Non Photorealistic Techniques with Focus and Context Volume Rendering

Peter Benilov

Supervisor: John Dingliana

School of Computer Science and Statistics

Trinity College Dublin

M.A.I. Computer Engineering

21/5/2015

Declaration

I, the undersigned, delcare 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.

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

Peter Andrew Benilov

May 21, 2015

Summary

This dissertation explores the use of non-photorealistic rendering techniques in focus and context volume rendering. The overall goal of this project is to develop and evaluate a rendering style which uses NPR techniques to effectively render focus and context images. This is done by developing an interactive volume rendering framework. The pipeline of the framework is designed in a modular fashion, making it flexible and allowing for a robust exploration of the use of NPR in focus and context visualisation.

The framework that I implemented uses raycasting to extract information from volume data. It then uses different combinations of colour abstraction, texture abstraction and line extraction to render the focus and context regions respectively.

For colour abstraction, I developed a novel tone shader which can interactively switch between two-tone shading and smooth tone shading (meaning a gradual shift from light to dark).

Texture abstraction is achieved through the use of an edge preserving blurring filter called a Kuwahara filter. An image space Sobel filter is used to extract silhouettes and contours.

The implementation of theses techniques was evaluated and discussed, and an effective NPR rendering style was identified. According to this style, the focus region is rendered using smooth tone shading and is enhanced with image space lines. The context region is rendered using two-tone shading and is then passed through the Kuwahara filter for texture abstraction.

To combine the focus and context regions, I developed novel implementations of popular focus and context paradigms, which are the Magic Lens and the 3D Cutaway Lens respectively. It was found that my 3D cutaway lens is more effective at focus and context visualisation than my Magic Lens, as it provides a stronger depth relationship cue. The NPR style mentioned in the previous paragraph was combined with the 3D cutaway lens interface to create the final focus and context rendering style which was evaluated.

The evaluation of the the focus and context style consisted of a visual examinations of the output images rendered in this style. Following this, an automatic

salience metric called 'Saliencey Toobox' was used to test the output images to see if they meet the requirements set out in the design chapter of this dissertation. Finally, the framework is evaluated according to its frame rate when rendering with certain style features switched on/off.

Overall, the focus and context style which was developed held up well in the evaluation. It met the requirements of increasing salience and details in the focus region while abstracting and removing high frequency detail in the context region. The framework also met its requirements of being flexible and interactive, as it maintained framerates above 8fps.

Acknowledgements

I would like to thank my dissertation supervisor, John Dingliana, for his availability, assistance and guidance throughout the implementation and write-up of this dissertation. I would also like to thank my parents for their support and patience, and all my friends and colleagues for their comradery.

Abstract

Volume rendering is a challenging problem due to the difficulty of rendering such complex information in a manner which is easy for a user to understand. Focus and context volume visualisation attempts to solve this issue by allowing the user to specify a focus area in the data which is displayed with a high level of detail while the surrounding data is sufficiently abstracted to retain context.

Non-photorealistic rendering techniques are regarded as powerful tools in manipulating view perception. These techniques are often used abstract a scene by removing unnecessary detail, while highlighting important information. In fields such as medicine and science, NPR has been used by artists in the production of technical illustrations. For this reason, the field of volume rendering (VR) is heavily invested in NPR techniques, especially for visualisation of medical data sets.

This dissertation explores the use on NPR techniques in focus and context volume rendering. It describes the implementation of a volume rendering framework which renders compelling focus and context images with the use of colour abstraction, texture abstraction and silhouette extraction. Two focus and context interfaces are implemented, which combine focus and context regions. An effective style for focus and context rendering is identified and evaluated visually and with the use of an automatic salience metric.

Contents

List of Figures

Chapter 1

Introduction

This dissertation explores the use of non-photorealistic rendering (NPR) techniques in focus and context volume rendering. This is done by developing an interactive volume rendering framework which uses NPR techniques to enhance the focus region and abstract the context region. Using this framework, an effective focus and context style is created and evaluated. This chapter describes the motivations behind this dissertations, and gives an in-depth introduction to relevant fields.

The majority of research in the field of real-time computer graphics has always been focused on photorealism. As the gap between current state-of-the-art and real-time photorealism is narrowing, and the remaining gap is largely due to hardware constraints, a greater amount of research is being dedicated towards rendering simplified, stylistic scenes with an illustrated look. The field of non-photorealistic rendering (NPR) is concerned with depicting information using techniques inspired by artistic media, such a paintings and animated cartoons. Because NPR techniques are not bound by reality, they are considered a key tool in developing unique visual identities for animated movies and games. Some of the oldest and most famous NPR techniques are repeatedly used in the entertainment industry, namely cel-shading, enhanced silhouettes, texture abstraction, water colouring and other painterly techniques.

NPR research attracts attention from many creative and technical areas since it provides great expressive capabilities to its user. The strength of NPR lies in its ability to convey complex scenes in a computationally cheap and visually uncluttered manner, whilst still presenting the same level of relevant information as photorealistic techniques. NPR techniques abstract a scene by removing unnecessary detail, while important information is highlighted. In fields such as medicine and science, NPR has been used by artists in the production of technical illustrations. This is especially true in medical imagery. For this reason, the field of volume rendering (VR) is heavily invested in NPR techniques, especially for visualisation of medical data sets.

In computer graphics, volume rendering (VR) is a set of techniques used to display a three-dimensional data set (e.g., from an MRI scan) onto a two-dimensional projection (e.g., computer screen). Volume data sets contain a lot of information which cannot be shown and understood at once. For this reason, a primary problem in VR is interpreting sampled volume data and displaying it in a way that would expose certain properties of the information. With the advancement of technology, the resolution of volume data sets has reached a level where VR has attracted the attention of the medical industry.

Volume rendering of Computed Topography (CT) and Magnetic Resonance Imaging (MRI) datasets dominates in application areas such as diagnosis and surgery planning. One of the biggest advantages of VR of medical data is that it provides all necessary information in a single radiologic study in cases that would previously require several studies. VR generates clinically accurate and immediately available images from a full CT or MRI dataset without extensive editing. Furthermore, it allows medical professionals to interactively explore the data set. The sub field of VR in medicine is growing rapidly, with untapped potential in the form of interactive e-learnings for students and further growth in the professional field. VR of medical data sets is becoming a relevant and profitable area of research.

Recent developments in image modalities, such as Multislice-Spiral CT in medical imaging, have led to a substantial increase in resolution in slice direction. Data sets with 300 or more slices will be the standard in the next years. On the other hand, the structure of interest (tumours, arterial structures etc.) often occupy less than 10% of the voxels in a volume data set. In practical applications, the large number of slices per study requires fast, versatile and efficient methods for reducing the data for information extraction. All these require an alternative rendering approach which integrates both: on the one hand, focusing on a specific structure and, on the other hand, context information.

My proposal concentrates on a particular visualisation model called focus and context rendering. This model splits an image into two areas: the focus area and the context area. The focus area is detailed and displays relevant information. The context area is abstracted to hide high frequency detail, whilst preserving enough information to provide spatial context to the focus. This focus and context paradigm is an effective method of solving the issue of information overloading in volume rendering.

This project aims to explore the use of state-of-the-art NPR techniques in focus and context visualisation of volume data. A flexible, real-time framework is implemented and used to develop an effective NPR style for focus and context VR. This style is then evaluated visually and with an automatic salience toolkit, to determine the effectiveness of the proposed NPR style.

I believe that the core components of an NPR focus and context style are the following:

- Colour Abstraction: the careful selection of colour and shading style to better conveying surface shape and curvature.
- Texture Abstraction: the reduction of high frequency noise and detail.
- Silhouettes and Outlines: the use of boundary lines to convey surface shape and orientation.
- Focus and Context Interface: the mechanism by which the focus and context regions are visualised relative to each other.

In most technical illustrations, shape information is valued above precise reflectance information, so hue changes can be used to indicate surface orientation rather than reflectance. Examples of colour abstraction include Gooch shading and Toon shading. Both of these shading models are implemented by mapping a Lambertian term to a 1D texture. Gooch shading obeys a gradual warm-to-cool hue shift. Toon shading is characterized by a very small set of colours with a hard transition at the terminator. For the purpose of this project, I have developed a flexible tone shading algorithm which can computer a Toon or Gooch texture on the fly, according to a user defined input parameter. Using this shading technique, the user can transition from Toon style shading to smooth tone (Gooch) shading (or somewhere in between) in real time.

Texture abstraction flattens shading in a way that resembles drawings, painting and comics. By omitting the high frequency details, more attention is drawn to the important information in the rendering and avoids information overloading. Blurring and smoothing features can achieve this, however, edges must be preserved as they hold critical information about object shape. In this dissertation, I describe the implementation of an edge preserving smoothing filter from state of the art, and outline its use in a focus and context VR framework.

Illustrators frequently employ silhouettes contours to enhance the depiction of objects. Contours help to clearly delineate object shape and resolve ambiguities due to occlusion by emphasizing the transition between front-facing and back-facing surface locations. In this dissertation, I outline the use of a state-of-the-art image space edge detector in highlighting or darkening contours.

Focus and context visualisation is often used by illustrators to convey all important details without overloading the image with information. It is important that the interface between focus and context areas facilitates the clear depiction of the focus area, preserves enough of the context area to keep it relevant and informative, and conveys the depth relationship between the two areas. In this dissertation, I present novel implementations of two established focus and context interfaces, which are the 'Magic Lens' and the '3D Cutaway Lens'. The Magic Lens modifies the opacity of obstructing layers in order to enhance data of interest and suppress distracting information. It facilitates smooth transitions between focus and context areas. The 3D Cutaway Lens uses a 3D clipping frame to completely expose a focus region. It provides strong depth relationship cues between the focus and context areas.

1.1 Dissertation Layout

This dissertation is structured as follows:

Chapter 2 (Background and Related Work) presents a general introduction to VR and NPR, as well as an exploration of various techniques in these fields. This chapter also contains reviews of state-of-the-art research in the areas of VR and NPR.

Chapter 3 (Design) gives an overview of the goals and requirements of the work described in this dissertation. A rough implementation approach is also detailed here.

Chapter 4 (Implementation) gives an in depth description of how the framework was put together.

Chapter 5 (Evaluation) reviews and analyses the implemented framework. The evaluation is based on visual examinations. Success is measured by whether or not I have achieved the goals set out in chapter 3, and how well the framework fulfils the requirements described.

Chapter 6 (Conclusion) reviews the proposal, and discusses the success of the project. Future work and improvements are proposed, which bridge the gap between the implemented framework and an ideal framework which satisfies the proposal objectives completely.

Chapter 2

Background and Related Work

This chapter introduces the theoretical background the reader should be acquainted with before moving onto the design and implementation sections. The research which is relevant to and inspired this dissertation is also discussed. This chapter is split into three sections:

- Volume Rendering: This section introduces the techniques used to display VR data. Both isosurface and direct volume rendering techniques are discussed.
- Non-Photorealistic Rendering: Research done in the field of NPR is introduced, especially work which applies to VR
- Focus and Context Rendering: Focus and Context frameworks are discussed, particularly those which work with VR.

2.1 Volume Rendering

Volume rendering is used to display a 2D image of a 3D dataset. The majority of datasets are discretely sampled along a 3D grid and contain scalar values usually acquired from medical imaging devices such as CT or MRI machines. The data then takes the form of a 3D array of voxels, a 3D extension of pixels. Volume rendering can be performed using two main techniques, either by extracting a number of surfaces from the data and rendering these surfaces to the screen, called isosurface rendering, or by rendering the volume itself as a complete block of data with no intermediary structures, usually called direct volume rendering (DVR). A wide variety of techniques have been developed in order to improve volume rendering performance, enhance perception of the data, and increase the rendering speed.

2.1.1 Splatting

Splatting, proposed by Westover in 1990 [1] , is a direct VR technique which was used in the early days of VR. Splatting projects voxels onto the 2D viewing plane. It approximates this projection by a so-called Gaussian splat, which depends on the opacity and on the colour of the voxel (other splat types, like linear splats can be used also). A projection is made for every voxel and the resulting splats are composited on top of each other in back-to-front order to form the final image. These techniques benefit from improved speed of calculation over other VR techniques such as ray casting. However, volume splatting is no longer widely utilised as modern hardware limitations comfortably allow for more advanced VR methods.

2.1.2 Isosurface Extraction

Isosurface volume rendering techniques (classed as indirect VR) were introduced by Lorensen and Cline [2]. They work by extracting a polygonal representation of a specific isosurface from a volume dataset. This polygon mesh is then rendered using traditional rasterisation techniques. A common indirect VR technique is the marching cubes algorithm, proposed by Lorensen and Cline and developed by Müller and Wehle [3].

2.1.3 Texture Slicing

The shear warp approach to volume rendering was developed by Cameron and Undrill, and popularized by Philippe Lacroute and Marc Levoy [4]. In this technique, the viewing transformation is transformed such that the nearest face of the volume becomes axis aligned with an off-screen image buffer with a fixed scale of voxels to pixels. The volume is then rendered into this buffer using the far more favorable memory alignment and fixed scaling and blending factors [5]. Once all slices of the volume have been rendered, the buffer is then warped into the desired orientation and scaled in the displayed image.

This technique is relatively fast in software at the cost of less accurate sampling and potentially worse image quality compared to ray casting. There is memory overhead for storing multiple copies of the volume, for the ability to have near axis aligned volumes.

With the development of programmable GPUs, researchers began to exploit hardware approaches in order to achieve significant increases in computational speed. This resulted in the development of texture based volume rendering [6]. This technique divides the volume into slices and then uses the graphics hardware to render slice polygons textured with volume data. These polygons are then rendered back to front and alpha blended to form the final image.

Initial implementations of texture slicing were limited to 2D textures and object aligned textures. This resulted in flickering when the slice axis changed. With the advent of hardware support for 3D textures, view-aligned slice based rendering became possible which provided superior image quality and removed many of the drawbacks inherent in a 2D texture based implementation.

2.1.4 Maximum Intensity Projection (MIP)

As opposed to direct volume rendering, which requires every sample value to be mapped to opacity and a colour, maximum intensity projection only picks out and projects the voxels with maximum intensity that fall in the way of parallel rays traced from the viewpoint to the plane of projection.

An improvement to MIP is local maximum intensity projection. This technique considers the first maximum value that is above a certain threshold, instead of the global maximum value. Because the ray is terminated earlier, this technique is faster and also gives somewhat better results as it approximates occlusion [7].

2.1.5 Ray Casting

Ray casting is currently the most popular approach for state-of-the-art VR methods [8], as it provides the same image quality as texture slicing methods but with a lot more flexibility. Modern techniques utilise programmable GPUs and 3D textures in order to produce high quality computationally efficient raycast renderings of volume data. Most ray casting techniques consist of two passes. In the first pass, the front and back faces of the volume bounding box are rendered to an on screen buffer in order to generate the start and end positions for each ray. In the second pass, for each pixel on the screen, a ray is cast into the volume using the start and end positions generated in the first pass of the algorithm then iterates along the ray, sampling the volume data and accumulating opacity and colour information until the ray terminates and the final result is output to the screen. Multiple optimisations to this approach have been proposed to further improve rendering speeds [9].

The graphics processing unit has become a powerful computational tool for parallelisable tasks such as ray casting. As a result, ray casting has become a viable and attractive alternative to slice-based methods. Sliced based methods often produce artefacts due to step size variation caused by perspective projection. These approaches are also not suitable for large datasets as the number and position of slices depends on the volume data set. In contrast, ray casting methods are simple and robust techniques which produce accurate, high quality results. For these reasons, ray casting is considered a future-proof alternative.

2.2 Non-Photorealistic Rendering

2.2.1 Colour and Shading

Traditional Volume Rendering (VR) has relied on the use of transfer functions to produce artificial views of the data to highlight regions of interest [10]. These transfer functions require in-depth knowledge of the data and need to be adjusted for each data set. Physics based opacity and illustrative opacity transfer functions can circumvent the need for in-depth knowledge, instead relying on a couple of user defined parameters to set voxel opacity [11][12]. Zhou et al [13] proposed the use of distance from the camera to modify opacity in the context region. Marchesin et al [14] eliminate opacity from the transfer function and instead modify volume opacity based on a function quantifying the relative importance of each voxel.

There has been extensive research for illustrating surface shape using non-photorealistic colouring techniques. Adopting a technique found in painting, Gooch et al. developed a tone-based illumination model that determined the hue and intensity from orientation of a surface element to a light source [15]. Lake et al. built upon the Gooch shading model by utilizing hardware accelerated texture mapping from a 1D texture to simulate the limited colour palette cartoonists use for painting cels [16]. Barla has extended this technique by using a 2D texture lookup to incorporate view-dependent and level-of-detail effects [17]. Baolong [18] describes methods for performing colour quantization on full color RGB images, using an octree data structure.

2.2.2 Texture Abstraction

Expressive textures have been applied to surfaces to convey surface shape. Lu creates textural abstraction by rendering brush strokes as point sprites with textures. By combining layers of strokes with different levels of coarseness, a painterly style is achieved [19].

A typical approach to automatically create stylized abstractions from images or videos is the use of an edge-preserving filter. Classical filters in this field are the well-known Kuwahara filter [20] and its variants [21][22]. The general idea behind these filters is to replace the input pixel with the mean of the subregion with minimum variance. The variants differ in the way the subregions are defined. The most advanced and effective Kuwahara adaption is the Anisotropic Filter [23] which removes clustering artefacts by adapting shape, scale and orientation of the filter to the local structure of the input figure 2.1.

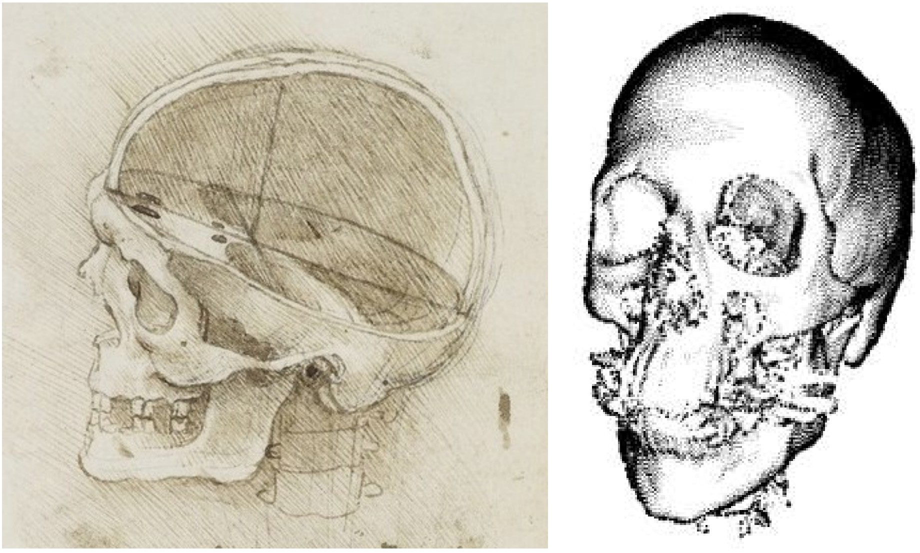

Figure 2.1: Da Vinci Hatching (Left) and Stippling (Right). Source: http://bmia.bmt.tue.nl/

Another approach for texture abstraction is the use of illustrative techniques which imitate how an artist would draw an illustration. One such technique is called stippling. This technique occurs often in pen drawings, where dots characterise the surface structure of an object. Stippling techniques are known to emphasise curved areas of the object surface, either by changing the density, or by changing the size of the dots. Manually stippled drawings consist of a large amount of uniformly scaled dots. Varying the density of the dots allows a change of tone in different areas of the drawing.

Another illustrative technique is called hatching. Hatching is also a pen drawing technique, however strokes are applied instead of dots. Hatching, as well as stippling, are illustrative techniques that occur quite often in technical, architectural and medical illustrations. The efficient and concise communication of information brought forward by these techniques was already known in the 15th century by Leonardo da Vinci, whose caricature is depicted in figure 2.1.

2.2.3 Silhouettes and Outlines

Suggestive contours and silhouettes are a common theme in volume illustration. There are two approaches to line extraction; image space lines and object space lines. There are numerous algorithms to extract lines from image space but some of the more common methods are: Difference of Gaussian edge detection, Sobel edge detection, and Canny edge detection [24].

Difference of Gaussian (DoG) edge detection works by applying a Gaussian blur to the target 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 over a user specified threshold. The blurred images are obtained by convolving the original images with Gaussian kernels having differing standard deviations. Sobel edge detection uses 3x3 kernels which are convolved with the original image to calculate approximations of pixel gradient. The Canny edge detector is uses a 5-stage algorithm to detect a wide range of edges in images. This method detects edges with a low error rate and minimizes false edges, but is computationally expensive.

Object space lines are extracted by considering the 3D nature of the dataset. This allows object space techniques to extract lines according to more advanced metrics such as surface normal. Object space line extraction is generally more complicated to implement and suffers an increased in computation time when compared to image space techniques. Csebfalvi et al. [25] extracted object space lines by using accelerated ray-casting and shear-warp rendering for fast object contour rendering. Erbert achieves volume illustration through the use of boundary enhancement and silhouettes [12]. Burns [26] proposes a volumetric drawing system that directly extracts sparse linear features, such as silhouettes and suggestive contours, using a temporally coherent seed-and-traverse framework. Kindlmann [27] proposed to regulate contours based on the normal curvature along the view direction. Bruckner [11] et all introduces an approximation of this measure which is less accurate but adds little additional computation to standard ray-cast based volume renderers.

2.3 Focus and Context Rendering

2.3.1 Distortion Lenses

LaMar et al. [28] propose a technique for focus and context volume rendering which utilises a magnifying lens to allow the user to zoom in on selected areas of the volume while keeping the surrounding area at the standard magnification level.

Ikits and Hansen [29] propose using a lens tied to the mouse position to allow the user to eliminate the context rendering and display the focus region without obscurance.

2.3.2 Two-Level Rendering

Two-level volume rendering, which was proposed by Hauser et al [30], allows for selective use of different rendering techniques for different subsets of a 3D data set. Different structures within the data set are rendered by either direct volume rendering (DVR), maximum-intensity projection (MIP), surface rendering or non-photorealistic rendering (NPR). DVR is used to render the focus area, and a transparent techniques such as MIP is used to render the focus. 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.

2.3.3 Cutaway Views

Mcguffin et al. [31] propose the use of exploded views for exploring data. Unfortunately, their technique relies on a point based rendering system which results in very poor quality renderings and is unable to support transparent voxels.

Chen et al. [32] introduce the concept of spatial transfer functions, which use geometric transforms for volume data to render exploded views for ray cast based volume rendering. Correa et al [33] extend this work to allow their splitting techniques to follow surface and segment alignment. Islam et al [34] further extend this work by combining spatial transfer functions with volume splitting.

2.3.4 Combined Methods

Tietjen et al. [35] combine surface rendering, DVR, and line drawings to create a combined volume rendering system for surgery planning. Their technique produces visually appealing results but requires pre-segmented datasets and an object space line extraction algorithm.

Bruckner et al. [36] combine a number of metrics including the current accumulated opacity, gradient magnitude, and distance to the camera in order to produce a context-preserving volume rendering system which does not require pre-segmented data. In other work [37], they suggest a combined system for focus and context rendering.

Bruckner also introduced the concept of style transfer functions [11]. His approach enables flexible data-driven illumination which goes beyond using the transfer function to just assign colors and opacities. An image-based lighting model uses sphere maps to represent non-photorealistic rendering styles. Style transfer functions allow us to combine a multitude of different shading styles in a single rendering. We extend this concept with a technique for curvature controlled style contours and an illustrative transparency model.

Krger et al [38] present a focus and context visualisation framework called Clearview, which assigns varying levels of opacity across a pre-segmented volume and renders the volume adaptively using volume ray casting. Their technique takes into account a user specified focus region and modifies the transparency of the context region according to absolute distance, view distance or surface curvature to produce a number of different renderings. Their technique can be utilised with both an isosurface and a DVR context layer.

Early attempts at focus and context VR required pre-segmented datasets, which had to be classified manually in a time consuming pre-processing step. Most modern approaches have been developed in such a way that the segmentation is done automatically or a simple user guided interface. Despite the large amount of research in the field of focus and context, its effect on user perception is unclear. The renderings created by focus and context systems are visually compelling, but their effectiveness at improving visualisation has not be properly tested through user studies.

Chapter 3

Design

3.1 Goals

From literature [39] and observation of media such as commercial illustrations, it is apparent that the base components of an NPR style are the following:

- Colour Abstraction: The selection of colour to better convey surface geometry and feature information.
- Texture Abstraction: The reduction of high frequency noise and details.
- Silhouettes and Outlines: The use boundary lines to convey surface orientation.

The aim of this project is to investigate the use of NPR techniques in volume rendering for the purpose of emphasizing an area of interest, whilst abstracting the context information. A focus and context style combines two NPR styles (one for focus rendering, and one for context rendering) through a focus and context interface. A focus and context interface is the mechanism by which the focus and context regions are visualised relative to each other. A focus and context volume rendering framework will be implemented and evaluated. In doing so, a focus and context style which is effective for focus and context rendering will be identified and evaluated.

3.2 Requirements

For effective investigation of NPR techniques in focus and context rendering, 3D data rendered with different techniques and input parameters must be evaluated from multiple angles. With this in mind, the implemented framework should have the following characteristics:

- Flexible: The framework should be able to render different style images, depending on the input parameters.
- Interactive: The framework should allow for camera movement, lens movement and parameter changes at interactive frame rates (>5 frames per second).

In this dissertation, the word detail is defined as an indication of surface shape, surface orientation, or surface curvature. Techniques used to render the focus area should do the following:

- Clearly display a high level of detail.
- Increase salience in the region by emphasising the above mentioned details.

Techniques used to render the context area should do the following:

- Remove high frequency detail.
- De-emphasise remaining detail.

The interface between focus and context areas must convey the depth relationship between them.

3.3 Volume Rendering

3.3.1 GPU Raycasting

In order to process 3D volume data, the framework uses a simple GPU raycasting algorithm based on the work of Stegmaier [40]. Stegmaier used a single-pass pixel shader to project rays into a volume and sampled each voxel along the way to determine the final pixel colour.

3.3.2 Transfer Function

Direct volume rendering requires every voxel sample value to be mapped to opacity and a colour. This is done with a transfer function (TF) which can be a simple ramp, a piecewise linear function or an arbitrary table. Once converted to an RGBA (red, green, blue, alpha) value, the composed RGBA result is projected on corresponding pixel of the frame buffer.

Quite often, these TFs need to be user defined for a specific data set. The creation of a good transfer function can be a long and involved process. As stated in section 3.2, one of the main requirements of the framework is flexibility. The user should be able to easily change the appearance of the rendered image in real time, to fully explore NPR in focus and context rendering. For this reason, the framework utilises a simple TF for which the opacity given by the pseudo code in figure 3.1, where the user specifies the values Dmax, Dmin and UnifromO at runtime and D is the sampled voxel density. Essentially, this TF looks for voxels which contain a value within a certain threshold and assigns a uniform opacity. The resulting surface is called an iso-surface, and the threshold value is called the iso-value. Colour assignment is discussed in section 4.1.

```
If (D<Dmax AND D>Dmin)
        Opacity = UnifromO
else
        Opacity = 0
```

Figure 3.1: Generic transfer function pseudo code

The framework introduced in this report renders both the focus and the context regions using the method outlined above. The result is that one opaque surface each is rendered for the focus and context. This is contrary to the usual paradigm of displaying the context as many transparent layers (i.e. the skin, bone and underlying organs are all visible), which requires a well-made transfer function. The generic transfer function outlined above can be easily used to extract information out of any data set. The lack of many transparent layers means that the context region is also clear and uncluttered. I believe that one strong context surface is enough to provide spatial context to the focus region.

3.3.3 Gradient calculation

Local illumination techniques use gradient information during rendering. In the context of this dissertation, gradient in volume rendering is roughly equivalent to a surface normal in traditional discrete surface rendering. Gradient is typically calculated at each voxel. In the case of this project, gradient is calculated using the method of central differences. The method of central differences approximates the gradient as the difference of data values of two voxel neighbours along a coordinate axis, divided by the physical distance.

3.4 Pipeline Design

3.4.1 Focus and Context Shaders

The focus and context rendering pipeline I have developed relies on two shaders, one shader renders the focus area only and the other shader renders the context area only, herein referred to as the focus shader and the context shader respectively. Each shader has a separate generic TF (see section 3.2.2). Both shaders use ray casting to render to a texture each, which will be referred to as the focus texture and the context texture. This pipeline design allows the focus and context textures to undergo separate image space filters and techniques before being combined for the final image. The focus and context shaders can be modified in order to implement different focus and context interfaces. For the remainder of the dissertation, the phrase focus/context shader pair refers to a focus pixel shader and a context pixel shader, which render to textures that are later combined to form a cohesive focus and context image.

3.4.2 Seperating the Focus and Context Regions

During volume rendering, a sphere is inserted into the volume to dictate the lens position. This sphere is used as a reference to determine which voxels should be considered by the focus and context shaders. How exactly this is done differs between interface implementations (see section 4.5), but it is important to note that the volume of the sphere implies the focus region, and everything else is considered to be the context region. As a general heuristic of this framework, only voxels that reside in the focus region make a contribution to the focus texture. Context shaders may consider all voxels, or may treat voxels in the focus region differently, depending on the focus and context interface they are implementing.

3.4.3 The Rendering Pipeline

Figure 3.2 shows the proposed pipeline. The rendering pipeline is made up of two branches; one branch for focus region rendering and one branch for context region rendering. The pipeline is designed in a modular fashion to allow for the rendering of images with different combinations of colour, texture and silhouettes.

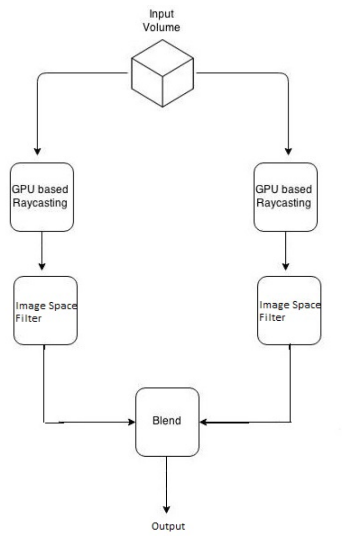

Figure 3.2: Propsed rendering pipeline.

Chapter 4

Implementation

This chapter is broken down by modules of the pipeline. The implementation of each module is outlined in detail. This is followed by a discussion and evaluation of the implementation. The purpose of the module evaluation is to determine the final focus and context style to be evaluated in chapter 5.

4.1 Colour Abstraction

4.1.1 Flexible Tone Shader

The flexible tone shader achieves colour abstraction by sampling a 1D texture according to according to surface normal and light direction. The texture itself is constructed on the fly according to an input float parameter ts ranging from 0 - 0.5, and two input colours. This tone shader works by taking samples of length ts from both sides of a smooth gradient texture, and stitching the samples together. The shader then samples from this stitched together texture according to the lambertian term (i.e. the dot product of normal and light direction). This concept is explained in figure 4.1.

Keeping in mind the interactivity and flexibility requirements of the framework, the Flexible Tone shader can achieve both Toon and Gooch shading by setting the input ts to 0 or 0.5 respectively. Incrementally shifting the value of ts from 0 to 0.5 will result in gradual softening of the terminators of each of the two input colours.

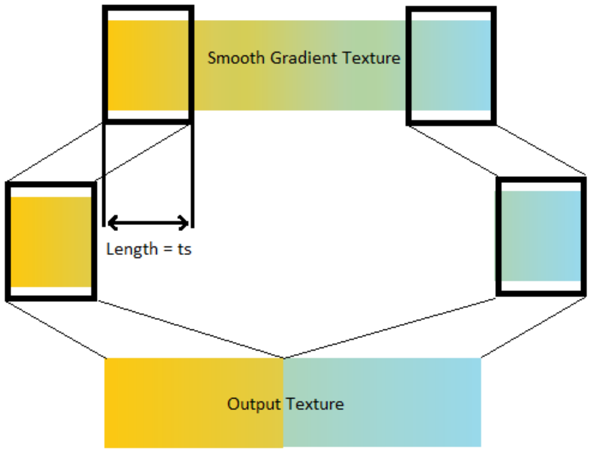

Figure 4.1: Two sample of length ts are taken from either end of a smooth gradient texture. These two samples are then stiched together to create the output texture. A lambertian term is then used to sample from this output texture. If ts=0.5, (i.e. detail=1), then the output texture will be the entire smooth gradient. If ts=0, then the output texture will only have two tones with a hard transition in the middle.

4.1.2 Module Evaluation

The images in figure 5.2 were rendered using the flexible tone shading model, each with a different ts input parameter. Toon shading (ts = 0) depicts the data set in only two colours, and thus produces the most highly abstracted image. The skin folds around the face are completely hidden, and curvature information is obscured. The effect is a flat image with a cartoony look, which is suitable for context area rendering.

The main caveat of this Toon shading is that irregularities in the dataset are very obvious due to their contrast to the surroundings, such as the dark spot above the lip. The harsh change in colour can also result in jagged edges between light and dark areas.

In the intermediate image (ts = 0.25), the skins folds underneath the chin are still visible, but the shading at the eyes (and other curved areas) appears to be split into light and dark areas.

The image rendered with 'smooth tone' shading (ts = 0.5) seems to provide the most information and the least abstraction. The skin folds underneath the chin are clearly visible and the curvature of the eyelids and sockets is shown as a subtle shift in shade. The apparent increased level of detail over Phong shading makes smooth tone shading suitable for rendering a focus area.

Figure 4.2: Tone shading, with ts values of 0, 0.25, 0.5 (from left to right)

4.1.3 Colour Selection

Further to the idea of using a gradual tone shift to present curvature information more clearly, figure 5.3 presents a tone shaded (detail = 1) image with carefully chosen warm and cool colours, akin to Gooch shading. Curvature is depicted with a hue shift rather than a luminance shift, which results in the information benefits of smooth tone shading but with more emphasis on curved regions. Out of all the shading styles mentioned, Gooch shading presents itself as the ideal style for focus rendering.

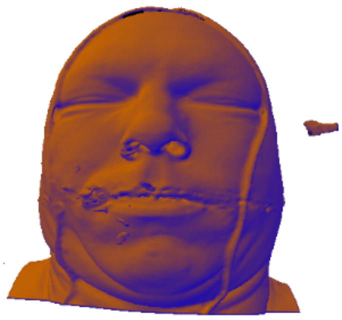

Figure 4.3: Gooch Shading

4.2 Texture Abstraction

4.2.1 Kuwahara Filter

The Kuwahara filter is a non-linear smoothing filter used in image processing for adaptive noise reduction. It blurs an image and reduces high frequency detail and noise, whilst preserving edges in the image.

The Kuwahara filter works by sampling the pixels around the input pixel, according to a window size input parameter. The window (square area of sampled pixels) is split into 4 sub windows, which overlap in the middle. The variance in pixel brightness of each sub-window is calculated. The Kuwahara filter then assigns the mean colour of the sub-window with the lowest variance (in brightness) to the input pixel. Figure 5.4 shows an example of a 5x5 Kuwahara, and how the window is split into sub-windows.

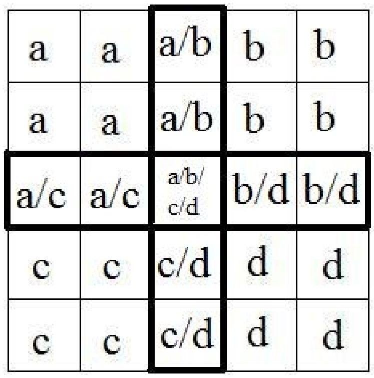

Figure 4.4: A Kuwahara window split into sub-windows

4.2.2 Module Evaluation

The Kuwahara filter appears to work well with Toon shading, as it complements the flat shading style by removing small irregularities in the data set and smoothing out the jagged edges between light and dark areas. We see this in figure 4.5, where a window size of 3 is enough to achieve this effect. As we increase the window size to 5 and 7, major features such as the eyelids and lips begin to fade. Increasing the window size to 10 results in a negligible difference for this data set.

Toon shading appears to benefit more from this simple implementation of the Kuwahara filter than Tone shading. Considering my earlier observation that Toon shading is quite useful for context rendering, and the fact that blurring is generally used to de-emphasize an area, the obvious conclusion is that the Kuwahara filter belongs in the context rendering pipeline.

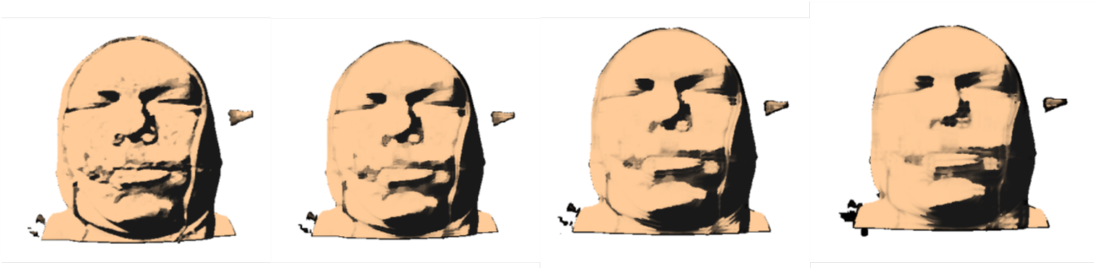

Figure 4.5: Toon shading with a Kuwahara filter with windows sizes of 2, 5, 7, 10 (from left to right)

4.3 Silhouettes and Contours

4.3.1 Sobel Filter

The Sobel operator performs a 2-D spatial gradient measurement on an image and so creates an image which emphasizes regions of high spatial frequency that correspond to edges. Typically it is used to find the approximate absolute gradient magnitude at each point in an input image. The filter uses two 33 kernels which are convolved with the original image to calculate approximations of the derivatives - one for horizontal changes, and one for vertical. The vertical and horizontal gradients are then summed to get the final pixel colour.

The RGB pixel values of each pixel in the filtered image are added to the corresponding pixels of the input image, to get the output image. In order to add flexibility to the framework, the filtered image is scaled by user controlled edge strength input parameter. The framework also has the ability to invert the colours of the filtered image, which results in darkened edges rather than brightened edges. This can be seen in diagram 4.6.

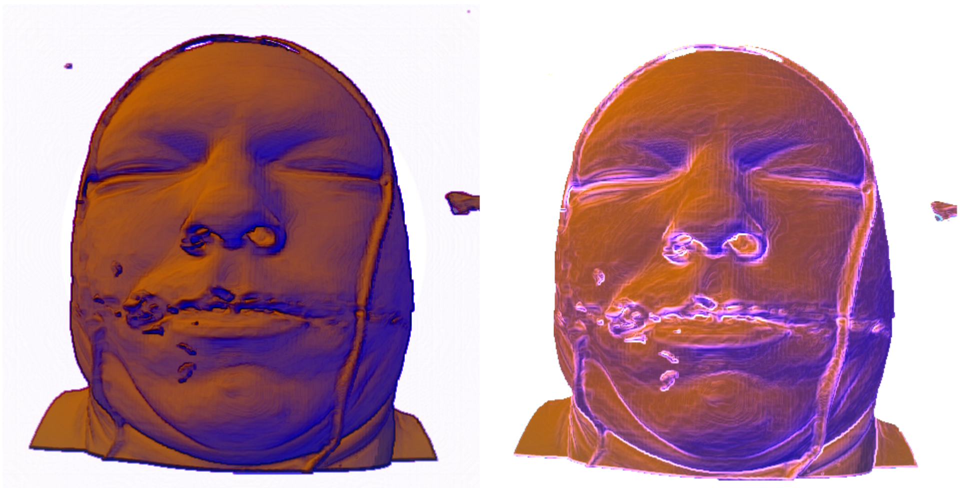

Figure 4.6: Gooch shading with a sobel filter used to darken lines(left) and brighten lines (right)

4.3.2 Module Evaluation

A Sobel filter was used to render silhouettes and edges. It was tested with both Toon and Tone shading styles and with different settings of the edge strength parameter.

A Sobel filter applied to a Toon shaded image picks up on changes in surface orientation (edge between light and dark areas) because of the hard change between colours, seen in figure 4.7. Changes in surface orientation are already the most obvious feature of Toon shading, and so the combination of Toon shading and image space edge detection appears redundant.

Figure 4.7: Toon shading with a sobel filter

In figure 4.8, we see the Sobel filter applied to a smooth Tone shaded image. Even with an edge strength of 0.3, contours, silhouettes and folds are noticeably emphasised at the lips, nostrils, eyes, and strands (wires?). Increasing the edge strength to 0.7 and 1.0 increases this emphasis, though a high edge strength parameter may also highlight unwanted high frequency details and imperfections in the data set.

The Sobel filter appears to emphasise features more strongly in Tone shaded images than it does in Toon shaded images. In general, edges and lines bring attention to an area, and so image space lines are effective for focus rendering.

Figure 4.8: Tone shading with a Sobel filter with line strength 0.3, 0.7, 1.0 (left to right)

4.4 The Pipeline

This section summarises the final pipeline design, according to the module evaluations. Figure 4.9 shows the rendering pipeline, and figure 4.10 summarises how each NPR technique fits into the pipeline.

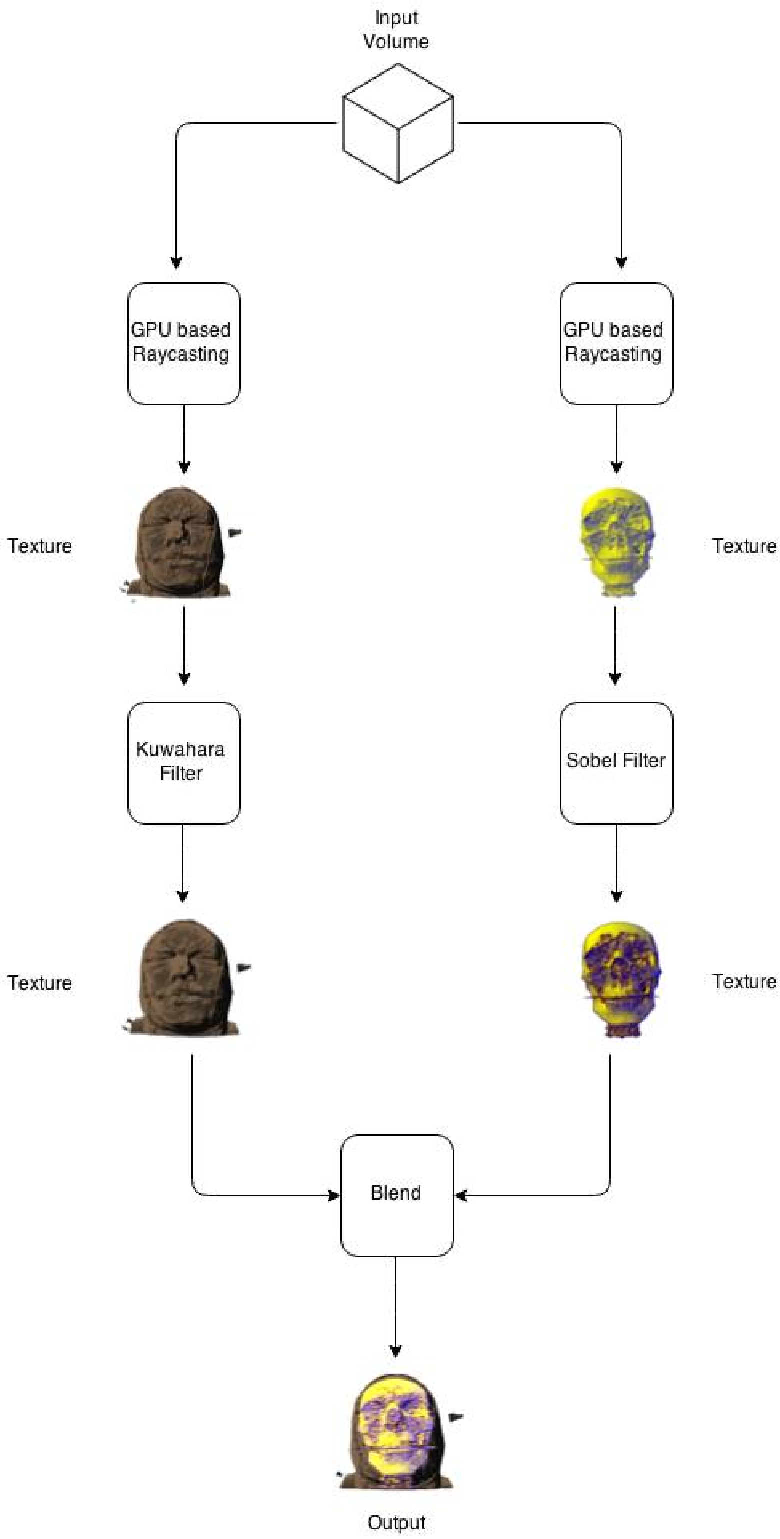

Figure 4.9: The final rendering pipline

Style Element	Strength	Weakness	Pipeline
Colour			
Gooch	More emphasis on details than Tone (detail=1)	---	Focus
Tone (detail = 1)	---	Details in shaded regions are visible	Subtle
Phong	Curvature emphasized by specular highlights	Loss of detail in shaded regions	-
Tone (0 < detail < 1.0)	Abstracts surface curvature whilst still presenting surface orientation and shape	Middle ground between emphasis and abstraction of detail	-
Tone (detail = 0), i.e. Toon	Highest level of detail abstraction	Jagged edges between shades and highlighting high frequency detail	Context
Texture			
Kuwahara	Gets rid of high frequency detail. Works well with Toon shading	---	Context
Silhouettes and Outlines			
Sobel	Emphasises feature shape	Does not work well with Toon shading	Focus

Figure 4.10: A table summarising how each module fits into the final pipeline

4.5 Focus and Context Interface

As mentioned in chapter 3, the focus and context interfaces described in this project rely on two shaders each. The pipelines for both interfaces are the same, however the VR shader-pairs differ. This section describes the implementation of both types of lenses.

4.5.1 Magic Lens

The Magic Lens paradigm modifies the opacity layers which obstructing data of interest and suppresses distracting information. It facilitates smooth transitions between focus and context areas.

During ray casting, any rays that intersect the 3D sphere are said to be making contributions to pixels in the focus area. In practice, this makes the Magic Lens a 2D lens which sits on the image plane. It is used as a mask for both the focus and context shaders. During ray casting, only the rays that pass through the lens make a contribution to the focus texture. All rays cast by the context shader make a contribution to the final image, but the contributions of rays that pass through the Magic Lens are reduced to facilitate blending in the final stage. So, the contribution of the focus area in the focus texture is inversely proportional to the contribution of the focus area in the context texture. A 'lens strength' input parameter determines the strength of the focus. A blending function is used to soften the edges of the Magic Lens, so that the focus area slowly fades into the context. The rendering of the magic lens is visualised in figure 4.11.

Figure 4.11: How the magic lens is composed from focus and context textures

4.5.2 Module Evaluation

In figure 4.12 we see that the Magic Lens is quite effective at focus and context visualisation. The focus area is clearly visible and enough of the context is preserved to indicate that the lips are close to the teeth, and the seamless blend between focus and context is visually appealing. However, the depth relationship between the teeth and the lips is not clear here. The smooth fade of the skin over the bone suggests that the bone is underneath, but this is only a weak depth cue, and is only effective at certain angles and lens sizes. It is clear here that a 2D technique (2D texture blending) is not effective at showing 3D relationships (depth).

Figure 4.12: The magic lens focus and context interface

4.5.3 3D Cutaway Lens

The 3D Cutaway Lens uses a 3D clipping frame to completely expose a focus region. It provides strong depth relationship cues between the focus and context areas.

The focus/context shader pair for my 3D Cutaway Lens extract iso-surfaces directly from the volume and perform all further calculations directly on the resulting surfaces. In iso-surface VR, the rays terminates once it samples a voxel with an index with a certain threshold, otherwise known as an iso-value. The surface that is extracted in this manner is called an iso-surface. The 3D Cutaway Lens algorithm considers two iso-values; one iso-value each for the focus and context iso-surfaces.

The sphere defines the focus volume, so any voxel that happens to be inside the sphere is consider to be in the focus region. All voxels around the sphere are considered to be in the context region. Only focus iso-surfaces are rendered in the focus region, and only context iso-surfaces are rendered in the context region.

There are two iso-surfaces, two regions and hence four cases that each shader needs to deal with. These cases and their appropriate actions are summarised in figure 4.13.

Iso-value	Inside Sphere	Outside Sphere
Context Iso-value	Ignore Continue	Colour the voxel according to the context transfer function
Focus Iso-value	Terminate Ray Return empty pixel colour	Ignore Continue

(a) Focus Table

Iso-value	Inside Sphere	Outside Sphere
Context Iso-value	Ignore Continue	Terminate Ray Return empty pixel colour
Focus Iso-value	Colour the voxel according to the focus transfer function	Ignore Continue

(b) Context Table

Figure 4.13: These tables detail what actions the focus and context shaders take, according to the sampled voxel and the region the sampled voxel is in. The top table represents the actions of the focus shader, adn the bottom table represents the context shader

4.5.4 Module Evaluation

The 3D Cutaway Lens presents much stronger depth cues between the focus and context than the Magic Lens, especially when rotating the model in real time. In figure 4.14, we see an example rendering of my 3D Cutaway Lens implementation. In all three data sets, it is obvious that the skin goes over the bone, as skin can be seen behind the focus.

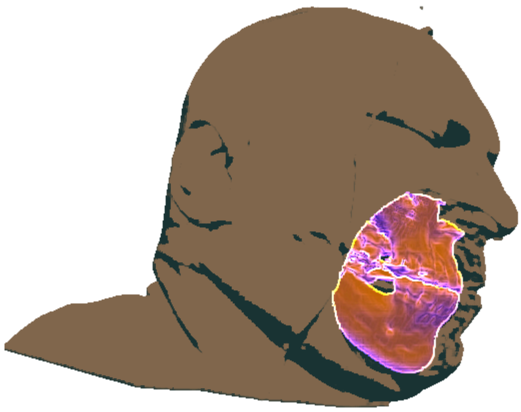

Figure 4.14: 3D Cutaway Lens

Chapter 5

Evaluation

5.1 Visual Evaluation

To evaluate the implemented focus and context rendering style, the final visual outputs of the framework were visually examined. This evaluation determines whether the requirements of the focus and context style outlined in chapter 3 are met. For the focus area, the requirements of the chosen style are that a high level of detail is presented clearly, and that salience is increased by emphasis of these details. The visual evaluation only deals with the first point, as salience is handled in subsection 5.2. For the context region, the focus and context style must remove high frequency detail and de-emphasise the remaining detail. Finally, the chosen focus and context interface should provide a strong indication as to the depth relationship between the focus ad context regions.

The images in figure 5.1 were rendered using the focus and context style deemed most effective in chapter 4. Detail was defined in section 3.2 as indications of feature shape, surface curvature and surface orientation. In the mentioned images, we see that the focus area is indeed quite detailed. The smooth tone shading gives a good indication of surface curvature. Since the surface orientation change is reflected by a hue shift rather than luminance shift (due to the careful selection of colour), surfaces facing away from the light are still clearly visible and detailed. The enhanced edges (brightened/darkened) and silhouettes in the focus outline the shape of a surface. In figure 5.1 (a), the curvature of the bone is apparent, and the bone shape has a bright outline against the context in the background.

In terms of the context area, the chosen focus and context style renders a very flat and abstracted context region. The combination of Toon shading and a Kuwahara filter removes all high frequency detail. In figure 5.1 (a) , only major surface features on the abdomen are left in the context region, such as the navel and the crease between the abdomen and right upper leg. The selection of colours relies on a slight shift in luminance only, reducing contrast (compared with the contrast in the focus) and hence de-emphasising the region.

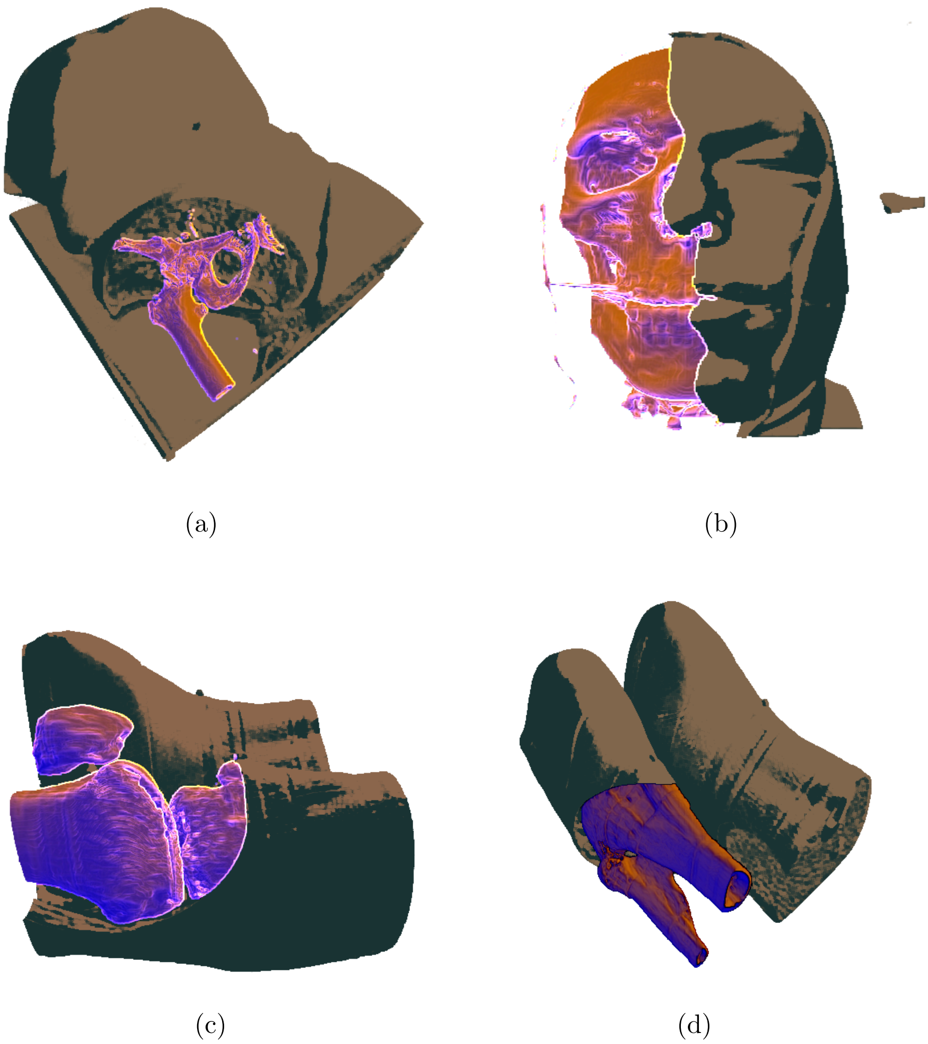

Figure 5.1: Visual examples of the final style rendering a) abdomen b) head c) knees d) knees

5.2 Saliency Test

According to object-based attention theory, the human brain groups similar pixels into coherent regions, which are called proto-objects. Viewer attention is given to the most salient image regions. The Saliency Toolbox estimates saliency according to colour, intensity and edge orientation. The Saliency Toolbox was used to determine the most salient proto-objects in images rendered by my frame work and in some control images, which were rendered with a Phong shader. This allows me to determine whether the focus and context style I am presenting in this thesis achieves the goal set out in chapter 3, which is to increase salience in the focus region by emphasising details.

Figure 5.2 shows the results of the saliency evaluations. The most salient region(s) in each image is circled in yellow. The images in the top row were rendered using my focus and context framework and the bottom row of images are Phong shaded control images. The evaluation of the control images allows me to confirm that the high level of salience in the focus region is indeed a result of my focus and context style, and not an inherent characteristic of the data set.

The most salient region in images b) and c) (figure 5.2) fall squarely in the focus region. The control images e) and f) depict the most salient region to be outside of the focus of their corresponding focus and context rendering, suggesting that the high salience of the focus region is a product of my focus and context rendering style.

In image a), we see that the focus is not always picked as the most salient region by the Saliency Toolbox. The eye was chosen as the most salient region, and the border between focus and context regions at the mouth is chosen as the second most salient. Intuitively, I would expect the purple and orange mouth region to stand out the most. This mismatch between the expected result and the actual result could be due to error in estimating salience, as salience estimation has not been perfected yet. Furthermore, the most salient regions in a) are still quite close to the focus area. This evaluation proves that the salience requirement of the focus and context style is met.

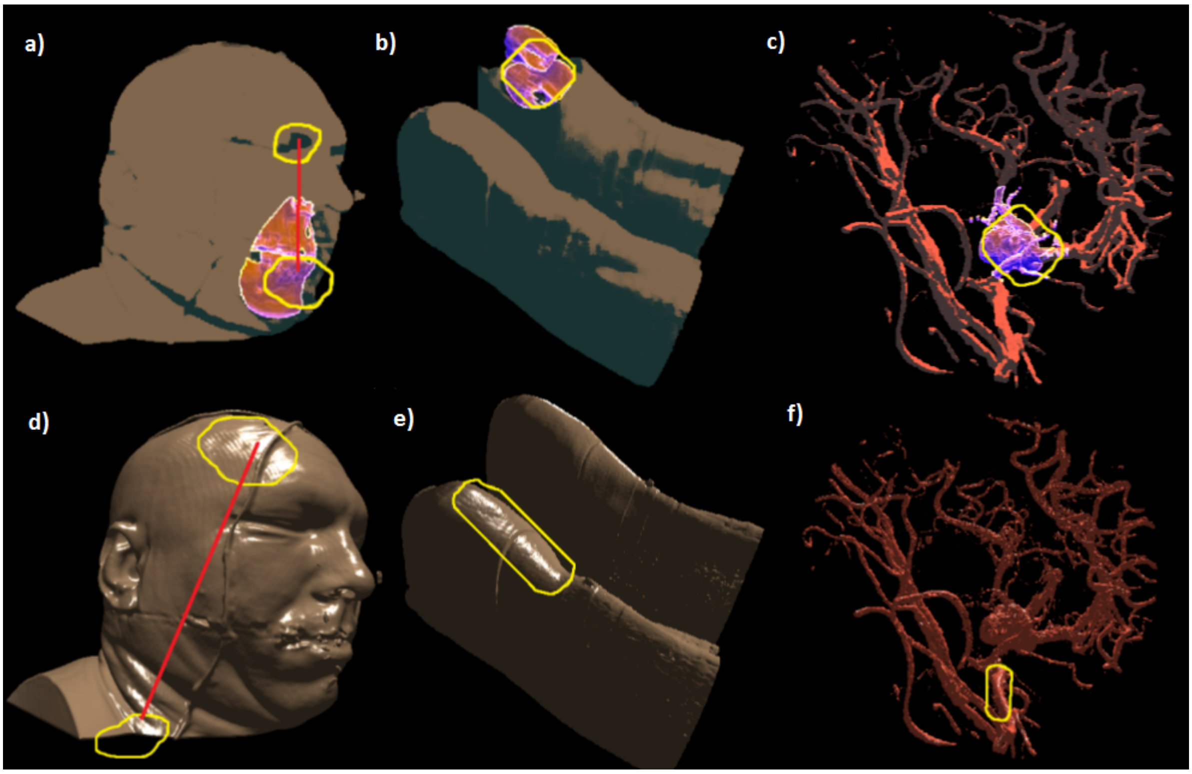

Figure 5.2: Salience test results

5.3 Performance

Performance was evaluated by measuring the frames per second (FPS) of the frame work while rendering different data sets, with various features switched on or off. Figure 5.3 summarises these measurements. Please note that the Kuwahara windows size for all tests was set to 5.

Dataset	Phong (control)	Clipping Lens	Clipping Lens with Kuwahara (5)	Clipping Lens with Sobel	Lens with Sobel, Kuwahara
Head	22	18	15	17	15
Knees	15	13	12	11	10
Abdomen	14	10	9	10	9

Figure 5.3: Performance Table

From figure 5.3 we see that all frame rates meet the 8 FPS requirement of the framework, set out in chapter 3. As expected, the most expensive feature is the lens, as we see a drop of roughly 20% in FPS when rendering the 3D Cutaway Lens (compared with the control FPS). This is due to the fact that two rendering passes are required to render the lens, as opposed to one rendering pass for Phong shading. The application of a Kuwahara filter results in a drop off of roughly 2 FPS. The application of a Sobel filter does not impact performance as heavily as a Kuwahara filter (with a window size of 5). This is because the Sobel filter considers fewer pixels in each iteration than the Kuwahara filter.

Chapter 6

Conclusions

6.1 Discussion

The goal of this dissertation is to investigate the use of NPR techniques in focus and context volume rendering. My aim was to do this by creating an interactive frame work that could render different focus and context styles. NPR techniques such as colour abstraction, texture abstraction and silhouettes and contours were explored evaluated with multiple focus and context interfaces in order to define an effective focus and context rendering style. Finally, this style was evaluated to prove its effectiveness in delivering compelling focus and context volume renderings.

I believe that I was successful in reaching these goals. The framework that I developed allowed me to interactively modify all three modules of an NPR style in both the focus and context regions. These were combined by a number of different focus and context interfaces to create images rendered in multiple focus and context rendering styles. Finally, a flagship style was identified and evaluated. The final focus and context rendering style rendered visually pleasing images which met the requirement outlined in chapter 3.

On a more practical level, this dissertation outlines novel implementations of well-known focus and context interface paradigms, namely the Magic Lens and the 3D Cutaway Lens. A novel tone shader which can interactively switch (and interpolate) between smooth tone shading and Toon shading was developed.

6.2 Future Work

While I am pleased with the results and contributions made in this dissertation, there is much room for improvement. This section describes improvements to the focus and context volume rendering framework which were not implemented either due to time constraints, or due to the scope of this dissertation.

6.2.1 Colour and Opacity

In general, the context region in focus and context volume rendering is an abstracted region which provides details about the rest of the data set apart from the focus. Thus, DVR which shows a transparent mixture of skin, vessels, muscles etc. could be more effective for context region visualisation than the iso-surface approach adopted in this dissertation. It would be interesting to extend my framework to use a user built transfer function for the focus, which assigned opacities that would make the entire structure of the volume visible, as opposed to just the outside surface (such as skin) as it was done in this dissertation. Further to this point, more advanced opacity models could be explored in the context.

6.2.2 Silhouettes and Contours

Object space lines are more effective for visualisation than image space lines because they allow for stylistic control. The implementation of style contours, proposed by Bruckner[11], was attempted for my framework but left in-completed due to time constraints. It works by to regulating contours based on the normal curvature along the view direction, and does not add much computation to the raycasting algorithm. Completing this implementation would be a good step towards improving the focus and context style presented in this thesis, as it is easy to implement in the given framework and would not slow the framework bellow interactive frame rates.

6.2.3 Focus and Context Interface

The 3D Cutaway Lens focus and context interface used in the final style provides the strongest depth cues of the interfaces implemented in this dissertation, but removes too much context information at large lens sizes. The focus region can also be obstructed by the context region. An improvement on the 3D Cutaway Lens would be to reduce the opacity of context information which falls in the focus, rather than remove it entirely. It would also be interesting to experiment with different lens shapes, such as a cylinder or a cone.

References

[1] L. Westover, "Footprint evaluation for volume rendering," *SIGGRAPH Comput. Graph.*, vol. 24, no. 4, pp. 367–376, Sep. 1990. [Online]. Available: http://doi.acm.org/10.1145/97880.97919 7

[2] W. E. Lorensen and H. E. Cline, "Marching cubes: A high resolution 3d surface construction algorithm," *SIGGRAPH Comput. Graph.*, vol. 21, no. 4, pp. 163–169, Aug. 1987. [Online]. Available: http://doi.acm.org/10.1145/37402.37422 7

[3] H. Müller and M. Wehle, "Visualization of implicit surfaces using adaptive tetrahedrizations," in *Dagstuhl '97, Scientific Visualization*. Washington, DC, USA: IEEE Computer Society, 1999, pp. 243–250. [Online]. Available: http://dl.acm.org/citation.cfm?id=647367.723598 7

[4] P. Lacroute and M. Levoy, "Fast volume rendering using a shear-warp factorization of the viewing transformation," in *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*, ser. SIGGRAPH '94. New York, NY, USA: ACM, 1994, pp. 451–458. [Online]. Available: http://doi.acm.org/10.1145/192161.192283 7

[5] A. Corcoran, "Enhancing real-time focus and context direct volume rendering," Ph.D. dissertation, Trinity College Dublin, 2012. 8

[6] B. Cabral, N. Cam, and J. Foran, "Accelerated volume rendering and tomographic reconstruction using texture mapping hardware," in *Proceedings of the 1994 Symposium on Volume Visualization*, ser. VVS '94. New York, NY, USA: ACM, 1994, pp. 91–98. [Online]. Available: http://doi.acm.org/10.1145/197938.197972 8

[7] Y. Sato, Y. S. Phd, S. Nakajima, N. Shiraga, S. T. Phd, and R. K. Md, "Local maximum intensity projection (lmip): A new rendering method for vascular visualization," 1998. 8

[8] J. Meyer-Spradow, T. Ropinski, J. Mensmann, and K. Hinrichs, "Voreen: A rapid-prototyping environment for ray-casting-based volume visualizations," *Computer Graphics and Applications, IEEE*, vol. 29, no. 6, pp. 6–13, Nov 2009. 9

[9] G. Marmitt, H. Friedrich, and P. Slusallek, "Efficient cpu-based volume ray tracing techniques," *Computer Graphics Forum*, vol. 27, no. 6, pp. 1687–1709, 2008. [Online]. Available: http://dx.doi.org/10.1111/j.1467-8659.2008.01179.x 9

[10] J. Kniss, G. Kindlmann, and C. Hansen, "Interactive volume rendering using multi-dimensional transfer functions and direct manipulation widgets," in *Visualization, 2001. VIS '01. Proceedings*, Oct 2001, pp. 255–562. 9

[11] S. Bruckner and M. E. Grller, "Style transfer functions for illustrative volume rendering. computer graphics forum," Tech. Rep. 9, 12, 14, 42

[12] D. Ebert and P. Rheingans, "Volume illustration: Non-photorealistic rendering of volume models," in *Proceedings of the Conference on Visualization '00*, ser. VIS '00. Los Alamitos, CA, USA: IEEE Computer Society Press, 2000, pp. 195–202. [Online]. Available: http://dl.acm.org/citation.cfm?id=375213.375241 9, 12

[13] J. Zhou, M. Hinz, and K. D. Tnnies, "Focal region-guided feature-based volume rendering," 2002. 9

[14] S. Marchesin, J.-M. Dischler, and C. Mongenet, "Per-pixel opacity modulation for feature enhancement in volume rendering," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 16, no. 4, pp. 560–570, July 2010. 10

[15] A. Gooch, B. Gooch, P. Shirley, and E. Cohen, "A non-photorealistic lighting model for automatic technical illustration," in *Proceedings of the*

25th Annual Conference on Computer Graphics and Interactive Techniques, ser. SIGGRAPH '98. New York, NY, USA: ACM, 1998, pp. 447–452. [Online]. Available: http://doi.acm.org/10.1145/280814.280950 10

[16] A. Lake, C. Marshall, M. Harris, and M. Blackstein, "Stylized rendering techniques for scalable real-time 3d animation," in *Proceedings of the 1st International Symposium on Non-photorealistic Animation and Rendering*, ser. NPAR '00. New York, NY, USA: ACM, 2000, pp. 13–20. [Online]. Available: http://doi.acm.org/10.1145/340916.340918 10

[17] P. Barla, J. Thollot, and L. Markosian, "X-toon: An extended toon shader," in *Proceedings of the 4th International Symposium on Non-photorealistic Animation and Rendering*, ser. NPAR '06. New York, NY, USA: ACM, 2006, pp. 127–132. [Online]. Available: http://doi.acm.org/10.1145/1124728.1124749 10

[18] G. Baolong and F. Xiang, "A modified octree color quantization algorithm," in *Communications and Networking in China, 2006. ChinaCom '06. First International Conference on*, Oct 2006, pp. 1–3. 10

[19] J. Lu, P. V. Sander, and A. Finkelstein, "Interactive painterly stylization of images, videos and 3d animations," in *Proceedings of the 2010 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, ser. I3D '10. New York, NY, USA: ACM, 2010, pp. 127–134. [Online]. Available: http://doi.acm.org/10.1145/1730804.1730825 10

[20] A. K. Jain, *Fundamentals of Digital Image Processing.* Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1989. 10

[21] "Extraction of multiple regions by smoothing in selected neighborhoods," *Systems, Man and Cybernetics, IEEE Transactions on*, vol. 7, no. 2, pp. 107–109, Feb 1977. 10

[22] P. Donitson, "Quantitative evaluation of edge preserving noise-smoothing filter," in *Geoscience and Remote Sensing Symposium, 1996. IGARSS '96. 'Remote Sensing for a Sustainable Future.', International*, vol. 3, May 1996, pp. 1590–1591 vol.3. 10

[23] J. E. Kyprianidis, "Image and video abstraction by multi-scale anisotropic kuwahara filtering," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering*, ser. NPAR '11. New York, NY, USA: ACM, 2011, pp. 55–64. [Online]. Available: http://doi.acm.org/10.1145/2024676.2024686 10

[24] J. Canny, "A computational approach to edge detection," *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, vol. PAMI-8, no. 6, pp. 679–698, Nov 1986. 12

[25] B. Csbfalvi, L. Mroz, H. Hauser, A. Knig, and E. Grller, "Fast visualization of object contours by non-photorealistic volume rendering," *Computer Graphics Forum*, vol. 20, no. 3, pp. 452–460, 2001. [Online]. Available: http://dx.doi.org/10.1111/1467-8659.00538 12

[26] M. Burns, J. Klawe, S. Rusinkiewicz, A. Finkelstein, and D. DeCarlo, "Line drawings from volume data," *ACM Trans. Graph.*, vol. 24, no. 3, pp. 512–518, Jul. 2005. [Online]. Available: http://doi.acm.org/10.1145/1073204.1073222 12

[27] G. Kindlmann, R. Whitaker, T. Tasdizen, and T. Moller, "Curvature-based transfer functions for direct volume rendering: Methods and applications," in *Proceedings of the 14th IEEE Visualization 2003 (VIS'03)*, ser. VIS '03. Washington, DC, USA: IEEE Computer Society, 2003, pp. 67–. [Online]. Available: http://dx.doi.org/10.1109/VISUAL.2003.1250414 12

[28] B. H. Eric LaMar and K. I. Joy, "A magnification lens for interactive volume visualization," 2001, pp. 223–. 13

[29] M. Ikits and C. D. Hansen, "A focus and context interface for interactive volume rendering," 2004. 13

[30] M. Hadwiger, C. Berger, and H. Hauser, "High-quality two-level volume rendering of segmented data sets on consumer graphics hardware," in *Visualization, 2003. VIS 2003. IEEE*, Oct 2003, pp. 301–308. 13

[31] M. McGuffin, L. Tancau, and R. Balakrishnan, "Using deformations for browsing volumetric data," in *Visualization, 2003. VIS 2003. IEEE*, Oct 2003, pp. 401–408. 13

[32] M. Chen, D. Silver, A. S. Winter, V. Singh, and N. Cornea, "Spatial transfer functions - a unified approach to specifying deformation in volume modeling and animation," 2003. 13

[33] C. Correa, D. Silver, and M. Chen, "Feature aligned volume manipulation for illustration and visualization," *Visualization and Computer Graphics, IEEE Transactions on*, vol. 12, no. 5, pp. 1069–1076, Sept 2006. 13

[34] S. Islam, D. Silver, and M. Chen, "Volume splitting and its applications," *IEEE Transactions on Visualization and Computer Graphics*, vol. 13, no. 2, pp. 193–203, Mar. 2007. [Online]. Available: http://dx.doi.org/10.1109/TVCG.2007.48 13

[35] C. Tietjen, T. Isenberg, and B. Preim, "Combining silhouettes, surface and volume rendering for surgery education planning," in *In IEEE VGTC Symposium on Visualization, 2005, 303310.* 14

[36] S. Bruckner, S. Grimm, A. Kanitsar, and M. E. Gr"oller, "Illustrative context-preserving volume rendering," 2005. 14

[37] S. Bruckner, M. E. Gröller, K. Mueller, B. Preim, and D. Silver, "Illustrative Focus+Context Approaches in Interactive Volume Visualization," in *Scientific Visualization: Advanced Concepts*, ser. Dagstuhl Follow-Ups, H. Hagen, Ed. Dagstuhl, Germany: Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2010, vol. 1, pp. 136–162. [Online]. Available: http://drops.dagstuhl.de/opus/volltexte/2010/2702 14

[38] J. Krger, J. Schneider, and R. Westermann, "Clearview: An interactive context preserving hotspot visualization technique." 14

[39] J. L. Mitchell, M. Francke, and D. Eng, "Illustrative rendering in team fortress 2," in *ACM SIGGRAPH 2007 Courses*, ser. SIGGRAPH

'07. New York, NY, USA: ACM, 2007, pp. 19–32. [Online]. Available: http://doi.acm.org/10.1145/1281500.1281666 15

[40] S. Stegmaier, M. Strengert, T. Klein, and T. Ertl, "A simple and flexible volume rendering framework for graphics-hardware-based raycasting," in *Proceedings of the Fourth Eurographics / IEEE VGTC Conference on Volume Graphics*, ser. VG'05. Aire-la-Ville, Switzerland, Switzerland: Eurographics Association, 2005, pp. 187–195. [Online]. Available: http://dx.doi.org/10.2312/VG/VG05/187-195 16