modelled.

Capacity refers to the maximum number of memory entries that a memory stage can store at any one time. The *threshold* parameter is used to decide when memory entries should be removed from a memory stage or *forgotten*. While the trace strength of a memory entry is greater than this threshold, the item remains in memory and the probability of its recall is presumed to be 1. Memory entries are removed from a memory stage when their trace strength falls below the threshold value. *Duration* is an indication of the amount of time that a memory entry will persist in a memory stage without being rehearsed in any way. *Rehearsal* is another threshold, but is used to decide when an entry should be moved to the next memory stage. Rehearsal may be increased by attending to an object.

In line with the modal model of memory, three memory stages are presented based on variations of these parameters: short-term sensory storage, short-term memory and long-term memory.

Short-Term Sensory Storage

The short-term sensory storage, or *STSS*, is the first storage stage for stimuli that have passed through initial visual filtering. It therefore contains memory entries relating to objects that are in the agent's view-frustum and are not totally occluded by other objects.

Operationally, the *STSS* consists of a list of proximal stimuli from the false-coloured renderings for the periphery and fovea renderings (see Section 4.2.2 and Table 4.2). Unlike observations, proximal stimuli do not contain resolved database object information and do not persist for very long between visual system updates. That is, the entries in the *STSS* correspond only to the 2D optical images deemed to be on the retina of the virtual human.

The capacity of the *STSS* is regarded as being quite large, allowing storage of a large number of proximal stimuli. The *STSS* is updated with each refresh of the virtual human's viewpoint rendering. When a proximal stimulus is encoded in the *STSS*, a time-stamp is updated to the current time and the trace-strength is set to the maximum value for the *STSS* memory stage. Memory entries in the *STSS*

Successive Stages of Filtering

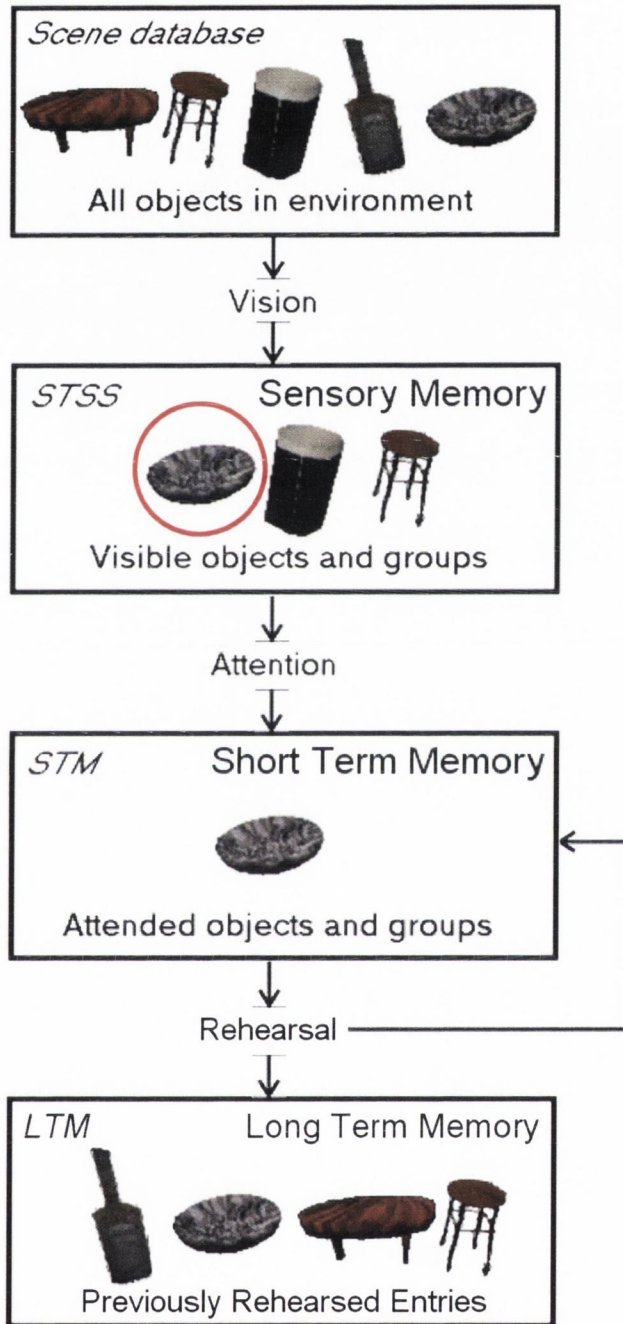


Figure 4.8: Successive stages of filtering. Synthetic vision and attention act as filters, selecting only certain information for further processing by the virtual human. Short-term sensory storage and short-term memory allow filtered items to persist for temporal processing. Rehearsed entries are moved into the long-term memory.

are regarded as having a very fast rate of decay. Because of this, entries in the STSS correspond to objects that are currently in view, or were in the agent's view very recently. It is therefore essential that information in this stage be *attended to* in order to transfer it to the next stage of memory, the short-term memory. The application of attention to memory entries is deemed to be the chief way of moving them into the next memory stage.

Short-Term Memory

The short-term memory, or *STM*, relates to our thoughts at any given moment in time. It is created by attention to an external stimulus or internal thoughts. Short-term memory is limited both in duration and by the number of units of information that can be processed at any one time. Research suggests that the STM can process between 7 ± 2 and 5 ± 2 units or chunks of information [Mil56]. These units correspond to letters, numbers, and also larger units such as words and phrases. The STM is therefore of limited capacity, but *chunking* may allow for a better encoding of information.

Memory entries are removed, or forgotten, from the STM under two conditions: they are displaced by newer memories when the STM is full and they also decay over time. Rehearsal occurs when attention is paid to a specific object over a period of time and is necessary for it to remain in the STM as well as move into the next memory stage. The rate of decay of information in the STM is slower than the STSS, but still relatively quick, with items decaying after 200 milliseconds to 20 seconds, with rehearsal.

As noted in Itti and Koch [IK00], operationally, information can be said to be attended to if it enters short-term memory and remains there long enough to be voluntarily reported. Because the STM relates to our awareness of an object, memory entry rehearsal may only take place when the entry is in the STM. In a similar way, object uncertainty may only be reduced when it is in both the STSS, i.e. being looked at, and in the STM, i.e. is also being attended to.

Long-Term Memory

Long-term memory, or *LTM*, provides storage for information over a long period of time. The capacity of long-term memory is thought to be infinite. In fact, it is possible that forgetting in the LTM is due primarily to the loss of retrieval

cures rather than actual decay of memories. For the purposes of the current model, memories that can no longer be retrieved due to lack of sufficient cues are considered to be the same as decayed entries. That is, such items decay and are completely removed when they fall below the threshold trace-strength.

There are three main activities related to long-term memory: storage, deletion and retrieval. For the purposes of the current model, memories that can no longer be retrieved, due for example to a lack of retrieval cues, are considered to be the same as deleted entries.

Long-term memory is considered to have a 'back and forth' relationship with short-term memory; items may be moved into short-term memory for processing and then moved back into long-term memory for storage. Information from short-term memory is stored in long-term memory by rehearsal. Thus, repeated exposure to or rehearsal of a stimulus will result in it being encoded in long-term memory.

Items are presumed to move from the STM to the LTM when they have been rehearsed to a sufficient degree. Unlike the other memory stores, there are no strict upper bounds on trace-strength in the LTM. If an item is frequently rehearsed in the STM, then it can remain easily accessible in the LTM.

Memory Dynamics

Provided with the three memory storage areas described, some methods are needed to decide how memory entries move from one stage to another, are deleted and are updated. For example, memory entries with varying attributes such as trace strength and uncertainty have been proposed, but when should a memory decay and when should a virtual human's uncertainty about an object be reduced?

We propose eight different possibilities for memory updates depending on the memory stages that a memory entry occupies (Table 4.4). These rules are based on the idea that objects in the STSS are, or were recently, visible, that objects in the STM are being attended to and objects in the LTM have previously been attended to and are now in long-term storage. These rules are as follows:

1. If there is no memory entry for an object in the STSS, STM or the LTM, then it is presumed that the agent has no internal representation of the object whatsoever.
2. A memory entry that exists only in the STSS signifies that an object was

Case	STSS	STM	LTM	Result
1	X	X	X	No internal representation
2	✓	X	X	Decay memory trace in STSS
3	X	✓	X	Decay memory trace in STM, increase rehearsal
4	✓	✓	X	Increase trace strength and rehearsal. Reduce uncertainty.
5	X	X	✓	Decay memory trace in LTM
6	✓	X	✓	Decay memory trace in STSS
7	X	✓	✓	Decay memory trace in STM, increase rehearsal
8	✓	✓	✓	Increase trace strength and rehearsal. Reduce uncertainty.

Table 4.4: Indication of memory dynamics for a single object based on the memory stages that contain an entry referring to the object. A tick indicates that a memory entry exists for the object in that particular memory stage. Rules for eight different possibilities are presented.

recently within the agent’s view field, but is not being attended to and has not been rehearsed before. In such a case, the trace strength of the memory entry in the STSS decays according to the memory decay function.

3. When a memory entry is only in the STM then it means that the agent is processing it, or *attending* to it, even though the object is no longer in the agent’s view. As long as this is the case, the entry’s trace strength continues to decay. However, the rehearsal value is increased in order to signify internal processing of the entry.
4. When an item is present both in the STSS and in the STM, then it is or was recently in the view of the agent and is also being attended to by the agent. In this case, the trace strength and rehearsal values for the entry in the STM are increased. The uncertainty value for the entry is also decreased in the STM to represent new incoming sensory information that is being used to contribute to an improved internal representation of the item. Uncertainty is not reduced since no new sensory information is available.
5. Memory entries that exist only in the LTM are subject to decay according to the decay function. Rehearsal and uncertainty parameters do not change since the agent is neither receiving new information about the object from the STSS nor paying attention to it through the STM.

6. Memories that are present in both the STSS and LTM result in normal STSS decay. In the current model, LTM decay is paused while corresponding entries exist in the STSS. This presumes some sort of connection between the STSS and LTM that bypasses the STM. Although such a connection was not part of the original version of the model proposed in [AS68], the abandonment of strict serial information flow between each memory component is seen as a necessary modification to support experimental evidence (see Section 3.3.1 or [Pas98] and [BB96]).
7. If a memory is present in both the STM and the LTM, then trace strength decay takes place in the STM, but not in the LTM. Rehearsal of the entry takes place since the entry is being attended to, although entry uncertainty does not change since the agent is not receiving new information about the object from its senses.
8. When memories exist for an object in all three stages, then the entry trace strength and rehearsal is increased. Object uncertainty is also reduced. This case represents a scenario where the agent is or has recently looked at a stimulus, is paying attention to it and has also previously rehearsed the stimulus and paid attention to it.

These rules form the basis of all updates that take place in the memory system. When used in conjunction with decay thresholds and rehearsal thresholds, they provide a way for items to move between memory stages so that only the most important environmental stimuli are retained in memory.

4.4 Discussion

This chapter presented a model for sensing and storing information about the environment by applying successive stages of filtering to objects in the scene database in order to approximate an agent's internal model based on its perception of the environment. When implementing synthetic senses, we seek to approximate the equivalent senses in the human system while minimising the amount of computational overhead; thus many necessary simplifications are made.

There are a number of compelling reasons for adopting a computer vision technique for synthetic sensing. First of all, it is a simple and fast way to extract useful

information from the environment [Blu97]. Underlying graphics hardware can be taken advantage of and since the object visibility calculation is fundamentally a rendering operation, all of the techniques that have been developed to speed up the rendering of large scenes can be adopted. These include scene-graph management and caching, hierarchical level-of-detail (LOD), and frame-to-frame coherency [KL99]. Secondly, synthetic vision may scale better than other techniques in complex environments. By its very nature, the visual system provides a controllable filtering mechanism so as not to overwhelm our limited cognitive abilities. Thirdly, the approach makes the actor less dependent on the underlying implementation of the environment because no direct queries of the database are made. Finally, as pointed out in [Blu97], “believable behaviour begins with believable perception”. There are also further beneficial side effects to this method: occluded objects are implicitly handled in a static scene and have been shown to be extendible to dynamic scenes through the use of memory [Kuf99].

Although the synthetic vision component is inspired by Noser et al. [NRTT95], the current work extends it by incorporating some crude forms of visual perception. Firstly, the acuity of the human eye is roughly approximated in order to provide a higher resolution scene perception nearer to the foveal region of the eye. Since the false-colour renderings are of limited resolution in order to provide efficiency, the use of a second higher resolution rendering over a smaller field of view means that small or distant objects will still be detected by the virtual human. A mechanism is also provided for altering the granularity at which objects within the scene are segmented. By either predefining groups or running the automatic grouping algorithm, items may be grouped according to Gestalt principles, primarily proximity, something that has consequences for visual chunking in short-term memory.

Unlike other approaches, we have also implemented a multi-stage model of memory inspired by research in psychology. This memory model is based on the work of Atkinson and Shiffrin [AS68] and incorporates present-day modifications suggesting the existence of a specialised visual short-term memory and allowing some latitude from strict serialisation of information flow between memory components, as suggested by experimental evidence and as outlined by Pashler [Pas98]. In practical terms, the memory further refines the notion of visual information filtering, initiated by the visual system, so as to limit the amount of environmental information that must be processed by the virtual human.

One of the main arguments against incorporating an independent memory system for virtual humans appears to be that of storage. It has been suggested that a memory model entails the storage of multiple copies of the world database [Gil01]. In the worst case, this would be the same size as the world database for an agent that had perceived every object in the world. This need not be the case, however, if filtering and forgetting mechanisms are applied to the memory system. Filtering makes sure that only important information makes it through to the storage stages, while forgetting clears out information that is no longer as valuable as it used to be. It appears that this filtering and forgetting allows real humans to cope with the huge amounts of data that they sense [Hil99]. In our framework, synthetic vision is just the first in a number of filters that reduce the amount of data that must be stored and processed by each agent. Memory is implemented as the main filtering mechanism; only information that is deemed to be important will be stored for any length of time.

Kuffner and Latombe [KL99] present a flat memory system for objects and point to rule-based models from artificial intelligence for facilitating forgetting processes. Such methods require rules, goals and other top-down motivations on the part of the character in order to decide what is of relevance and therefore what to forget. The memory system presented here has a number of advantages over a flat-memory system by providing filtering so that only that important items get through to later, longer term memory in the first place. In this case, importance is not defined by explicit rules, but rather implicitly by those stimuli that are attended to by the agent. This approach is more generic and would be compatible with and augmented by rule-based approaches for forgetting.

Chopra-Khullar [CK99] does not provide an explicit memory model for the storage of previously seen objects: object-related memory thresholds are used to provide a measure of uncertainty, but only for monitoring type attention behaviours. For example, when an agent is walking on a path, it will pay attention to the path when its uncertainty of it has exceeded some threshold value. In this work, uncertainty is stored for all objects in the environment and represents the totality of an agent's internal representation of that object. This is useful for creating explorative behaviours.

Unlike previous systems, the system presented here also allows for the elaboration of an agent's internal perception of visual stimuli. Proximal stimuli represent

early and partial internal models that have not received considerable attention by the agent and therefore contain only limited information. In contrast, observations represent more complete internal representations, where geometric object data is made available from the scene database, while a local copy of positional and other information is also stored by the agent. In this way, observations contain more information than proximal stimuli, but may also be imperfect if they are not frequently attended to.

Algorithm 1 Extracts proximal stimuli from false-coloured and full-scene renderings.

Input : False-coloured rendering FR , full-scene rendering FS

Output : List of proximal stimuli PS

EXTRACTPROXIMALSTIMULI(*Image FR, Image FS*)

for all pixels p in FR **do**

$falseColour \leftarrow p$

if $falseColour$ exists in PS **then**

$PS[entry][size] \leftarrow PS[entry][size] + 1$

$PS[entry][bounding\ box] \leftarrow$ pixel coordinates of p

$PS[entry][average\ colour] \leftarrow PS[entry][average\ colour] + FS[p][colour]$

 { $FS[p]$ is the corresponding pixel in the full-scene rendering}

else

 {*new proximal stimulus*}

$PS \leftarrow$ new proximal stimulus entry

$PS[entry][false\ colour] \leftarrow falseColour$

$PS[entry][size] \leftarrow 1$

$PS[entry][bounding\ box] \leftarrow$ pixel coordinates of p

$PS[entry][average\ colour] \leftarrow FS[p][colour]$

end if

end for

Algorithm 2 Forms proximal stimuli into perceptual groups.

Input : List of proximal stimuli PS , perceptual graph PG

Output : List of perceptual groups GL

PERCEPTUALGROUP(*Proximal stimuli* PS , *Perceptual graph* PG)

{*Add the proximal stimuli as new vertices in the perceptual graph*}

for all proximal stimuli ps in PS **do**

 Add a new vertex to PG for the stimulus ps

end for

{*Create an edge between each vertex*}

for all vertex pairs $(v1, v2)$ in PG **do**

$e = \text{edge}(v1, v2)$

$e.\text{proximity} \leftarrow \text{eye-space distance}(v1, v2)$

$e.\text{colourDiff} \leftarrow \text{CIEluv difference}(v1, v2)$

$e.\text{sizeDiff} \leftarrow |v1.\text{size} - v2.\text{size}|$

$PG \leftarrow e$

end for

{*Normalise all edge feature dimensions, and weight colour and size by proximity*}

for all edges e in PG **do**

$e.\text{proximity} \leftarrow (1 - e.\text{proximity}/\text{largestProximity})$

$e.\text{colourDiff} \leftarrow (1 - e.\text{colourDiff}/\text{largestColourDiff})$

$e.\text{colourDiff} \leftarrow e.\text{colourDiff} * e.\text{proximity}$

$e.\text{sizeDiff} \leftarrow (1 - e.\text{sizeDiff}/\text{largestSizeDiff})$

$e.\text{sizeDiff} \leftarrow e.\text{sizeDiff} * e.\text{proximity}$

end for

form perceptual groups GL by iteratively comparing all edge differences along a single feature dimension

Chapter 5

Visual Attention

5.1 Introduction

Establishing a sense of presence in virtual environments is a vital step towards providing users with immersive and meaningful experiences. This is particularly true when the artificial environments in question are intended to represent the real world in some way; most of us have expectations that are based on our experiences of the real world and when these are not fulfilled adequately within the virtual world, immersion is lost. Collision detection and response algorithms are a good example of this, where the user can be prevented from walking through virtual structures or allowed to manipulate virtual objects in rough accordance to their expectations from the real world.

In this thesis, we are particularly interested in how the behaviour of virtual humans can contribute towards the sense of presence. Humans tend to notice where other humans look and evidence indicates that infants pay attention to the direction of the gaze of others from as young as three months [FJBS00]. Thus, a key notion is that, by providing virtual humans with a mechanism for noticing things in their environment, the user's sense of presence will be enhanced as they become immersed in what they perceive to be a living place. Ideally, users should be given the impression that virtual humans are interacting with their environment in the same way as living, thinking creatures would.

In a computer animation context, this notion is not uncommon; in CGI films, such as *Toy Story* by Disney Enterprises and Pixar Animation Studios, characters look at each other and can be seen to pay attention to important or interesting events

in the environment. Unfortunately, these behaviours must be created manually by teams of animators using conventional methods. When considering virtual humans, who are expected to be autonomous in a number of ways, an automatic method for generating looking behaviours must be found. At first, this would seem to be rather trivial since the whole notion of attention appears to be obvious; “everyone knows what attention is” said famous 19th century psychologist and philosopher William James. Yet the concept of attention is probably somewhat more illusive, as is highlighted by the more cautious approach of more recent research: “there may not even be an ‘it’ there to be known about” [Pas98]. Keeping this in mind, it would seem wise to take proper account of work from psychology, where there has been a great deal of research into attention and related areas, such as memory. In a similar vein to our methodology behind the synthetic sensing and memory modules, our approach here is to apply simplified versions of attention models from research in the fields of psychology and cognitive engineering. By doing this, we attempt to retain the behavioural essence of the system while avoiding the inherent computation overhead.

In this chapter we present a biologically plausible model of bottom-up visual attention from the field of cognitive engineering. This model forms the basis of the attention module which is the main control process in the system. The attention model operates by processing the full-scene rendering from the synthetic vision component in a bottom-up manner and outputs a map encoding locations in the scene likely to attract one’s attention based on the colour, intensity, orientation and motion contrast in the scene. We also extend the basic scene-based model to provide object-based information, necessary for providing dynamic scene memory and for querying object state information from the scene database.

5.2 Attention

The human information processing system is a limited resource system [And95]. At any time, our environment contains a great deal more information than we could possibly hope to process. Nonetheless, we seem to manage despite the limits of our cognitive resources. A key factor in helping us to do this is our ability to filter out and attenuate useless information and to deploy our senses to likely areas of important information in order to elaborate on it. Attention forms this information-processing

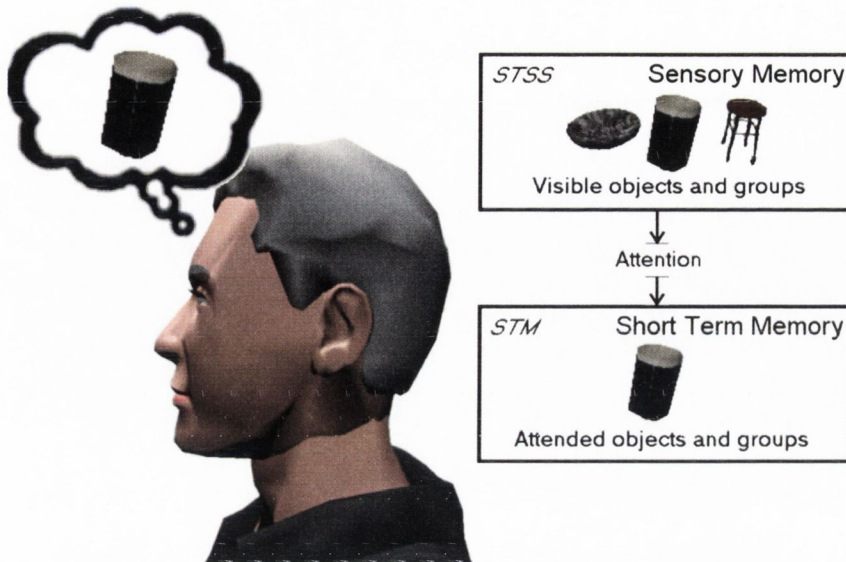


Figure 5.1: Attention acts as a filter controller to determine what items are entered into short-term memory.

bottleneck, in that it allows only a fraction of the sensory input, estimated to be in the region of 10^7 - 10^8 bits per second on the optic nerve, to reach short-term memory (see Figure 5.1) and awareness [IK01a].

Attentive behaviours are often the focus of interest of other creatures inhabiting an environment, since motive and thought can often be derived from the eyes and gaze [AC76]. Attention is not just behaviourally significant however - it is of functional necessity for sensory systems that are not omnidirectional and therefore must be directed in some way. Also, because sensory systems essentially act as filters, some mechanism must be in place to decide what information gets filtered and, as a consequence, what information is retained for further processing. This is perhaps most evident in the human visual system due to its inherent structure (see Section 3.1.1).

5.2.1 Visual Attention

Our attention model is based on the two-component framework of attention (see Section 3.2), and focuses on the pre-attentive, bottom-up component as opposed to the top-down volitional component. Pre-attentive processing is thought to run in parallel across the visual input and does not represent all parts of such a scene

equally well. Instead, there are strong responses to a few parts of the scene while there are poor responses to everything else. The processing of a scene in this manner depends very much on the contrast between different elements in the scene, or *context*.

As Yantis points out, stimulus-driven attention is often “faster and more potent” than goal-driven attention, since conscious processing effort is required for goal-driven attention [Yan98], whereas bottom-up attention is effortless. Both types of attention can operate at the same time so that visual stimuli can get into higher levels of awareness by either being wilfully brought into the focus of attention, or by winning the competition of saliency [IK00]. Since this thesis focuses on bottom-up visual attention, we will now consider this in more depth.

Bottom-Up Visual Attention

Bottom-up attention refers to the way in which stimuli in the environment may elicit our attention. Such attention is involuntary and is captured in an effortless manner, guided by local feature contrast. Important feature dimensions include orientation, colour, intensity, size, shape, texture, flicker and motion. Bottom-up attention is behaviourally significant, since it constitutes a powerful alerting mechanism that allows primates to instantly become aware of unexpected predators or dangers [Itt00].

Many models of bottom-up attention are centered about the concept of a *saliency map*. This saliency map provides an indication of areas in the image that are likely to attract attention in an exogenous, or stimulus-driven, manner due to the basic features contained in that image.

The Saliency Map

Koch and Ullman [KU85] proposed the idea of a saliency map in order to accomplish pre-attentive processing. The saliency map is a two-dimensional, topographic map encoding the amount of *pop-out* present over a visual scene. Pop-out can be thought of as being a measure of how much a location in an image stands-out, or contrasts, with its surroundings (see Figure 5.2). Evidence suggests that such locations grab one’s attention in a bottom-up manner (see Section 3.2.2). The amount of pop-out present in a location in a scene would appear to be the result of a combination of contrasting low-level visual features. Important feature dimensions include colour,

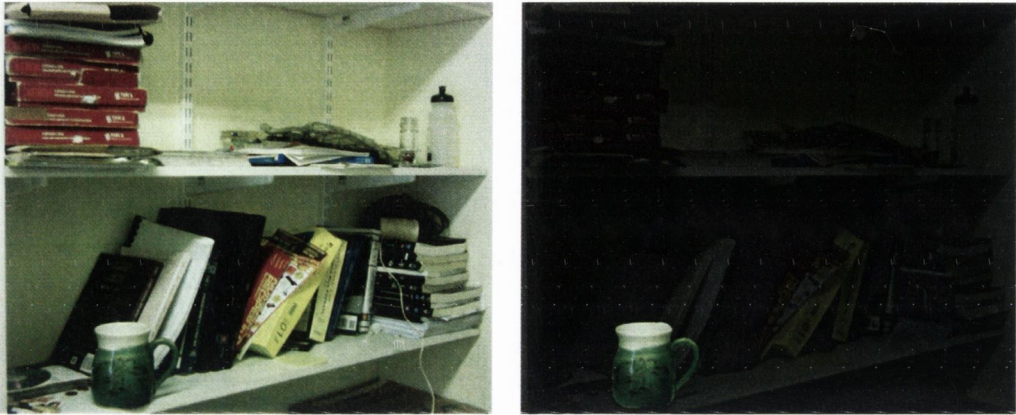


Figure 5.2: Photograph of a natural, cluttered scene. Such scenes may be processed for saliency by combining feature maps for dimensions such as colour, orientation and intensity. Right photograph illustrates the concept of pop-out; in this case all objects, apart from the mug in the bottom left corner, are darkened. The mug may elicit attention in a bottom-up manner.

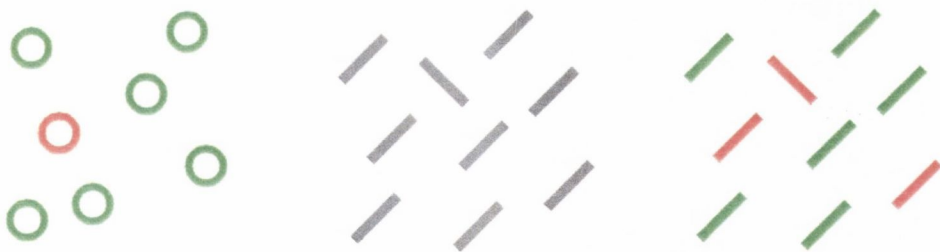


Figure 5.3: Example of feature dimensions for (from left to right) colour contrast, orientation contrast and conjunction of orientation and colour.



Figure 5.4: An example saliency map (center) for an input scene (left) and saliency map superimposed on scene (right). Lighter regions denote areas of higher saliency, signifying a high probability that such areas will attract attention in a bottom-up manner.

intensity, orientation, texture, shape, size and distance from viewer (see Figure 5.3 for examples). The saliency map is a useful concept since it provides a single unified measure of such pop-out, or saliency, in the form of a single map (Figure 5.4 illustrates a saliency map). Thus, the saliency map may be used as an indicator of conspicuous locations in a scene; by suppressing the most conspicuous location in the map, further targets of interest may be acquired.

5.3 Modelling Visual Attention

We use a computational model of bottom-up visual attention based on a biologically plausible architecture provided by Itti, Koch and Niebur (see [IKN98] [IK99] [Itt00] for relevant research). This model has its foundations in an architecture proposed by Koch and Ullman [KU85] and by Nieber and Koch [NK98]. It attempts to mimic the low-level, automatic mechanisms responsible for attracting our attention to interesting, or *salient*, locations in our environment and closely follows the neuronal architecture of the earliest hierarchical levels of visual processing. Primarily, the model considers three basic feature dimensions that contribute to the pop-out in a scene: colour, orientation and intensity. Other feature dimensions, such as shape, texture, motion and size are also important when establishing pop-out, and may be added to the model as long as a saliency map representation is possible. For example, in dynamic scenes, motion is usually an important factor in determining pop-out: we were able to add a quick motion contrast detector to the model with

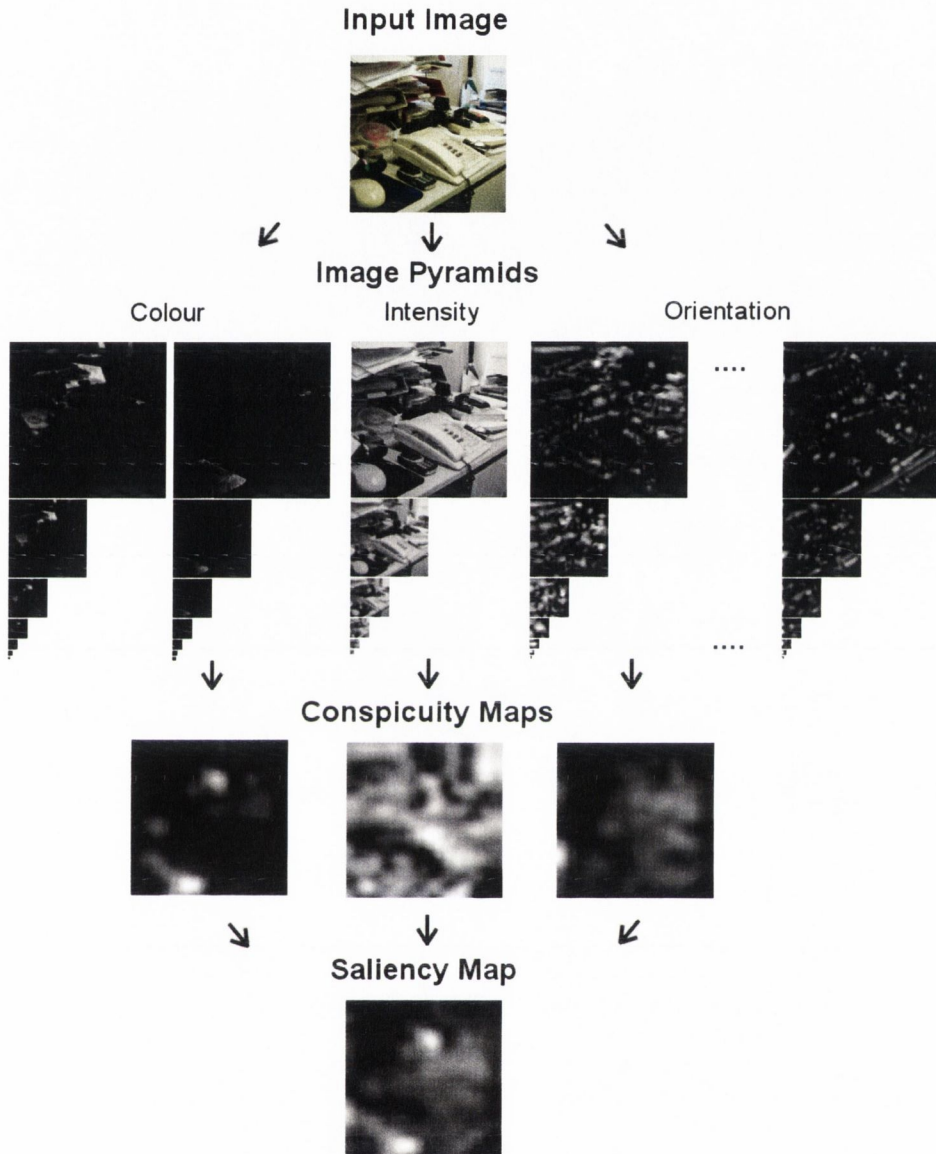


Figure 5.5: High-level schematic of the processing performed by the attention model. An input image is first split into constituent channels for entry into a number of image pyramids. Each pyramid is used to create a single respective conspicuity map. Conspicuity maps are combined at the end of the process to create the output of the attention model: a saliency map encoding pop-out across the input image.

relative ease (see Section 5.4.1).

There are a number of compelling reasons for choosing this computational model. First of all, it coarsely replicates early levels of visual processing and does so in a biologically plausible manner. Secondly, it has been shown to be effective with both natural [Itt00] and graphically rendered [YPG01] scenes. The saliency map representation provides an efficient and plausible strategy for calculating targets. Finally, various components in the model use fast implementations. Although this means that the model only approximates neurobiological vision systems, it does make its use feasible for the real-time animation of a virtual human. The model therefore appears to provide a good trade-off between computational intensity and the plausibility of results. A summary of the processes involved in the Itti model follows.

Calculating the Saliency Map

The saliency map is the primary output of the attention model and the description of its calculation therefore constitutes an overview of the model itself. This calculation takes place as follows: an input image is decomposed into a number of constituent channels. Each channel acts as the root level in a filtered image pyramid that facilitates center-surround operations for each of the features of colour, intensity and orientation to produce a number of *feature maps*. These maps are then combined into three separate *conspicuity maps*, one for each feature type. The conspicuity map can be thought of as a master map indicating the salience for that particular feature dimension. The three conspicuity maps are finally combined to produce a single saliency map that unifies pop-out for the entire scene over all feature dimensions (see Figure 5.5).

5.3.1 Center-surround Processing

The attention implementation presented by Itti processes an input image and calculates local contrast for intensity, orientation and colour features respectively as summarised in Algorithm 3 (see also Figure 5.5). These feature types are computed and combined in a centre-surround fashion using image pyramids and filtering techniques, to provide a biologically plausible system that is sensitive to local spatial contrast rather than amplitude in a given feature.

The process proceeds as follows: an input image is passed to the algorithm

and is decomposed into three constituent channels, one for intensity, two for colour (Figure 5.6). The intensity channel is obtained as the average value of the red, green and blue values in each input image location. There are two colour channels; a red-green opponency channel and a blue-yellow opponency channel. The colour channels respond maximally to the hue to which they are tuned and respond minimally to white and black values. They provide rough approximations of the sensitivity of the eye, but may be quickly computed. These three channels act as base images for a number of image pyramids, which are used to create the feature maps.

Pyramid Construction

Image pyramids are constructed from the three constituent image channels for the different feature dimensions, intensity, colour and orientation. Each image pyramid is a dyadic hierarchy of images, such that the root of the hierarchy is an input image and each successive level of the pyramid is a decimated and filtered version of its predecessor. Each successive level is scaled down by a value of two (hence *dyadic*) and filtered according to the filtering scheme in operation for that pyramid. The final level of the hierarchy is an image consisting of a single pixel.

A single image pyramid is calculated for intensity from the intensity channel, two pyramids are created for colour from the red-green and blue-yellow opponency channels, and four orientation pyramids are formed from four filtered versions of the intensity channel. The intensity and colour pyramids are filtered by a *Gaussian filter* and are referred to as *Gaussian pyramids* (see Figure 5.9 for an illustration of a Gaussian filter). These image pyramids are used for simulating center-surround processing: because each level of the pyramid is essentially a blurred version of the previous level, the contents of image locations further down in the hierarchy will correspond to an averaging operation over areas in preceding pyramid levels (see Figure 5.7 for an example of a Gaussian pyramid for an intensity channel). By computing the difference between a location in the input image and its corresponding location further down the pyramid, one can measure the difference between a central area (a pixel) and its surrounding area (the neighbouring pixels). Thus, a convolution by a Difference-Of-Gaussians, or *DoG*, filter imitates the centre-surround receptive field properties of neurons in early visual processing.

In order to calculate the orientation pyramid, orientation selective filters are first applied to the original intensity channel. Neurons that are tuned to certain

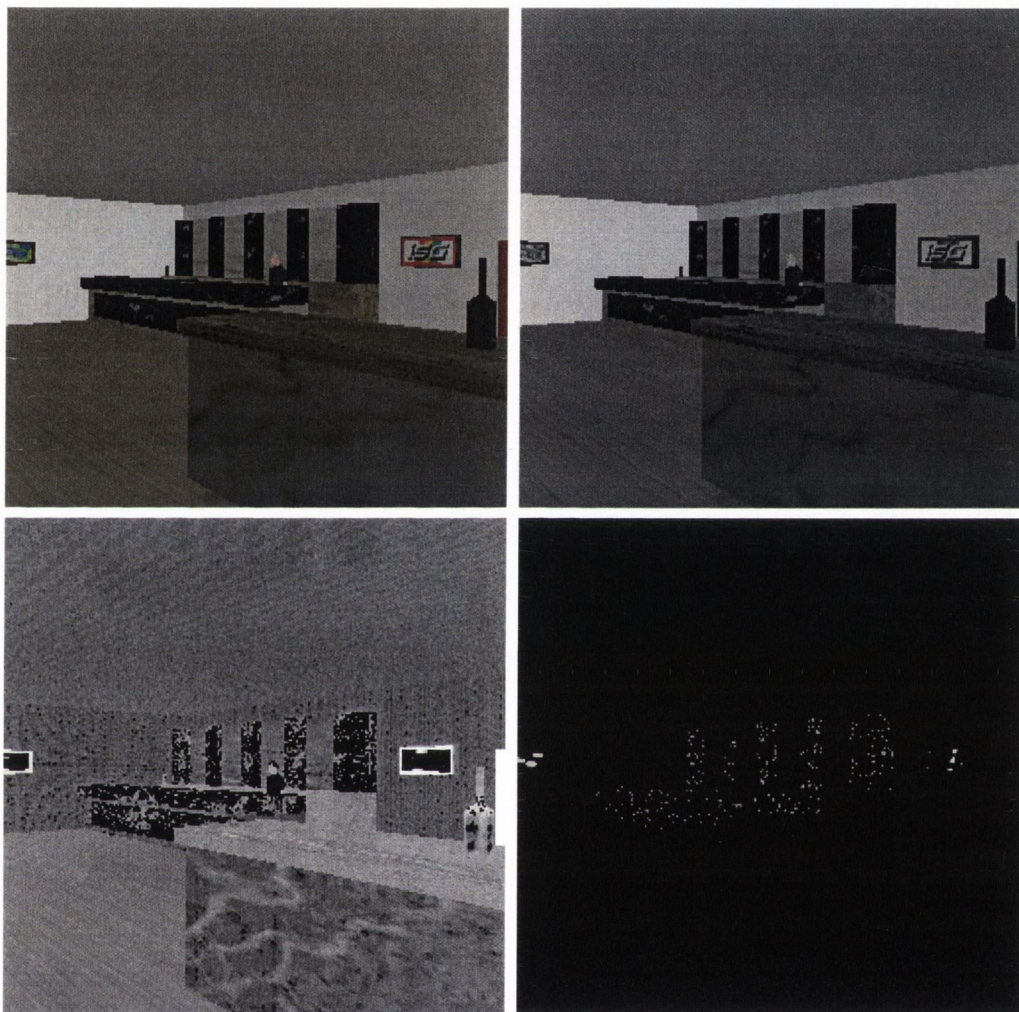


Figure 5.6: Input image and constituent channels. Clockwise, from top-left: input image, intensity channel, red-green opponency channel and blue-yellow opponency channel.



Figure 5.7: Gaussian pyramid for an input image (top left) based on the intensity channel of the image. Each level is successively filtered and decimated (from right towards left).

orientations may be computed by convolution with *Gabor wavelets*, which resemble biological response functions [IK01a]. *Gabor filters* are the product of a cosine grating and a 2D Gaussian envelope (Figure 5.9). In a similar manner to Gaussian pyramids, a *Gabor pyramid* may be constructed from an input image. Unlike Gaussian pyramids, however, Gabor pyramids are tuned to certain orientations; for example, a Gabor pyramid tuned to 0° will respond to horizontal lines in an image. Gabor filters process the image through orientations of angles 0° , 45° , 90° and 135° (Figure 5.8). Although more than four orientations may be used for filtering purposes, Itti notes that using more oriented filters does not affect the model in a significant way [Itt00]. The result is four intensity channels tuned to their respective orientations, where each of these channels forms the root level in a Gabor pyramid.

Feature Map Computation

Once the Gaussian and Gabor pyramids have been constructed, a number of feature maps are computed from the filtered images. One feature type accounts for on-off intensity contrast, two types account for red-green and blue-yellow opponency and four feature types account for orientation contrast. Orientation contrast requires

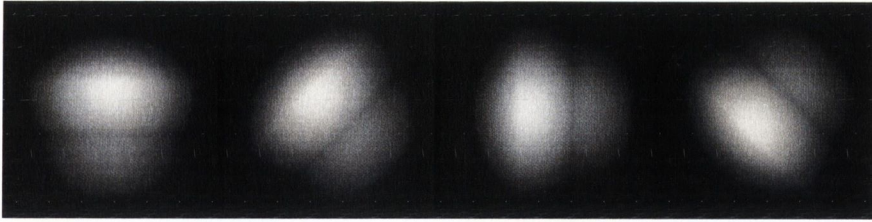


Figure 5.8: Illustration of Gabor filters for the selection of orientations 0° , 45° , 90° and 135° .

four feature types since four different orientations are handled. These feature maps represent the *centre-surround* differences (see Section 3.2.2) calculated from the different levels, or scales, in the pyramids. Levels from the pyramids are chosen at scales $F(c,s)$, where F is the feature map, c is the centre, fine scale in the image pyramid and s is a surrounding coarser scale from the image pyramid:

$$c \in \{2, 3, 4\} \quad (5.1)$$

$$s = c + \delta, \delta \in \{3, 4\} \quad (5.2)$$

$F(c,s)$ is calculated as the absolute result of the difference between the two image scales. This means that six feature maps are computed from each image pyramid, providing multi-scale feature extraction. Essentially, this process calculates the absolute difference between the feature intensity value at the center scale and feature intensity value at the surround scale. In all, a total of 42 feature maps are computed: 6 for intensity, 12 for colour, split into 6 for red-green and 6 for blue-yellow opponency, and 24 for orientation, 6 each for orientations of 0° , 45° , 90° and 180° . These feature maps are then combined to provide a measure of the total pop-out for each feature dimension.

Conspicuity Maps

Once the feature maps have been computed from each pyramid, they are combined to produce master maps for the respective feature dimension (see Figure 5.14 for an example). These master maps are referred to as *conspicuity maps* and encode the saliency of the image in terms of a single feature dimension, for example, intensity. At the end of this process, three conspicuity maps have been produced, one each

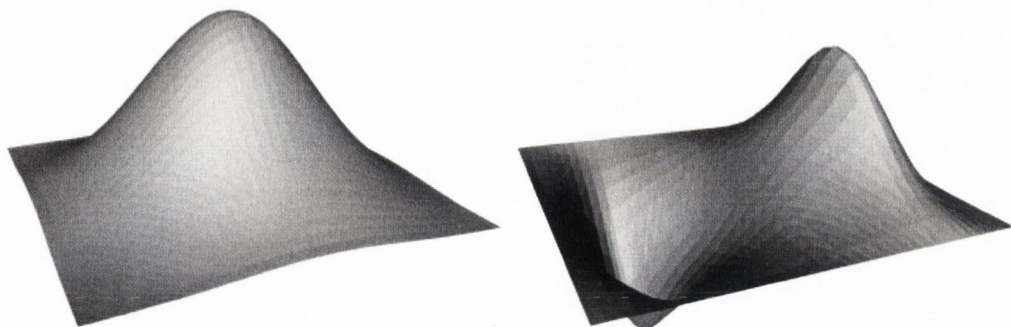


Figure 5.9: Gaussian and Gabor filter kernels, as used in the Gaussian and Gabor pyramids for center-surround processing.

for intensity, colour and orientation. By further combining the three resultant conspicuity maps, a unified measurement of pronounced parts of the entire scene, the *saliency map*, is produced. Figures 5.10 and 5.11 illustrate conspicuity maps and the resulting saliency map for sample input images using the attention model.

Feature Combination

One of the most important factors to be considered when creating the saliency map is how multiple maps should be combined. First of all, feature maps need to be combined in some way into conspicuity maps. Secondly, the conspicuity maps must be combined into the single saliency map. The main difficulty inherent in combining feature maps is that the maps may arise from differing visual modalities. As Itti points out, “how should a 10 degree orientation discontinuity compare to a 5% intensity contrast?” [Itt00]. Itti presents four different feature combination strategies: naive summation, learning linear combinations, contents-based global non-linear amplification and iterative localised interactions (see Itti and Koch [IK01b] for a comparison of these different schemes).

Global non-linear amplification is of particular interest to our research, since it is a simplified, computationally efficient approximation of an iterative localised interactions combination scheme. This method is automatic, requiring no top-down supervision, and works by favouring feature maps with a small number of high activity peaks over feature maps containing larger numbers of responses (see Figures

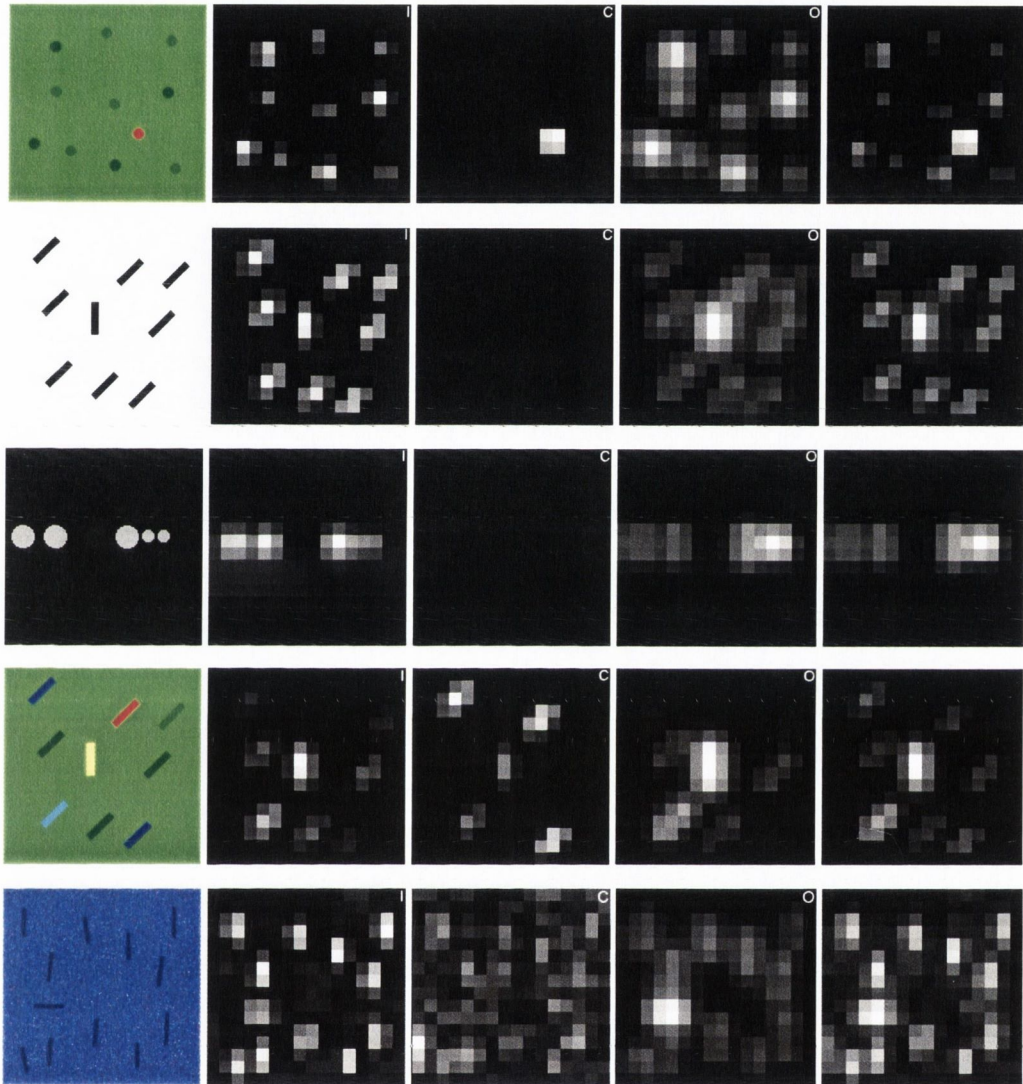


Figure 5.10: Results of test runs for sample images from our implementation of the Itti attention model [Itt00]. Depicts (from left to right) the input scene, the intensity conspicuity map, the colour conspicuity map, the orientation conspicuity map and the resulting saliency map. These examples demonstrate pop-out in standard and noisy conditions.

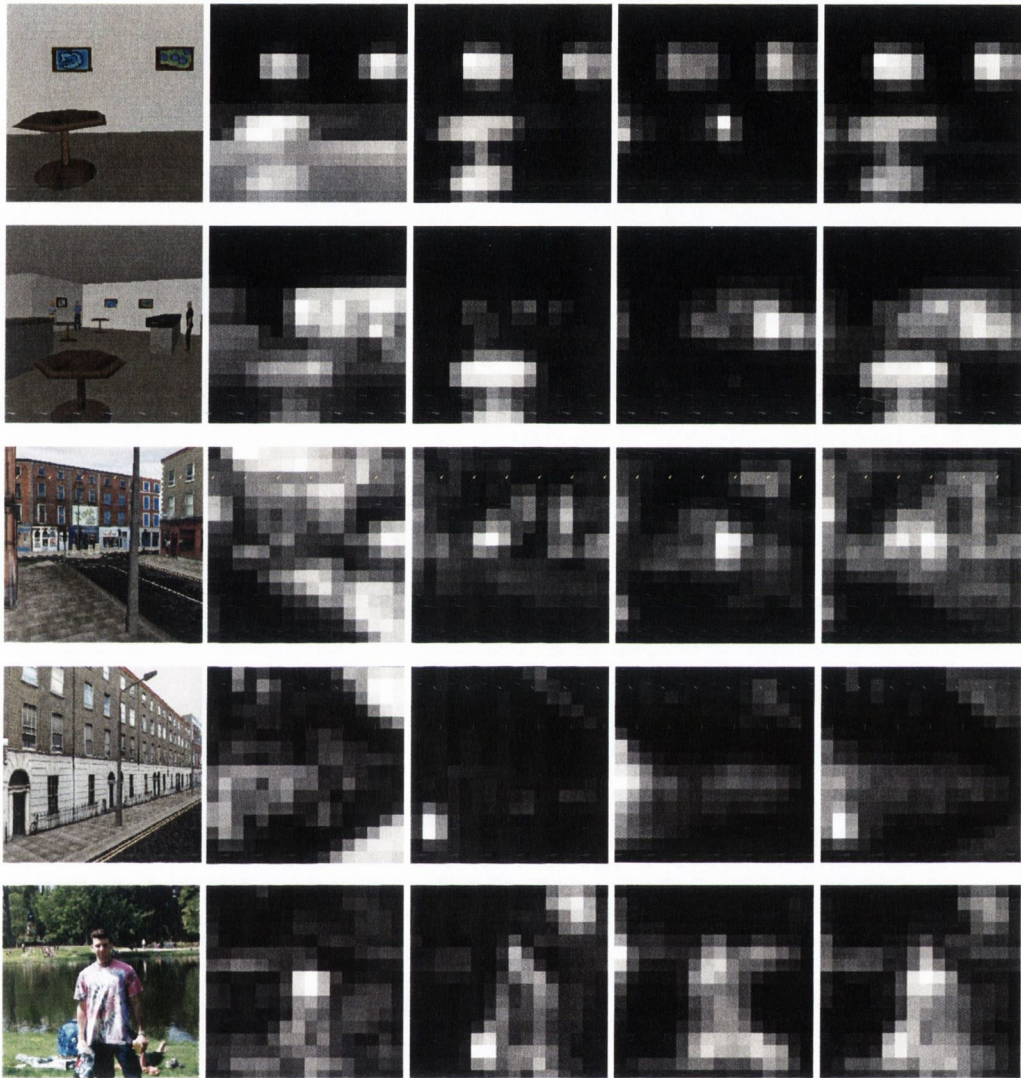


Figure 5.11: Test runs of the attention model on rendered images from a virtual environment and a photograph. Depicts (from left to right) the input scene, the intensity conspicuity map, the colour conspicuity map, the orientation conspicuity map and the resulting saliency map.

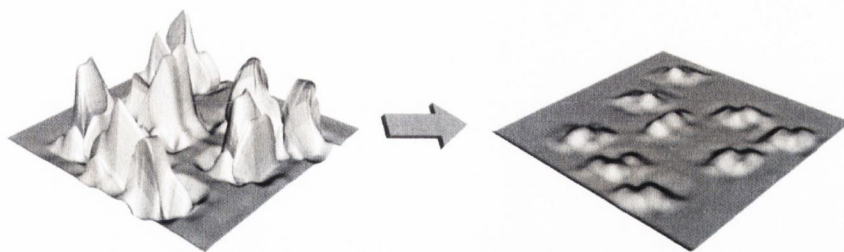


Figure 5.12: Illustration of the normalisation process for a feature map (left) containing multiple peaks of activity and (right) the normalised map after suppression.

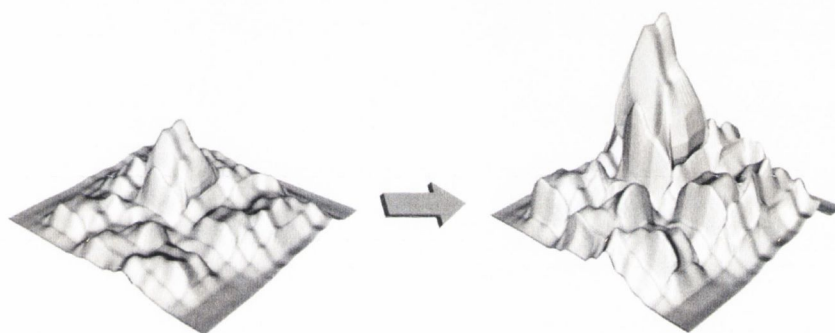


Figure 5.13: Illustration of the normalisation process for a feature map (left) containing a single strong peak of activity and (right) the normalised map after promotion.

5.12 and 5.13 for examples). It coarsely replicates the biological mechanism of *localised excitation* and *lateral inhibition* (see Section 3.2.2), although it is not strictly biologically plausible, since global computations are used for speed reasons.

5.3.2 Inhibition of Return

The result of the calculations described in the previous sections is the saliency map, the primary output of the attention model. The region of maximal value in this saliency map is the stimulus in the scene with maximum *pop-out* and corresponds to the area of the scene to which attention will be paid. However, in reality, the focus of attention tends to be time-varying: attention needs to be shifted from one location in a scene to another, otherwise a single location in the scene will continuously monopolise the focus of attention. The resultant behaviour would correspond to attention being constantly paid to the same location in a scene and no other stimuli would be able to capture attention. Thus, the selection of the focus of attention must be time-varying: if a static model is used then the same location will be repeatedly

chosen as the maximum. Because of this, IOR, or *inhibition-of-return*, is introduced into the model. Locations that have been attended to previously are inhibited so that new locations can win the competition for saliency.

Winner-take-all Network

A neural winner-take-all (*WTA*) network is placed at the output of the saliency map and provides inhibitory feedback based on previous winning locations. The winner-take-all network is a two-dimensional layer of neurons where a strong global inhibition may be activated by any neuron in the layer. When a location in the saliency map is selected as a winner, the corresponding location in the winner-take-all network acts as an inhibitory centre. The centre location and its neighbours become inhibited and the focus of attention may therefore switch to other salient locations. Areas do not remain inhibited indefinitely; the focus of attention is free to return to them after a short period of time. This process has been demonstrated for covert attentional shifts in humans [Itt00] and is used to generate scanpaths within the image.

5.4 Implementation

Our attention system uses our own version of the bottom-up model proposed in [Itt00] and handles colour, intensity and orientation feature contrast. Although the implementation source code has been made available on the internet, the full implementation consists of a large body of code, with only a portion related to the bottom-up attention model. Since our model uses only a subset of the full project provided, implementing our own version was a natural choice for establishing a good understanding of the underlying principles, reducing code complexity and maintaining code coherency. Furthermore, our implementation provides a number of additions to the original model in order to handle moving stimuli in a fast manner, integrate memory and also to ensure the correct operation of the inhibition of return mechanism in an environment where the camera may be moving.

Our implementation uses a 256x256 full scene rendering from the synthetic sensing system (see Figure 4.4) as the primary input to the bottom-up attention model. This rendering is separated into a number of 256x256 image channels as detailed in Section 5.3.1: one for intensity, one for red-green colour opponency and one for

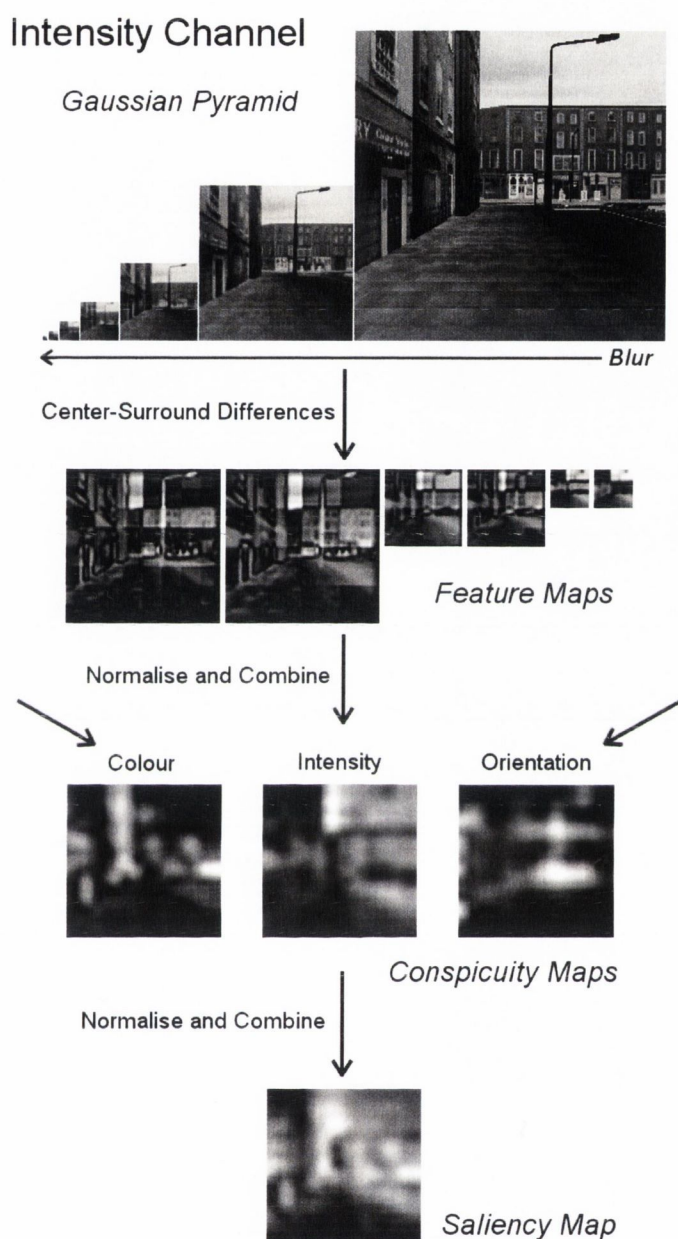


Figure 5.14: Schematic of the attention model. An input image is split into consecutive channels, forming a number of dyadic image pyramids (the image pyramid for intensity is depicted here). Feature maps are created by performing across-scale differencing between different levels in the pyramids. Conspicuity maps are then created from the feature maps using a combination method. Conspicuity maps are further combined to form the final output of the model: the saliency map.

blue-yellow colour opponency. The intensity channel is used as the first level in the intensity Gaussian pyramid and four orientation Gabor pyramids. The red-green channel acts as the first level in a red-green Gaussian pyramid and the blue-yellow channel as the first level in a blue-yellow Gaussian pyramid. Since the pyramids are dyadic in nature, each pyramid therefore contains nine levels; one level for each of the dimensions from 256x256 for the root level image, down to 1x1 for the image of smallest scale.

Images from the feature pyramids are differenced and scaled by 4; that is, 16x16 pixels. Feature maps are then normalised and combined separately for each feature dimension using the global non-linear amplification operator. The result is three 16x16 pixel conspicuity maps, one for each of the primary feature dimensions. These maps are further normalised and combined to form the final saliency map, which also has dimensions of 16x16.

It should be noted that the original input image corresponding to the virtual human's retinal view was 256x256, whereas the saliency map generated by the attention system is 16x16. Because of this, the attention model serves to indicate locations of interest within the agent's view-frustum at a resolution of 16x16, or roughly in 6 degree blocks, given the 100 degree field-of-view of the agent.

5.4.1 Dynamic Scenes

The original model provided by Itti has been shown to work well for static scenes (see [YPG01][HMYS01] for implementations and [MD02] for an evaluation with respect to human scanpaths). In such scenes, the camera is not rotating or translating, although objects in the environment are not necessarily static. Indeed, the original attention architecture has also been updated by Itti in order to handle important temporal events in the environment that may also elicit attention in a bottom-up manner, such as flicker and motion.

When dealing with a purely scene-based technique, as is the case with the attention module presented here, the detection of temporally variant phenomena involves the comparison of multiple frames of the scene, an approach taken in the work described above by Itti. However, for the purposes of animating characters in a virtual environment, such an approach may be over-complicated and costly, especially when a scene database is available. For example, when calculating flicker in a scene-based manner, flicker contrast is calculated using six feature maps derived from a flicker

pyramid based on the change in luminance between the current frame and the previous frame of the scene [IDP03]. However, rather than processing the scene over multiple frames and adding to computational overhead with extra image pyramids, object state from the scene database may be interrogated for temporally significant visual events, an approach taken for top-down attention in [CK99] and here for the purposes of bottom-up attention.

Motion

The motion contribution of an object is based on both its velocity vector and the size of the object. The object's current and projected positions are queried from the database and are perspective projected into the two dimensional view-space of the virtual human. A 2D velocity vector is then constructed based on the two resulting positions. The magnitude of this velocity vector provides an estimate of the amount that the object is moving from the point of view of the virtual human. This magnitude is then weighted by the size of the object's proximal stimulus representation from short-term sensory storage (see Table 4.2 or Figure 4.6). This models a system where attention will tend to be elicited by large or near, fast moving objects, as opposed to small or distant, slow moving objects. Here, fast moving is defined in terms of stimulus movement on the virtual human's retina as opposed to the object's world-space velocity.

The process described above takes place for every moving object in the scene. Once a list of object motion contributions has been created, it is then normalised between 0 and 1. A motion map is then created in such a manner that all pixels corresponding to the object in the false-colour image are given values corresponding to their motion contribution (Figure 5.15). This map is then normalised according to the *Global Non-linear Amplification* normalisation procedure (see Section 5.3.1).

5.4.2 Object-Based Attention

Unfortunately, a number of problems are encountered when the camera must change its position or orientation. The main problem is that IOR in the original attention model is view-based; if a location becomes inhibited in the scene and the view position or orientation changes, then the original IOR view-space location no longer corresponds to the same location within the environment and becomes invalid. This does not cause a problem with static scenes since view-space always corresponds to

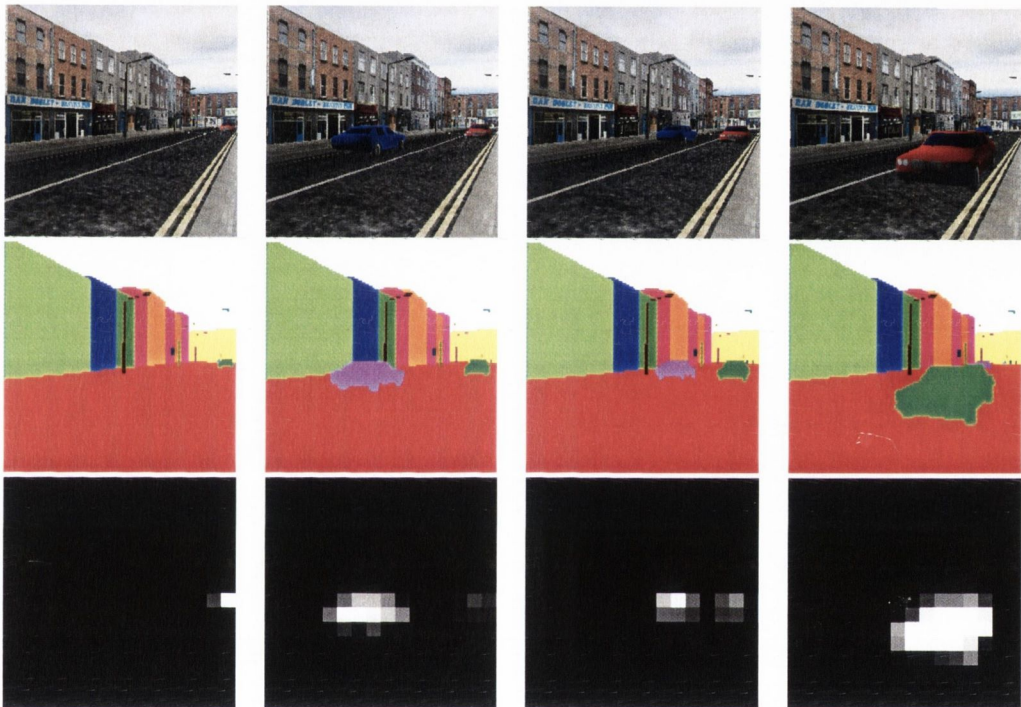


Figure 5.15: Four frames of animation from a city scene containing two moving vehicles. In this case, the red vehicle is moving towards the virtual human while the blue vehicle is moving away from the virtual human. The top row illustrates the view from the perspective of the virtual human, the middle row is the false-colouring peripheral rendering and the bottom row is the normalised motion map.

locations within the environment from one frame to the next. However, when successive frames are not the same, synchrony between view-space and environmental locations is lost. Essentially, inhibition-of-return should take into account locations *within the environment*, rather than locations *within view-space*.

This is of particular significance when we seek to apply the attention model to virtual humans; looking and moving around the environment are natural human behaviours and a system that couldn't handle them would be restrictive. One proposed method of handling the problem of a moving viewer in the attention model is to completely disable inhibition-of-return. The basis of this approach is that when the viewer is moving, the algorithm will be presented with a completely new image each time and that, as pointed out by Itti, there is little evidence for inhibition-of-return across eye movements in humans or monkeys.

In contrast to this approach, Marmitt and Duchowski [MD02] have found that the correlation between human and artificial scanpaths generated by the Itti model is lower than expected for dynamic scenes. They point out that the problem may lie in the algorithm's lack of memory; when a human has already seen many parts of a scene, they are free to distribute visual attention to new areas. The conjecture is that the attentional model should be augmented with a memory of previously seen image regions or inspected virtual objects. Regardless of what approach is taken, some form of memory of parts of the environment that have been attended to is desirable, whether the purpose is to inhibit those locations from being looked at or to note objects of interest that should be looked at again. In order to solve the problem of attentional memory in dynamic scenes, we use an object-based memory system.

There is evidence from psychology that attention can be deployed in both a scene-based manner and an object-based manner [Dun84]. One commonly cited instance providing evidence for object-based attention is Balient syndrome. A component of this syndrome, called simultanagnosia, is the inability to perceive more than one object at a time, despite normal visual processing. This neglect appeared to be unconnected to particular spatial locations: patients often perceived only one of two separately coloured overlapping triangles (see [Sch01] for an overview).

In contrast to the location-based IOR model discussed earlier, our implementation resolves winning locations in the saliency map to the object level so that the object-based memory system (Section 4.3) can be used to remember what objects

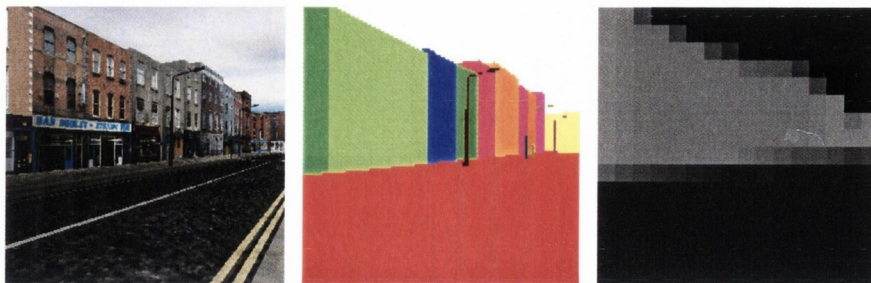


Figure 5.16: The memory uncertainty map is constructed from the false-colour rendering and applied to the attention output to modulate the saliency map. From left to right, fully rendered scene, false-coloured retinal rendering and memory uncertainty map. In this example, the sky and ground have low uncertainty levels in contrast with the building objects.

have been attended to in the scene. By remembering what objects have been attended to, rather than the locations in view-space, the problem of a moving viewer problem becomes easier to account for. Recall that the 256×256 retinal rendering is passed to the attention model for processing and that a corresponding false-colour peripheral rendering is used to encode object identification information. Because the saliency map is of a lower resolution than the retinal rendering input into the attention model, a single saliency map location will correspond to more than one pixel from the retinal rendering and the peripheral rendering. This means that a single winning saliency map location may contain more than one object. That is, multiple objects may be the focus of attention at any one time.

Memory Uncertainty Maps

We use the object-based memory system to modulate the visual salience of objects based on the number of times that they have been brought into the focus of attention. Every object in memory has an uncertainty level, ranging from 0 to 1, where a level of 1 denotes that the agent's uncertainty of the object is high (see Section 4.3.1). When an agent attends to an object, the uncertainty level of the object decreases to simulate the agent forming an internal representation of it. When the uncertainty levels of all objects in the scene are taken together, they provide an estimate of scene uncertainty, which is useful for guiding higher level behaviours, such as exploration.

Once the saliency map has been calculated, an object memory uncertainty map is created at the same resolution of 16×16 pixels (see Figure 5.16). Each position in the

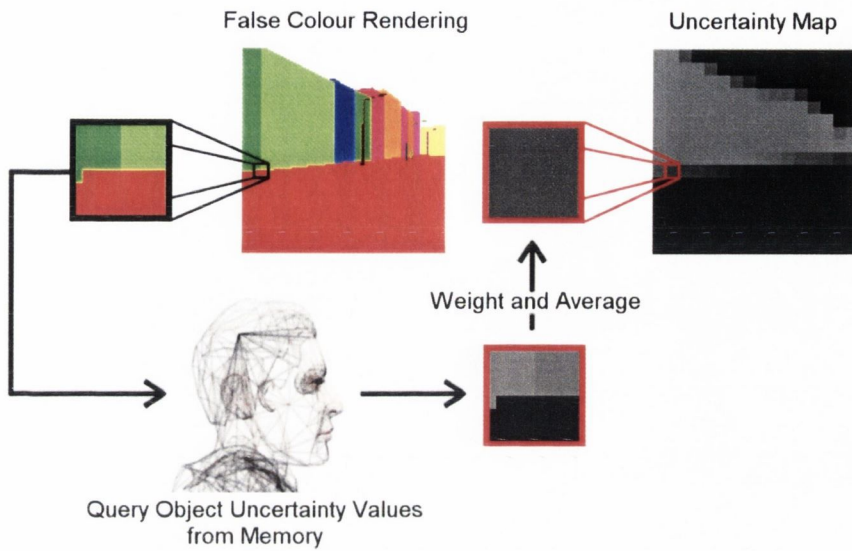


Figure 5.17: The uncertainty map provides a measure of uncertainty for a scene based on an agent’s memory of its surroundings. Since the uncertainty map is at a lower resolution than the false-colour rendering, a location in the uncertainty map is calculated as the average value of the uncertainty levels of each object within that location weighted by the number of pixels that the object occupies.

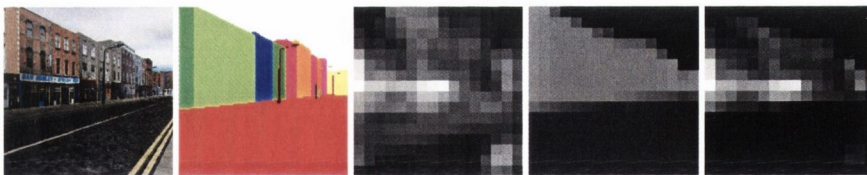


Figure 5.18: The modulated saliency map (far right) is obtained by point-by-point multiplication of the saliency map and the memory uncertainty map. From left to right: fully rendered scene, false-coloured retinal rendering, unmodulated saliency map, memory uncertainty map and modulated saliency map, referred to as the attention map.

object uncertainty map is a measure of the uncertainty for all objects falling within that spatial region. It is calculated as the average value of the sum of the weighted object uncertainty levels within the spatial location. Each object uncertainty level is weighted by the number of pixels that the object actually occupies within that spatial location (see Figure 5.17 and Equation 5.3).

$$\frac{\sum_{n=1}^{numObjects} \left(uncertainty^n * \frac{screenspace^n}{256} \right)}{numObjects} \quad (5.3)$$

The result is a two dimensional map defining spatial locations containing values denoting how much the contained objects have been attended to before. This object uncertainty map is then combined with the saliency map using point-by-point multiplication to produce an *attention map* (see Figure 5.18). The attention map represents the combined saliency and uncertainty in a scene. This attention map may then be parsed for local maxima (that is, positions of interest) using, for example, the winner-take-all network described in Section 5.3.2.

5.5 Discussion

We have adopted a bottom-up model of visual attention from the field of cognitive engineering for the purposes of locating areas of interest based on *pop-out* in rendered images. This model is based on the colour, intensity, orientation and velocity features in a scene. The use of attention in our system is dual-role. Not only does it link into the gaze controller to provide eye and head movements, but it also forms an integral part of the multi-stage memory system, deciding what visual scene information should be passed from a virtual human's short-term sensory storage to short-term memory. Thus a feedback loop is created between attention and memory.

There are a number of reasons for using the bottom-up attention model presented by Itti. It replicates early visual processing in a biologically plausible manner and it has been shown that the algorithms that form a basis for the architecture can indeed predict eye fixations over static images [PS00b]. The model itself has been shown to be effective with both natural and graphically rendered scenes [Itt00] [YPG01] [HMYS01] and has also been specifically evaluated for dynamic virtual reality settings by Marmitt and Duchowski [MD02], who concluded that it could be improved by the use of memory, something that is presented in this work.

Although the attention module presented here only considers a subset of possible attentive behaviours, that is, behaviours stemming from bottom-up attention, it improves on current work in this area. Chopra-Khullar's excellent work presents a three tier architecture, where the tiers represent deliberate, task-related behaviours, peripheral motion events and spontaneous looking respectively [CK99]. Task-related attention competes with peripheral events in a probabilistic manner. Spontaneous looking has the lowest precedence, only occurring if there are no tasks at hand and no peripheral motion events. It will also be interrupted by higher precedent events when they occur. The type of attention implemented here corresponds to Chopra-Khullar's peripheral motion events and spontaneous looking behaviours. Although computationally efficient, Chopra-Khullar's model only considers colour contrast in RGB space for spontaneous looking and does not take orientation or luminance into account. Furthermore, spontaneous looking only takes place if there are no tasks or peripheral motion events pending. Such a scheme inherently prioritises the motion dimension over all other basic feature dimensions, regardless of the magnitude or contrast in the motion. This may not always be correct; for example, a single bright location may attract attention over: a small or distant moving object, a moving object that has been looked at before, a moving object that is partially occluded, or a large number of moving objects (even if such objects constitute distractors). The feature map methodology used in our model provides a homogeneous representation for all feature dimensions, including motion. Because of this, the same feature combination technique is applied across all feature maps, providing a useful, unified scheme for prioritising between colour, intensity, orientation and motion targets.

Finally, although Chopra-Khullar's model implements uncertainty levels for object monitoring behaviours, it does not consider object-based memory for spontaneous looking behaviours or peripheral motion events. Some form of memory is important in ensuring that a single location does not consistently dominate the agent's attention.

In a similar manner to the methods presented in this paper, Courty et al. [CMA03] use a saliency map approach based on synthetic vision to guide the actions of an autonomous actor. Unlike the attention model presented in this chapter, the saliency map consists of two different feature maps, based on orientation and depth, in order to provide quick determination of regions of interest in the input image. Courty et al. note that their system could be improved by the use of a visual

memory system, something that is presented in this work.

The system provided by Gillies does not implement any form of bottom-up attention mechanism, and operates in a solely task-oriented, endogenous manner on objects in the environment [Gil01]. Such a system would appear to be computationally efficient; however, a scene-based attention scheme may be desirable over a purely object-based system since it allows rendering effects, such as lighting and textures, to contribute in an implicit manner to the calculation of where agents attend to. The reason for this is that a scene-based method takes into account the environment from the perspective of the agent; proximal stimuli on the agent's retina are considered, rather than just the objects as they exist in the environment, the distal stimuli. Because of this, scene-based perception forms a more accurate approximation of the agent's perception of the environment. Usually lighting effects, such as spotlights and flares, are not represented in the scene database as objects, yet they are still of perceptual importance to viewers and should therefore contribute to the attention process. In the same way, textures may be highly significant. This is particularly true for virtual environments containing objects that may be large; buildings in a city environment, for example. In order to improve rendering speeds and storage constraints, visual elements such as doors and windows are often not explicitly represented as geometry, or may only be represented as geometry according to a perceptual metric, such as distance. Rather, these elements are represented by textures: therefore, they will be missed by a purely object-based attention scheme. A scene-based scheme is also desirable because it is not dependent on the representation of the objects in the scene database. It is flexible in that any scene description may be used, such as geometry or billboards, without modification of the attention component. Previous approaches either do not consider scene-based attention at all, operating in a strictly object-based manner [Gil01], or only consider a coarse colour contrast computation for a single type of attentive behaviour [CK99].

Although the original attention model presented by Itti acts in a scene-based manner, selecting locations of interest in the environment, we use the synthetic vision module to obtain information about the corresponding objects in the agent's field-of-view. This permits modulation of the results from the attention model by object-based information from the scene database, thus providing an attention system that is both scene-based and object-based. This is more plausible from a human standpoint: although a scene may elicit attention, humans usually conduct further

processing in order to establish constructs, or objects, that allow more compact processing and storage (see *Feature Integration Theory* in Section 3.2). For example, our attention may be attracted to a part of the environment containing a red car on the basis of its colour, edges and other features; however, we don't usually store a red area and an edge list in memory, but rather remember a construct that we refer to as a 'red car'. In terms of virtual human animation, the concept of objects is a necessity, since it allows for the storage of environmental information in a way that is semantically significant and related to the scene database. The storage of coordinates from the environment is not useful since it only denotes where the agent was attending to, but not what the agent was attending to. Furthermore, in a dynamic environment, a scene-based memory system such as that included in the Itti model (inhibition of return) is not useful, since inhibited locations will be invalidated as soon as the agent's viewpoint changes. Also, the motion of objects will not be handled in a proper fashion, since the previous locations of any moving object will be inhibited rather than its current location. This has been catered for in the Itti model by disabling IOR for dynamic scenes. Nonetheless, some form of memory of what the agent has looked at before is desirable in behavioural animation. An object-based scheme also has the advantage of allowing the scene database to be queried for information on the internal state of objects. Access to state information, such as position, orientation and velocity, ensures that no scene information is lost during the synthetic vision process. This sort of information also allows for quicker processing methods to be devised for the agent, since environmental information does not have to be guessed or reconstructed after it reaches the agent.

Algorithm 3 Create a saliency map from an input image.

Input : Full-scene Rendering FS

Output : Saliency Map SM

CREATESALIENCYMAP(*Full-scene Rendering* FS)

 Split channels(FS)

 {split the full-scene rendering into three channels}

$RGColourPyr \leftarrow$ GaussianPyramid($RGchannel$)

$BYColourPyr \leftarrow$ GaussianPyramid($BYchannel$)

$IntensityPyr \leftarrow$ GaussianPyramid($Intensitychannel$)

$OrientationPyr \leftarrow$ GaborPyramids($Intensitychannel$)

 {calculate image pyramids}

$RGColourFMaps \leftarrow$ CenterSurround($RGColourPyr$)

$BYColourFMaps \leftarrow$ CenterSurround($BYColourPyr$)

$IntensityFMaps \leftarrow$ CenterSurround($IntensityPyr$)

$OrientationFMaps \leftarrow$ CenterSurround($OrientationPyr$)

 {calculate feature maps}

$RGColourCMap \leftarrow$ Combine($RGColourFMaps$)

$BYColourCMap \leftarrow$ Combine($BYColourFMaps$)

$ColourCMap \leftarrow$ Combine($RGColourCMap, BYColourCMap$)

$IntensityCMap \leftarrow$ Combine($IntensityFMaps$)

$OrientationCMap \leftarrow$ Combine($OrientationFMaps$)

 {calculate conspicuity maps}

$SM \leftarrow$ Combine($ColourCMap, IntensityCMap, OrientationCMap$)

Chapter 6

Attentive Behaviours

6.1 Introduction

All of the proposed components thus far, such as synthetic vision and memory, have been internal to the virtual human. Yet, in terms of providing virtual human animation, these systems mean little if they do not provide some form of outward manifestation to viewers. Perhaps the most important external manifestations of underlying cognitive processes are looking and gaze behaviours. Contemporary agents lacking in such behaviours appear to be absent from their environment in some way, although many seem to be able to sense the scene as if by telepathy. Ideally, viewers would like to be given the impression that they are interacting with their environment in the same way as living, thinking creatures. Although as humans we cannot directly see the thoughts of others, we do obtain an impression of thought-processes at work through the way in which other people behave. We can often infer what people are thinking by observing where they look as they perform their everyday activities. In this way, if the agents in virtual environments do not appear to pay attention to anything in the scene, then our sense of presence may be severely degraded. As noted by Blumberg et al.: “If we provide characters with the means to express their mental state, an observer can infer their beliefs and desires” [Blu97]. Resultant motions, such as gaze motions, derived from underlying cognitive simulations, including attention and memory, are important factors in achieving this goal.

In this chapter, we present a component for automatically generating appropriate looking behaviours for a virtual human given a target location in the environment.

This component is concerned with the animation of the virtual human and is comprised of a gaze module for generating eye and head motions, and a blinking module for generating appropriate blinking motions. In addition, we also discuss how the internal models may be integrated with gaze generators in order to produce a range of high-level behaviours.

6.2 Gaze

As pointed out by Poggi et al. [PPdR00], eyes have at least four different functions: seeing, looking, thinking and communicating. Seeing means that we use our eyes for storing incoming information from the environment through visual perception. Even though we may see things in our environment, we do not necessarily pay attention to them. Looking is the orienting of the direction of one's gaze with the goal of obtaining visual information about points of interest. Another function of the eyes is thinking. We may raise our eyes up or close them when concentrating. Finally, we may also communicate using our eyes with the sole purpose of giving information to somebody.

In this work, the primary concern is how the eyes act in the roles of seeing and looking. When looking around, targets may appear at different eccentricities within our environment; if these eccentricities are outside of the mechanical rotational limits of our eyes, the *oculomotor range* or *OMR*, then the recruitment of the head in the final motion is mandatory in order to align the visual axis with the target. Such *gaze shifts* consist of coordinated eye and head movements in order to bring the foveal region onto such a target.

In terms of animating plausible virtual humans, gaze is not only important in allowing us to shift our attention to different areas of our environment, but it is also a behaviour that is salient to others. Real humans expect intelligent creatures to make gaze motions from time to time. Such motions seem to suggest that a creature is aware of its surroundings and paying attention to them, thus signalling the existence of underlying cognitive processes.

Gaze, G , is regarded as being the direction of the eye in space, which in turn is the direction of the eye in the head, E , and the direction of the head in space, H . Thus,

$$G_{WS} = H_{WS} + E_{HS} \quad (6.1)$$

describes gaze direction where WS is the world-space coordinate frame and HS is the head-space coordinate frame.

6.2.1 Eye and Head Movements

Since both the eye and the head contribute to gaze motions, modelling gaze raises the question of how much the eye and the head actually contribute to the total gaze motion. It has been found that in species with a small OMR, cats for example, the eye and head trajectories are tightly coupled and stereotyped [Gui92].

In contrast, for humans and Rhesus monkeys who have larger oculomotor ranges, the contribution of the eyes and head to the final motion can show a great deal of variability. Afanador and Aitsebaono [AA82] found that half of their subjects consistently moved their heads, even when targets were well within the OMR and head movements were not mandatory. The other half only produced head movements for eccentricities beyond 20° to 30° . Subjects were subdivided into two distinct groups: those showing head movements at or below 10° and those presenting head-movements only up to 20° . Bard et al. [BFP91] therefore classified humans as either *head-movers* or *non-movers* in making gaze-shifts, noting two contrasting strategies for eye-head coordination that were apparent in the normal population.

Fuller [Ful92] further detailed a number of effects that may be used to explain these behaviours. As noted, horizontal human head movements are generally mandatory when the gaze shift demands an ocular orbital eccentricity exceeding the OMR, roughly $\pm 40^\circ$ in humans. If a gaze saccade can be executed within this orbital threshold, then the amplitude of the head movement is regarded as being *discretionary* and the extent to which the head moves is called *head movement propensity*. Individuals are thought to have an innate behavioural trait which determines their propensity to move their head, something that is thought to be linked to an individual's method of constructing central nervous coordinate systems. Fuller details three effects based on observations of trials involving head-movers and non-movers, where *gain* is the relative contribution of the head to the total gaze motion.

1. Midline attraction refers to a resistance to head movement away from the midline (see Figure 6.1) in non-movers and an increase in the head movement

amplitude if a jump in gaze starts eccentrically. When the head is off the midline and gaze is stepped to the opposite side, the head will be moved ipsiversively (to the opposite side), unlike when the jump is initiated with the head on the midline.

2. Resetting occurs when the eccentricity of a jump is varied, resulting in the stopping position being reset closer to the target. The pooled mean head contribution for the jump from midline to a 40° target was 0.51 (stopping position of 19.6° from the target), while it was 0.73 (stopping position of 10.8° from the target) for a -20° to $+20^\circ$ jump. Fuller notes that this can be thought of as a momentum effect that brings the head closer to the target when the head and the target are on opposite sides of the midline.
3. An *arousal* effect increases the gain in the tests where the head has been aligned. This may be due to instructions given to participants that increase their awareness of their own gaze motions.

It was also noted that starting and stopping positions can dramatically affect the extent to which the head takes part in a gaze shift. Head movement amplitude is also affected by the position of the eyes in the head at the time of the gaze step.

Given the task of rotating the head and eyes onto a target position such that the view direction is directed towards that target, one of the main questions that arises is how much the head and eyes should contribute to the final motion. Experiments in psychology have indicated a relationship between the contribution of the head to the motion, referred to as *gain*, and the position of the target with respect to the person's viewpoint. However, as noted above, the gain tends to vary from individual to individual. We present a gaze controller that animates the gaze motions of different virtual humans in different ways based on head movement propensity in order to provide diversity between the motions of characters.

The Gaze Controller

A *gaze controller* is used to provide a model of gaze behaviour for a virtual human. Target positions in the environment are passed to the gaze controller, which is responsible for creating an animation of the agent's eyes and head towards that target position.

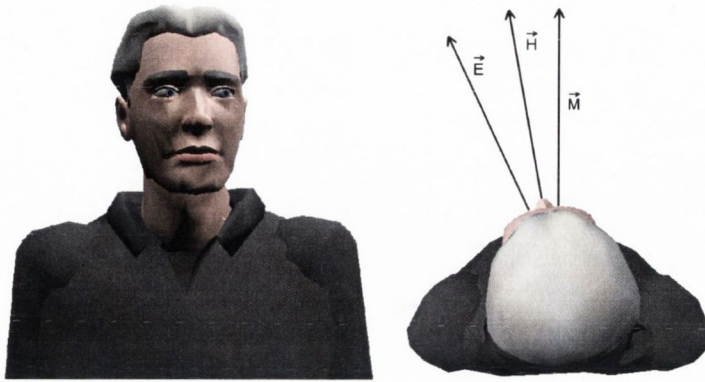


Figure 6.1: Virtual human configuration (left) and the corresponding top-down view (right) showing vectors that are used for determining gaze. \vec{M} is the midline vector, \vec{H} is the head vector and \vec{E} is the eye vector. The object of the gaze controller is to align the eye vector onto a target position with head contributions that vary between different virtual humans.

At the lowest level, the task of the gaze controller is to animate the skeleton of the virtual human so that the final configuration is such that the eyes are oriented towards the target position. The virtual human's skeleton contains a large number of bones and in the real world situation, many of these could contribute towards the motion itself. The gaze controller presented here is simplified in that it only affects a subset of the virtual human's bones, primarily the head and eyes of the character. Since the virtual human's vision is monocular, a further bone is defined as part of the skeleton that specifies a camera position approximating the viewpoint of the virtual human. The orientation of this viewpoint is locked to the orientation of the eyes of the actor (see Figure 4.5). Thus, when the eyes rotate, so too does the viewpoint.

Virtual humans are tagged with a *head movement* factor, HMF , ranging from 0 to 1, signifying their propensity towards either head movement or eye movement (see Table 6.1 for a list of definitions that will be used to describe gaze computations). That is, virtual humans range between being *head-movers* and *non-movers*. This attribute is used by the gaze controller in providing a simplified model of midline attraction and resetting effects as described by Fuller. Essentially, this attribute is used to determine the gain, or head contribution, to gaze motions made by the virtual human; if the gain is high, then gaze motions will consist primarily of head movements, while a low gain will result in mainly eye movements.

Attribute	Meaning
\vec{M}	Midline Vector
\vec{HM}	Extreme head-mover vector
\vec{NM}	Extreme non-mover vector
\vec{H}_c	Current head vector
\vec{H}_f	Final head vector
\vec{E}_c	Current eye vector
\vec{E}_f	Final eye vector
\vec{T}	Target vector from head position to target position
OMR	Oculomotor Range between -40° and $+40^\circ$ horizontally
HMR	Head Movement Range
HMF	Head Movement Factor ranging in value from 0.0 to 1.0

Table 6.1: Definitions to describe gaze computations used throughout this chapter.

The gaze algorithm operates by creating a final head vector \vec{H}_f and final eye vector \vec{E}_f (see Algorithm 4 for an overview of the full algorithm). Initially, two head vectors are created with respect to the target. The first vector, \vec{HM} , represents the most extreme head movement vector. That is, gain will be maximum when orientating with the target and the eyes will therefore not make any contribution to the gaze unless the target is eccentric beyond that afforded by head movement limits alone. In such cases, the head movement is maximised and eye movement composes the remainder of the orientation (see Figure 6.2). A second vector, \vec{NM} , is then constructed that represents extreme non-movers. If the target is within oculomotor range, OMR , and eye movement is thus discretionary, then extreme non-movers will always use an eye movement to conduct the gaze [AA82]. If the target is outside of oculomotor range, then \vec{NM} will be aligned such that the eye vector will be at its maximum, that is, the range of OMR . Note that any targets that are outside of $HMR + OMR$ can not be looked at by the current algorithm, since eye and head contributions alone are not enough to generate alignment with the target (see Figure 6.3).

For cases where the current head vector \vec{H} is on the opposite side of \vec{M} with respect to \vec{T} , midline resetting is simulated for non-movers by moving \vec{NM} closer to \vec{HM} (and thus, closer to \vec{T}) (see Figure 6.4).

The final head vector \vec{H}_f is then calculated as an interpolation of HMF between \vec{NM} and \vec{HM} (see Section 6.2.1). The eye vector \vec{E}_f will then be the vector from

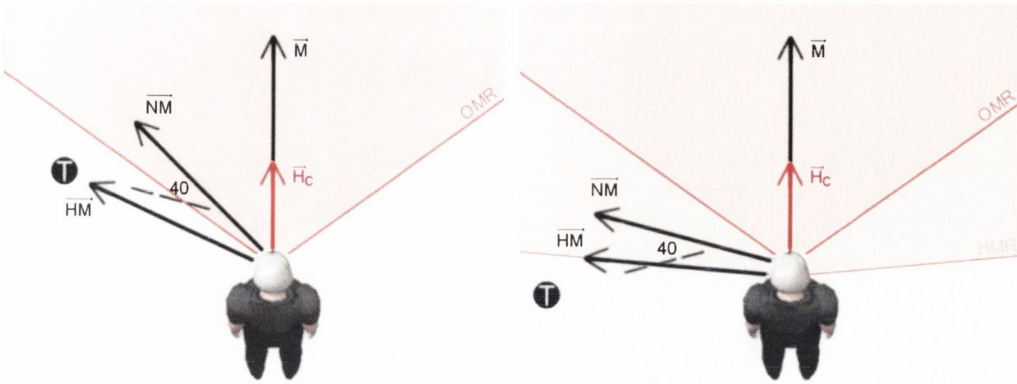


Figure 6.2: Calculation of extreme movement vectors when targets are outside of OMR and outside of HMR . (Left) When a target is outside of OMR , head movements are mandatory, even for non-head movers. (Right) \overrightarrow{HM} can not extend beyond the maximum head rotation range HMR . Therefore, when the target is outside HMR , \overrightarrow{HM} is set to HMR and eye movements are mandatory. In both cases, the extreme non-mover vector \overrightarrow{NM} remains as close as possible to the midline vector \overrightarrow{M} .

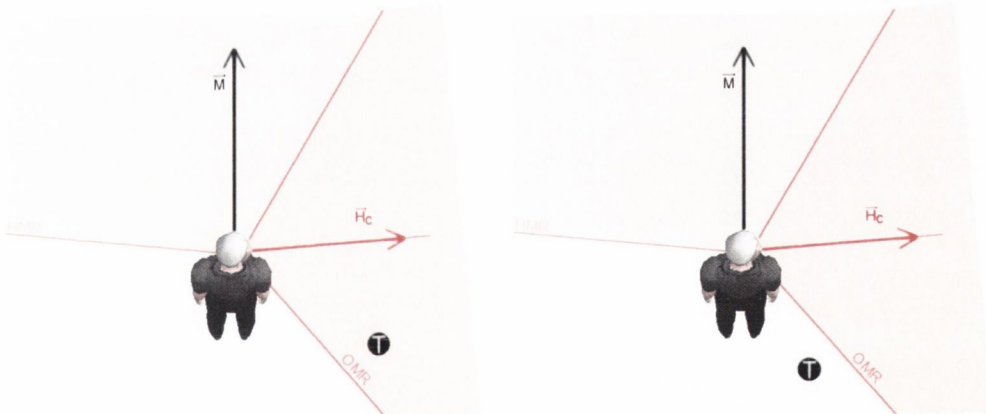


Figure 6.3: Two cases where the target is outside of the HMR . (Left) The target is outside of the HMR , but is inside $HMR + OMR$ and can therefore still be looked at. (Right) The target is outside of $HMR + OMR$ and can not be looked at by head and eye contributions alone.

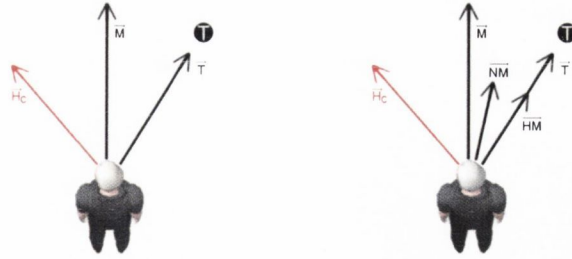


Figure 6.4: The resetting effect. (Left) The target vector \vec{T} is on the opposite side of the midline \vec{M} to the current head vector \vec{H}_C . (Right) The extreme non-mover vector \vec{NM} is brought closer to the target vector \vec{T} in what can be thought of as the effects of momentum.

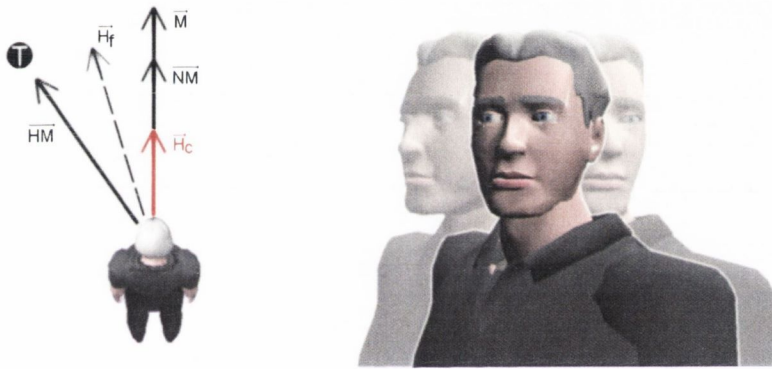


Figure 6.5: The final head vector \vec{H}_f is calculated as an interpolation between \vec{NM} and \vec{HM} . In this example, the interpolating factor, corresponding to HMF , is 0.5.

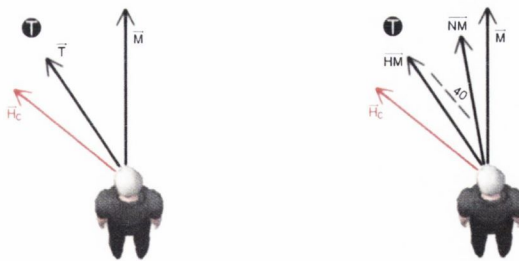


Figure 6.6: Illustration of midline attraction. (Left) The target vector is between the midline and current head vectors. (Right) The extreme non-mover vector \vec{NM} is attracted towards the midline such that the eyes are at the limits of OMR .

the eye position to the target (and will always be between 0 and the oculomotor range OMR) (see Figure 6.5).

When the target vector \vec{T} is within OMR , non-movers will tend to use mainly their eyes to fixate the target. However, non-movers are more likely to maintain their head position close to the midline due to midline attraction. This means that in certain cases where the target vector \vec{T} is between the current head vector \vec{H}_c and the midline \vec{M} will result in the head being attracted back towards the midline (see Figure 6.6).

Animating Gaze

Provided with current and desired head and view orientations, a gaze motion consists of interpolations of these orientations over a number of frames until the desired end orientations have been achieved. In the gaze controller, orientations are encoded as *Quaternions* (see [Sho85]). Interpolation between gaze positions is achieved through the use of a Quaternion SLERP, or *spherical linear interpolation*, operation. Provided with a starting orientation Q_{so} , an end orientation Q_{oe} , and a time factor t , the SLERP operation will provide an interpolation of factor t between Q_{so} and Q_{oe} using Bezier curves. The SLERP operation provides a quaternion that represents the shortest path on a unit sphere (or *great arc*) from one rotation to the other.

The SLERP operation is also used to interpolate between extreme-head mover vectors \vec{HM} and extreme non-mover vectors \vec{NM} in order to produce an intermediate orientation based on HMF . Since HMF has a range of 0.0 to 1.0, passing it to the SLERP operation as the t parameter will produce an orientation somewhere between \vec{NM} and \vec{HM} . A HMF of 0.0 will correspond to the start orientation in the SLERP, and a HMF of 1.0 will correspond to the final orientation in the SLERP. Intermediate HMF values will therefore correspond to intermediate orientations between the start and final orientation, that is, \vec{NM} and \vec{HM} respectively. A high HMF value will therefore produce large head contributions to the final gaze animation as required.

6.3 Blinking

Blinking motions, although subtle, are an important and often overlooked way of conveying a sense of realism in gaze motions. Blinks may be interpreted in many

Algorithm 4 Gaze motion towards a target in the environment.

Input : Current head vector \vec{H}_c Target vector \vec{T} Head movement factor HMF
Output : Final head vector \vec{H}_f

 GAZE(Current head vector \vec{H}_c , Target vector \vec{T} , Head movement factor HMF)

```

if  $\vec{T}$  within  $HMR$  then
  if  $\vec{T}$  is within  $OMR$  then
    if  $\vec{T}$  is between  $\vec{M}$  and  $\vec{H}_c$  then
      {Midline attraction brings head back towards midline for non-movers}
       $\vec{NM} \leftarrow$  head as close to midline as possible while  $\vec{T}$  is within  $OMR$ 
    else
      {Eye movement will suffice:  $\vec{NM}$  not changed}
    end if
  else
    {Target is outside oculomotor range}
     $\vec{NM} \leftarrow$  head as close to midline as possible while  $\vec{T}$  is within  $OMR$ 
  end if
   $\vec{HM} \leftarrow \vec{T}$ 
  if  $\vec{T}$  was on opposite side of  $\vec{M}$  w.r.t.  $\vec{H}_c$  then
    { $\vec{NM}$  is moved  $re\%$  closer to target due to resetting effect}
     $\vec{NM} \leftarrow$  SLERP ( $\vec{NM}, \vec{T}, re$ )
  end if
else
  {target is outside of head movement range}
  if  $\vec{T}$  is within  $OMR$  then
    if  $\vec{T}$  is between  $\vec{M}$  and  $\vec{H}_c$  then
      {Midline attraction brings head back towards midline for non-movers}
       $\vec{NM} \leftarrow$  head as close to midline as possible while  $\vec{T}$  is within  $OMR$ 
    else
      {Eye movement will suffice:  $\vec{NM}$  not changed}
    end if
  else
     $\vec{NM} \leftarrow$  head as close to midline as possible while  $\vec{T}$  is within  $OMR$ 
  end if
  {Target outside of head range, so head moves to maximum head range possible}
   $\vec{HM} \leftarrow HMR$ 
  if  $\vec{T}$  was on opposite side of  $\vec{M}$  w.r.t.  $\vec{H}_c$  then
    { $\vec{NM}$  is moved  $re\%$  closer to  $\vec{HM}$  due to resetting effect}
     $\vec{NM} \leftarrow$  SLERP ( $\vec{NM}, \vec{HM}, re$ )
  end if
end if
 $\vec{H}_f \leftarrow$  SLERP( $\vec{NM}, \vec{HM}, HMF$ )

```

different ways. Individuals who don't blink or move their eyes appear to be pre-occupied, while high rates of blinking often indicate unfocused or rapidly altering internal states such as frustration or emotional excitement. Professional animators and artists have long recognised this and blinking is treated as an important ingredient for characters [Moo01]. Despite this, in-depth research into *automated* multi-modal blinking for autonomous virtual humans based on scientific literature appears to have received little attention.

Blinking has been considered when applying eye movements to conversational agents. Most work incorporating blinking in conversational agents has been based on an agent's internal emotional parameters. For example, in the A.C.E. system, an agent's anxiety level has an effect on the frequency of eye blinking [KMCT00]. Unlike previous work, we focus here on synthesising blinking that is invoked in a direct way by changes in the agent's gaze. Such blinking, referred to as *gaze-evoked blinking*, is now considered.

6.3.1 Gaze-Evoked Blinking

Gaze-evoked blinking motions are those that accompany saccadic eye and head movements, in contrast to normal blinks that are elicited by external stimulation such as wind. Many vertebrates generate gaze-evoked blinks as a component of saccadic gaze shifts. As noted in Evinger et al. [EMP⁺94], such blinks would appear to serve the purpose of protecting the eye during the movement, while also providing lubrication to the cornea while vision is impaired due to the saccade.

Gaze-evoked blinks, as described in a study by Evinger et al., have three main characteristics:

1. The probability of a blink increases with the size of the gaze shift. In the study conducted by Evinger et al., blinks were found to occur with 97% of saccadic gaze shifts larger than 33°, while the occurrence of eyelid activity for 17° eccentricity was only 67%.
2. The initiation of the activation of the eye muscle is dependent on the magnitude of the head or eye movement. Activation of the eye muscle started after the initiation of the head movement, but there was a clear tendency for activity to start before the head movement with large amplitude gaze shifts.

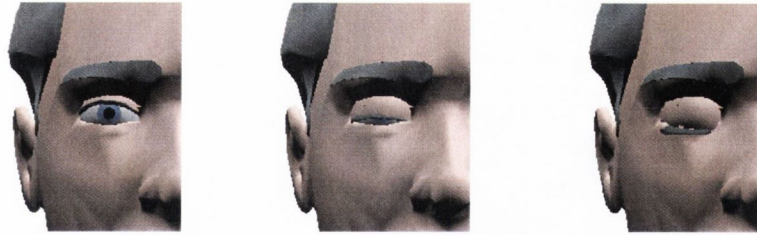


Figure 6.7: Illustration of blinking magnitudes for the virtual human. Magnitudes from left to right are 0%, 50% and 100%.

3. The amplitude of the head movement has been shown to affect the magnitude of the blink, or how much the eyelid closes (see Figure 6.7). In general, the more that the head moves, the more the eyelid tends to close. For example, during a head movement of 5° eccentricity, the blink magnitude will usually be quite small, meaning that the eye-lids will not close very far. In contrast, eccentricities of 50° or more will often result in a full closure of the eye-lid.

The Blink Controller

The general characteristics of gaze-evoked blinking detailed in Evinger et al. are implemented in our *blink controller*. Ordinarily, approximately twenty blinks occur every minute [Kar88]. However, the blink controller also takes input from the gaze controller. When a gaze request is made, relevant information from the gaze controller, such as the starting time of the gaze and the gaze amplitude, is passed to the blink controller which then plans and schedules the blink motion for the virtual human's eyelids.

The blink motion is planned as follows. First of all, it is determined, based on head movement amplitude, if a blink occurs at all. This is done probabilistically, where the probability of a gaze-evoked blink occurring is modelled as having a simple linear relationship with respect to the amplitude of the head movement, such that the probability of a blink for targets of 20° eccentricity is 20%, and the probability of a blink for targets of 50° or more eccentricity is 60%. In this way, a gaze evoked blink will always take place for head movement amplitudes of 75° or more.

If a blink event is forecast, then the blink magnitude is calculated. Blink magnitude i.e., how much the eyelids close, is modelled as a linear relationship between the amplitude of the head movement, such that the magnitude of a blink for targets of 17° eccentricity is 67% and the magnitude for targets of 33° eccentricity is 97%.

6.4 High-Level Behaviours

There are a large number of applications for internal models of vision, memory and attention that are not possible without such models. Here, three suggestions are provided as road-maps for the use of such mechanisms in the construction of higher level behaviours that are integrated with the internal models.

Vision-based reaching illustrates how the introduction of internal models that approximate an agent's world can lead to errors. When an agent is not sensing (e.g. looking at) a target, reaching actions towards that target must be made on the basis of memory and therefore become more prone to error. This means that an agent will often make an initial approximate reach for an object; if an error is made, it may need to look at the object in order to correct the reach.

Gaze behaviours can be obtained by integrating the gaze and blinking systems discussed in this chapter with the visual attention model presented previously. There are a number of ways in which gaze targets may be queried from the attention model and these are important for affecting the appearance of the final animation.

Once the integration has taken place, curiosity and explorative behaviour can be simulated for agents by considering what is in the agents memory, i.e. what it has and hasn't seen before and to what extent. Such a system may be useful for simulating the difference between agents that are familiar with their environment and don't look around a lot and virtual tourists who have arrived at a new destination and are eager to explore the new environment by looking around a lot and creating internal representations of the new environment.

6.4.1 Vision-based Reaching

An example of a higher-level behaviour that may be constructed from the sensing and memory components is *visual reaching*. Visual reaching distinguishes between prehension motions made towards those objects that are actually visible to the agent and objects that are not visible to the agent, in which case reaching must be made towards memorised object locations. Neurophysiological results have shown that the computations performed by humans vary under these two conditions [SF89].

The term *sensorimotor transformation* refers to the process by which sensory stimuli are converted into motor commands [PS00a]. Reaching towards a visual stimulus with the hand is one example of a sensorimotor transformation. It has



Figure 6.8: Three frames of a reaching animation based on vision and memory. In this case, the target, marked in red, is not within the senses of the virtual human. In this case, memorised information about the target is used along with an approximation of the sensorimotor transformation to define an initial hand position (right picture).

been suggested that there is a reasonably accurate internal representation of a target's location in a body-centred frame of reference (see [SF89] for references). The coordinates of this representation are referred to as extrinsic coordinates. There also appears to be an internal representation for the current orientation of the upper arm and the forearm. These are referred to as intrinsic coordinates and allow the calculation of the hand position.

Under normal, visually guided circumstances, the mathematical relationships between intrinsic and extrinsic coordinates were found to be highly non-linear [SLT86]. However, research shows that the relations between intrinsic and extrinsic coordinates are close to linear when subjects point at virtual target locations that are based on visual memory. It was hypothesised that these movements would be less accurate than those made towards visible objects because of the use of a linear approximation to the more accurate, but non-linear, sensorimotor transformation.

The shoulder coordinate frame is centered on the shoulder. The x-axis of the frame is along the line connecting both shoulders, the y-axis goes outward from the chest, and the z-axis points upward toward the head. Arm posture is encoded in the shoulder frame by four parameters: Θ is the upper arm elevation, β is the forearm elevation, η is the upper arm yaw, and α is the forearm yaw. The extrinsic coordinates are defined in terms of spherical coordinates with the origin at the

Algorithm 5 Approximate reaching towards a target.

Input : Target Object *TObject*, Short-term sensory storage list *STSS*,
Long-term memory list *LTM*

Output : Reach animation towards the target object

APPROXIMATE REACH(*Target Object TObject*, *Short-term sensory storage list STSS*, *Long-term memory list LTM*)

if *TObject* exists in *STSS* **then**

 {the object is visible}

 visual position \leftarrow *STSS*[*TObject*][position]

 Accurate reach(visual position)

else

 {use memorised object information from *LTM*}

 remembered position \leftarrow *LTM*[*TObject*][position]

 Inaccurate reach(remembered position)

 Look at(remembered position)

 Search(*TObject*)

 {the object will now enter the *STSS*}

 visual position \leftarrow *STSS*[*TObject*][position]

 Accurate reach(*TObject*[position])

end if

shoulder. These coordinates consist of the target azimuth χ , the target elevation Ψ , and the radial target distance R . Given these definitions, Soechting et al. found that the following relations between intrinsic and extrinsic object coordinates describe a pointing movement towards a *remembered* target location:

$$\begin{aligned}
 \Theta &= -4.0 + 1.10R + 0.90\Psi \\
 \beta &= 39.4 + 0.54R - 1.06\Psi \\
 \eta &= 13.2 + 0.86\chi + 0.11\Psi \\
 \alpha &= -10.0 + 1.08\chi - 0.35\Psi
 \end{aligned}
 \tag{6.2}$$

whereas the equations for visually guided reaches are thus defined:

$$\begin{aligned}
 \Theta &= -6.7 + 1.09R + 1.10\Psi \\
 \beta &= 47.6 + 0.33R - 0.95\Psi \\
 \eta &= 67.7 + 0.68R + 1.00\chi \\
 \alpha &= -11.5 + 1.27\chi - 0.54\Psi
 \end{aligned}
 \tag{6.3}$$

These equations, from [SLT86], approximate a highly non-linear function. In our system, when a target has been visually acquired by the agent, a reach is calculated correctly towards that target. A correct reach approximation is first obtained using Equation 6.3. This acts as the first guess for a simple inverse kinematics algorithm that positions the arm correctly on the object. When a target is not in the view of the agent, the remembered location of the target is used with Equation 6.2, which is the inaccurate reach approximation. A reach animation is conducted towards this incorrect location first and then a corrective movement is made by passing the actual target coordinates to the cyclic coordinate descent inverse kinematics module (see [Wel93] for a description). This inverse kinematics technique was used due to its simplicity, numerical stability and computational efficiency. An overview of an approximate reaching behaviour that may be based on sensing and memory is provided in Algorithm 5.

Although the reaching animation as implemented here is simplistic and reaching for a remembered item is a behaviour that probably is not frequent or exceptionally noticeable in daily life, it does illustrate an important point in relation to internal representations of environments based on synthetic vision and memory; internal representations are only approximate and therefore the actions of virtual humans using such models are prone to error. Such errors may be fruitful in providing enhanced plausibility, since humans also make mistakes, and also provide more diversity in the generation of animations.

6.4.2 Integration with Attention

The attention module presented in Chapter 5 inputs an image and provides an estimate of areas that are likely to be looked at. When integrated with the vision and memory modules, a scene can be sensed and memorised in an object-based manner and targets of attention can be generated over time. We now consider how gaze motions and behaviour can be integrated with the attention module, since there are a number of possible ways that the attention module could be used to generate such behaviour.

When generating behaviours, a number of issues need to be considered. These include consideration of when the synthetic vision module should be updated, when new targets of attention may be processed and to what extent the attention module should interact with the gaze generator presented in Section 6.1.

The first issue involves synchronising visual sensing and attention updates with head and eye movements. For example, it would not be appropriate for the visual system to update while the virtual human's eye lids were closed. As mentioned in Section 3.4.1, eye movements consist of fast movements called *saccades* and periods where the eye stays relatively still in order to comprehend the target, called *fixations* (see Figure 6.10). The constraints that we place on a fixation is that the eyes and head must be still and the eyelids must be open. That is, a gaze animation must not be in progress and the virtual human must not be in the process of blinking.

Generating Fixation Targets

Fixation targets are obtained from the uncertainty map generated by the visual attention system (see Section 5.4.2). At any time, a character's fixation target will correspond to the maximal value of the uncertainty map, which is a combination



Figure 6.9: Human fixations and scanpaths from a scene in a virtual environment captured using eye tracking equipment. The raw eye data (top image) is processed into saccades, denoted by yellow lines, and fixations, denoted by red circles (bottom image). It is the task of the attention system to provide analogous data to the gaze generation module based on the visual input from the scene.

of the saliency and memory maps (see Section 5.2.1 for description of the saliency map). Since attending to an object reduces its uncertainty level in memory, a virtual human's interest in an object will decrease over time as it pays attention to it. If at any time a new location in the uncertainty map becomes maximal, then a change of fixation is regarded as having occurred and the virtual human must perform a gaze shift to look at the scene location corresponding to the new maximal value in the map.

Algorithm 6 Conduct a fixation towards a new region of interest.

Input : False-coloured rendering FR , full-scene rendering FS

Output : Fixation Vector FV

```

FIXATENEWTARGET(Image FR, Image FS)
   $PS \leftarrow \text{extractProximalStimuli}(FR, FS)$ 
   $Memory.EncodeInSTSS(PS)$ 
   $Memory[STM] \leftarrow \text{applyAttention}(Memory[STSS])$ 
   $UM \leftarrow \text{createUncertaintyMap}(Memory)$ 
   $SM \leftarrow \text{createSaliencyMap}(FS)$ 
   $AM \leftarrow SM.Modulate(UM)$ 
  {create the attention map}
   $Winner \leftarrow \text{getWinningLocation}(AM)$ 
   $RICoords \leftarrow \text{transform}(Winner)$ 
   $FV \leftarrow \text{transform}(RICoords)$ 

```

It should be noted that this process replaces the Winner-Take-All network in Itti's original model, which consists of a neural network (see Section 5.3.2). When a winning location emerges from the Winner-Take-All network, neighbouring locations within a certain radius of the winning location are inhibited, or suppressed. Because of this, neighbouring regions are unlikely to be selected as subsequent winners and fixation targets, meaning that locations further to the periphery will be selected as regions of interest. This would seem to match experimental observations: Privitera and Stark [PS00b] use clustering as an eccentricity weighting procedure as they had found that human subjects focused attention on significant eccentric loci.

The typical operation of the system will consist of switches between targets of interest in the environment, where different locations will be looked at for different periods of time based on their relative interest level relative to other locations. This operation is analogous to that of human scanpaths (see Section 3.4.1) where the eye

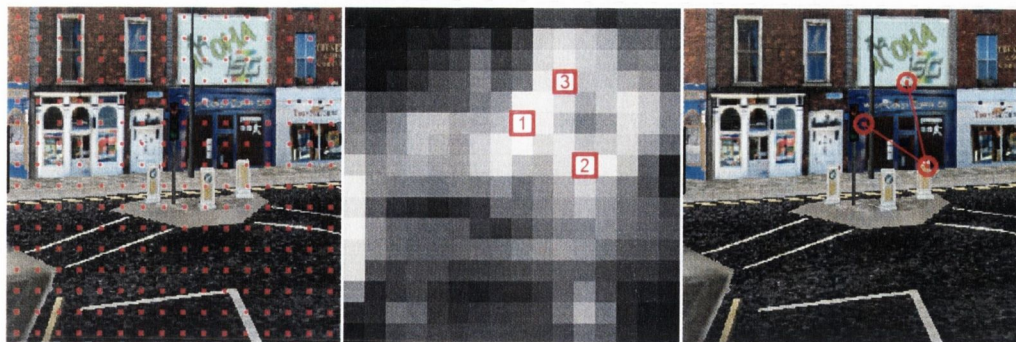


Figure 6.10: Illustration of artificial fixation generation for a sample image. The saliency map, which has a resolution of 16×16 , is scaled up to the dimensions of the input image (left). Red dots mark potential artificial fixation positions. Provided with an uncertainty map for the image, three maxima are chosen as fixation targets in ascending order (center) to produce a scanpath over the original scene (right). The uncertainty map can be queried for as many fixation targets as required: in this case only three were chosen.

fixates on regions of interest and dwells on them for variable periods of time before conducting gaze saccades to new locations.

Since the output of the attention system is a two dimensional greyscale map of 16×16 resolution across the character's view-field, in order to obtain a fixation within the character's view-field, the winning position in the uncertainty map is transformed into retinal image coordinates ranging from 0 to 255 along the x and y axes. Given these eye-space coordinates as well as the camera frustum parameters corresponding to the virtual human's eye, a retinal image coordinate can be transformed into a three dimensional eye-space vector and from there to a world-space vector. Such vectors form the input to the gaze generator and are useful for constructing orienting gaze motions during changes of fixation.

Synchronisation of Gaze and Attention

The uncertainty map provides a basis for obtaining a large number of possible fixations. Since a single update of the attention module generates an uncertainty map, the attention system does not necessarily have to be updated in order to obtain a number of fixation sequences. This means that the attention system does not have to be updated on a per frame basis in order to provide a number of frames of gaze animations. However, as the amount of time since the last attention update increases,

so too does the error inherent in the uncertainty map. That is, the uncertainty map will correspond to locations that were of interest a number of frames previously. The validity of such an approach depends on how much the character's view of the environment changes: if the environment remains constant and the character does not change position or orientation, then only one update will be required to provide the character with saliency information for generating all gaze and fixation motions.

It should be noted that attention is shifted to the new location before the saccade that moves the gaze takes place. Therefore, the movements of the eyes are a result of the attentional selection processes preceding the actual eye-shifts; the eye-shifts should not be considered as the selection process.

6.4.3 Curiosity And Exploratory Gaze Behaviour

There are many reasons why creatures pay attention to their environment and many ways in which they can do so. Curiosity is the motivation to discover new knowledge when faced with an unfamiliar situation [Ber60] [Ber71]. It enables man, and indeed many animals, to gain information about their environment for the purposes of survival and would appear to be a driving force behind many types of exploratory behaviour. In this way, exploration is used to describe an observable behaviour, while curiosity is regarded as being the underlying construct or drive that is responsible for the behaviour.

There are a number of distinct types of explorative behaviour, ranging from curiosity about an environment to philosophising and game playing. In this work, a less general view of curiosity is taken, in that it is presumed to be the tendency to explore or investigate a novel environment. Of particular interest here is *inspective behaviour* or *specific curiosity* [Ber71]. This sort of behaviour is aimed at the reduction of uncertainties in the environment, as curiosity is thought to be induced by a lack of information or uncertainty resulting from exposure to novel or complex stimuli. An animal that finds itself within an unusual or novel environment will often display inspective behaviour in order to analyse it for the purposes of noting possible escape routes and dangers.

We endow virtual humans with curiosity thresholds that are used in conjunction with the attention system to provide differing types of exploratory behaviour (see Figures 6.11 and 6.12). Referring back to the way in which uncertainty values are indicative of the completeness of a character's internal representation of an object

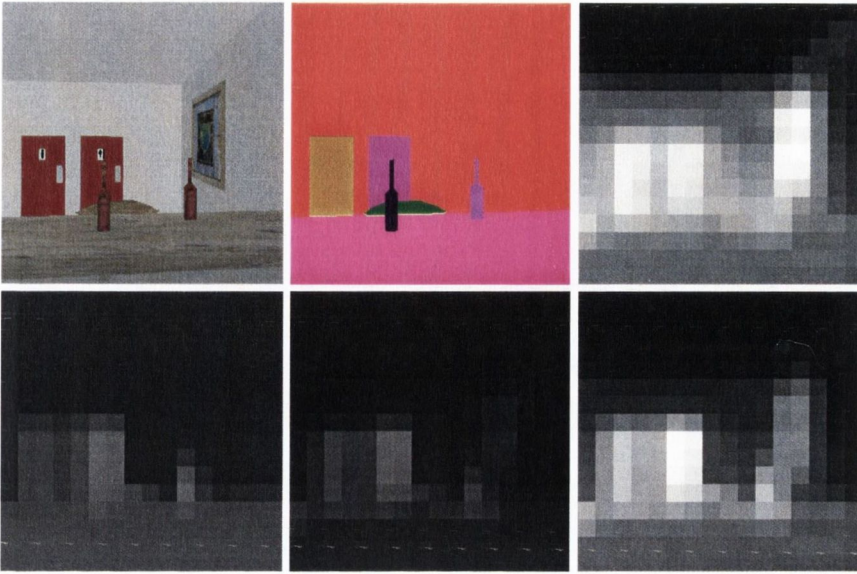


Figure 6.11: Illustration of curiosity calculation for an input scene. From left to right, top to bottom: the input scene, the false-coloured scene rendering, the saliency map, the memory uncertainty map, the unnormalised attention map and the normalised attention map. These images correspond to a scene that has been looked at and memorised already by a virtual human, since all uncertainty levels are relatively low. In such a case, a curious character may continue to inspect the scene when other characters would not.

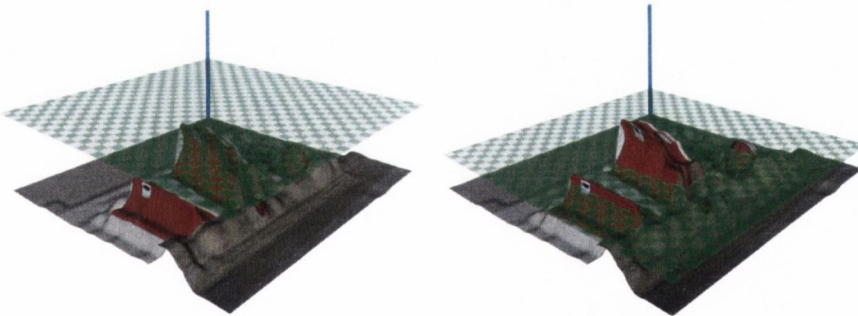


Figure 6.12: Illustration of two different curiosity thresholds (green checkerboard planes) for the unnormalised attention map depicted in Figure 6.11, where scene uncertainty levels are used as height values. Despite the fact that the attention maps are the same, since the threshold in the left image is higher, no scene elements attract attention. In contrast, the threshold in the right image means that some scene elements still elicit attention, even though they have already been looked at and have low uncertainty levels. Low threshold values correspond to a high curiosity weighting for that character.

(see Section 4.3.1), curiosity thresholds model the way in which some characters will be content with sparser internal object representations as oppose to others who will continually inspect an object in an attempt to find out new details about it. We may regard the latter characters as having the trait of curiosity. In practice, the curiosity threshold is a value between 0 and 1 that corresponds to the minimum saliency level in the unnormalised attention map that will invoke a gaze response (see Section 5.4.2 for a description of the attention map). That is, if the attention level of a stimulus is beneath a character's curiosity threshold, then the character will not find that stimulus appealing for further inspection and will not pay attention to it.

Curiosity thresholds, when combined with memory uncertainty maps, are especially useful for determining when virtual humans should *stop* looking around. According to the saliency map generation procedure detailed in Section 5.3, the output saliency map will always be normalised in order to provide winning locations. Since the attention system deals with feature contrast, it will always provide potential fixation targets for the virtual humans to look at. A nave application of this model results in a virtual human that has continuously high levels of arousal: that is, it continually fixates on objects in the environment without rest as the Winner-Take-All network continually generates them. While such situations do occur in the real world, for example, when we find ourselves in novel or dangerous environments, in many cases when we have a relatively good memory of our surroundings we do not need to inspect objects with such frequency or enthusiasm. By integrating the saliency map and the uncertainty map, a fixed measure of an agent's familiarity with its environment may be gauged. In this way, agents can look around frequently when encountering a new environment and then decrease the frequency of their looking motions as familiarity increases and arousal decreases. Free time without significant gaze evoking stimuli can be spent on task related goals or internal processing (analogous to introspection) and may be interrupted by sudden onsets of new stimuli that were not previously sensed or in memory.

6.5 Discussion

In this chapter we presented a motion generator for creating gaze and blink motions for a virtual human provided with a target location in the environment. We also

demonstrated how higher level behaviours can be based on the internal components presented in Chapters 4 and 5. The synthetic vision module is used to provide vision-based reaching, where errors in reach motions are present when remembered target information is used in contrast to currently sensed target information. Gaze fixations are also modelled by integrating the gaze module with the output of the attention module. By incorporating memory and basing looking frequency on an agent's uncertainty values for objects in its memory, exploratory behaviour may be generated.

Chopra-Khullar [CK99] uses a gaze algorithm based on attention, as does Itti [IDP03]. These approaches define final orientations for the eyes and head based on an input location. Unlike these approaches, our approach is novel in that it provides variation in the contribution of the head and eyes to gaze motions based on internal attributes, in particular, a head-movement attribute based on research from psychology. Provided with two virtual humans in the same state making gaze motions towards the same target, the resultant head and eye contributions will differ based on their particular head movement propensities. This is useful for creating diversity in the final animations, something that is particularly important for scenes involving crowds of virtual humans.

Despite the fact that eye movements are a necessary component of gaze behaviours, we only consider eye movements as they relate to gaze motions, *saccadic gaze shifts*, and do not consider small-scale saccadic eye movement models, for example, the model presented by Lee et al. [LBB02]. In our model, it is presumed that eye movements only take place in the context of saccadic gaze shifts triggered by shifts in attention. The work presented here is by no means mutually exclusive with respect to small-scale eye movement models and would most likely benefit greatly from the addition of such a model.

Although most contemporary animation systems contain a blinking model of some sort, blinking is often random or based on internal emotional models. Unlike previous work, our blink controller links eye-blink to gaze motions with the specific purpose of enhancing the realism of such motions. This type of gaze-evoked blinking could be used in conjunction with other forms of blinking, for example, those based on emotion or attention levels [PPdR00].

Vision-based reaching illustrates how internal systems may be used for controlling higher level motions. The concept of automatically introducing error into

reaching motions to targets based on senses that are not visible has not been the focus of previous work in virtual human animation. While the results may be subtle, they may often add to the plausibility of an animation. One good example of this sort of error-based animation can be seen in the short animation *Geri's Game* by Pixar Animation Studios, where an old man is preoccupied with a chess game and makes erroneous reaches towards a chair; he is only able to grasp the chair accurately when he turns around and acquires it visually. While Pixar's example uses hand-crafted motion and the automated approach here does not reach the same standard of animation, it does highlight the potential of using synthetic sensing to introduce higher-level human error correction behaviours where they could enhance plausibility. For example, such work could be extended by having characters trip over objects on the ground that they have not sensed while they are walking.

The attention system presented here differs from those previously presented in a number of ways. In particular, our system is dual-role. Not only does it link into the gaze controller to provide eye and head movements, but it also forms an integral part of the multi-stage memory system; it decides what visual scene information will be passed through each stage of the virtual human's memory. Thus, a feedback loop is created between attention and memory.

Chapter 7

Evaluation

As has been noted in previous publications that have tackled the problem of attention for directing virtual human gaze animation [CK99] [Gil01], evaluating the naturalness of gaze motions generated from such systems is challenging, since human behaviour is both complex and diverse.

This work has considered the problem of generating plausible gaze motions as being composed into two subproblems: where to look, and how to look. First of all, provided with a virtual environment, what determines the locations of the where the virtual human will look? These locations should make sense and ideally be similar to where a real human would look under the same conditions. The second problem is how the looking, or gaze motion, should take place. Given a target location in the environment and the virtual human's current configuration, how should a gaze motion be generated towards that location? This includes the question of how much the head and eyes should contribute to the movement, as well as blink motions. Keeping these issues in mind, we evaluate our system in terms of speed, the accuracy of targets generated by the attention system and the plausibility of the gaze generation system.

7.1 Results

Although a large amount of this work has focused on the construction of internal models, such as attention, one of the main ways in which the overall operation of these components may be grouped is through the resulting outward behaviours that



Figure 7.1: Two screenshots from the street environment. A number of pedestrians are walking in the street as well as the main virtual character (dressed here in black).



Figure 7.2: Screenshots taken from the street environment.

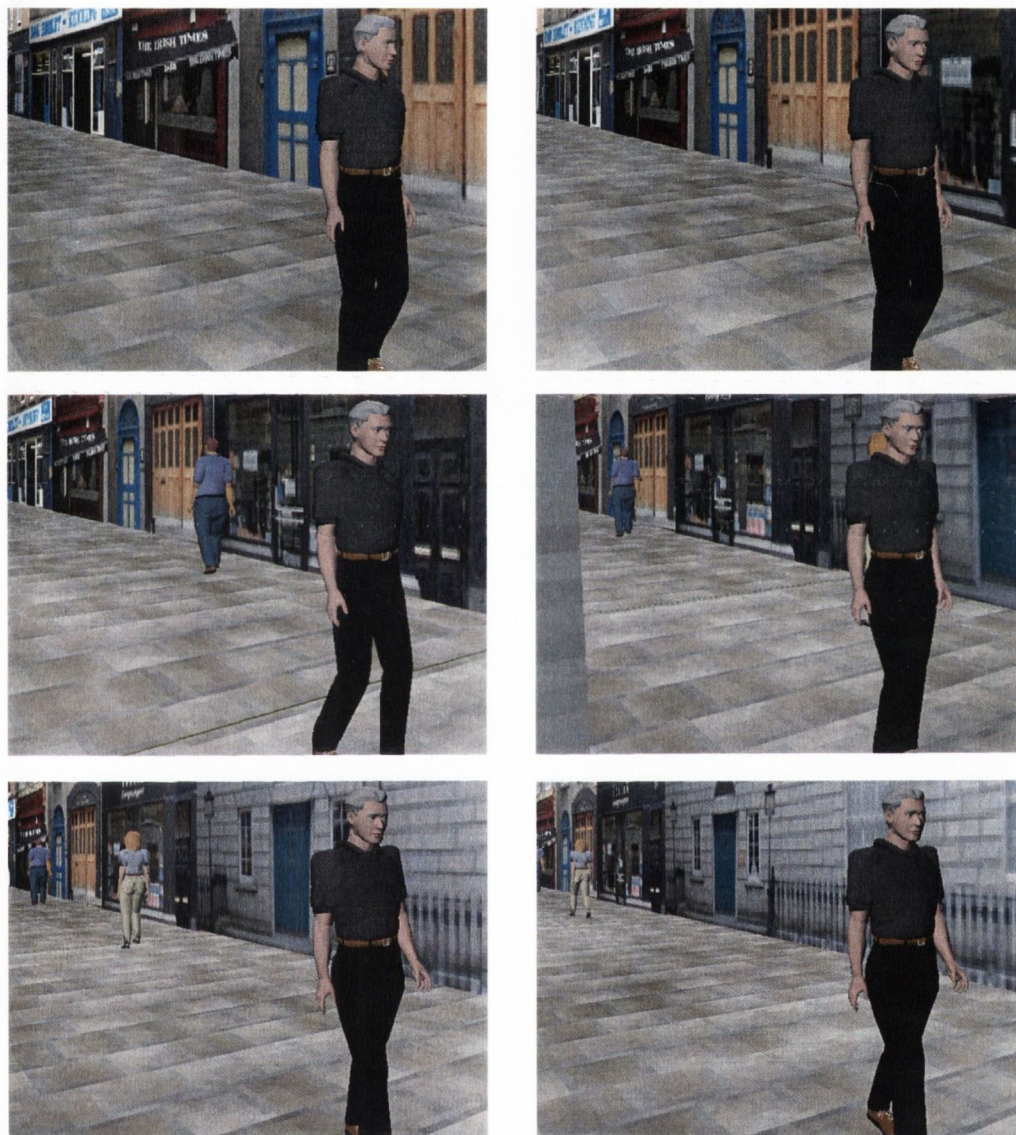


Figure 7.3: Successive frames of animation for gaze behaviours in the street scene.



Figure 7.4: Successive frames of animation for gaze behaviours in the street scene, continued from Figure 7.3.

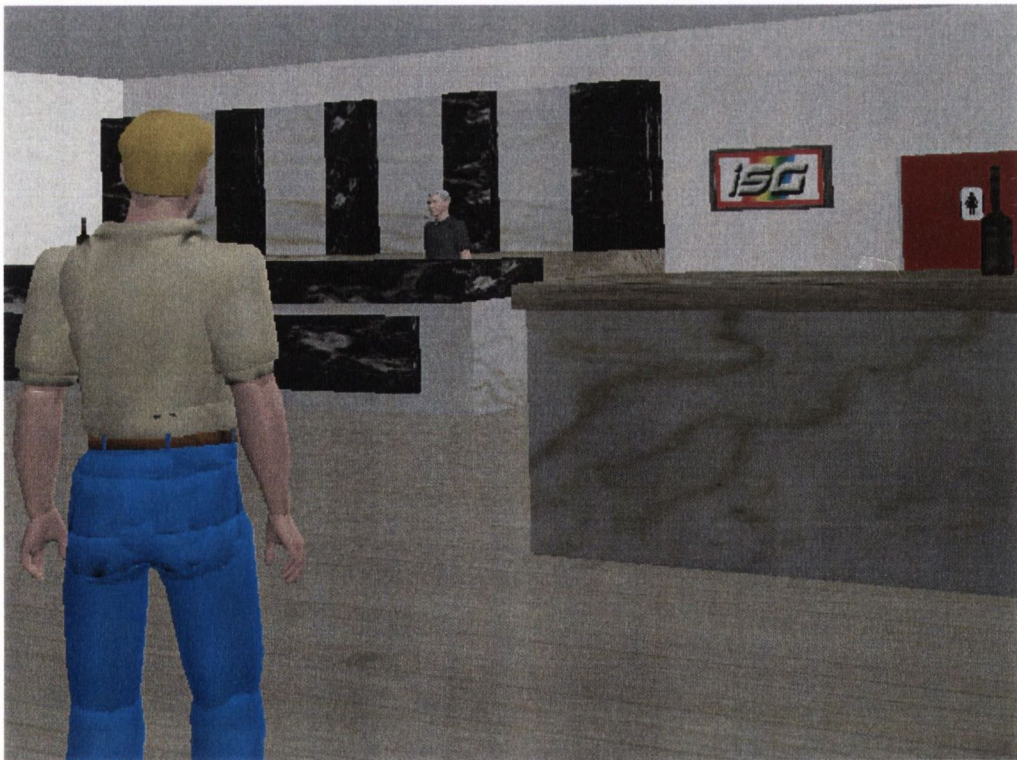


Figure 7.5: Screenshots from the pub scene.



Figure 7.6: Screenshots from the pub scene.



Figure 7.7: Successive frames of animation for gaze behaviours in the pub scene. In this scene, curiosity thresholds have not been implemented, so the character will continue to survey the scene indefinitely.



Figure 7.8: Successive frames of animation for gaze behaviours in the pub scene, continued from Figure 7.7.

are generated. Two virtual environments were created for testing and demonstrating the system, a street scene and a pub scene. The vision, memory and attention models were used to generate gaze behaviours for a single virtual human in these environments. Depictions of the environments and frames from the resulting animations are presented in Figures 7.1 through to Figures 7.8. Figures 7.3 and 7.4 show successive frames of an animation of gaze behaviours in the street scene, while Figures 7.7 and 7.8 show successive frames of animation for gaze behaviours in the pub scene.

7.2 Speed Evaluation

Speed is an important factor when considering virtual human animation, especially in the context of real-time applications.

A number of timing tests were conducted in order to evaluate the system presented here, in particular our implementation of the attention model which forms the core of the system. Of particular importance is how the model scales with respect to the complexity of the environment that the virtual human inhabits.

Each timing trial consisted of 50 separate tests on a Dell Dimension 8200 2Ghz Intel Pentium 4 Processor, with 512 MB RAM fitted with an Nvidia Geforce 4 Ti 4600 graphics accelerator. Results of each set of 50 tests were then averaged in order to provide the final speed metrics.

Timing results were conducted for a number of different cases. The stimuli extraction and resolution phases of the synthetic vision model and the uncertainty and saliency map generation phases were chosen as the basis of the speed calculations since they form the core processing stages that sensed information progresses through on its way to conversion into gaze motions.

As such, the information reaching the visual system will already have been filtered due to the implicit nature of the synthetic vision process. Since the false-views are rendered based on the viewpoint of the virtual human, any objects depicted in the false-views as stimuli will be those that have not already been culled due to being outside the viewpoint of the vision system. In addition, it should be noted that the occlusion of objects that lie within the view frustum is also handled implicitly; thus, further queries and calculations are not necessary to test if an object within the view-frustum is totally obscured by another, nearer object.

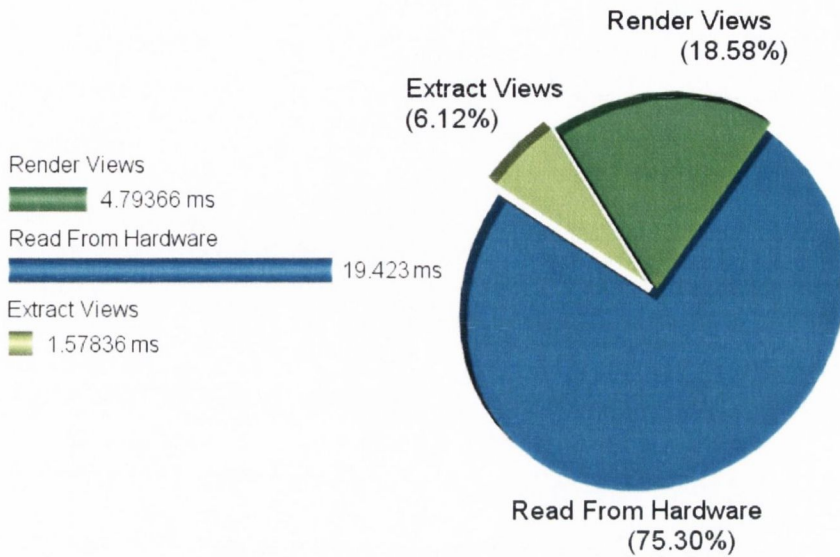


Figure 7.9: Amount of time taken to render, read and process the three scene renderings in order to prepare them for stimulus extraction and attention processing. All timings were recorded as the average of 50 tests moving through the street scene environment.

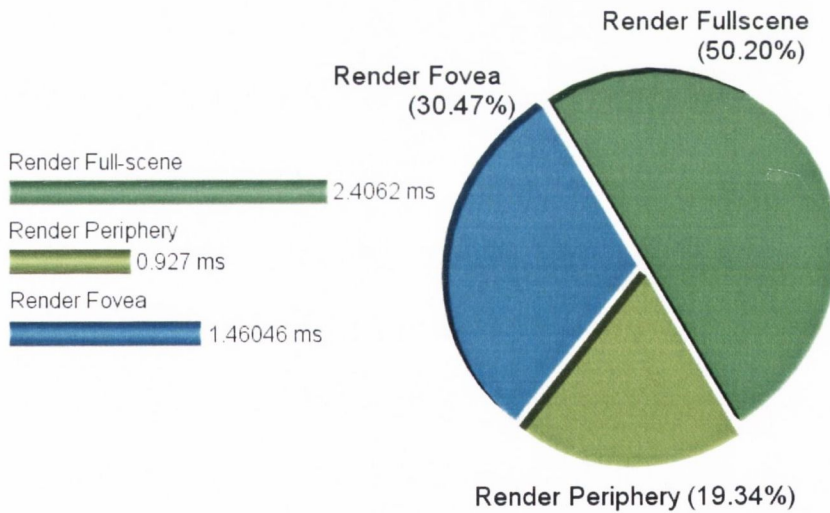


Figure 7.10: Illustration of the amount of time taken to conduct each of the three synthetic vision renderings, corresponding to *Render Views* in Figure 7.9. As would be expected, the full-scene rendering requires a significant amount of the overall rendering time, as the fovea and periphery are flat-shaded with texturing and lighting disabled.

Figure 7.9 illustrates timings for the process of rendering views from the perspective of the virtual human and making them available for the stimulus extraction stage of the synthetic vision process. The actual view rendering stage consists of three sub-stages: rendering the environment three times from the view of the virtual human (see Section 4.2.1 and Figure 4.4) and reading the rendering scenes back from the graphics hardware. Because the reading procedure often disrupts the operation of the graphics pipeline, it can be time-consuming. Because of this, in our system, the three renderings are composed as a single large rendering that needs only to be read from the hardware once per visual update. This is in contrast to three separate read procedures, which proved to be more time consuming. This method does necessitate a further extraction sub-phase where the respective renderings are cropped from the main rendering and thus made ready for use by the vision processing systems. Figure 7.9 details the timings for the three sub-stages of view rendering. It can be seen that reading the scene information back from the graphics card consumes a significant amount of processing time in comparison to conducting the multiple scene renderings.

Although multiple renderings of the scene environment would appear prohibitive, it should be noted that, of the three renderings, both of the false-colour renderings could be speeded up by using a multitude of visibility and level-of-detail techniques, none of which are present in the current implementation (see Figure 7.10). This complements the fact that these renderings are simplified in the first place since lighting and texturing effects are not applied. In terms of the visual system, the most prohibitive component would appear to be the full-scene rendering. The quality of this rendering is important as it forms the direct input to the attention system and determines the selection of salient scene locations.

Timing tests were conducted by progressively adding objects to the scene database. Objects were ensured to be in the virtual human's view frustum and not occluded in any way by directly generating the false-colour view; the unique false-colour for every object in the scene database was entered directly into the false-colour display. In this way, each object was ensured of occupying at least a single pixel in the final false-colour rendering and would therefore be considered as a stimulus during the proximal stimuli extraction phase. The number of objects tested in each trial was varied from 1 up to 16,384. This value was chosen as the maximum as it forms a worse-case scenario for the vision system. Since the false-colour renderings are

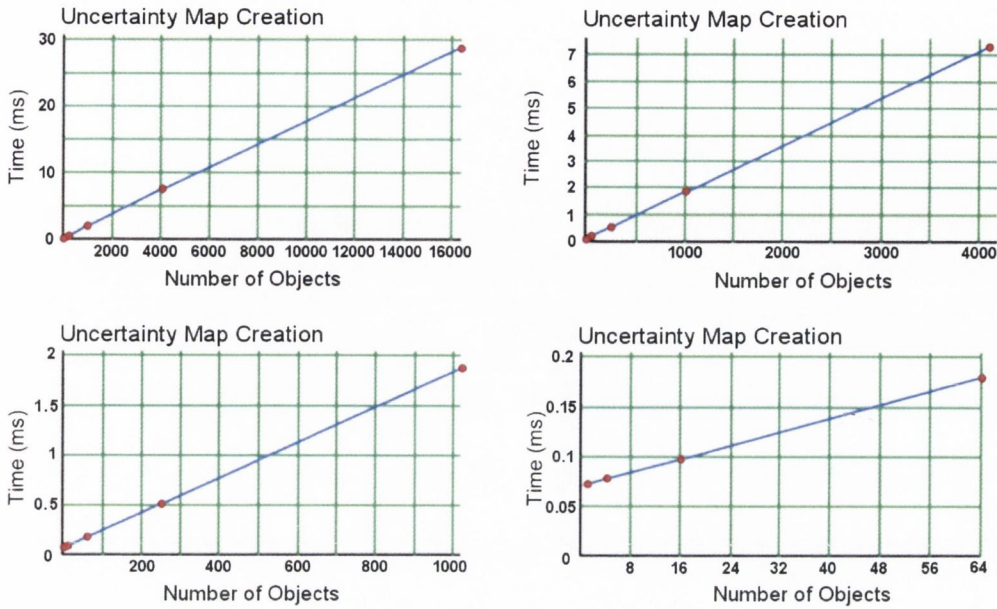


Figure 7.11: Graph of timing tests for uncertainty map creation. Red dots illustrate timing tests, where each dot represents the average of 50 separate tests. From left to right, top to bottom: timings for 0 to 16,384, 0 to 4,096, 0 to 1,024 and 0 to 64 objects respectively.

128x128 in resolution, this means that the maximum possible number of objects sensed is 128x128, where each object occupies a single pixel. Object false-colours were generated in an incremental manner and the concatenated false-colour red, green and blue values were used as a unique object name for referencing in the database.

Timings were conducted for 1, 16, 32, 64, 256, 1,024, 4,096 and 16,384 objects. This distribution of object numbers was chosen as most envisaged scenarios involving the vision system would involve between 1 and 256 objects; that is, up to 256 objects in view of the virtual human at any one time. Higher numbers of objects were only used to test for worse-case conditions that would be highly unlikely in all but the most complex virtual environments. In addition to this, these numbers were also chosen as they correspond to worse case scenarios of lower resolution false-colour renderings. For example, the 1,024 objects case corresponds to a worse case scenario for a 32x32 pixel false-colour rendering (the current false-colour rendering is 128x128). The results for extraction of peripheral stimuli are presented in Figure

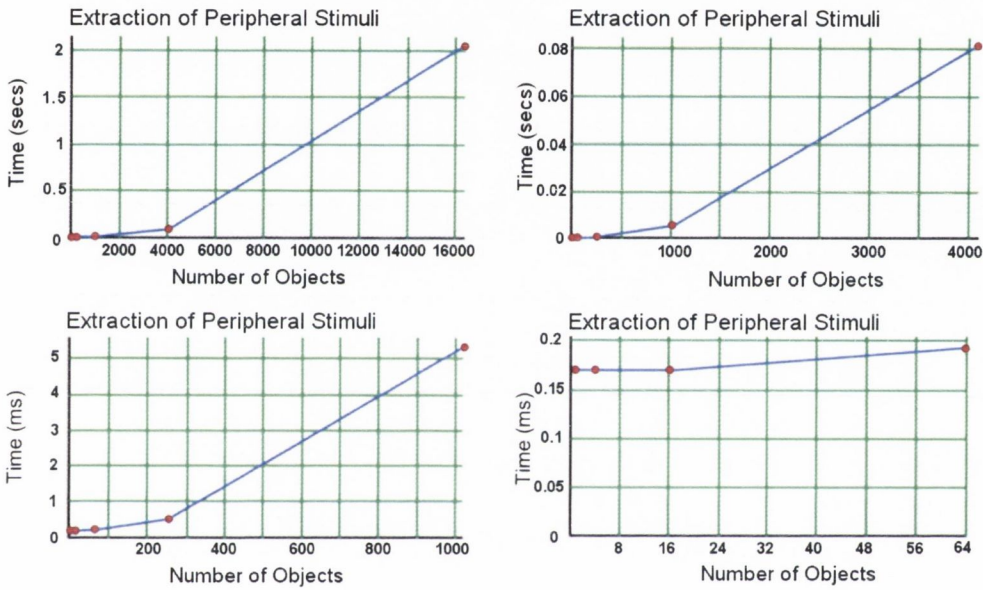


Figure 7.12: Graph of timing tests for the extraction of peripheral stimuli. Red dots illustrate timing tests, where each dot represents the average of 50 separate tests. From left to right, top to bottom: timings for 0 to 16,384, 0 to 4096, 0 to 1024 and 0 to 64 objects respectively. The extraction phase scans each pixel in the false-coloured view and constructs a new proximal stimulus if one does not exist already in the agent’s proximal stimulus list. This involves a search of the proximal stimulus list.

7.12. Timing profiles according to the tests described above for stimuli extraction in the foveal rendering are the same as those for the peripheral rendering since both renderings contain the same number of pixels and therefore can sense, at the most, 16,384 different objects. In terms of processing, it should be noted that each pixel of the false-coloured renderings must be scanned regardless of whether it contains an object. When a pixel containing an object is detected, a search must be conducted on the current stimuli list to see if the stimulus is already present (see Algorithm 1). This is necessary to update the stimulus details for stimuli already detected, such as bounding box and average colour information.

Timings for the stimulus resolution stage are detailed in Figure 7.13. When the number of stimuli increases dramatically, search time also increases since, in the test situation, the resolution phase has to search the world database. This is a worse case scenario however; in practice the memory component acts as a cache that is first searched for resolved object data. The full scene database is only searched if the resolved stimulus is not present in memory. A naming scheme has been introduced to facilitate full scene database searches. The scene database uses a hashing scheme based on concatenated object false-colour values.

The attention module was also tested for a variable number of objects in the visible area of the virtual human (Figure 7.16). As would be expected, the attention system ran in constant time from the input phase of the algorithm up to the saliency map generation phase (the fixation resolution phase involving, for example, the WTA network, was not included in these timings). This is to be expected, since the attention algorithm is image based as opposed to being object based like, for example, the stimulus extraction phase. That is, the attention algorithm has a fixed running time that is independent of the amount of objects visible to the agent; each pixel in the input image goes through the same processing procedure regardless of whether an object is occupying it.

A lower-level analysis was conducted of the attention algorithm averaged over 50 tests (see Figure 7.14). It is evident from these tests that the calculation of the orientation conspicuity map consumes most of the algorithm's processing time (81.93%). This is explained by the fact that orientation conspicuity calculation requires processing over four different base images, derived from the intensity channel of the input image and filtered by Gabor filters of orientation 0° , 90° , 180° and 270° . Because these base images act as root levels in Gabor pyramids, which must

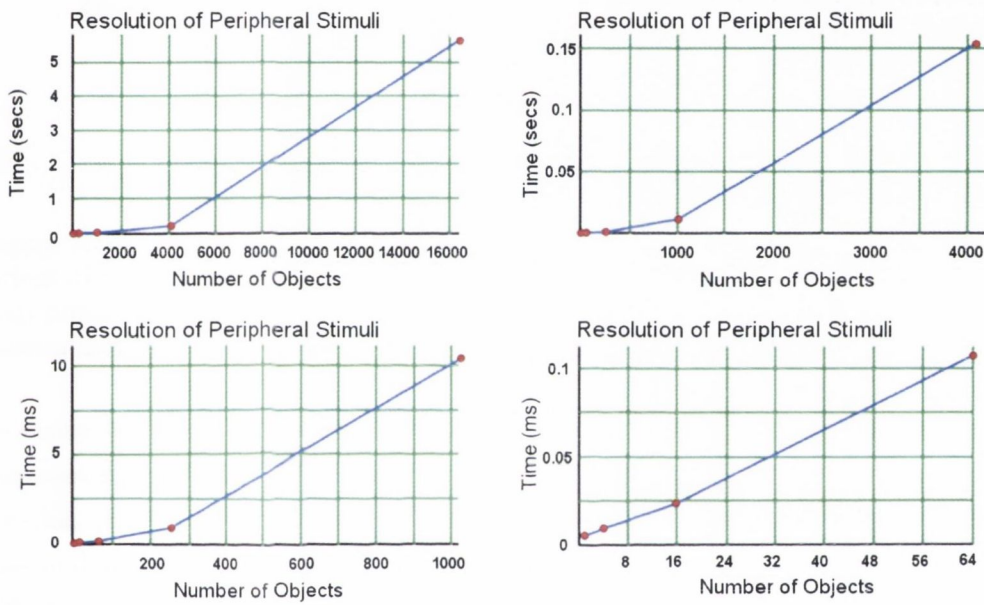


Figure 7.13: Graph of timing tests for the resolving peripheral stimuli. Red dots illustrate timing tests, where each dot represents the average of 50 separate tests. From left to right, top to bottom: timings for 0 to 16,384, 0 to 4096, 0 to 1024 and 0 to 64 objects respectively.

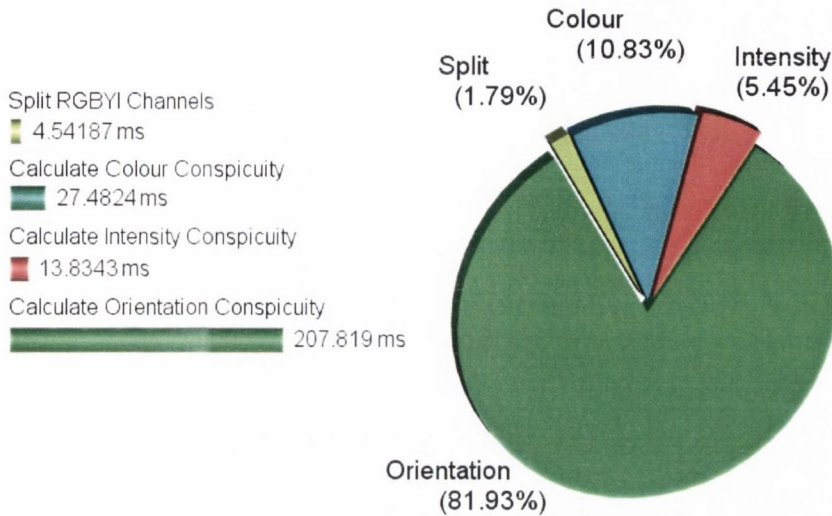


Figure 7.14: Depiction of timings for saliency map generation in the street scene. Orientation conspicuity calculation consumes a large amount of the overall saliency map processing time. Timings values are the average of 50 samples taken as the camera moved through the scene.

be processed into feature maps and then combined, the computational overhead increases significantly for each additional base image. This is the case also in the creation of the colour conspicuity map which takes 10.83% of attention processing time and consists of two base images: a blue-yellow difference image and a red-green difference image. In contrast, the calculation of the intensity conspicuity map takes only 5.45% of processing time, as there is only one base image, and thus one image pyramid, involved.

The saliency generation section of the attention algorithm runs in constant time with respect to the number of objects in the scene (see Figure 7.16) and had an average run time of 0.257983 seconds in the street scene (see Figure 7.15). This is the time taken between passing the input retinal image to the attention model and receiving the saliency map (excluding processing conducted by the artificial fixation extraction routine, such as the winner-take-all network). Although the creation of the saliency map generation routine averaged 90.63% of the total time, it should also be pointed out that the attention algorithm does not have to be run every frame. When a new gaze location is needed but the attention model does

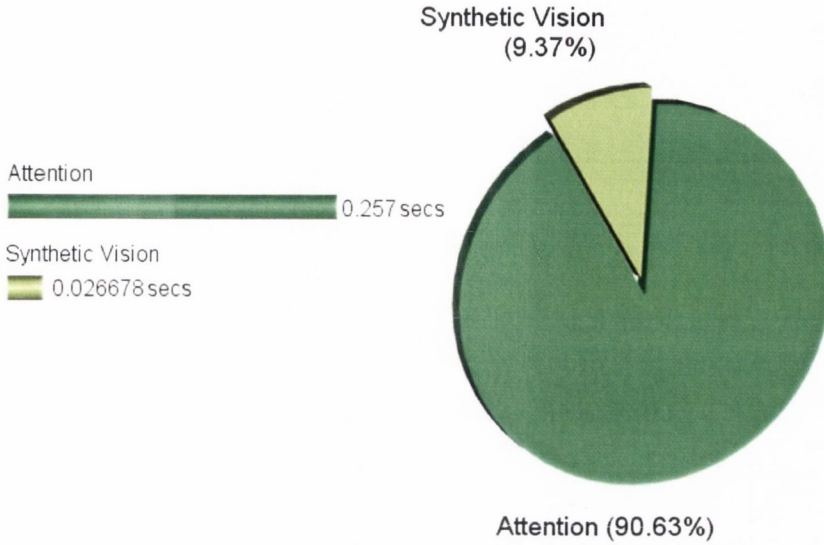


Figure 7.15: Overall timings for the synthetic vision and attention systems.

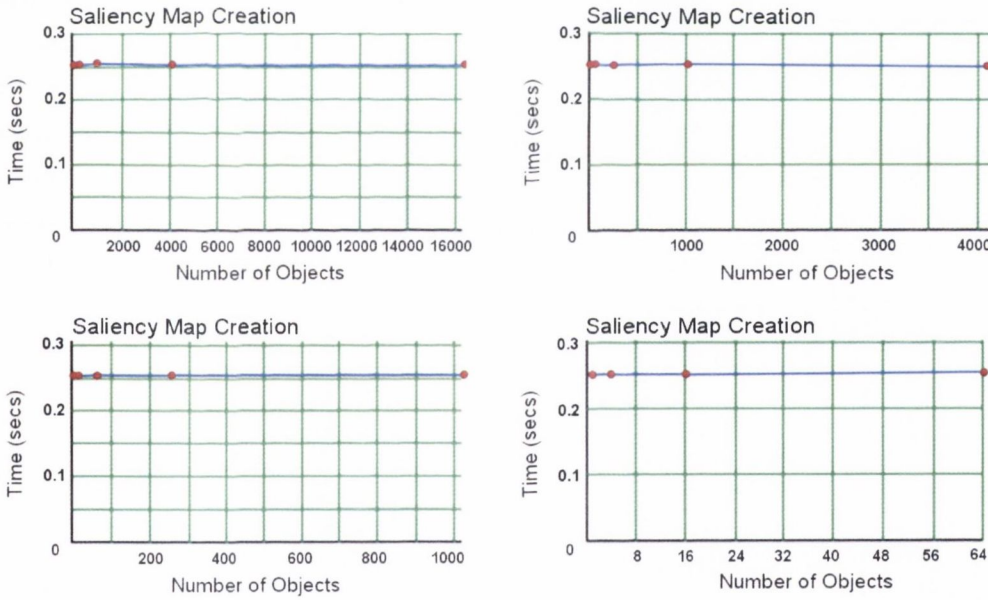


Figure 7.16: Graph of timing tests for saliency map creation with a varying number of visible objects. Red dots illustrate timing tests, where each dot represents the average of 50 separate tests. From left to right, top to bottom: timings for 0 to 16,384, 0 to 4096, 0 to 1024 and 0 to 64 objects respectively.

not need to be updated, only the sensing and winner-take-all network components need to be updated to obtain a new salient location. Since the retinal (full-scene rendering) image does not need to be taken in this case, further savings are made. In our implementation, when orientation calculations are enabled for saliency map generation, we distribute the attention processing over a number of frames (i.e. conduct time-slicing) in order to provide updates of the scene at interactive rates, thus enabling a real-time visualisation of the simulation.

7.3 Evaluation of Gaze Targets

This section deals with the evaluation of locations in the environment that are to be looked at, as dictated by the attention component. These locations, referred to in psychology literature as Regions-of-Interest, or *ROIs*, are brought into the field of the fovea during eye fixations - that is, when the eye is focused relatively steady between saccadic eye movements. Only a limited number of *hROIs* (human Regions-of-Interest) need to be focused on by a human in order for the brain to be able to recognise complex visual input.

Although the behaviours generated by our system consist of the interactions between multiple components, the key component to be focused on is the bottom-up visual attention algorithm, which acts as the primary controller of gaze. This algorithm belongs to a class of image processing algorithms, *IPAs*, that algorithmically detect Regions-of-Interest, or *aROIs*, in a visual input.

There are a number of reasons as to why the evaluation of gaze targets is difficult:

- A single subject looking at the same scene multiple times may look at different areas of the scene and therefore generate different *hROIs*. Although this may be due to top-down task influences, even if the subject is asked to look at the scene in a free-viewing situation, unfamiliarity with the scene may affect their eye movement patterns, as shown by Zangemeister et al. [WZS95].
- There are variations in eye movements between different people when they view the same scene. In a general viewing situation, Privitera and Stark found that when different people viewed the same image, only an average of 54 percent of their *hROIs* cohere [PS00b]. Algorithms cannot be expected to predict human fixations better than the coherence among fixations of different people.

- In Chapter 5 we considered a model for generating likely areas of interest in a scene based on the properties of the scene and in Chapter 6.2.1 we saw how covert orienting behaviours, such as eye movements, could be made towards locations selected from the attention model. Unfortunately, in practice, the link between the underlying attention processes and outward eye-movements is more complicated. The current debate on the link between attention and eye movements for idle viewing conditions centers around the issue of whether scene memory or the external scene stimuli determine fixation locations; in other words, whether attention during idle viewing conditions is directed or elicited. It is acknowledged that most likely a mixture of both memory and environmental stimuli play a role in the generation of fixations.
- The mapping between where a person looks and where they are attending to is not always one-to-one. Due to *covert attention*, a person may be using peripheral vision to pay attention to a location without fixating it and may be looking at a location without really attending to it.
- Finally, experiments that consider fixation targets for subjects during idle viewing conditions are difficult since one can never be sure whether a participant is really viewing a scene in a task independent manner. If a subject is not provided with a task, it doesn't necessarily mean that they are viewing the scene without a task: perhaps task is self-defined and the person may tend to look at, for example, objects relating to personal experiences or interests.

Our evaluation is aided in some way by our choice of the attention model by Itti et al. In particular, the saliency map model has been validated by Itti and Koch [IK00]. The model was tested in a number of ways on a wide variety of real colour images ranging from natural outdoor scenes to artistic paintings and showed good qualitative agreement with human psychophysical literature in classical search tasks, such as pop-out and conjunctive search. As pointed out by Itti and Koch, the model performs remarkably well at detecting salient targets in cluttered natural and artificial scenes.

Furthermore, the model has been used successfully in the graphics community as part of a number of perceptually-driven rendering applications [YPG01] [HMYS01]. Also, Privitera and Stark [PS00b] showed, using both qualitative and quantitative

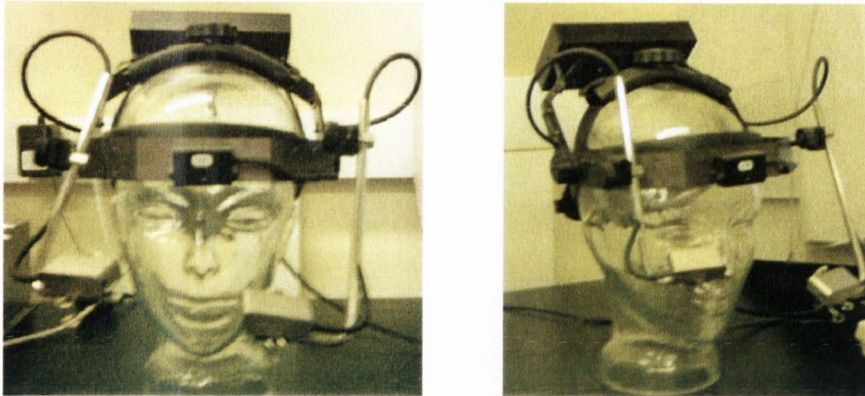


Figure 7.17: View of the SMI EyeLink eye-tracker head-mounted band used in the experiments. The band weights 600 grams and is designed to be balanced and comfortable to wear.

tests, that image processing algorithms, such as the attention algorithm investigated in this thesis, could indeed predict eye fixations over static images. As noted by Marmitt and Duchowski [MD02], their findings appear to indicate that multiresolution strategies, such as that adopted by the Itti et al. model, are very efficient for several classes of images.

7.3.1 Gaze Targets for Motion

In order to analyse and compare the aROIs of the attention system to those fixations made by real people, experiments were carried out using an eye-tracking device on a number of virtual scenes. The results of these experiments were compared with the results of attention system in order to provide an insight to the validity of the approach pursued. A description of the experimental procedure follows.

Participants

Five subjects, one woman and four men, between the ages of 22 and 30, participated in the experiment. All had normal or corrected normal vision.

Apparatus and Methods

Eye movements were recorded using a remote eye-tracking device that uses infrared light. The EyeLink Eyetracker, built by SMI-SensoMotoric Instruments from Teltow Germany, is connected to a PC. The eye-tracking system has a video sampling



Figure 7.18: The eye-tracker in operation on a test subject during data collection. The subject looks at the output monitor: data from the eye-tracker is transferred to the operator PC (seen in the background) to provide real-time feedback for the operator.

rate of 250HZ, providing a spatial resolution of better than 0.01° with a display-to-eye working distance of up to 240cm. The eye position tracking range is $\pm 30^\circ$ horizontally and $\pm 20^\circ$ vertically. It features true gaze-position accuracy of between 0.5° and 1.0° average error, using head tracking in the ranges of $\pm 20^\circ$ horizontally and $\pm 17^\circ$ vertically, allowing head position compensation. It is non-invasive in its acquisition of eye movements from subjects. It also provides online analysis of eye movement data into saccades, fixations and blinks.

The Eyelink eye-tracking system consists of a 600 gram head-mounted band that is designed to be balanced, comfortable and stable. Two miniature high-speed cameras simultaneously take 250 images per second of both eyes. A third camera tracks 4 infrared markers that are mounted on the monitor in order to provide head motion compensation and true gaze position tracking. The subject's PC and monitor output are linked to an operator console. On the operator console, an image processing system synchronously analyses the images from all three cameras in real-time to provide pupil position of both eyes and the marker position.

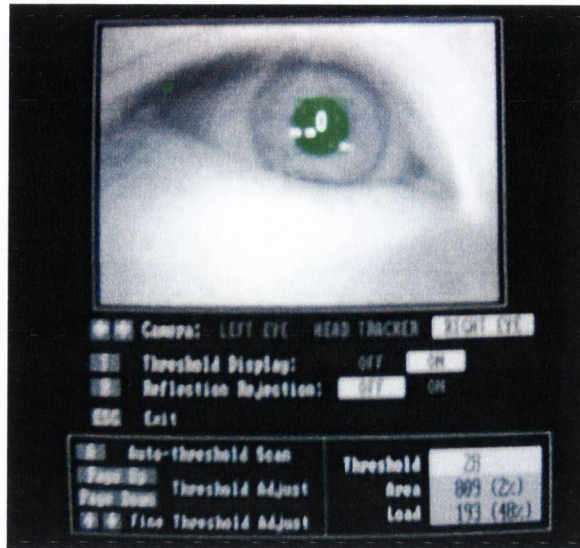


Figure 7.19: Calibration phase of the eye-tracking experiment on the operators PC. Calibration consists of finding a suitable threshold for detection of the subject's pupil so that accurate eye tracking may take place.

Environment

The experimental environment is shown in Figure 7.18. There was electric light in the room during the experiment, which was not interfering with the recording capabilities of the apparatus.

Experimental Procedure

Each session began with the seating of the subject approximately 60cm in front of the output monitor and receiving instructions about the experiment. An instruction guidelines checklist was followed in order to ensure consistency both in subject instructions and in the experiment set-up procedure. Subjects were informed of their task and asked to minimise their head movements.

The eye-tracker headband was then mounted on the subject and calibrated using the Eyelink calibration procedure (see Figure 7.19). The experimental session started as soon as the subject indicated their readiness by pressing the return key. During the experiments, calibration was only carried out for the subject's dominant eye and only this eye was tracked. A record of the subject's dominant eye was made at the beginning of the calibration procedure. The experiment did not proceed until

a *good* calibration was acquired to ensure a high quality of eye tracking (calibration results may be *failed*, *poor* or *good*).

Six sets of images were displayed to the participants on the screen and were clearly visible (see Figure 7.18). Each set consisted of ten separate images from the virtual environment displayed sequentially at a resolution of 1024x768. The source image resolution and the monitor resolution were the same. Each frame was displayed for 500 milliseconds, during which eye movements were captured. After the 500 milliseconds, the next image in the set was then displayed directly after the previous image. At the beginning of each new set, a *drift correction* procedure took place. Almost all eye-tracking systems have a tendency to drift over time, meaning that the error between the computed gaze position and the subjects actual gaze position is increasing. Drift can be caused by a number of factors, such as head movement. The drift correction procedure displays a grey screen with a black-dot at the center and requires the subject to look at this position and press a key to continue; they are not permitted to continue the experiment until they are looking straight at the dot.

Set sequences were randomised before being shown to each participant, although image order within each set remained constant. This was necessary, since the ordering of the images provide an appear of a scene that is moving in slow motion.

7.3.2 Data Analysis

The raw eye movement files capturing by the eye-tracking device were converted into fixation data files using the Eyelinks iView software package. After being processed by this software package, the fixation files were saved as plain text and loaded into the system for comparison with the aROIs generated by the attention model using a visual analysis procedure. This procedure grouped all fixations made by all subjects over a single image into a batch. In this format, each batch of fixations could be displayed as overlays on the sample scene to provide a clear insight into where all subjects were looked on a particular frame.

A number of aROIs were then generated for these frames using three different fixation calculation strategies:

1. A *winner-take-all* strategy uses the winner-take-all neural network as proposed by Itti [Itt00] in order to inhibit previously selected areas. The WTA network

must be reset for each frame and thus maintains no memory over multiple frames.

2. A *winner-take-all without orientation* is the same as the winner-take-all strategy outlined above, but does not include the orientation conspicuity calculations in the saliency map generation procedure.
3. An *object-based memory* uses the memory system proposed in Chapter 4 to modulate the saliency map calculated each frame by an uncertainty map maintained over the entire sequence of frames. Fixations in the image are resolved into the objects that are being fixated upon and these object's uncertainty values are then reduced in memory. In this way, the object-based memory method has a lower likelihood of looking at parts of the scene that have previously been looked at, even if such areas still had a high conspicuity.

Experiments were conducted on scenes where the viewer appeared to be moving. That is, each frame was captured from the virtual environment as the camera moved through the environment. Frames were also shown for only a small amount of time (500ms) in order to provide a sense of continuity in the scene. Moving images were chosen as the basis for tests since Marmitt and Duchowski evaluated the attention model proposed by Itti for dynamic scenes and found that the correlation between human and artificial scan paths was lower than expected [MD02].

A number of the results from the experiments are shown in Figures 7.20 through 7.27. Results for a street scene are shown in Figures 7.20 through 7.25 and are shown for the bar scene in Figures 7.26 and 7.27. There are ten frames per trial, representing motion through a virtual scene. Each frame was shown for 500ms during which eye movements were recorded. Frames are temporally ordered for display from the top row to the bottom row in each figure. Fixations are represented by filled circles and saccades by the lines joining the circles. Each frame is represented by a row in the Figures: the first column contains the frame overlaid with the human scan paths from the experiment. All scan paths are colour coded according to subject. Dotted lines indicate saccades that connect the last fixation of the previous frame to the first fixation of the current frame. Fixation sequence is indicated by descending colour intensities, for example, a dark yellow dot indicates a later fixation than a bright yellow dot.

The second column contains the saliency map calculated for that frame with

the attention model from Chapter 5. The saliency map that is shown encodes colour, orientation and intensity features in the image and gives an overall impression of the correspondence between the human scan paths and the raw output of the computational model of attention. Brighter areas in the saliency maps correspond to areas deemed to elicit attention.

The third column illustrates the frame overlaid with three varieties of aROIs generated from the results of the attention model, according to Section 6.4.2. Red indicates fixation generation using the winner-take-all strategy, green using the winner-take-all without orientation strategy and blue using the object-based memory strategy. In cases where green fixations are not visible, they are the same as red fixations, indicating the results for the winner-take-all and winner-take-all without orientation strategies were the same. The number of artificial fixations generated in each frame was the average of the number of human fixations that occurred during that frame.

Figures 7.20 and 7.21 show cases where the attention models did not correspond well with the human scan paths. The artificial fixations in this test were all very eccentric in the scene, with few fixations occurring near the center of the screen. In contrast, human fixations tended to be clustered nearer to the center of the screen. This could have been for a number of reasons. First of all, the direction of movement was towards a position near the center of the screen. This may indicate that subjects tend to look in the direction that they are travelling; the attention algorithm did not take the direction of motion into account. Secondly, participants tended to look at buildings at the end of the street: it is known that distance can play a role in eliciting attention, but the attention algorithm tested here also does not take this into account. Finally, central clustering of fixations for human subjects has also been noted in other work [MD02]; in some cases, human subjects just naturally look less at eccentric regions of images. The same situation appears to be reflected in Figures 7.24 and 7.25, although viewers did tend to look at more eccentric positions to the right of the frames.

Artificial scan paths in Figures 7.22 and 7.23 did appear to correspond better with their human counterparts. This is evident by looking at the output of the saliency map (center column) which indicates a band of high saliency running across the center horizontal of the scene. Again, most human fixations appeared to be made towards the center areas of the scene, which was also the direction of motion. This

occurred even when the locations had been looked at previously and it appears that a few choice locations were looked at multiple times, something pointed out by Yarbus for static imagery [Yar67].

In contrast to the previous examples, Figures 7.26 and 7.27 consist of rotational motion as well as translational motion. In these examples, the attention system appears to emulate human scan paths a much better way. There are a number of reasons for this. First of all, the human scan paths during these frames were spread out and occurred in more eccentric areas than in previous examples. This is emulated in a better way by the attention system, since it is not biased towards central areas of a scene. Secondly, the pub scene appears to be less complex than the street scene: because of the uniform texture of the wall, objects tend to stand out more, whereas in the street scene, individual objects are harder to discern due to texture. This is also evident by considering the saliency maps: for the bar scene, there are a few single, well defined areas with high saliency, whereas areas of saliency tended to be more uniform over the street scene images.

Another interesting factor in this case was the position that human scan paths appeared to be clustered around. During the rotational component of the scene (frames 5 to 10), although human scan paths were still relatively grouped together, they were grouped towards the direction of the rotation. This finding suggests that the viewers attention is directed towards newly uncovered parts of the scene during rotations. It is likely that this is related to information gathering and uncertainty: new parts of the scene have not been seen before and therefore provide new information, whereas older scene elements have been looked at before and the viewer therefore has a higher confidence with regards to scene certainty.

Overall, the experiments raised a number of important issues:

1. Human fixations in all of the scenes tended to be far less eccentric than those generated by the attention algorithm and, in many cases, appeared to be clustered together. In contrast, the attention algorithm tended to generate fixations that were further apart.
2. When motion was translational, human fixations tended to cluster near to center of the screen in the direction of motion. This may be explained in the street scenes by elaborated information being available (as textures got nearer to the viewer, they became clearer), In general, there may be a human

propensity to look near the center of the view in virtual environments (see [MD02]).

3. When motion was rotational, human fixations were more eccentric. Clustering of fixations tended to take place nearer the direction of the rotation. This could be explained by the introduction of new information from that direction.

In general, the IPAs appeared to provide very good results for the bar-scene (Figures 7.26 and 7.26), but provided mixed results in the street scene examples. In most cases, the attention algorithm produced fixations that were highly eccentric, and perhaps more suitable to search tasks than idle viewing conditions for scenes involving motion. Most human fixations during idle viewing appeared to be clustered near the center of the view: this suggests that the output of the attention model could be improved by an eccentricity based weighting, for example, a Gaussian centered at the view direction.

Overall, the *object-based* attention strategy appeared to perform equally with both the *winner-take-all* and *winner-take-all without orientation* strategies for the street scene. The strategy appeared to contend particularly well in the bar scene however (see Figures 7.26 and 7.26) where the camera was rotating: rather than continuing to fixate on previously seen objects, as the other strategies tended to do, the *object-based* strategy paid more attention to the newer areas of the scene coming into view, in line with behaviour displayed by most of the subjects.

During these experiments, a comparison between human scan paths and aROIs was complicated by the fact that subjects were not fully immersed inside the virtual environment and had to view the window into the environment from a distance of 60cm over a set of still images to provide the effect of moving. This would appear to have a number of consequences:

1. Since human subjects were seated at a distance from the monitor, peripheral vision played a smaller role in the outcome of the experiments. This could be quite significant as peripheral vision may play an important role in bottom-up attention and is known to be important for motion detection. To elaborate, areas in the periphery of view are harder to recognise than those in the center of view due to the nature of the human eye. Therefore, when looking at a scene for the first time, one may be more inclined to generate fixations towards eccentric stimuli in a stimulus-based, i.e. bottom-up, manner in order

to provide some form of recognition. When a scene covers a smaller field-of-view, scene components may be more readily recognised and bottom-up attention plays a smaller role in the dictation of eye-movements as attention guided by internal motivation assumes control.

2. Subjects were not permitted to move their heads or control their movement in the virtual environment during the experiment. Instead they were forced to follow preplanned paths and rotations, which was constraining. Under normal circumstances, a gaze motion to a target may also uncover further peripheral fixation targets outside of the initial field-of-view. The experimental method adopted here could not account for these type of targets.
3. Due to the nature of the software used for the experiment, scan paths could only be captured for still images. The impression of motion was obtained by displaying frames taken from the scene in sequential order at a very low frame-rate (2 frames/sec). Frame rates in the region of 30 frames per second would have been ideal in order to ensure persistence of vision.

All of these limitations could be resolved by the use of a virtual reality head mounted display incorporating an eye tracker. Ideally, a *string editing* approach would then be used to provide a more elaborate, quantitative, comparison of human and algorithmically defined scan paths (see [PS00b] for details on string editing).

7.4 Evaluation of Gaze Animations

In order to analyse the plausibility of the animations generated by the gaze module, human participants were asked to rate a number of animations. A description of the experimental procedure follows. Six subjects, one woman and five men, between the ages of 22 and 30, were shown multiple animations. Participants were not informed of the precise nature of the experiment and were simply asked to gauge their overall impression of the realism of the animation. They did this by rating each animation on a scale of 1 to 10, 1 denoting low realism and 10 denoting high realism. A written description of the reason for the choice made by the participant was also requested in each case. Participants were provided with two sets of animations, each set consisting of five separate animations. These animations consisted of a human-like character making gaze motions to a number of eccentric targets using the gaze



Figure 7.20: Images from trial *a* in the motion experiment. This trial is continued on Figure 7.21. (Left column) Human fixations are overlaid on the original frame. (Middle column) The corresponding saliency map scaled to the same size as the test image. (Right column) Artificial fixations generated by the three different attention strategies. Each image was shown for 500ms during which eye movements were recorded.

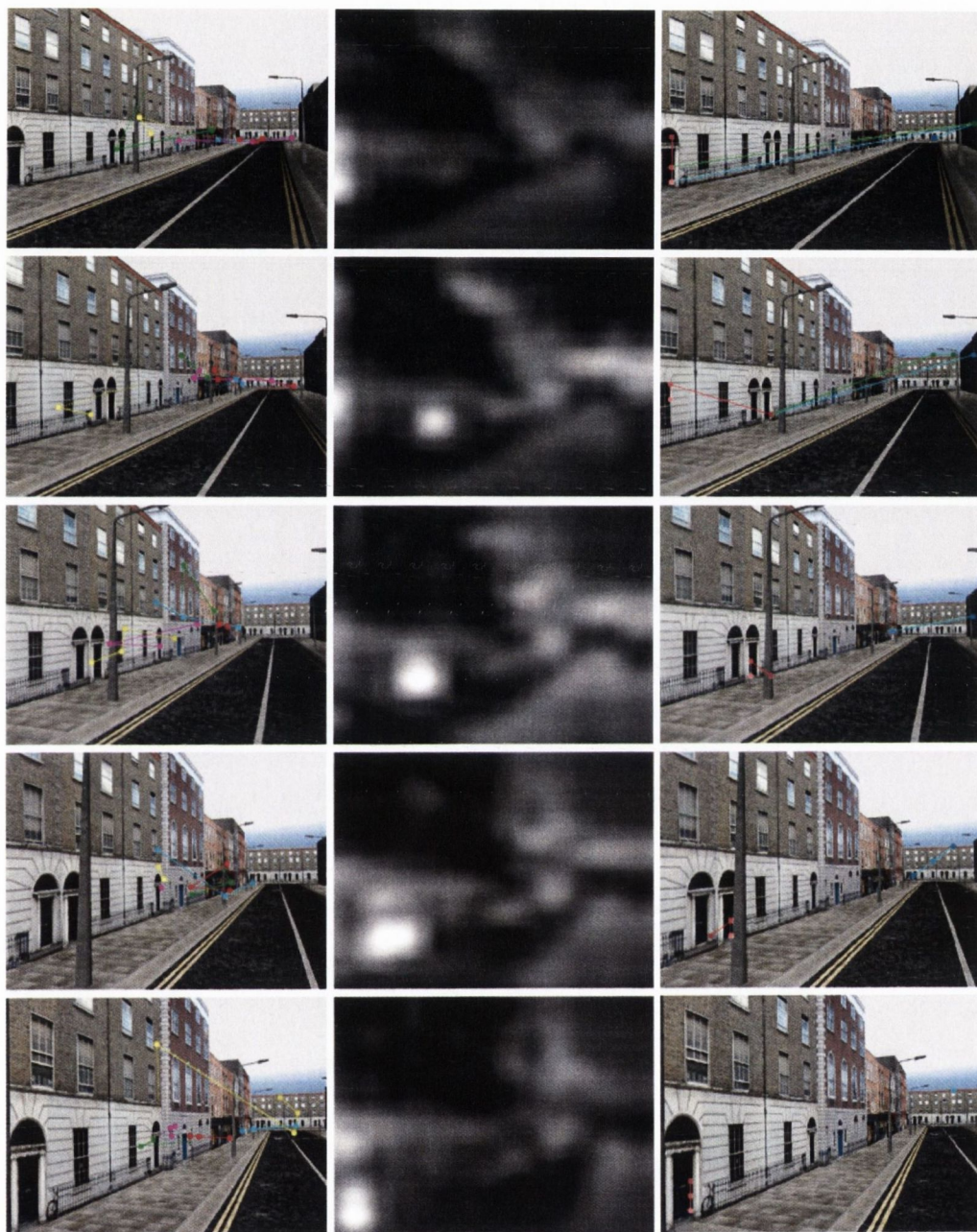


Figure 7.21: Images from trial *a* in the motion experiment, continuing on from Figure 7.20. The temporal ordering is left to right, top to bottom. Each image was shown for 500ms during which eye movements were recorded.

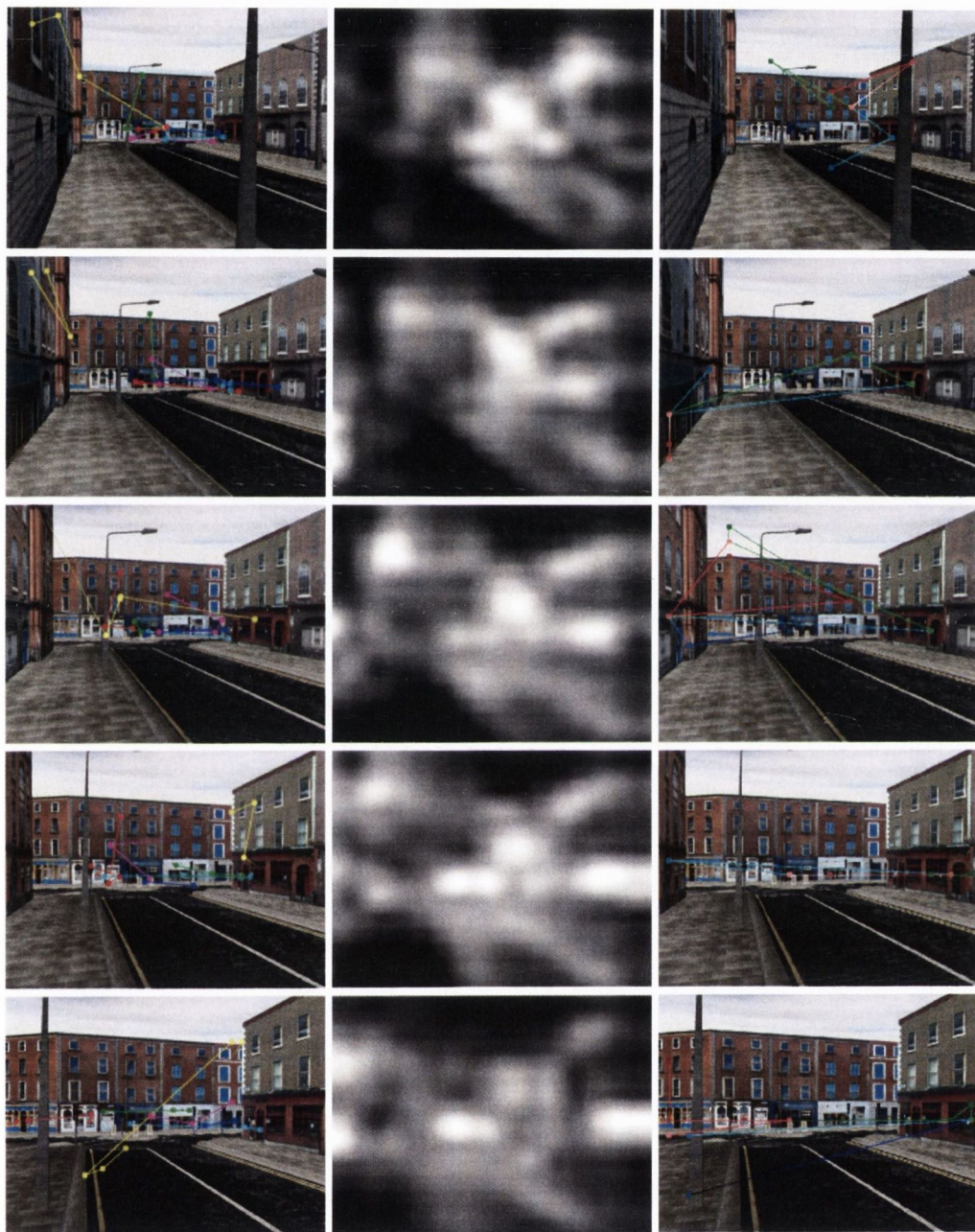


Figure 7.22: Images from trial *b* in the motion experiment in the motion experiment. This trial is continued on Figure 7.23. Each image was shown for 500ms during which eye movements were recorded.

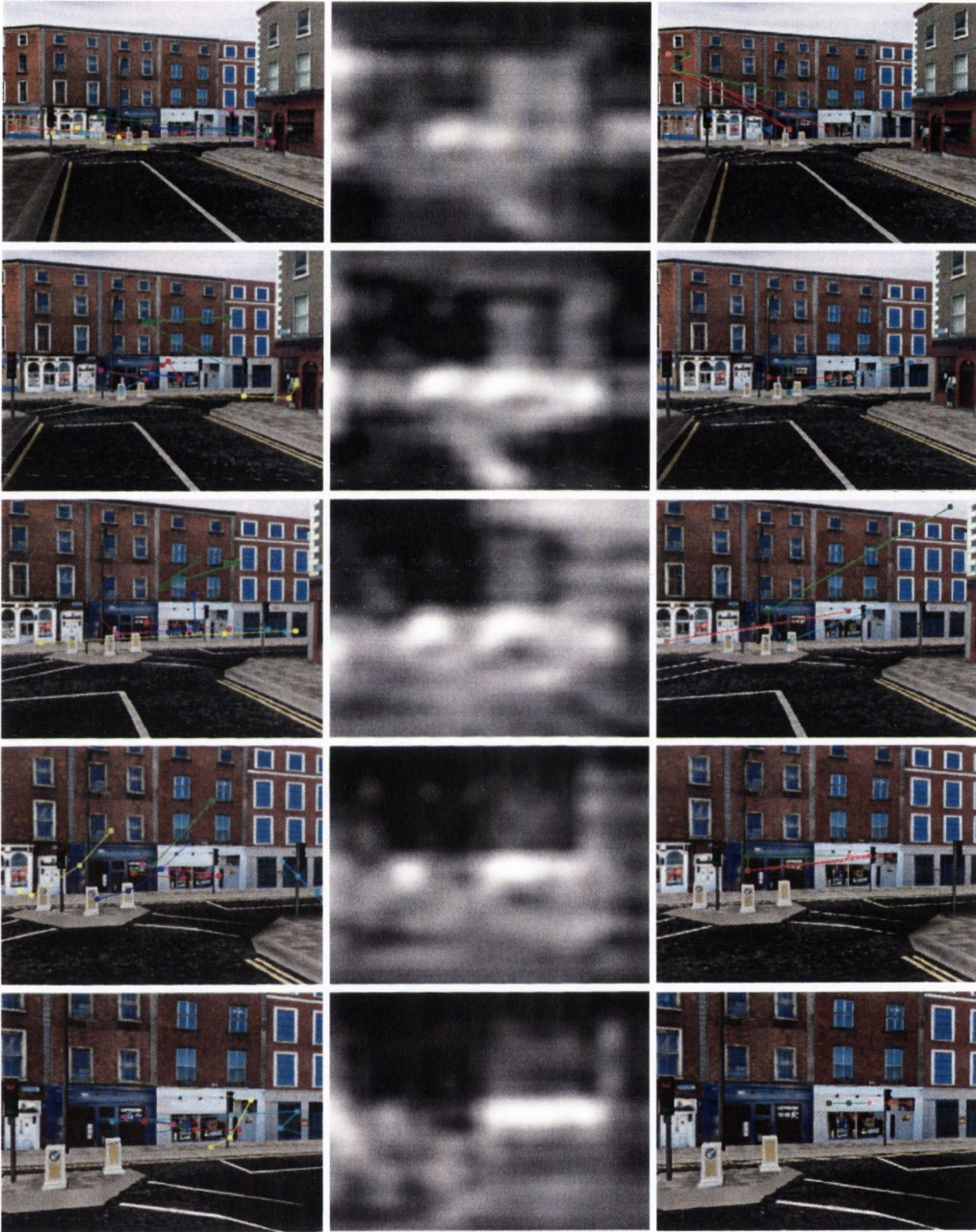


Figure 7.23: Images from trial *b* in the motion experiment, continuing on from Figure 7.22. Each image was shown for 500ms during which eye movements were recorded.



Figure 7.24: Images from trial *c* in the motion experiment. This trial is continued on Figure 7.25. Each image was shown for 500ms during which eye movements were recorded.

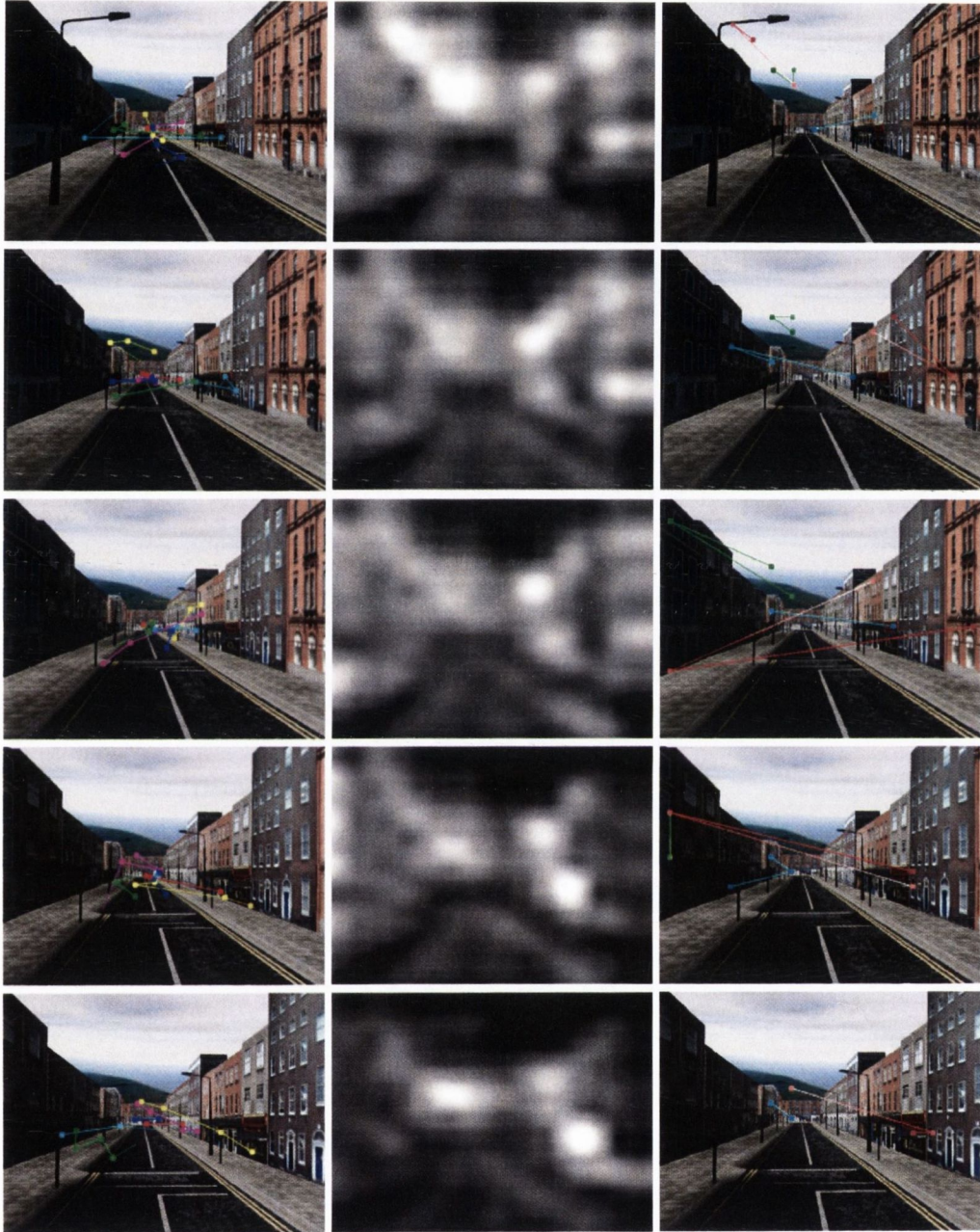


Figure 7.25: Images from trial *c* in the motion experiment, continuing on from Figure 7.24. Each image was shown for 500ms during which eye movements were recorded.

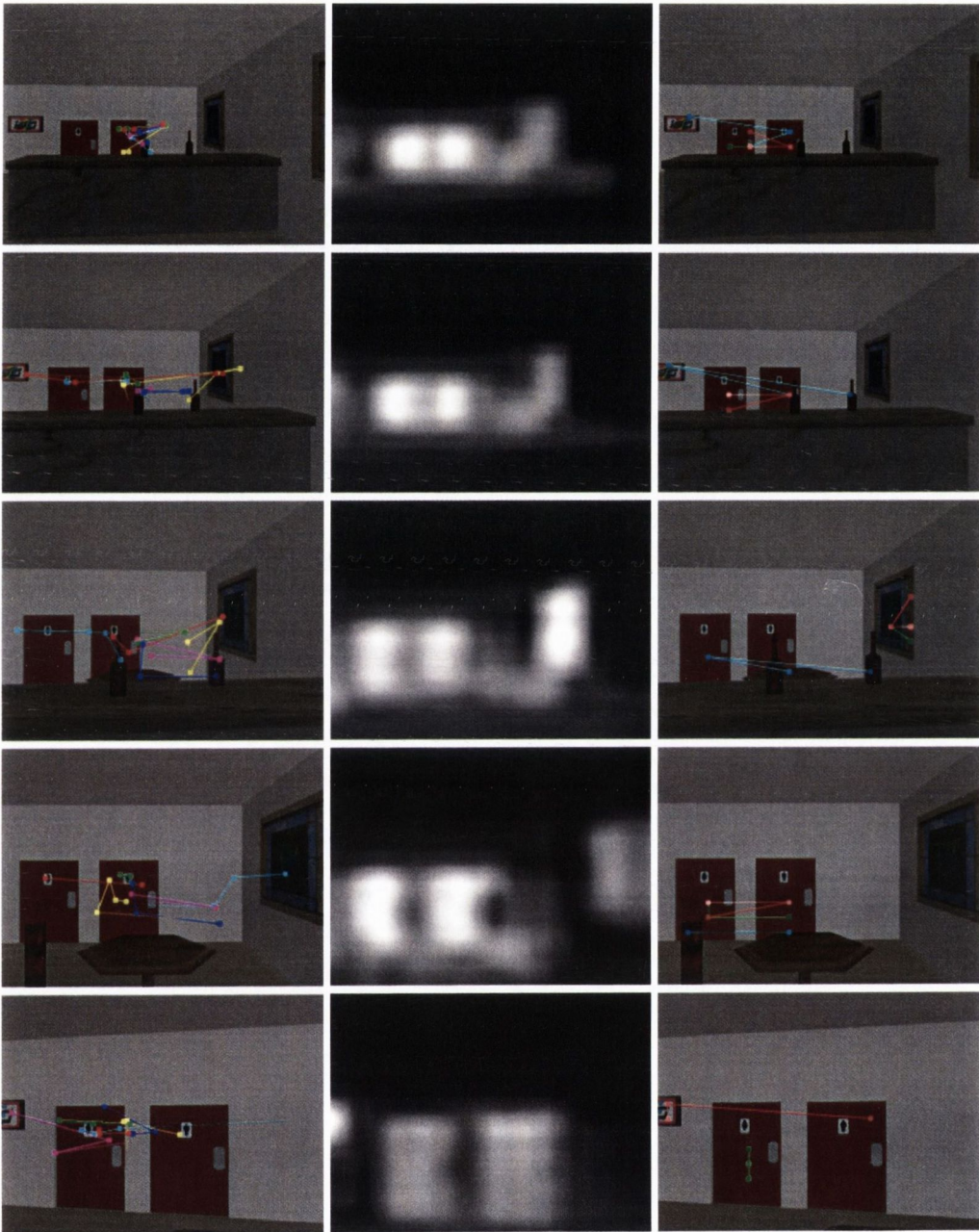


Figure 7.26: Images from trial *f* in the motion experiment. This trial is continued on Figure 7.27. Each image was shown for 500ms during which eye movements were recorded.

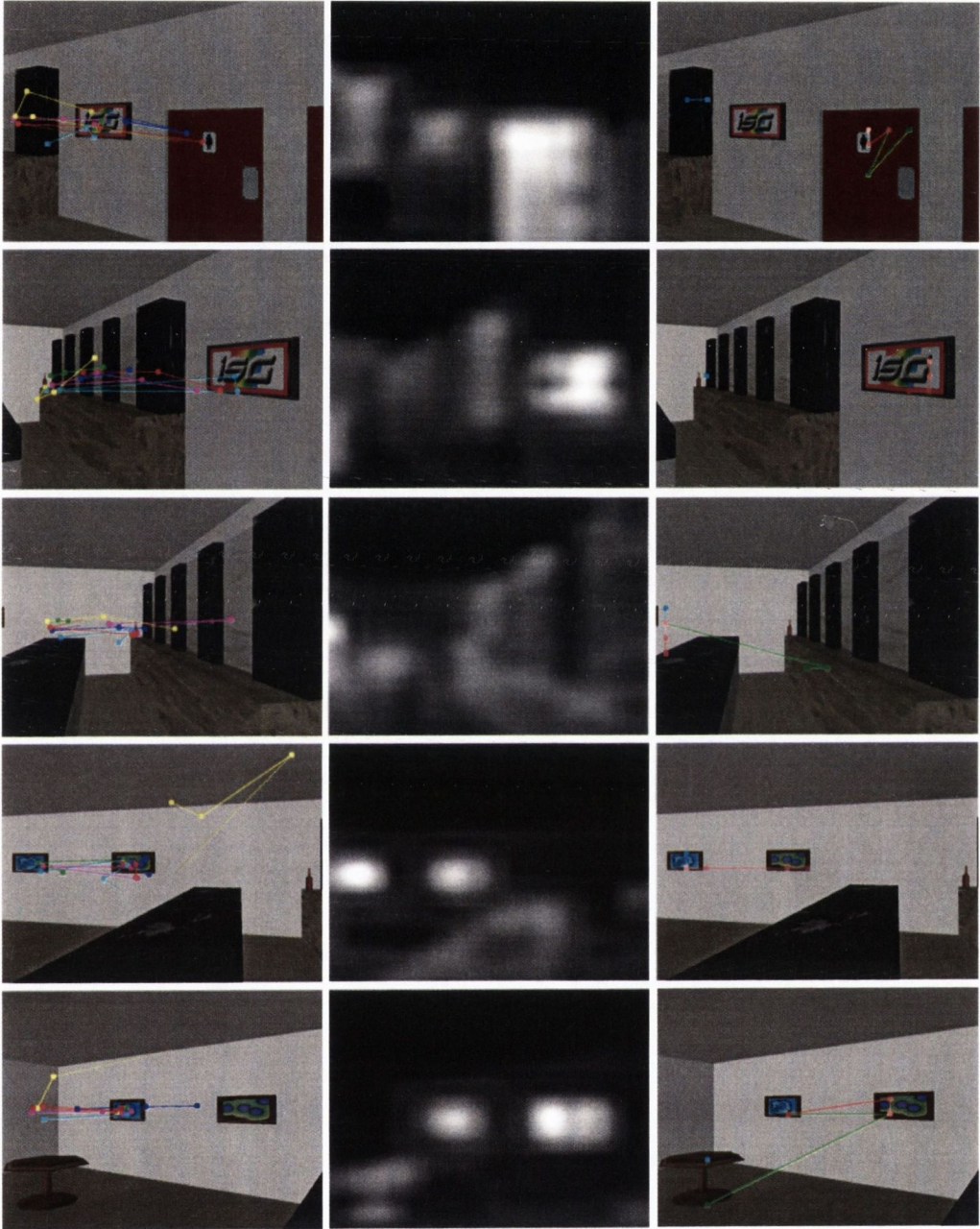


Figure 7.27: Images from trial f in the motion experiment, continuing on from Figure 7.26. Each image was shown for 500ms during which eye movements were recorded.

module discussed in Section 6.3.1. Each animation lasted for 30 seconds and the virtual human made gaze changes at regular 1.5 second intervals. The ordering of the animations was randomised for each participant and differed only according to the parameters passed to the blinking module: that is, head and eye movements were the exact same in every animation, only the blinking was altered.

7.4.1 Evaluation

Five different cases of blinking were tested: no blinking, regular unsynchronised, gaze-evoked with a blink probability of 1, probabilistic gaze-evoked, and a mixture of regular blinking and probabilistic gaze-evoked blinking. These cases are now explained in more detail:

1. No blinking: the character did not blink at all during the animation.
2. Regular unsynchronised: the character blinked according to a set interval of one blink every 3 seconds. Blinks were not synchronised in any way with gaze changes.
3. Gaze-evoked: the character only blinked when a gaze change took place. The probability of blink occurring with a gaze change was set to 1, so a blink was made for every gaze change event.
4. Probabilistic gaze-evoked: the character only blinked when a gaze change took place. The probability of blinks occurring with gaze changes as given in Section 6.3.1.
5. Probabilistic and regular: a mixture of the probabilistic gaze-evoked and regular cases. The character blinked both at regular intervals and when a gaze change took place with a probability according to that given in Section 6.3.1.

Each animation consisted of only one type of blinking.

7.4.2 Results and Analysis

The averaged results obtained over all six subjects are depicted in Figure 7.28. The case where the virtual human did not blink at all was rated the lowest by all participants, who generally agreed that the total lack of blinking was disconcerting; all participants noticed its absence.

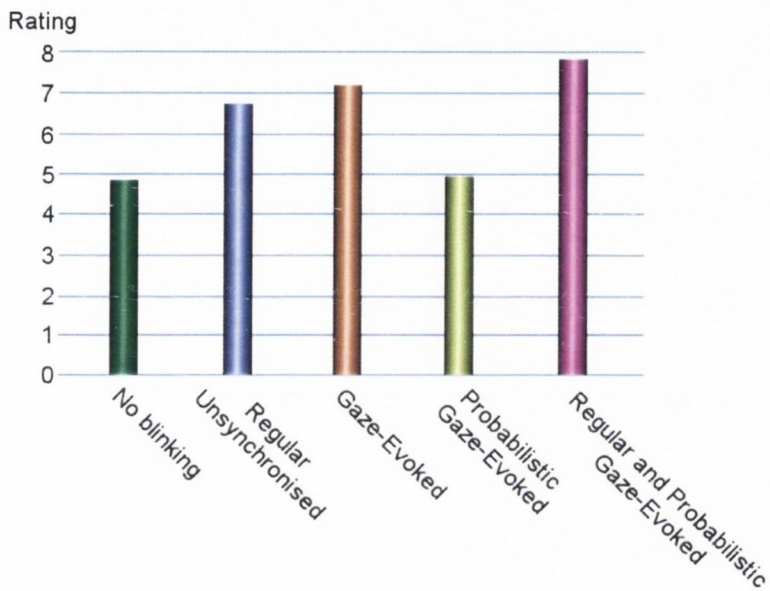


Figure 7.28: Averaged results of gaze and blink animation experiments. Participants were asked to rate a number of animations in terms of realism. Animations consisted for a virtual human making gaze-shifts between a number of target locations in the environment. The various blinking approaches listed above were used for each animation. A mixture of regular and probabilistic gaze-evoked blinking was rated as the most realistic method by participants.

The regular interval unsynchronised case produced a more interesting result, ranking as the third most plausible case. First of all, this seems to indicate that frequent blinking, in the order of one blink every 3 seconds, is an important component for plausible gaze motions. This is also outlined by the results of the case where no blinking took place. More interestingly, however, a number of participants noted that some of the blinks “didn’t feel natural”. This seems to indicate that frequent blinking is not the only factor leading to the perceived realism of gaze motions. In this case, blinks were not synchronised with head movements, so some blinks occurred at the start of, during, or at the end of a gaze motion or between gaze motions.

Participants ranked the gaze-evoked case as the second most realistic case, mainly due to the fact that they found the movement to be “synchronised better” and commented that the timing “feels more natural”. Despite this, the deterministic nature of the regular blink pattern, which took place every time a gaze change was invoked, detracted from the realism of the animation in a number of cases; as one participant mentioned, the blinks were “too regular, like a metronome”. This can be accounted for by the regularity at which the gaze changes took place. Since the blinking system is linked to a gaze change, if gaze changes take place at regular intervals and the blink probability is 1, then blinks will also appear to be programmed. The results for this case seem to indicate that synchronisation of blinking with gaze changes is desirable, but should be probabilistic in nature in order to avoid the impression of a robot-like agent.

Probabilistic gaze-evoked blinking was judged to be of low realism, with an overall score of 4.9, which seems surprising at first. However, this can be explained by the fact that, according to the formula in Section 6.3.1, for most low eccentricity gaze changes, there is a low probability that a blink will occur. Therefore, when probabilistic gaze-evoked blinking was used on its own, even though blinking was synchronised with gaze changes, very few blinks occurred throughout the entire 30 seconds of the animation. This led to a disconcerting impression similar to the case where no blinking took place. In this instance, it was not the synchrony of the blinks that participants found unrealistic, but the fact that there were few blinks. The results of this case indicate that probabilistic gaze-evoked blinking, as modelled in Section 6.3.1 and when used in isolation, does not provide plausible gaze motions due to the lack of frequency of blinking. This would be especially evident in systems where gaze changes of high eccentricity were not commonplace.

The final case, which mixed probabilistic gaze-evoked blinking with regular blinking, provided the most plausible results for the participants, with a score of 7.7. When considered with the results of the other cases, it would seem to provide three assertions:

1. Synchrony of blinking with gaze movements (i.e. gaze-evoked blinking) is an important factor for the plausibility of gaze animations.
2. Gaze-evoked blinking should be probabilistic in nature so as to avoid the impression of programmed or robotic motions.
3. Probabilistic gaze-evoked blinking alone is not enough to ensure plausibility. Normal, frequent blinks are a necessary component of plausible gaze motions.

Some of the limitations of the gaze generation system were also highlighted in the experiments, as five of the six participants recorded eye and head movements as being relevant factors during the blinking animations discussed earlier. Most of these comments centered on the timing of the eye movements with respect to the head and the neck and torso remaining stationary during some motions. We will now discuss both of these issues.

Recall from Section 6.2.1 that, given an initial head and eye configuration as well as a target location, the gaze generator will generate a final head and eye orientation based on head movement propensity. A spherical linear interpolation is then conducted from the current eye and head orientations to the final orientations in order to generate the animation. Firstly, this means that there are constant rotational velocities during the gaze movement and, secondly, that the eye and head movements begin and end at the same time. Both of these factors could be improved upon, as it is known that gaze motions do not consist of constant velocities and that the timing of eye and head movements is variable under a number of different circumstances. Indeed, in most cases, the eyes do not move gradually towards their target, but instead quickly acquire it and are locked onto it using the *vestibulo-ocular reflex*, or VOR, while the head continues towards its final destination. The above additions could be implemented with relative ease into the gaze model proposed here and would no doubt improve the plausibility of the resultant animations.

As mentioned in Section 6.2.1, a more elaborate model would be required to account for the contributions of other parts of the body in addition to the head

and eyes. The current model cannot make gaze motions towards objects outside the head movement range and the lack of contribution of the neck and torso is likely to be noticeable even for highly eccentric targets that are within the head movement range of the virtual human.

7.5 Discussion

This Chapter provided an evaluation of the results of the models presented in Chapters 4, 5 and 6. The system was evaluated in terms of speed, the gaze targets generated by the attention system and the plausibility of the gaze motions generated by the eye, head and blinking controllers.

In terms of the gaze and blinking controller, the motions that were generated were rated with high plausibility when blinks were frequent and were synchronous with gaze motions. Also, gaze motions were also rated higher when blinks did not occur with an obvious regularity. However, frequency and synchrony of blinking were not enough in isolation to ensure natural looking motions. When probabilistic gaze-evoked blinking was used in isolation, participants reported that the agent did not blink enough and when frequent blinking was used, participants reported that the blinks appeared to be unnatural when head motions were made. A mixture of regular and probabilistic gaze evoked blinking appeared to provide the best results in terms of both frequency and synchrony with gaze motions and was deemed to provide the most realistic results by the participants.

Given the nature of the speed timings taken for the attention system, it is evident that in environments containing a modest number of visible objects (< 256) the orientation section of the attention module consumes a significant amount of the processing time. Given that the orientation conspicuity calculations comprise over 80% of attention processing time, it would appear that orientation calculations are the main bottleneck in the system when the number of objects is within the 256 range. Therefore, it would be beneficial to investigate alternative methods of orientation conspicuity calculation that are not as time consuming or perhaps to decrease the amount of calculations undertaken in the current orientation processing model. Some authors have also taken the approach of disabling the orientation calculations altogether and only using intensity and colour for saliency calculation [HMYS01]. In our implementation, when orientation calculations are enabled for saliency map

generation, we distribute the attention processing over a number of frames (i.e. conduct time-slicing) in order to provide updates of the scene at interactive rates enabling a real-time visualisation of the simulation.

Disabling the calculation of certain feature dimensions is a viable approach since it has been found that image processing algorithms tend to be more successful at predicting fixations for certain types of scenes than for others (see [PS00b]). That is, one IPA may be quite successful at predicting scan paths for paintings, but may be a lot less successful at detecting scan paths in other classes of images, such as natural scenes. The challenge therefore seems to be in finding an automatic way to decide what IPAs will be the most effective for a given image and then employing those IPAs to generate the scan paths.

There are a number of issues to be considered when generating fixations during movement in a natural scene, as seen in Section 7.3.1. Evaluating gaze targets for a certain type of attention, in this case, attention during idle viewing, is difficult for a number of reasons: First of all, it has been shown that the unfamiliarity of a viewer with a scene affects their eye movements patterns. One has to be careful, therefore, to consider the affects of memory on the scene that is being viewed, as previous exposure to the scene may bias participants. Indeed, an interesting experiment may use abstract imagery to minimise the effects of memory and semantics on subject viewing patterns.

Secondly, a vitally important point is that the link between attention and eye-movements is not always clear and is difficult to establish. That is, when considering eye movements, a participant may be fixating on one area of the scene overtly, but may be paying covert attention to a different part of the scene. This difficulty can be attributed to the flexibility of the human attention system. In the attention module presented in Chapter 5, it leads to the practical consideration of how fixations should be extracted from the saliency map. This is because a given saliency map may have regions of interest extracted from it in a number of different ways. A rudimentary extraction method will directly select the first n maxima from the saliency map and directly specify these as aROIs. This is the approach that was adopted in this thesis. However, this type of extraction technique can lead to a large number of closely spaced fixation points, or the selection of salient local maxima in the image. Although this method is fast, it can be detrimental to the functioning of the system as it may result in fewer fixations to other areas of the image i.e. the selection of

globally salient locations that are suitable for comparison with human ROIs. As such, previous systems that use IPAs to generate aROIs have queried large numbers of local maxima from images. A clustering algorithm was then incorporated to form these local maxima into the final aROIs (see, for example, [PS00b] and [MD02] who use a clustering process to generate aROIs). It is noted that the clustering procedure is actually more of an eccentricity-weighting procedure that allows lower value regions to have the opportunity to be selected as aROIs. For speed reasons, the approach presented in Section 6.4.2 only generates a small number of maxima from the saliency map and uses these directly as aROIs: no clustering is performed. A more accurate system would undoubtedly incorporate clustering of larger numbers of maxima from the saliency map, but maintaining real-time performance under these conditions would be challenging.

It should also be noted that only bottom-up attention was studied in this thesis. Top-down, task and emotion related attention was not considered, but is also a vital component in determining where humans look. As mentioned however, even if a subject is not given a task and is told to look at a scene in any manner they wish, it is still possible that they will delegate themselves a task, whether it is intentional or not. This seems to say something about the possible nature of globally successful IPAs. As noted by Privitera and Stark [PS00b]:

“We suspect that the brain, with its enormous top-down approach, has the ability to synthesize image processing algorithms well beyond those that have been incorporated (by evolution) into the bottom-up mechanism of the retina and early vision cortical processing ... even those algorithms that do not have an intuitive and straightforward biological plausibility may be successful in predicting eye fixations.”

Indeed, even during idle viewing conditions, one could say that the viewer is considering the scene with the task of looking at the most interesting or informative locations. How *interesting* and *informative* are specified would appear to be the crux of the matter.

Chapter 8

Conclusions and Future Work

This section reviews the contributions made by this thesis to the domain of virtual human animation in terms of the synthetic vision, memory and attention models provided. Limitations of some of the adopted approaches are also discussed as well as possible avenues and improvements leading to intriguing future research in the area.

8.1 Contributions

This thesis presents a number of novel contributions to the domain of virtual human animation:

1. Enhance a current synthetic visual sensing technique in order to better account for visual perception while retaining computational flexibility. The visual sensor also allows for the amalgamation of object-based information, necessary for the memory component, with scene-based information from the attention component, to provide a system that supports both scene-based and object-based attention paradigms.
2. Present a synthetic memory model to filter and store visual information of relevance to the virtual human. The memory component further builds on the concept of successive filtering of environmental information and acts to modulate the output of the attention module. The filtering in the memory module is also controlled by the attention component, both directly and indirectly, in order to create a feedback loop.

3. Introduce a biologically plausible model of bottom-up selective visual attention from the field of cognitive engineering to the domain of virtual human animation in order to act as both a control mechanism for filtering in memory and as a gaze controller for head and eye movements.
4. Present a psychologically inspired model of gaze and blink coordination for gaze animations generated by the system. Eye and head contribution to the final motions are based on head movement propensity in order to provide variation in the animations, while blinking is gaze-evoked, with eyelid closure probability and magnitude dependent on head movement amplitude.

8.2 Limitations and Future Work

The system here is by no means intended to be a complete description of attention for virtual humans. Indeed, the broad scope of the research involved, encompassing a large body of work in psychology, makes it challenging to construct even simplified models of such systems. This thesis attempts to take a step in this direction by providing a starting point for demonstrating how such behaviours may be generated and by raising questions that have not been previously considered in this domain of work.

This section looks at limitations inherent in the approaches taken and highlights important questions, the answers to which, it is hoped, will contribute to future work in the area.

8.2.1 Synthetic Vision and Perception

One question that needs to be addressed is that of object recognition. In the synthetic vision module, if pixels relating to an object appear in the false colour rendering, then the object is automatically presumed to have been recognised. This is the case even when an object occupies only a single pixel in a false-colour rendering. As noted by Thalmann and Monzani [TM02], this is not an ideal situation. Determining when an object is recognised need not necessarily rely on complex approaches, such as those from the field of AI, but could be based on simplified approaches. For example, a simple probability metric for object recognition could be based on a mixture of pixel size thresholds and memory information about whether the object type or the object instance had been seen before.

The automatic perceptual grouping scheme detailed here could be improved to include other Gestalt grouping features, such as shape. Although the current system is automatic, it is also somewhat limited since it is calculated anew with each vision update since perceptual grouping is view dependent. The method presented in this thesis constitutes a first step towards a perceptual grouping system for virtual humans. Its main limitation is its volatility: frequent updates with view changes means that objects will tend to be associated with different groups from update to update. This is in contrast with the static grouping mode, where objects always remain in the same groups, but which is less realistic. Employing a grouping system where object groups remain more coherent over multiple frames is an important route for future research.

8.2.2 Synthetic Memory

The integration of automatic groups generated by the perceptual grouping system into memory could be improved. It is known that *chunking* of visual stimuli allows more items to fit into short-term memory (see Section 3.3.1). However, in terms of memory storage, it would be beneficial to investigate how grouping methods could reduce memory storage. For example, when looking at a haystack, storing a single group object as oppose to hundreds of individual grass objects would be preferable. This currently works with the manual grouping method since the groups produced remain static over multiple frames. It would be interesting to investigate a scheme that handles groups where members change dynamically, as with those produced by the automatic scheme.

Another area of important future research and one that is linked to perceptual grouping is the granularity at which objects are perceived. Currently, the lowest possible granularity is that defined by the geometry of the object in the geometry database. That is, the smallest perceptible unit will always correspond to the object's geometry, since false-colours are assigned on a per object basis. This means that a building object in the database will never be perceptible as a door, windows and wall objects and thus the virtual human will never have a separate memory of these components. One suggested approach to this sort of segmentation problem would be to introduce the ability to define sub-objects in a special object perception file. Although such a system would be manual, it would provide a fast solution to what otherwise is a complicated and time-consuming task. Such work could be

related to assigning attention-based attributes to objects using the smart-object paradigm (see Section 8.2.3).

The current memory system is object-based, in that it stores only a memory for object instances that have been previously sensed in the environment. However, it may be of interest to provide virtual humans with a storage of previously sensed object types, in order to allow virtual humans to distinguish between types of objects that they have and have not seen before. This may be particularly useful for deriving some form of novelty measurement so that objects in the scene that were not of a type previously sensed would elicit more attention.

8.2.3 Visual Attention

Perhaps the most significant addition to the work presented in this thesis would be support for some form of task-related, or *top-down* attention. While the system presented here is based on early visual processing in a bottom-up manner, the real system is thought to prescribe attention based on a combination of top-down and bottom-up methodologies. Such a system would represent a more complete model of attention and would impact many of the other modules here. For example, during a task, those objects relating to that task would be maintained in the STM a working set until task completion and would also receive enhanced attentive modulation. The model of visual attention presented by Itti [Itt00], and used in this thesis, has already been shown to be easily extendible to include notions of top-down, task-based, attentional control [NI02]. In addition, this author has already contributed to suggesting how object-based information may be used to support a virtual human's attentive processes during tasks for the purposes of behavioural animation [PDMS03].

Another important area of future work involves finding suitable ways of speeding up the visual attention algorithm. It is possible that other less computationally expensive algorithms may provide suitable results (see [PS00b] for example algorithms). Other shortcuts may also be taken once object-based information is available due to a synthetic vision resolution, as described in this thesis. A good example of this is querying actual object velocity information from the scene database rather than using an optical flow approach when calculating attention to motion.

The current model does not differentiate between covert and overt targets of attention. As mentioned in Section 3.2, we may pay attention to different locations

in our visual field without necessarily orienting towards them using eye or head movements. In this thesis, we simplify current theories by presuming a more direct linkage between covert attention locations output by the attention models and the target locations that correspond to overt orienting through eye and head movements. In reality, the connection between covert and overt attention would appear to be less direct, although the nature of the association between covert attention and eye movements is still an area of active research and debate.

Time varying stimuli, such as flicker, have been accounted for in the updated model outlined by Itti [Itt00] and provide important contributions to bottom-up attention. These features can be added to our system with minor modifications. Perhaps the most significant future improvement will be an integrated attention model featuring a top-down attention component. This could allow us to consider subtle factors such as object novelty and task relevance when planning gaze motions. A simple first approach may be to use the long-term memory to store a field representing whether the agent had encountered an object-type before. Objects that had not been encountered before may elicit more attention. Task-relevant attention could be achieved by increasing the importance of certain object types. More work would be beneficial on the management of the saliency map generator in order to provide a more scalable system. In our implementation, the computation of the final orientation conspicuity map took on average 82.2% of the total calculation time for the saliency map. There are perhaps cases where only an intensity channel and colour channel computation would suffice.

8.2.4 Attentive Behaviours

There are a number of ways in which the current gaze model could be improved. First of all, the gaze model does not take the timing or velocity of gaze motions into account; interpolation is conducted in such a way that the velocity of the head is always constant. This means that the initial movement of the head can be somewhat abrupt, as can the final movement when the head comes to a standstill. Experimental research has been conducted into the timing of head movements with respect to eye movements and of particular interest is the connection between eye linking and maximum head velocity [EMP⁺94]. Producing proper head timings and velocities, as well as introducing interaction between the head and blinking for such factors, would lead to considerable improvement in the current model.

Another limiting factor of the current head model is that it is unable to track moving objects. That is, the gaze algorithm (see Algorithm 4) is only executed once after being provided with a target position. If the target position changes after gaze calculation, the current gaze model will fail to track it. In relation to tracking objects, interesting future work would also consider the contributions of other bones to the gaze motion. The current model only considers the contributions of the head and eyes, but more realistic orienting movements would be obtained by also considering, for example, the spine.

In terms of creating behaviours linked to uncertainty reduction and exploration, *inspective behaviour* is only one type of exploratory behaviour identified by Berlyne [Ber71]. Other behaviours, such as *diversive behaviour* and *affective exploration*, are also detailed. These behaviours are thought to arise out of a competence of certainty of one's environment and concern boredom relief, philosophising and play. Behaviours are thought to be prioritised in the order of inspective, diversive and affective. That is, one is most likely to perform inspective behaviour before diversive and diversive before affective. These types of behaviours may be useful for providing a controller for different behaviour types. For example, a character might only engage in play or internal processing behaviours if the attention map was below a certain threshold (i.e. there weren't any particularly interesting external stimuli present).

An interesting route for future work concerns attempts to bridge the gap between attention in natural environments and attention for social interactions. A key question that has not received a great deal of research is how conversational situations arise in the first place. For example, a group of virtual humans are engaging in conversational behaviour. But how did they meet in the first place? An interesting possibility would be to allow virtual humans to recognise each other and use this as the impetus for early social interaction behaviour. By augmenting the systems proposed in this thesis (i.e. sensing, memory and attention) with special maps for discriminating faces and figures from the rest of the scene, virtual humans could be made more social in the sense that they would pay more attention to other virtual humans. This would allow for scenarios where virtual humans may recognise colleagues and attempt to gain *their* attention in an effort to initialise social contact. If social contact was successful, then conversational attention systems could then take over the attention process (for example, see [PPdR00]).

Related Publications

1. Attention-Driven Eye Gaze and Blinking for Virtual Humans. C. Peters and C. O'Sullivan, SIGGRAPH 2003 Sketches Program, San Diego, July 2003.
2. Bottom-up Visual Attention for Virtual Human Animation. C. Peters and C. O'Sullivan, Proceedings of the Computer Animation and Social Agents (CASA) 2003, Rutgers University, New York pp 111-117, May 2003.
3. Smart Objects for Attentive Agents. C. Peters, S. Dobbyn, B. MacNamee, and C. O'Sullivan, Proceedings of the International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, Plzen, Czech Republic, 2003.
4. Synthetic Vision and Memory for Autonomous Virtual Humans. C. Peters and C. O'Sullivan, Computer Graphics Forum, 21(4) pp 743-75, 2002.
5. Levels of Detail for Crowds and Groups. C. O'Sullivan, J. Cassell, H. Vilhjmsson, J. Dingliana, S. Dobbyn, B. McNamee, C. Peters, and T. Giang, Computer Graphics Forum, 21(4) pp 733-742, 2002.
6. Crowd and Group Simulation with Levels of Detail for Geometry, Motion and Conversational Behaviour. C. O'Sullivan, J. Cassell, H. Vilhjmsson, S. Dobbyn, C. Peters, W. Leeson, T. Giang, and J. Dingliana, Eurographics Irish Chapter Workshop Proceedings 2002, pp 15-20, 2002.
7. A Memory Model for Autonomous Virtual Humans. C. Peters and C. O'Sullivan, Eurographics Irish Chapter Workshop Proceedings 2002, pp 21-26, 2002.
8. Vision-Based Reaching for Autonomous Virtual Humans. C. Peters and C. O'Sullivan, Proceedings of the AISB'02 symposium: Animating Expressive Characters for Social Interactions, Imperial College, London, pp 69-72, 2002.
9. ALOHA: Adaptive Level Of Detail for Human Animation: Towards a new framework. T. Giang, R. Mooney, C. Peters, and C. O'Sullivan, Eurographics 2000 short paper proceedings, Interlaken, Switzerland, pp 71-77, 2000.

Bibliography

- [AA82] A.J. Afanador and A.P. Aitsebaomo. The range of eye movements through progressive multifocals. *Optometry Monographs*, 73:82–88, 1982.
- [AC76] M. Argyle and M. Cook. *Gaze and Mutual Gaze*. Cambridge University Press, Cambridge, 1976.
- [And95] J.R. Anderson, editor. *Cognitive psychology and its implications*. Freeman, New York, 1995.
- [AS68] R.C. Atkinson and R.M. Shiffrin. Human memory: a proposed system and its control processes. *The psychology of learning and motivation: advances in research and theory*, 2:89–195, 1968.
- [Bad97] N.I. Badler. Real-time virtual humans. pages 4–14, 1997.
- [BB96] E.L. Bjork and R.A. Bjork. *Memory*. Academic Press, Hove, 1996.
- [BBT99] C. Bordeaux, R. Boulic, and D. Thalmann. An efficient and flexible perception pipeline for autonomous agents. *Computer Graphics Forum*, 18(3):23–30, 1999.
- [BBZ91] N. I. Badler, B. A. Barsky, and D. Zeltzer. *Making Them Move: Mechanics, Control, and Animation of Articulated Figures*. Morgan Kaufmann, San Mateo, CA, 1991.
- [Ber60] D.E. Berlyne. *Conflict, Arousal and Curiosity*. McGraw Hill, New York, 1960.
- [Ber71] D.E. Berlyne. *Aesthetics and Psychobiology*. Appleton-Century-Crofts, New York, 1971.

- [BFP91] C. Bard, M. Fleury, and J. Paillard. *Different patterns in aiming accuracy for head-movers and non-head movers*. Oxford University Press, Oxford, 1991.
- [BG90] V. Bruce and R.G. Green. *Visual Perception: Physiology, Psychology and Ecology*. Lawrence Erlbaum Associates, Hove and London, 1990.
- [Blu97] B. Blumberg. Go with the flow: Synthetic vision for autonomous animated creatures. In W. Lewis Johnson and Barbara Hayes-Roth, editors, *Proceedings of the First International Conference on Autonomous Agents (Agents'97)*, pages 538–539, New York, 5–8, 1997. ACM Press.
- [BPW93] N. I. Badler, C. B. Phillips, and B. L. Webber. *Simulating Humans: Computer Graphics Animation and Control*. Oxford University Press, New York, NY, Oxford, 1993.
- [Cha67] M.R.A. Chance. Attention structure as the basis of primate rank orders. *Experimental Brain Research*, 2:503–518, 1967.
- [CK99] S. Chopra-Khullar. *Where to look? Automating certain visual attention behaviors of human characters*. PhD dissertation, University of Pennsylvania, January 1999.
- [CMA03] N. Courty, E. Marchand, and B. Arnaldi. A new application for saliency maps: Synthetic vision of autonomous actors. IEEE Int. Conf. On Image Processing, Icip'03, 2003.
- [CT99] J. Cassell and K.R. Thórisson. The power of a nod and a glance: envelope vs. emotional feedback in animated conversational agents. *Applied Artificial Intelligence*, 13(4):519–538, 1999.
- [CTP99] J. Cassell, O. Torres, and S. Prevost. Turn taking vs. discourse structure: how best to model multimodal conversation. In Yorick Wilks, editor, *Machine Conversations*, pages 143–154. Kluwer, The Hague, 1999.
- [CV99] J. Cassell and H. Vilhjálmsón. Fully embodied conversational avatars: Making communicative behaviors autonomous. *Autonomous Agents and Multi-Agent Systems*, 2(1):45–64, 1999.

- [CVB01] J. Cassell, H. Vilhjálmsón, and T. Bickmore. BEAT: The behavior expression animation toolkit. In Eugene Fiume, editor, *SIGGRAPH 2001, Computer Graphics Proceedings*, pages 477–486. ACM Press / ACM SIGGRAPH, 2001.
- [Dun84] J. Duncan. Selective attention and the organization of visual information. *Journal of Experimental Psychology: General*, 113(4):501–517, 1984.
- [Ebb13] H. Ebbinghaus. *Über das Gedächtnis*, 1913.
- [EK00] M.W. Eysenck and M.T. Keane. *Cognitive Psychology: A Students Handbook*. Psychology Press, Hove, 2000.
- [EM02] T.F. Evers and S.R. Musse. Building artificial memory to autonomous agents using dynamic and hierarchical finite state machine. In *Proc. of Computer Animation 2002*, pages 164–169, June 2002.
- [EMP⁺94] C. Evinger, K. Manning, J. Pellegrini, M. Basso, A. Powers, and P. Sibony. Not looking while leaping: the linkage of blinking and saccadic gaze shifts. *Experimental Brain Research*, 100(2):337–344, 1994.
- [EY69] R.V. Exline and A. Yellin. Eye contact as a sign between man and monkey, 1969.
- [EY87] C.W. Erikson and Y.Y. Yeh. Allocation of attention in the visual field. *Journal of Experimental Psychology: Human Perception and Performance*, 11:583–597, 1987.
- [FE91] D.J. Felleman and D.C. Van Essen. Distributed hierarchical processing in the primate cerebral cortex. *Cerebral Cortex*, 1:1–47, 1991.
- [FJBS00] T. Farroni, M.H. Johnson, M. Brockbank, and F. Simion. Infants’ use of gaze direction to cue attention: The importance of perceived motion. *Visual Cognition*, 7(6):705–718, 2000.
- [FTT99] J. Funge, X. Tu, and D. Terzopoulos. Cognitive modeling: Knowledge, reasoning and planning for intelligent characters. In Alyn Rockwood, editor, *Siggraph 1999, Computer Graphics Proceedings*, pages 29–38, Los Angeles, 1999. Addison Wesley Longman.

- [Ful92] J.H. Fuller. Head movement propensity. *Experimental Brain Research*, 92:152–164, 1992.
- [Fun98] J. Funge. *Making Them Behave: Cognitive Models for Computer Animation*. PhD thesis, 1998.
- [Gib79] J. J. Gibson. *The Ecological Approach to Visual Perception*. Mifflin, Boston, 1979.
- [Gil01] M. Gillies. *Practical behavioural animation based on vision and attention*. PhD dissertation, University of Cambridge Computer Laboratory, 2001.
- [Gol96] E.B. Goldstein, editor. *Sensation and perception*. Wadsworth, 1996.
- [GS84] G. Gillund and R.M. Shiffrin. A retrieval model for both recognition and recall. *Psychological Review*, 91:1–67, 1984.
- [Gui92] D. Guitton. Control of eye-head coordination during orienting gaze shifts. *Trends In Neuroscience*, 15(5):174–179, 1992.
- [Hec98] E. Hecht, editor. *Optics*. Addison-Wesley, Massachusetts, third edition, 1998.
- [Hil99] R. Hill. Modeling perceptual attention in virtual humans. In *Proc. of the Eighth Conference on Computer Generated Forces and Behavioral Representation*, May 1999.
- [HMYS01] J. Haber, K. Myszkowski, H. Yamauchi, and H.P. Seidel. Perceptually guided corrective splatting. *Computer Graphics Forum*, 20(3), September 2001.
- [IDP03] L. Itti, N. Dhavale, and F. Pighin. Realistic avatar eye and head animation using a neurobiological model of visual attention. In *Proc. SPIE 48th Annual International Symposium on Optical Science and Technology*, Aug 2003.
- [IK99] L. Itti and C. Koch. A comparison of feature combination strategies for saliency-based visual attention systems. In *Proc. SPIE Human Vision and Electronic Imaging IV (HVEI'99)*, San Jose, CA, volume 3644, pages 473–82, Jan 1999.

- [IK00] L. Itti and C. Koch. A saliency-based search mechanism for overt and covert shifts of visual attention. *Vision Research*, 40(10-12):1489–1506, May 2000.
- [IK01a] L. Itti and C. Koch. Computational modeling of visual attention. *Nature Reviews Neuroscience*, 2(3):194–203, Mar 2001.
- [IK01b] L. Itti and C. Koch. Feature combination strategies for saliency-based visual attention systems. *Journal of Electronic Imaging*, 10(1):161–169, Jan 2001.
- [IKN98] 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 (PAMI)*, 20(11):1254–1259, 1998.
- [Itt00] L. Itti. *Models of Bottom-Up and Top-Down Visual Attention*. PhD thesis, Pasadena, California, Jan 2000.
- [Jac95] R. Jacob. Eye tracking in advanced interface design, 1995.
- [Kar88] C. Karson. Physiology of normal and abnormal blinking. *Advances in Neurology*, 49:25–37, 1988.
- [KL99] J. Kuffner and J.C. Latombe. Fast synthetic vision, memory, and learning for virtual humans. In *Proc. of Computer Animation*, May 1999.
- [KMCT00] M. Kallmann, J.S. Monzani, A. Caicedo, and D. Thalmann. Ace: A platform for the real time simulation of virtual human agents, 2000.
- [KU85] C. Koch and S. Ullman. Shifts in selective visual attention: towards the underlying neural circuitry. *Human Neurobiology*, 4:219–227, 1985.
- [Kuf99] J. Kuffner. *Autonomous Agents for Real-time Animation*. PhD dissertation, Stanford University, 1999.
- [LBB02] S.P. Lee, J.B. Badler, and N.I. Badler. Eyes alive, 2002.
- [MD02] G. Marmitt and A.T. Duchowski. Modeling visual attention in vr: Measuring the accuracy of predicted scanpaths. *Eurographics 2002 Short Papers Programme*, 2002.

- [Mil56] G.A. Miller. The magic number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological Review*, 63:81–97, 1956.
- [Moo01] G. Moore. Talking heads: Facial animation in the getaway. Gamasutra Feature, April 2001.
- [Nei67] U. Neisser, editor. *Cognitive Psychology*. Appleton-Century-Crofts, New York, 1967.
- [Nem93] A. Nemcsics. *Color Dynamics*. Akademiai Kiad, Budapest, 1993.
- [NI02] V. Navalpakkam and L. Itti. A goal oriented attention guidance model. *Lecture Notes in Computer Science*, 2525:453–461, Nov 2002.
- [NK98] E. Niebur and C. Koch. Computational architectures for attention, 1998.
- [NRTT95] H. Noser, O. Renault, D. Thalmann, and N.M. Thalmann. Navigation for digital actors based on synthetic vision, memory and learning. *Computer and Graphics*, 19(1):7–19, 1995.
- [NS71] D. Noton and L.W. Stark. Scanpaths in eye movements during pattern perception. *Science*, 171:308–311, 1971.
- [Par00] A.J. Parkin, editor. *Essential Cognitive Psychology*. Psychology Press, 2000.
- [Pas98] H.E. Pashler, editor. *The psychology of attention*. MIT Press, Cambridge, Massachusetts, 1998.
- [PDMS03] C. Peters, S. Dobbyn, B. MacNamee, and C. O’ Sullivan. Smart objects for attentive agents. *Proceedings of the International Conference in Central Europe on Computer Graphics*, 2003.
- [PNK76] M.I. Posner, M.J. Nissen, and R.M. Klein. Visual dominance: an information-processing account of its origins and significance. *Psychological Review*, 83:151–171, 1976.
- [PPdR00] I. Poggi, C. Pelachaud, and F. de Rosi. Eye communication in a conversational 3d synthetic agent. *AI Communications*, 13(3):169–182, 2000.

- [PS00a] A. Pouget and L.H. Snyder. Computational approaches to sensorimotor transformations. *Nature Neuroscience supplement*, 3:1192–1198, 2000.
- [PS00b] C.M. Privitera and L.W. Stark. Algorithms for defining visual regions of interest: Comparison with eye fixations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(9):970–982, 2000.
- [Rey87] C.W. Reynolds. Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics*, 21(4):25–34, 1987.
- [RTT90] O. Renault, D. Thalmann, and N.M. Thalmann. A vision-based approach to behavioural animation. *Journal of Visualization and Computer Animation*, 1(1):18–21, 1990.
- [SB75] M. Scaife and J.S. Bruner. The capacity for joint visual attention in the infant. *Nature*, 253:265–266, 1975.
- [Sch01] B.J. Scholl. Objects and attention: The state of the art. *Cognition*, 80:1–46, 2001.
- [SF89] J.F. Soechting and M. Flanders. Errors in pointing are due to approximations in sensorimotor transformations. *Journal of Neurophysiology*, 62(2):595–608, 1989.
- [Sho85] K. Shoemake. Animating rotation with quaternion curves. In Eugene Fiume, editor, *SIGGRAPH 2001, Computer Graphics Proceedings*, volume 19, pages 245–254. ACM Press / ACM SIGGRAPH, 1985.
- [SLT86] J.F. Soechting, F. Lacquaniti, and C.A. Terzuolo. Coordination of arm movements in three-dimensional space. *Neuroscience*, 17:295–311, 1986.
- [Spe60] G. Sperling. The information available in brief visual presentations. *Psychological Monographs*, 74(498), 1960.
- [TCW⁺95] J. K. Tsotsos, S. M. Culhane, W. K. Wai, Y. Lai, N. Davies, and F. Nuflo. “Modeling visual attention via selective tuning”. *Artificial Intelligence*, 78:507–545, 1995.
- [TG80] A. Treisman and G. Gelade. A feature integration theory of attention. *Cognitive Psychology*, 12:97–136, 1980.

- [Thó94] K. Thórisson. Simulated perceptual grouping: An application to human computer interaction, 1994.
- [TM02] D. Thalmann and J.S. Monzani. Behavioural animation of virtual humans: What kinds of laws and rules? *Proceedings of Computer Animation 2002*, pages 144–153, 2002.
- [TT94] X. Tu and D. Terzopoulos. Artificial fishes: Physics, locomotion, perception, behavior. *Computer Graphics*, 28(Annual Conference Series):43–50, 1994.
- [Wal42] G.L. Walls. *The Vertebrate Eye and Its Adaptive Radiation*. The Cranbrook Press, Michigan, 1942.
- [Wel93] C. Welman. *Inverse Kinematics and Geometric Constraints for Articulated Figure Manipulation*. MSc dissertation, University of Cambridge Computer Laboratory, 1993.
- [WS82] G. Wyszecki and W.S. Stiles. *Color Science, Concept and Methods, Quantitative Data and Formulae*. John Wiley and Sons, New York, 1982.
- [WZS95] K. Sherman W.H. Zangemeister and L. Stark. Evidence for a global scanpath strategy in viewing abstract compared with realistic images. *Neuropsychologia*, 33(8):1009–1025, 1995.
- [Yan98] S. Yantis. *Control of Visual Attention*, pages 223–256. Attention. Psychology Press, London, 1998.
- [Yar67] A.L. Yarbus. *Eye Movements and Vision*. Plenum Press, New York, 1967.
- [YPG01] H. Yee, S. Pattanaik, and D.P. Greenberg. Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. *ACM Transactions on Graphics (TOG)*, 20(1):39–65, 2001.