Unreal Realities:
Non-Photorealistic Rendering in Virtual Reality

Mr Peter Mills
University of Canberra
66 Earnest St
South Brisbane 4101
Peter.Mills@heliumdreams.com.au

Mr Justin T Carter
University of Canberra
66 Earnest St
South Brisbane 4101
Justin.Carter@canberra.edu.au

Abstract

Virtual Reality (VR) is a rapidly growing field, disrupting many industries, such as video games, engineering, architecture, and medical visualization. Designing VR experiences involves the use of digital technology and rendered 3D graphics to create immersive virtual environments. While traditional user interfaces require users to view and interact with a screen, VR systems place the user inside a virtual environment through the use of a head mounted display (HMD). This form of user interface has implications on how rendered graphics are perceived and interpreted. One rendering technique used extensively in the design and construction of virtual environments is Non-Photorealistic Rendering (NPR). NPR is primarily concerned with providing opportunity for a wide variety of expressive rendering styles such as toon, hatching and outline shaders.

This paper examines Non-Photorealistic Rendering techniques for virtual reality experiences, specifically focusing on strategies applied to achieve characteristics of toon, hatching and outline shaders, in virtual reality contexts. Through first identifying the common features traditionally used for NPR and then reconstructing these features in a virtual reality context the project illuminates unique considerations for practitioners implementing NPR effects in VR.

Keywords

Virtual Reality, Non-Photorealistic Rendering, Virtual Environments

Introduction

The notion of virtual reality has existed for some time. It was the technical advancements of the late 1980's that made the concept a reality [11]. VR is distinct from traditional forms of screen based media such as film, in that the user is immersed within a virtual environment [3]. This effect is achieved through technology such as "three-dimensional, stereoscopic, head-tracked displays, hand/body tracking, and binaural sound [3]. Over the last decade this technology has continued to advance, allowing virtual reality systems to become mass produced increasing availability to consumers and independent developers. As a result, these systems are rapidly being adopted by developers of digital content for purposes such as education and entertainment. When creating virtual world experiences designers transitioning from traditional screen media projects are now faced with the challenge of rendering digital content within a head mounted display. One field of rendering that is relevant to virtual reality projects is non-photorealistic rendering, as it allows designers to express many different artistic styles [10]. Non-photorealistic rendering encapsulates many forms and is derived from styles within the arts such as animated cartoons, painting, drawing and technical illustration [9]. This project examines non-photorealistic rendering as an important component of virtual reality creation, in order to identify implications for the designers of these experiences.

Existing Knowledge

Non-Photorealistic Rendering

Non-photorealistic rendering is a field of computer graphics described as having "two complimentary goals: the communication of information using images; and rendering images in interesting and novel visual styles which are free of the traditional computer graphics constraint of producing images which are 'life-like'" [9]. Non-photorealistic rendering can allow images to communicate information more effectively because the images are "free of the complications of shadows and reflections" [9]. This problem is described by Gooch, Gooch, Shirley, & Cohen [4] who state that realistic "phong-shaded 3D imagery does not provide geometric information of the same richness as human-drawn technical illustrations". Human drawn technical illustrations are frequently used for non-photorealistic rendering due to its need to communicate information effectively [4]. Another aspect of non-photorealistic rendering is the ability to create computer graphics effects that mimic "aesthetic human artistic styles"[9]. In their paper 'State of the Art Non-Photorealistic Rendering (NPR) Techniques' Sayeed & Howard[9] categorize non-photorealistic rendering techniques into five broad sections: stylized lighting, silhouettes and edges, volume rendering, pen-and-ink hatching, and engraving. These categories are then further defined by whether they are calculated in object space or screen space and if they have frame coherence. Three prevalent categories used in virtual reality projects are: stylized lighting, silhouettes and edges, and pen-and-ink hatching.

Non-Photorealistic Rendering in Virtual Reality

Existing knowledge surrounding the use of non-photorealistic rendering for virtual reality has already identified several important considerations. These are largely related to the methods in which virtual reality systems influence our visual perception. The nature of this is described by Richardt, Kyprianidis, & Dodgson[8] stating that "In the real world, the two views created by our eyes are inherently consistent, as they are both projections of the same 3D world. However, this is not necessarily the case for NPR techniques". As a result, the project involved the core features of NPR effects that indicate compatibility with virtual reality environments. These are temporal coherence, object space coherence and stereoscopic coherence[10].

Temporal Coherence

The notion of temporal coherence is an important consideration when creating non-photorealistic effects in virtual environments[7][8][9][10]. There are a number of different terms used to describe temporal coherence in contemporary literature such as frame coherence[7] and fluent image rendering[10]. This study applies the definition by Sayeed & Howard[9] who describe temporal coherence as a sequence's ability to maintain appearance over multiple frames without appearing to 'jump' or distort. They consider temporal coherence to be influential, given that most implementations of VE's are not static. Therefore, it is important that effects show consistency over time.

Object and Image Space Coherence

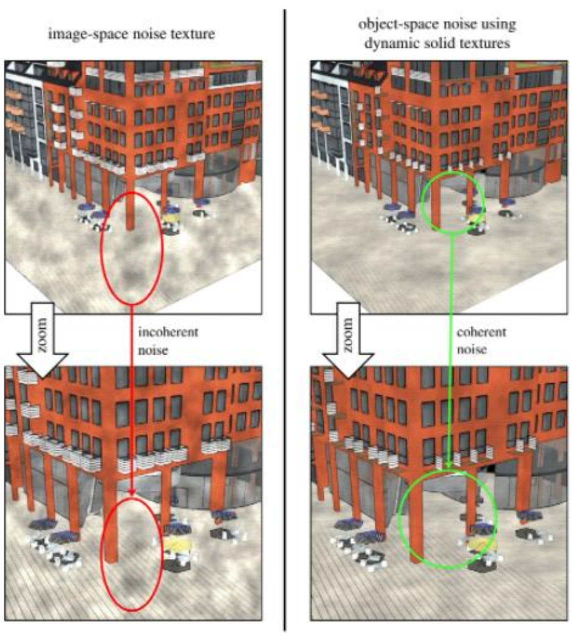

Figure 1. Comparing image-space and object-space to demonstrate the shower door effect[8].

When examining temporal coherence effects within virtual environments consideration must be given to the concepts of object and image space coherence. Object space coherence in virtual environment rendering involves a process in which the system calculates and renders levels of visual abstraction based on the position and relationship of objects being rendered in the scene. Image space coherence however calculates levels of visual abstraction based on the position of the viewers coordinates within the scene. When investigating these two forms of effects rendering, Richardt, Kyprianidis & Dodgson[8] found that the image space rendering methods were more likely to experience unwanted side effects such as the shower door effect. The shower door effect describes a virtual environment that appears as if viewed through a shower door or similar filter as shown in figure 1[8]. The shower door effect has previously been identified as having negative implications on temporal coherence[7]. Richardt, Kyprianidis, & Dodgson[8] also found that effects rendered using object space methods were more likely to be temporally coherent than those derived from image space effects. The current study therefore considers the object space method of image rendering as being more likely to offer positive experiences for viewers in regard to temporal coherence and reducing incidences of issues like the shower door effect.

Stereoscopic Coherence

Stereoscopic Coherence is identified by Schultz[10] as describing effects which are compatible with stereoscopic rendering and is a key feature of NPR effects in VR. All the features of NPR effects discussed are critical to creating stereoscopically coherent NPR effects in VR. Richardt, Kyprianidis, & Dodgson [8] identify temporally coherent effects as more likely to be stereoscopically coherent. Their work aimed to test the compatibility of object space versus image space non-photorealistic effects in stereoscopy. Their study found that object space effects were much preferred.

> "Overall, participants also found the object-based method more comfortable to watch. … It appears that stereo coherence is closely related to temporal coherence, as object-based techniques are also more likely to be temporally coherent than image-based techniques. However, further research is warranted to study this relationship in more detail" [8].

The effects created by this work should use these features as a template when considering how to create NPR effects which are suitable for VR.

Non-Photorealistic Rendering in Virtual Reality

This research project was conducted through a participatory action research approach in line with Kemmis & McTaggart[6]. In this approach the researcher is embedded in the study and critiques the research process and journey through collaborative discussions, reflection and reflexivity. Therefore the investigation involved the implementation of non-photorealistic effects in virtual reality systems through the creation and rendering of a virtual environment. Data collection and analysis was gathered through contemporary literature combined with a series of rendering prototypes. These prototypes focus on the implementation of non-photorealistic rendering in the context of virtual reality and utilize the Unity game engine.

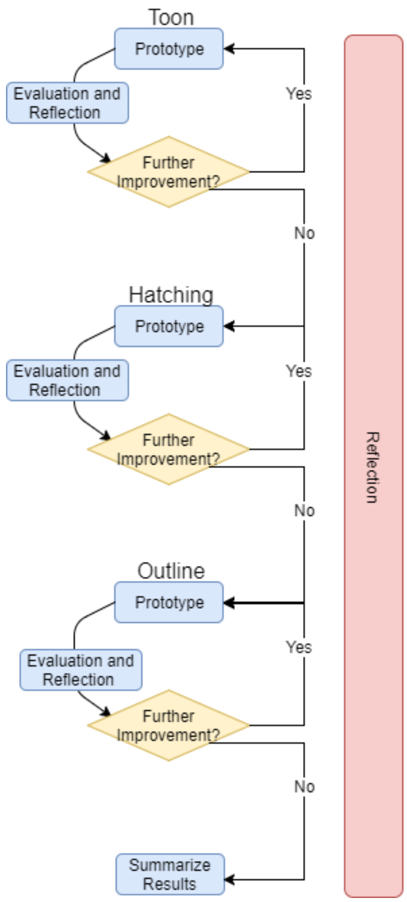

Figure 2. Overview of Research Methods

During each phase of evaluation, as shown in figure 2, a new implementation of the effect was created with any changes to improve compatibility of the effect with VR implemented. Compatibility is measured though the effect presenting with the key features of NPR described above; Object Space, Temporal, and Stereoscopic Coherence. This process was applied to three specific non-photorealistic effects commonly found within virtual environment design; toon shading, hatching and outlines. Although other forms of non-photorealistic effects are also found within virtual environments they lie outside the scope of this project. These effects were selected as they represent a diverse set of features of non-photorealistic rendering. Data for this study was derived from observation, reflection and discussion and is compiled in a development journal. Regular build snapshots were created throughout development in addition to a source controlled repository of the project.

Prototyping, Result, and Evaluation

Toon Shaders

Toon shading was selected as an example of stylized lighting for this study. Stylized lighting effects use a calculation to replace realistic shading models with false colours. A cartoon shading model normalises lighting values to give a sharp cut off between differently lit areas of a model reminiscent of cartoon scene as shown in figure 3[10]. Cartoon lighting is a distinctive style characterised by large uniformly or widely banded coloured surfaces with detail largely restricted to outlines" [9]. While there are many different methods described to create this effect, all can be broken down into two sections; the lighting model, and the abstraction. Most effects base their lighting model calculations on either Lambertian[5][1]. or Phong[4][2]. shading. Regardless of the lighting model used, the output of this stage is a luminosity value for each fragment. In traditional 3D rendering techniques this would be the value output for each fragment.

Figure 3. Example of toon shading[10].

The second part of an NPR toon effect, is the abstraction of the luminosity value to create the distinctive cell shaded effect. Again, two methods are popular; texture-lookup and procedural. The texture-lookup technique uses the luminosity value as a lookup value from a 1-dimensional texture which defines the shading of the fragment as shown in figure 4. This technique allows one shader to produce many different effects if using different 'toon ramp' textures and gives a lot of control to artists.

Figure 4. A black and white one-dimensional texture, with pixel positions from 0 to 1[5]

The procedural method contrasts the texture-lookup method by assigning colour values to ranges from the luminosity value in code. Figure 5 shows an example of this where luminosity values greater than 0.5 are shown as white, values less than 0.5 and greater than 0 are a dark gray and values less than 0 are black. Luminosity is represented in this example as 'nl', describes the dot product of the normal and the light vectors, representing how luminosity is calculated in the Lambertian lighting model[5].

```
if (nl > 0.5)
        color = vec4(1.0,1.0,1.0,1.0);
else if (nl > 0.0)
        color = vec4(0.33,0.33,0.33,1.0);
else
        color = vec4(0.0,0.0,0.0,1.0);
```

Figure 5. Code example for procedural toon ramp [5]

There is a consensus that specularity is an important part of toon shading effects as they convey "… information about

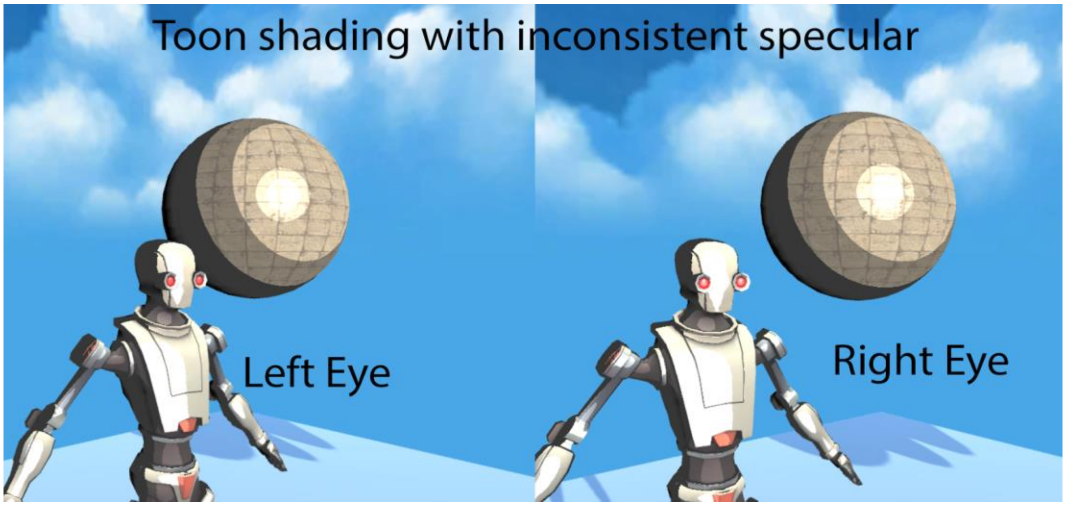

Figure 6. Example of stereoscopically inconsistent specular highlights in a toon shader.

surface type" [5]. The level of abstraction in this effect, especially when not using texture can lead to a lack of understanding in the view about the objects being rendered. Specular highlight help resolve this issue by "…provid[ing] visual feedback about the material type of an object (flat, shiny, metallic…)" [2].

Due to the use of traditional lighting models at the core of this effect, it was predicated that there would be minimal issues in a VR implementation. The exception to this is specularity. Specular highlights are a view-dependent effect that represents the properties of the material through the angle between the viewing direction and the light reflection vector [1].

The initial VR implementation of toon shading in this work used a Phong lighting model and procedural abstraction of luminosity for ease of implementation [5].

This version worked well in VR, despite a slight loss of depth information due to the flat shading with the exception of specularity. Specularity is a measure of how reflective a material is and is an important element in communicating information about surface textures in toon shaders [9]. Rendering specular highlights caused a cross eyed effect on non-uniform surfaces as predicted since each eye calculated a unique specular highlight. This can be seen in the specular highlight on the robot's head in figure 6. The distinctly different shapes of the highlight would lead to the area to the left of the robot's right eye being reflected in the right view and flat shaded in the left view. This causes a cross eyed effect when viewing this area. On uniform surfaces, such as a sphere, the highlight was stereoscopically coherent, since despite the view vector being different, there was consistency in the surface of the object. This meant the specular highlights stereoscopically resolved as a solid circle inside the sphere. This did not look like a specular highlight.

The main issue to resolve in the next iteration of the effect was the stereoscopically inconsistent specular highlights. To resolve the cross-eyed issue, and keep with the toon nature

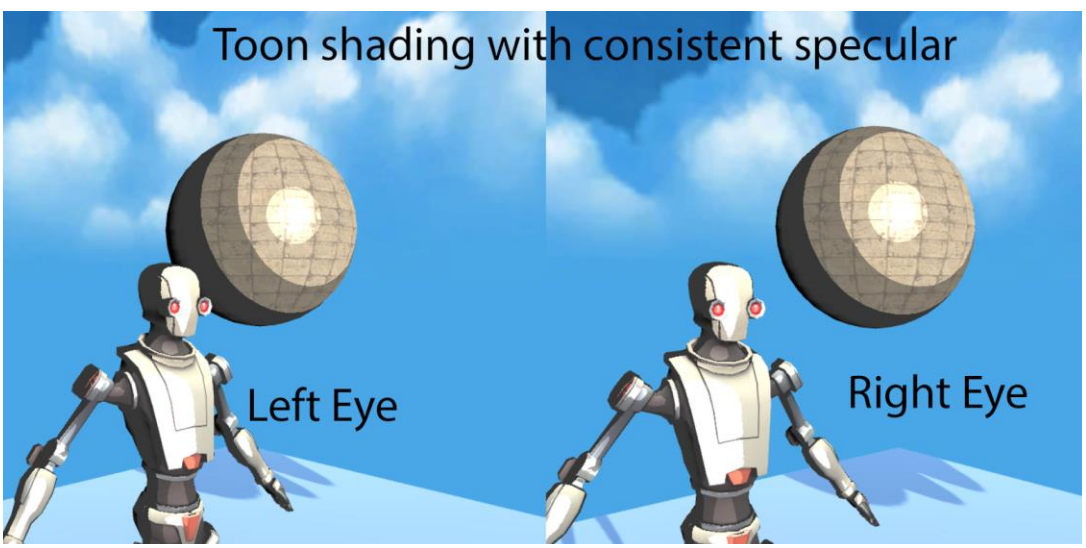

Figure 7. Example of stereoscopically consistent specular highlights in a toon shader.

of the effect, the specular highlight should not be stereoscopic and rather appear to be painted on the surface of the object. The loss of depth information is consistent with the cartoon goal of the effect.

To achieve this, the camera position used to calculate the specularity for each eye was replaced with the same global point representing a midpoint between the two eyes. This resulted in a less accurate specularity abstraction that was close enough to an accurate specular highlight to be believable but was also stereoscopically coherent because the same calculation was used for each eye. Figure 7 shows a consistently shaped specular highlight on the robot's head. The specular highlight on the sphere in figure 7 is also now consistent which can be identified by examining the constant distance from the left edge of the specular highlight and the darker band of diffuse lighting. This is opposed to the different distances in the inconsistent example in figure 6. This resolved the major issue identified with toon NPR effects in VR.

Hatching Shaders

Unlike, other NPR effects, Pen and Ink Hatching is a purely artistic effect, that does not aid in the communication of information. This is due to the limitations of the style, since it consisted only of monochrome pen stokes. The style can struggle to communicate the properties of a material. "This is solved through hatching, which is a technique that indicates curvature through stroke direction and surface texture, and lighting by stroke weight, density and rotated layering (also known as cross-hatching)"[9].

This work implemented a version of this effect described by Praun, Hoppe, Webb & Finkelstein[7]. Their method for Real-Time Hatching is calculated in object space so that it will be temporally coherent. Both of these features were previously identified as relevant features for VR. An important feature of Hatching Shaders identified by Praun, Hoppe, Webb & Finkelstein is consistent stroke width and density, even as an object moves further away from the viewer[7]. This has traditionally been a feature of image space hatching techniques, however, Praun, Hoppe, Webb & Finkelstein presented a method for achieving this consistency in object space using a Tonal Art Map (TAM). The TAM is a sequence of hatching textures, representing different levels of luminosity, with custom mipmaps that reduce the resolution of the texture, but not the scale of the strokes at each mipmap level as shown in figure 8. Mipmapping is a common technique in 3D graphics rendering to allow objects to sample a lower resolution version of a texture, if the object is further away from the camera. Using a TAM, as the object moves away from the viewer and different levels of the mipmap are sampled, the stroke width and density remains constant[7].

For our VR implementation, we were aware that the hatching shader constitutes a much stronger abstraction of the lighting model than toon shading. Thus, if specularity is included in the effect it should be calculated in a stereoscopically consistent fashion using the centre eye technique. However, as Praun, Hoppe, Webb & Finkelstein point out the strokes "character and aggregate arrangement suggests surface texture" [7]. Since the role of conveying information about the material can be left the TAM, specularity in the lighting calculation was left out of our implementation.

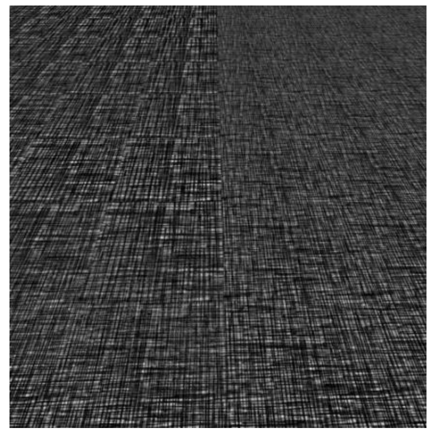

Figure 9. Comparing pen and ink hatching with and without TAMs. Left: with TAMs implemented for constant stroke width. Right: using traditional mipmaps without TAMs.

Since we predicted no complications with stereoscopic coherency, because the technique is object space coherent, our first iteration of this effect was setup to test the relative effectiveness of TAM's and traditional mipmaps. Figure 9 is a side by side of the two techniques on two planes receding into the distance. The left side shows TAMs implemented, as the strokes at the top of the image have the same width despite receding into the distance. The right side of the image shows traditional mipmapping where the texture reduces in resolution but not scale as the object recede.Both methods

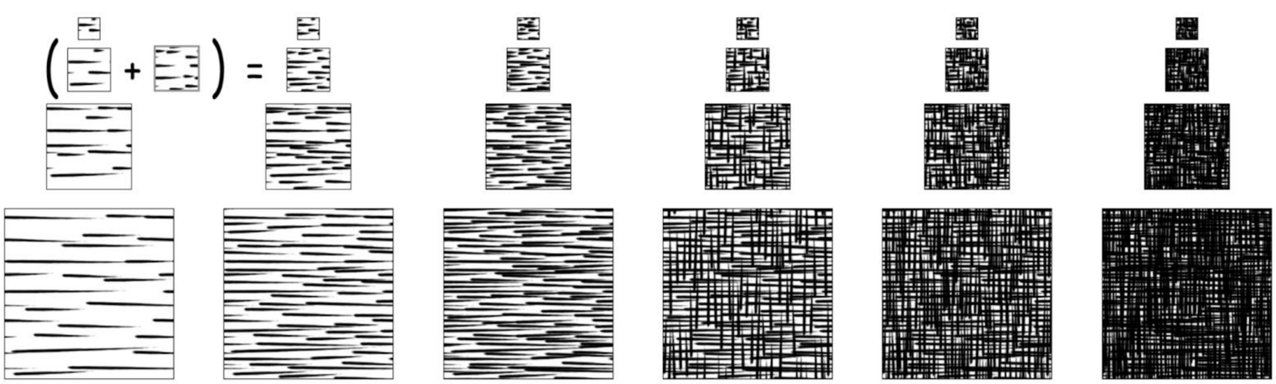

Figure 8. Example of stereoscopically inconsistent specular highlights in a toon shader[7].

presented no fundamental issues in terms of the core features of NPR compatibility with VR since both are Object Space, Temporally and Stereoscopically coherent. However, the need for consistent stroke width and density is part of making the effect appear as a 2D drawing, and while using the effect in VR this is still the goal in principle, we found that the stroke width remaining constant removed a vital depth cue, used to determine the size and scale of one's environment. We therefore found that the experience was better when the TAMs were not implemented, leaving the user with more depth cues.

Outline Shaders

Silhouettes and edges are a collection of techniques that mimic traditional line drawing methods because "line drawings constitute one of the most common and effective illustration styles" [9]. Rendering techniques in this category are used in "formal contexts such as technical illustration, architectural design and scientific diagrams" [9]. The delineation of internal edges, in addition to silhouettes, greatly increases the internal readability of the object [4]. This category of non-photorealistic rendering is also valuable in artistic contexts for its ability to evoke cartoons and sketches [9].

Our first implementation of outline shaders used a standard technique of rendering the object to be outlined in an initial shader pass, that is flat shaded in the desired outline colour, where the object vertices are scaled along their normals by the outline amount. The object can then be rendered on top of this silhouette normally if desired. The method works well in VR, as it is entirely object space coherent and temporally coherent. The limitations of this method in VR are the same as in traditional executions; models with hard edges the outline shape becomes discontinuous at the corners. This is demonstrated in the outline of the robots shoulder in figure 10. A range of other techniques for rendering outlines exist, however few seem to present issues specific to stereoscopic coherency based on their underling feature set as they are object space and temporally coherent.

Figure 10. Example of extruded normals outline shader without fill texture.

Results

NPR effects that do not feature spatial abstractions, like tone shading, should be naturally compatible with VR. However, NPR effects with spatial abstractions should use objects space calculations to prevent each eye from getting a different response where possible. Just as Image Space rendering in Real Time environments can be described as having a 'screen door' effect, effects in VR that are not stereoscopically coherent can have a 'cross eyed' effect where each eye is finding images that do not resolve stereoscopically. If an effect needs to use the view position or direction, thus being calculated in image space, it may benefit from calculating the abstraction for both eyes from a global centre point so that the abstraction is consistent across both eyes, forcing an images space effect to appear stereoscopically coherent, at the cost of accuracy.

Future Research

This work expounds the effect virtual reality has on a number of core features of NPR effects when implemented in a VR context. However, the importance of maintaining these features in the new medium was not investigated. This work discusses techniques for ensuring that the maximum amount of depth cues are maintained when implementing NPR effects in VR but does not assess the importance of these depth cues, or recommend a specific amount to maintain immersion. As this work focuses on the implementation of NPR effects in VR, issues such as user experience and immersion are outside of our scope. As an early Further work investigating the effect that different features of NPR effects have on users in VR could build off the results of this study.

Conclusion

The project illuminates unique considerations for practitioners implementing NPR effects in VR. Most effects which are compatible with real time environments will already have all the features needed to be suitable for VR. The main consideration specific to VR is large spatial abstractions, such as those found in the colour banding of toon shaders, or the addition of outlines. These abstractions only cause an issue if they are calculated using the view position or direction, since this means each eye will get a completely different image resulting in a cross eyed effect. However as the results of this study suggest, it may be possible to replace the view position with a global centre eye position that is consistent across each eye to make the effect stereoscopically coherent.

References

[1] Barla, P., Thollot, J., & Markosian, L. (2006). X-toon. *Proceedings Of The 4Th International Symposium On Non-Photorealistic Animation And Rendering - NPAR '06*, 127-132. http://dx.doi.org/10.1145/1124728.1124749

[2] Decaudin, P. (1996). Cartoon-Looking Rendering of 3D-Scenes. *Research Report INRIA*, (2919), 1 - 11.

[3] Earnshaw, R., Gigante, M., & Jones, H. (1993). Virtual reality systems (1st ed.). London: Blackwell Science. 3 -4

[4] Gooch, A., Gooch, B., Shirley, P., & Cohen, E. (1998). A Non-Photorealistic Lighting Model For Automatic Technical Illustration. Computer Graphics, 32(Annual Conference Series), 447-452.

[5] Hutchins, A., & Kim, S. *Advanced Real-Time Cel Shading Techniques in OpenGL. Computer Science Science at Rensselaer*. Retrieved 3 October 2017, from https://www.cs.rpi.edu/~cutler/classes/advancedgraphics/S12/final_projects/hutchins_kim.pdf

[6] Kemmis, S. & McTaggart, R. (2005). Participatory Action Research: Communicative Action and the Public Sphere. In N. K. Denzin & Y. S. Lincoln (Eds.), The SAGE Handbook of Qualitative Research. 3rd ed. (pp. 271-330). London: SAGE Publications.

[7] Praun, E., Hoppe, H., Webb, M., & Finkelstein, A. (2001). Real-Time Hatching. In ACM SIGGRAPH (pp. 581-586). Los Angeles.

[8] Richardt, C., Kyprianidis, J., & Dodgson, N. (2010). Stereo Coherence in Watercolour Rendering. In Symposia on Computational Aesthetics, and on Non-Photorealistic Animation and Rendering.

[9] Sayeed, R., & Howard, T. (2006). State of the Art Non-Photorealistic Rendering (NPR) Techniques. In EG UK Theory and Practice of Computer Graphics (pp. 1-10). Manchester: School of Computer Science, The University of Manchester

[10] Schulz, D. (2016). Non-Photorealistic Rendering in Virtual Reality (Masters). Institut Fur Informatik.

[11] Stein, C. (2016). Virtual Reality Design: How Upcoming Head-Mounted Displays Change Design Paradigms of Virtual Reality Worlds. Media Tropes, VI(1), 52 - 85.