Influencing User Perception Using Real-Time Adaptive Abstraction

by

Niall Redmond, B.A. (Mod)

Dissertation

Presented to the

University of Dublin, Trinity College

in fulfillment

of the requirements

for the Degree of

Doctor of Philosophy

University of Dublin, Trinity College

September 2010

Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work.

Niall Redmond

April 18, 2011

Permission to Lend and/or Copy

I, the undersigned, agree that Trinity College Library may lend or copy this thesis upon request.

Niall Redmond

April 18, 2011

Acknowledgments

Firstly, thanks to my supervisor John Dingliana, for all his help, encouragement and support. This work would not have been possible without him.

Secondly, I would like to thank to my friends and colleagues in GV2, past and present, for making my time there thoroughly enjoyable and memorable. Thanks to Andrew, his collaboration in my later work was a huge help.

Finally, I would like to thank to my Mum, Dad and Niamh. I don't know what I would do without their support, interest and encouragement, in this work and in everything I do.

Niall Redmond

University of Dublin, Trinity College
September 2010

Abstract

Real-time applications such as games, medical or technical visualisations and urban simulations can often be highly complex in nature. This can lead to too much visual data being presented to the user at once, which can make scenes cluttered and difficult to interpret. Complex scenes can be particularly problematic when it is required to highlight certain scene data to the user, as is often necessary in visualisations and interactive scenes. This thesis describes research towards developing a solution for this problem by creating perceptually optimised approaches using non-photorealistic rendering (NPR).

NPR is a research area within computer graphics which is driven by the desire for both aesthetic stylisation and effective communication. The majority of existing non-photorealistic research focuses on creating stylistic and convincingly artistic renderings. Other NPR techniques borrow from the artistic world to use stylistic rendering to draw a user's attention to certain parts of an image while reducing the impact of less important areas. This can be done effectively by using multiple levels of abstraction within an image. We use such techniques in real-time scenes to create renderings which successfully influence user behaviour in interactive scenes.

In this thesis, we explore effective real-time multi-level abstraction methods for interactive scenes. We examine a variety of stylistic approaches, to find which ones are suitable for real-time scenes and which can be altered to create different levels of abstraction. We propose an adaptive abstraction approach, which can be used across scenes to emphasise particular objects and influence user perception. A number of non-photorealistic abstraction techniques exist which can be defined as image-space or object-space approaches. We implement the most suitable styles for the proposed adaptive abstraction approach, while

using the strengths of both image-space and object-space techniques to retain important perceptual scene cues.

We investigate how a variety of non-photorealistic styles can affect user perception of real-time scenes through a number of user experiments. We show how adaptive abstraction can be used as an effective tool in facilitating user guidance and understanding in scenes by examining a number of aspects of user perception such as eye-gaze behaviour and shape perception. We also investigate how adaptive abstraction can affect task performance in interactive scenes. We present a number of guidelines, learned from perceptual experiments, on how adaptive abstraction can be best used in varying contexts.

We produce adaptive non-photorealistic styles, which could be smoothly integrated into a traditionally modelled environment. This allows for the adaptive abstraction approach to be easily incorporated into any existing application where particular scene data needs to be highlighted to users. We show how the adaptive abstraction approach can be useful for a variety of applications, including volume data. Volume visualisation is an example of a type of application that suffers from an excessive amount of data being simultaneously presented to a user. This problem can make the clear presentation of volume data a difficult task. The adaptive abstraction approach suits this type of visualisation as we aim to simplify complex scenes to focus on particular parts of the scene to make clear, stylised renderings.

Related Publications:

1. **Perceptual Enhancement of Two-Level Volume Rendering**: Andrew Corcoran, Niall Redmond and John Dingliana; Computers & Graphics, Volume 34, issue 4, pp. 388-397, 2010

2. **A Hybrid Approach to Real-Time Abstraction**: Niall Redmond and John Dingliana; Eurographics Ireland '09: 9th Irish Workshop on Computer Graphics, pp. 43-49, 2009

3. **Investigating the Effect of Real-Time Stylisation Techniques on User Task Performance**: Niall Redmond and John Dingliana; APGV '09: 6th symposium on Applied Perception in Graphics and Visualization, pp. 121-124, 2009

4. **Influencing User Attention Using Real-Time Stylised Rendering**: Niall Redmond and John Dingliana; TPCG '09: Theory and Practice of Computer Graphics, pp. 173-180, 2009

5. **A Hybrid Technique for Creating Meaningful Abstractions of Dynamic 3D Scenes in Real-Time**: Niall Redmond and John Dingliana; WSCG '08: International Conferences in Central Europe on Computer Graphics, Visualization and Computer Vision, pp. 105-112, 2008

6. **Adaptive Abstraction of 3D Scenes in Real-Time**: Niall Redmond and John Dingliana; Eurographics '07: Short Presentations, pp. 77-80, 2007

Related Posters:

1. **Evaluating Stylistic Enhancement of Two-Level Volume Renderings**: Niall Redmond, Andrew Corcoran and John Dingliana; CA '10: Sixth International Symposium on Computational Aesthetics in Graphics, Visualization and Imaging, Poster Session, 2010

2. **Evaluation of Non-Photorealistic Abstraction Techniques in Influencing User Behaviour**: Niall Redmond and John Dingliana; APGV '08: 5th symposium on Applied perception in graphics and visualization, Poster Session, 2008

3. **Adaptive Real-Time Abstractions of Large Urban Simulations**: Niall Redmond and John Dingliana; NPAR '07: 5th International Symposium on Non-Photorealistic Animation and Rendering, Poster Session, 2007

Contents

Acknowledgments **iv**

List of Tables **xii**

List of Figures **xiii**

Chapter 1 Introduction **1**

1.1 Motivation 2
1.2 Methodology 3
1.3 Scope 3
1.4 Contributions 4
1.5 Summary of Chapters 5

Chapter 2 Background and Related Work **6**

2.1 NPR Techniques 7
2.1.1 Painterly Rendering 8
2.1.2 Edge Detection 12
2.1.3 Artistic Shading 15
2.2 Data Types 17
2.2.1 Polygonal Data 17
2.2.2 Volumetric Data 18
2.2.3 Images and Video 19
2.2.4 Multidimensional Datasets 20
2.3 Application Fields 20
2.4 Meaningful Abstraction 23
2.5 Evaluation of NPR 27
2.5.1 Psychology of NPR 27
2.5.2 Observational Studies 28

2.5.3 Timed User Tasks 29
2.5.4 Space Perception 29
2.5.5 Shape Perception 29
2.5.6 Eye-Tracking 30
2.5.7 Saliency measurement 31
2.6 Discussion 32

Chapter 3 Adaptive Abstraction **33**
3.1 Introduction 33
3.2 Real-Time Styles 34
3.2.1 Painterly Rendering 35
3.2.2 Edge Detection 37
3.2.3 Shading Stylisation 39
3.3 Scene Segmentation 40
3.4 Combining Styles 43
3.4.1 Example Effect 1: Hybrid Cel-Shading 45
3.4.2 Example Effect 2: Hybrid Pencil-Shading 46
3.5 Creating an Adaptive Abstraction 47
3.5.1 Multi-level abstraction 47
3.5.2 Stylised Focus 49
3.5.3 Frame Times 52
3.6 Summary and Conclusion 54

Chapter 4 Perceptual Evaluation **55**
4.1 Introduction 55
4.2 Experimental Setup 56
4.3 Automatic Saliency Tests 58
4.4 Two-Alternative Forced Choice Task 62
4.5 Simple Context Experiments 64
4.5.1 Rendering Styles 65
4.5.2 Timed search experiments 66
4.5.3 Eye-tracking Experiments 75
4.5.4 Discussion 80
4.6 Matching Pairs Experiment 81
4.6.1 Rendering Techniques 82
4.6.2 Experiment Framework 84
4.6.3 Results 85

4.6.4 Discussion 89
4.7 Interactive Search Experiment 90
4.7.1 Rendering Techniques 91
4.7.2 Experiment Framework 92
4.7.3 Results 94
4.8 Summary and Conclusion 95

Chapter 5 Case Study: Volume Visualisation System 98
5.1 Introduction 98
5.2 Segmented Volume Data 99
5.3 Two-Level Volume Rendering 104
5.3.1 Volume Rendering Techniques 105
5.3.2 Non-Photorealistic Rendering Techniques 106
5.4 Evaluation 108
5.4.1 User Study 108
5.4.2 Results and Analysis 110
5.4.3 Saliency Tests 114
5.5 Summary and Conclusion 119

Chapter 6 Conclusions and Future Work 121
6.1 Summary of Contributions 121
6.1.1 New techniques 121
6.1.2 New Experimental Results 121
6.1.3 New Guidelines 123
6.1.4 Example Applications 124
6.2 Reflections on Experimental Design 124
6.2.1 Timed Search Tasks 124
6.2.2 Eye-Tracking Data 125
6.2.3 Shape Perception Experiments 125
6.2.4 Experiment Conditions 125
6.3 Future Work 126

Bibliography 128

List of Tables

3.1 Frames per second achieved by the system using various abstraction options on a complex urban simulation (over 120,000 polygons) at different resolutions. These frame times were calculated using a Pentium D, 3.72GHz computer with 2GB of RAM using an NVidia GeForce 7950 GX2 GPU . . 53

4.1 List of each experiment in the chapter, showing the section it is in, the task given to the participant, the styles tested and the reason for the experiment 57

4.2 Description of the various styles used in Experiment one, within the simple context experiments. 67

4.3 Description of the various combination styles used in Experiment two, within the simple context experiments. 72

4.4 Description of the various combination styles used in Experiment three, within the simple context experiments. 74

5.1 Description of the various styles used in the shape perception experiment. . 111

List of Figures

2.1 Example of the painterly approach proposed by Hertzmann which uses curved brushstrokes of various sizes [Her98] 8
2.2 Examples of image-based painterly filters. Left: Original image, Centre: Image created by Kuwahara filter, Right: Image created by using a more detailed, but slower filter as described by Kyprianidis et al. [KKD09]. Note that there is more fine detail preserved and fewer clustering artefacts using the approach by Kyprianidis (right image). 11
2.3 A number of edge-detection approaches, as described in [DR07]. Left: Contours and Suggestive Contours combined, Centre: Contours, Suggestive Contours and Suggestive Highlights, Right: Image-Space approach to finding Suggestive Contours. As can be seen, object space approaches such as suggestive contours and suggestive highlights add important shape detail to the image while image-space approaches do not represent object edges as accurately. 13
2.4 Cel-shading exhibited on a model, as described in [LMHB00] 16
2.5 Example of how volumetric data can be visualised using non-photorealistic techniques. The right image shows a traditional image of blood vessels while the left image uses silhouettes to help clarify the spatial relationship between vessels [LM02]. 18
2.6 Image from Healey et al. where weather over South America is visualised using brushstrokes. Each brushstroke property is linked to an attribute of the dataset; Wind speed - Hue, Temperature - Luminance, Radiation - Size, Precipitation - Orientation [THE07] 22
2.7 Example image from Santella and DeCarlo, who produce painterly renderings by creating coloured regions of varied size depending on the visual importance of each area [DS02] 25

2.8 Image showing a stylised focus on the top left part of the scene. Multiple levels of abstraction are implemented to focus the user of certain areas [CDF+06] 26
2.9 Image from a study which explored the relationship between NPR and psychology. Users tended to choose the more detailed routes with line drawing as more interesting for exploration [HMH+03a] 28
2.10 Example of the results from the shape perception study in [CSD+09]. The gauges displayed show how well user's estimated the surface normal at that point, coloured by how far they deviate from the ground truth (Left: contours, Right: apparent ridges) 30

3.1 Example of the layout of a 5x5 Kuwahara filter containing 3x3 sampled regions. Each of the four regions is indicated using a separate colour 36
3.2 Edge detection examples; Difference of Gaussian (Left), Sobel (Centre), Canny (Right) 37
3.3 Various types of abstraction; (a) Original rendering of scene (b) Kuwahara filter (c) Pencil-shading (d) Cel-shading. As can be seen, the shading styles do not take into account colour and texture information, as the painterly approach does. 41
3.4 Example of the false colour pass; Left: Original rendering. Right: False colour rendering 41
3.5 Edge detection example: 3D Silhouette edge image, from false colour rendering pass (Left), Image space edge image (Middle) and the two overlaid on the original image to display all scene edges (Right) 42
3.6 Workflow for combining object-space shading and image-space abstraction and edges. The example cel-shading effect is used to demonstrate the workflow. 44
3.7 Examples demonstrating the advantage of the hybrid edge detection approach;(a) Scene rendered using Sobel image-space edges (b) Scene rendered using both image-space edges and silhouette edges found using unique colour ID rendering pass 45
3.8 Abstraction examples: (a) Hybrid cel-shading effect (b) Hybrid pencil-shading effect 46
3.9 A section of an image when there is no abstraction(top left), a 3x3 Kuwahara filter(top right), a 3x3 Kuwahara filter applied twice(bottom left) and a 5x5 Kuwahara filter(bottom right) 48

3.10 An example of a scene with object-based segmentation. One object is rendered clearly while the rest of the scene is abstracted. The Kuwahara filter, difference of Gaussian edges and saturation variation were used in this image (two levels of abstraction). 50
3.11 Example of stylised focus without scene segmentation information. Left: original rendering, Right: a 2D distance based abstractive rendering using difference of Gaussian edges and brightness variation (8 levels of abstraction). 50
3.12 Adaptive abstraction examples. (a) Two-level adaptive abstraction using Kuwahara filter, with focus on the character in the background wearing stripes (b) Multi-level adaptive abstraction using adaptive hybrid pencil shader, using object-space distance as a guide for abstraction. The focus is on the character in yellow in the foreground 52
3.13 Examples of three of five abstraction levels using the Kuwahara filter, Difference of Gaussian edges and brightness variation. Left (level 1, most salient), Middle (level3) and Right (level 5) 52

4.1 Example of two scene images with and without abstraction. The yellow markers indicate the most salient points of each image, as indicated by the saliency metric, described in [IKN98]. In each stylised image the most salient point is found to be the object which is focused on using adaptive abstraction. The right images use the Kuwahara filter, difference of Gaussian edges and saturation variation. These images show that the most salient point can be changed by using our adaptive abstraction approach. 59
4.2 Examples of the Kuwahara filter with (right) and without (left) abstraction. The yellow markers indicate the most salient points of each image, as indicated by the saliency metric. The most salient point in the stylised image is found to be the target object. 60
4.3 An example image (Kuwahara filter without adaptive abstraction) from the 2AFC experiment. The target character is wearing a white shirt and green pants, and is located in the back of the left hand group. 62
4.4 Average response times for each style from the forced choice experiment. The adaptive toon shading technique performed significantly faster than all other styles including standard rendering. 63

4.5 The range of possible target sphere appearances shown to users before all experiments. From left-to-right the styles shown are; Top Row: No stylisation, saturation variation, difference of Gaussian edges and brightness variation. Bottom row: Kuwahara painterly filter, luminance variation, colour quantization and the comic-book filter. 67
4.6 Example frame from Experiment 1 where focus is on the target sphere, which has a brick texture. The style used in this example is luminance variation (*Lum*). 68
4.7 Example frame from Experiment 1, where stylised focus is on some other random sphere, using saturation variation (*Sat*). 69
4.8 Average reaction times from Experiment 1 for all styles. Note error bars represent standard errors of the mean in all graphs. 70
4.9 Average reaction times from Experiment 1 for each individual style. The graph shows that each style resulted in faster times when the focus was on the target object rather than on some other scene object. It can also be seen that some styles perform significantly better than others, for example the Kuwahara filter and saturation variation performed particularly well. . 71
4.10 Average reaction times for the focused abstraction styles from Experiment two. The style containing the Kuwahara filter, Saturation variation and brightness variation (*KSB*) resulted in the fastest times. 73
4.11 Average reaction times for the focused abstraction styles from Experiment three. The style containing edges (*KSBD*) results in significantly longer reaction times than the other two styles tested. 75
4.12 Example frame from an animation shown to participants in eye-tracking experiment (Style *KSBD*). 76
4.13 Distance from gaze position to abstraction centre through time for each abstraction style in Experiment four. As can be seen, the style containing edges (Style *KSBD*), results in closer fixations to target objects. It is also seen that for all styles tested, there are more fixations near target objects early in trials. 77
4.14 Example frame from a scene in Experiment five. 78
4.15 Average fixation amounts for Experiment five. When using adaptive abstraction with stylised focus on the target objects, there are significantly more fixation hits on the target object. 79

4.16 Average fixation amounts for each individual style in Experiment five. Difference of Gaussian edge detection and the comic-book filter result in significantly more fixations on target objects. 80
4.17 Examples of the styles used in the experiment.(a) No abstraction (b) Edge detection (c) Shading stylisation (d) Texture stylisation (e) Saturation variation. 82
4.18 Example trials from the experiment (a) Texture Variation on the character set (b) Saturation Variation on the fruit set (c) Shading stylisation on the simple shape set. 84
4.19 Examples of each level of abstraction for each style. Row A: Edge detection, row B: Shading stylisation, row C: Texture stylisation, row D: Saturation variation. 85
4.20 Average reaction times for each model type for all abstraction types. 87
4.21 Average reaction times for optimal stylisation levels on the fruit set of models. 87
4.22 Average reaction times for optimal stylisation levels on the character set of models. 88
4.23 Average reaction times for optimal stylisation levels on the basic set of models. 89
4.24 Examples of the environment and each style used in the experiment. The target character can be seen in each image and is in stylised focus in each abstracted image. (Top-left) No abstraction (Top-right) Saturation variation (Bottom-left) Texture abstraction (Bottom-right) Edge detection. 92
4.25 Image of the target character for the experiment, shown to particpants before the experiment began. 93
4.26 Average response times for interactive search experiment. Each of the adaptive abstraction styles returned significantly faster times than standard rendering. 94

5.1 A pre-segmented dataset showing each distinct object rendered with a solid isosurface (left) and with one object rendered with a solid isosurface while the rest of the volume is rendered with a high opacity. As can be seen, the right image with high opacity does not display the background data as clearly as the left image. 100

5.2 Various renderings of a torso with pre-segmented bones and organs. Top-left: Traditional rendering. Bottom-Left: Distance transform calculated with the spine as the target object. Top-Middle: Adaptive abstraction with multiple levels of saturation variation. Top-Right: Adaptive abstraction with two levels of saturation variation. Bottom-Middle: Adaptive abstraction with multiple levels of Difference of Gaussian edges. Bottom Right: Adaptive abstraction with two levels of Difference of Gaussian. 101
5.3 Various renderings of a torso focusing on the heart (top) and the spine (bottom). Left images: Traditional rendering. Middle images: Cut away approach showing entire target object. Right images: Fade-away approach focusing on target objects using Sobel edge detection and saturation variation (top), and Canny edge detection (bottom). 103
5.4 Pipeline overview for the two-level volume rendering system. In comparison to the methods described in previous chapters, the system does not use a false colour rendering pass but instead renders the target isosurface as well as the background volume. Object space lines are also added to the system. 105
5.5 Object space line examples, Left: Original volume rendering, Middle: Silhouettes, Right: Suggestive Contours with Silhouettes. 107
5.6 Example from the two-level volume rendering system, Sobel edge detection combined with suggestive contours and the Kuwahara filter applied to a volumetric model of a skull. 109
5.7 An example image shown to participants in the user study. The controllable gauge can be seen in red. 109
5.8 Average normal estimation errors for each dataset. 111
5.9 Average normal estimation errors for each style type over all datasets. . . . 112
5.10 Average normal estimation errors for each style type for the brain dataset. . 113
5.11 Average frames per second for the brain dataset at varying resolutions and styles. 115
5.12 Some sample images from the experiment: First Row - Normal rendering; Second Row - silhouettes, suggestive contours, Canny; Third Row - Sobel, Canny combined with silhouettes, Sobel combined with suggestive contours; Fourth Row - Difference of Gaussian combined with suggestive contours, Canny combined with suggestive contours, Difference of Gaussian combined with suggestive contours and Kuwahara filtering. 116

5.13 Examples of how adding stylisation can affect colour contrast within scenes. The left images are normally rendered while the right images use the Sobel filter (top) and a combination of suggestive contours and saturation variation (bottom). Colour contrast is indicated in purple and measured using the automatic saliency metric presented by Itti et al. [IKN98]. These images show how the saliency of the target isosurface can be increased and manipulated using various styles. 117
5.14 Example of how the adaptive abstraction approach can change the most salient point in the image, marked with a yellow circle in each image; Left: Normal rendering. Right: Canny. As can be seen, adding stylisation can change the most salient point to the target isosurface. 118
5.15 Examples of how multi-level adaptive abstraction affects segmented datasets. The most salient point in each image marked with a yellow circle; Left: Normal rendering. Right: Adaptive Canny edges combined with saturation variation. 118
5.16 Images of an engine block showing how the system, described in Section 5.3, can be adapted to create different types of renderings. The first image shows an engine block with the target isosurface highlighted using Difference of Gaussian edges and the surrounding volume enhanced using suggestive contours. The second image shows the target isosurface highlighted using suggestive contours and the Kuwahara filter while also using a multi-dimensional transfer function. 120

Chapter 1

Introduction

Non-photorealistic rendering (NPR) is a research area within computer graphics which adopts techniques long used by artists to emphasise specific objects and features within a scene. NPR is driven by the desire for stylisation and efficient communication instead of the traditional graphics goal of producing lifelike renderings. Computer-generated images and scenes have long been judged on how realistic they appear; however, there can be a need within such scenes to emphasise subtle attributes to the viewer which would not be obvious in photorealistic renderings. This can be done by varying non-photorealistic abstraction across a scene to focus a user's attention on particular parts of the scene. While such techniques have been explored for off-line rendering, there is a lack of research that deals with using NPR for influencing user attention in interactive scenes. The goal of this thesis is to develop methods and guidelines to utilise multi-level abstraction, and to influence user behaviour in real-time scenes, primarily through the use of perceptual principles.

Real-time applications such as games or urban simulations are often highly complex in nature. Within interactive scenes such as this, it can be necessary for a particular scene object to be highlighted to the user, whether that is an object of interest or a designated goal. This is also true for real-time visualisations of medical, technical and scientific data, where emphasising particular parts of the imagery can help to more effectively communicate vital information to users.

While enhancing certain objects in a scene may be necessary, it is also vital that all available scene data be taken into account in some way to faithfully represent a scene. Displaying background information for reference while ensuring that important scene objects are emphasised is a difficult problem, especially in real-time. In this thesis, we describe a variety of real-time non-photorealistic rendering styles which use variation in abstraction, looking at both image-space and object-space approaches. Each of the NPR filters and techniques discussed is referred to as a *style* in the remainder of this thesis. A hybrid method is also proposed that uses strengths from both classes of approach to produce stylistic and adaptable results in real-time. We propose an adaptive abstraction approach that uses these styles to create a stylised focus within real-time scenes. This approach produces real-time renderings of scenes, which uses the stylised focus to highlight particular objects within scenes.

An adaptive abstraction approach would be considered to be successful if it was able to influence how a user reacts within interactive scenes. Therefore, we perform a number of perceptual experiments to analyse how each style can change user behaviour within real-time scenes. Specifically, we test how each style, and combinations of styles, can change a user's task performance, gaze behaviour and shape perception. We perform a number of tests to determine how each style can be modified to achieve the most effective results for different scene types of varying levels of complexity. The results of these tests establish guidelines on how each style can be best used in terms of scene type and stylisation level. We also look in detail at how adaptive abstraction can be used with a variety of data types, such as volume data, to help user interpretation of complex datasets. We integrate the adaptive abstraction approach with volume rendering of complex datasets such as medical data, to help user understanding of object shape.

1.1 Motivation

The main goal of our research is to investigate how real-time non-photorealistic rendering styles can be adapted to enable multiple levels of abstraction within a scene. We hypothesise that the creation of such an adaptive abstraction in interactive scenes may allow for certain objects to be highlighted and user perception to be influenced. A small number of approaches have been proposed for using non-photorealistic rendering techniques with multiple levels of abstraction to attempt to create a stylised focus on particular objects within scenes. However, most available approaches are not suitable for real-time scenes or large scene sizes as they cannot perform at real-time rates. We investigate whether effective adaptive abstraction could be performed on real-time scenes.

Another motivation is that in current systems, somewhat ad-hoc styles and parameters are used when creating a stylised focus. Little is known about which styles have the most significant effect on a user's perception of a scene. Similarly, there has not been research into how to get the best performance from each individual style by varying parameters. It is also of interest to discover how scene type affects the success of such styles, as different tasks may require different aspects of a scene to be highlighted. These are aspects of non-photorealistic rendering which have not been explored in the literature and there may be significant benefit in exploring such perceptual issues.

1.2 Methodology

At this stage in the development of the field of non-photorealistic rendering, great progress has been made in generating renderings that are convincingly artistic, for both offline and online rendering. Different types of styles have been researched including painterly rendering, edge stylisation, sketch-shading, cel-shading, hatching and many others. While a large number of these styles are suited only for offline rendering, there are many styles that can be implemented in real-time. Advances in graphics hardware have increased the amount of stylisation that can be done per-frame in such renderings. We use shaders, programmed on the GPU, to create such real-time styles and integrate them with existing applications.

The opportunities that come with creating real-time non-photorealistic renderings also bring more challenges. How can real-time NPR styles be best used and what are the limitations of such styles? Perceptually adaptive graphics involves the investigation of all of these issues. Psychophysics is a useful tool for perceptually adaptive graphics, as it explores the connection between physical stimuli and sensory responses. This allows the effect of graphics approaches to be fully understood and, in turn, altered to achieve optimal results.

Throughout this thesis, we use psychophysics to address a number of questions relevant to the efficient and effective use of adaptive abstraction. To maximise the effectiveness of each NPR style, we decided to conduct a series of perceptual investigations to discover how to alter techniques to best create adaptive abstraction, how to use the most effective parameters to improve the effectiveness of various techniques and which techniques are most effective for particular styles. We wanted to directly explore these issues, thereby filling in the gaps in our understanding of the relationships between non-photorealistic rendering, adaptive abstraction and perception.

1.3 Scope

A large focus of this thesis is on using adaptive abstraction to influence a user's perception of real-time scenes through evaluation of the influencing factors and the development of perceptual metrics. The perceptual experiments on non-photorealistic rendering are implemented with a variety of stylistic approaches. When choosing styles to be tested within the perceptual experiments, we picked approaches that would represent a particular category of stylisation, as well as ensuring each style was aesthetically pleasing and was suitable for real-time adaptive abstraction. There are a vast number of existing NPR styles which were not tested that could also be effective for the adaptive abstraction; however we hypothesise that their effects would not be significantly different from the styles implemented. In particular we focused on methods for interactive scenes, although non-real-time methods may be effective for static images or for offline renderings. There exists a number of user-assisted methods for creating NPR, although we focus only on automatic abstraction.

We investigate the success of adaptive abstraction when applied to complex volumetric models such as medical datasets and detailed polygonal model sets such as urban simulations and crowd simulations. The adaptive abstraction approach could potentially be beneficial when applied to other application areas. These areas include route finding applications, architecture visualisation, flow visualisation and any other real-time application where scene understanding needs to be influenced to focus a viewer on particular parts of the scene.

1.4 Contributions

We have developed a real-time approach for creating a stylised focus in scenes using **adaptive abstraction** [publication numbers 5 and 6][poster number 9]. We show how existing **real-time styles** are suitable for such an approach and how they can be adapted to create multiple levels of abstraction. We develop a variety of methods for visually enhancing specific target objects within a scene. We also show that different types of important scene information can be taken into account by combining the strengths of image-space and object-based approaches [publication number 2].

Through a series of **perceptual experiments**, we have gained new insights into how to best utilise non-photorealistic approaches to affect scene understanding. The creation of adaptive abstraction is shown to affect a user in interactive scenes by altering their **task performance**, **gaze behaviour** or **perception of shape** [publication numbers 3 and 4][poster number 2]. Results also show that choice of style, style parameters and scene type are important factors in achieving the best results.

We have shown that adaptive abstraction is effective in a variety of real-time applications, including **volume data**, (as used in one case-study), with segmented and two-level volume rendering systems [publication number 1][poster number 1]. We develop an approach to using adaptive abstraction to emphasise certain features within complex datasets to make them easier and faster to comprehend. This is shown to be true for a number of datasets with varying levels of complexity.

Developers and practitioners will be able to create fast and effective non-photorealistic renderings that have the ability to influence user perception of a scene. In addition they will be able to use metrics and guidelines that we have derived, to create more effective stylistic renderings.

1.5 Summary of Chapters

The rest of the thesis has been divided up into the following chapters:

- **Chapter 2** provides an overview of the previous background and related work in non-photorealistic rendering techniques, with particular focus on real-time approaches. A detailed description is also included of the current state of the art for using NPR to alter scene understanding.

- **Chapter 3** outlines an adaptive abstraction approach to creating a stylised focus within real-time scenes by altering existing NPR techniques. Image-space abstraction techniques are shown and combined with object-space shading and 3D information to create stylistic and reliable representations.

- **Chapter 4** presents a series of experiments on the perception of scenes when adaptive abstraction is applied. Firstly, adaptive abstraction is evaluated using an automatic saliency metric, demonstrating the promise of such an approach. Timed search tasks, eye-tracking and task based experiments were then conducted to fully investigate how a variety of non-photorealistic styles perform in altering user behaviour.

- **Chapter 5** gives a detailed description of how adaptive abstraction styles could be integrated with complex datasets, using volume data as a case-study. Styles from previous chapters are combined with object-space line-drawing styles, to provide stylistic rendering of complex volume data. The adaptive abstraction approach is implemented with both segmented datasets and a two-level volume rendering system. User evaluations are run to investigate how styles can affect user perception of object shape for these visualisations.

- **Chapter 6** summarises our contributions, and provides a discussion of future work.

Chapter 2

Background and Related Work

Non-Photorealistic rendering (NPR) has two main goals; the first is to produce aesthetically pleasing images using styles that do not follow the traditional graphics goal of creating realistic results. The second is to use such styles to better communicate scene information to the user, which can be valuable as there can be a need in complex interactive scenes to draw user attention to appropriate objects and regions within the scene. This can be done by using multiple levels of abstraction within a scene to increase the saliency of certain elements while reducing the visual impact of unimportant scene detail. This chapter presents the relevant work that has been done in real-time NPR, paying particular attention to styles that can use multiple levels of abstraction.

- We first evaluate real-time non-photorealistic techniques that have been presented in previous research. NPR literature has presented a large number of styles, some for aesthetics and some for efficient communication. In Section 2.1 we will focus on NPR styles that are relevant to creating results that can increase user understanding of a scene. We will look at the three main classes of real-time NPR, which are painterly renderings, line drawings and artistic shaders.

- We then focus on how NPR styles can be a valuable tool in a large variety of fields such as visualisations for medical, technical and architectural data. We will also look at how NPR can be used with various types of data such as polygonal or volumetric datasets.

- It will then be shown how previous research involved the use of particular NPR styles to affect user gaze and user scene understanding. This research has shown that gaze behaviour can be successfully influenced by increasing the saliency of particular objects and areas within scenes and images.

- This research area is closely linked with perception and there have been a number of papers that deal with the link between NPR and perception. When dealing with NPR techniques that influence user behaviour, it is important to be able to evaluate such techniques. Different forms of evaluation, which have been used in past research, will be reviewed in this chapter.

2.1 NPR Techniques

NPR is a class of rendering that places emphasis on the creation of convincing stylistic results as well as efficient communication of vital scene information. A vast body of work exists that deals with producing non-photorealistic images for specific applications or simulating specific artistic styles in computer imagery. In this section we discuss each of the major classes of non-photorealistic style. We also focus on fully automatic NPR methods, rather than techniques that are designed to assist a user in the artistic process. A full discussion of a variety of NPR techniques can be found in [GG01] and [SS02], which provide extensive surveys on the area.

There has been much research into various types of automatic non-photorealistic rendering, including both image-space and object-space techniques. The most heavily researched categories of NPR are painterly rendering, line drawing and artistic shading. We will address the most important previous research for each of these areas and assess the strengths and weaknesses of each. There is a number of other non-photorealistic styles, such as simulation of stained glass [Mou03], pen-and-ink rendering [WS96], copper plate engraving [Ost99], mosaic simulation [Hau01, KP02], cubist renderings [CH03], and charcoal rendering [MG02]. However, these styles are not widely researched in comparison with more popular styles, such as the ones described later in this section, and are not usually suitable for real-time rendering as they are focused on creating static images.

Many different techniques exist for creating different types of **painterly renderings** such as watercolour, impressionist and oil painting. These styles can be created by using various techniques such as creating brushstrokes in image-space or utilising particle systems using 3D data. While there is a variety of approaches, the majority of existing painterly styles only use 2D information.

Edge detection and edge stylisation is one of the most common and effective illustration styles. Line drawings can effectively represent an object by using techniques such as highlighting silhouette edges or areas of high curvature. Indeed, examples of line drawings are found in scientific visualisation, technical illustration and architectural design. Perceptual tests by Christou et al. have shown that humans are good at inferring the shape

Figure 2.1: Example of the painterly approach proposed by Hertzmann which uses curved brushstrokes of various sizes [Her98]

of an object from its line drawing alone [CKD96]. This research also shows that adding edges to the scene can help create clearly distinct regions and bold edges can help to direct user focus to particular objects.

A basic method for conveying the shape of a 3D object is shading and there have been a number of **artistic shading** techniques proposed such as cel-shading, pencil-shading, cool-to-warm shading and hatching. These techniques traditionally use available lighting information to calculate and stylise shades on scene objects, which can result in renderings that display convincing artistic shading.

2.1.1 Painterly Rendering

Painterly rendering is a very popular research area within NPR. Both image-space and object-space approaches exist for creating renderings that simulate hand painted results. Painterly styles such as impressionist, halftoning and oil painting can be effectively simulated. While the more expensive approaches, [Her98], offer extremely convincing results, they are only suitable for single images. Faster approaches, [WOG06], can trade-off some authenticity to provide real-time abstraction.

Among the earliest work on painterly rendering was the research by Haeberli [Hae90], where a simplistic semi-automated system was introduced for creating impressionist results from photographs. The system creates brushstrokes by point-sampling the source image in user-specified areas. While the system relies on user input and is suitable only for single images, the results prove that authentic painterly renderings are possible. In later research, fully automatic methods were presented for creating painterly renderings for both static images and 3D scenes.

One popular approach for creating painterly renderings using object-space information is to use particle systems such as in [Mei96, KGC00, HS04]. These approaches use particles in 3D to model screen-space brushstrokes by relating particles with the geometry and surface properties of the model. While such approaches take advantage of scene data not used in purely image-space approaches, they can be expensive and dependant on the complexity of the scene, so frame times can increase hugely for large interactive scenes.

A large number of purely image-based painterly rendering approaches exist. Hertzmann creates images that have the appearance of being hand-painted, created by modelling brushstrokes using spline curves [Her98]. A number of layers are built up where each layer is composed of smaller brushstrokes than the previous one (See Figure 2.1). Similar image-space, brushstroke-based styles which build on this work have since been presented [KMM^{+}02, PY08, HE04]. These approaches can simulate convincing looking long, continuous brushstrokes with varying size and shape, although they are not suited to real-time rendering due to the heavy performance impact involved with creating splines and multiple layers. These styles would also result in poor frame-to-frame coherence in a video or animation. Creating temporally coherent videos and animations can be difficult when creating painterly renderings, as brushstrokes have to be re-calculated for each frame. This can then lead to a jittering effect in the resulting renderings.

These temporal coherence issues have been addressed in a number of papers to create painterly renderings from video. Impressionist renderings from video have been created by Litwinowicz, by calculating brushstroke orientation using optical flow and then clipping each brushstroke using a Sobel filtered image for reference [Lit97]. By using optical flow, brushstrokes move with pixel movement to achieve frame-to-frame coherence in the resulting video. A similar optical flow method for painterly rendering is proposed by Hertzmann and Perlin [HP00], where some brushstrokes move with optical flow and some are painted over where the source is changing considerably. Collomosse et al. [CRH05], obtain temporally coherent painterly renderings from video by using a region based approach. By tracking regions over frames by processing the video as a spatio-temporal voxel volume, a smooth and abstracted result is achieved. More recently, Bosseau et al. have improved the use of optical flow by adding texture advection and mathematical morphology operators [BNTS07]. While these papers show that convincing and smooth animations can be created using image-space approaches, the computational cost of creating such results mean that the type of approach is not suitable for interactive applications.

Santella and DeCarlo [SD02] were among the first to use painterly rendering to create abstractions which are dependent on a user's attention. They use eye tracking data to guide automatic painterly abstraction to create what they refer to as *meaningful abstraction* of images. Visually salient parts of an image are determined with the use of an eye-tracker before the appropriate abstraction is performed depending on the importance of different points. The abstraction uses brushstrokes of various sizes based on the visual importance of each point, allocating smaller brushstrokes to more important areas and thereby adding more detail. DeCarlo and Santella [DS02] expand on their previous work by presenting a region based abstraction system. This system still uses eye-tracking data to generate abstractions, but a region-based system is used. Multiple regions are generated and appropriate detail is displayed depending on each regions importance. The renderings produced by the system are convincing although the technique is only applicable to single images due to its complexity. The images can also suffer from unwanted background detail near focus points in the image due to the constraints of the purely image-space technique.

Other work has demonstrated that real-time interactive painterly rendering is also possible [KLK$^+$00], with virtual environments. Some compromises have to be made in choosing abstracted rendering techniques for lines and fills that are fast enough to run in real-time and that exhibit reasonable frame-to-frame coherence. Later work by the same author, [KSFC02], investigates how to produce various stylised renderings from video, such as cubism and impressionism. The video input is treated as a video cube, a space-time volume of image data which is used to give interactive control to the user. This research shows how a variety of styles is possible in real-time, although these types of highly artistic styles would not be suitable for creating scenes that emphasise certain aspects of the scene, as is our goal.

Winnemöller et al. [WOG06], use image-based abstraction techniques, accelerated on the GPU to deliver real-time abstraction of video input. Each frame of the video is put through a series of image processing filters. First an anisotropic diffusion filter is used to produce a painterly like effect, followed by difference of Gaussian edge detection and finally a colour quantization filter. The combination of these filters produces an effective cartoon style which can be easily adapted to achieve different abstraction results. A similar, more detailed, but slower cartoon abstraction method is later described in [SB09].

A similar filter to the anisotropic diffusion filter used by Winnemöller et al. was earlier put forward by Kuwahara [KHEK76] in which an edge preserving smooth filter that gives a painterly effect is presented. The filter, known as the Kuwahara filter, calculates the mean colour for four surrounding regions for each input pixel, and the output for that pixel is then calculated as the mean for the surrounding region with the smallest variance.

Figure 2.2: Examples of image-based painterly filters. Left: Original image, Centre: Image created by Kuwahara filter, Right: Image created by using a more detailed, but slower filter as described by Kyprianidis et al. [KKD09]. Note that there is more fine detail preserved and fewer clustering artefacts using the approach by Kyprianidis (right image).

This allows low-contrast regions to be smoothed while preserving high-contrast edges. The filter does not produce as convincing results as can be generated by more computationally expensive painterly approaches but some trade-offs must be made to achieve real-time rates. The main drawback of the filter is that it can result in some clustering artifacts, which has resulted in several extensions of the algorithm. The most notable are presented by Papari et al. [PPC07], where a larger number of subregions are used, and more recently by Kyprianidis et al. in [KKD09, KSKD10], where the scale, shape and orientation of the filter adapts to the local structure of the input. While the latter method produces more convincing painterly results than the Kuwahara filter (see Figure 2.2). However, while real-time rates are possible, the detail associated with it means it runs significantly slower than the original Kuwahara filter and would not be suitable when run in combination with other styles.

Cole et al. [CDF+06] create a painterly like style using image-space filters and use an automatic *stylised focus* technique to direct gaze in 3D models. They use simplistic blur and colour saturation contrasts combined with more complex line variations to achieve effective looking results. Their system, which achieves interactive rates (3-5 fps), uses some object-space information by utilising 3D distance as a guide for abstraction. This work confirms that NPR can successfully direct user attention in interactive 3D scenes, although it tests just one style and one scene type. This research also shows how fast abstraction approaches such as blurring and variation of saturation can be effective at influencing user perception.

Previous research has demonstrated that authentic painterly rendering can be achieved, although many of the more effective techniques can be too slow for interactive applications. The most recent research shows that when trade-offs are made, painterly rendering can be achieved that produces good results and runs at real-time rates. Research has also shown that the extent of painterly stylisation can be controlled, which would allow a number of levels of abstraction to be created within a scene. This would then be a valuable tool in creating styles that could influence user perception.

2.1.2 Edge Detection

Edges have been shown to be an important factor in non-photorealistic rendering as the addition of edges to a scene can help create clearly distinct regions and bold edges can help to direct user focus to particular objects and paths. Perceptual tests by Halper at al. [HMH$^+$03a, HMH$^+$03b] have shown that humans are good at inferring the shape of an object from its line drawing alone. Results from that research indicate that users tended to focus more on objects with bold edges. Much previous research effort has gone into the generation and stylisation of edges, both in image-space and object-space.

For object-space edges, a large number of definitions exist for each type of edge. Well known classes of line-drawing that can be automatically extracted include: silhouettes [IFH$^+$03], occluding contours, geometric ridges and valleys, suggestive contours [DFRS03], apparent ridges [JDA07], and suggestive and principal highlights [DR07] (See Figure 2.3). Silhouette edges are the most popular type of edge for NPR and have been heavily researched. Silhouette edges are defined as the edge between a back-facing polygon and a front-facing polygon. Silhouette edges, like most of the classes of line-drawing, are view dependent and therefore must be calculated every frame and are extremely computationally expensive, although there are methods of speeding up the process by making some assumptions [KSG03], or by using information from the previous frame [NM00].

While many approaches concentrate on detection of certain object-space edges, some research has focused on both the detection and stylisation of scene edges. Such an approach is proposed by Northrup and Markosian [NM00]. Firstly silhouette edges are found using an algorithm that takes advantage of temporal coherence. This is done by using the silhouettes from the previous frame as a starting point for the current frame. While finding all silhouette edges cannot be guaranteed using this method, large speed-ups are achieved in comparison to the brute force method for finding these edges. This method was first proposed in earlier work by Markosian et al., [MKG$^+$97]. Once silhouettes are found, Northrup and Markosian determine the visibility of each stroke using an ID-buffer, before creating artistic brushstrokes by combining edges into paths based on their orientation. The paths are then represented by creating 2D rib vectors aligned with the screen. These paths can then be stylised using various effects such as tapering, flaring and alpha fade. By doing this, highly convincing brushstrokes can be created for silhouette edges in a

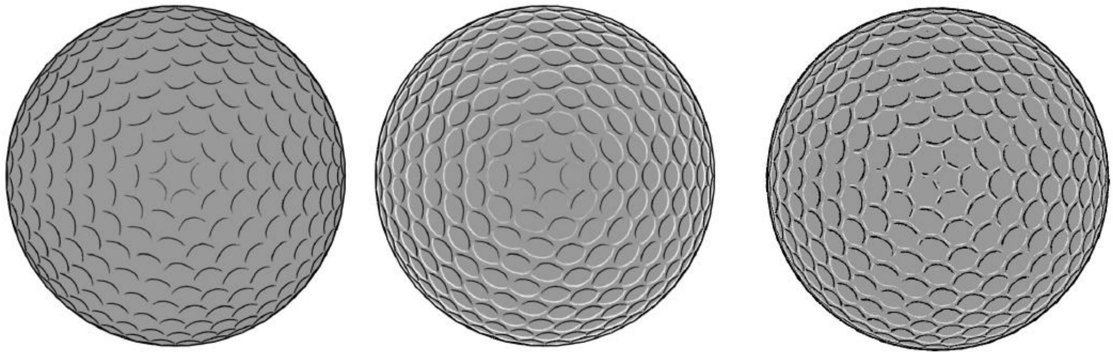

Figure 2.3: A number of edge-detection approaches, as described in [DR07]. Left: Contours and Suggestive Contours combined, Centre: Contours, Suggestive Contours and Suggestive Highlights, Right: Image-Space approach to finding Suggestive Contours. As can be seen, object space approaches such as suggestive contours and suggestive highlights add important shape detail to the image while image-space approaches do not represent object edges as accurately.

variety of styles.

A similar approach to creating stylised edges from silhouettes is presented by Isenberg et al [IHS02], where, unlike Northrup and Markosian's approach, the finding of all silhouettes can be guaranteed by using a costly brute force technique. The visibility of the resulting edges is then calculated using a depth buffer approach, which is faster than the ID-buffer technique, before they are combined and stylised. A slightly different approach is proposed by Lake et al. [LMHB00], where three textures are implemented to represent each type of silhouette edge and create an artistic rounded effect. These textures are a "leftward", "rightward" and "straight" stroke which are chosen based on the view-dependent angle between one edge and the next.

Further approaches to creating stylised silhouettes are put forward [LFX+05, KDMF03], where brush paths are generated and stylised in a smooth and temporally coherent manner. Like the method proposed by Northrup and Markosian, the results are effective, although it is found to be a computationally costly process and can only be applied to small scenes in real-time. A faster approach to creating stylised edges is to add some uncertainty values in image-space to the edges to achieve a sketchy look [ND04b]. This approach allows for an artistic look on the edges to be achieved without having to combine polygon edges into strokes.

As discussed, object-space algorithms can very effectively find edges but are very expensive and dependent on the complexity of the scene. While some object-space edge detection mechanisms can display scene edges in real-time [CF09], these are restricted by scene size and would not be suitable for large complex scenes. Image-space edge detectors, despite not being as accurate as object-space edge detectors, can achieve good results and run at interactive rates. There is a large number of image-space edge detectors, which usually use one of two methods. They find edges, by either measuring edge strength using a simple first-order derivative expression such as the gradient magnitude, or by searching for zero crossings in a second-order derivative expression.

Simplistic first-order expressions can find edges by using gradient operators before searching for local maxima of the gradient magnitude. The most popular example of such operators is the Sobel filter, which calculates gradient magnitude in the horizontal and vertical directions. Other first-order difference operators include the Prewitt operator and also the Roberts cross operator, which finds the gradient magnitude in each diagonal direction.

Second-order approaches include the Marr-Hildreth algorithm [MH80] for detecting edges, which uses the Laplacian of Gaussian function on an image and then looks for the zero crossings in the filtered result. The Difference of Gaussian method is a fast approximation of the Marr-Hildreth algorithm where two Gaussian blurred images are created and the difference between them is calculated to find the magnitude of the gradient.

The Canny edge detector [Can86] is widely regarded as the optimal image-space edge detector due to its accurate and noise-free results. It requires a number of passes over the image including blurring, non-maximal suppression and hysteresis, so the complexity associated with it makes it difficult to run the full algorithm in real-time. When dealing with non-photorealistic rendering both the accuracy and style of the edge detection must be taken into account, which can mean no particular edge detector is suitable for all rendering styles.

Hybrid techniques exist that use image-space edge detectors to find discontinuities in the depth buffer or a normal buffer and thereby find 3D edges [ST90, Cur98, ND03]. However, these approaches can be unreliable as they are dependent on the size of the depth buffer. Many edges can be missed when two objects are at the same depth level; this is especially true for objects distant from the camera where the depth buffer values become less accurate. Other approaches use a normal buffer to find scene edges [Her99]. Methods for generating a normal buffer image can be complex and computationally expensive.

While the vast majority of edge detectors work either in object-space or image-space, there can be a need for both. This is especially true in scenes with a large number of textures, where object-space edge detectors can be effective at finding silhouettes, ridges and suggestive contours, which image-space approaches may miss. Conversely, object-space edge detectors miss edges within the textures of an object, which image-space filters would find successfully. Mitchell et al. [MBC02] present a method for creating real-time non-photorealistic renderings that contain both texture and silhouette edges. This is done by combining edges between textures with edges detected using a normal rendering pass. However, the technique can only find edges between textures and not within textures, which is necessary when scenes contain large complex textures. Also the normal rendering pass can be expensive to create for large scenes. Despite this, the results are promising and they show how combining image-space and object-space edges can produce effective results.

Cole et al. perform experiments to determine correspondences between automated line drawing techniques and hand-drawn line illustrations [CGL+08]. The results provide an indication of lines that artists believe would most intuitively indicate relevant shape and structure and it is observed that these tend to largely correspond to image-space lines and occluding contours. A more recent study by Cole et al. suggests that users are able to interpret certain shapes almost as well from line drawings as from shaded images and that current computer generated line drawing techniques can match the effectiveness of artists' drawings in depicting shape [CSD+09]. This result was discovered by testing perception of shape using user studies described in previous experiments [KDK92, CKD96]. These results show that edge detection techniques can be effective tools in increasing saliency and understanding of important scene objects.

2.1.3 Artistic Shading

Shading is a fundamental method for conveying the shape of a 3D object through a 2D image. A number of NPR methods exist for creating artistic shading. Traditional cel-shaded animations reduce the number of shades used in images and use large areas of block colour and minimal detail to add emotional appeal and stylisation (See Figure 2.4). This type of shading has been recreated by Claes et al. [CFVR01] and Lake et al. [LMHB00] who present a number of real-time object-space approaches for stylising 3D scenes. The Gouraud shading algorithm [Gou98] is used to calculate shading at each vertex before clamping each value to a certain cel-shade from a 1D texture. This work is extended in [BTM06], where the 1D texture is extended to a 2D texture to allow for specular highlights and other tone detail. While these methods produce effective stylised results, the cartoon shader can only take into account an object's diffuse colour and misses any existing texture information.

Lake et al. also propose a technique for creating results which look like they have been shaded in pencil [LMHB00]. A similar approach to cel-shading is implemented where the

Figure 2.4: Cel-shading exhibited on a model, as described in [LMHB00]

Gouraud shading algorithm is again used to calculate the shading value at each vertex. The shading value is then clamped to one of a number of pre-generated pencil shades, which represent different levels of pencil sketch textures. It is then ensured that all vertices from each polygon have the same shading value; if they do not, then the polygon is subdivided linearly and the test is run again. Finally, each polygon is mapped to a specific pencil texture, dependent on its shading value. This shading approach results in very stylistic and visually distinct renderings, although the pencil shader ignores all colour and texture information that may exist in a scene.

Gooch et al. [GGSC98] present a cool-to-warm lighting model suitable for technical illustration of 3D scenes. The lighting model alters the traditional lighting algorithm to allow only mid-tones to be displayed. This highlights the object shape while the detected edges also become visually prominent. Similar techniques are also used to produce metal shading by altering object luminance, hue and tone. The authors hypothesise that these approaches can effectively highlight the shape and structure of objects.

Another method that emphasises shape and detail is described by Rusinkiewiicz et al. [RBD06], who exaggerate shading throughout a model by dynamically changing the effective position of the light for each area of the surface. This has the effect of enhancing the shading regardless of surface orientation. These lighting models produce results that successfully emphasise shape and structure, although much like cel-shading, texture information cannot be taken into account and the systems are designed for creating illustrations of individual objects and not large scenes.

Hatching has been a popular way of illustrating shape by shading where colour and texture cues are not available. A number of techniques exist for creating renderings using hatching. It has been shown, by Saito and Takahashi, that it is possible to use image-space information to generate hatching through the use of geometric buffers [ST90]. It is more popular, however, to use geometric information to create hatching effects such as [HZ00], where hatching fields are created on smooth surfaces dependent on surface curvature and shading. A similar hatching approach was proposed using a sequence of pre-processed mip-map hatch images called tonal art maps [PHWF01]. These textures are applied to models dependent on light and blended together to form convincing results. This type of rendering is not designed for generic objects with colour or texture and is only suitable for generating hatching for small scenes. While artistic shading styles such as the ones discussed are not directly suited to influencing gaze within interactive scenes, it would be of interest to investigate whether changing the levels of stylistic shading between scene objects can influence user interaction with the scene.

2.2 Data Types

NPR has been shown to be a useful tool in the visualisation of a variety of types of content, scene sizes and data types. In this section we discuss previous works that show how different types of data such as polygonal, volumetric and complex multi-dimensional datasets can be combined with non-photorealistic techniques.

2.2.1 Polygonal Data

The majority of object-space NPR techniques are designed specifically for polygonal data. This is because polygonal meshes are the 3D representation most suited to real-time simulation due to their relative simplicity in comparison to NURBS, subdivision surfaces or other types of representation. Many NPR techniques rely on the attributes of each polygon for creating the artistic style. For example, most object-space line-drawing approaches use the edge data from each polygon to determine where important edges exist in the scene [DFRS03, DR07, NM00, MG01]. Similarly, artistic shaders such as [GGSC98, LMHB00] use vertex information to calculate shading values for each polygon. While most painterly rendering techniques use image-space data, many use polygonal data to produce artistic results [LMHB00, GGSC98].

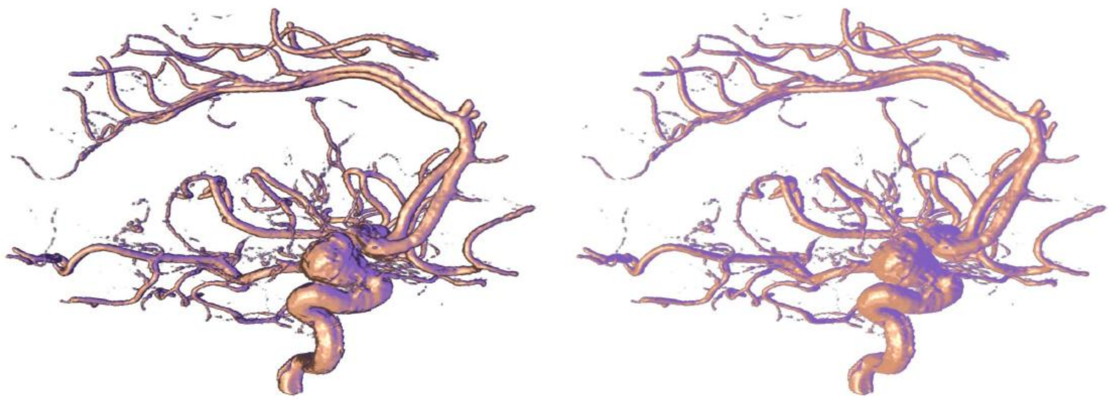

Figure 2.5: Example of how volumetric data can be visualised using non-photorealistic techniques. The right image shows a traditional image of blood vessels while the left image uses silhouettes to help clarify the spatial relationship between vessels [LM02].

2.2.2 Volumetric Data

NPR has been shown to have widespread use in fields in which volumetric visualisations are prevalant such as engineering, science and medicine. To create non-photorealistic renderings of volume data, specialised approaches need to be taken as volume datasets cannot be handled in the same way as 3D surface representations. Numerous approaches exist for volume visualisation, which range from the very accurate to the very interactive [CKY01, PB07, NSW02]. In recent times, the gap between these two extents has narrowed with technologies now enabling the interactive visualisation of very complex volume data. While some approaches involve first obtaining a surface representation from the volume before applying standard 3D renderings, direct volume rendering (DVR) techniques operate directly on the volume data. This can be achieved through the use of 3D textures, generated from view aligned slices of volume data and optimised using GPU hardware for real-time performance [CN94, CCF94]. DVR techniques use a transfer function to select which voxels are displayed from within the dataset.

Other approaches employ techniques such as volume ray-casting [HSS$^+$05], facilitated through adaptive optimisations for real-time visualisation of implicit surfaces defined by a volumetric grid of sample data. Volume visualisations can also be rendered using cutaway approaches where layers of data are cut back to reveal more relevant data to the user [MRH08, LRA$^+$07, VKG05]. This cutaway approach is also applied to polygonal data [BF08], but it is more commonly used for volume visualisations, due to the multi-layered nature of volume data. Volume datasets can be pre-segmented to allow users to control individual transfer functions for each segmented object [HBH03, RGW+03]. This allows more control over how each object is displayed, which can lead to clearer renderings of complex datasets, although some problems can arise with regards to the accuracy of the boundaries of each object [TSH98]. This is because voxel resolution segmentation can result in obvious and jagged boundaries.

Existing works that have explored NPR techniques in volume rendering include that of Ebert and Rheingans [ER00], who use sketchy lines and silhouettes to visualise volume data, and Burns et al. [BKR+05], who extract a number of classes of lines directly from volume data. Treavett and Chen employ a stylistic rendering approach resembling pen-and-ink illustrations for volume visualisation [TC00]. Lum and Ma employ hardware-accelerated parallel non-photorealistic styles in their volume renderings [LM02], utilising object-space GPU techniques for enhancement of the image (See Figure 2.5). They use tone shading, silhouettes and depth cues to provide perceptual improvements to the rendered volume data. Contour enhancement on the fragment shader and tone-shading are employed in combination with MIP compositing and DVR approaches for focus and context two-level volume rendering by Hadwiger et al. [HBH03]. In separate work, Hadwiger et al. integrate ridges and valleys and tone shading into their system for real-time volume ray casting [HSS+05]. Svakhine et al. [SEA09] present image-based outlining techniques that can generate pure line drawings or helpful feature enhancements for illustrative volume visualisation. These papers show that non-photorealistic approaches, particularly line-drawing, can be combined with volumetric data to create convincing results of complex data at interactive speeds.

2.2.3 Images and Video

As discussed in previous sections, a large number of non-photorealistic approaches are designed specifically for images and work purely on a pixel-by-pixel basis. Unlike object-space techniques created for specific types of data, image-space approaches can be used with any type of scene. Painterly rendering and edge detection are the two image-based stylistic approaches most heavily researched, due to the ability to create such styles with no information about the 3D structure of scene objects, which styles such as artistic shading need. Painterly techniques. such as those put forward by Hertzmann [Her98] or Santella and DeCarlo [DS02], are suitable for single images only due to the complexity of the brushstroke based systems and the problem of temporal coherence. Other painterly systems can create artistic results from video with good frame-to-frame coherence but are still not real-time [Lit97, BNTS07, CRH05]. Finally a class of non-photorealistic rendering exists that can create painterly results at real-time rates using less complex approaches, examples of which presented by Kuwahara [KHEK76], or Winnemöller et al. [WOG06]. Image-space edge-detectors can usually work effectively in real-time, with the exception of complex multi-pass techniques such as the Canny edge detector. While image-space edge detectors are suitable for interactive applications, they can miss vital edges in the scene due to similarities in colour between objects.

2.2.4 Multidimensional Datasets

A difficult research problem in computer graphics has been how to effectively visualise datasets that contain a number of different dimensions. Non-photorealistic approaches, particularly painterly rendering, are particularly suited for this type of visualisation, as each brushstroke has a number of properties that can represent a different attribute of the dataset. Examples of this include the work of Healey et al. [Hea01, HTER04, THE07], where complex weather maps are visualised using various properties of each brushstroke such as luminance, size, hue and orientation. Interrante et al. [Int00] use similar methods to visualise agricultural datasets.

2.3 Application Fields

In this section we show how NPR can be used within a variety of fields including games, weather maps, urban simulation, medical imaging, scientific visualisation and technical illustrations.

Entertainment Industry

Non-photorealistic rendering has been used in both games and movies more and more frequently in the last ten years. Within the games industry, titles such as *The Legend of Zelda: The Wind Waker* 2002 [Nin02], *XIII* 2003 [Ubi03], and *Grand Theft Auto: Chinatown Wars* 2009 [Gam09], have shown how cel-shading can be used to create stylised and effective renderings, both on home consoles and handheld consoles . Handheld consoles in particular can benefit from cel-shading renderings as they can help show complex scenes on screens of limited size (e.g. *The Legend of Zelda: Phantom Hourglass* [Nin07] on the Nintendo DS). Cel-shading is the most popularly used style within games although other approaches have been used such as ink and wash painting in *Ōkami* 2006 [Nin06], an illustrative comic book style in *Borderlands* 2009 [Sof09], *Prince of Persia* 2008 [Ubi08] and watercolour painting in *Valkyria Chronicles* 2008 [Seg08]. Past research has also shown how existing games, such as *Quake III* 1999 [iS99], can be stylised using pencil sketch, blueprint or brushstroke styles [MG01].

NPR has also been used in the movie industry, beginning with the painterly effects seen in *What Dreams May Come* 1998 [Ent98]. Movies such as *Sin City* 2005 [Fil05], and *300* 2007 [Pic07], exhibit an NPR style that tries to recreate comic book illustrations by adopting various methods including manipulation of colour contrast and colour saturation. A cel-shading style is shown in movies such as *Waking Life* 2001 [Pic01], and *A Scanner Darkly* 2007 [Pic06]. This style is created by using a frame-by-frame rotoscoping tool over live action. The increased use of NPR in both games and movies shows that it is a viable way of presenting information to a user.

Medical, Scientific and Technical Imaging

It has been shown that non-photorealistic approaches can be effective for the visualisation of medical and technical data. Particularly, there has been a large body of research with focus on medical visualisations using volume data. In [PTD05], a technique is presented for using NPR for medical visualisations with emphasis and focusing methods. Silhouettes and hatching are used among other methods to emphasise a certain object within the medical dataset. Similar stylistic techniques are implemented to create interactive renderings of medical and technical data, by Nagy et al. [NSW02], where hatching, silhouettes and toon shading are tested. Stylistic edges and shading are utilised by Li et al. [LRA+07], where both medical and technical imagery is visualised using cutaway views of each object. Medical and technical images are also rendered using non-photorealistic techniques by Lum and Ma [LM02] such as colour variation and line drawing to create clear and stylistic renderings.

Burns et al. [BKR+05] extract different classes of line, such as contours, to render line drawings of various types of volume data, with a focus on medical visualisations. Line drawings are also used by Lum and Ma [LM02], in combination with tone shading and depth cues to create stylised renderings of medical data. Treavett and Chen [TC00] also use NPR methods with medical volume data to create pen-and-ink illustrations of complex datasets. These types of visualisation show how NPR can be of use in complex datasets to provide simpler renderings which can be easier to comprehend.

Weather, Agricultural and Oceanography Maps

Complex datasets such as weather maps or oceanography can be effectively visualised as shown by Healey et al. [Hea01, HTER04, THE07] using stroke attributes such as orientation, hue, texture and shape to signify wind speed, temperature, pressure and precipitation

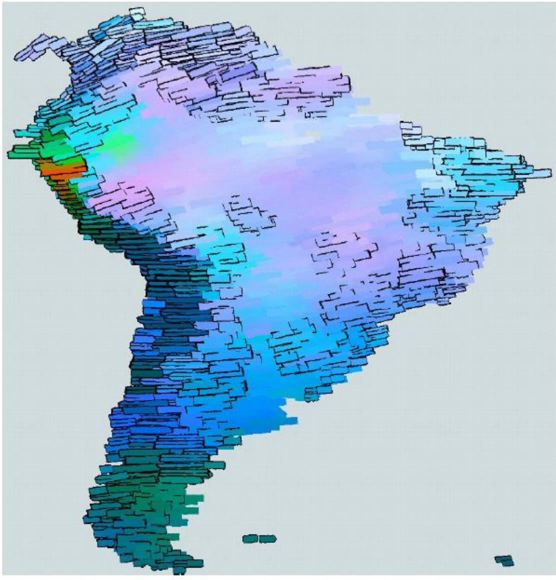

Figure 2.6: Image from Healey et al. where weather over South America is visualised using brushstrokes. Each brushstroke property is linked to an attribute of the dataset; Wind speed - Hue, Temperature - Luminance, Radiation - Size, Precipitation - Orientation [THE07]

(See Figure 2.6). The images generated have been shown to effectively illustrate multiple attributes in a clear manner.

Similar visualisations to those of Healey et al. have been proposed using various properties of non-photorealistic textures in 2D flows [KML99], where varying the colour and direction of strokes and arrows within an image represented varying properties for fluid mechanics. Agricultural maps can also be visualised using these methods [Int00], where overlapping natural textures and varying colour represent different variables. These visualisations of complex datasets are suitable for single images but show the strong potential NPR can have for communicating vital information to the user.

Architecture Datasets

NPR has also been seen as a good way to present unfinished architectural drawings. In [WS94] a pen-and-ink approach is proposed for visualising architectural models. In [SSLR96], over half of surveyed architects chose a non-photorealistic sketchy presentation of a building as a suitable way to present a first draft to a client over a simple shaded image or the complete detailed plot. This was because the sketches were seen as more lively and creative and less artificial. More examples of how the visualisation of architectural drawings can be aided by non-photorealistic rendering can be seen in both [Pom04] and [ND04a]. In the latter paper, it is shown that by using non-photorealistic techniques, perceptually important visible and occluded edges can be displayed to give perceptually efficient views of architectural structures.

Virtual World Representations

Using non-photorealistic styles in virtual reality worlds has been seen as a good way of presenting cultural heritage representations to novices in the field [RD03]. This is done by creating more "believable" scenes using stylisations such as pen-and-ink and a paper-grain method, which was previously shown to provide good representations for virtual heritage [$\mathrm{CTP^+03}$]. It has also been seen that NPR can complement augmented reality approaches where virtual objects need to be convincing rather than realistic [Hal04].

2.4 Meaningful Abstraction

Although a number of NPR techniques are motivated purely by aesthetics, a fundamental premise for most NPR styles has been to improve the effectiveness of images in conveying information. Artists and proponents of NPR can exploit well known perceptual mechanisms to provide informational cues to the viewer. This type of approach can be necessary, as when rendering complex data there can be a problem of information overload where the rendered images are too detailed for the human visual system to process quickly and usefully. This problem is further exacerbated in dynamic, real-time scenes, as the user has less time to process each frame. There have been a number of approaches for utilising NPR to highlight important features within scenes while removing extraneous detail. These can help in interactive scenes for the analysis of data or for visual search tasks. This can be done by increasing the saliency of target objects and areas by using perceptually effective non-photorealistic enhancement methods. Unimportant data can also be abstracted to further increase the saliency of the target features.

Outside of NPR literature, previous work has involved exploring ways to apply the rules of perception to create salient images [War04]. Kosara et al. [KMH01] present a technique called "semantic depth of field", to increase saliency of certain parts of an image by simply blurring out objects according to their importance, while sharply depicting the objects of interest. This technique was later shown to be an effective and efficient method for guiding the user's attention to certain scene objects [$\mathrm{KMH^+02}$]. Work has also been done to investigate how effects from top-down visual processing can affect a user's gaze behaviour

[CCL02]. This can be exploited to enable degradations in quality for unimportant objects to improve speed without the user perceiving any quality difference.

Related work has also been carried out by El-Nasr and Yan [MENY06], who analyse visual attention patterns within interactive 3D environments. Sundstedt et al. [SCCD04] investigate a user's perception in a visual task to enable certain degradations in quality and reduced frame-times. McNamara et al. [MBG08a] show how search task performance can be influenced by using subtle gaze direction. These papers show that influencing user gaze can be achieved using a variety of photorealistic techniques and while valuable lessons can be learned from this work, other research within NPR shows that stylistic techniques can also increase the saliency of important objects. Stylistic NPR approaches can be preferable to using simplistic and sometimes obvious techniques such as blurring and sudden changes in luminance. Other research has shown that non-photorealistic approaches can effectively highlight important parts of a scene while maintaining a natural and aesthetically pleasing appearance.

Setlur et al. use both photorealistic and non-photorealistic methods to resize images while displaying important parts of the scene in their original size [SLNG07]. This is done by using an importance map to extract important features before displaying the image without any additional style or with image-space NPR styles. The results show that this method can effectively pick out important parts of photos and medical images and display them effectively. Santella and DeCarlo [SD02] use eye tracking data to guide automatic painterly abstraction to create meaningful abstractions of images. Visually salient parts of an image are determined with the use of an eye-tracker before the appropriate abstraction is performed, depending on the importance of different points. The painterly approach described in [Her98] is used to abstract the images. Larger brushstrokes are used for unimportant areas while smaller brushstrokes are used to highlight vital detail.

In follow-up research, DeCarlo and Santella create a region based abstraction system in [DS02]. A similar approach to the previous paper is employed in order to obtain importance data, before photographs are transformed into line-drawing style images with bold lines and large regions of constant colour (See Figure 2.7). This approach creates a hierarchical structure of the image to allow for regions of varying detail to be created. Boundaries between regions are displayed using bold edges to enhance the artistic effect. The results in both these papers are convincing and show how NPR can be effective in creating images with multiple levels of abstraction; however, they are also extremely expensive and suited only to off-line renderings of photographs. The approach is also dependent on eye-tracking input which makes the transition to real-time video difficult.

A similar method for creating painterly renderings which preserves important features

Figure 2.7: Example image from Santella and DeCarlo, who produce painterly renderings by creating coloured regions of varied size depending on the visual importance of each area [DS02]

is proposed by Collomosse and Hall [CH05]. They use a saliency measure that combines feature rarity, feature visibility and feature class to return a final saliency value [HOC04]. With the resulting information, they use a spline based painting method [Her98], to create more detailed brushstrokes in visually salient regions. Using the approach, a large level of control over the stylisations can be exhibited. Despite the promising results, the technique is again only suitable for off-line rendering of single images.

Cole et al [CDF+06] describe a technique for directing gaze with stylised focus. Strong object-space edges are utilised by varying their density and texture. This effect is combined with colour variation and contrast to create an artistic focus on a particular area within a scene (See Figure 2.8). The colour variation used in this paper is a simplistic way of influencing gaze within a scene but it was found to be effective. The edges presented in the system require a large amount of pre-processing but the system does achieve interactive rates. This work in directing user focus shows the promise that NPR has in this area, both for complex styles like stylistic edges and simplistic colour variations. Colour variation has been used in other research, for example, altering colour saturation to focus user's attention on particular parts of objects is shown to be an effective mechanism in [LRA+07].

There is also a body of research into creating stylised focus in volumetric visualisa-

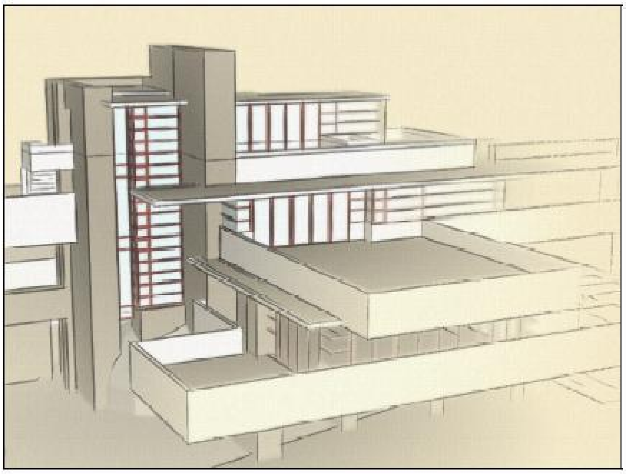

Figure 2.8: Image showing a stylised focus on the top left part of the scene. Multiple levels of abstraction are implemented to focus the user of certain areas [CDF+06]

tions. Creating clear renderings from volume datasets can be particularly hard due to the complexity of high-resolution 3D data. Volumetric data such as medical imaging and scientific visualisations are particularly prone to this problem and research has investigated whether non-photorealistic methods can help create perceptually clearer renderings. Engel et al. describe a process for pre-integrating volume data, which eliminates some artifacts related to low volume sampling resolution [EKE01]. Using a stylised pen-and-ink approach, Treavett and Chen note that their renderings can be overlaid with photorealistic rendering to create selective abstraction of detail and to improve, for instance, user understanding of translucent parts of the volume [TC00].

Viola et al. discuss methods for highlighting and enhancing pre-segmented volume data through the use of cut-away and ghosted views of volumetric data [VKG05]. Hauser et al. propose the use of two-level volume rendering, which merges DVR and MIP techniques in an interactive tool [HMBG00]. They argue that such an approach, based on a *focus-and-context* strategy, provides intuitive benefits to users, allowing them to peer inside inner structures, while keeping surrounding objects integrated for spatial reference.

Focus and emphasis is also discussed by Preim et al [PTD05], who survey a number of existing areas where various enhancement and NPR techniques can be used to improve tasks in medical visualisation. Kruger et al. present a focus and context visualisation framework called Clearview, assigning varying levels of importance across an object and rendering the object adaptively using texture-based volume raycasting [KSW06]. Kim and Varshney use a visual saliency operator to compute an emphasis field which is used to direct viewer attention to relevant parts of a volume rendering [KV06a]. This research, which focuses on volume visualisation, emphasises that the approach of creating meaningful renderings using multiple levels of abstraction is useful for a variety of implementation areas, especially those that contain a large amount of complex data.

2.5 Evaluation of NPR

While we have shown that rendering approaches, which use multiple levels of abstraction, are well researched and are considered a viable way of altering scene understanding, we must also focus on the evaluation of such methods. It is necessary to understand the strengths, weaknesses and limits of using such approaches and there has been research into evaluating adaptive abstraction approaches in the past. In the research already discussed, a large number of styles have been used to create stylised focus on particular objects within scenes. Examples of these are descriptive lines and brushstrokes to enhance silhouettes and contours, and abstractions to texture and fill areas, which are designed to remove extraneous detail. Given the abundance of stylisation choices, it is important to have measurable indicators of quality for evaluating and comparing the effectiveness of different abstraction styles.

There are a number of different ways to measure how scenes using non-photorealistic rendering can affect user's scene perception. It has been shown that the affect of NPR styles can be evaluated through observational studies [INC$^+$06, SSLR96], by timing task performance [GRG04], by observing user eye-tracking behaviour [SD04, WOG06], by testing user distance perception [GW02, PRI$^+$09] or by testing user shape perception [CKD96, CDF$^+$06]. The variety of evaluations shows that NPR can successfully change a user's perception of a scene in different ways. We are particularly interested in research that involves measurable evaluations of scenes with adaptive abstraction.

2.5.1 Psychology of NPR

Halper et al. explore the relationship between NPR and psychology [HMH$^+$03a, HMH$^+$03b] and observe that NPR can influence user perception and judgement. Users tend to select objects with bold edges over other objects within a scene. It is shown that NPR can be used to guide user behaviour in navigation and exploration, where users view paths with higher detail as more interesting for exploration relative to paths with low detail (See Figure 2.9).

Further research into how NPR can affect the psychology of scenes was presented by Duke et al. [DBHM]. In a number of user studies, a forced choice paradigm was used to

Figure 2.9: Image from a study which explored the relationship between NPR and psychology. Users tended to choose the more detailed routes with line drawing as more interesting for exploration [HMH+03a]

determine how rendering style affected high-level choices such as danger awareness, awareness of strength and weakness and path choice. It was again found that such high level choices were influenced greatly by rendering style, and that such effects can be controlled, thus demonstrating it is possible to exert significant influence over user judgement and how scene meaning is conveyed.

2.5.2 Observational Studies

Isenberg et al. [INC+06] present an observational study to examine users' understanding and assessment of hand-drawn pen-and-ink illustrations of objects in comparison with non-photorealistic renderings of the same 3D objects. Their results showed that people perceive differences between those two types of illustration but that those that look computer-generated are still highly valued as scientific illustrations.

Similarly, an observational user-study is described by Schumann et al. [SSLR96], where architects were asked to observe a number of architectural designs with either normal computer aided design (CAD), artistic shading or a NPR sketch renderer. It was found that the reaction to each type of image differed and that the NPR images were seen as a good way of presenting architectural information as it encouraged interaction. These types of observational study are extremely valuable [WBC+07], especially when performed with experts from the area [JAL+03, ALD05], as it was in this case. While such qualitative research can certainly increase the knowledge of the perceptual effect of NPR styles, further quantitative research is necessary to fully understand the advantages and limitations of each style.

2.5.3 Timed User Tasks

Gooch et al. [GRG04] prove that NPR has distinct advantages over photorealistic imagery in communication and identification by timing facial learning and recognition responses. Non-photorealistic renderings of facial images were created by omitting extraneous detail and focusing attention on relevant features. It is proven that the speed of user recognition compares well with non-photorealistic images compared with normal photographs and the speed and accuracy of user learning increases significantly with non-photorealistic images.

2.5.4 Space Perception

Gooch and Willemsen [GW02] perform experiments to investigate how NPR styles affect space perception, specifically in immersive environments. The first of the experiments tested how people gauged distance within a physical hallway and also a non-photorealistic version of the hallway using a head mounted display. A variety of edge styles are tested such as silhouettes, boundaries and creases. The results showed that NPR styles, with only edges and no shading, can communicate distance almost as well as traditionally rendered environments.

Further research by Philips et al. used similar line-drawing styles to further investigate space perception within non-photorealistic scenes [PRI+09]. Results indicate that people tend to misjudge distance in NPR environments in comparison to real environments, although this result was also found by Gooch and Willemsen and can be explained by the fact that this misjudgement is not exclusive to NPR renderings and very similar results can be seen in traditionally rendered environments [TWG+04].

2.5.5 Shape Perception

It is also of value to determine how the perception of shape is affected by various NPR styles. Early experiments judged how well images of shaded, textured and line-drawn images portrayed the shape of simplistic primitives [CKD96]. Users were asked to control a probe so that it appeared coplanar with the surface at that point. Results show that shape perception of objects is quite accurate using shaded and line-drawing styles when objects were altered using slant and tilt components.

Cole et al. [CSD+09] use the same experimental procedure to investigate how well different types of NPR line-drawing depict shape in comparison with traditional shading and artistic images (See Figure 2.10). It was found that non-photorealistic line-drawing methods perform equally as well in conveying shape in comparison with artistic images. It is also seen that line-drawings can almost match the efficiency of shaded images for

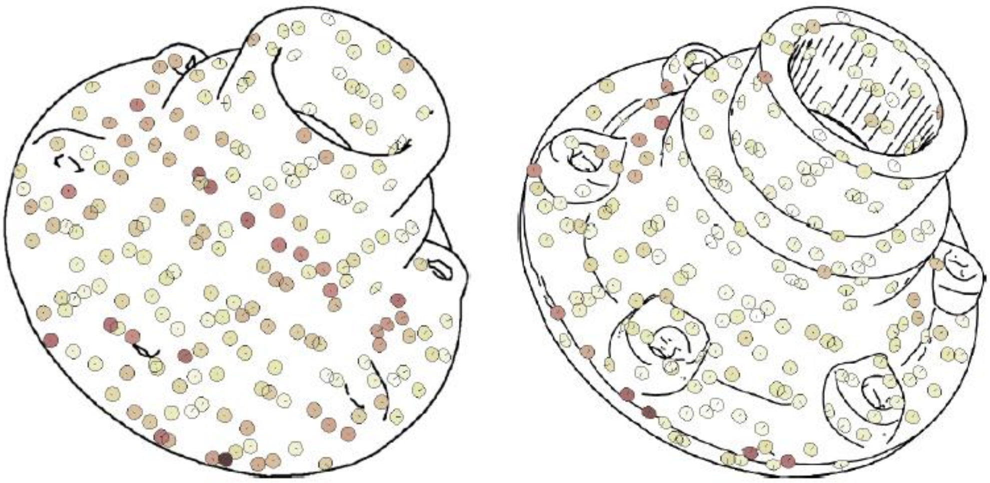

Figure 2.10: Example of the results from the shape perception study in [CSD+09]. The gauges displayed show how well user's estimated the surface normal at that point, coloured by how far they deviate from the ground truth (Left: contours, Right: apparent ridges)

depicting shape. Further work has investigated how well different texture components affect user shape perception [KHSI04], where it was found that applying a basic directional grid onto an object increased the performance of shape perception.

2.5.6 Eye-Tracking

Eye-tracking mechanisms have become a popular way to investigate user gaze behaviour in interactive scenes. For example, Sundstedt et al. propose utilising eye tracking as part of the computer game design cycle to predict fixation behaviour of players [SSWR08]. Santella and DeCarlo evaluate their region based multi-level abstraction system [DS02], using an eye-tracker [SD04]. Eye-tracking data is used to investigate whether images created using the original system successfully focused a user's interest on particular parts of an image. The results validate the design of their original system and show that a user's focus can be guided using NPR.

Cole et al. [CDF+06] validate the stylised focus approach for directing user gaze using an eye-tracker to gauge if user fixations are guided successfully. As in [SD04], eye-tracking experiments were used on single images from the system to prove that gaze was directed effectively. The system was not tested on a user interacting with the scene, which would have provided valuable feedback; however, eye-tracking was proved to be a successful approach for evaluation of non-photorealistic images with varying levels of abstraction.

2.5.7 Saliency measurement

Previous research shows that knowledge of low-level saliency can be used to create perceptually influential renderings and also to create valid forms of automatic saliency measurement for existing scenes. Healey et al. [Hea01, HTER04, THE07] applied perceptual strategies in his work to create highly effective results. Perceptually salient features are created within visualisations by using knowledge of the low-level human visual system. Their research states that a small number of basic visual features can be used to guide attention, such as the first-order properties of luminance and hue and the second order properties of orientation, texture and motion.

While many of the techniques mentioned in this and previous sections aim to increase the saliency of target objects and areas, there must be quantitative measures to evaluate whether this goal is achieved. Itti et al. [IKN98] create a saliency mapping technique that measures saliency within an image by using orientation, colour and intensity metrics. Another saliency measure, by Hall et al., uses a combination of the rarity, visibility and class of features to determine visually important parts of an image [HOC04]. These metrics are useful tools for quickly finding the most salient parts of an image.

A method for saliency based enhancement of volume-visualisation was proposed in [KV06b] and, while NPR techniques were not used, it was shown that saliency-guided visualisation can be helpful for guidance through complex datasets. There has been much work on fully understanding saliency cues in the human visual system [Hen03, CI06, Tre03, KB01] and this research shows that by manipulating certain image properties we can effectively influence user gaze direction.

Many of the papers mentioned in this section so far largely focus on bottom-up processing i.e. manipulating user gaze through low-level visual cues such as colour or shape. However, research has shown that when a user has a specific task, the visual search of a scene is affected. When previous knowledge of a scene influences how a user perceives that scene, this is known as top-down processing. Knowledge of top-down processing can be exploited with visual tasks or goals to determine where a user will look. Mack and Rock [MR98] presented a principle known as *inattentional blindness*, where users tended to ignore visible objects if they were not important to the task at hand. This principle has since been used to speed-up real-time scenes by lowering the quality of less important objects, as they would be attended to less [CCL02, SCCD04]. This type of research shows that both low-level and high-level factors must be taken into account when evaluating saliency of objects within interactive scenes.

2.6 Discussion

A review of the literature suggests that non-photorealistic rendering techniques can be used to influence user gaze behaviour and user scene understanding. While there is a large amount of NPR styles available, past research shows that many object-based techniques for abstraction are too expensive to run in real-time. Instead it may be necessary to use a combination of faster, simpler object-based methods with image-based abstraction styles on each frame. To accomplish the goal of influencing user perception, adaptive abstractions are needed using real-time styles. This necessitates stylisation techniques that can employ multiple levels of abstraction. Existing research on the human visual system and user psychology gives us valuable information on how to create salient objects within a scene. Existing research in creating stylised non-photorealistic renderings with focus also helps us to better understand how to affect user behaviour within an interactive scene.

Research on the strategies for evaluation, reviewed above, has shown that NPR can be a successful tool for improving the speed and the interpretation of images and data sets to facilitate better user understanding. While the past evaluations of gaze direction are promising, it would be of interest to continue the research to explore many unanswered questions. Further studies are clearly required in order to evaluate the effectiveness of NPR in animated scenes and in particular real-time interactive animations. It would also be of interest to compare how a variety of styles perform against each other in affecting user gaze, how combining styles affects performance and how scene type influences the results.

Chapter 3

Adaptive Abstraction

3.1 Introduction

NPR can be used to create simple renderings of complicated scenes to make them faster and simpler to comprehend. This can be achieved by borrowing techniques that have long been used by artists to control a viewer's focus on an image. To successfully affect the way users perceive interactive scenes, it is possible to use real-time rendering styles and ensure that these styles can be altered to create multiple levels of abstraction across an image. These levels of abstraction can then be used within interactive scenes, combined with segmentation information, to create adaptive abstraction across a scene. This adaptive abstraction approach enables a stylised focus to be created around target objects and therefore allows user perception to be influenced.

As discussed in Chapter 2, past research indicates that in order to create stylistic renderings of large scenes in real-time, image-space NPR techniques are currently most suitable. The majority of existing automatic NPR techniques for 3D scenes are not applicable to generic scenes as they are too expensive to run in real-time. This is due to the fact that NPR approaches for 3D scenes use scene data to produce stylisations such as paint strokes or edge representations, which then results in the frame-times being dependent on the size and complexity of the scene. In addition, many methods will only work for a single type of 3D model, such as polygon models [Mei96], parametric surfaces [WS96] or volumetric models [TC00]. These types of object-based methods require knowledge of the geometry detail in the scene and they are often unsuitable where important detail is only available in texture.

Image-based abstraction techniques can solve these problems as access to the model and scene data is not necessary. Image-space approaches can guarantee frame-rates, regardless of scene size or complexity. These approaches also take into account important visual detail that can be missed by object-space approaches, such as texture information. In addition, image-space methods can be adapted to create multiple levels of abstraction for the scene. However, image-based abstraction techniques, by their nature, work on a pixel-by-pixel basis. This means that image-space techniques can ignore available 3D object information, such as shading, 3D-shape or depth information. That data is important for representing scene objects faithfully or for achieving the goal of highlighting certain objects within the scene.

We propose a solution to this problem by creating a system using a hybrid approach, utilising the speed of image-space styles combined with object-space information. This object-space information is integrated in the form of fast and effective shading styles as well as a false colour rendering pass, which is used for object identification. Image-space techniques are used to ensure interactive rates while object-space data is also used to create renderings that can adapt to the scene. While the main goal of the approach is to create renderings that can influence user perception, we also ensured that the styles used exhibited aesthetically pleasing results. In this chapter we describe the implementation and alteration of a number of real-time styles, with a view to creating renderings that successfully use adaptive abstraction to affect saliency within scenes. We will address these questions:

- Can convincing non-photorealistic abstraction be performed, for a variety of styles, in real-time?
- Can such real-time styles be adapted to create multiple levels of abstraction?
- Can we use multiple levels of abstraction in scenes to create renderings with adaptive abstraction, using object importance as a basis for abstraction?

3.2 Real-Time Styles

The goal of creating a system that successfully affects how a user perceives a scene necessitates the implementation of a number of styles and approaches. While there is evidence that computationally expensive image-space approaches would be successful in achieving this goal [DS02], there is no research into whether this is possible for real-time approaches. Therefore, it was first investigated which real-time styles could be effectively modified to create renderings that increase the saliency of important areas while abstracting extraneous detail in the scene. While it was important that each style was both real-time and adaptable, it was also necessary to choose a wide range of styles, each of which we hypothesised would be effective in influencing user perception. We implemented a number of different types of style to enable evaluation of each type of stylisation in future experiments.

3.2.1 Painterly Rendering

As discussed in Chapter 2, painterly rendering is one of the areas of stylisation that is most heavily researched within the field of NPR. Painterly abstraction may be a useful tool for influencing user gaze behaviour in real-time scenes as past research has shown that images containing different levels of painterly abstraction can successfully cause a user to focus on particular objects [SD02, CH05]. We hypothesise that results found using painterly abstraction on singular images will hold true in real-time scenes, enabling multiple levels of painterly abstraction to be used to direct the user's attention to particular objects in interactive scenes.

There are many methods for achieving authentic painterly results by simulating the creation of brushstrokes throughout an image. However, the complexity associated with creating such brushstrokes means that these methods can be difficult to run in real-time [Her98, Lit97], although some researchers have managed to produce frame rates of 1-4 fps using a brushstroke based approach [HP00]. Additionally, brushstroke based approaches can cause problems with temporal coherence.

To avoid these problems and achieve real-time abstraction, it was decided that a filter-based approach would be suitable. Such approaches [KKD09, WOG06, LSF10], work on a pixel-by-pixel basis to abstract the image in a painterly manner without creating brushstrokes. **The Kuwahara filter** [KHEK76], an edge-preserving smoothing filter, was implemented as it is suitable for real-time applications, it gives a convincing painterly effect and it has good frame-to-frame coherence, with no flicker.

The Kuwahara filter works on a per-pixel basis throughout the image. The filter calculates the mean colour and variance in each of four adjacent regions for each pixel. These regions are generated by dividing the surrounding area into four overlapping windows, each containing the centre pixel itself (see Figure 3.1). The output colour for each pixel is the mean colour for the adjacent window with the smallest variance. This has the effect of smoothing internal regions while keeping the edges between colours sharp. The filter is typically 5x5 in size although it is altered within our work to change the amount of abstraction desired. The filter can be applied with varying sizes, from 3x3 upwards, although a filter of 7x7 or larger greatly reduces frame times.

Further research, which has been done in parallel to our work, has investigated how

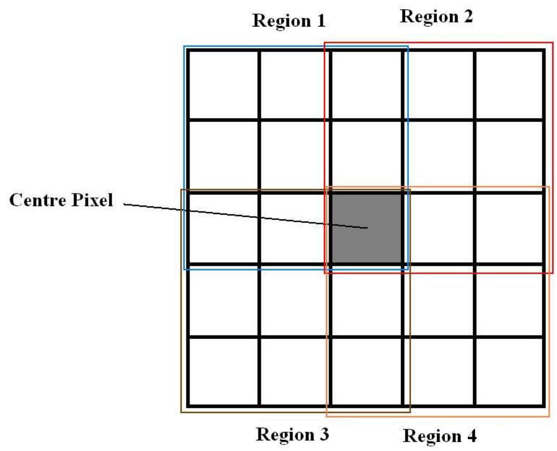

Figure 3.1: Example of the layout of a 5x5 Kuwahara filter containing 3x3 sampled regions. Each of the four regions is indicated using a separate colour

to improve the results obtained from the filter, by using a larger number of subregions [PPC07] or by adapting the filter to the local structure of the input [KKD09]. However, while these modifications to the original filter improve results, they also slow down the speed of the filter. Due to this problem, we use the original filter as speed is an important element within this research, especially as we intend to combine stylisations of different types.

We also implement an approach based on the technique used to produce **comic-book** like effects, in [WOG06], which employs a multi-pass system using a bilateral diffusion filter, colour quantization and edge detection. This filter does not try and re-create any particular painterly style; the effect is more stylised and comic-book like, and it is more complex than the Kuwahara filter. It was seen as a suitable style that could be used adaptively in real-time scenes to create a stylised focus. As is described in the paper, a high level of stylisation is possible with the techniques described, although some of the quality and the artistic authenticity of the full approach must be sacrificed to achieve real-time rates.

We also isolate a component of the comic-book approach described in [WOG06], a **colour quantization filter**, to investigate whether an effective style can be created using only that form of abstraction. Colour quantization reduces the number of colours used in the palette, which would help eliminate unnecessary detail in background objects.

Figure 3.2: Edge detection examples; Difference of Gaussian (Left), Sobel (Centre), Canny (Right)

3.2.2 Edge Detection

As discussed in Chapter 2, edge detection and stylisation approaches have been heavily researched, and edges are an excellent way of directing user focus [HMH+03a]. Research also shows that users can interpret the 3D shape of an object well from line-drawings [CSD+09]. These results indicate that edge detection may be a good method for use in directing user focus to particular parts of a real-time scene, as well as helping user understanding of the shape of important scene objects. Image-space edge detectors are not as accurate as object-space edge detectors but are preferable as the latter are computationally more expensive to extract, a problem which is exacerbated in large scenes. A number of edge detectors, of varying complexity, were implemented in our system. This allowed us to choose suitable styles for particular scenes and also to investigate the trade-off between computational cost and effectiveness.

When applying image-space edge detection within real-time scenes, there can be a problem of flickering where edges are found in some frames and not others. When implementing each edge detector within our system, scales and parameters were chosen to minimise flicker while ensuring that the majority of important scene edges were found. This was important in producing temporally coherent results and ensuring that the flicker did not distract users and cause a confounding factor in experiments.

Difference of Gaussian (DoG) edge detection was implemented as it was seen as a good compromise between speed and accuracy. Difference of Gaussian edge detection works by simply applying a Gaussian blur to an image using two different blur levels. The resulting edges can then be found by calculating the difference between the two blurred images and finding all pixels with a difference under a certain threshold.

$$G(u,v) = \left(\frac{1}{2\pi\sigma_1}\right) e^{\frac{-(u^2+v^2)}{2\pi^2}} - \left(\frac{1}{2\pi\sigma_2}\right) e^{\frac{-(u^2+v^2)}{2\pi^2}} \quad (3.1)$$

where: G is the difference of Gaussian function,
σ is the standard deviation of each Gaussian distribution, and
(u,v) is the distance from centre of mask.

The Gaussian blur function is used for creating a Gaussian mask from a σ value. The mask is then passed through the image to blur each pixel. Each of the two blurred images is created using a different σ value before the difference between the images is calculated to find the edges within the original image. This function can be seen in function can be seen in Equation 3.1. The Marr-Hildreth [MH80] ratio of 1:1.6 between the blurred images was used to give the most effective result. Figure 3.2 (left) shows an example of difference of Gaussian edge detection.

The **Sobel** edge detection filter was also implemented in the system (see Figure 3.2 (centre)). Sobel edge detection is a simplistic and very quick edge detection filter which computes the colour gradient in horizontal and vertical directions. The operator uses two 3x3 kernels, which each convolve the image to calculate approximations of the gradient (See Equation 3.2). The resulting gradient approximations can then be combined using the formula in Equation 3.3.

$$S_x = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix} * A \qquad S_y = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} * A \quad (3.2)$$

where: S_x is the horizontal derivative approximation,
S_y is the vertical derivative approximation, and
A is the Source image.

$$S = \sqrt{{S_x}^2 + {S_y}^2} \quad (3.3)$$

where: S is the gradient magnitude.

A reduced form of the highly effective but costly edge detector, **Canny**, was also implemented. Canny edge detection requires a number of iterations over the image, with each iteration performing a different filter. First a Gaussian blur eliminates noise (Equation 3.1) before a Sobel filter is run on the image to find the basic edges (Equations 3.2

and 3.3). The edges are then thinned using non-maximal suppression, which operates on each edge dependent on its direction, and can be calculated by operating on the Sobel edge image using Equation 3.4. The calculated angle is then rounded to 0, 45, 90 or 135 degrees, before non-maximal suppression takes place by eliminating any edges that are not at a local maximum in the given direction. Finally, the edges are traced and thresholding takes place using hysteresis.

The final step, tracing through the edges and using hysteresis thresholding, is far too expensive to run in real-time and, therefore, it was not used. While the results are not as reliable as "full" Canny edge detection, the reduced method still provides good results, as shown in Figure 3.2 (right). As can be seen, there is much less noise in the Canny example in comparison to the difference of Gaussian or Sobel examples. The non-maximal suppression is also obvious in this example which is responsible for producing thin edges.

$$\Theta = arctan\left(\frac{S_x}{S_y}\right) \tag{3.4}$$

where: Θ is the gradient direction.

3.2.3 Shading Stylisation

Image-space NPR techniques can be very effective when abstracting the colour and texture information, but they fail to take into account any object-space information, which is vital in 3D scenes. To convey the shape of an object reliably it is necessary to use shading upon the surface of the object. Stylistic shading is a well-researched part of non-photorealistic rendering and the available methods may be valuable tools for altering how objects are perceived. We hypothesise that by altering shading styles within a real-time scene, we may be able to improve user understanding of particular objects. In addition we believe that some shading styles may be suited to altering object saliency by quantising the amount of shades within an object, thus increasing the colour contrast and therefore, object saliency. However, stylistic shading is an approach that cannot be implemented in image-space, which means that frame-times may be hard to guarantee for many of the published techniques.

We implemented two shading styles that are relatively fast compared to other approaches, such as hatching or cool-to-warm shading, and yield diverse and potentially valuable results. The first, **cel-shading**, is a simple style that first calculates the shading value at each vertex using the well-known Gouraud shading algorithm. The shading value obtained is then clamped to one of n discrete values, where n is the number of shades desired. This gives the effect of a cartoon shaded result (see Figure 3.3(d)).

A **pencil-shading** style is also implemented to create stylistic results. The technique is a variation on the pencil shading technique presented by Lake et al. [LMHB00]. The pencil shading style is similar to cel-shading in that the shading value at each vertex is calculated and the resulting value is then quantized into the desired numbers of levels. However, with the pencil shading approach, if a polygon has different shading values at its vertices, linear subdivision is applied to that polygon until all vertices within each polygon match the same shading value. Then depending on which level the shade value falls within for each polygon, a corresponding pencil shade texture is applied to that polygon. This effect gives the illusion of the scene being hand-drawn by pencil (see Figure 3.3(c)). A similar technique is presented in [PHWF01], using tonal art maps to produce hatching effects.

It should be noted that fixed polygon texture coordinates are used in preference to projecting the texture onto the object from the camera viewpoint. This is to preserve frame-to-frame coherence and avoid the shower-door effect. The shower-door effect, described fully in [Mei96], is caused when textures are applied to an object from the camera's viewpoint, then as the object moves, the texture seems to "swim" over it. This can cause poor temporal coherence such as in [LMHB00].

While the two styles described produce a stylistic rendering of scenes, a lot of scene data can be lost, especially in scenes with a large amount of texture information. This can be seen in Figure 3.3(c) and (d) where the shading styles fail to represent the colour and texture information for the characters and buildings contained within the scene. Figure 3.3(b) shows the advantage of using image-space approaches in this way, as the Kuwahara filter preserves the colour and texture information within the scene after stylisation.

3.3 Scene Segmentation

We manipulate the real-time styles, described in the previous section, to allow for each approach to produce different levels of abstraction (described later, in Section 3.5.1). These different abstraction levels can then be used simultaneously within one scene to stylise certain information more than others. While image-space approaches are suitable for creating real-time abstractions, the lack of object-space information results in poor segmentation possibilities within the scene. To adequately render a scene using such an approach and render each object in a suitable style, some scene information must be known.

To allow for such scene information to be used, we implemented a fast and reliable

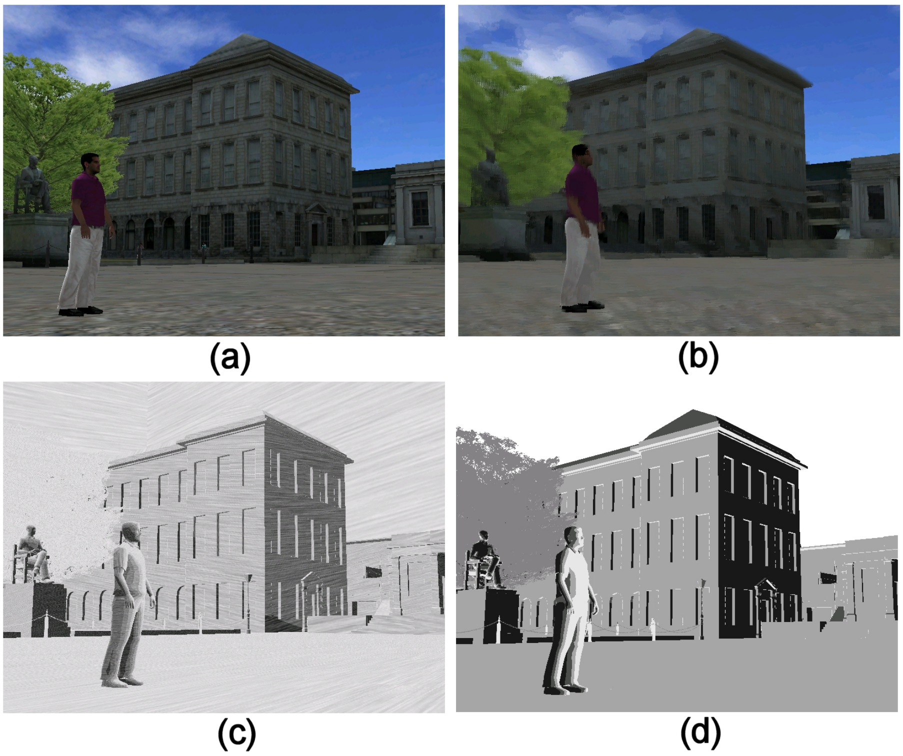

Figure 3.3: Various types of abstraction; (a) Original rendering of scene (b) Kuwahara filter (c) Pencil-shading (d) Cel-shading. As can be seen, the shading styles do not take into account colour and texture information, as the painterly approach does.

Figure 3.4: Example of the false colour pass; Left: Original rendering. Right: False colour rendering

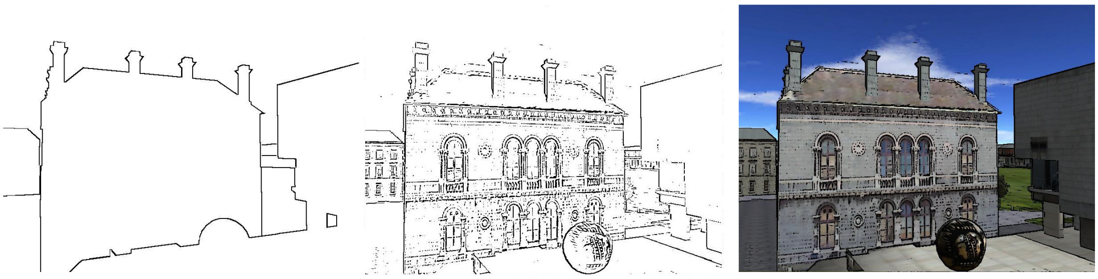

Figure 3.5: Edge detection example: 3D Silhouette edge image, from false colour rendering pass (Left), Image space edge image (Middle) and the two overlaid on the original image to display all scene edges (Right)

method, which uses a false colour rendering of a scene to guide abstraction and find edges (see Figure 3.4). False colour rendering is a popular technique used in for object identification [NRTT95] and techniques in NPR use a similar ID-buffer to segment a scene, e.g., in [VB99], to give each object a different halftoned texture.

To create this false colour rendering, we use a two-pass system. We first render the scene with each object rendered in a distinct colour. The amount of unique colours available depends on the depth of each component in the colour buffer. When rendering during the false colour pass it must be ensured that all objects are rendered in flat colour, therefore lighting, textures etc. must be disabled. The false colour rendering pass is a useful tool for identifying which object is present at each pixel position. This information can then be used to render each scene object in a suitable style. The simplicity of the rendering pass makes it fast enough for interactive applications.

This false-colour rendering pass is designed to help scene segmentation; however, it can also be useful for increasing the amount of significant edges found when using the edge filters. Despite their advantages, image-space edge detectors can be unreliable due to the fact that some 3D edges can be missed because of colour similarities. We employ a hybrid edge detection system that uses both object-space and image-space techniques to reliably find edges in the scene, including 3D external object silhouettes and edges that exist within textures. We combine image-space edges, found by using image-space filters on the framebuffer, with external silhouettes, found by detecting edges within the false colour rendering.

The Difference of Gaussian edge detection is run for the framebuffer image to find the majority of the scene edges, including any edges that might occur within object textures. The same DoG edge detection process is then run on the false colour image generated from the first-pass rendering. By doing this, more reliable external silhouettes edges are found for scene objects, although any silhouettes caused by self-occlusion are not found. As can be seen from Figure 3.5, the 3D silhouette image finds edges which the image-space detector might miss. In the example edges are missed particularly around the roof. Implementing these two edge detection passes gives a reliable method of finding more significant edges within a scene. In past research, a similar approach was used by Kolliopoulos et al. where 2D image segments were generated to provide segmentation information for 3D scenes [KWH06].

There are other hybrid methods for finding silhouettes, such as using the depth image as input to an edge detector [ST90] [ND03], although these methods can be unreliable as they depend on the accuracy of the depth buffer. This approach can miss silhouette edges when two nearby objects are at the same depth level. This can be a problem in interactive scenes, especially within large scenes, where the depth buffer can become less accurate.

DoG edge detection is run for both the framebuffer image and the false colour image using our technique, although a different edge threshold is used for each image. This is because all edges from the simple false colour image are desirable, whereas only the strongest edges from the framebuffer image are needed. DoG edge detection can be implemented on a pixel-by-pixel basis making it ideal for GPU pixel shaders. The edges found in the edge detection process can then be overlaid on the original image to give convincing results containing all scene edges (see Figure 3.5).

3.4 Combining Styles

The image-space styles presented in Section 3.2.1 and Section 3.2.2 can guarantee real-time rates when speed is dependent on screen resolution and not scene complexity, as is the case with object-space approaches such as stylistic shading. While we have shown that image-space styles have much promise for creating multiple levels of abstraction, it may also be possible to use some object-space styles and use more object information to better represent the whole scene. We already take into account some object-space information to segment the scene, as described in the previous section, but other approaches such as object-space shading, described in Section 3.2.3, may add valuable cues to the image-space styles.

There is currently a lack of approaches that use both object-space shading information and image-space colour information to produce results. If either of these types of information is not used, it can lead to valuable scene information being ignored. This is especially true for interactive scenes when reliable abstraction can be traded off for speed. It would therefore be of interest to see if the object-space shading styles described in Section 3.2.3,

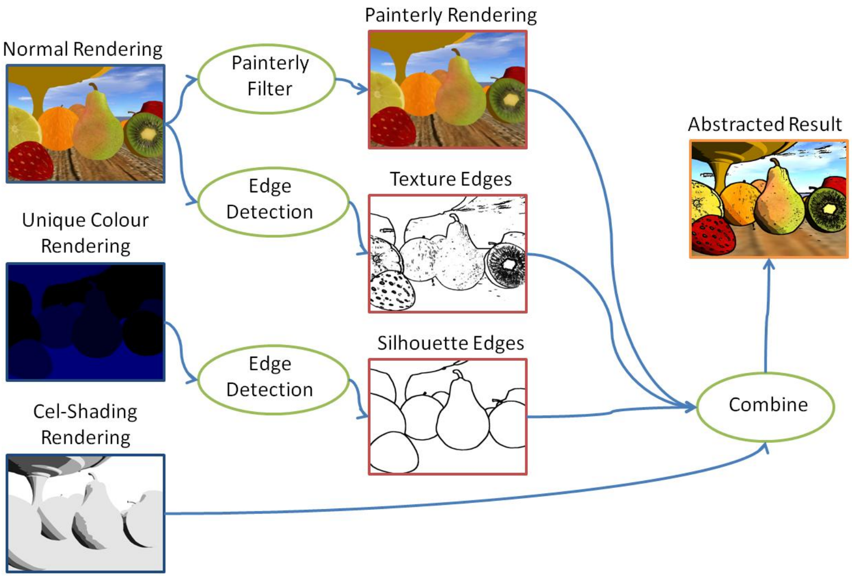

Figure 3.6: Workflow for combining object-space shading and image-space abstraction and edges. The example cel-shading effect is used to demonstrate the workflow.

such as cel-shading or pencil shading can be combined with the image-space approaches while retaining real-time rates.

The approach is based on the premise that utilising more scene information will yield better NPR representations. Employing the combination of both object-space and image-space information as described here allows for the creation of real-time effects which are aesthetically pleasing and also accurately represent scene objects. The technique could be used to create a large number of effects, using various combinations of image-space abstraction methods and object-space shading approaches.

Two example effects are presented as a case-study to investigate the advantages of such an approach. By creating those example effects, we are able to show how real-time rates can be achieved using the framework. A hybrid toon effect is shown that uses three individual styles; these are cel-shading, image-space painterly abstraction, in the form of the Kuwahara filter, and edge detection. A hybrid pencil-sketch effect is also described that uses pencil-sketch shading combined with the painterly style.

Using the approach, image-space abstraction techniques can be applied to the images from the framebuffer to add colour and texture information to shading information, which is calculated earlier in the pipeline. The Kuwahara filter, as described in Section 3.2.1, was employed in the example effects presented. This style gives the impression of a painted image as can be seen in Figure 3.3(b) and eliminates some of the fine detail within textures

Figure 3.7: Examples demonstrating the advantage of the hybrid edge detection approach;(a) Scene rendered using Sobel image-space edges (b) Scene rendered using both image-space edges and silhouette edges found using unique colour ID rendering pass

which can look out of place within non-photorealistic styles, while retaining all important scene information.

While the examples described in this section exhibit the strengths of the framework, alternate effects could easily be created using different object-space shaders or image-space filters. The workflow of the framework is shown in Figure 3.6, using the presented cel-shading style as an example. First, the scene is rendered normally without any shading, then the false colour rendering pass is calculated. Another pass then renders the scene using the cel-shading style. The image-space abstraction techniques (the painterly filter and edge detectors) are then run on the outputs of the rendering passes. Finally, the image-space and object-space results are combined to create the desired effect. The workflow is then slightly changed for the pencil shading effect, removing the edge detection style from the final effect (Figure 3.8).

3.4.1 Example Effect 1: Hybrid Cel-Shading

Image-space edge detection was included in the first example effect to add stylistic and potentially perceptually influential lines to the scene. As described in Section 3.4, the edge detection filter was used both on the original rendering pass and the false colour rendering pass to get more reliable results. Adding lines from the false-colour rendering was especially beneficial when dealing with multiple objects of the same colour. This improvement can be seen in Figure 3.7. Each of the edge detection styles described in Section 3.2.2 was tested with the cel-shading effect to see which was most successful at simulating suitable artistic outlines. It was desirable for the effect to provide reliable edge detection as well as fitting in with the style of the shader. For this reason, Sobel edges were chosen as it was decided that the bold simplistic edges complemented the cartoon

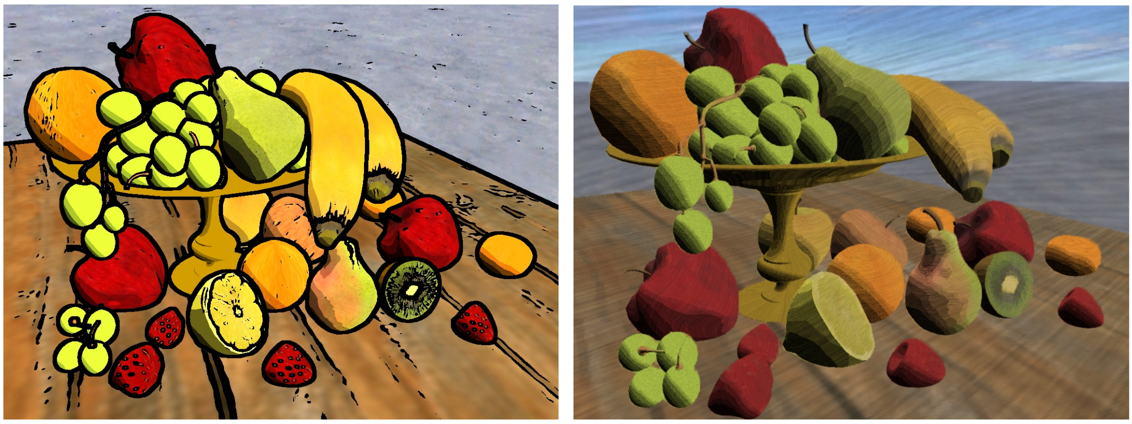

Figure 3.8: Abstraction examples: (a) Hybrid cel-shading effect (b) Hybrid pencil-shading effect

effect most effectively.

$$F(i, j) = min((P(i, j) * \epsilon * C(i, j)), E(i, j)) \tag{3.5}$$

where: F is the hybrid cel-shaded image,
ϵ is the brightening multiplier,
P is the painterly image,
C is the cel-shaded image, and
E is the edge image.

To create the example cel-shading style, the results obtained from the object-space cel-shading pass are combined with the image-space edge detection and painterly filters. The painterly filter is used to eliminate extraneous detail from the final results and enhance stylisation while keeping the basic colours and shapes of scene textures. The renderings are combined by multiplying the pixel values from the cel-shading and painterly images together as can be seen in Equation 3.5. The result is multiplied by a factor of ϵ, as it was found that brightening the combined image enhanced the visual effect to make the cel-shading clearer and the result more stylised. A ϵ factor of 1.5 was used in all images in this section.

3.4.2 Example Effect 2: Hybrid Pencil-Shading

The second example style is created by combining the pencil-shaded values with the painterly rendering. The style of the result varies depending on how the two images are combined, as described in Equation 3.6. The result gives a coloured pencil effect, which can be seen in Figure 3.8(b). This example uses a pencil-shading factor of 2.0 when combining the image to get the most aesthetically pleasing results for this scene. For this effect, no edges were added. This is because it was thought that edges did not suit the pencil-sketch effect and also, we wanted to investigate the differences in future tests between the styles when edges were present in only one of the effects.

$$L(i,j) = (P(i,j) * S(i,j)^{\gamma}) \tag{3.6}$$

where: L is the hybrid pencil-shading image,
P is the painterly image,
S is the pencil-shaded image, and
γ is the pencil-shading factor.

3.5 Creating an Adaptive Abstraction

The goal of this chapter is to use non-photorealistic abstraction to produce an adaptive abstraction that emphasises vital features of an image and reduces the impact of unimportant information. To do this, varying levels of abstraction must be achieved for each of the styles, which have been described in Section 3.2. On top of this, methods must be created for applying those levels to a scene in a meaningful manner, which can be done by utilising the segmentation information described in Section 3.4. By bringing these elements together we can create an effective and stylistic focus in interactive scenes.

3.5.1 Multi-level abstraction

To create an adaptive abstraction, we must first alter each style to enable it to alter its abstraction throughout a scene. For the Kuwahara filter, using varying sampled region size can give different abstraction levels within the scene. Large adaptations of the Kuwahara filter, such as 9x9, produce more smoothing but do not preserve edges as accurately. Large filters are also significantly slower, as described in papers that use the Kuwahara filter as a basis for more convincing painterly styles [KKD09, PPC07]. If a high level of abstraction is required eserand real-time rates are still necessary, it is preferable to apply a smaller filter iteratively (e.g., 5x5). This has the effect of preserving edges and increasing the abstraction. Using various filter sizes in combination with iterative application of the filters can produce many different levels of abstraction (see Figure 3.9).

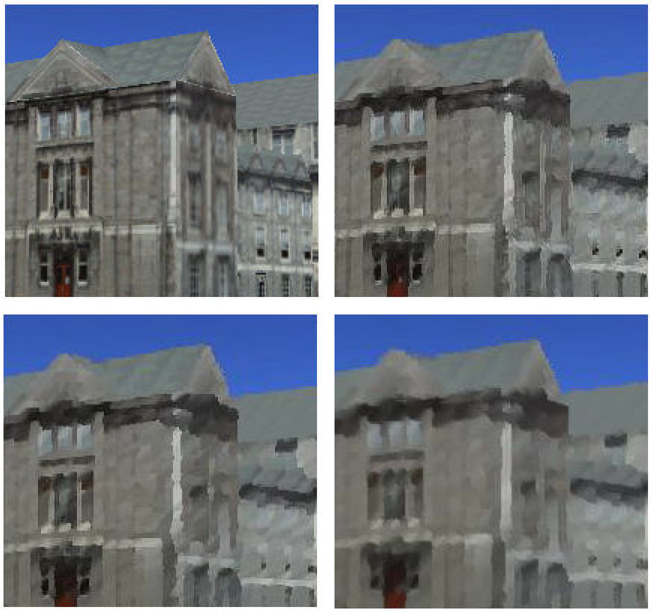

Figure 3.9: A section of an image when there is no abstraction(top left), a 3x3 Kuwahara filter(top right), a 3x3 Kuwahara filter applied twice(bottom left) and a 5x5 Kuwahara filter(bottom right)

For the image-space comic-book filter, we can alter parameters within the algorithm to create adaptive levels of abstraction. This has the effect of preserving edges better in lower levels of abstraction while getting more of a blur effect in high levels. It is also possible within the algorithm to change the bin widths of the colour quantization filter to alter how extreme the comic-book effect is.

Creating adaptive levels for each of the edge detectors can be done by altering the parameters within the filters to display only the most prominent edges for low abstraction levels and all edges for high abstraction levels. This can then be used to increase the number of edges found in target objects to attempt to increase their saliency. These parameters can be altered for difference of Gaussian, Canny and Sobel edge detectors.

Colour variation algorithms are also implemented to further affect object saliency. The brightness of an object can be altered by uniformly changing the RGB values of the object. It has been shown that brightness variation causes an object's colour to fade, and therefore lowers its visual saliency in the scene [KV06b]. It is also possible to alter colour attributes using the HSL (Hue, Saturation and Luminance) colour model, which is a more perceptually relevant and more intuitive model. The conversion from RGB to the HSL colour space allows for easy adjustment of effects like luminance and saturation variation.

There is a question as to how best to influence user attention using colour quantization and quantized shading styles such as cel-shading. When utilising these types of styles, high levels of stylisation for extraneous objects preserve detail better for target objects while abstracting extraneous objects. However, this may actually increase colour contrast for these unimportant objects, making them stand out visually in the scene. It may be more effective to stylise the target objects to increase colour contrast and therefore make them more salient within the scene.

To examine the effect of colour quantization, two styles were used with different approaches. The isolated colour quantization filter was tested with a decrease in the number of colours for high levels of abstraction in order to attempt to simplify unimportant objects. The cel-shader creates more shading levels for less important scene objects, making their shading smoother while leaving the number of cel-shades for the important scene object low. This is to test whether the increase in colour contrast improves the visual saliency of the target object in comparison to the rest of the scene.

The nature of the pencil-sketch shader makes it difficult to alter attributes to create multiple levels of abstraction. There is no way to easily create such levels, apart from using less dense sketch textures for higher levels of abstraction, although it is doubtful whether this would be an obvious enough change to affect user perception.

The hybrid effects created in Section 3.4 are adapted by varying the blending ratio between the stylistic shader and the painterly abstraction when the two images are combined. This allows for the shading to be displayed with varying prominence. For the hybrid cel-shading effect this can be done by altering the brightening multiplier in Equation 3.5, combined with altering the number of cel-shades. For the hybrid pencil-sketch effect, adaptive levels can be obtained by altering the pencil-shading factor in Equation 3.6. Increasing the amount of painterly style shown in the final version of the pencil-sketch effect can add more colour and detail to the object thus increasing the visual appeal.

3.5.2 Stylised Focus

It is desirable to render each object in an abstraction style that suits that object's importance in the scene. This allows the effective rendering of scenes that we hypothesise will draw a user's attention to certain objects within the scene. We propose an object classification technique for doing this based around the most visually important objects within the scene. These visually important objects can be pre-defined, set by the current mouse position or image centre, or be dependent on a user's current goal or interest.

The information gained from the false colour rendering can be used to create a stylistic focus in a number of ways. A simplistic method is to use a two-level approach where

Figure 3.10: An example of a scene with object-based segmentation. One object is rendered clearly while the rest of the scene is abstracted. The Kuwahara filter, difference of Gaussian edges and saturation variation were used in this image (two levels of abstraction).

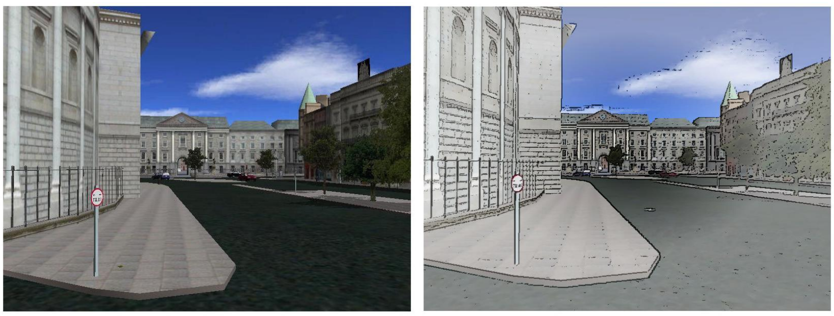

Figure 3.11: Example of stylised focus without scene segmentation information. Left: original rendering, Right: a 2D distance based abstractive rendering using difference of Gaussian edges and brightness variation (8 levels of abstraction).

pre-defined objects of interest are identified by their unique colour ID and rendered in an enhanced style while abstracting the rest of the scene using a high level of stylisation. This creates a high contrast between target objects and the rest of the scene, which increases the saliency of the target objects. The resulting images appear to be quite effective, although the large contrast between the two levels can make the scenes look unnatural (See Figure 3.10). While this technique is not always suitable, it is hypothesised that it may be sufficient for scenes dominated by a single object.

Another adaptive abstraction method is to render the target objects in a enhanced style but then use a number of levels of abstraction in the rest of the scene to create a smooth stylised focus on the target objects rather than one harsh contrast. This can be done using simplistic screen distance from the centre of the target object to decide on which abstraction level should be used (Figure 3.11), although a drawback to this approach is that by altering abstraction dependent only on screen distance, renderings can have a flat, depthless feel. We propose a system that uses discrete levels of abstraction to render each object in a suitable style, by calculating which level every object should be rendered at.

The first step to doing this is calculating the centre of each visible target object in screen-space. After this is done, the abstraction levels for each of the remaining scene objects must be calculated. The approach uses the false colour image and distance information for each pixel to calculate the average distance from each regular scene object to the centre point of the target object. This is done by first inspecting the false colour image to identify which pixels belong to a certain object. These pixels are then used to calculate an average distance to the target object. Depending on the result, each object can then be quantized into one of a number of abstraction levels. This information can then be passed to the pixel shader to do the necessary abstraction.

We also implemented a similar method that used object-space distance rather than image-space distance. This approach calculates the centre points of each object in object-space and again calculates the distance between each object, although for this method the distance is in world-space and not pixel distance. The objects are then rendered using the unique colour ID as per usual, but the ID also stores the distance to the target object, allowing the styles to access the information in image-space as an input to the abstraction method. An example of this object-space distance approach can be seen in Figure 3.12(b).

When using multiple degrees of abstraction, rather than just two, smooth abstraction transitions are possible within a scene. Within the initial system, five distinct levels were used as we found it to be a sufficient number to successfully abstract a scene smoothly while also abstracting the background areas enough to create an effective stylised focus.

Figure 3.12: Adaptive abstraction examples. (a) Two-level adaptive abstraction using Kuwahara filter, with focus on the character in the background wearing stripes (b) Multi-level adaptive abstraction using adaptive hybrid pencil shader, using object-space distance as a guide for abstraction. The focus is on the character in yellow in the foreground

Figure 3.13: Examples of three of five abstraction levels using the Kuwahara filter, Difference of Gaussian edges and brightness variation. Left (level 1, most salient), Middle (level3) and Right (level 5)

It is possible to introduce more levels of abstraction if necessary. An example of three of the finished abstraction levels, using a combination of the Kuwahara filter, Difference of Gaussian edges and brightness variation can be seen in Figure 3.13. It is also possible to use the adaptive abstraction approach with a larger number of abstraction levels (Figure 3.12(b)) and as low as 2 levels of stylisation (Figure 3.10 and Figure 3.12(a)).

3.5.3 Frame Times

The system was integrated with a number of different types of applications, including the Metropolis system, which is a large-scale immersive urban simulation with realistic crowds, developed in Trinity College Dublin. The frame-times in Table 3.1 are for a version of this system that contains over 120,000 polygons. This system was chosen as an initial test system because the abstraction methods could be used effectively with a system such as this to improve pathfinding or highlight particular objects such as historical landmarks or popular sights for tourism purposes.

	FPS for different resolutions		
Rendering Style	640 x 480	800 x 600	1280 x 1024
Photorealistic rendering (no abstraction)	317	301	206
Kuwahara painterly filter	107	66	27
Comic-book filter	95	61	25
Sobel edge detection	246	228	138
Difference of Gaussian edge detection	228	182	92
Canny edge detector	177	132	59
Pencil shading (object-space only)	154	136	79
Cel shading (object-space only)	309	272	184
Hybrid pencil effect	96	73	41
Hybrid toon effect	88	71	38
Two-pass rendering with false colour pass	194	150	96
Adaptive Kuwahara (5-levels)	78	50	22
Adaptive Sobel (5-levels)	168	142	79
Adaptive Canny (5-levels)	125	95	46
Adaptive hybrid toon (5-levels)	51	34	16
Adaptive hybrid pencil (5-levels)	51	38	19

Table 3.1: Frames per second achieved by the system using various abstraction options on a complex urban simulation (over 120,000 polygons) at different resolutions. These frame times were calculated using a Pentium D, 3.72GHz computer with 2GB of RAM using an NVidia GeForce 7950 GX2 GPU

We implemented the styles using shaders written in CG on GPU pixel shaders. The implementation of the abstraction techniques on the GPU coupled with the fast false colour rendering pass means that interactive frame times are achieved for all approaches, which can be seen from Table 3.1. This is even true for the hybrid effects described in Section 3.4, which use one object space shading stylisation and two image-space styles each. The hybrid toon effect is created by combining cel-shading, Kuwahara filtering and Sobel edge detection. The hybrid pencil effect is made up of pencil-shading, Kuwahara filtering and saturation variation. Both these example effects are shown to be able to run on a complex scene at frame rates of over 35 fps for large resolutions.

Table 3.1 also shows examples for various adaptive abstraction approaches that use the false colour rendering pass (the final five styles in the table). As can be seen, there is a slow-down, albeit minimal, when calculating the false colour pass. The simplicity of the pass means that real-time rates can be retained. The most computationally expensive techniques are the adaptive hybrid styles, although they still run at interactive rates.

3.6 Summary and Conclusion

In this chapter, we have presented a novel adaptive abstraction approach for generating meaningful renderings of large 3D scenes, using non-photorealistic styles in real-time. Our main contribution is a hybrid method for rendering each scene object in a particular abstraction level which is suitable for its importance in the scene. This adaptive abstraction approach is created by manipulating existing NPR techniques to create multiple levels of abstraction. Scenes can then be rendered in real-time with each scene object displayed using an appropriate, importance-driven stylisation level, which is dependent on that object's importance within the scene.

We have shown how stylistic and temporally coherent results can be created, while retaining all important scene information. A variety of image-based abstraction methods can be used individually or in combination with each other or with stylised lighting approaches to create final renderings. We also present a novel hybrid method for effectively combining object-space edges and edges that exist within textures, to find all important scene lines.

The system provides variable abstractions of scenes in real-time, although to fully understand the benefits and drawbacks of the system, there must be a number of tests and perceptual evaluations run on each style. The next chapter describes validation approaches such as task-based experiments and eye-tracking studies, which would allow for comparison and evaluation of a variety of styles. This will help to give an accurate measure of the effectiveness of the abstraction system.

Chapter 4

Perceptual Evaluation

4.1 Introduction

Chapter 3 introduced an approach for creating adaptive abstractions using real-time non-photorealistic styles. We hypothesise that by using such an approach, user perception can be influenced. However, quantitative evaluations must be run to prove that this is true, as well as to investigate the details and limitations of such techniques. Detailed user studies are required to understand how NPR styles can influence a user in real-time scenes. We are interested in measuring how a number of real-time styles perform in perceptual experiments to compare styles and find the most effective techniques.

A large number of NPR techniques have been proposed in past literature to improve the effectiveness of images in conveying informational content and drawing user attention to important objects in an image. However, with the wide range of available stylisations, each with their respective objectives and side-effects, measurable indicators of the effectiveness of each NPR style are vital in optimising the perceptual fidelity of interactive NPR systems and in particular, our adaptive abstraction approach.

In this chapter we investigate the success of the real-time NPR approaches, described in Chapter 3, and perform a number of user studies to evaluate how a number of non-photorealistic styles perform in affecting user task performance. Determining the effectiveness of various types of non-photorealistic abstractions should enable us to develop the most efficient NPR techniques for increasing the saliency of important scene objects. It is vital that we understand in detail, the limits and strengths of each non-photorealistic approach. With this in mind, we address the following questions in this chapter:

- Can saliency tests and various types of user studies give a basic indication as to the success of the adaptive abstraction approach?

- Can user task performance be improved using such adaptive abstraction?
- Do certain styles perform better than others in influencing user behaviour?
- Can styles be altered by varying parameters to achieve the best results in this context?
- Does the scene type affect how successfully each style can influence user behaviour?
- Can adaptive abstraction be successful in influencing user behaviour in complex interactive scenes?

4.2 Experimental Setup

Each of the experiments performed in this chapter is listed in Table 4.1, showing what styles were tested and what task was given to the user in each experiment. Before performing the user studies, we used an automatic saliency metric to get preliminary insights into the success of the adaptive abstraction approach in altering object saliency (Section 4.3). These early tests gave us an indication of which styles perform better than others in affecting saliency. We then performed a two-alternative forced choice task, which showed that task performance can be improved using the adaptive abstraction approach and that certain styles outperform others in improving search times (Section 4.4). A set of timed search experiments then investigated which styles and combinations of styles were the most effective in improving user search times (Section 4.5.2). Eye-tracking studies then examined how each style could affect user gaze behaviour and whether there was a correlation between gaze behaviour and the effects on task performance seen in previous experiments (Section 4.5.3). The next experiment investigated how scene type and style parameters can affect the success of each style in influencing user behaviour (Section 4.6). The final experiment in the chapter tested whether adaptive abstraction can be effective in a complex environment when the user has interactive control (Section 4.7).

Each of the experiments, apart from the final experiment (Section 4.7) was run on a workstation with 2GB of RAM, a 7950 GeForce graphics card and a 19 inch LCD monitor. The final experiment was run using an NVidia ION graphics card, 2GB of RAM and a 17 inch LCD monitor. For our analysis of each experiment, we used ANalysis Of VAriance (ANOVA) with a varying number of factors. Post-hoc analysis was conducted using Newman-Keuls tests for comparison of means and only significant results at the 95% level are reported. A number of the experiments were performed using the Metropolis system, an urban simulation developed in Trinity College. Within each of these experiments, the

Experiment	Task	Purpose	Styles
Saliency Tests (Section 4.3)	Automatic Saliency Metric	To get preliminary insights into the effectiveness of each style in affecting object saliency	Kuwahara filter Comic-book filter DoG edges Sobel edges Saturation variation Luminance variation Brightness variation Colour quantization Hybrid cel-shading Hybrid pencil-shading Combinations of above styles
2AFC (Section 4.4)	Pick out particular character from one of two groups	To investigate how hybrid styles can affect task performance	Normal rendering Kuwahara filter Hybrid pencil-shading Hybrid cel-shading
Search Task 1 (Section 4.5.2)	Find particular sphere in a rotating animation	To analyse how each individual image-space style could affect search task performance	Normal Rendering Kuwahara filter Comic-book filter DoG edge detection Saturation variation Colour quantization Brightness variation Luminance variation
Search Task 2 (Section 4.5.2)	Pick out particular sphere in static image	To investigate how combinations of styles can affect search task performance	Normal Rendering Kuwa/Sat/Bright Kuwa/DoG/Quant/Lum Comic/Sat/Lum Comic/DoG/Bright
Search Task 3 (Section 4.5.2)	Pick out particular sphere in static image	To isolate what styles perform better than others	Normal Rendering Kuwa/Sat/Bright Kuwa/Sat/Bright/Quant Kuwa/Sat/Bright/DoG
Eye-Tracking Study 1 (Section 4.5.3)	Count instances of particular sphere in rotating animation	To investigate if user eye gaze behaviour correlated to previous search time performance	Normal Rendering Kuwa/Sat/Bright Kuwa/Sat/Bright/Quant Kuwa/Sat/Bright/DoG
Eye-Tracking Study 2 (Section 4.5.3)	Observe urban walkthrough	To see how each individual style performed in affecting eye-gaze	Normal Rendering Kuwahara filter Comic-book filter DoG edge detection Saturation variation Colour quantization Brightness variation
Matching Pair Experiment (Section 4.6)	Find matching pair in grid of objects	To see how scene type and style parameters affected success of each style	Normal Rendering Kuwahara filter Saturation variation Hybrid cel-shading Sobel edge detection
Interactive Search Experiment (Section 4.7)	Find two instances of a character in complex interactive scene	To see if past results held true in a more complex, interactive environment	Normal Rendering Kuwahara filter Saturation variation Sobel edge detection

Table 4.1: List of each experiment in the chapter, showing the section it is in, the task given to the participant, the styles tested and the reason for the experiment

ground plane was rendered using only one level of stylisation. This was done to avoid a halo effect on the cobbled street texture around target characters when using the adaptive styles, which would have caused a confounding effect. In each of the experiments we tested the effect of adaptive abstraction when applied to scenes on a per-object basis. Although we later show it is possible to use adaptive abstraction to enhance certain parts of an object (Chapter 5), we first investigate whether user gaze can be drawn to whole objects.

We anticipate that each of the types of styles implemented will have a differing effect on user perception, with some styles performing better than others in influencing user gaze behaviour and task performance. We make the following hypotheses:

- We hypothesise that through the use of *edge detection*, we can add bold edges to target objects which will increase their saliency and influence user gaze and performance in search tasks. We suggest that edge detection will be one of the most effective of the proposed styles in influencing user behaviour, due to results found in past research [CKD96]. We also believe that the addition of edges can help users understand the shape of target objects.

- We believe that through the use of *colour variation* techniques, we can reduce colour contrast in background objects to make target objects more salient to the human visual system. We expect that colour variation approaches will perform consistently well in drawing user gaze to target objects, although the appearance of such styles does not seem to be as aesthetically pleasing as other styles.

- We hypothesise that by using *painterly abstraction* for background objects, we can lower the saliency of unimportant scene data, thereby leading user gaze to be drawn to the target object, improving search times and task performance. We anticipate that painterly rendering will be a more subtle way of influencing user behaviour in comparison to the others styles.

- As discussed in Section 3.5.1, it is interesting to discover whether quantizing colours and shades within an object either increases colour contrast and therefore saliency, or whether the resulting loss of detail actually reduces the object's saliency.. We intend to test this using differing adaptive abstraction approaches for *stylistic shading* and *colour quantization*. We hypothesise that the increase in colour contrast will outweigh any decrease in detail, which will result in objects becoming more salient with colour and shade quantization.

4.3 Automatic Saliency Tests

Through studies and experiments we aim to evaluate how real-time adaptive abstraction can alter the visual saliency of objects within a scene. To gain preliminary insights into the success of various non-photorealistic styles in influencing user gaze, we used a saliency mapping algorithm [IKN98]. This technique uses a multi-scale approach for finding the location of salient regions based on local image structure. Regions of high contrast are found for colour, intensity and orientation of an image on a number of scales. The results are then normalised and summed to create the final saliency map for an image.

The saliency metric was used to get an early insight into how well the adaptive abstraction approach can influence scene saliency. However, results from the saliency metric

Figure 4.1: Example of two scene images with and without abstraction. The yellow markers indicate the most salient points of each image, as indicated by the saliency metric, described in [IKN98]. In each stylised image the most salient point is found to be the object which is focused on using adaptive abstraction. The right images use the Kuwahara filter, difference of Gaussian edges and saturation variation. These images show that the most salient point can be changed by using our adaptive abstraction approach.

Figure 4.2: Examples of the Kuwahara filter with (right) and without (left) abstraction. The yellow markers indicate the most salient points of each image, as indicated by the saliency metric. The most salient point in the stylised image is found to be the target object.

may not always give accurate results. This is because many of the styles implemented enhance contrast of the target object, which is a property that the saliency map is based on. This circularity may mean that there is some bias inherent in the tests. The saliency metric also measures only low-level saliency which means it doesn't take into account any semantic connotations that may exist in the scene. While these issues mean the results should be interpreted with care, we felt the metric would give sufficiently accurate results to gain preliminary indications of the success of the adaptive abstraction approach.

The saliency mapping algorithm confirms that our adaptive abstraction approach can successfully increase the saliency of certain objects within a scene. We tested a number of styles on two different scenes. We used the Metropolis system for testing the styles using the saliency metric. The urban scene was chosen as we wanted to investigate the saliency effects of the styles in an example application which we hypothesise would benefit from such an approach. We also used a simplistic grid of characters to see if saliency was affected when the scene was a group of similar objects with no background distractions.

Figure 4.1 shows two scene images both with and without our abstraction methods. While all styles were tested with varying levels of success, the Figure shows the Kuwahara filter combined with difference of Gaussian edges and brightness variation. This figure shows that adding abstraction to a scene image can effectively change the most salient point to any particular object that needs to be emphasised. In this scene we used a number of levels of stylisation, on a per-object basis decided by image-space distance to the centre of the target object.

Each of the styles, described in Section 3.2, was tested with the saliency metric, as were various combinations of each type of styles. We tested each style using varied target objects. We also tested each style using both two-level abstraction and multi-level abstraction (5 levels), performed on a per-object basis. It was ensured that the target object was not the most salient point in the image before stylisation. Each of the styles and style combinations was tested using five test images or more. It was found that, over all styles, when multi-level abstraction was used, the adaptive abstraction was successful at altering the most salient point approximately 34% of the time. When two-level abstraction was used, the approach was successful approximately 63% of the time. The most consistent styles tested were the Kuwahara filter, difference of Gaussian edges, Sobel edges and saturation, brightness and luminance variation. All if these methods changed the most salient point to the target object over 70% of the time when using two-level abstraction.

The success of the Kuwahara filter in a simplistic scene is shown in Figure 4.2. In this scene a grid of characters is displayed over a white background. It is seen in this example, when a target character is reduced with minimal painterly abstraction and the rest of the scene is heavily abstracted, then one of the target characters becomes the most salient object in the scene. In this example only two levels of abstraction were used to test the approach, although multi-level abstraction provided similar, although not quite as consistent results. When testing the multi-level approach, using a stylised focus, nearby objects with slightly lower levels of detail were often found to be the most salient point. This was expected and indicates that a trade-off exists between high contrast two-level abstraction and more natural multi-level abstraction.

These studies gave us indications that some of the styles implemented can influence saliency more than others. Some styles, such as the Kuwahara filter, colour variation and difference of Gaussian edge images resulted in more consistent results than other styles, such as the hybrid shading approaches and colour quantization. These results allowed us make some design choices for future experiments, such as choosing to test difference of Gaussian edges for initial experiments ahead of other edge detectors like Sobel edges. Despite this, all styles are tested in some form in future sections, using perceptual experiments to ensure reliable comparisons could be made. While user studies would provide more worthwhile information about the effectiveness of our system these preliminary tests showed that our methods are successful at increasing object saliency and could be used to guide a user's focus in a scene.

Figure 4.3: An example image (Kuwahara filter without adaptive abstraction) from the 2AFC experiment. The target character is wearing a white shirt and green pants, and is located in the back of the left hand group.

4.4 Two-Alternative Forced Choice Task

A task-based test study was run to determine whether the hybrid styles with stylised shading (described in Section 3.4), which performed inconsistently in the automatic saliency tests, could affect user gaze patterns in real-time scenes and influence task performance in a timed search task. While the automatic saliency metric can give a good idea of how adaptive abstraction affects user gaze, it is preferable to test them using user studies in interactive environments. Twelve naive participants (7 Male, 5 Female) from different educational backgrounds took part in the study which measured user's speeds at recognising virtual characters. In the experiment users were asked to pick a particular character out of a crowd.

Each participant was first shown an image of the target character that was to be found in all trials. The character was the same for all trials and was present in each trial. We used a two-alternative forced choice paradigm. Participants were instructed to find the character on screen as fast as possible and indicate which side of the screen he appeared on. Participants clicked on the left mouse button if they saw the target character on the left of the screen, and the right if they saw him on the right. Upon choosing which side the character was on, the time and the result were recorded and the next trial began. The character appeared on each side of the screen an equal number of times with the trials randomly ordered.

Six types of style were investigated during the study. These styles were normal Gouraud

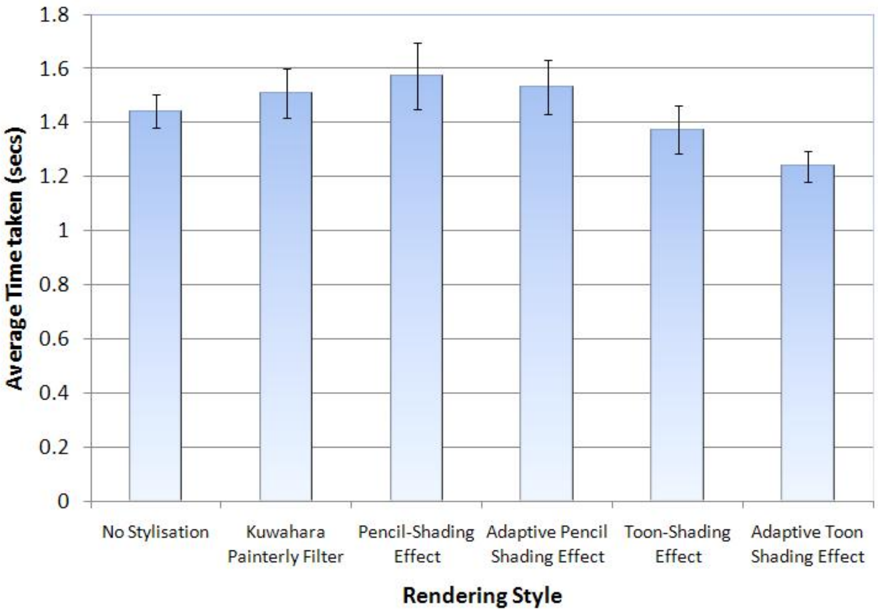

Figure 4.4: Average response times for each style from the forced choice experiment. The adaptive toon shading technique performed significantly faster than all other styles including standard rendering.

shading, one-level Kuwahara filter (i.e., no adaptive abstraction), one-level hybrid toon-shading, one-level hybrid pencil-shading, adaptive hybrid toon-shading and adaptive hybrid pencil-shading. In this experiment we were particularly interested in comparing the difference between styles with and without adaptive abstraction. The addition of the Kuwahara filter without adaptive abstraction was done to investigate whether the filter could affect user gaze without adaptive abstraction, as we had already seen that the style with adaptive abstraction performed well in the saliency tests. In the adaptive styles the target character was at the centre of the stylised focus, which was created using multiple levels of abstraction, calculated using object-space distance. We then used 10 repetitions for each style so therefore 60 trials were shown to each participant. An example image from the experiment is shown in Figure 4.3.

The reaction times of the participants were averaged for all styles, and a one way ANOVA was then performed on the results. Results showed (see Figure 4.4), that there was a main effect of style ($F(1;19) = 6 : 3778; p < 0 : 01$). Post-hoc analysis showed that the adaptive toon-shading technique ($t = 1 : 238$) performed significantly faster than all other styles in the task ($p < 0 : 05$), while the regular toonshading technique also performed well in comparison to the other styles, although not significantly so ($t = 1 : 374$).

Both the pencil-shader ($t = 1 : 572$) and adaptive pencil-shader ($t = 1 : 532$) produced times that were not significantly different from the one-level Kuwahara style ($t = 1 : 510$) or the normal rendering ($t = 1 : 442$). Some participants noted after the experiment that while the pencil style was aesthetically good, it was not as easy to pick out the character as in the cel-shaded styles. This may be because of the smoother nature of the pencil shading styles in comparison to the high colour contrasts in the cel-shading styles. Despite this, the study showed that the framework for combining image-space and object-space styles can successfully produce effects that increase user recognition times, particularly when there is stylised focus on the target object.

The results give us a good initial indication of the promise of using non-photorealistic rendering techniques in real-time scenes. In particular, it is seen how adaptive abstraction can be an effective tool for influencing user task performance. While the pencil-shading style produced poor results in influencing user performance, both with and without adaptive abstraction, the cel-shaded style performed well. We also saw that the Kuwahara filter without adaptive abstraction performed no better than normal shading; this indicates that the adaptive approach is important for influencing user behaviour, as the style on its own has no effect. This is an area that we would like to further investigate with more perceptual experiments. While this brief experiment gave a good idea of the possible success of using NPR styles to alter the saliency of scene objects, further experiments are necessary to investigate which styles are more successful at such abstraction.

4.5 Simple Context Experiments

The next set of experiments were designed to determine how effective a variety of styles were in affecting user scene perception in a simplistic scene with minimal semantic connotation. This type of scene was chosen as it was decided that within such a scene the effect of adaptive abstraction would be the major influencing factor, which would mean the styles could be conclusively isolated and examined. For this reason the styles were tested on a scene containing a number of simplistic textured spheres.

The series of experiments was conducted to investigate the effect of multi-level abstraction on a user's perception of a scene. In these studies the focus was on the performance of image-space styles and their performance with adaptive abstraction when the focus was either on the target object in the task or some other scene object. We also investigated the effect of combining such styles. This set of experiments consisted of five individual studies, composed of two distinct types. Firstly, three experiments analysed what effect various NPR techniques could have on a timed search task. The final two experiments used eye-tracking data to investigate whether NPR techniques had a major effect on gaze behaviour.

4.5.1 Rendering Styles

The previous section looked at the success of the stylised shading approaches in influencing task performance. That study showed the promise of adaptive abstraction and in particular, the success of the adaptive cel-shading approach in comparison to the pencil-shading style. In this set of experiments we focus on only image-space approaches, such as those described in Section 3.2, which use image-space stylisation coupled with object-space segmentation. We require a representative sample of specific classes of stylisations that are commonly employed. We identify these as edge stylisation, texture abstraction and colour variation.

As previously discussed, **edge detection** can play an important part in NPR. The initial experiments in the previous sections showed that emphasising edges in a scene enhances the distinction between regions and can increase the saliency of a particular object or area. The *Difference of Gaussian (DoG)* filter, applied to the frame buffer image of a rendering of the 3D scene, was chosen for this set of experiments as it produced good results in the initial saliency tests, in comparison to Sobel edge detection.

Painterly rendering, which we classify as **texture abstraction**, is the most well known device used in NPR. Two types of real-time painterly styles were implemented for our experiments. Both were described in full in Section 3.2.1. The first, the *comic-book style*, is based around an anisotropic diffusion filter and colour quantization. The second style we use, the *Kuwahara filter*, is a simpler edge-preserving, smoothing filter that also yields painterly-like results [KHEK76].

Several different styles were implemented for testing the impact of **colour variation**. One of these was a *colour quantization* filter, where colours are quantized into a set of bin boundaries whose size is dependent on the object's importance. We test how user perception changes when colour depth in unimportant objects is reduced whilst leaving the significant objects in full colour-detail. Secondly, a *saturation variation* style was used to reduce the colour intensity of particular unimportant scene objects. Finally two simple styles were additionally used in this experiment, classified under colour variation: one was a *brightness variation* style while the other was a *luminance variation* style. As discussed in Section 3.5.1, brightness is changed using simple operations on each of the pixel's three RGB values while luminance is a single value measured within the HLS colour space.

4.5.2 Timed search experiments

Three experiments were performed to determine whether real-time non-photorealistic rendering styles can have an effect on recognition speeds in interactive scenes. The experiments also measured which styles of NPR were more effective than others. Within the timed search experiments, scenes containing randomly generated textured spheres were used as they provided a scene with minimal semantic connotations. When there is little semantic information attached to the scene it allows us to obtain a good general indication of what effect various abstraction techniques can have on a user's behaviour. While more meaningful scenes are used in experiments described in later sections, the initial intention was to reduce the number of external factors that could affect the measured impacts on user behaviour. Previous published works have successfully employed a similar strategy of reducing scene content to simplistic objects, such as spheres, to effectively measure users' responses and draw useful conclusions about more general scenarios [OD01] [NLB+07]. In each of our experiments all participants had normal or corrected vision and were naive to the purpose of each experiment.

Experiment One

Procedure Twenty participants (12 male, 8 female) took part in this experiment. Each participant was shown a set of sixty-four animations each containing fifty spheres of the same size (see Figure 4.7). In each scene the spheres revolved around a vertical axis directly in front of the camera. The distance from the axis to the spheres was fixed, but otherwise their placement was random. This was to ensure that all spheres were visible for approximately the same amount of time while rotating. Each sphere was given a random texture and in each trial all spheres rotated around the centre of the scene at a constant speed in the same direction. Eight of the scenes shown were rendered using normal 3D local illumination throughout the scene with no stylisations, and the other fifty-six were rendered using seven different types of non-photorealistic abstraction (eight images per style). The styles used for the experiment are seen in Table 4.2.

Users were instructed to click upon a particular textured sphere as quickly as possible in each animation. The same textured sphere was the target object in each image, although in many of the images the focus was on a different object, leaving the target sphere abstracted using one of the styles. A brick texture was chosen for the target sphere as it was visually distinct from the rest of the textures. Before the experiment started the user was shown examples of the appearance of the sphere normally and in each of the possible styles (see Figure 4.5). It was ensured that there was no clipping of the

Figure 4.5: The range of possible target sphere appearances shown to users before all experiments. From left-to-right the styles shown are; Top Row: No stylisation, saturation variation, difference of Gaussian edges and brightness variation. Bottom row: Kuwahara painterly filter, luminance variation, colour quantization and the comic-book filter.

Style	*Style Description*
Kuwa	Kuwahara filter
Comic	Comic-book filter
DoG	Difference of Gaussian edge detection
Sat	Saturation variation
Quant	Colour quantization
Bright	Brightness variation
Lum	Luminance variation

Table 4.2: Description of the various styles used in Experiment one, within the simple context experiments.

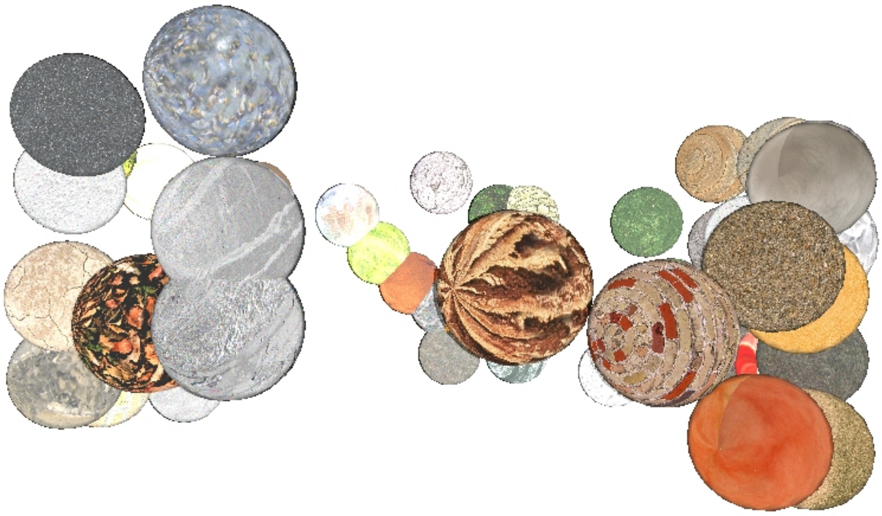

Figure 4.6: Example frame from Experiment 1 where focus is on the target sphere, which has a brick texture. The style used in this example is luminance variation (*Lum*).

target object in each animation and the animation played for a full scene rotation. The trials were also randomly ordered. The starting point of the sphere was random although it was ensured that the starting points were balanced throughout the experiment i.e. foreground/background and left/right. If the user successfully clicked on the target object a new trial began, otherwise the animation continued for a maximum of twenty seconds. Half of each of the sets of abstracted images used multiple levels of abstraction within the image to eliminate unwanted detail and focus on the target sphere i.e. increase its saliency (see Figure 4.6). The other half were focused randomly on a different scene sphere, thus abstracting the target sphere (see Figure 4.7).

Results The reaction times of the participants were averaged for all abstraction styles focused on the target and for the normal images, and a one-factor ANOVA was then performed on the results where the conditions were *Focused/Normal* (2). It was found that there was a main effect in the reaction times, for the class of scenes (normal or focused) ($F(1, 19) = 6.3778, p < 0.02$). Post-hoc analysis then showed that when the target object was focused, average recognition speeds ($2.53s$) were faster than normally rendered images ($2.79s$) (see Figure 4.8). These results indicate that the NPR styles, in general, used were effective at improving task speeds when adaptive abstraction focused

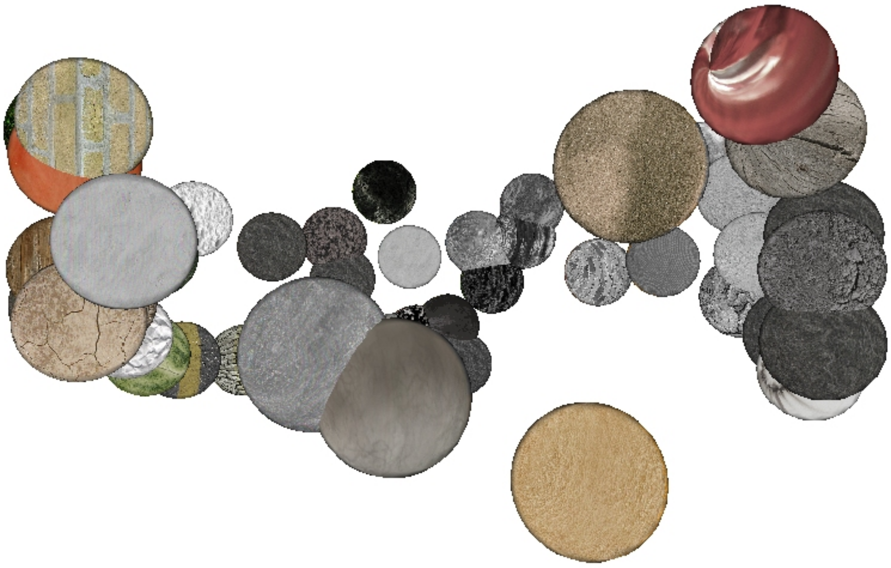

Figure 4.7: Example frame from Experiment 1, where stylised focus is on some other random sphere, using saturation variation (*Sat*).

on the target object.

Reaction times were also averaged for scenes when both the stylised focus was on the target object and when the focus was on some other random object. These times were averaged for each style to discover how they performed individually. A two factor ANOVA was performed where the conditions were *Focused/Not Focused* (2) and *Style* (8). It was seen that there was a main effect between focused and not focused (focus on a random sphere, not the target sphere) scenes for all of the styles ($F(1, 19) = 5.7467, p < 0.025$). Post-hoc analysis then showed that the adaptive abstraction styles used produced consistently faster times with focus on the target object than when the focus was on another object as can been seen from Figure 4.9. It can also be seen from this figure that some styles resulted in faster reaction times than others. For example the Kuwahara filter (*Kuwa*) produced significantly faster times ($2.18s$) than the quantization filter (*Quant*) ($2.72s$).

Experiment Two

Procedure It was seen from experiment one that the individual NPR styles used can effectively influence user recognition times. A second set of experiments was run that

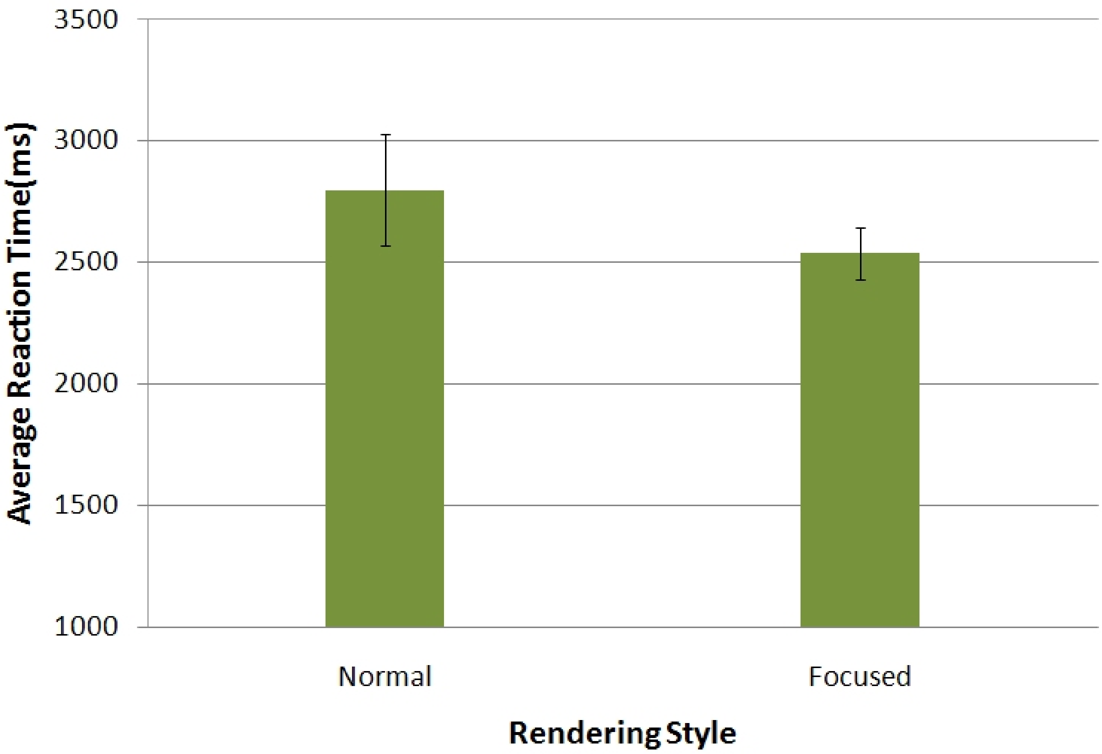

Figure 4.8: Average reaction times from Experiment 1 for all styles. Note error bars represent standard errors of the mean in all graphs.

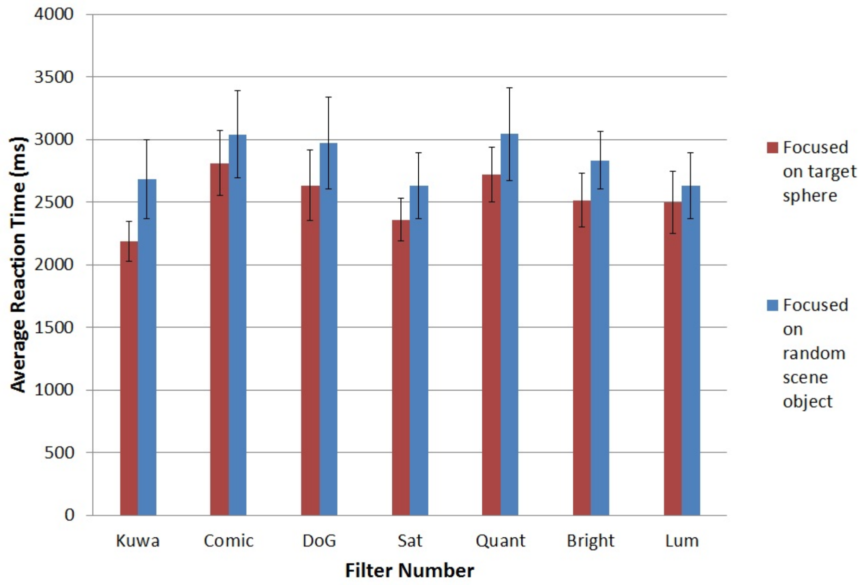

Figure 4.9: Average reaction times from Experiment 1 for each individual style. The graph shows that each style resulted in faster times when the focus was on the target object rather than on some other scene object. It can also be seen that some styles perform significantly better than others, for example the Kuwahara filter and saturation variation performed particularly well.

Combination style	*Filter description*
KSB	**K**uwahara filter, **S**aturation variation and **B**rightness variation
KDQL	**K**uwahara filter, **D**oG edges, colour **Q**uantization and **L**uminance variation
CSL	**C**omic-book filter, **S**aturation variation and **L**uminance variation
CDB	**C**omic-book filter, **D**oG edges and **B**rightness variation

Table 4.3: Description of the various combination styles used in Experiment two, within the simple context experiments.

looked at how combining styles can influence user times. This experiment was run with static scenes with spheres of varying size. Twenty participants (13 male, 7 female) who had not taken part in the first experiment volunteered to participate in the experiment. In this experiment users were shown a group of one hundred images each of which contained one hundred and fifty randomly generated spheres of varying size. The placement and size of each sphere was random and each sphere was given a random texture. Twenty of the images were again rendered using normal 3D local illumination, and the other eighty images were rendered using four different types of NPR (twenty images per style). It was ensured that there was no clipping or occlusion of the target object in each image. Like the first experiment, half of the stylised images were focused on the target object while the other half were focused on a random other object within the scene. Four combinations of varied styles were chosen for the experiment ensuring that each filter within a combination style provided a different type of abstraction. For example, two of the combination styles used luminance variation while the other two used brightness variation. This was also done for the painterly styles, the Kuwahara and comic-book filters. We chose such combinations to observe the comparative effect of each similar stylisation approach. The styles selected are listed in Table 4.3.

Results Participants' reaction times were averaged for each abstraction style and for the normally rendered images. A one-factor ANOVA was then performed on the results where the condition was *Normal/Focused/Not Focused* (3). It was found that there was a main effect in the reaction times for normal, focused and not focused images ($F(2, 38) = 78.503, p < 0.01$). Post-hoc analysis was then performed using Newman-Keuls and it was seen that when the target object was focused, average recognition speeds ($1.45s$) were faster than normally rendered images ($1.64s$), which were in turn much faster than

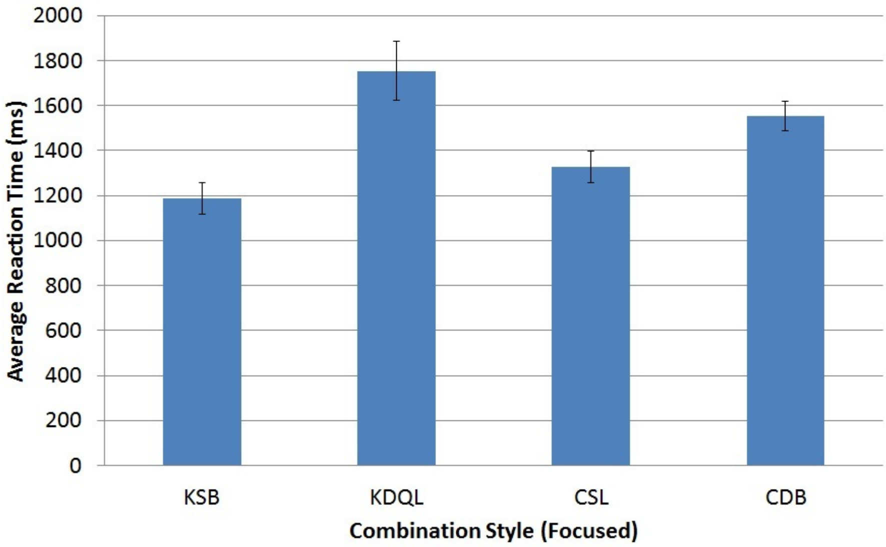

Figure 4.10: Average reaction times for the focused abstraction styles from Experiment two. The style containing the Kuwahara filter, Saturation variation and brightness variation (KSB) resulted in the fastest times.

the images in which the focus object was some random sphere and the target object was abstracted ($2.28s$) ($p < 0.02$ in all cases). These results confirm what was already discovered in experiment one.

It was also seen that some style combinations performed significantly better than others. Reaction times were averaged for when the target sphere was in stylised focus and a one-factor ANOVA was then performed on the results with the condition of *Style* (5). It was found there was again a main effect of abstraction style ($F(3, 57) = 13.677, p < 0.01$). Post-hoc analysis was performed using Newman-Keuls pairwise comparison of means. Combination Style *KSB* resulted in excellent average times ($1.19s$) while the times from Combination Style *KDQL* ($1.75s$) were considerably slower than the average times for the other styles as well as the times for the normally rendered images. Combination Style *CSL* resulted in good times ($1.33s$) while Combination Style *CDB* ($1.54s$) did not perform quite as well ($p < 0.04$ in all cases except Combination Style *KDQL* vs. Combination Style *CDB*). This difference in abstraction reaction times can be seen in Figure 4.10. Reaction times were generally faster than experiment one although this is due to the fact that this experiment was performed on static images and not moving scenes.

It was also seen from these experiments that the projected size of the target sphere in the scene was important. When times were averaged according to size a one-factor ANOVA was performed, with the condition of *size* (8) and it was seen that size was a significant

Combination style	*Filter description*
KSB	**K**uwahara filter, **S**aturation variation and **B**rightness variation
KSBQ	**K**uwahara filter, **S**aturation variation, **B**rightness variation and colour **Q**uantization
KSBD	**K**uwahara filter, **S**aturation variation, **B**rightness variation and **D**oG edges

Table 4.4: Description of the various combination styles used in Experiment three, within the simple context experiments.

factor ($F(1, 19) = 256.00, p < 0.01$). Screen distance between the centre of the focus object and screen centre was also seen to be a significant factor ($F(1, 19) = 237.20, p < 0.01$) as users tended to find the sphere faster if the sphere was centred. Despite these results, the distributions of sphere size and screen distance were evenly spread throughout each style, so they did not have an impact on the overall style results.

Experiment Three

Procedure It was seen from experiment two that some abstraction combinations worked significantly better than others. In an attempt to isolate the reasons for this, another timed search experiment was run. Twenty participants (11 male, 9 female) volunteered, each of which had not participated in previous experiments. The same format as experiment one was again used. This time there were eighty images, twenty of which were normally rendered. The remaining sixty images were divided into three sets of twenty images each and, as before, half were focused on the target object and half on a random object. The most successful style from the first experiment, Combination Style *KSB*, was again used. The other two combination styles in the experiment were generated by using Combination Style *KSB* again but adding one other type of abstraction. Colour quantization and DoG edges were added as they generated slow results when incorporated as part of other styles in the first experiment. This was done to allow direct comparison between styles and show what effect each style addition had on recognition speeds.

Results The reaction times of participants were again averaged for each style when the image was in focus and a one-factor ANOVA was performed on the results, with the condition of *Style* (4). It was shown there was a main effect of abstraction style ($F(2, 38) = 42.649, p < 0.01$). Post-hoc analysis showed that there was no significant difference between the average times between Combination Style *KSB* (1.30) and Com-

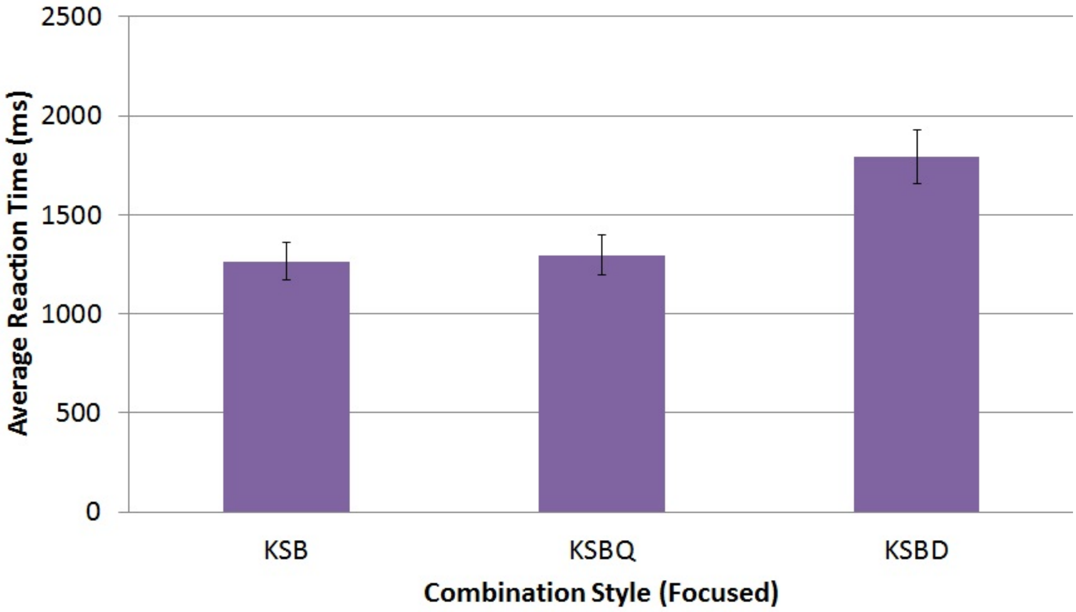

Figure 4.11: Average reaction times for the focused abstraction styles from Experiment three. The style containing edges (*KSBD*) results in significantly longer reaction times than the other two styles tested.

bination Style *KSBQ* (1.27s). However, as can be seen from Figure 4.11 there was a significant difference in average reaction times between the set of abstracted images containing edges, Combination Style KSBD (1.79s) and the other two sets ($p < 0.002$ in each case).

It was noted that there was no significant difference in reaction times for recurring styles in experiment two and three. This was despite having different participants from experiment one and a new set of images. This is true for Combination Style *KSB* when the target object was in focus (1.19s for experiment one and 1.27s for experiment two), Combination Style *KSB* when the target object was abstracted and focus was on some other object (2.01s for experiment one and 2.28s for experiment two) and normally rendered images (1.64s for experiment one and 1.53s for experiment two).

4.5.3 Eye-tracking Experiments

Experiment Four

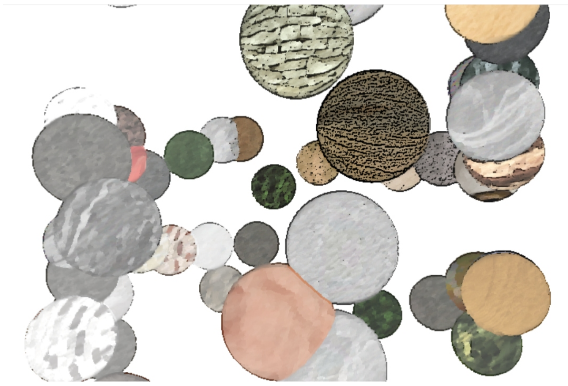

Figure 4.12: Example frame from an animation shown to participants in eye-tracking experiment (Style *KSBD*).

Procedure The next and subsequent experiments used an eye-tracker to investigate how implementing NPR within real-time scenes can affect a user's gaze behaviour. An SR-Research Eye-link II eye-tracker was used. In the first of these, ten participants (7 male, 3 female) participated and were all positioned 64cm away from an 18inch monitor using a chin rest. It was ensured the participants were focusing on the centre of the screen before each trial. Each trial contained fifty spheres rotating around the centre of the screen as in experiment one (see Figure 4.12). The participants were shown forty animations, each lasting eight seconds. In each animation the abstraction was focused on some sphere as the objects rotated. In half of the abstraction sets, the focus object was a brick textured sphere, while the other half were focused on a random object. Ten of the animations were rendered using normal 3D local illumination. The other thirty were divided into three sets of ten animations, each with a different abstracted rendering style. The combination styles were the same as in experiment two, i.e. Combination Style *KSB*, *KSBQ* and *KSBD*.

The users were asked to count the number of brick textured spheres in each scene and state the number after each trial. The task was given to ensure participants were searching the entire scene throughout the animation and their performance in the task was not actually used in the evaluation. There was between one and three brick textured spheres in each animation. The purpose of the experiment was to investigate whether the focus sphere and its surrounding area drew more attention than the rest of the scene during a search task, especially in the first few seconds when the user is first presented with the scene.

Results For the eye-tracking experiment the distance between the centre of abstraction and the user's gaze point was measured for each frame. A two-factor repeated measures

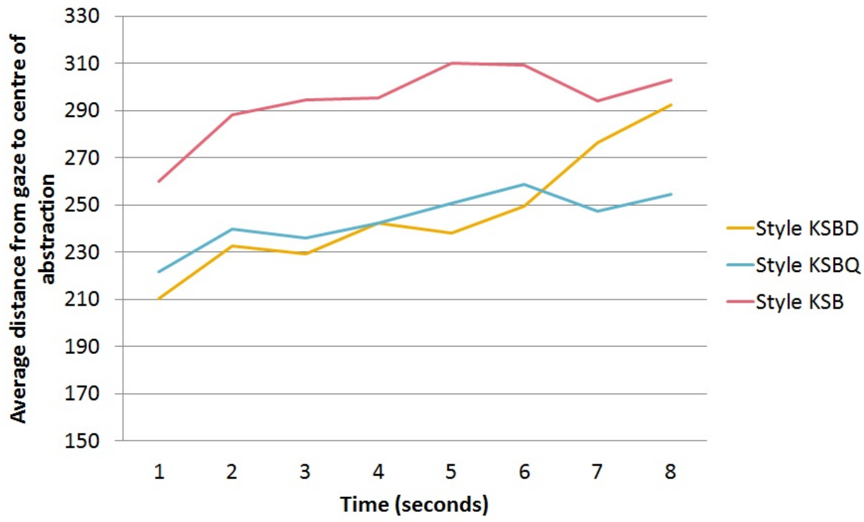

Figure 4.13: Distance from gaze position to abstraction centre through time for each abstraction style in Experiment four. As can be seen, the style containing edges (Style *KSBD*), results in closer fixations to target objects. It is also seen that for all styles tested, there are more fixations near target objects early in trials.

ANOVA was performed with conditions of *Style* (3) and *Time* (8), using the average distances per frame grouped into abstraction styles. The results showed that there was a main effect of abstraction style ($F(2,18) = 20.916, p < 0.01$), indicating that some styles drew gaze more effectively than others. There was also a main effect of time ($F(7,63) = 7.9485, p < 0.01$), where user gaze was drawn to the focus of the abstraction early on in each animation. Post-hoc analysis was again performed using a Newman-Keuls test for pairwise comparisons among means. This showed that there was a significant difference between Combination Style *KSBD*, containing edges, and the other two styles ($p < 0.002$), although the difference in Combination Style *KSB* and *KSBQ* was not significant. It was also proven that the tendency for the gaze to be drawn to the objects in focus was significant for the first second of the animations ($p < 0.02$).

Our results strongly indicate that the abstraction styles can effectively draw user gaze to specific objects in a simple scene. As can be seen from Figure 4.13, there is a trend for user gaze to be immediately drawn to the area surrounding the focus object, regardless of whether that object is the search target or not. Gaze then generally moves away from these focus areas as users search the rest of the scene. From this experiment we can see that Combination Style *KSBD*, which contains edge detection, performs significantly better in drawing user gaze than Combination Style *KSB*, which contains the same styles without

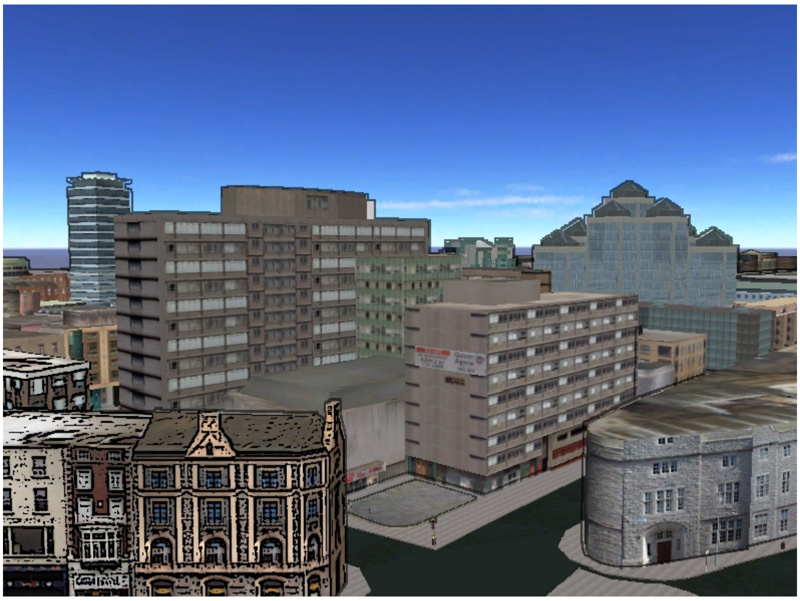

Figure 4.14: Example frame from a scene in Experiment five.

edge detection, and Combination Style *KSBQ*, which contains the same styles and colour quantization.

Experiment Five

Procedure The final experiment looked at how the NPR styles used performed within an real-time scene containing semantic information. While the results of the previous experiments, with simplistic scenes, were useful it was interesting and worthwhile to see if those results are consistent when using the same styles within a more detailed and meaningful scene. The eye-tracking equipment described in the last experiment was again used. For this experiment the scene chosen was an interactive urban simulation [HO03]. Ten new participants (5 male, 5 female) were shown thirty-two scenes, each six seconds long and randomly ordered. An example frame from a scene can be seen in Figure 4.14.

Each scene was rendered in real-time during the experiment and showed one of four walkthroughs of the urban environment. Each walkthrough was shown eight times, once rendered normally and also with each of the individual NPR styles used in experiment one. In each abstracted scene the focus was on one of four buildings in each walkthrough. The buildings chosen were of varying prominence within the scene. The users were given no task but asked to investigate each scene normally. Eye tracking fixations were then extracted and it was determined which object was being inspected at each fixation. In

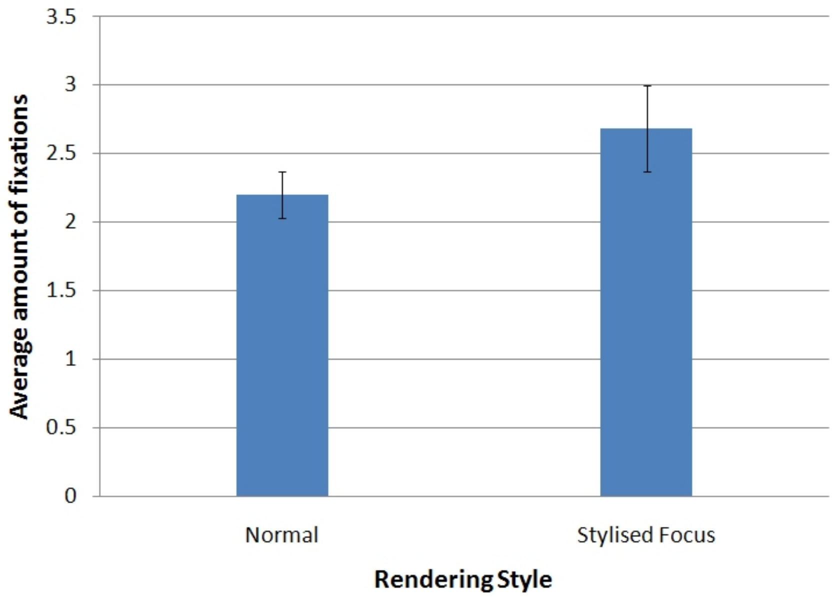

Figure 4.15: Average fixation amounts for Experiment five. When using adaptive abstraction with stylised focus on the target objects, there are significantly more fixation hits on the target object.

this experiment individual styles (see Table 4.2) were used rather than the combination styles used in previous experiments as we wanted to isolate the effects of each style on gaze behaviour. As no significant differences in results were observed in the previous experiments, when using the similar filters of luminance and brightness variation (see Figure 4.9), we decided at this point to test only brightness variation for future studies. The filter was chosen as there was a slight improvement in times when using it in past experiments, albeit not significant.

Results In the last eye-tracking experiment, the user was given a search task, which meant that the first few seconds of gaze behaviour were relevant. In this experiment, the user was simply asked to inspect each scene, so overall gaze behaviour was more interesting. The total amount of fixations on each object was averaged for all NPR styles and for the normal renderings. A one-factor ANOVA was then performed, with the condition of *Normal/Focused/Not Focused*, and it was found that was a main effect between the normal and focused renderings ($F(1,9) = 12.081, p < 0.01$). As can be seen from Figure 4.15 the user spent a significant longer time looking at the target object if the building was in stylised focus. The average times for each individual style were then averaged and after a one-factor ANOVA was performed on the results with the condition of *Style* (6), it was

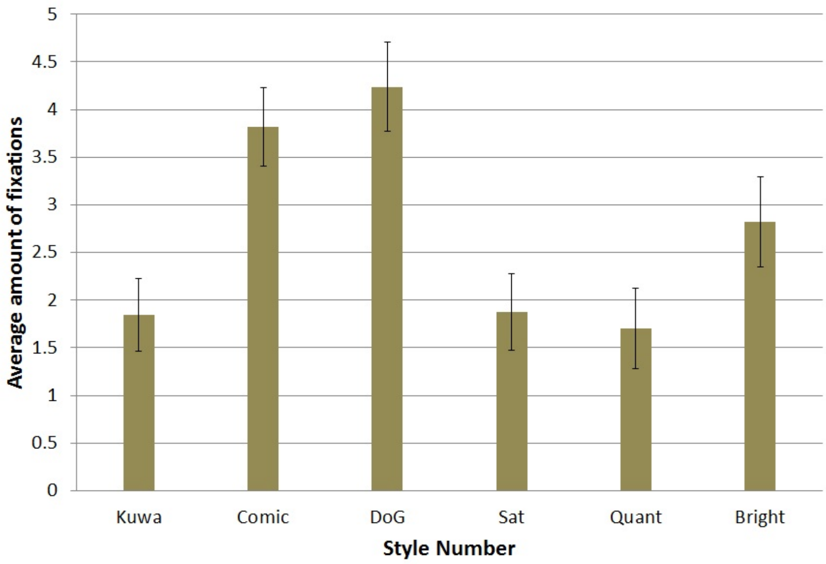

Figure 4.16: Average fixation amounts for each individual style in Experiment five. Difference of Gaussian edge detection and the comic-book filter result in significantly more fixations on target objects.

seen that there was a main effect of rendering style ($F(6, 54) = 11.564, p < 0.01$). Post-hoc analysis was then performed using a Newman-Keuls test for pairwise comparisons among means and it was seen that some styles performed significantly better than others (as can be seen from Figure 4.16). In particular the comic-book filter and difference of Gaussian edges were significantly better at drawing user attention to target objects than the rest of the filters used ($p < 0.02$ for each case).

4.5.4 Discussion

With this set of experiments we have shown that using image-space real-time NPR techniques can successfully influence user recognition times and gaze behaviour in real-time scenes. We have also seen that the effectiveness of different NPR techniques can vary largely depending on the style and also the task. For example the addition of edges was very successful at increasing object saliency during animations but performed badly in the timed search task in static frames. While edges do seem to make objects more salient within scenes a number of participants noted that recognition of objects containing edges took slightly longer than when it was rendered normally. Similarly the Kuwahara filter was very successful at improving reaction times in all search tasks whereas it performed significantly worse in drawing user gaze in both the eye-tracking experiments. Some styles, such as colour quantization, performed poorly in comparison to other styles in all tasks. The results from this set of experiments shows conclusively that: (a) stylised abstraction can improve visual search times of users, (b) stylisation can be used to effectively draw user gaze to intended objects in a scene and (c) certain stylisations consistently perform better than others.

4.6 Matching Pairs Experiment

After investigating how styles, both individually and combined, affect user behaviour in basic scenes, we present a perceptual experiment that builds on the results found in the previous section. In a visual search task designed to test people's perception of abstracted scenes, different individual styles are again used to investigate how reaction times can be affected. While we have already investigated how different styles perform against one another, here we try to answer important other questions such as how scene type affects the success of each style and how the parameters from each style can be altered to maximise its success.

The work we have presented in this chapter adheres to the traditional NPR goal of creating aesthetically pleasing results while focusing on the creation of styles that optimally affect user performance. In this section we show how NPR techniques compare against non-stylised renderings, compare the effectiveness of different styles, determine how varying each style can affect performance and investigate how these styles perform with objects of varying complexity.

There has been some evaluation of NPR systems for affecting user attention earlier in the chapter as well as past research [SD04] [CDF+06]. Few systems, however, have dealt with the design choices involved, such as why particular styles were used above others and why certain parameter values were chosen in each style. There is also a lack of research into how scene type and complexity can affect the performance of NPR styles. These issues must be addressed to best affect the visual communication of an interactive scene.

The perceptual study in this section investigates how NPR styles can influence user task performance by varying non-photorealistic stylisation levels within an interactive scene. In particular, we examine the factors which determine the success of such an approach including style type, stylisation level and scene type. In these studies we investigated how adding and altering a variety of NPR styles can influence users' ability to pick out a single matching pair from a number of objects. Three levels of object complexity were tested with the styles to determine how scene type affected results. We use basic shapes with simplistic colours, a set of fruit models with semi-complex textures and varying shape and

Figure 4.17: Examples of the styles used in the experiment.(a) No abstraction (b) Edge detection (c) Shading stylisation (d) Texture stylisation (e) Saturation variation.

finally a character set. The character set is used to see how the styles perform with objects of similar shape and large semantic connotations. While characters are a special case, it has been shown in past work that the addition of abstraction can help user recognition of faces [WOG06].

4.6.1 Rendering Techniques

In this perceptual study, four varied styles were tested which were chosen to represent four major types of abstraction. Each of the three types of abstraction used in the previous set of experiments was again employed, which are edge stylisation, texture abstraction and colour variation. In addition to these three types of stylisation, shading stylisation was also implemented. The shading stylisation was not tested in the previous set of experiments, which focused on image-space approaches. However, shading stylisation was included in this experiment due to the results described in Section 4.4, where cel-shading, combined with other styles, showed some promise at influencing user times in a simplistic task. The four styles described are split into two categories in how they are used to alter scene saliency; Edge and shading stylisation increase the saliency of the target object, leaving the background objects rendered normally, while texture abstraction and colour variation decreases the saliency of background objects through abstraction leaving the target object in full detail.

One type of stylisation was chosen to represent each class of abstraction from the available styles. In previous experiments we have tested a number of similar styles but as many of the styles outperformed others in these studies, we tested only the most successful styles from previous sections, which we felt were representative of each abstraction class. *Saturation variation* was chosen ahead of other colour variation techniques such as brightness and luminance due to the good comparitive results from the experiments in Section 4.4. The *Kuwahara filter* [KHEK76] was used to represent texture abstraction methods, above the comic-book filter, after producing fast times in earlier search task experiments (Section 4.4.2).

Sobel edge detection was implemented as it is a simplistic filter that produces reliable and unstylised bold edges. While the earlier experiments used Difference of Gaussian edges, we felt displaying basic edges without major stylisation would result in generalisable results. Finally, an object-space algorithm, *cel-shading*, was implemented as it was hypothesised that quantizing the shading values within each object could alter object saliency. This style was chosen over other shading styles, such as pencil-shading, due to its performance in an earlier experiment (Section 4.2). The cel-shading was combined with any existing textures on the objects to achieve a shading style that does not lose any existing object information, as described in Section 3.2.3.

Each style was tested using 8 different levels of abstraction to determine how the degree of stylisation affected reaction times. In the case of the edge and shading stylisation the background characters were always rendered with basic minimum abstraction while the level of abstraction of the target pair was increased systematically. In the case of shading stylisation this resulted in background objects rendered with 10 shades while the number of shades of the target object was decreased from 9 to 2 shades. The edge detection style rendered background objects with a small number of thin lines i.e. setting the threshold for displaying edges on the Sobel filter to 0.95 on a scale from 0 to 1. This meant only a small number of detected edges were displayed. The threshold on the focus characters was then decreased from 0.9 to 0.2 on the target objects to increase the thickness and frequency of internal edges and silhouettes. This resulted in investigating two-level adaptive abstraction, which we have yet to test after using multi-level adaptive abstraction for the previous experiments.

In contrast to edge and shading stylisation, the colour variation and texture abstraction styles rendered the target characters with minimum abstraction while the abstraction of the background characters was increased. For saturation variation the target object was rendered with full saturation and the background objects were rendered with reduced saturation from 90 percent down to 20 percent. The painterly effect altered the size of the filter to increase the abstraction of the background objects while leaving the target objects in a clearer style (see Figure 4.18(a)). Examples of each NPR style implemented can be seen in Figure 4.17.

Figure 4.18: Example trials from the experiment (a) Texture Variation on the character set (b) Saturation Variation on the fruit set (c) Shading stylisation on the simple shape set.

Three types of object were used with varying complexity. Simple shapes such as cubes and cones, each containing one block colour, comprised the first and most basic level. The next level of object complexity contained fruit models with relatively basic shapes and some simplistic textures. The most complex model tested was a set of character meshes, each with an identical stance and highly detailed textures. The three model types were chosen due to the varied characteristics between them.

4.6.2 Experiment Framework

The experiment task was to pick out a matching pair from a grid of 15 objects set in three rows of five (see Figure 4.18). This simple search task was chosen as it would measure user's basic reaction times within an interactive scene while keeping the semantic information to a minimum. In this way it would be easy to measure what effect each of the non-photorealistic styles had on the user's task. A similar experiment framework was used by McDonnell et al. [MLD+08] to identify how to most effectively hide clones of characters. In that experiment the user was instructed to find appearance clones in a grid of characters in a short as time as possible. While adding background to the scene could make a difference in reaction times, a number of factors were already being tested (model type, style, abstraction level) and we believed that the addition of another would make the results over-complicated and hard to interpret.

In [MLD+08] it was found that if the clones were not directly adjacent in the grid then their position had no effect. It was also found that there was no significant difference in reaction times regardless of which target model was chosen. Therefore, in this study, the target object for each trial was selected randomly and the position of the matching pair was random provided that they were not adjacent. We separated the experiment into three blocks, each of which tested a different complexity of model. The blocks were randomly ordered for each participant.

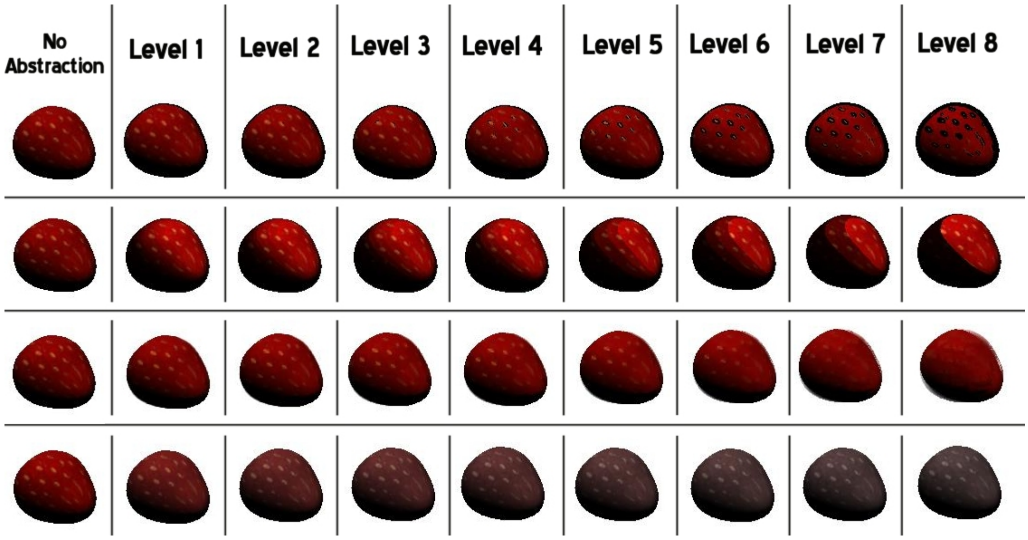

Figure 4.19: Examples of each level of abstraction for each style. Row A: Edge detection, row B: Shading stylisation, row C: Texture stylisation, row D: Saturation variation.

There were 40 trials in each block, each containing 8 trials for 5 different types of rendering. 4 of these rendering types were the NPR styles described, while one was basic rendering with no stylisation, included for comparison with each style. In each trial the target objects were in stylised focus using the various styles apart from trials in which there was no stylisation. Thirty naive participants from different educational backgrounds volunteered for the experiment (16M-14F).

Participants sat at a distance of 60cm from the display throughout the experiment. A mouse was used to control an onscreen cursor and participants clicked to select objects, whereupon a bounding box appeared surrounding the chosen object. After a pair was chosen the bounding boxes would disappear if an incorrect pair was chosen and the trial would end if a correct pair were chosen. The reaction times were recorded for each trial and if the correct pair was not found then the maximum time of 30 seconds was recorded as the reaction time. Between trials a fixation cross appeared in the centre of the screen for the participants to look at between trials ensuring that they were always looking at the same screen position before each trial.

4.6.3 Results

We first analysed the data using the reaction times from the four abstracted styles, leaving out the times from the normally rendered set. A three factor ANOVA was performed where the conditions were *Model type* (3), *Style* (4) and *Level of abstraction* (8). It was observed that there was a main effect of model type where the complexity of the model affected user reaction times ($F(2, 58) = 119.33, p < 0.00001$). Post-hoc analysis was then performed using a standard Newman-Keuls test for pairwise comparison among means. As can be seen from Figure 4.20 it took significantly longer to identify matching pairs within scenes of characters than it did within the two simpler scenes ($p < 0.0002$ in both cases). There was no main effect of style indicating that none of the styles aided or hindered reaction times across all model types.

It was also found that there was an interaction between Model type and Level of abstraction ($F(6, 174) = 3.3025, p < 0.0045$), where the success of the different levels of abstraction varied significantly depending on which model type was shown. This implies that no optimal parameter values exist that are valid across all scene types and that effective stylisation parameters are context-dependant. To further investigate this result, the average times for each level of style were analysed for each model type. The times from the most effective levels of style were then extracted for each model type and compared with the reaction times from the unstylised trials for that model type.

An ANOVA was then conducted on the average reaction times for each model set with the optimised style levels and the standard renderings. It was found that there was a main effect of style for both the character set ($F(4, 116) = 3.2878, p < 0.015$) and the fruit set ($F(4, 116) = 3.6641, p < 0.008$). Post-hoc analysis was then performed using a standard Newman-Keuls test and it was found that all styles produced significantly faster times than standard rendering in the fruit set when the detected optimal parameters were used ($p < 0.043$ in all cases). The success of all optimal styles on this set is illustrated in Figure 4.21. The optimal levels of abstraction are also shown in this figure, with level 1 being the lowest level of abstraction and 8 being the highest.

It was also seen that, in the character set, altering the saturation of the background characters using the optimal parameter level (level 8, highest level of abstraction) produced significantly faster results than standard rendering ($p < 0.008$). While the other styles produced lower average times, as can be seen in Figure 4.22, only saturation variation was considered statistically significant. We had hypothesised other styles such as texture variation and edge stylisation would make a significant difference in this set, although the outcome may be explained by the fact that faces are perceived very differently to static objects. Past research has discovered that the upper torso and head are attended to most in search tasks that involve characters [MLH$^+$09]. This indicates that scenes involving characters may have more semantic connotations and therefore are less susceptible to preattentive processing cues than either of the simpler object scenes. Therefore, by abstracting the upper bodies and faces, especially when the character is at a low scale, an

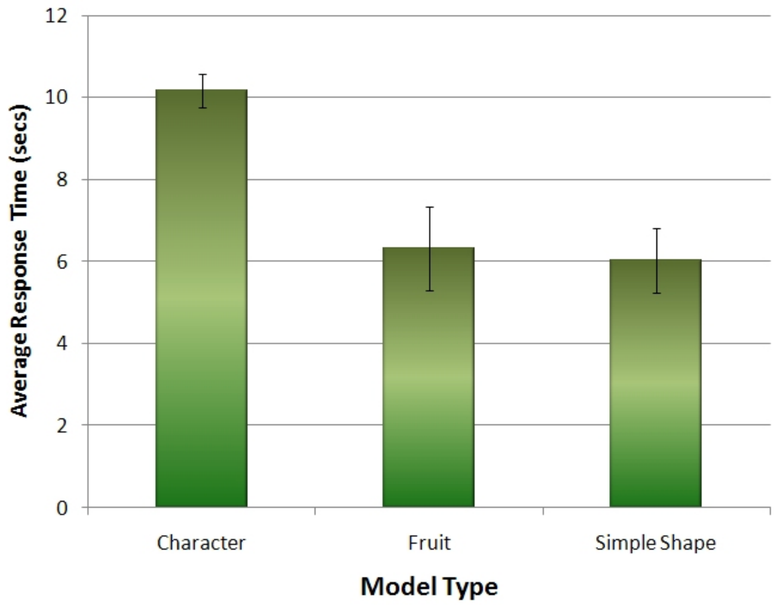

Figure 4.20: Average reaction times for each model type for all abstraction types.

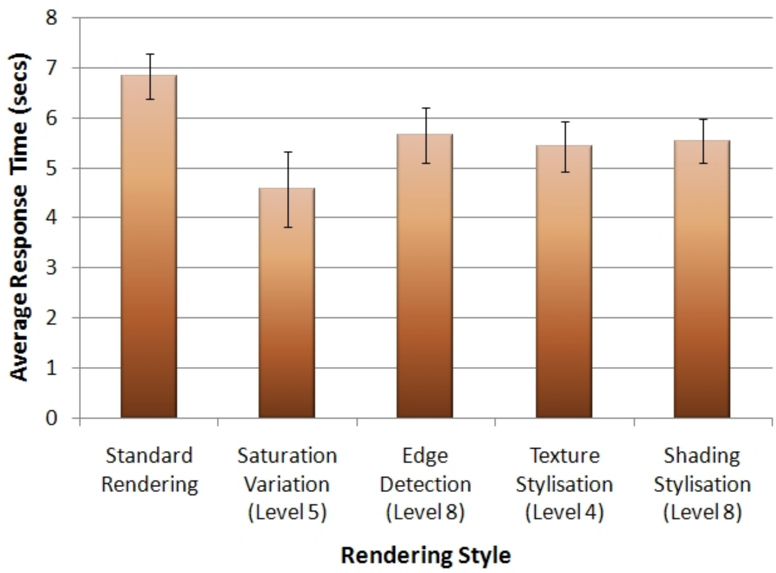

Figure 4.21: Average reaction times for optimal stylisation levels on the fruit set of models.

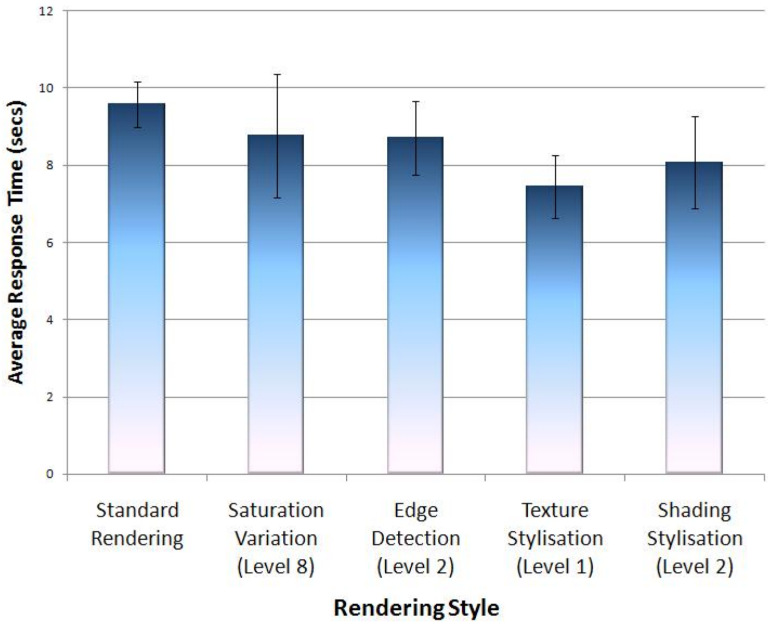

Figure 4.22: Average reaction times for optimal stylisation levels on the character set of models.

important visual cue for virtual characters may be taken away. This hypothesis is backed up by the results found in Section 4.4, where the style containing edges did not result in any significant improvement in task performance on a crowd scene.

The average times for all optimal styles did produce faster average times than those for the standard renderings within the simple shape set, however there was no main effect of style. This can be seen in Figure 4.23. This indicates that none of the styles applied to the simple shapes had a significant effect on reaction times in comparison to each other or the standard renderings. This was somewhat expected in that the simplicity of the shapes in their colour and shape meant that the impact of the NPR styles was lessened, although it was surprising that the saturation variation did not have an effect. This may be because, as several participants noted after the experiment, shape was used as the main visual cue in this set.

As can be seen from the results, there was much variation in which levels of stylisation were considered optimal for each set. For example the optimal stylisation levels for texture stylisation varied from level 1 (lowest level) for the character set to level 8 (highest level) for the fruit set. Similarly, the optimal level for edge detection varied from level 1 in the basic set of models to level 8 for the fruit set. These results strongly indicate that scene context is the most important factor when choosing abstraction types and stylisation

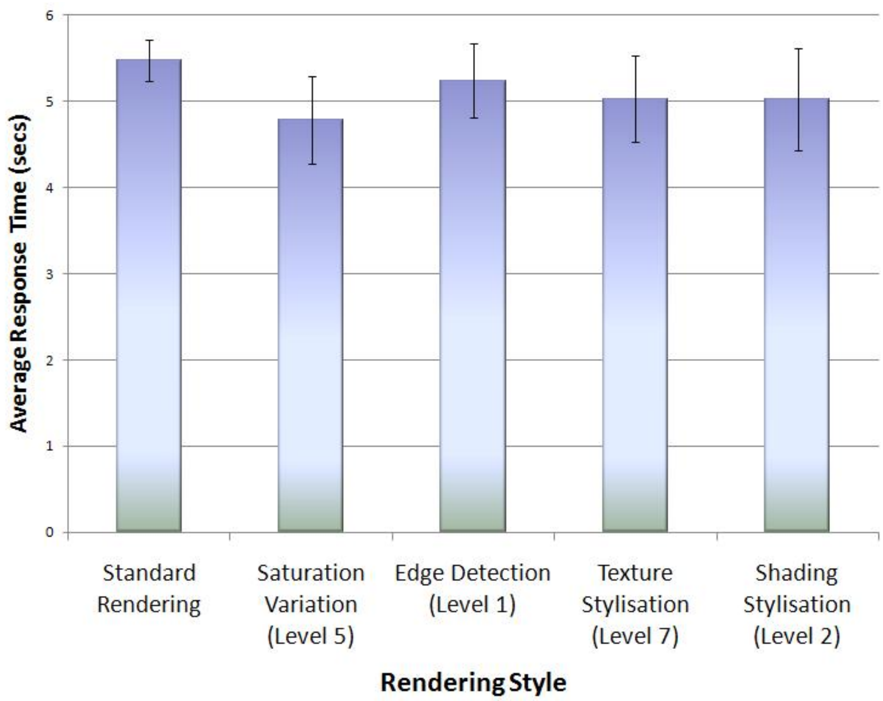

Figure 4.23: Average reaction times for optimal stylisation levels on the basic set of models.

levels.

The results from the experiment show that NPR can be a useful tool for increasing the saliency of target objects while reducing the visual impact of the rest of the scene. However, it is also shown that the success of each style depends largely on the scene context and also on the level of stylisation used. We believe the results from this study can help in the creation of effective NPR styles in the future.

4.6.4 Discussion

In this experiment, we have confirmed the results from earlier experiments in showing how non-photorealistic rendering techniques can be used to successfully influence task performance in visual search tasks. However, it has also been shown that scene type has a large effect on the success of such styles. The results from the simple object set show that styles, such as the ones implemented, are not effective at drawing user attention in very simplistic scenes that contain only basic colours and shapes. The results from the character set demonstrate that semantic information within a scene must be taken into account when choosing abstraction styles as only one style, saturation variation, resulted in significantly faster times with human models. The success of all styles within the set of fruit models shows the promise of using such stylisation in scenes containing semi-complex shapes and textures.

We have also determined that choosing both the right style and the correct levels of abstraction are vital in producing the best visual search performance within a scene. Some style types were successful in producing significantly faster reaction times in some scenes while proving ineffectual with other model types. The most effective stylisation levels also vary largely dependent on the model type, for example, texture abstraction performed most effectively in the most extreme level for the fruit set while showing the best results at the minimum level of abstraction for the character set.

While we see that the times found vary largely between style and object type, there are still conclusions to be drawn from the experiments. We can see that the use of bold lines and edges, while drawing user attention to objects as found in Section 4.5, can have an adverse effect in complex models. This may be because detail within textures in such models can be used to identify that object and the addition of edges can slow down the recognition process rather than aid it. Results for very simplistic objects also produce no significant increase in reaction speed when edges are added due to the lack of any detail within the objects, and therefore lack of any internal edges. Our results indicate that using edge detection to enhance the object saliency may be better suited to objects with simplistic shapes and textures where edges can enhance object saliency without inadvertently disguising the object.

We also see from the results that scene abstraction techniques, which preserve the target object, while stylising background objects, are better suited for smaller more complex models such as characters. This is due to the success of saturation variation and texture abstraction on these models. This explains results found in earlier sections where complex textures were used in similar perceptual experiments. We believe that the results presented here show that while NPR can be very successful in this area, there are many design choices that must be carefully selected when creating such styles. We also believe the results make significant progress in understanding what factors determine the success of NPR as an effective tool in influencing user perception of a scene.

4.7 Interactive Search Experiment

The results from the previous experiment showed how style choice, scene type and the level of abstraction can affect the success of the adaptive abstraction approach. From this study we were able to determine how the parameters from each style can be altered to maximise success with particular object types. In this section, we describe a further experiment which was run to determine whether the optimal levels of stylisation found in the previous experiment could be successful in a more complex, interactive environment.

A visual search task is again used in this experiment, with three individual adaptive abstraction styles, using the optimal levels of stylisation found in the previous experiment. In the majority of our previous experiments we have used basic scenes without complex backgrounds to isolate contributing factors. However, in this experiment we add a complex background, the Metropolis system, and allow the user interactive control, to verify that past results will hold true in complex, interactive applications.

To build upon the results from the previous section, the character set was again used for this experiment, which was one of the three object types tested in the previous experiment. The results from that experiment showed that characters must be treated differently to simpler objects as there are many semantic connotations associated with characters. In particular, the results showed that by abstracting the characters, especially the target characters, important visual cues were taken away. We were interested to see whether these results could be used to improve the performance of each style in this experiment.

4.7.1 Rendering Techniques

We used three styles in this perceptual study, which were tested in the previous experiment. Saturation variation and texture abstraction (the Kuwahara filter), which abstract background characters, were again used as they gave good results in the previous experiment with the character set, with saturation variation performing significantly better than standard rendering. Edge detection and shading stylisation, which stylise target objects to make them more salient, did not perform well in the previous experiment, as it was found that they can disguise faces and torsos, making character recognition more difficult. However, we decided to again test edge detection (Sobel filter), but alter the filter to try and improve results. The edge detector was adapted to display only the external silhouettes, leaving the character with no edge detection on the face and internal textures. We hypothesise that this silhouette edge will draw eye-gaze, but the lack of edge detection on the character itself will ensure recognition speeds are not slowed down. We decided not to test shading stylisation in this experiment, as it did not perform well in the previous experiment with the character set and it could not be altered to leave the target character in full detail.

Each of the styles described was implemented using the optimal styles found in the previous experiment. This meant the Kuwahara filter was implemented using level one of stylisation. This low level of stylisation abstracts the background characters and scene somewhat, but ensures that texture detail remains in all objects. The edge detection

Figure 4.24: Examples of the environment and each style used in the experiment. The target character can be seen in each image and is in stylised focus in each abstracted image. (Top-left) No abstraction (Top-right) Saturation variation (Bottom-left) Texture abstraction (Bottom-right) Edge detection.

filter was also used with a low level of stylisation (level 2), which displays each object with a thin silhouette, leaving the target object rendered normally otherwise. Saturation variation was implemented using level 8, the highest level of abstraction, which resulted in rendering the background characters and scene with very low saturation, leaving the target object in full saturation. Examples of each of these styles, as well as a standard rendering, can be seen in Figure 4.24.

4.7.2 Experiment Framework

The experiment task was similar to the previous experiment in that each participant was asked to pick out two matching characters from a set of 15. However, in this task the character to be found was set beforehand and each of the characters were spread around an urban simulation. The participant was given interactive control within the scene and could use the keyboard to move (arrow keys) and the mouse to look around. Though the participant could use the keyboard to move around, all characters were viewable without movement, although changing the direction of the camera with the mouse was necessary to complete each trial. Before the experiment began, the participants were shown four

Figure 4.25: Image of the target character for the experiment, shown to particpants before the experiment began.

initial trials without stylisation to ensure that they were comfortable interacting with the scene.

The target character was constant for all trials and was shown to each participant before the experiment started, with each of the styles and standard rendering to make the participant familiar with the character. The particular character was chosen as he was visually distinct from the rest of the characters yet did not contain any bright colours that might be a significant visual aid in a search task (See Figure 4.25).

Each trial began with the user looking at a section of the scene which did not contain any characters. The user was made look for two instances of each character, instead of one, to ensure a thorough search of the scene took place. It was ensured that the target characters were never within close range of each other. The position of each of the target characters was balanced for each style to ensure that position was not a factor. When the participant clicked on one of the target characters, a bounding box appeared around them. Once the participant clicked on the second of the target characters, the trial ended. If the participant did not find each of the target characters, the maximum time of 30 seconds was taken as the reaction time.

Three styles which were texture abstraction, edge detection and saturation variation were tested, along with standard rendering. Each of these rendering styles was using 10 different target character positions, leading to 40 trials in total. The trials were randomly ordered and a fixation cross was displayed for 2 seconds in between trials to ensure the user was looking at the same spot before each trial. Ten naive participants (7 Male, 3 Female) from different educational backgrounds took part in the experiment.

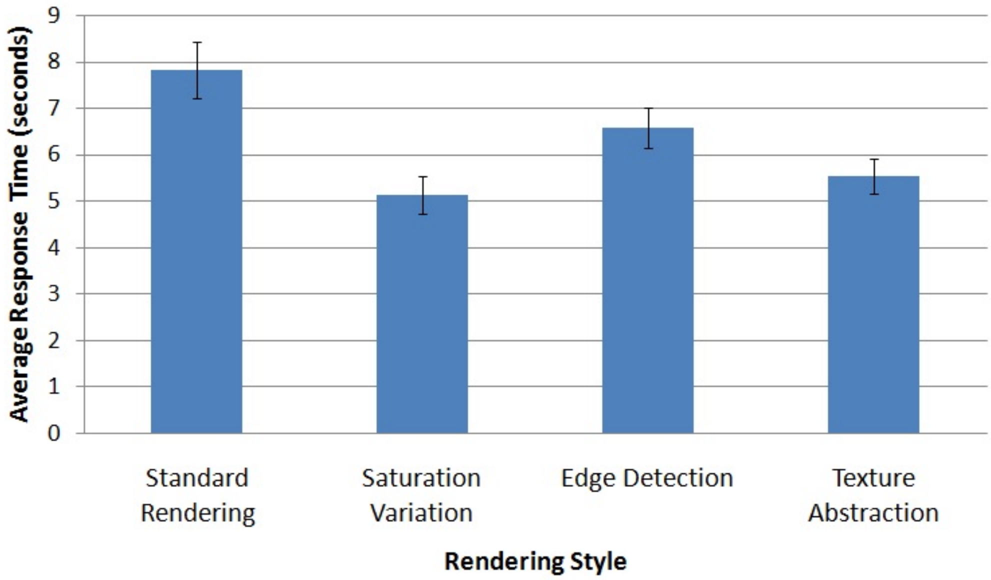

Figure 4.26: Average response times for interactive search experiment. Each of the adaptive abstraction styles returned significantly faster times than standard rendering.

4.7.3 Results

The reaction times of the participants were averaged for each of the four styles, and a one way ANOVA was then performed on the results with a condition of *Style* (4). Results showed (see Figure 4.26), that there was a main effect of style ($F(3, 21) = 22.927, p < 0.01$). Post-hoc analysis showed that all adaptive abstraction styles performed significantly better than standard rendering ($p < 0.002$ for all cases). This is an improvement on the results from the previous experiment, where texture abstraction and edge detection did not perform significantly better than standard rendering for the character set. Saturation variation again produced the fastest reaction times ($time = 5.12s$), as it did in the previous experiment.

The improvement in performance from the edge detection style ($time = 6.58s$) may be explained by the fact that the filter was adapted for this experiment. The filter was altered to display only external silhouettes, in an attempt to draw user focus to a character without occluding any perceptually important features. This style was significantly faster than standard rendering ($time = 7.81s$), although times were slower than the other two adaptive abstraction styles. The improvement in times from the texture abstraction approach ($time = 5.53s$) may be due, in part, to the fact that in this experiment, the participants were shown a pre-defined character to search for in the scene. This may have meant that they did not need to attend to background characters as much as in the previous experiment, where they were not aware which character was the target character. This may mean that abstraction of background characters has less of a negative effect on search times when the user is aware of the target character.

The results from this study show how the insights gained from previous experiments can be used to create effective adaptive abstraction styles in complex environments. It is interesting to note that the addition of a complex environment did not lessen the effects of adaptive abstraction. This experiment also shows how adaptive abstraction can be successful within interactive environments, where the user has full control over position and camera orientation. It was noted during the experiment that while only rotation of the camera was necessary to complete each trial, 3 of the participants also moved the camera position when searching the scene. The performance of each of the styles in this experiment shows how adaptive abstraction can be tailored for specific scenes and object types to successfully influence user behaviour in interactive scenes.

4.8 Summary and Conclusion

In this chapter we have presented a series of perceptual studies that increase the understanding of how non-photorealistic styles can influence user behaviour in real-time scenes. Using the adaptive abstraction approach, we have compared a variety of techniques to find which types of stylisation are most suitable for particular scenes and how to alter each style to get the most effective results. We have shown how different aspects of user perception can be influenced and we have evaluated each style in a number of ways. This enables us to propose a number of **guidelines** for choosing effective styles and enhancing their success using adaptive abstraction:

- The most successful styles for increasing object saliency and affecting search task performance were edge detection, painterly abstraction (through the Kuwahara filter) and saturation variation.
- Certain styles are not recommended for increasing object saliency within adaptive abstraction, such as stylistic shading, luminance variation and colour quantization.
- Edge detection can be an extremely effective method for influencing user gaze in any scene, although the drawback of this approach is that within complex objects, edges can obscure vital detail which can make object clarity and recognition difficult. This problem can be overcome by displaying only external silhouettes for texturally

complex objects, which can draw gaze to objects without obscuring important detail. Difference of Gaussian and Sobel edges produced similar results to each other.

- Painterly stylisation can be used to effectively abstract background data to make target objects clearer. It performed well over all experiments but we found that it was largely ineffective in the matching pairs task for character sets (Section 4.5), although it performed significantly better when the user was aware of the target object they were searching for (Section 4.6). This indicates that painterly stylisation is particularly effective in helping users search for specific known objects.

- Saturation variation performed consistently well in all tasks. The approach outperformed similar colour modulation techniques such as colour quantization or luminance variation. The approach was found to be a fast and effective method of influencing user behaviour and perception, although it may not produce as aesthetically pleasing results as other more complex styles.

- Two-level adaptive abstraction was seen to be a more effective approach than multi-level adaptive abstraction for changing scene saliency and improving times of user search tasks. Despite this, we believe that multi-level abstraction has a more natural appearance due to the gradual change in stylisation. This means that there may be a trade-off between effective, high-contrast two-level renderings and less effective, more natural multi-level abstraction.

- When using multi-level adaptive abstraction, we found that five levels of abstraction were enough to highlight target objects while ensuring scenes were rendered smoothly to avoid salient contrasts between abstraction levels.

- Determining the correct parameter values for each style depends largely on the scene context. This is especially true for the best edge thresholds, which vary largely depending on the scene. It was found that displaying all edges on a target object can draw user gaze effectively, but to stop the by-product of slowing down object recognition, displaying fewer edges for texturally complex objects is necessary. This is shown in the experiment results from Section 4.5, where the highest degree of edge detection was the most effective for the fruit set yet the second lowest degree of edge detection was found to be the most effective for the characters set.

- The best painterly parameter values for background objects were found when background data was abstracted to a degree that eliminates fine detail but preserves the basic context of the background objects. These values differed depending on object type as more complex objects require a lesser degree of abstraction than simpler objects to ensure important detail is preserved. This is shown in Section 4.5, where the character set was most successful using the lowest level of painterly abstraction, the basic set of models benefited from one of the highest levels of abstraction and the fruit set was most successful using a medium level of abstraction.

- Optimal thresholds for saturation variation were found to be consistently high for all scenes, which shows that a large colour contrast between target object and the rest of the scene can increase user task performance and object saliency.

While we have so far tested only Difference of Gaussian and Sobel edge detection, we hypothesise that the Canny edge detector will produce better results in object recognition tasks as the non-maximal suppression step ensures thin edges that would not obscure the object. This may also come at the cost of being less effective in drawing user gaze but it is an area we will research further in the next chapter.

This information will be useful in creating and developing NPR techniques which can effectively influence user reaction times and behaviour in interactive scenes. These perceptual studies also demonstrate how to obtain quantitative results on how a particular style performs in perceptual tasks, should future styles need evaluation. Real-time NPR techniques that can affect user behaviour would be useful within any interactive application where it is necessary to emphasise a certain point or object to the user while eliminating extraneous detail.

Chapter 5

Case Study: Volume Visualisation System

5.1 Introduction

The previous chapters have shown that real-time adaptive abstraction is possible and that it can have a significant effect on user perception in real-time scenes. As a case-study, to demonstrate how such an approach can be of use in a real-world application, we apply the adaptive abstraction approach to volume rendering. 3D volume visualisation is an important technique with widespread use in a number of fields. Example application areas that use volume data include medical imaging and scientific visualisation.

In recent years, advancing graphics hardware and new algorithms have enabled fast renderings of complex volume data. However, with this ability to interactively view volume data, comes a need to be able to render such data in a clear manner. Volume visualisation is an example application where there is a large amount of information to be displayed to the user. Such renderings are often affected by problems with visually complex images as volume datasets are often multi-layered and overlapping. This excessive detail can cause significant demands on the human vision system to process what can be perceptually demanding visual input.

By integrating volume rendering systems with the adaptive abstraction approach we show how adaptive abstraction can be used in existing applications. The adaptive abstraction approach is well suited to such complex visualisations as it can successfully highlight certain data while de-emphasising less important features. This case-study proves how the adaptive styles created in Chapter 3 and the perceptual knowledge learned from Chapter 4 can be of use in existing applications to influence user perception and render complex scenes clearly. We address the following questions in this chapter:

- Can the adaptive abstraction approach be integrated with an example application area, such as volumetric visualisation? We show that adaptive abstraction can easily be integrated with volumetric data, with some minor alterations to the adaptive abstraction system (Section 5.3).
- Can this case study show the advantages of the approach in a real-world application? This chapter shows how the adaptive abstraction approach can be used in an example application where a user's understanding of relevant data is vital.
- Can the approach be used with different types of volumetric data, such as segmented and un-segmented datasets? It is found that adaptive abstraction can be a valuable tool in visualising different types of volume data, although some types are more suited to the approach (Sections 5.2 and 5.3).
- Can scene understanding be improved for complex datasets through the use of the adaptive abstraction approach? We conduct a user study to show how the approach can improve shape perception of important detail within complex datasets (Section 5.4).

5.2 Segmented Volume Data

In this section we show how the adaptive abstraction approach can be used with volume data, specifically segmented volume data. In Section 5.3, we will investigate complex single-object volume datasets, where it is necessary to highlight certain layers within the dataset. This type of dataset comprises the majority of existing volume data but we first focus on segmented datasets as we hypothesise they will particularly benefit from the adaptive abstraction approach. This is because when it is necessary to highlight a particular segmented object within the dataset, this target object will usually make up a small portion of the screen amongst other objects, in comparison to single-object datasets, where it is necessary to highlight a particular layer of the dataset using a single transfer function. Target objects within segmented datasets therefore need to be highlighted more effectively to both increase user understanding of the object and draw user attention to it. In this section we integrate the adaptive abstraction approach with an existing volume rendering system [MSMH09] to show the advantage of the adaptive abstraction approach with segmented volumetric datasets.

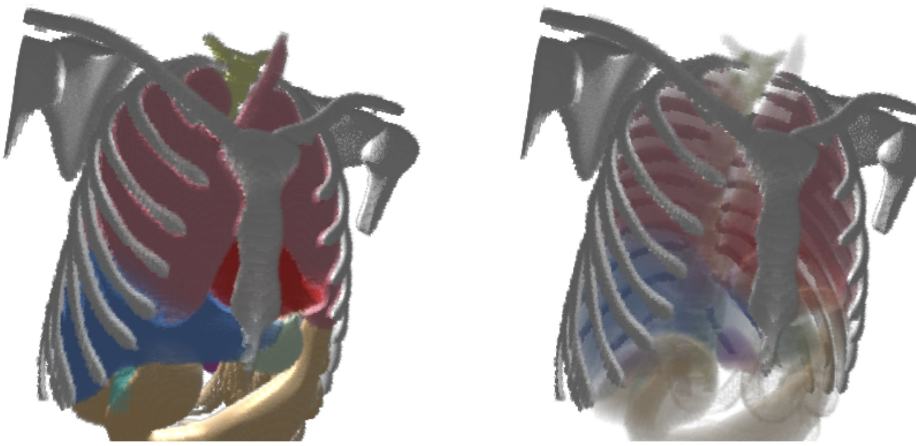

Figure 5.1: A pre-segmented dataset showing each distinct object rendered with a solid isosurface (left) and with one object rendered with a solid isosurface while the rest of the volume is rendered with a high opacity. As can be seen, the right image with high opacity does not display the background data as clearly as the left image.

Within such datasets, pre-segmentation takes place which allows each individual object to be given its own transfer function and change how each object is displayed. To utilise pre-segmented data, two inputs are required, which are the complete volume and a separate volume that distinguishes which voxel belongs to which object. A common problem with segmented data is that, even with accurate classification of each voxel, artifacts can appear and lead to poor quality renderings. Such artifacts can be seen in Figure 5.1, where the outlying voxels of each object appear jagged, despite accurate segmentation. There are various ways to combat such unwanted effects, such as using pixel resolution boundary filtering [TSH98, RGW$^+$03]. However, implementing such techniques was not the focus of this research and we treat the segmentation information without added computation.

To help highlight particular objects within segmented datasets, it is possible to adjust the transfer function of each object to make all objects highly transparent apart from the target object, which is rendered using a solid isosurface. While this is a valid form of visualisation, and it is discussed further in the next Section, it can make the structure of the surrounding objects unclear (see Figure 5.1). Therefore, after initial tests, it was decided to implement the adaptive abstraction approach with the segmented data by using a solid isosurface rendering. This was then combined with a fade-away technique, where the transparency of occluding voxels is increased to display target objects fully.

The segmentation information provided with these datasets allows us to integrate the false colour rendering pass, (described in Section 3.4), with the system. This is done

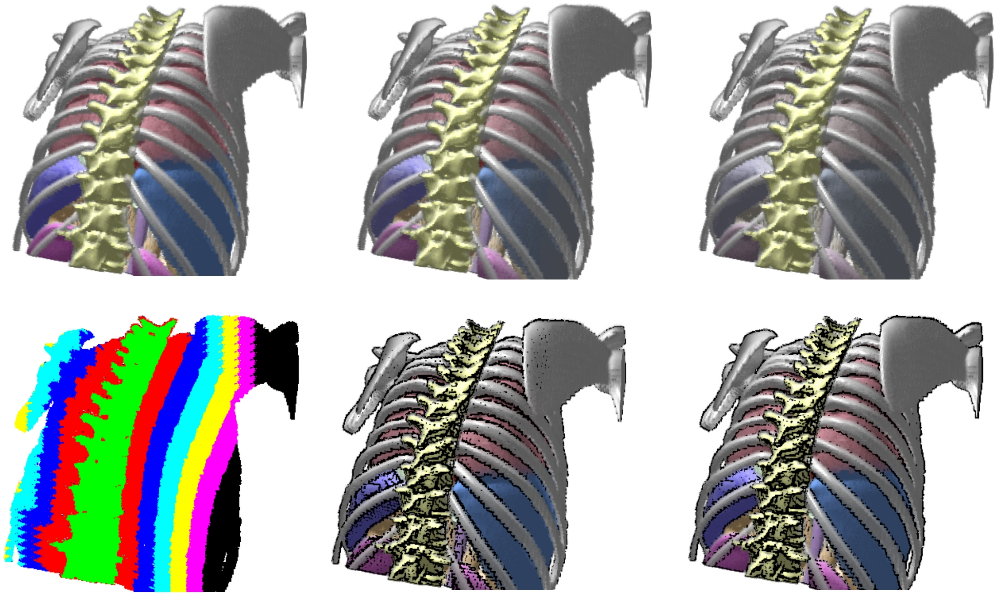

Figure 5.2: Various renderings of a torso with pre-segmented bones and organs. Top-left: Traditional rendering. Bottom-Left: Distance transform calculated with the spine as the target object. Top-Middle: Adaptive abstraction with multiple levels of saturation variation. Top-Right: Adaptive abstraction with two levels of saturation variation. Bottom-Middle: Adaptive abstraction with multiple levels of Difference of Gaussian edges. Bottom Right: Adaptive abstraction with two levels of Difference of Gaussian.

by using the segmentation information volume to render each object in a distinct colour. This in turn allows us to render each pixel in a suitable rendering style to achieve adaptive abstraction around a target object.

With the implementation of adaptive abstraction comes the choice of how many levels of stylisation should be used. Possible choices are a high contrast 2-level abstraction or a smoother, more subtle multi-level abstraction. To achieve more than 2 levels of abstraction, choices must be made as to how best to achieve a stylised focus. As discussed in Chapter 3, the adaptive abstraction can be calculated by using either screen-space distance or object-space distance. An image-space approach is more likely to give more suitable results as it gives smoother abstraction compared to object-space distance where the contrast might be large as the dataset may be comprised of a small number of large objects.

To achieve an adaptive abstraction for multiple-levels, a 2D distance transform is calculated using Euclidean distance to the nearest pixel of the target object. This is created by propagating outwards to find how far each pixel is away from the target object. To speed up the procedure, the search skips pixels when propagating outwards, which improves speed significantly but slightly decreases accuracy. However such a decrease in accuracy is acceptable as precision within the adaptive abstraction levels is not vital. An example of the distance map can be seen in Figure 5.2 (bottom-left).

Once the distance transform has been calculated, an adaptive abstraction can be created around the target object by using different levels of abstraction dependent on which distance level is at that pixel. An example of the adaptive abstraction approach combined with a segmented dataset can be seen in Figure 5.2. In these images, both two-level and multi-level abstractions are demonstrated with a stylised focus on the spine.

Our approach for rendering segmented datasets means each segmented object is rendered as a solid isosurface. While this allows for the entire dataset to be rendered clearly, when focusing on a particular object, there can be occlusion problems. This is demonstrated in Figure 5.3 where renderings show how some segmented objects are not visible from certain angles. To solve this problem, a fade-away approach is employed to show the target objects through occluding voxels.

A number of complex cut-away approaches exist for volumetric datasets, such as in [LRA$^+$07] or [VKG05]. We implement a more basic proof-of-concept technique to show that such types of approach can be successful and work well in tandem with adaptive abstraction. We use a ray-casting approach to check if any voxels are occluding the target object. If they are occluding, the opacity of the voxel is increased to let the target object display through the occluding voxels. An example of how this approach can help show

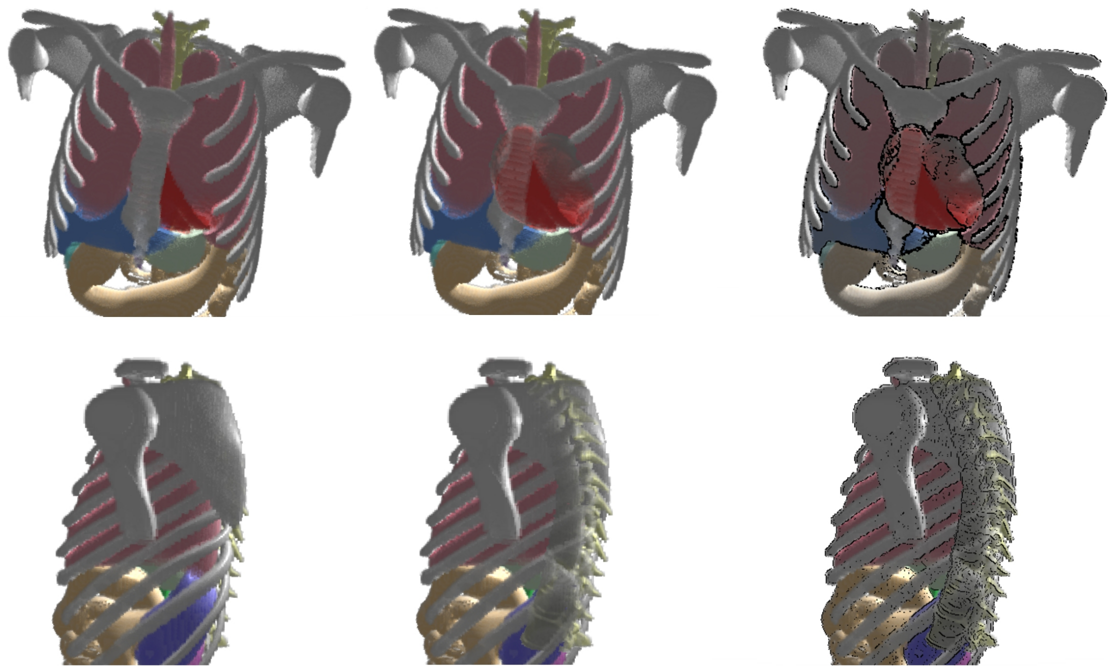

Figure 5.3: Various renderings of a torso focusing on the heart (top) and the spine (bottom). Left images: Traditional rendering. Middle images: Cut away approach showing entire target object. Right images: Fade-away approach focusing on target objects using Sobel edge detection and saturation variation (top), and Canny edge detection (bottom).

objects of interest and be used in combination with adaptive abstraction can be seen in Figure 5.3.

This section has shown how the adaptive abstraction approach can be used with volume data to create clearer renderings of pre-segmented data. As discussed, this type of dataset benefits particularly from the adaptive abstraction approach due to the multi-object nature of the scenes. However, in practice, only a small percentage of existing volume data is pre-segmented, so we move our focus to unsegmented data and investigate whether the adaptive abstraction approach can be adjusted to suit more common datasets.

5.3 Two-Level Volume Rendering

We integrate the adaptive abstraction approach with a volume rendering system to highlight specific voxels within complex un-segmented datasets. We use create our own system for two-level direct volume rendering [HMBG00] in combination with the adaptive abstraction approach, to enhance the saliency and fidelity of the most pertinent data within datasets. The system exploits enhanced direct volume rendering (DVR) techniques in combination with multiple levels of abstraction to provide enhanced detail without complete loss of peripheral information, which is retained for spatial context. Previous research [KSW06] has demonstrated that there are strong potential benefits to adopting such a paradigm in applications of volume visualisation.

We can employ a combination of image-space and object-space line drawing as well as textural and colour abstraction of peripheral data using image-space techniques. We integrate a two-level adaptive abstraction to fit with the two-level volume rendering system. Two-levels of abstraction are optimal for most unsegmented volume datasets as the target voxel data can often take up large portions of screen-space. This means the finished renderings can consist of a small amount of background data and a large percentage of target voxels. Within such images, a multi-level adaptive abstraction approach is unnecessary as focus is already on the target voxels. Instead it is preferable to use a two-level adaptive abstraction and focus on how best to stylise the data to increase user understanding.

The 3D data is first segmented and rendered into two layers of high and low importance and the layers are then post-processed with stylised enhancements and merged. We provide a mechanism by which the salience of the two layers can be modulated and integrated to provide optimal balance of focus and context. Figure 5.4 shows the pipeline used in order to generate the final perceptually enhanced renderings. The work described in this section was done in collaboration with a colleague, who provided the implementation of the volume rendering system (Section 5.3.1) as well as the implementation of the object-space edge detectors for volume data (Section 5.3.2).

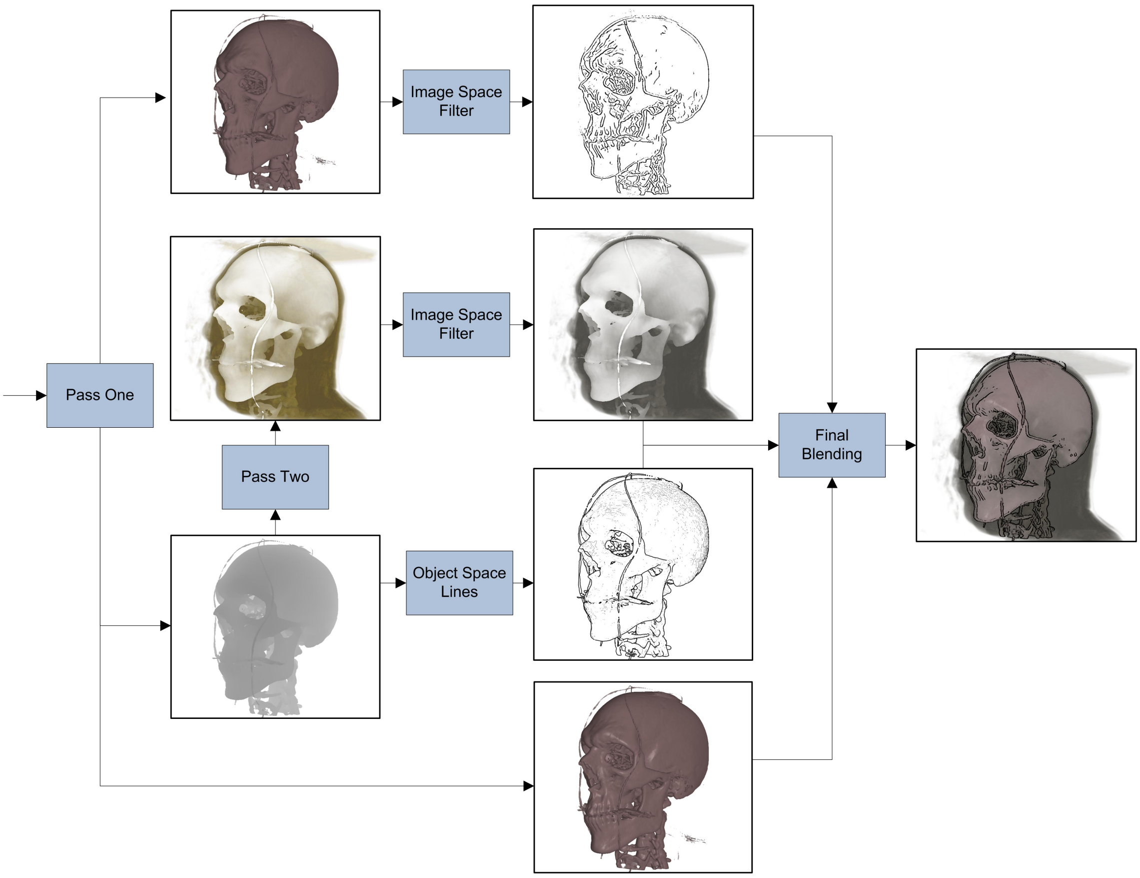

Figure 5.4: Pipeline overview for the two-level volume rendering system. In comparison to the methods described in previous chapters, the system does not use a false colour rendering pass but instead renders the target isosurface as well as the background volume. Object space lines are also added to the system.

5.3.1 Volume Rendering Techniques

While we used an existing volume rendering system in the previous section, in this section we create our own system to provide the necessary two-level renderings. The first step of our implementation is to perform a pre-processing step on the volume data using a 3D Sobel filter in order to calculate the normal vector for each voxel in the dataset. Since we focus our efforts on static volume data this normal information does not change from frame to frame and we are able to dramatically reduce our system's overall rendering time by performing this calculation once and saving it for future use in subsequent frames.

Our implementation performs two-pass volume rendering. In order to apply different abstractions to the two levels of the volume we modify a standard 3D texture slice volume renderer to output two images: one of the areas of the volume that the user has chosen to focus on, defined by a chosen isosurface value, and one of the remaining sections of the volume. We modified the basic algorithm so that the different segments of the volume are rendered in separate passes, enabling us to use custom rendering parameters for each segment in the volume. We further modify the output images from each pass by applying various image-space filters as a post-processing step. In a final stage the two images are blended together and rendered to the screen resulting in a complete volume view.

First Rendering Pass

Our first pass is used to generate a solid rendering of a user selected isosurface. We flag voxels that are part of the isosurface and render these voxels as normal. The pixel shader utilises multiple render targets in order to render multiple images in one pass. We generate an image using phong lighting rendering parameters, an image using phong lighting without any specular terms and also a depth image.

Second Rendering Pass

The second pass is used to render the remaining section of the volume. These voxels are rendered with high opacity to accompany the solid isosurface rendering of the target object. Our approach is to render all parts of the volume that were not displayed in the first rendering pass. This is done by using a custom pixel shader that uses the depth buffer generated in the first pass to perform depth testing. This discards any voxels that would normally be occluded and results in an accurate final image.

5.3.2 Non-Photorealistic Rendering Techniques

We integrate edge detection, texture abstraction and colour modulation with the volume visualisation system. In these renderings, the target object will comprise a large part of the visualisation. This means that making the target object more salient is a secondary goal to improving user understanding of the object itself. This is a goal that the stylistic shaders tested in earlier chapters are not suited to as these approaches, such as cel-shading, quantize the number of shades, which would make shape recognition harder. We instead focus on edge detection for the target object, with texture abstraction and colour modulation used to abstract the background data.

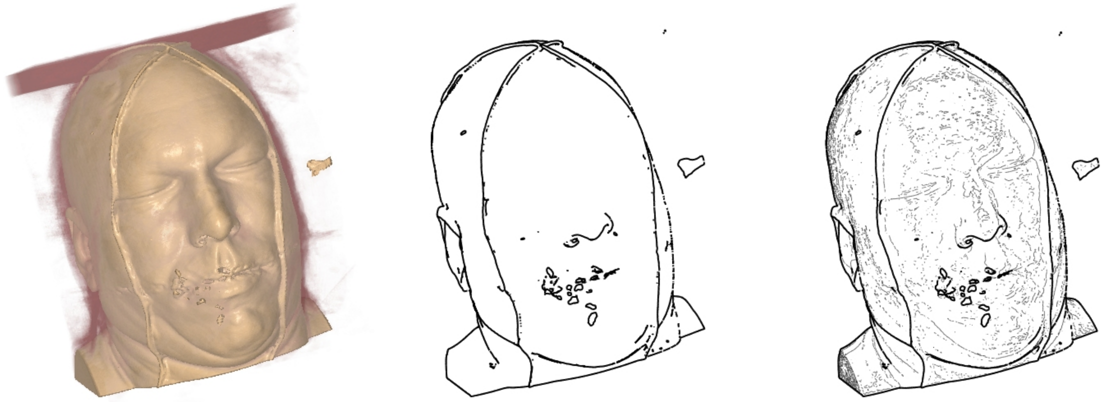

Figure 5.5: Object space line examples, Left: Original volume rendering, Middle: Silhouettes, Right: Suggestive Contours with Silhouettes.

As we have seen in the previous chapter, perceptual studies prove that adding edges to a scene can significantly increase the saliency of an object and positively affect search task performance. We also saw in past chapters that adding edges can make object recognition a more difficult prospect; however this is not a factor for most volumetric datasets. This is because there is typically only a single object in volumetric visualisations, and it is only a subsection of that object that needs to be highlighted to the user. Usually, this leads to the target voxels occupying a large part of the screen space in comparison to typical objects in polygonal scenes. In volumetric datasets, it is usually a layer of the whole dataset that needs to be highlighted, whereas in the polygonal scenes it can often be an individual, smaller object.

Five edge detection techniques were implemented for testing with the system. Three of these were image-space edge detectors; Sobel, Difference of Gaussian and Canny, while the other two were object-space edge detection techniques; silhouettes and suggestive contours. Previously we have not implemented object-space edge detectors with our adaptive abstraction technique, as they are techniques too expensive to run in real-time for large scenes. However, for volume rendering, the datasets are limited in size so frame-times can be guaranteed for most scene sizes.

The three image-space detectors were of varying complexity, as were the two object-space edge detectors. This allowed direct comparison between techniques and therefore we were able to determine whether more expensive techniques outperformed the simpler, faster methods in perceptual tests.

Edge Detectors

The limited size of volume data allows for the addition of object-space lines to the adaptive abstraction approach. We add silhouettes, which are defined as the edge separating a back facing and front facing surface. We also add suggestive contours, which are the surface locations where the radial curvature is zero, and its directional derivative is positive. Both line drawing styles can be seen in Figure 5.5. Our current implementation, created in collaboration with a colleague, for extracting object-space lines uses a parallelised version of the line extraction algorithm as described by Burns et al. [BKR+05]. We were able to optimise this algorithm in a number of different ways to provide real-time navigation of the volume dataset with object-space line enhancement. The image-space edge detectors, described in Section 3.2.2, were also integrated with the system.

Peripheral Abstraction Techniques

Experimental results from the previous chapter have shown that a user's perception of a scene can be altered by not only adding detail to the target object, but also by abstracting extraneous information. These styles are achieved by implementing two varied abstraction styles chosen due to their success in past work: Saturation variation and the Kuwahara filter. When determining the thresholds to use for each of these abstraction styles, as well as the edge detectors, the guidelines proposed in the previous chapter were used to attempt to achieve the best result.

5.4 Evaluation

5.4.1 User Study

An experiment was performed to investigate how each style, described in Section 5.3.2, affected a user's ability to determine the shape of the target isosurface. The primary aim of the visualisations is to use all volumetric data to give a clear impression of a certain part of each dataset while showing the rest of the object data for reference. For this reason it was necessary to test how well each style represented each dataset, which meant a shape perception experiment was the most suitable test and it was chosen over task-based or eye-tracking experiments. We hypothesised that by adding edges and abstractions to the dataset, the target isosurface level would be clearer to the user and the background volume would be de-emphasised, therefore increasing a user's understanding of the dataset.

To test a user's perception of shape, an experiment was run that involved placement of gauges on static images. This protocol was described in previous experiments such

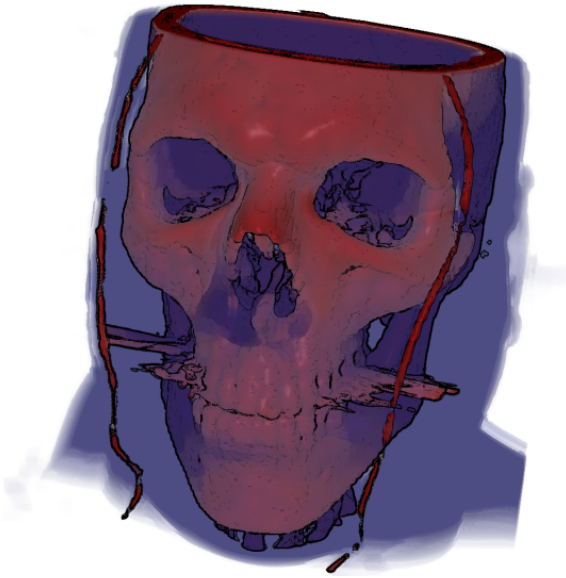

Figure 5.6: Example from the two-level volume rendering system, Sobel edge detection combined with suggestive contours and the Kuwahara filter applied to a volumetric model of a skull.

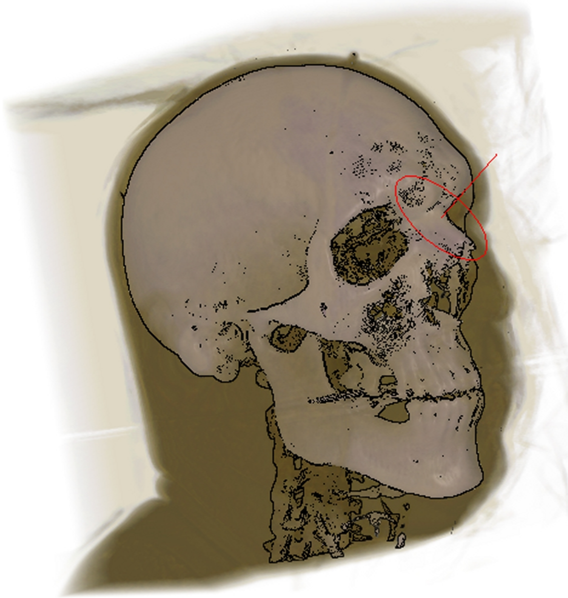

Figure 5.7: An example image shown to participants in the user study. The controllable gauge can be seen in red.

as [KDK92] and [CSD+09]. Participants were shown a series of images from the system and asked to rotate gauges that overlaid the images to match the surface normal to the isosurface at that point. Participants had no control over gauge position, only orientation. The orientation of the gauge was controlled by the mouse and the space bar was used to indicate that the participant was happy with the gauge position and ready to move on to the next trial. Each gauge was drawn as an ellipse and a single line as seen in Figure 5.7.

Gauge placement was pre-determined and each gauge was placed in an area of interest for each dataset. Placement was determined manually to ensure balance between datasets. Three diverse datasets were used in the experiment. These were scans of a male head, a brain and a foot. The scale of each dataset in each image was constant although the rotation of the dataset was changed slightly to avoid any learning curve with regards to the normals. Each image was shown 8 times with different gauge placements in each trial to fully test a user's shape perception with each style. This resulted in a total of 360 trials. 14 naive participants took part in the experiment (10M-4F), each with a knowledge of graphics and therefore surface normals.

There were 15 styles tested on each dataset, both individual and combination styles, which resulted in 45 images. A sample of these images can be seen in Figure 5.12. These styles can be seen listed in Table 5.1. Each of the 2 object-space line drawing, 3 image-space edge detection and 2 abstraction techniques were tested individually on each dataset. It was also investigated how combining the image-space and object-space edges would affect user estimations, which resulted in 6 additional styles. Also tested was a style containing suggestive contours, Difference of Gaussian edges and Kuwahara painterly effect. This style was added to determine how using 3 different stylisation types together could affect user perception of an image. Finally a set of images using the two-level volume rendering system was run with no stylisation at all as a baseline comparison. The two-level rendering was chosen as a baseline instead of a single shaded isosurface as our focus is on investigating the effectiveness of different NPR techniques in order to emphasise and abstract data. A single isosurface, with no background data, has only one area of interest and does not allow for both emphasis and abstraction to be applied at the same time. Table 5.1 describes each of the styles, and combinations of styles, tested in the experiment.

5.4.2 Results and Analysis

For each trial the angle between the correct surface normal and the estimated surface normal was calculated. The average angle was then calculated for each style across each dataset and an ANalysis Of VAriance (ANOVA) was then performed on the results where the conditions were *Dataset*(3) and *Style*(15). It was found that there was a main effect

Style number	*Style description*
Style 1	Sobel edge detection
Style 2	Difference of Gaussian (DoG) edge detection
Style 3	Canny edge detection
Style 4	Silhouettes
Style 5	Suggestive contours
Style 6	Sobel / Silhouettes
Style 7	DoG / Silhouettes
Style 8	Canny / Silhouettes
Style 9	Sobel / Suggestive contours
Style 10	DoG / Suggestive contours
Style 11	Canny / Suggestive contours
Style 12	DoG / Suggestive contours / Kuwahara filter
Style 13	Kuwahara filter
Style 14	Saturation variation
Style 15	No Style (Normal rendering)

Table 5.1: Description of the various styles used in the shape perception experiment.

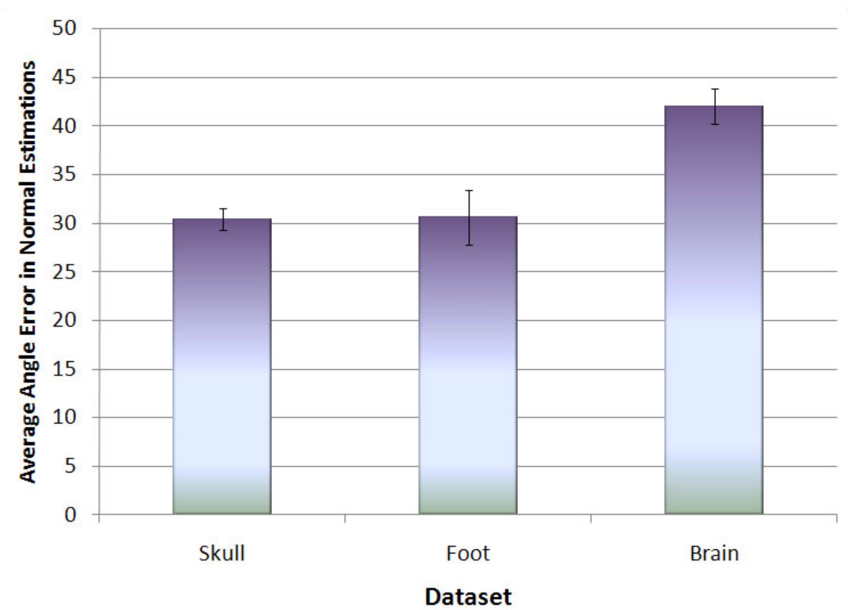

Figure 5.8: Average normal estimation errors for each dataset.

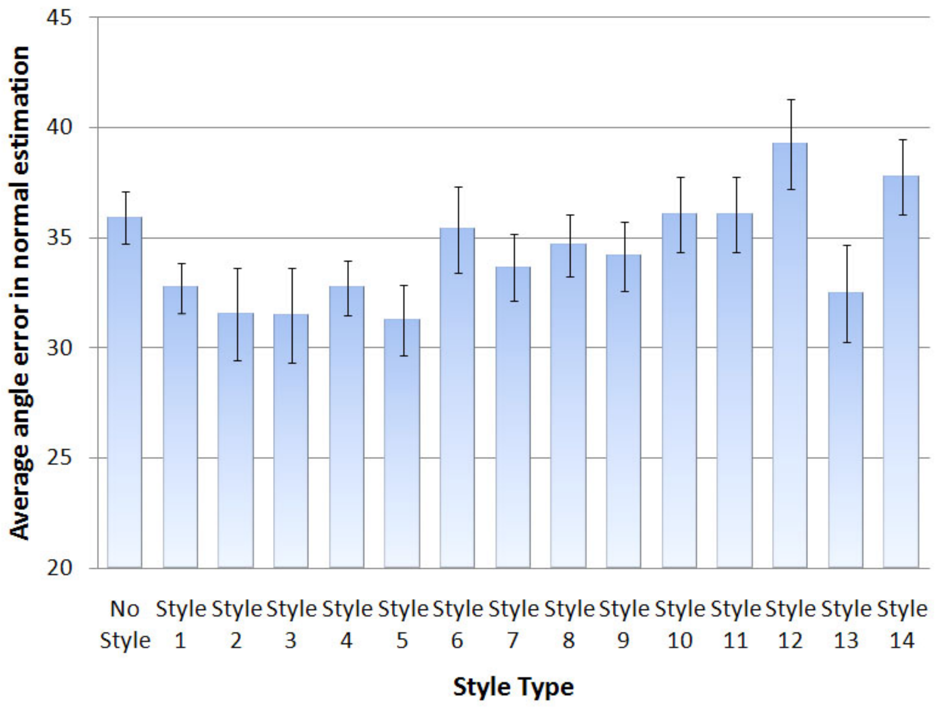

Figure 5.9: Average normal estimation errors for each style type over all datasets.

of dataset where the type of dataset shown affected user accuracy ($F(2, 26) = 15.253, p < 0.0005$). Post-hoc analysis was then performed using a standard Newman-Keuls test for pairwise comparison among means. As can be seen from Figure 5.8, error was significantly higher for the brain dataset than the skull or foot datasets ($p < 0.0003$ in both cases). Several participants noted after the experiment that they had prior knowledge of what shape a skull or foot skeleton should be, whereas they had no knowledge of the localised shape of a brain dataset. Therefore, we hypothesise that this previous knowledge affected the experiment results.

It was also seen that there was a main effect in style ($F(14, 182) = 7.6138, p < 0.00001$), where certain styles performed significantly better than others. Post-hoc analysis was performed on the data using a Newman-Keuls test and it was seen that the angle estimations from images with Difference of Gaussian edges, Canny edges ($p < 0.015$ in both cases) and suggestive contours ($p < 0.007$) were significantly better than those from renderings with no stylisation. These differences can be seen in Figure 5.9. Results also show that estimations from the images that used either basic silhouettes or Sobel edges did not have any significant difference with those from the unstylised renderings. This was somewhat expected as Sobel edge detection is a far simpler and faster image-space technique than

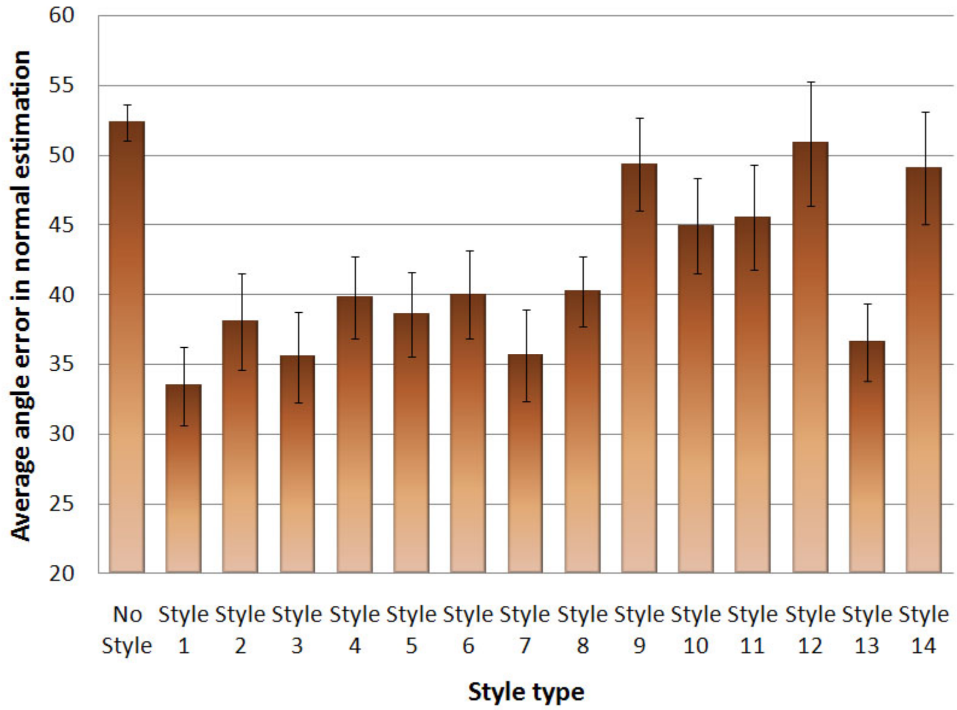

Figure 5.10: Average normal estimation errors for each style type for the brain dataset.

either Difference of Gaussian or Canny edge detection. Similarly, object-space silhouette detection is a much faster method than suggestive contours. These results imply that speed cannot be traded off if reliable and effective results are to be obtained.

Results also showed that there was no significant difference between results from images with no stylisation and those both with the Kuwahara filter and saturation variation. This shows that abstracting the background volume data does not have a large effect on user perception of the main volume data over all datasets. It can also be seen from our results that combining multiple styles had no significant impact on user accuracy. This result is quite surprising, especially in the case of combining suggestive contours with Difference of Gaussian edges or Canny Edges, as each one performed well individually. However, the results may be explained by the fact that adding too many edges of different types may clutter the object and obscure vital shape cues. This is a result that would reinforce earlier findings, from the previous chapter, where edge detectors appeared to hide vital information on smaller scale objects.

It was also found that there was an interaction between dataset and style ($F(28, 364) = 13.576, p < 0.00001$) which indicates that certain styles performed better within certain datasets. To investigate this effect, the results from each style were isolated and compared. While it was found that there was no change in the styles that performed well for the foot and skull datasets, a number of styles performed significantly better than normal rendering in the brain dataset. As Figure 5.10 illustrates, all styles apart from the mixed style, saturation variation and the combination of Sobel edges and suggestive contours performed significantly better than normal rendering in this set ($p < 0.02$ in all cases). Within the brain dataset there was much occlusion of the isosurface from the background volume data; this meant there were less cues for shape than in the other datasets and the styles were therefore more effective in conveying shape. Also, as mentioned earlier, participants were less familiar with the shape of the brain dataset so the cues added made more of a difference than in the other sets.

It was noted that there was an effect of gauge orientation in the experiments. It was found that if the correct gauge position was less than 15 degrees from the camera view vector then user accuracy was significantly increased. If the angle between the correct gauge position and the camera view vector was above 60 degrees then the gauge estimation was significantly harder. There was no significant difference in accuracy if the angle was between 15 and 60 degrees, which represented approximately 80 percent of the gauges tested in the experiment. Within the experiments the gauge angles tested were distributed randomly which means that more experiments would be necessary to determine what effect, if any, gauge position has on how each style performed. This is an interesting result, although it is unexpected as such an effect was not noted in previous work using the same experimental system [CSD$^+$09].

Figure 5.11 shows the average frames per second for the brain dataset at varying resolutions and styles. Our two pass modification incurs a computation penalty of between 16-40% in comparison with an unmodified volume render but allows custom rendering parameters and post-processes to be applied separately to each volume segment. Once the volume has been segmented, applying the majority of the image-space NPR filters is negligible; however both the object-space lines and Kuwahara filtering incur a relatively large performance penalty.

5.4.3 Saliency Tests

While the main focus of the volume rendering system was on increasing user understanding and not on affecting user gaze behaviour, as is described in earlier chapters, it was still of interest to test saliency using the automatic saliency metric described in Section 4.3 [IKN98]. The target object comprises a large portion of the rendering so the user finding the target object is trivial. However, we show the background data for reference but do

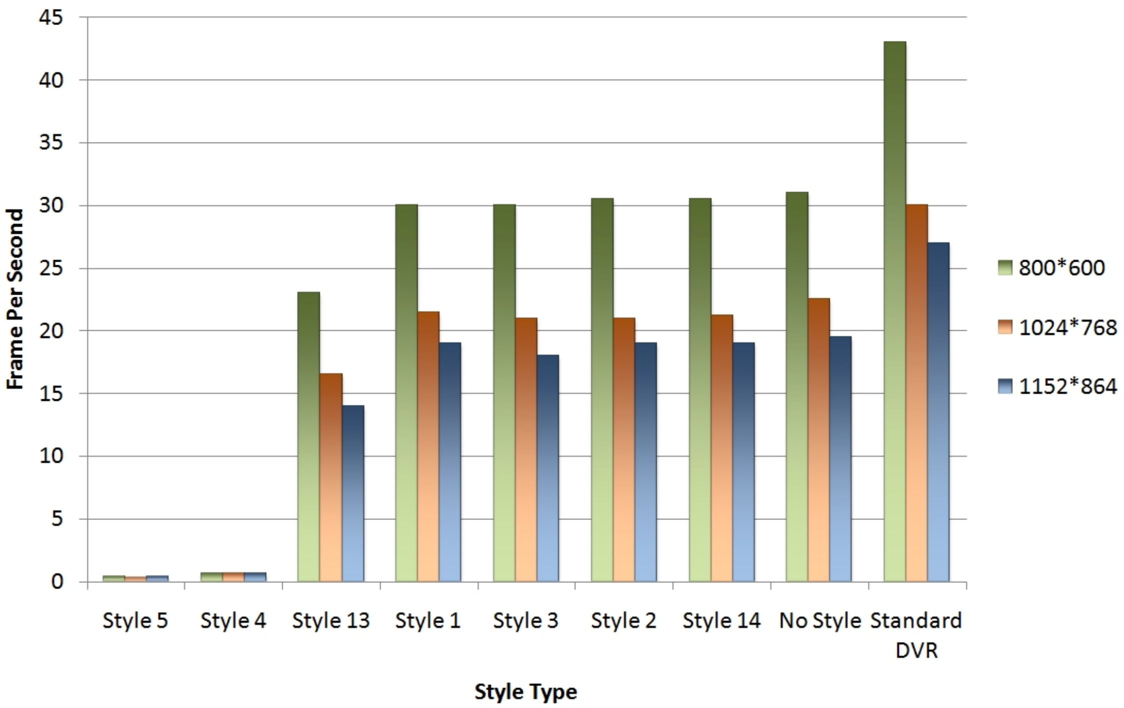

Figure 5.11: Average frames per second for the brain dataset at varying resolutions and styles.

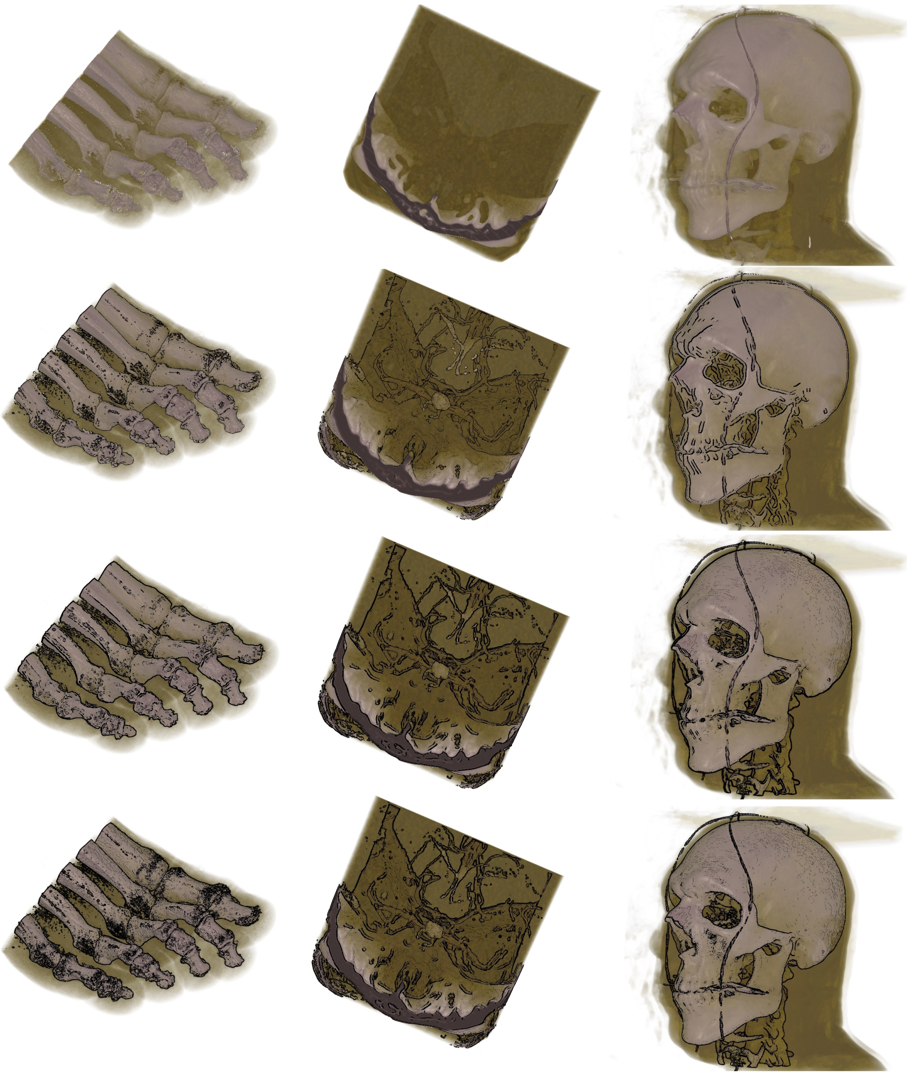

Figure 5.12: Some sample images from the experiment: First Row - Normal rendering; Second Row - silhouettes, suggestive contours, Canny; Third Row - Sobel, Canny combined with silhouettes, Sobel combined with suggestive contours; Fourth Row - Difference of Gaussian combined with suggestive contours, Canny combined with suggestive contours, Difference of Gaussian combined with suggestive contours and Kuwahara filtering.

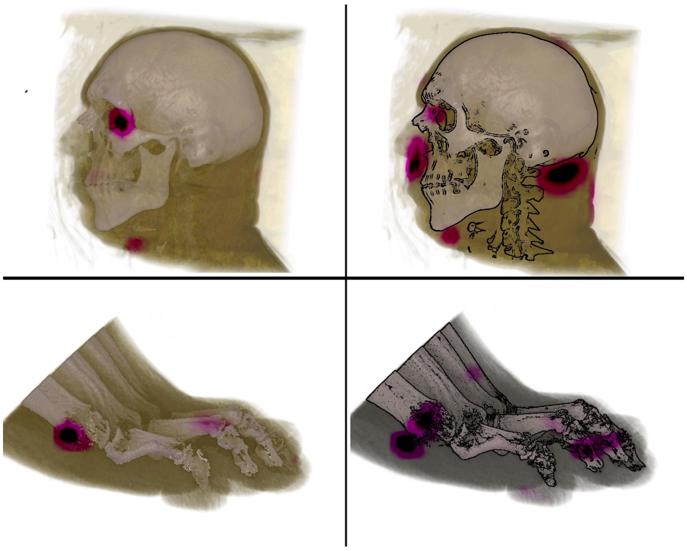

Figure 5.13: Examples of how adding stylisation can affect colour contrast within scenes. The left images are normally rendered while the right images use the Sobel filter (top) and a combination of suggestive contours and saturation variation (bottom). Colour contrast is indicated in purple and measured using the automatic saliency metric presented by Itti et al. [IKN98]. These images show how the saliency of the target isosurface can be increased and manipulated using various styles.

not want it to be salient in comparison to the target isosurface.

A number of images were taken from both the two-level volume rendering system and the segmented volume data system (Section 5.2). Each set of images, taken from identical viewpoints, was tested with both unstylised renderings and adaptive abstraction images, and compared. The images were then tested using the saliency metric and while results were somewhat mixed, it was seen that by adding stylisation to the images, saliency could be effectively influenced.

For the two-level system, the saliency of the isosurface within the image can be increased while the impact of the surrounding volume can be reduced. This is especially true for increasing colour contrast within images using edge detectors, as can be seen in Figure 5.13. There was no large difference in how the edge detectors performed against each other although suggestive contours increased the saliency within the centre of the isosurface more than others. Similar results were found when it was investigated whether multi-level adaptive abstraction can have the same effect on pre-segmented data. Figures 5.14 (for the two-level rendering system) and Figure 5.15 (for the segmented data system) show that the most salient point of each image (marked by a yellow circle) can be changed using these stylisations on the two-level volume renderings.

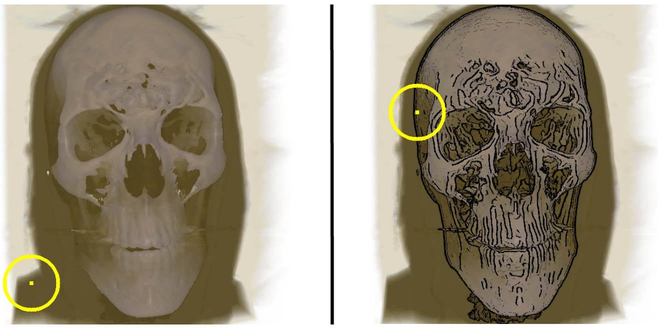

Figure 5.14: Example of how the adaptive abstraction approach can change the most salient point in the image, marked with a yellow circle in each image; Left: Normal rendering. Right: Canny. As can be seen, adding stylisation can change the most salient point to the target isosurface.

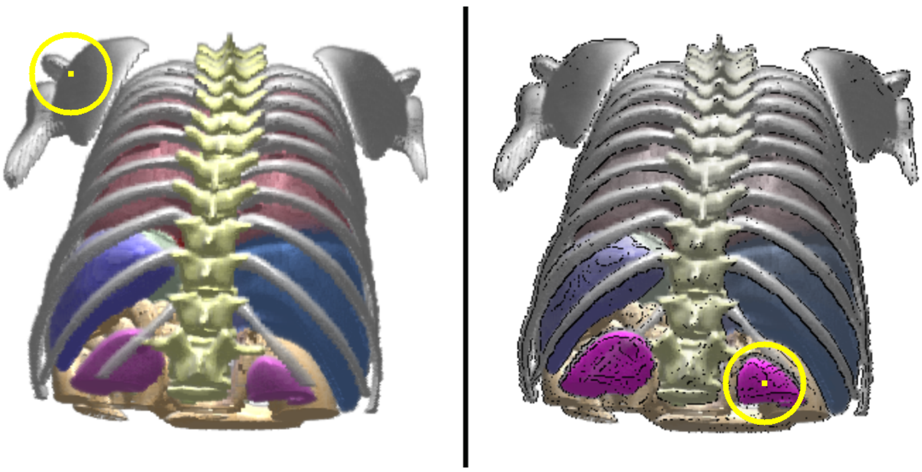

Figure 5.15: Examples of how multi-level adaptive abstraction affects segmented datasets. The most salient point in each image marked with a yellow circle; Left: Normal rendering. Right: Adaptive Canny edges combined with saturation variation.

5.5 Summary and Conclusion

In this chapter we have shown how the adaptive abstraction approach presented earlier in the thesis can be used to create clear, stylised renderings in the practical context of volume visualisation systems. Volume rendering poses some unique problems due to the complex nature of the datasets. The adaptive abstraction approach has been shown to work well with segmented datasets where important objects from within the dataset can be highlighted using the stylised focus and a fade-away view. We have also shown how the approach can be effectively combined with a two-level volume rendering system, using a transfer function to select certain voxel densities before highlighting the target voxels using various line drawing styles while displaying and abstracting the rest of the volume dataset for reference. The system described can be altered to achieve different types of results, as can be seen in Figure 5.16.

We also carried out experiments that show that our adaptive abstraction approach is capable of increasing object saliency and improving user understanding of shape whilst preserving context with the surrounding peripheral data. Results from the experiments showed that certain edge detectors can have a significant effect on a user's perception of object shape. It was also shown that the techniques described here are especially effective for datasets that contain a large amount of peripheral volume data, and also for datasets that users are not familiar with.

The work described in this chapter shows how the adaptive approach and experimental results from previous chapters can be used with complex datasets to produce clear, stylised renderings that effectively highlight certain data. Visualisation of volume data can be combined with the adaptive abstraction approach to produce perceptually influential renderings that can increase user understanding of target objects. Volume visualisation is an example application where the proposed approach would be a valuable tool in increasing the clarity of renderings.

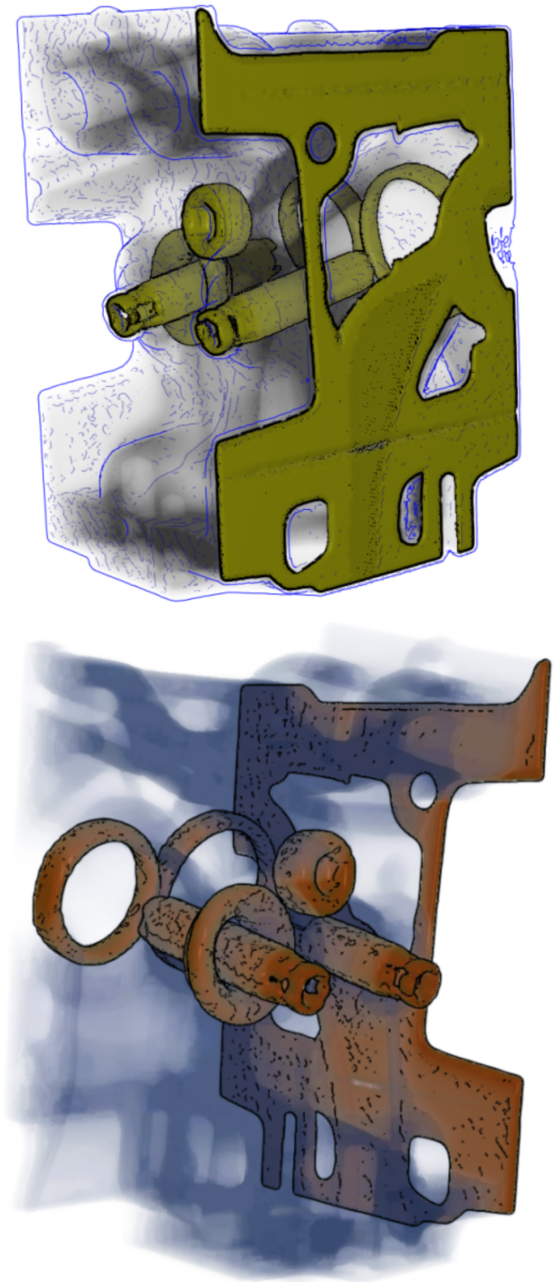

Figure 5.16: Images of an engine block showing how the system, described in Section 5.3, can be adapted to create different types of renderings. The first image shows an engine block with the target isosurface highlighted using Difference of Gaussian edges and the surrounding volume enhanced using suggestive contours. The second image shows the target isosurface highlighted using suggestive contours and the Kuwahara filter while also using a multi-dimensional transfer function.

Chapter 6

Conclusions and Future Work

In this chapter, we provide an overview of the contributions of this thesis, along with some avenues of future work.

6.1 Summary of Contributions

6.1.1 New techniques

We have presented a new approach for creating adaptive abstraction and influencing user perception within real-time scenes. We have shown how a variety of image-space styles can be adapted to create a number of levels of abstraction and how such styles can then be applied to a scene to highlight important information and remove unnecessary detail. We also add object-space shading information to create styles that use both image-space and 3D data.

We have demonstrated how the adaptive approach can be utilised with just 2-levels of abstraction or with multiple levels. This flexibility allows the methods to be used successfully on a variety of types of datasets. While the target objects must be predefined, the rest of the process is fully automatic and applicable to real-time scenes. The system has been shown to work well with complex scenes to make information clearer and quicker to understand.

6.1.2 New Experimental Results

We have gained new insights into the effects of using variable levels of non-photorealistic rendering to affect user perception in real-time scenes. We found that, in general, throughout the series of experiments, non-photorealistic styles affected scene perception of the participants. These results are useful in terms of proving that non-photorealistic styles are a valid way of presenting complex data. Results can also help to improve the effectiveness of the approaches by exhibiting how scene context and stylisation levels can affect the success of non-photorealistic techniques.

- **Context**

  Our main finding was that the success of each style in affecting user perception relied largely on three factors. We found that the effectiveness of using non-photorealistic stylisations depended greatly on the task, the style and the scene type. While this was somewhat expected, we did not foresee the extent to which this was true. We have shown how styles can have a positive influence in one task while having no effect, or even a negative effect, in another. We believe our results show that stylisation choice is a hugely important one, and our results can give guidelines to ensure that the most suitable styles can be chosen for particular types of task and scene.

  We tested a variety of styles in a number of different ways to discover how NPR approaches affected different aspects of user perception. Eye-tracking studies, timed search tasks and shape perception experiments were run during the course of our research. Using these varied experiments, we were able to determine how adaptive abstraction approaches can affect different aspects of user perception. The results from these experiments show how each style performed very differently depending on the task.

  During the experiments we tested two distinct types of non-photorealistic approach, those that attempt to increase the saliency of important objects, such as edge detection and stylistic shaders, and those that abstract background objects to get rid of extraneous data, such as colour variation or painterly styles. We found that both can be effective in altering how a user perceives a scene, and such styles can be combined and tailored for a specific task.

  Other interesting findings were that there was a high sensitivity to changing the internal thresholds and levels for each style with respect to how each style performed. Altering stylisation levels can affect the success of that style in drawing user gaze or affecting task performance. This indicates that care should be taken when choosing the level of stylisation for particular scenes, as the most aesthetic is not necessarily the most effective. We found that scene context and complexity has a large bearing on how to pick the best style thresholds. The results from our research give guidelines on how to choose the best thresholds for particular styles and scenes.

- **Style Performance**

  Our findings indicate that for certain tasks, especially those that require user eye gaze to be drawn to particular objects, edge detectors can be particularly effective. While there has been past work in how well different edge styles alone represent shape [CSD$^+$09], we have shown that overlaying edges onto existing scenes can increase the saliency of particular objects and draw the eye. We have also seen that certain edges, such as Canny edges or suggestive contours, can help the user perceive the shape of objects. However, we have also seen that the same styles can make object recognition a more difficult prospect as the edges may inadvertently disguise the object, which can affect task performance.

  We found that texture abstraction, such as painterly filtering, can effectively remove unnecessary detail, which can improve task performance. While this style does not directly influence the target objects, removing detail from unimportant regions can help make certain tasks, especially search tasks, quicker to comprehend and complete. Similar results were discovered for simpler colour modulation filters such as saturation variation which performed well for all datasets and tasks. However, somewhat surprisingly, neither of these styles affected initial eye gaze in scenes, which leads us to believe that they do not increase the low-level saliency of target objects hugely, yet can still help in search tasks.

  It was seen in the two-level abstraction experiments (Section 4.6) that stylising the background data using saturation and painterly styles led to no change in the shape perception of the target object. This result was expected and confirms that this type of abstraction is suited to helping to increase the saliency of target objects but cannot increase understanding of those objects. This is a result that edge detection and stylistic shading can achieve. The stylistic shading techniques that were tested gave mixed results. While there was some improvement in certain tasks, the types of shading tested did not perform significantly better than the other types of stylisation tested. Having said this, certain existing shaders would be suited to particular tasks such as the exaggerated shading approach of Rusinkiewicz et al. [RBD06], which might be suitable for helping shape perception.

6.1.3 New Guidelines

Our studies not only furthered the knowledge on perception of NPR with adaptive abstraction, we also provide some perceptual guidelines in order to provide effective and stylistic renderings. From our experimental studies we have come up with a number of recommendations on how to create effective renderings using adaptive abstraction and how to best utilise a variety of style types. We have shown what factors must be taken into account when choosing styles and thresholds. We have established when certain styles are best suited and which levels of abstraction should be used in scenes of varying complexity. These results can be valuable for anyone intending to use non-photorealistic styles in real-time scenes.

6.1.4 Example Applications

We have shown how our adaptive abstraction approach can be successful in a number of different types of application. It was seen that NPR styles can be used effectively in scenes of varying complexity. The scenes tested included character scenes, which indicates that the methods could be successful in interactive applications which render large crowds such as large scale games. The approach was also successfully tested with large urban simulations, which shows that either architectural visualisations or virtual world representations could benefit from adaptive abstraction. In this type of application, there can be a need to highlight certain objects within the scene such as buildings, tourist attractions or current objects of interest.

We have also focused on volumetric visualisations as such highly complex datasets can be hard to render in a clear and comprehensible manner. This type of dataset exhibits the strength of the adaptive abstraction approach as the amount of information contained in large volumes can make them extremely difficult to present. We have shown how adding non-photorealistic styles can increase the clarity of volumetric visualisations. This highlights the potential of the approach in medical, scientific, biological and technical imaging, which are areas that use volumetric data.

6.2 Reflections on Experimental Design

A number of different psychophysical techniques were used for the different experiments in this thesis. There were some lessons learned in this process that may be of use to others wanting to perform similar studies.

6.2.1 Timed Search Tasks

A variety of designs were used throughout the experiments to test a number of aspects of user perception. Two-alternative forced choice and timed-search tasks were used to investigate how using adaptive abstraction around objects crucial to a task could change user performance. Within the search tasks, users were asked to find a particular object and, in another experiment, asked to find matching pairs from within a number of objects. These experiments tested different aspects of user reactions. The simple search task involved quickly scanning a scene until a target object was found, whereas the more difficult matching pairs task meant inspecting the scene more carefully.

6.2.2 Eye-Tracking Data

When retrieving eye-tracking data from users the main obstacle was what task or instructions were given to the user viewing a scene. While it would be interesting to combine an interactive task with eye-tracking data we felt that to allow the user control of the keyboard or mouse would interfere with the user's attention and therefore result in unsatisfactory eye-tracking data. We instead used two distinct approaches. For the first we gave the user a search task where they were asked to count the occurrences of a particular object which ensured the user was constantly searching the scene. For a complex urban scene we simply asked the user to view the scene as we were interested in how eye-gaze would be affected without any task or outside influence. While interesting and worthwhile results were achieved from both tasks, there was no definitive answer to what method is best for gauging eye behaviour.

6.2.3 Shape Perception Experiments

We chose a gauge orientation experiment [KDK92] [CSD+09] to determine how well a user could perceive the shape of objects when non-photorealistic styles were applied. The user was instructed to orient fixed gauges on an object surface until they matched the surface normal. One large question when setting up the experiment was how to place the gauges for the experiment. Random placement may result in easier orientation decisions for particular datasets which would skew the results. We instead decided to manually place the gauges in similar positions for each style and dataset to ensure a balanced study.

6.2.4 Experiment Conditions

During the course of the experiments it was found that due to the nature of adaptive abstraction, there was a large number of conditions to be tested such as style, dataset as well as various style parameters. While this large number of parameters was necessary for our experiments, it may have made it hard to pinpoint exact ideal parameters for styles. The high number of variables is a reason we can only provide guidelines, and not exact metrics, on how to get the most effective results from various styles.

6.3 Future Work

In our current approach, we created a stylised focus in real-time scenes using adaptive abstraction. We used a number of methods to determine how to vary the abstraction across an image to create a focus. We tested approaches such as gradual abstraction using image-space distance from a particular point or from a target object, abstracting the scene on a per-object basis using either screen distance or object-space distance, or limiting the scene to two-levels of abstraction for the target object and the rest of the scene. The choices about the abstraction approach were made during the course of the research dependent on the particular scene and which method suited the data type and scene complexity. This may be an avenue for further research to compare the perceptual difference between the types of adaptive abstraction and how users react to the aesthetic difference between each one and normally rendered scenes.

In our research we focused on the creation of adaptive abstraction and its possible perceptual influence, with aesthetic value being a secondary goal. We ensured throughout our research that styles were both convincing and effective, and also got feedback from the experiment participants on the authenticity of each style. However this was not our focus and further qualitative research regarding aesthetic real-time styles would be valuable.

Because we were investigating how adaptive abstraction could affect a large number of types of data with various styles, it proved impossible to produce style metrics that would be suitable for all applications. We did produce a number of guidelines on how to choose the most suitable styles and thresholds for particular types of scenes. We also showed how styles performed on scenes with minimal semantic connotations. Finding perceptual metrics for all styles was not our aim but we feel with additional research it would be possible to produce definitive threshold values for various styles and datasets to produce optimal results. Our research has shown thresholds would vary between scenes so this type of experiment would need to be done for each type of dataset for which the metrics were required.

We have seen in our research that volumetric datasets benefit from our approach due to the complex nature of the visualisations. We are currently investigating how volumetric data can be rendered to enhance this benefit using a larger number of styles, such as ridges and valley lines, curvature stylisations and more complex shading approaches. While these styles are more complex than many of the styles tested in this thesis, the size of volumetric datasets is limited so more expensive techniques can be employed while retaining interactive rates.

During the course of the research we used non-experts as experiment participants;

which allowed us to discover how users reacted to our methods without being affected by prior knowledge of scene data. While this was desired when showing certain datasets to non-experts, such as the medical data tested, it would be useful to get the feedback of professionals who are very familiar with the data. While this familiarity may mean the effect of adaptive abstraction would be lessened, as shown in our experiments, their evaluation of the images would certainly be useful in understanding how to best present this data to non-experts. This would allow doctors and experts to evaluate the visualisations and give their views on the strengths and weaknesses of such images. This feedback would be invaluable in producing renderings that could improve user perception while retaining valuable scene information.

We focused in our research on real-time non-photorealistic styles, suitable for interactive applications. The generation of off-line non-photorealistic rendering is a well researched subject but the perceptual questions raised in our studies also apply to more complex styles. While some of the results found in the experiments in this thesis will give indications of how best to use computationally expensive techniques, user perception will be affected differently in static photos as well as animations and videos with no user interaction. It would be valuable to compare perceptual differences in adaptive abstraction for various styles and scene types such as those described in [DS02] and [SD02]. This is certainly an area that would benefit from further perceptual studies.

A premise of our research was that we wanted to influence user perception while maintaining a natural looking scene and preserving background information for context. A further avenue of research would be to compare user reaction to adaptive abstraction and more obvious saliency enhancing techniques, such as simple blurring or flashing to direct user gaze [MBG08b]. A qualitative comparison combined with perceptual differences would give a clear indication of the advantages of an adaptive abstraction approach.

Bibliography

[ALD05] D. Acevedo, D. Laidlaw, and F. Drury. Using visual design expertise to characterize the effectiveness of 2d scientific visualization methods. In *VIS 2005: 16th IEEE Visualization Conference*, page 101, 2005.

[BF08] M. Burns and A. Finkelstein. Adaptive cutaways for comprehensible rendering of polygonal scenes. In *SIGGRAPH Asia '08: ACM SIGGRAPH Asia 2008 papers*, pages 1–7, 2008.

[BKR+05] M. Burns, J. Klawe, S. Rusinkiewicz, A. Finkelstein, and D. DeCarlo. Line drawings from volume data. *ACM Transanctions on Graphics*, 24(3):512–518, 2005.

[BNTS07] A. Bousseau, F. Neyret, J. Thollot, and D. Salesin. Video watercolorization using bidirectional texture advection. *ACM Transactions on Graphics*, 26(3):104, 2007.

[BTM06] P. Barla, J. Thollot, and L. Markosian. X-toon: an extended toon shader. In *NPAR '06: Proceedings of the 4th international symposium on Non-photorealistic animation and rendering*, pages 127–132, 2006.

[Can86] J.F. Canny. A computational approach to edge detection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8:769–798, 1986.

[CCF94] B. Cabral, N. Cam, and J. Foran. Accelerated volume rendering and tomographic reconstruction using texture mapping hardware. In *VVS '94: Proceedings of the 1994 symposium on Volume visualization*, pages 91–98, 1994.

[CCL02] K. Cater, A. Chalmers, and P. Ledda. Selective quality rendering by exploiting human inattentional blindness: looking but not seeing. In *Proceedings of* *the ACM symposium on Virtual reality software and technology*, pages 17–24, 2002.

[CDF+06] F. Cole, D. DeCarlo, A. Finkelstein, K. Kin, K. Morley, and A. Santella. Directing gaze in 3d models with stylized focus. In *EGSR '06: Proceedings of the 17th Eurographics Symposium on Rendering*, pages 377–387, 2006.

[CF09] F. Cole and A. Finkelstein. Fast high-quality line visibility. In *I3D '09: Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 115–120, 2009.

[CFVR01] J. Claes, F. Di Fiore, G. Vansichem, and F. Van Reeth. Fast 3d cartoon rendering with improved quality by exploiting graphics hardware. In *no wait, theyre not but they get put inProceedings of Image and Vision Computing New Zealand (IVCNZ) 2001*, pages 13–18, 2001.

[CGL+08] F. Cole, A. Golovinskiy, A. Limpaecher, H. Stoddart Barros, A. Finkelstein, T. Funkhouser, and S. Rusinkiewicz. Where do people draw lines? *ACM Transactions on Graphics*, 27(3):1–11, 2008.

[CH03] J.P. Collomosse and P.M. Hall. Cubist style rendering from photographs. *IEEE Transactions on Visualization and Computer Graphics*, 9(4):443–453, 2003.

[CH05] J.P. Collomosse and P.M. Hall. Genetic paint: A search for salient paintings. *Lecture Notes in Computer Science (Proc. EvoMUSART)*, 3449:437–447, 2005.

[CI06] R. Carmi and L. Itti. Causal saliency effects during natural vision. In *ETRA '06: Proceedings of the 2006 symposium on Eye tracking research & applications*, pages 11–18, 2006.

[CKD96] C. Christou, J.J. Koenderink, and A.J.V. Doorn. Surface gradients, contours and the perception of surface attitude in images of complex scenes. *Perception*, 25:701–713, 1996.

[CKY01] M. Chen, A. Kaufman, and R. Yagel. *Volume Graphics*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2001.

[CN94] T.J. Cullip and U. Neumann. Accelerating volume reconstruction with 3d texture hardware. Technical report, University of North Carolina at Chapel Hill, 1994.

[CRH05] J.P. Collomosse, D. Rowntree, and P.M. Hall. Stroke surfaces: Temporally coherent artistic animations from video. *IEEE Transactions on Visualization and Computer Graphics*, 11(5):540–549, 2005.

[CSD+09] F. Cole, K. Sanik, D. DeCarlo, A. Finkelstein, T. Funkhouser, S. Rusinkiewicz, and M. Singh. How well do line drawings depict shape? *ACM Transactions on Graphics*, 28(3):1–9, 2009.

[CTP+03] M. Cunzi, J. Thollot, S. Paris, G. Debunne, J-D. Gascuel, and F. Durand. Dynamic canvas for immersive non-photorealistic walkthroughs. In *Proceedings: Graphics Interface 2003*, pages 121–130, 2003.

[Cur98] C. J. Curtis. Loose and sketchy animation. In *SIGGRAPH '98: ACM SIGGRAPH 98 Electronic art and animation catalog*, page 145, 1998.

[DBHM] D.J. Duke, P.J. Barnard, N. Halper, and M. Mellin. Rendering and affect. *Eurographics '03: Proceedings of Eurographics, Computer Graphics Forum*, 22.

[DFRS03] D. DeCarlo, A. Finkelstein, S. Rusinkiewicz, and A. Santella. Suggestive contours for conveying shape. *ACM Transactions on Graphics*, 22(3):848–855, 2003.

[DR07] D. DeCarlo and S. Rusinkiewicz. Highlight lines for conveying shape. In *NPAR '07: Proceedings of the 5th international symposium on Non-photorealistic animation and rendering*, pages 63–70, 2007.

[DS02] D. DeCarlo and A. Santella. Stylisation and abstraction of photographs. *ACM Transactions on Graphics*, 21:769–776, 2002.

[EKE01] K. Engel, M. Kraus, and T. Ertl. High-quality pre-integrated volume rendering using hardware-accelerated pixel shading. In *HWWS '01: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS workshop on Graphics hardware*, pages 9–16, 2001.

[Ent98] PolyGram Filmed Entertainment. What dreams may come. Motion Picture, 1998.

[ER00] D. Ebert and P. Rheingans. Volume illustration: Non-photorealistic rendering of volume models. In *Proceedings of the conference on Visualization '00*, pages 195–202, 2000.

[Fil05] Dimension Films. Sin city. Motion Picture, 2005.

[Gam09] Rockstar Games. Grand theft auto: Chinatown wars. Nintendo DS game card, 2009.

[GG01] B. Gooch and A. Gooch. *Non-Photorealistic Rendering.* A.K. Peters Ltd, Natick, MA, 2001.

[GGSC98] A. Gooch, B. Gooch, P. Shirley, and E. Cohen. A non-photorealistic lighting model for automatic technical illustration. In *ACM SIGGRAPH 1998: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 447–452. ACM Press / ACM SIGGRAPH, 1998.

[Gou98] H. Gouraud. Continuous shading of curved surfaces. In *Seminal graphics: pioneering efforts that shaped the field*, pages 87–93. ACM, New York, NY, USA, 1998.

[GRG04] B. Gooch, E. Reinhard, and A. Gooch. Human facial illustrations: Creation and psychophysical evaluation. *ACM Transactions on Graphics*, 23(1):27–44, 2004.

[GW02] A.A. Gooch and P. Willemsen. Evaluating space perception in NPR immersive environments. In *NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering*, pages 105–110, 2002.

[Hae90] P. Haeberli. Paint by numbers: abstract image representations. In *Computer Graphics (Proceedings of SIGGRAPH '90)*, pages 207–214. ACM, 1990.

[Hal04] M. Haller. Photorealism or/and non-photorealism in augmented reality. In *VRCAI '04: Proceedings of the 2004 ACM SIGGRAPH international conference on Virtual Reality continuum and its applications in industry*, pages 189–196, 2004.

[Hau01] A. Hausner. Simulating decorative mosaics. In *SIGGRAPH 2001: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 573–580. ACM Press / ACM SIGGRAPH, 2001.

[HBH03] M. Hadwiger, C. Berger, and H. Hauser. High-quality two-level volume rendering of segmented data sets on consumer graphics hardware. In *VIS '03: Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, page 40, 2003.

[HE04] J. Hays and I. Essa. Image and video based painterly animation. In *NPAR '04: Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering*, pages 113–120, 2004.

[Hea01] C.G. Healey. Combining perception and impressionist techniques for non-photorealistic visualization of multidimensional data. In *SIGGRAPH 2001 Course 32*, chapter Nonphotorealistic Rendering in Scientific Visualization, pages 20–52. ACM, 2001.

[Hen03] J.M. Henderson. Human gaze control during real-world scene perception. *Trends in Cognitive Sciences*, 7(11):498–504, November 2003.

[Her98] A. Hertzmann. Painterly rendering with curved brush strokes of multiple sizes. In *ACM SIGGRAPH 1998: Proceedings of the 25th annual conference on Computer graphics and interactive techniques*, pages 453–460. ACM Press / ACM SIGGRAPH, 1998.

[Her99] A. Hertzmann. Introduction to 3d non-photorealistic rendering: Silhouettes and outlines. In *SIGGRAPH 1999 Course Notes*, chapter Non-Photorealistic Rendering, pages 205–242. ACM, 1999.

[HMBG00] H. Hauser, L. Mroz, G. Bischi, and E. Gröller. Two-level volume rendering-fusing mip and dvr. In *VISUALIZATION '00: Proceedings of the 11th IEEE Visualization 2000 Conference (VIS 2000)*, pages 211–218, 2000.

[HMH+03a] N. Halper, M. Mellin, C.S. Herrmann, V. Linneweber, and T. Strothotte. Psychology and non-photorealistic rendering: The beginning of a beautiful relationship. In *Mensch & Computer 2003, Interaktion in Bewegung*, pages 277–286, 2003.

[HMH+03b] N. Halper, M. Mellin, C.S. Herrmann, V. Linneweber, and T. Strothotte. Towards an understanding of the psychology of non-photorealistic rendering. *Comp. Visualistics, Media Informatics, and Virtual Communities*, 11:67–78, 2003.

[HO03] J. Hamill and C. O'Sullivan. Virtual dublin - a framework for real-time urban simulation. *Journal of WSCG: The 11th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, pages 221–225, 2003.

[HOC04] P. Hall, M. Owen, and J. Collomosse. A trainable low-level feature detector. In *ICPR '04: Proceedings of the Pattern Recognition, 17th International Conference on (ICPR'04) Volume 1*, pages 708–711, 2004.

[HP00] A. Hertzmann and K. Perlin. Painterly rendering for video and interaction. In *NPAR '00: Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*, pages 7–12, 2000.

[HS04] M. Haller and D. Sperl. Real-time painterly rendering for MR applications. In *GRAPHITE '04: Proceedings of the 2nd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 30–38, 2004.

[HSS+05] M. Hadwiger, C. Sigg, H. Scharsach, K. Bhler, and M. Gross. Real-time ray-casting and advanced shading of discrete isosurfaces. *Eurographics '05: Proceedings of Eurographics, Computer Graphics Forum*, 24(3):303–312, 2005.

[HTER04] C.G. Healey, L. Tateosian, J.T. Enns, and M. Remple. Perceptually based brush strokes for nonphotorealistic visualization. *ACM Transactions on Graphics*, 23(1):64–96, 2004.

[HZ00] A. Hertzmann and D. Zorin. Illustrating smooth surfaces. In *SIGGRAPH 2000: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 517–526. ACM Press / ACM SIGGRAPH, 2000.

[IFH+03] T. Isenberg, B. Freudenberg, N. Halper, S. Schlechtweg, and T. Strothotte. A developer's guide to silhouette algorithms for polygonal models. *IEEE Computer Graphics and Applications*, 23(4):28–37, July/August 2003.

[IHS02] T. Isenberg, N. Halper, and T. Strothotte. Stylizing silhouettes at interactive rates: From silhouette edges to silhouette strokes. In *Eurographics '02: Proceedings of Eurographics, Computer Graphics Forum*, volume 21(3), pages 249–258, 2002.

[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*, 20(11):1254–1259, 1998.

[INC+06] T. Isenberg, P. Neumann, S. Carpendale, M.C. Sousa, and J.A. Jorge. Non-photorealistic rendering in context: an observational study. In *NPAR '06:*

Proceedings of the 4th international symposium on Non-photorealistic animation and rendering, pages 115–126, 2006.

[Int00] V. Interrante. Harnessing natural textures for multivariate visualization. *IEEE Computer Graphics and Applications*, 20(6):6–11, 2000.

[iS99] id Software. Quake iii. CD-ROM, 1999.

[JAL+03] C.D. Jackson, D. Acevedo, D.H. Laidlaw, F. Drury, E. Vote, and D. Keefe. Designer-critiqued comparison of 2d vector visualization methods: a pilot study. In *SIGGRAPH 2003: ACM SIGGRAPH 2003 Sketches & Applications*, page 1, 2003.

[JDA07] T. Judd, F. Durand, and E. Adelson. Apparent ridges for line drawing. *ACM Transactions on Graphics*, 26(3):19, 2007.

[KB01] T. Kadir and M. Brady. Saliency, scale and image description. *International Journal of Computer Vision*, 45(2):83–105, 2001.

[KDK92] J.J. Koenderink, A.J. Van Doorn, and A.M.L Kappers. Surface perception in pictures. *Perception and Psychophysics*, 52(5):487, 1992.

[KDMF03] R.D. Kalnins, P.L. Davidson, L. Markosian, and A. Finkelstein. Coherent stylized silhouettes. *ACM Transactions on Graphics*, 22(3):856–861, 2003.

[KGC00] M. Kaplan, B. Gooch, and E. Cohen. Interactive artistic rendering. In *NPAR '00: Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*, pages 67–74, 2000.

[KHEK76] M. Kuwahara, K. Hachimura, S. Eiho, and M. Kinoshita. Processing of riangiocardiographic images. In *Digital Processing of Biomedical Images*, pages 187–203. Plenum Press, 1976.

[KHSI04] S. Kim, H. Hagh-Shenas, and V. Interrante. Conveying shape with texture: Experimental investigations of texture's effects on shape categorization judgments. *IEEE Transactions on Visualization and Computer Graphics*, 10:471–483, 2004.

[KKD09] J.E. Kyprianidis, H. Kang, and J. Döllner. Image and video abstraction by anisotropic kuwahara filtering. *Computer Graphics Forum - Special Issue on Pacific Graphics*, 28(7):1955–1963, 2009.

[KLK+00] A.W. Klein, W. Li, M.M. Kazhdan, W.T. Corrêa, A. Finkelstein, and T.A. Funkhouser. Non-photorealistic virtual environments. In *SIGGRAPH 2000: Proceedings of the 27th annual conference on Computer graphics and interactive techniques*, pages 527–534. ACM Press / ACM SIGGRAPH, 2000.

[KMH01] R. Kosara, S. Miksch, and H. Hauser. Semantic depth of field. In *INFOVIS '01: Proceedings of the IEEE Symposium on Information Visualization 2001 (INFOVIS'01)*, page 97, 2001.

[KMH+02] R. Kosara, S. Miksch, H. Hauser, J. Schrammel, V. Giller, and M. Tscheligi. Useful properties of semantic depth of field for better F+C visualization. In *VISSYM '02: Proceedings of the symposium on Data Visualisation 2002*, pages 205–210, 2002.

[KML99] R.M. Kirby, H. Marmanis, and D.H. Laidlaw. Visualizing multivalued data from 2d incompressible flows using concepts from painting. In *VIS '99: Proceedings of the conference on Visualization '99*, pages 333–340, 1999.

[KMM+02] R.D. Kalnins, L. Markosian, B.J. Meier, M.A. Kowalski, J.C. Lee, P.L. Davidson, M. Webb, J. F. Hughes, and A. Finkelstein. Wysiwyg npr: drawing strokes directly on 3d models. In *SIGGRAPH 2002: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 755–762. ACM Press / ACM SIGGRAPH, 2002.

[KP02] J. Kim and F. Pellacini. Jigsaw image mosaics. In *SIGGRAPH 2002: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, pages 657–664. ACM Press / ACM SIGGRAPH, 2002.

[KSFC02] A.W. Klein, P.J. Sloan, A. Finkelstein, and M.F. Cohen. Stylized video cubes. In *SCA '02: Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, pages 15–22, 2002.

[KSG03] D. Kirsanov, P.V. Sander, and S.J. Gortler. Simple silhouettes for complex surfaces. In *SGP '03: Proceedings of the 2003 Eurographics/ACM SIGGRAPH symposium on Geometry processing*, pages 102–106, 2003.

[KSKD10] J.E. Kyprianidis, A. Semmo, H. Kang, and J. Döllner. Anisotropic kuwahara filtering with polynomial weighting functions. In *TPCG 10': Proceedings of EG UK Theory and Practice of Computer Graphics*, 2010.

[KSW06] J. Kruger, J. Schneider, and R. Westermann. Clearview: An interactive context preserving hotspot visualization technique. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):941–948, 2006.

[KV06a] Y. Kim and A. Varshney. Saliency-guided enhancement for volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):925–932, 2006.

[KV06b] Y. Kim and A. Varshney. Saliency-guided enhancement for volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):925–932, 2006.

[KWH06] A. Kolliopoulos, J.M. Wang, and A. Hertzmann. Segmentation-based 3d artistic rendering. In *EGSR '06: Eurographics Symposium on Rendering 2006*, pages 361–370, 2006.

[LFX+05] J.P. Lewis, N. Fong, X. XueXiang, S.H. Soon, and T. Feng. More optimal strokes for NPR sketching. In *GRAPHITE '05: Proceedings of the 3rd international conference on Computer graphics and interactive techniques in Australasia and South East Asia*, pages 47–50, 2005.

[Lit97] P. Litwinowicz. Processing images and video for an impressionist effect. In *SIGGRAPH 1997: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 407–414. ACM Press / ACM SIGGRAPH, 1997.

[LM02] E.B. Lum and K-L. Ma. Hardware-accelerated parallel non-photorealistic volume rendering. In *NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering*, pages 67–ff, 2002.

[LMHB00] A. Lake, C. Marshall, M. Harris, and M. Blackstein. Stylized rendering techniques for scalable real-time 3d animation. In *NPAR '00: Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*, pages 13–20, 2000.

[LRA+07] W. Li, L. Ritter, M. Agrawala, B. Curless, and D. Salesin. Interactive cutaway illustrations of complex 3d models. *ACM Transactions on Graphics*, 26(3):31, 2007.

[LSF10] J. Lu, P.V. Sander, and A. Finkelstein. Interactive painterly stylization of images, videos and 3D animations. In *I3D '10: Proceedings of the ACM*

SIGGRAPH Symposium on Interactive 3D Graphics and Games, pages 127–134, 2010.

[MBC02] J.L. Mitchell, C. Brennan, and D. Card. Real-time image-space outlining for non-photorealistic rendering. In *SIGGRAPH 2002, Conference Abstracts and Applications, ACM Press*, page 239, 2002.

[MBG08a] A. McNamara, R. Bailey, and C. Grimm. Improving search task performance using subtle gaze direction. In *APGV '08: Proceedings of the 5th symposium on Applied perception in graphics and visualization*, pages 51–56, 2008.

[MBG08b] A. McNamara, R. Bailey, and C. Grimm. Improving search task performance using subtle gaze direction. In *APGV '08: Proceedings of the 5th symposium on Applied perception in graphics and visualization*, pages 51–56, 2008.

[Mei96] B.J. Meier. Painterly rendering for animation. In *ACM SIGGRAPH 1996: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 477–484. ACM Press / ACM SIGGRAPH, 1996.

[MENY06] S. Magy M.S. El-Nasr and S. Yan. Visual attention in 3d video games. In *ACE '06: Proceedings of the 2006 ACM SIGCHI international conference on Advances in computer entertainment technology*, page 22, 2006.

[MG01] A. Mohr and M. Gleicher. Non-invasive, interactive, stylized rendering. In *I3D '01: Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 175–178, 2001.

[MG02] A. Majumder and M. Gopi. Hardware accelerated real time charcoal rendering. In *NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering*, pages 59–66, 2002.

[MH80] D. Marr and E. Hildreth. Theory of edge detection. *Royal Society of London Proceedings Series B*, 207:187–217, February 1980.

[MKG+97] L. Markosian, M.A. Kowalski, D. Goldstein, S.J. Trychin, J.F. Hughes, and L.D. Bourdev. Real-time nonphotorealistic rendering. In *ACM SIGGRAPH 1997: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 415–420. ACM Press / ACM SIGGRAPH, 1997.

[MLD+08] R. McDonnell, M. Larkin, S. Dobbyn, S. Collins, and C. O'Sullivan. Clone attack! perception of crowd variety. *ACM Transactions on Graphics*, 27:1–8, 2008.

[MLH+09] R. McDonnell, M. Larkin, B. Hernandez, I. Rudomin, and C. O'Sullivan. Eye-catching crowds: Saliency based selective variation. *ACM Transactions on Graphics*, 28:1–10, 2009.

[Mou03] D. Mould. A stained glass image filter. In *EGRW '03: Proceedings of the 14th Eurographics workshop on Rendering*, pages 20–25, 2003.

[MR98] A. Mack and I. Rock. *Inattentional blindness.* MIT Press, Cambridge, MA, USA, 1998.

[MRH08] J. Mensmann, T. Ropinski, and K.H. Hinrichs. Interactive cutting operations for generating anatomical illustrations from volumetric data sets. *Journal of WSCG: The 16th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*, pages 86–96, 2008.

[MSMH09] J. Meyer-Spradow, T. Ropinskiand J. Mensmann, and K.H. Hinrichs. Voreen: A rapid-prototyping environment for ray-casting-based volume visualizations. *IEEE Computer Graphics and Applications*, 29(6):6–13, 2009.

[ND03] M. Nienhaus and J. Doellner. Edge-enhancement - an algorithm for real-time non-photorealistic rendering. *Journal of WSCG: The 11th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision' 2003*, pages 346–353, 2003.

[ND04a] M. Nienhaus and J. Döllner. Blueprints: illustrating architecture and technical parts using hardware-accelerated non-photorealistic rendering. In *GI '04: Proceedings of Graphics Interface 2004*, pages 49–56, 2004.

[ND04b] M. Nienhaus and J. Döllner. Sketchy drawings. In *AFRIGRAPH '04: Proceedings of the 3rd international conference on Computer graphics, virtual reality, visualisation and interaction in Africa*, pages 73–81, 2004.

[Nin02] Nintendo. The legend of zelda: The wind waker. CD-ROM, 2002.

[Nin06] Nintendo. Okami. CD-ROM, 2006.

[Nin07] Nintendo. The legend of zelda: Phantom hourglass. Nintendo DS game card, 2007.

[NLB+07] M. Nusseck, J. Lagarde, B. Bardy, R. Fleming, and H.H. Bülthoff. Perception and prediction of simple object interactions. In *APGV '07: Proceedings of* *the 4th symposium on Applied perception in graphics and visualization*, pages 27–34, 2007.

[NM00] J.D. Northrup and L. Markosian. Artistic silhouettes: a hybrid approach. In *NPAR '00: Proceedings of the 1st international symposium on Non-photorealistic animation and rendering*, pages 31–37, 2000.

[NRTT95] H. Noser, O. Renault, D. Thalmann, and N.M. Thalmann. Navigation for digital actors based on synthetic vision, memory, and learning. *Computers and Graphics*, 19(1):7–19, 1995.

[NSW02] Z. Nagy, J. Schneider, and R. Westermann. Interactive volume illustration, 2002.

[OD01] C. O'Sullivan and J. Dingliana. Collisions and perception. In *ACM SIGGRAPH 2001: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 151–168. ACM Press / ACM SIGGRAPH, 2001.

[Ost99] V. Ostromoukhov. Digital facial engraving. In *ACM SIGGRAPH 1999: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 417–424. ACM Press / ACM SIGGRAPH, 1999.

[PB07] B. Preim and D. Bartz. *Visualization in Medicine: Theory, Algorithms, and Applications (The Morgan Kaufmann Series in Computer Graphics)*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2007.

[PHWF01] E. Praun, H. Hoppe, M. Webband, and A. Finkelstein. Real-time hatching. In *SIGGRAPH '01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, page 581, 2001.

[Pic01] Fox Searchlight Pictures. Waking life. Motion Picture, 2001.

[Pic06] Warner Independent Pictures. A scanner darkly. Motion Picture, 2006.

[Pic07] Warner Bros. Pictures. 300. Motion Picture, 2007.

[Pom04] G. Pomaska. Between photo-realism and non-photorealistic rednering - modeling urban areas for real time VR. In *International Workshop on Vision Techniques applied to the Rehabilitation of City Centres*, 2004.

[PPC07] G. Papari, N. Petkov, and P. Campisi. Artistic edge and corner enhancing smoothing. *Image Processing, IEEE Transactions on*, 16(10):2449–2462, October 2007.

[PRI+09] L. Phillips, B. Ries, V. Interrante, M. Kaeding, and L. Anderson. Distance perception in NPR immersive virtual environments, revisited. In *APGV '09: Proceedings of the 6th Symposium on Applied Perception in Graphics and Visualization*, pages 11–14, 2009.

[PTD05] B. Preim, C. Tietjen, and C. Dorge. Npr, focussing and emphasis in medical visualizations. In *SimVis '05: In Proceeedings of Simulation and Visualisierung*, pages 139–152, 2005.

[PY08] Y. Park and K. Yoon. Painterly animation using motion maps. *Graphical Models*, 70(1-2):1–15, 2008.

[RBD06] S. Rusinkiewicz, M. Burns, and D. DeCarlo. Exaggerated shading for depicting shape and detail. *ACM Transactions on Graphics*, 25(3):1199–1205, 2006.

[RD03] M. Roussou and G. Drettakis. Photorealism and non-photorealism in virtual heritage representation. In A.Chalmers, D.Arnold, and F. Niccolucci, editors, *Proceedings of the International Symposium on Virtual Reality, Archeology and Cultural Heritage*, November 2003.

[RGW+03] S. Roettge, S. Guthe, D. Weiskopf, T. Ertl, and W. Strasser. Smart hardware-accelerated volume rendering. In *VISSYM '03: Proceedings of the symposium on Data visualisation 2003*, pages 231–238, 2003.

[SB09] C. Sauvaget and V. Boyer. Abstraction of photographs: a comics style approach. In *ECC'09: Proceedings of the 3rd international conference on European computing conference*, pages 215–220, 2009.

[SCCD04] V. Sundstedt, A. Chalmers, K. Cater, and K. Debattista. Top-down visual attention for efficient rendering of task related scenes. In *VMV '04: In Proceedings of the 9th annual workshop on Vision, Modeling and Visualization*, pages 209–216, 2004.

[SD02] A. Santella and D. DeCarlo. Abstracted painterly renderings using eye-tracking data. In *NPAR '02: Proceedings of the 2nd international symposium on Non-photorealistic animation and rendering*, pages 75–ff, 2002.

[SD04] A. Santella and D. DeCarlo. Visual interest and npr: an evaluation and manifesto. In *NPAR '04: Proceedings of the 3rd international symposium on Non-photorealistic animation and rendering*, pages 71–150, 2004.

[SEA09] N. A. Svakhine, D. S. Ebert, and W. M. Andrews. Illustration-inspired depth enhanced volumetric medical visualization. *IEEE Transactions on Visualization and Computer Graphics*, 15(1):77–86, 2009.

[Seg08] Sega. Valkyria chronicles. Blue-Ray Disc, 2008.

[SLNG07] V. Setlur, T. Lechner, M. Nienhaus, and B. Gooch. Retargeting images and video for preserving information saliency. *IEEE Computer Graphics and Applications*, 27(5):80–88, 2007.

[Sof09] Gearbox Software. Borderlands. CD-ROM, 2009.

[SS02] T. Strothotte and S. Schlechtweg. *Non Photorealistic Computer Graphics: Modeling, Rendering and Animation.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2002.

[SSLR96] J. Schumann, T. Strothotte, S. Laser, and A. Raab. Assessing the effect of non-photorealistic rendered images in cad. In *CHI '96: Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 35–41, 1996.

[SSWR08] V. Sundstedt, E. Stavrakis, M. Wimmer, and E. Reinhard. A psychophysical study of fixation behavior in a computer game. In *APGV '08: Proceedings of the 5th symposium on Applied perception in graphics and visualization*, pages 43–50, 2008.

[ST90] T. Saito and T. Takahashi. Comprehensible rendering of 3-d shapes. *Computer Graphics (Proceedings of SIGGRAPH '90)*, 24(4):197–206, 1990.

[TC00] S.M.F. Treavett and M. Chen. Pen-and-ink rendering in volume visualisation. In *VIS '00: Proceedings of the conference on Visualization '00*, pages 203–210, 2000.

[THE07] L.G. Tateosian, C.G. Healey, and J.T. Enns. Engaging viewers through non-photorealistic visualizations. In *NPAR '07: Proceedings of the 5th international symposium on Non-photorealistic animation and rendering*, pages 93–102, 2007.

[Tre03] S. Treue. Visual attention: the where, what, how and why of saliency. *Current Opinion in Neurobiology*, 13(4):428–432, 2003.

[TSH98] U. Tiede, T. Schiemann, and K.H. Höhne. High quality rendering of attributed volume data. In *VIS '98: Proceedings of the conference on Visualization '98*, pages 255–262, 1998.

[TWG+04] W.B. Thompson, P. Willemsen, A.A. Gooch., S.H. Creem-Regehr, J.M. Loomis, and A.C. Beall. Does the quality of the computer graphics matter when judging distances in visually immersive environments. *Presence: Teleoper. Virtual Environ.*, 13(5):560–571, 2004.

[Ubi03] Ubisoft. Xiii. CD-ROM, 2003.

[Ubi08] Ubisoft. Prince of persia. Blue-Ray Disc, 2008.

[VB99] O. Veryovka and J. Buchanan. Comprehensive halftoning of 3D scenes. In *Eurographics '99: Proceedings of Eurographics, Computer Graphics Forum*, volume 18(3), pages 13–22, 1999.

[VKG05] I. Viola, A. Kanitsar, and M.E. Gröller. Importance-driven feature enhancement in volume visualization. *IEEE Transactions on Visualization and Computer Graphics*, 11(4):408–418, 2005.

[War04] C. Ware. *Information Visualization: Perception for Design.* Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 2004.

[WBC+07] C. Wallraven, H.H. Bülthoff, D.W. Cunningham, J. Fischer, and D. Bartz. Evaluation of real-world and computer-generated stylized facial expressions. *ACM Transactions on Applied Perception*, 4(3):16, 2007.

[WOG06] H. Winnemoller, S. C. Olsen, and B. Gooch. Real-time video abstraction. *ACM Transactions on Graphics*, 25(3):1221–1226, 2006.

[WS94] G. Winkenbach and D.H. Salesin. Computer-generated pen-and-ink illustration. In *ACM SIGGRAPH 1994: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 91–100. ACM Press / ACM SIGGRAPH, 1994.

[WS96] G. Winkenbach and D.H. Salesin. Rendering parametric surfaces in pen and ink. In *ACM SIGGRAPH 1996: Proceedings of the 23rd annual conference* *on Computer graphics and interactive techniques*, pages 469–476. ACM Press / ACM SIGGRAPH, 1996.