

12-2016

Inter-color NPR lines: A comparison of rendering techniques

Donald G. Herring
Purdue University

Follow this and additional works at: https://docs.lib.purdue.edu/open_access_theses



Part of the [Computer Sciences Commons](#)

Recommended Citation

Herring, Donald G., "Inter-color NPR lines: A comparison of rendering techniques" (2016). *Open Access Theses*. 854.
https://docs.lib.purdue.edu/open_access_theses/854

This document has been made available through Purdue e-Pubs, a service of the Purdue University Libraries. Please contact epubs@purdue.edu for additional information.

INTER-COLOR NPR LINES:
A COMPARISON OF RENDERING TECHNIQUES

A Thesis
Submitted to the Faculty
of
Purdue University
by
Donald G. Herring

In Partial Fulfillment of the
Requirements for the Degree
of
Master of Science

December 2016
Purdue University
West Lafayette, Indiana

**PURDUE UNIVERSITY
GRADUATE SCHOOL
Thesis/Dissertation Acceptance**

This is to certify that the thesis/dissertation prepared

By Donald Grant Herring Jr

Entitled

INTER-COLOR NPR LINES: A COMPARISON OF RENDERING TECHNIQUES

For the degree of Master of Science

Is approved by the final examining committee:

Tim E. McGraw

Chair

Esteban Garcia

Voicu S. Popescu

David M. Whittinghill

To the best of my knowledge and as understood by the student in the Thesis/Dissertation Agreement, Publication Delay, and Certification Disclaimer (Graduate School Form 32), this thesis/dissertation adheres to the provisions of Purdue University's "Policy of Integrity in Research" and the use of copyright material.

Approved by Major Professor(s): Tim E. McGraw

Approved by: Mihaela Vorvoreanu

Head of the Departmental Graduate Program

9/29/2016

Date

ACKNOWLEDGMENTS

Special thanks go to my graduate-committee members – Prof. Tim McGraw, Prof. David Whittinghill, Prof. Esteban Garcia Bravo, and Prof. Voicu Popescu – for providing valuable feedback and suggestions for this project in numerous regards. I also wish to thank Prof. James Mohler for aid in revisions of this paper and for methodological and statistical guidance, as well as Rongrong Zhang for aid in choosing and applying the hypothesis tests in the study. Credit goes to Stephen Garver and William Huynh for co-developing the camera-track (flythrough) functionality with me that was utilized in the testing application. Finally, I want to thank the various subjects for volunteering their time and effort in the testing phase.

Parts of this thesis were published in *The Computer Games Journal*. The final publication is available at Springer via <http://dx.doi.org/10.1007/s40869-016-0022-3>

TABLE OF CONTENTS

	Page
LIST OF TABLES	v
LIST OF FIGURES	vi
ABSTRACT	ix
CHAPTER 1. INTRODUCTION	1
1.1 Scope	1
1.2 Significance	3
1.3 Problem Statement	4
1.4 Research Question	5
1.4.1 Sub-Question A	5
1.4.2 Sub-Question B	5
1.5 Assumptions	5
1.6 Limitations	6
1.7 Delimitations	8
1.8 Definitions	9
1.9 Summary	11
CHAPTER 2. REVIEW OF RELEVANT LITERATURE	12
2.1 History of Non-Photorealistic Line Generation	13
2.2 Inter-Color Lines	16
2.3 Thickness Variation in Rendered Lines	20
2.4 Color Differences	23
2.5 Summary	27
CHAPTER 3. METHODOLOGY	29
3.1 Research Questions and Hypotheses	30
3.2 Variables	31
3.3 Population and Sample	33
3.4 Testing Application	35
3.5 Procedure and Data Collection	36
3.6 Statistical Analysis	41
3.7 Threats to Validity	44
3.8 Summary	48
CHAPTER 4. RESULTS	50
4.1 Statistics	50

	Page
4.2 Demographics	53
4.3 Problems Encountered	54
4.4 Summary	56
CHAPTER 5. CONCLUSIONS AND RECOMMENDATIONS	58
5.1 Interpretations of Findings	58
5.2 Discussion of Problems Encountered	60
5.3 Future Research	61
5.4 Summary	63
LIST OF REFERENCES	64
APPENDIX	67
A.1 Lighting	67
A.2 Image Processing for Dynamic Method	67
A.3 Image Processing for Baked-Lines Method	71

LIST OF TABLES

Table	Page
4.1 This table sums up the choices of subjects between the NPR techniques with respect to each variable of interest. The middle and bottom sections tally the choices provided by the ten subjects with the longest interactive-phase times and the ten with the shortest times, respectively.	52

LIST OF FIGURES

Figure	Page
1.1 A non-photorealistic depiction (bottom) of a simple, heavily-textured cube on a textured plane is juxtaposed with an ordinary render of the same objects (top). The black lines are inter-color lines. Silhouettes are highlighted in red, and creases in blue. The color fade is unrelated to the present techniques, merely existing to emphasize the lines.	2
2.1 This render, which was created via a basic, image-space NPR algorithm, is composed of silhouette lines and crease lines. Labels indicate examples of each.	15
2.2 A ball rendered only with a silhouette (left) is ambiguous in shape. A visible texture on the non-planar surface (right) clarifies that the ball is not, for example, a disc facing the camera.	18
2.3 A textured cube rendered via the baked-lines technique is viewed from two angles. The thick line around the circle on the left side of the cube is clearly narrowed in the second view, as the left side of the cube experiences more perspective foreshortening in that view.	22
2.4 These pairs of baked-lines renders (left) and dynamic renders (right) were generated by the testing application used in this study. The lines in both varied with color differences and z-depth, but only those on the left varied with the view angle.	24
3.1 The two textures used for a pillow object in the 3D scene are juxtaposed for comparison. One features baked lines that were automatically generated by an external program, and the other does not.	37
3.2 This screenshot shows the initial view presented to subjects in the application. The upper-right area contains initial instructions.	38
3.3 This screenshot shows the automatic flythrough phase. The upper-right area contains the instructions displayed to subjects.	39
3.4 This screenshot shows Controlled-Flythrough Mode. The upper-right area contains the instructions displayed to subjects.	40
3.5 This screenshot shows Free-Camera Mode. The upper-right area contains the instructions displayed to subjects.	41

Figure	Page
3.6 This screenshot shows the final phase when the user is in Free-Camera Mode. The instructions tell the subject to switch to Controlled-Flythrough Mode to make the final selections, as there is no onscreen mouse cursor in Free-Camera Mode.	42
3.7 This screenshot shows the final phase when the user is in Controlled-Flythrough Mode. The instructional area now contains the final questions, the radio buttons, and the “Submit” button.	43
3.8 An example results file, “98897121.txt”, is shown. In each two-line block, the upper set of values, if present, are a set of position values for the camera, and the lower set, if present, are a set of Euler rotation values. These were recorded every 0.5 second during the interactive phase of a session. A line of dashes in place of a set of three values indicates that the position/rotation had not changed from the prior corresponding value in the last 0.5 second.	44
3.9 Some of the line thicknesses differ between the NPR techniques in this image, even though they were meant to appear equal. The grayscale color values in this book’s texture may not have been accurately preserved when the texture was loaded into Unity 3D.	46
3.10 In the baked-lines renders on the left side, some unwanted line-thickness variations were present on account of inconsistent UV spacing.	47
3.11 These renders were generated from a camera view on the default flythrough path. In the dynamic-method renders on the right, certain lines appear notably thick, particularly in view of the extreme view angles on the corresponding surfaces. It was possible that this could unintentionally mislead viewers into thinking that the dynamic technique was merely distinguished by overall thicker lines.	49
4.1 This MiniTab output, from a two-tailed power analysis starting from an α of 0.05, a power of 0.80, and a sample size of 49, indicates a threshold value of 0.695336.	51
4.2 This MiniTab output displays the results of two-tailed, single-proportion hypothesis tests that were carried out on each variable of interest. Each used an α of 0.05 and a power of 0.80.	53
A.1 The code for the first post-processing shader, which performs edge detection and comparison of colors across inter-color edges, is displayed here in full.	69
A.2 The code for the second post-processing shader, which generates NPR lines of varying thickness on the rendered image based on the output map from the first, is displayed here in full.	70

- A.3 These cropped portions of side-by-side NPR renders illustrate the partial-darkening effect for thin lines. On the right, the baked-lines method displays lines less than two pixels thick; the dynamic method draws thicker lines, but it dims them to compensate. 71

ABSTRACT

Herring, Donald G. M.S., Purdue University, December 2016. Inter-Color NPR Lines: A Comparison of Rendering Techniques. Major Professor: Tim E. McGraw.

Renders of 3D scenes can feature lines drawn automatically along sharp edges between colored areas on object textures, in order to imitate certain conventional styles of hand-drawn line art. However, such “inter-color lines” have been studied very little. Two algorithms for rendering these lines were compared in this study – a faster one utilizing lines baked into the textures themselves and a more complex one that dynamically generated the lines in image space on each frame – for the purpose of determining which of the two better imitated traditional, hand-drawn art styles and which was more visually appealing. Test subjects compared results of the two algorithms side by side in a real-time rendering program, which let them view a 3D scene both passively and interactively from a moving camera, and they noted the differences between each technique’s relative line thicknesses – the key visual disparity – in order to reach final judgments as to which better adhered to artistic conventions and which was more appealing. Statistical analysis of the sample proportions that preferred each algorithm failed to prove that any significant difference existed between the two algorithms in terms of either of the above metrics. Thus the algorithm using baked lines appeared to be more recommendable overall, as it was known to be computationally faster, whereas the dynamic algorithm was not shown to be preferred by viewers in terms of conventionality or aesthetics.

CHAPTER 1. INTRODUCTION

This initial chapter will explain the need for this research project, delineate its boundaries, and state the assumptions and definitions underlying its focus. It will first establish the scope and significance of the research, then lay out the problem and the question that requires investigation, and finally list various assumptions, limitations, delimitations, and definitions of terms.

1.1 Scope

This study delved into the topic of aesthetic, stylistic, and computational considerations between algorithms for a specific category of non-photorealistic rendering (NPR) techniques. This category involves automatically-generated line art that represents or accentuates object attributes in output renders. More narrowly, however, the lines in question are not of the commonly-studied types that define object shape attributes (e.g., silhouette or crease lines), but rather ones that correspond to edges between colored areas on object textures (see Figure 1.1 for examples). Techniques for rendering lines of this latter class, which in this text will be termed *inter-color lines* for the sake of brevity, may be implemented either via baking of the lines into object textures or via an image-processing algorithm that dynamically and interactively renders the lines along inter-color edges. Regardless of the implementation details, the lines may ideally vary in thickness according to the degree of difference between the base (i.e., unlit) texture colors on each side.

The researcher conducted a study comparing a computationally lightweight technique in the baked-lines category with a more complex technique in the dynamic, image-space category, investigating whether the latter outclasses the former in terms of viewer-perceived adherence to artistic convention and general

aesthetic appeal, given the existence of certain inherent differences between the two such as the undesirable perspective foreshortening of texture-baked lines in the former. This study focused on real-time NPR techniques, so that the findings would clearly be applicable both to static and to animated renders.

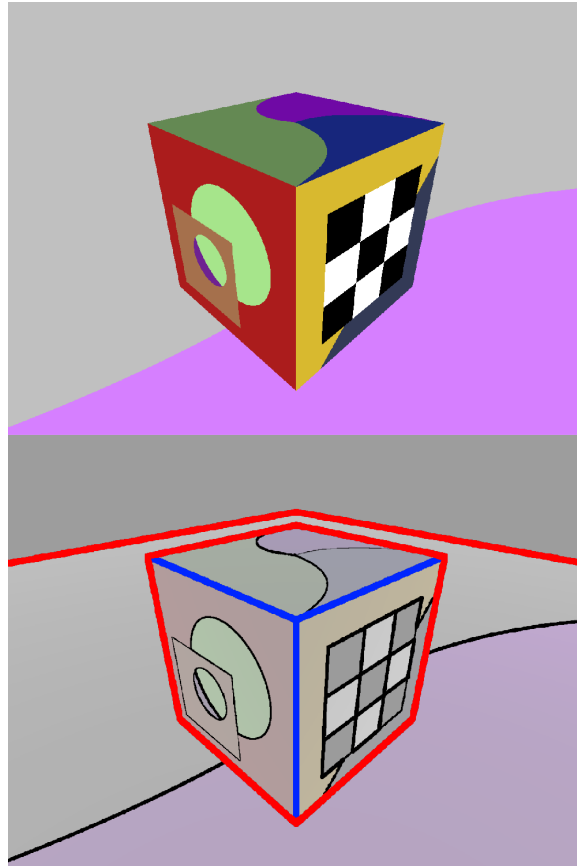


Figure 1.1. A non-photorealistic depiction (bottom) of a simple, heavily-textured cube on a textured plane is juxtaposed with an ordinary render of the same objects (top). The black lines are inter-color lines. Silhouettes are highlighted in red, and creases in blue. The color fade is unrelated to the present techniques, merely existing to emphasize the lines.

1.2 Significance

NPR has many real-world applications, especially in areas such as video games, animation, and technical illustration (Gooch, Sloan, Gooch, Shirley, & Riesenfeld, 1999; Magdics, Sauvaget, García, & Sbert, 2013); hence a great deal of research has taken place in this area. However, little has been conducted specifically on NPR techniques that place lines along inter-color edges on textures. The present research, by addressing this gap, paved the way for further discoveries and consequent technical improvements in the area.

Inter-color-line techniques are useful to artists and developers by virtue of adding to the pool of possible line art styles and thereby expanding the variety of options for establishing particular emotional impacts or atmospheres. This expansion also increases the likelihood that an artist will be able to find a style that fits his or her vision or matches an existing style in another medium.

In any NPR technique, computational efficiency is important, and this holds particularly true for real-time techniques like those used in video games, given the split-second time frame needed for each rendering cycle. However, it is likewise imperative to pay heed to aesthetic appeal, both for the gratification of the artist and of the viewer. Given that lines baked into textures are pre-rendered with predefined weights, techniques in the texture-baking category are decidedly far more computationally efficient. On the other hand, prior to this study it was considered highly possible that the inherent visual differences between the outputs of analogous techniques in each of the two categories in question (texture-baking versus dynamic, image-space generation) would lead to a disparity in levels of adherence to artistic convention and in levels of general aesthetic appeal in the eyes of viewers. Thus it was worthwhile to compare the outputs of analogous techniques from each of the two categories in terms of overall viewer perceptions of their adherence to artistic convention and aesthetic appeal, in order to properly evaluate the aesthetics-efficiency trade-off for each and to determine whether or not the

texture-baking technique presents a distinct overall advantage (i.e., equal or better perceived aesthetics and decidedly greater efficiency).

1.3 Problem Statement

As noted previously, many existing NPR styles for 3D scenes include lines denoting aspects of object form or boundaries (e.g., silhouettes and contours); this is especially true of those intended to imitate hand-drawn line art. However, some artists may also find it desirable for such styles to feature lines that correspond to edges between texture colors on object meshes. Such a technique would allow the output line art to represent scene objects in greater detail in cases where fill colors are absent from output renders. Additionally, certain styles – e.g., imitations of ones used in colored comic-book illustrations – may benefit from incorporation of these inter-color lines even if fill colors still appear in the output, with the lines superimposed.

For inter-color lines throughout a given render, diverse thicknesses are preferable over uniform ones for reasons of visual variety. Thick lines should indicate sharp, blatant color contrasts and closeness to the viewer, and thinner lines should indicate finer, more subtle color contrasts and greater distance from the viewer (Gooch et al., 1999). Thus, thickness of lines can be determined jointly via a scalar measure of color difference and via z-depth. This is possible both in a technique using texture-baked lines and in a dynamic, image-space technique; the thicknesses will be precomputed in the former and computed at run time in the latter. However, texture-baked lines will inevitably be foreshortened according to variation in polygons' view angles, and thus a confounding, view-dependent influence on line thickness will be introduced that is unlikely to correspond to any element of a hand-drawn art style.

In view of this, it seemed possible prior to this study that a technique using texture-baked lines would adhere less closely to artistic convention than an

analogous technique of the other type would. General viewer-perceived aesthetic appeal in this category of NPR is likely to depend in part upon adherence to conventions used in hand-drawn animations, given that such techniques are often clearly intended to emulate hand-crafted art, and given that viewers are likely to be highly familiar and comfortable with such conventions. Judgments expressed via viewer feedback can be utilized to gauge relative adherence to convention as well as aesthetic appeal. However, it was unknown before this study how well these two categories of techniques compared in these respects.

1.4 Research Question

When two NPR techniques are used to produce inter-color lines that vary in thickness according to color contrast and z-depth, and when one technique uses lines baked into textures and the other dynamically generates lines in image space, how will the two differ with regard to their resulting renders' viewer-perceived adherence to artistic convention and aesthetic appeal?

1.4.1 Sub-Question A

How will the two techniques differ with regard to the viewer-perceived adherence of their outputs to artistic convention?

1.4.2 Sub-Question B

How will the two techniques differ with regard to the viewer-perceived aesthetic appeal of their outputs?

1.5 Assumptions

The assumptions for this study included the following:

- Subjects would attentively observe all of the parts of the 3D scene in the testing application that were shown to them in flythroughs, so that they would be able to see and compare all of the lines that they were meant to see.
- Subjects would heed and understand the instruction to only devote attention to line-thickness differences when forming their judgments, as opposed to any other visual differences between the techniques.
- Subjects would not rush to submit their judgments between results of the two NPR techniques without full prior consideration.
- Subjects would be truthful in their judgment submissions, rather than choosing randomly or picking a render on a preselected side of the screen.

1.6 Limitations

The limitations for this study included the following:

- The only NPR lines associated with the research questions were ones drawn along sharp edges between colors on textures.
- The application drew some shape-based lines (silhouettes and creases) that did not vary in thickness, in addition to the inter-color lines, but only for the sake of visual completeness, visual clarity, and aesthetics.
- The study treated degree of color difference (defined as the difference across adjacent, unlit colors on a texture) and z-depth as factors that were common to both NPR techniques as influences upon output inter-color line thickness.
- The study treated degree of color difference and z-depth as having linear relationships with output inter-color line thickness.
- A specific color-difference metric based upon the RGB color model, which reasonably approximated perceptual uniformity while remaining

computationally simple, was treated as properly representative of color-difference metrics for real-time graphics applications.

- The key steps in the dynamic NPR technique that checked colors, determined line thicknesses, and generated the lines worked entirely in image space.
- The dynamic NPR technique output lines in a strictly rasterized format.
- These rendering techniques were applicable to any 3D scene with any object count and camera angle, provided that textures were unfiltered and not mipmapped.
- The application rendered in real time, making the study's results equally applicable in contexts of static and real-time rendering.
- All output line art was black.
- All subjects used the same computer for testing – an office computer which ran the program at a somewhat low and unstable framerate due to processor, GPU, or memory limitations.
- Total output resolution was limited specifically to 1920 x 1080, so that viewports would have the same dimensions for all viewers; this was a viable strategy because all testers used the same monitor.
- All textures used in the 3D scene were of high resolution, so that aliasing problems would not affect the findings.
- Persons over the age of 18 (primarily students at Purdue University) were eligible for participation regardless of any other demographic factors such as ethnicity or sex.
- For the sake of feasibility, recruitment of subjects primarily occurred within the Purdue Polytechnic Institute.

- User judgments regarding the output renders were expressed as two Boolean values – one expressing a preference between them in terms of adherence to artistic conventions in hand-drawn media and the other expressing a preference in terms of aesthetic appeal. Test subjects entered these via the program interface upon receipt of appropriate instructions.

1.7 Delimitations

The delimitations for this study included the following:

- While the application drew some shape-based lines, this study contained no new research concerning the determination of thicknesses for such line types.
- Potential line-rendering styles using lighting as an influence upon inter-color line thickness were not studied here.
- The line-generating algorithms and the application did not support texture filtering or mipmapping, and thus they would perhaps have worked poorly with low-resolution textures.
- The application did not feature vectorized output, and thus renders were not scalable independently of resolution.
- The study did not seek to examine how the varying of line colors may affect user perceptions.
- The study did not seek to examine how the varying of output resolutions may affect user perceptions.
- The study did not seek to examine how the varying of global line scaling may affect user perceptions.
- The dynamic, image-space NPR technique presented here did not incorporate any “content-aware” mechanism that would associate the maximum line

thickness with the maximum actual color difference found in an arbitrary scene’s textures (as opposed to the maximum possible difference).

- The study did not focus upon performance optimization for the presented algorithms.
- Persons who were under 18 years of age were not subjects in this study.

1.8 Definitions

The following are significant terms that merit precise definition:

Crease: A type of shape-based line that “indicate[s] discontinuities in the surface normals” and is “defined by the topology of the mesh itself, independent of the view direction or the camera settings” (Hajagos, Szécsi, & Csébfalvi, 2012). The ridge and valley lines discussed by Cole et al. (2012) fall into this category.

Delta E: (commonly: ΔE) The difference between two colors in a perceptually uniform color space, expressed as the Euclidean distance between them if they are regarded as points in 3D space and if the axes of the color space are the three dimensions (Mokrzycki & Tatol, 2011). It can also refer to a difference between colors in a non-uniform color space, but in such cases terms and coefficients are added to the formula to make the resulting difference values more uniform (Mokrzycki & Tatol, 2011).

Image-based: A term used in the context of line-based NPR for techniques that “focus on image processing methods such as edge detection to extract the lines at each pixel of the output image” (Cardona & Saito, 2014, p. 41) or in some cases at each pixel of one or more intermediate framebuffers.

Inter-color lines: NPR lines marking sharp discontinuities between areas of color on a texture applied to an object in a 3D scene. In the context of this study, in order to determine the existence and thickness of these lines, an algorithm

compares colors that are measured directly from texture pixel values, prior to lighting calculations.

Non-photorealistic rendering: (commonly: *NPR*) Rendering that “breaks free from the constraint of (photo-)realism that many other rendering techniques strive for” by producing “images and animations that at least in some parts appear to be made by hand” (Isenberg, Neumann, Carpendale, Sousa, & Jorge, 2006, p. 115). It focuses on stylization and abstraction.

Object-based: A term used in the context of line-based NPR for techniques that use 3D geometry characteristics to determine line placements and attributes before rasterization and typically generate polygonal strokes over the 3D models. They tend to utilize principles of differential geometry (Cardona & Saito, 2014).

Perceptual uniformity: (commonly: *perceptual linearity*) An ideal characteristic of a color space “in which the distance between the points defining individual colors would be proportional to the perceptual difference” seen by the human eye (Mokrzycki & Tatol, 2011, p. 4). It is also used here to describe color-difference metrics for which values are proportional to perceptual differences, even if they take parameters from non-uniform spaces and then adjust them.

Silhouette: A type of shape-based line that is generally drawn “where a continuous object surface appears to end” and “where the surface folds behind itself” – that is, where front-facing and back-facing surface areas meet (Hajagos et al., 2012, p. 71). Some authors, like Ao-yu, Min, and Jin-Xiang (2004), additionally use the term to describe other shape-based lines like creases (ridges or valleys).

1.9 Summary

This chapter has laid out the scope and boundaries of the project, presented the questions that need answers, summarized the problem and the importance of finding its solution, made underlying assumptions clear, and defined technical terms as necessary. The next chapter will begin with an overview of the researcher's approach to the reviewing of literature, which was conducted for the purpose of gathering background information in the field, narrowing the focus, and locating a knowledge gap. Then it will explore findings from relevant, prior studies that, when tied together, will justify investigation of the problem at hand as it was defined above.

CHAPTER 2. REVIEW OF RELEVANT LITERATURE

The researcher began this review of literature while compiling various papers that detailed prior studies and breakthroughs concerning non-photorealistic rendering (NPR) in general. The goal of this compilation process was familiarization with the existing state of the NPR field, as well as the accumulation of knowledge about various individual NPR techniques and algorithms and even the implementation of some of them in software. From a relatively early point, these searches tended to gravitate in particular toward NPR line-generation techniques, largely due to the researcher's personal interest in the topic. Over the course of this process, the researcher discovered most articles through Google Scholar – either via relevant keywords or via citation linkages – and accessed them via Purdue University's online library service.

A paper by Constable (2011), which offered guidance for new directions in NPR research from the viewpoint of an artist, was the key work that inspired the researcher to focus upon a little-studied type of NPR line. Constable's suggestions concerning a relative, contextual approach to determining the weights of NPR lines drawn between colors, based upon hue, saturation, and lightness (HSL) values, triggered the realization on the part of the researcher that precise determination of thicknesses of lines along edges between texture colors had been the subject of little, if any, prior research. Subsequently, the focus of this literature review narrowed significantly, with concentrations in the areas of line-based NPR in general, inter-color lines specifically, variations in thicknesses of rendered lines, and color-difference metrics usable for determining inter-color line thicknesses. The following sections of this review will correspond to each of these sub-areas, and the review will conclude by summarizing and linking together the various findings in order to clearly demonstrate the need for the present study.

2.1 History of Non-Photorealistic Line Generation

The field of non-photorealistic rendering for 3D scenes encompasses myriad styles that are generally oriented toward simplification and abstraction of visuals, limiting the output only to elements of high importance to human perception or central to the artist’s intended “message” (Magdics et al., 2013, p. 147). Many NPR styles directly emulate traditional art styles, such as pencil or ink sketches, paintings, or cel animation, as these styles depict objects in abstracted and aesthetically pleasing formats. Well-known classes of NPR techniques include automatic quantization of shading (i.e., “cel shading”), hatched shading, painterly rendering, and stylistic colorization.

However, one of the most heavily-researched areas is that of the generation of line art. Son, Kang, Lee, and Lee (2007) explain the value of line art as a perceptual aid by pointing out that line-based representation reduces objects to their defining characteristics (shape attributes), dispensing with insignificant details, and thereby makes visual processing faster and less burdensome to the mind. Manual line drawings by skilled artists that utilize relevant drawing conventions constitute the ideal targets for line-based NPR to match. Algorithms that automatically produce lines have been shown to generate passable approximations of these, featuring many of the same general line types, and these lines coincide with those in manual drawings to a significant degree (Cole et al., 2012). In addition, of course, automated algorithms can generate results on the fly, saving time and labor to an immense degree as well as reusing assets (i.e., the 3D objects in a scene). Opinions differ among researchers as to the efficacy of auto-generated NPR lines in representing comprehensible objects by themselves, but these differences appear to be attributable to variations between methods. Sanocki, Bowyer, Heath, and Sarkar (1998) found that certain conventional line-rendering methods of the time did not necessarily approximate line drawings well and did not always allow viewers to comprehend objects quickly and reliably, but Cole et al. (2009) presented evidence

that their own methods performed well – and similarly to hand-made drawings – in conveying shape details to viewers.

NPR lines can serve as plausible approximations to sketched lines or strokes from brushes or crayons, and when combined with fill colors and shading, they can emulate hand-drawn lines found in a colored sketch or a comic illustration. This can be particularly valuable in cases in which an artist or developer wants to find a rendering style that accurately matches a style previously defined in another medium; for instance, a video game developer may want a game’s graphics to resemble the art in a comic book from the same franchise. Stylistic choices depend upon the story and atmosphere that an artist wishes to communicate. For instance, an artist may desire a style evocative of children’s book illustrations in order to imbue the renders with an air of innocence. Magdics et al. (2013) explain how several types of NPR stylizations, including contour lines, can convey emotional impact through their attributes, and Duke, Barnard, Halper, and Mellin (2003) note that stylized NPR lines, through attributes such as thickness, can shape viewer expectations and emotional or cognitive assessments of scenes or objects. Thus one may infer that an increase in NPR style options constitutes an expansion of the range of possibilities for artistic expression. In addition, the representation of a scene as pure line art (i.e., lines that are not superimposed over a colored render) makes the printing or copying of a view of the scene more efficient, given the lack of tone variation (Saito & Takahashi, 1990).

Most of the seemingly countless studies on line-rendering techniques have focused on lines defining the shapes of scene objects. Shape-defining line types include silhouettes, creases (ridges and valleys), and suggestive contours (Cole et al., 2012). A silhouette is an outline marking the visual discontinuity between an edge of an object and a surface or background area behind that edge. Ao-yu et al. (2004) provide a survey of NPR methods for generating silhouettes. Ohtake, Belyaev, and Seidel (2004), as well as Nienhaus and Döllner (2004), provide methods for generating lines along ridges and valleys – that is, contours on an object along

which mesh curvature is high or angled. DeCarlo, Finkelstein, Rusinkiewicz, and Santella (2003), as well as Lin et al. (2015), present techniques for the rendering of suggestive contours, which DeCarlo et al. (2003) describe as “features that are almost contours, that become contours in nearby views” (p. 848). Figure 2.1 displays a render composed of silhouette lines and crease lines.

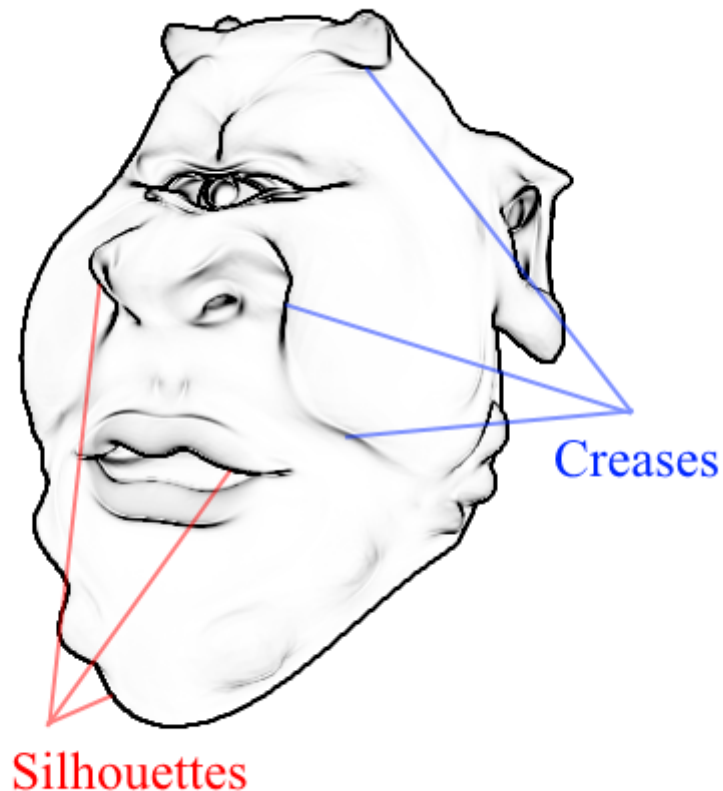


Figure 2.1. This render, which was created via a basic, image-space NPR algorithm, is composed of silhouette lines and crease lines. Labels indicate examples of each.

NPR techniques for line generation fall broadly into two categories: Object-space methods and image-space methods. Object-space methods directly utilize 3D geometric information, which is collected prior to the rasterization step of

rendering, to define the lines. Generally, as in the techniques of Hajagos et al. (2012) and of Cardona and Saito (2014), the lines themselves are initially created as polygonal structures, and they are subsequently rasterized along with the rest of the geometry. In contrast, image-space methods use information stored in the pixels of one or more framebuffers. Typically, such a method performs edge detection with neighborhoods of these pixels as inputs in order to generate the output line art. Some image-space techniques, such as the NPR edge-detection filters presented by Magdics et al. (2013) or Canny edge-extraction techniques similar to the one implemented by Sanoeki et al. (1998) for photograph processing, use the finished color map as an input buffer. However, many methods, such as those presented by Saito and Takahashi (1990) and by Nienhaus and Döllner (2004), instead use framebuffers containing geometric information, such as depth maps and surface-normal maps. The result of such techniques is that only discontinuities in depth and normal directions generate edges (lines), as opposed to object attributes unrelated to shape such as light intensity or texture coloration. Depth maps (i.e., the contents of the z-buffer) lend themselves well to silhouette generation, whereas normal maps lend themselves well to crease (ridge and valley) generation (Nienhaus & Döllner, 2004). The render in Figure 2.1 was generated through the use of Sobel edge detection on maps of these types.

2.2 Inter-Color Lines

However, despite the existence of copious research on generation of shape-based line art, very little NPR research appears to have taken place that specifically concerns inter-color edges on textures. While Constable (2011) offered a potential launching point for such studies in his recommendation of an algorithm that would provide “content-aware” establishment of inter-color line thicknesses, he conducted no research himself in this area.

In the context of the present study, the colors defining these edges were measured as they existed in the original texture images, prior to any blending with light values. (In the technique using texture-baked lines, these measurements were of course taken during texture creation as opposed to run time.) Many post-processing algorithms exist that dynamically generate lines for all sharp edges between colors in a final color map. For instance, Wang, Ma, Liu, Chen, and Wu (2012) utilize such an approach, and Magdics et al. (2013) combine such a method with one that uses depth and normal maps to generate shape-based lines. However, in this sort of approach, variation over time of light colors or intensities, as well as movements of objects or lights, can potentially result in unintuitively-sized shifts in color-difference values across a given texture edge due to the effects of color blending in the lighting step. Moreover, an abrupt, step-like change in lighting across a surface (e.g., in a render featuring quantized shading) can produce an output line where one should not exist. For this reason, algorithms that use unlit texture colors rather than light-blended colors to determine color differences across texture edges are worthy of study. With these algorithms, the color difference across any given edge will remain consistent between frames regardless of movement, the view angle, or a scene's lighting properties. It is additionally worth noting that influences from too many factors on line thickness would likely confound viewer observations regarding the precise effects of the factor(s) critical for study. In view of these considerations, lighting-influenced algorithms are impractical for a study such as the present one.

An algorithm that renders inter-color lines will add a new level of detail to a line render by carrying sharp edges between object texture colors over to the output, thereby preserving these often striking visual phenomena even in a render that lacks fill colors. While in some cases the designer of a line art style may prefer to forgo that level of detail, in other cases it can preserve critical visual information. For instance, in a lines-only render style in which only silhouettes, creases, and suggestive contours result in lines, a ball with a striking decal on it will merely appear as a circle, because the lines do not preserve the texture detail. Moreover, a

texture image can in fact aid viewers in gauging the shape of an object when ambiguity exists. In the previous style example, no shading will appear on the ball, and thus viewers may potentially perceive it as a flat disc facing the camera, but a spherically-distorted decal, if visible in the lines-only render, will clarify the ball’s form (see Figure 2.2). Geng (2010) makes a similar point when discussing how isoparametric curves can resolve shape ambiguity if added to a silhouette-line representation of a smooth object. Inter-color lines may also help viewers of a lines-only render to overcome issues with figure-ground segregation, which Sanocki et al. (1998) suspect to be a significant potential obstacle to comprehension of edge-detection-based renders; they speculate that “segregation may benefit from color and surface information because it could constrain possible groupings” (p. 344). On the other hand, it is important to recall that relative simplicity can be preferable in a line render on account of its visual-processing advantages, as Son et al. (2007) stated. Thus it is clearly important that artists strike an appropriate balance between detail and simplicity.

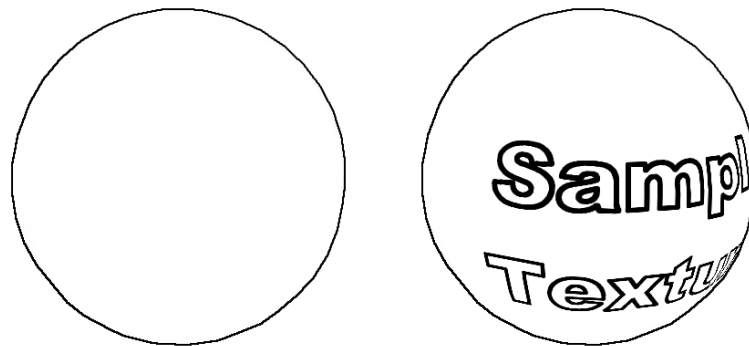


Figure 2.2. A ball rendered only with a silhouette (left) is ambiguous in shape. A visible texture on the non-planar surface (right) clarifies that the ball is not, for example, a disc facing the camera.

Mitchell, Brennan, and Card (2002) seem to be the only researchers on record as designers of a line-generating algorithm that deals specifically with

inter-color edges on unlit textures. In their unfinished study, they refer to the inter-color edges as “texture boundaries” (p. 239). They caution that their algorithm requires high-resolution textures in order to avoid potential issues with precise edge detection due to texture filtering. However, they do not lay out their algorithm in detail, do not specify a color-difference metric used to check whether an edge should have an output line, and do not provide any information as to whether or not their method handles variable edge thicknesses for inter-color lines.

Wang et al. (2012) present a method for generating inter-color lines of varying thicknesses via post-processing of renders. Although their approach does not examine colors from unlit textures, but rather ones from final color maps, the principles behind their method are still very useful for an approach of the former variety. In effect, their technique uses a specified color-difference metric to determine whether a line should be placed at any given inter-color edge, as well as how thick any given line should be (magnitude of color difference determines this). However, unlike the present study, their study does not compare their proposed image-filtering method to a lightweight alternative utilizing precomputed or baked values; instead, it compares it to a prior, similar image-filtering method.

With regard to the topic of NPR methods that dynamically generate inter-color lines, this researcher has found no other studies or projects as directly relevant as the aforementioned two works. However, certain principles applicable to techniques for shape-defining lines may also be applied in the area of inter-color line generation. Object-space line generation methods that depend on mesh topology do not apply where texture-based details are concerned, as the bounds and patterns of texture colors are independent of topology. Thus, algorithms that dynamically generate inter-color lines must operate in image space, which means that they can incorporate or adapt techniques used in ordinary image-space NPR, such as image processing techniques that operate on pixel neighborhoods.

As for NPR methods utilizing inter-color lines that were baked into the textures offline, this researcher has found no studies proposing or describing a

technique of this nature. However, it is an intuitive solution to the problem of implementing inter-color lines, given that precomputation of certain object or scene attributes inherently leads to run-time performance boosts, and given that no color-related attributes calculated in intermediate or final steps of the rendering process (e.g., light-blended colors) are required for definition of these lines. Despite the computational advantages, this type of method possesses certain limitations relative to dynamic NPR methods, given the latter’s inherent or potential invariance to certain scene attributes; the details of these differences will be discussed in the next section.

2.3 Thickness Variation in Rendered Lines

Ideally, NPR lines should vary in thickness, provided that available computing resources can handle any additional processing load that may ensue. Varying line thicknesses add visual interest, in contrast to the monotony of uniform thickness among all lines. Goodwin, Vollick, and Hertzmann (2007) note that “line style adds life and clarity to drawings” (p. 53) and that “a drawing with a plain or inappropriate stroke style can be lifeless or confusing” (p. 53), and they regard thickness as an element of line style. Also, line thicknesses can serve to highlight or downplay specific visual elements in a scene according to their intended importance or according to drawing conventions, and they can highlight shape characteristics of objects (Matsuo, Mikami, Watanabe, & Kondo, 2011) and perhaps aid viewers in discerning those attributes (Goodwin et al., 2007).

Historically, varying thicknesses of shape-defining NPR lines have often been influenced algorithmically by scene characteristics such as mesh shape attributes, spatial parameters, or lighting. An approach recommended by Gooch et al. (1999) will make lines thicker for components that are closer to the camera; in other words, it will let z-depth affect thicknesses. In a project by Matsuo et al. (2011), an algorithm sets NPR line thicknesses according to mesh curvature. Goodwin et al.

(2007) showcase a NPR method in which line thickness is a function of mesh curvature, depth, and lighting conditions. It is also worthy of note that certain traditional, image-space edge detection algorithms, such as the Sobel algorithm used to create the line render in Figure 2.1, naturally produce darker lines at sharp discontinuities between pixel values in input framebuffers; darkness of Sobel edges is a measure of line intensity, as is thickness, and when framebuffers containing depth and normal values are used (as in Figure 2.1), those two scene geometry attributes effectively determine output lines' relative intensities.

Masuch, Schlechtweg, Schönwälder, and Magdeburg (1997) recommend dark and intense lines for visual elements intended for emphasis by the artist, with lighter and fainter lines used for other elements; one can easily extend this principle to apply to thickness of lines as well as to darkness. NPR technology already allows the setting of line thicknesses according to artist-assigned importance of certain objects or object features. For instance, Cardona and Saito (2014) present a NPR method that permits continuous association of arbitrary line stylizations (including thicknesses) with specific regions on a mesh, regardless of camera view; this allows artists to add visual highlights, including increased stroke width, to certain parts of objects that they wish to emphasize.

In the specific case of NPR lines along texture-color boundaries (at least as defined in the present study), the only relevant scene attributes are the colors on either side of each of those boundaries and the z-depth. Even so, some of the preceding principles apply. Some inter-color boundaries are inherently more visually striking than others and thus are more visually important. Because thicker lines are associated with greater visual impact, it is natural to allow thickness to vary directly with degree of color difference. This model will not only allow the line art to augment the ability of viewers to quickly and intuitively distinguish between colors, but it will also allow the degree of color contrast across a given edge to be preserved even when an output render includes only line art without color fill. Moreover, varying thicknesses that convey color differences may subtly assist colorblind people

who ordinarily would incorrectly perceive certain color differences. If colorblind people observe the output line thicknesses, they will potentially comprehend the artist's intent more accurately regarding the visual contrasts in the scene.

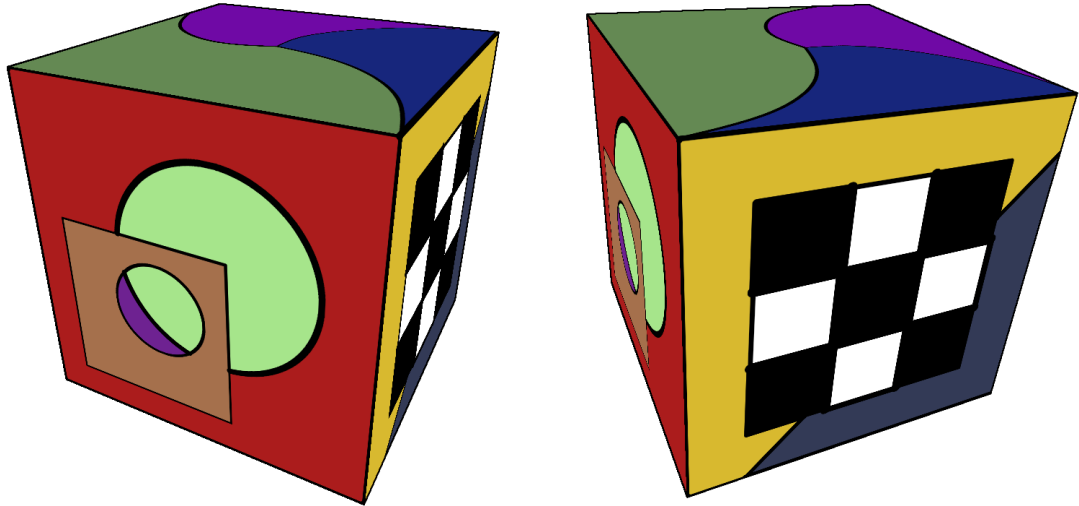


Figure 2.3. A textured cube rendered via the baked-lines technique is viewed from two angles. The thick line around the circle on the left side of the cube is clearly narrowed in the second view, as the left side of the cube experiences more perspective foreshortening in that view.

Line-thickness variation based on z-depth is possible in both classes of techniques under consideration in the present study (i.e., ones using inter-color lines baked into textures and ones dynamically generating such lines in real time). However, one inherent distinction between the two classes is that dynamic NPR methods optionally permit invariance of thicknesses to z-depth, whereas in methods using baked lines, perspective distortion will always make closer lines thicker. Another important distinction is that baked-lines methods, in which the inter-color lines are bound to textures on the planes of polygons, are inherently characterized by line-thickness variation based upon view angle, due to perspective foreshortening (see Figure 2.3). This limitation in particular is not to be ruled out as significant in

the eyes of viewers, as variation of inter-color line widths based on view angle is not known to correspond to any traditional drawing technique. As presently defined, these lines are not intended to convey texture details that are plastered on a surface, but rather to serve as abstract visual demarcations or accentuations of color discontinuity. As such, it is no more conventional to bind them to flat object-space surfaces than it is to bind abstract demarcations of shape characteristics to them, such as silhouette and crease lines. In contrast, line-width variation based on z-depth may often be regarded as an aesthetically pleasing option consistent with artistic conventions and therefore desirable for either class of technique, given that, in accordance with the aforementioned suggestion for NPR by Gooch et al. (1999), foreground objects in a scene are traditionally drawn with thicker lines than background objects are.

See Figure 2.4 for side-by-side comparisons between the two NPR techniques implemented in this study. Both featured line variation based on color differences and z-depth, but only the baked-lines method displayed perspective-based line variation.

2.4 Color Differences

For both classes of techniques, in any case, it is important to define a suitable metric for color differences across edges. A prior section addressed the fact that if inter-color boundaries on textures are to be defined in such a way that their output lines will remain consistent even under changing lighting and camera conditions, the only object attributes that will define them will be the base texture colors that appear on either side, measured prior to lighting calculations. (Z-depth may optionally influence line thicknesses but is not itself an inter-color-boundary attribute.) Therefore it appears logical to allow a scalar measure of color difference to affect inter-color line thickness linearly, provided that this metric is reasonably accurate and consistently proportional in value to human-perceived color difference

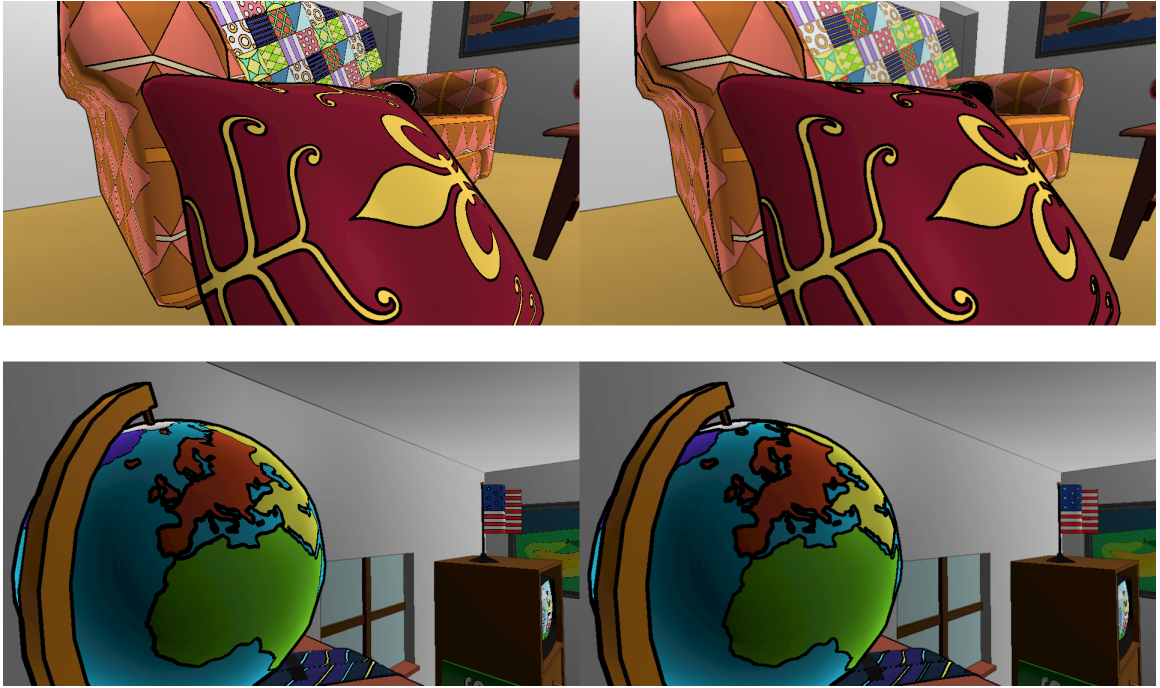


Figure 2.4. These pairs of baked-lines renders (left) and dynamic renders (right) were generated by the testing application used in this study. The lines in both varied with color differences and z-depth, but only those on the left varied with the view angle.

across all color pairings. Such *perceptual uniformity* is important; without it, a linear mapping would be inherently invalid to some degree. For example, to the human eye one pair of colors may appear to exhibit more contrast than another pair does, but with a non-uniform data representation the calculated difference values may be numerically equal, because some color components in that model may be more influential than others in determining human-perceived contrast.

No color-difference metric is perfectly uniform in reality; all are approximations at best. Mokrzycki and Tatol (2011) explain that in a truly uniform color space, represented as a 3D space with the color-space axes as the dimensions, color difference could simply be expressed as a Euclidean distance between two points that represent colors; this distance is called Delta E, often written as “ ΔE ”.

Because no real-world color spaces are truly uniform, existing Delta E formulations generally strive for approximate perceptual uniformity in their results by adding various terms or coefficients to the basic Euclidean distance formula. Mokrzycki and Tatol (2011) outline the history of Delta E formulations, beginning with the very simple Just Noticeable Difference (JND) formula from 1936 and documenting various new formulations that emerged up until 2000. They explain that many formulations directly superseded older ones on account of exhibiting demonstrably greater uniformity, and that older formulas tended to be modified with new coefficients or terms or through replacement of color-component parameters with new, derived ones.

Current state-of-the-art Delta E metrics include CIEDE2000 (also known as ΔE_{2000}) and CMC l:c, developed by two authorities on color standards – the Commission on Illumination (CIE) and the Colour Measurement Committee (CMC), respectively. These metrics are relatively complex to compute, and their parameters exist in the device-independent L*a*b* color space. They were primarily designed to distinguish accurately between small color differences, as in the field of colorimetry (Feldman, 1993).

In contrast, most color models used in computer graphics are neither perceptually uniform nor intended as such. The RGB color space, in which graphics applications internally represent colors, is a simple tristimulus model with intuitive primary colors; as Perrot, Habrard, Muselet, and Sebban (2014) point out, RGB is well understood to be a non-uniform space, and color distances in CIE-created color spaces, like that in which CIEDE2000 is calculated, are “more perceptual” (p. 97) than a Euclidean distance between RGB colors is. The same applies to HSL and the opponent-color model oRGB (Bratkova, Boulos, & Shirley, 2009), two other computing-oriented color models, given that they are derivations of RGB that were not designed for perceptual linearity either.

However, Mokrzycki and Tatol (2011) do describe a derivation of a RGB Euclidean distance formula that is in fact intended as an approximation of a

perceptually linear color model, albeit a rough one. Denoted as $\Delta E''_{\text{RGB}}$, this formula adjusts the parameters of the basic RGB Euclidean distance by assigning coefficients to the squared R, G, and B values, largely in order to account for the differing sensitivities of the human eye to the three primaries. The authors opine that this formula “gives results suitable for use in graphics software” (Mokrzycki & Tatol, 2011, p. 20) while still remaining computationally efficient.

High-end Delta E metrics like CIEDE2000 and CMC l:c appear ill-suited for real-time computer graphics applications, given their high computational complexity relative to their likely detectable advantages in their present application (i.e., as precise determinants of line thicknesses). Because the L*a*b* color space is device-independent, device-dependent RGB values must be converted to an intermediate format such as sRGB before translation to L*a*b* values, and the same is true in reverse. It is not even certain, given the above quote on the adequacy of $\Delta E''_{\text{RGB}}$ for graphics applications, that users will notice differences between line thicknesses set by a high-end Delta E algorithm and ones set by a fast but rough approximation thereof. Furthermore, as Feldman (1993) points out, Delta E metrics were mainly designed for examination of small color differences, and thus they may perform less consistently than expected in determining precise difference values between sharply-contrasting color pairs – perhaps no more consistently, or even less consistently, than some less uniform approximations.

Accordingly, the researcher has deemed it reasonable to choose the adjusted RGB Euclidean distance formula previously described ($\Delta E''_{\text{RGB}}$) as the color-difference metric influencing inter-color NPR line thicknesses. The high computational costs of using a high-end Delta E metric appear to significantly outweigh the likely benefits. On the other hand, the use of an even simpler and less uniform measure of color difference, such as a simple Euclidean distance formula using RGB components, would be likely to provide far lower perceptual uniformity but unlikely to provide a significant advantage over $\Delta E''_{\text{RGB}}$ in speed, in view of the fact that computation of each would differ only by a few arithmetical operations.

2.5 Summary

The length and complexity of the line of reasoning drawn out over the course of this review warrants a summarized reiteration.

NPR lines in general, like the hand-drawn lines that they emulate, provide a simplified, abstracted representation of a scene that is highly conducive to ease of perception, and artists and developers can choose from a broad range of NPR line styles that emulate hand-made drawings via quick and automatic processes. The utility of NPR lines is also clear in view of the wealth of literature already in existence concerning standard, shape-defining NPR lines. However, NPR lines generated along inter-color edges on textures additionally permit the transmission of certain scene details (e.g., striking patterns on textures) into line art – details that would be lost if only shape-defining line types were used. Moreover, these inter-color NPR lines further expand the range of possible art styles, and consequently they increase the capabilities of artists to accurately communicate the atmosphere that they want for a scene or to match an application’s NPR style to a style from another medium. Thus it is clear that lines of this type can be useful to artists and developers.

But if these inter-color lines are to be used, they should ideally vary in thickness, much as NPR lines that express shape should and frequently do. For inter-color lines, thickness variations according to color differences will not only prevent visual monotony but will also aid viewers in distinguishing between subtle and striking color edges or patterns. However, as this review has indicated, little research was conducted prior to this study regarding this type of line, and the researcher found none that examined NPR techniques in which such lines had been baked into textures prior to run time; thus, the researcher inferred that no comparison had been made in existing literature between the aesthetic qualities of such techniques relative to ones that generate inter-color lines in real time. Such a comparison of techniques was warranted, however, as it was possible that developers and artists would wish to decide on a trade-off between artistic quality and

complexity of computation. It was possible that dynamic NPR techniques would produce results more in tune with artistic conventions, due to the invariance of the lines to perspective foreshortening, but baked-lines techniques were clearly faster. Regardless of the outcome of the comparison, the meager knowledge base concerning optimal techniques for inter-color-line generation would be augmented.

A color-difference metric had to be chosen for both classes of techniques in order for such a comparison to be made. High-end Delta E metrics such as CIEDE2000 and CMC l:c closely approximate perceptual uniformity, which is optimal for linear mapping of color contrast to line thickness. However, less perceptually uniform metrics, while they provide a poorer mapping, are far more computationally lightweight and will not necessarily lead to renders that are perceptibly worse in artistic quality according to the human eye than high-end Delta E metrics will. $\Delta E''_{\text{RGB}}$, a rough approximation to a perceptually uniform Delta E metric, is both fast to compute and likely to set line thicknesses in a manner that appears uniform to observers. Thus it seemed warranted for use in this study's comparison of techniques for rendering inter-color lines.

In view of the above points, the researcher deemed the present questions – that is, the questions of how dynamic NPR techniques would differ from baked-lines techniques in terms of their resulting renders' viewer-perceived adherence to artistic convention and aesthetic appeal – to be worthy of examination.

In the next chapter, the experimental methodology will be outlined, in which human subjects used a testing application to examine the two proposed NPR techniques simultaneously and then provided feedback as to how the techniques compared in terms of the aforementioned two artistic qualities, thereby providing the needed information for answering the research questions.

CHAPTER 3. METHODOLOGY

This study utilized a rendering application as a testing tool to collect input from human subjects, who gave feedback regarding their preferences between two NPR algorithms – one using inter-color lines baked into textures and the other dynamically generating inter-color lines in image space. Each subject observed differences between two juxtaposed, real-time NPR renders of a pre-built scene, each demonstrating one of the techniques, over the course of two identical flythroughs and a subsequent session of arbitrary, interactive camera movement.

After this, each subject provided two binary responses via the interface, in response to textual prompts. One response indicated his or her preferred technique in terms of perceived adherence to artistic conventions used in hand-drawn media, and the other conveyed his or her preference in terms of aesthetic appeal. For each of these two variables individually, if subjects overall would prove to prefer one technique over the other by a significant margin – that is, if the proportion of the sample in favor of one technique would exceed a certain threshold determined by the sample size – a quality difference would be inferred to exist between the techniques. From thence conclusions would be able to be drawn regarding computation/aesthetics trade-offs between the techniques.

This chapter will lay out the specifics of this study in detail, beginning with the restated research question and its possible answers (the hypotheses), and continuing with the natures of the associated variables, the sampling approach, the testing application and procedures used for data collection, the planned statistical analyses of the resulting data, and some known threats to methodological validity.

3.1 Research Questions and Hypotheses

Broadly, as stated earlier, this research sought to establish whether or not one of the two proposed techniques for rendering inter-color lines of varying thickness – that is, the baked-lines technique or the dynamic, image-space NPR technique – emerged as clearly superior to the other in terms of aesthetics and in terms of adherence to conventions observable in other visual media.

Two pairs of hypotheses were tested, with each pair consisting of a null hypothesis and an alternative hypothesis.

H1₀: No significant difference exists between the proportions of the population that prefer each rendering technique in terms of adherence to artistic conventions.

H1_α: A significant difference does exist between the proportions of the population that prefer each rendering technique in terms of adherence to artistic conventions.

H2₀: No significant difference exists between the proportions of the population that prefer each rendering technique in terms of aesthetic appeal.

H2_α: A significant difference does exist between the proportions of the population that prefer each rendering technique in terms of aesthetic appeal.

In effect, each null hypothesis stated that both proportions were equal at 1/2 ($P = 0.5$, where P is the proportion in favor of either technique), and each alternative hypothesis stated that the proportions were unequal, with neither at 1/2 ($P \neq 0.5$). Thus, the hypothesis test needed to be two-tailed.

3.2 Variables

This study involved two binary, categorical variables of interest, each of which held tester feedback. They corresponded to the two viewer-perceived technique qualities from the research questions. The first variable stored each tester’s selection of the rendering technique that he or she found more adherent to artistic conventions, and the second stored each tester’s choice of the technique that he or she found more aesthetically appealing. For each variable, a tester selected only one value – either one corresponding to the baked-lines technique or one corresponding to the dynamic technique. Thus, each user only entered two Boolean data elements in total, after a single session of exploring the scene and observing the renders.

The prompt for the first response was worded as follows: “In terms of line thicknesses, which of the bottom two rendering styles more strongly resembles a conventional hand-drawn animation style, in your opinion?” The prompt for the second was worded thus: “In terms of line thicknesses, which of the bottom two rendering styles looks more appealing to you overall?” For each of these, the user was only given “Bottom Left” and “Bottom Right” as selectable options, as the bottom two onscreen viewports contained the two inter-color-line techniques. Furthermore, the left-right positioning of these viewports was randomized, so that some testers saw the dynamic technique on the left, whereas others saw it on the right. This feature ensured that any bias related to positioning on the part of users would not have a uniform effect on the results.

The researcher decided to limit the options for the users to two, rather than to provide them with a multi-value scale (e.g., a Likert scale), for three reasons: The impossibility of guaranteeing that different testers would similarly evaluate magnitudes of inter-technique quality gaps, the fact that Likert scales and similar devices could not be assumed to feature equal spacing between adjacent values, and the inherently low usefulness of a middle (“no preference”) option for gauging a difference between two things expected to be widely regarded as highly similar.

The sole intended visual differences between the techniques, which would form the basis of the judgments of testers, lay in the specific ways in which line thicknesses varied in each technique. While both featured variation with z-depth and with magnitude of color difference as measured via a rough Delta E metric ($\Delta E''_{\text{RGB}}$), the baked-lines technique also inescapably distorted line thicknesses according to perspective foreshortening. The researcher attempted to control all variables that could cause visual differences between the techniques but did not showcase a significant, inherent deficiency in one technique. For instance, line-thickness variation based on depth, which always had to be present in the baked-lines technique, was intentionally incorporated into the dynamic technique as well and matched between the techniques as exactly as seemed possible, because the baked-lines technique’s lack of an option for depth-invariant thickness was not a significant deficiency by definition (in many cases, depth-based variation could in fact be regarded as stylistically preferable over depth-invariance). As explained in Section 3.4, some unwanted differences between the implemented techniques did appear in the end, such as the decrease in quality (“disintegration”) of the texture-baked lines when they were viewed at great distances. Therefore, in an ad-hoc solution, subjects were explicitly instructed to ignore differences in any aspects other than line thickness, on the grounds that they were irrelevant to the study’s focus (i.e., confounding factors).

Testers were informed even prior to the start of their sessions – both via the written recruitment materials and via the spoken instructions of the researcher, who attended all the sessions – as to what the two variables of interest were, so that they would not potentially waste time viewing the differences between the renders without knowing the context or purpose of their observations. However, they were not explicitly informed of the influence (or lack thereof) of perspective foreshortening upon line thicknesses in each technique, as this may have biased them in favor of a particular feedback response when they might otherwise not have recognized that factor or deemed it sufficiently important to analyze.

Subjects all fell into a single group, with each individual evaluating both techniques jointly in order to provide each element of feedback. For this reason, the primary (ideally the sole) visual effect of the differences between the two techniques – that is, the presence or lack of perspective-based influence on line thicknesses – was not to be regarded as a grouping variable like that found in an experimental study.

3.3 Population and Sample

The population of inference consisted of all potential viewers of non-photorealistic renders that incorporated inter-color lines of varying thickness. Thus, this population had to be assumed to include any member of the public, given that, for example, NPR shaders of this type could eventually be used in a video game playable by individuals of all ages and demographics.

In view of this fact, the sampling procedure, while it was intended to approximate a simple random sample, necessarily fell short of the ideal due to limits of feasibility. In reality, a sample of convenience was used, in which most participants were Purdue Polytechnic Institute students, but in which participants could also be students in other Purdue University colleges, Purdue faculty or staff, or even individuals outside the university. Participants were required to be at least 18 years old, but recruitment occurred entirely independently of any demographic factors other than age, such as sex or ethnicity. Order of participation followed a “first come, first served” model. While in theory stratification of the sample based on artistic experience would have been desirable, it is unclear how such experience could have been measured. Moreover, because users were not sorted into groups, no need existed for vigilance as to equal representation in this area across subsets of users.

Subjects were originally to be excluded from participation if they had a visual impairment (i.e., colorblindness) that would potentially skew their perceptions of colors and thus of the relationships between base color differences

and line thicknesses, but this restriction was later deemed irrelevant and dropped, in view of the fact that color-thickness relationships were ideally identical across the two techniques, the fact that colorblindness would not affect perception of the relevant influence of perspective foreshortening, and the fact that the output colors and degrees of shading were already imprecise in the output renders, due in part to a program bug possibly related to limits on available color depths of textures (see Section 3.7 and the Appendix).

By conducting power analyses (see Section 3.6), the researcher determined that a sample size of 66 would permit an inference that the population proportion in favor of one technique would exceed the population proportion in favor of the other in a case where one sample proportion would exceed a threshold of 0.67. Similarly, a sample size of 47 would permit a threshold of 0.70. Given the researcher's desire to weigh feasibility concerns regarding sample size against extremity of the required threshold (a smaller sample size would have been associated with a higher threshold), the initial plan was to target a minimum sample size of 66 (threshold = 0.67), and if that were to prove infeasible, the backup plan was to reduce the targeted minimum sample size to 47 (threshold = 0.70). In the course of the testing phase of the project, the researcher eventually decided to target the more modest minimum figure of 47.

Recruitment of testers took place primarily through word of mouth. The researcher asked students and faculty in Knoy Hall (the primary location of the Purdue Polytechnic Institute) if they were interested in participation and handed out flyers that briefly summarized the study. Some non-student participants, initially contacted outside of Knoy Hall, came to the testing site as well. In addition, some faculty members spread the information in their classrooms, and two volunteered of their own accord to offer extra credit for participation. Secondary recruitment methods included a listserv-distributed email and copies of the aforementioned flyers hung at a few locations in Knoy Hall.

3.4 Testing Application

The real-time rendering application used for testing was assembled in the Unity 3D game engine. It utilized scripts written in C# and JavaScript as well as shaders written in the CG shader language.

It divided the screen into four quadrants – three viewports containing real-time renders from identical camera views and one quadrant containing written instructions for subjects. The first viewport, occupying the upper left quadrant, showed the scene rendered normally, without NPR lines of any sort. The other two viewports, located in the bottom quadrants, showed the scene rendered via the two NPR techniques (baked-lines and dynamic). As mentioned previously, the application ordered the two NPR viewports randomly so as to avoid any potential user biases involving left-right positioning. The instructions contained in the upper-right quadrant changed with each session phase. See Figures 3.2 through 3.7 for screenshots of the application in each of its various session phases, which will be described in the next section.

The output of each NPR technique was generated via an ordinary rendering pass and two subsequent image-processing passes. The image-processing passes for both techniques generated silhouette and contour lines that varied in thickness according to depth, and the passes for the dynamic method added inter-color lines that varied with color contrast and depth as well. Equation 3.1 is the $\Delta E''_{\text{RGB}}$ color-difference equation used in the dynamic method; it is based on that presented by Mokrzycki and Tatol (2011). The Appendix contains detailed descriptions of the rendering passes.

$$\Delta E'' = \frac{\sqrt{(2 + R_{\text{AVG}})\Delta R^2 + 4\Delta G^2 + (3 - R_{\text{AVG}})\Delta B^2}}{3} \quad (3.1)$$

Directly prior to the initial color-map rendering for each viewport on a given frame, each of the scene’s key objects was assigned one of two textures; each texture corresponded to one of the NPR styles. The textures in each pair were identical in resolution, but one had black lines baked into it via an external program that used a

similar line-generating algorithm to that employed in the dynamic method. See Figure 3.1 for an example of such a pair of textures.

None of the textures were mipmapped, as mipmapping would have caused unwanted blurring of inter-color edges in certain areas, and as the two NPR techniques were meant to look as visually similar as possible in terms of line quality. However, one effect of the lack of mipmapping was that baked lines would lose coherence (i.e., “disintegrate” into disconnected black pixels) at large distances from the camera. However, as mentioned previously, subjects were all verbally instructed by the researcher to solely consider differences in line thicknesses when making their judgments, ignoring any other differences that they would potentially notice. (This expectation was also conveyed in the “In terms of line thicknesses...” clause at the beginning of each of the user-feedback prompts discussed in Section 3.2.)

Information on the application’s session-phase functionality and data-recording mechanisms will be included in the next section. Certain technical flaws in the application that may have constituted threats to validity will be documented in Section 3.7.

3.5 Procedure and Data Collection

All data was collected via the testing application itself. During the interactive phase of each session, user-induced camera movements would be recorded at a very rough sampling rate, so that details of each user’s exploration choices could be examined at a later date or possibly played back via a future extension of the application. At the end of each session, the application prompted and accepted each user’s binary feedback on each variable of interest (i.e., adherence to artistic convention and aesthetic appeal) via its user interface. The camera-movement data and the final feedback for any given user were stored anonymously in a text file at the conclusion of his or her session.

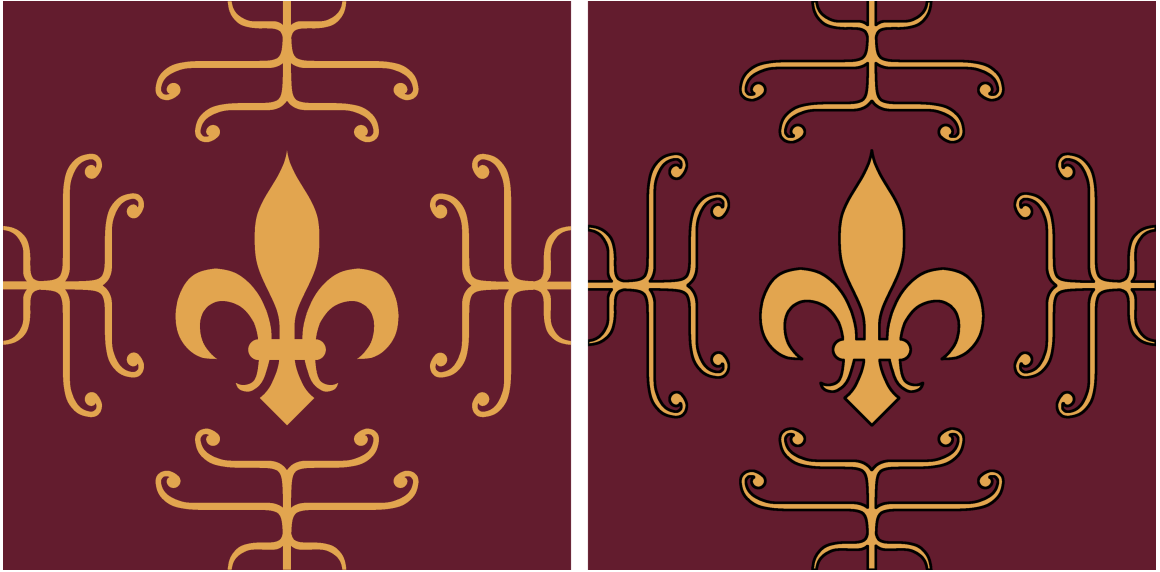


Figure 3.1. The two textures used for a pillow object in the 3D scene are juxtaposed for comparison. One features baked lines that were automatically generated by an external program, and the other does not.

Each subject would thoroughly examine the scene via the two bottom viewports. The “default” render with no superimposed lines was simply intended as a visual reference, which subjects could use to better understand detailed shapes, color relationships, and the precise nature of the visual additions provided by the NPR techniques. Users would focus most of their attention upon the NPR viewports in the bottom two quadrants. Given their immediate, horizontal juxtaposition, the subjects could easily compare the results of each in terms of inter-color line thicknesses.

Prior to testing, subjects were explicitly informed of the variables of interest, and they were asked to examine the differences in line thicknesses between the two NPR techniques in order to reach their overall conclusions concerning adherence to artistic conventions and aesthetic appeal. As mentioned previously, they were instructed to ignore any accidental differences between the implemented techniques

in aspects other than line thickness (e.g., in line quality or in line sharpness), so that such differences, being irrelevant to the study's focus, would not confound the results. The researcher also informed them in advance of the upcoming phases in the session – that is, two automatic flythroughs, a phase of interactive scene exploration, and the eventual contribution of feedback on the variables of interest. As the subjects received this initial verbal instruction, they were presented with an initial set of onscreen instructions along with an initial static view of the scene in each viewport, as shown in Figure 3.2.



Figure 3.2. This screenshot shows the initial view presented to subjects in the application. The upper-right area contains initial instructions.

In the first phase, a subject would watch two automatic flythroughs of a pre-built scene. The scene, a virtual living room, contained 26 textured objects of varied shapes, texture detail levels, color combinations, and color pattern types. The textures of all of said objects featured areas of solid, uniform color bordered by sharp edges. The second flythrough, identical to the first, was intended to help

viewers to avoid missing details that would potentially pass too quickly for them to analyze the first time. See Figure 3.3 for a screenshot of the automatic phase.

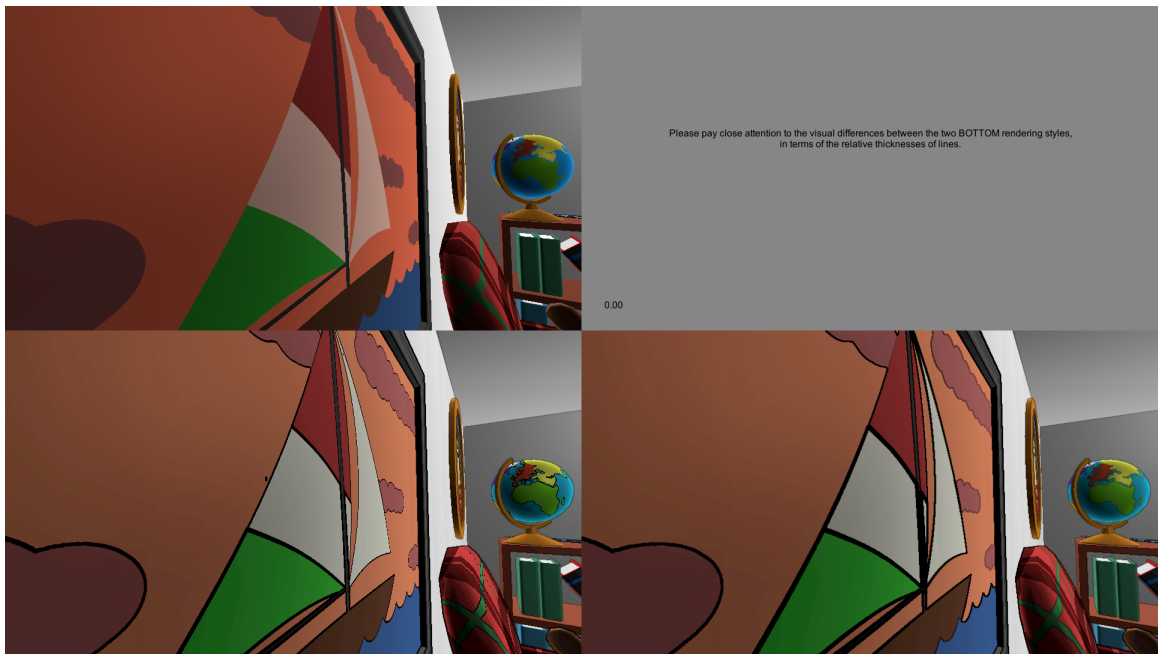


Figure 3.3. This screenshot shows the automatic flythrough phase. The upper-right area contains the instructions displayed to subjects.

After the flythroughs, subjects would interactively explore the scene in at least one of two modes: Controlled-Flythrough Mode (see Figure 3.4), in which they could use the keyboard to move the camera along the flythrough path at will and stop wherever they wished to examine differences, and Free-Camera Mode (see Figure 3.5), in which they could move the mouse to rotate the camera and the keys to translate it relative to its view direction. In this latter mode, the subjects could view any object in the room from largely arbitrary angles and distances, although colliders placed on objects in the room limited the camera's positioning to a degree. During this phase of exploration, the camera's positions and rotations over time were sampled behind the scenes at a rate of 2 frames per second and logged. Key presses and mouse movements were also counted, as each user was required to reach

a minimum of 100 key presses (collectively in both Controlled-Flythrough Mode and Free-Camera Mode) and 1000 frames that contained mouse movement (in Free-Camera Mode only) in order to proceed to the final phase. The researcher implemented this feature in order to ensure that each subject would inform his or her final judgments through at least a baseline amount of exploration and comparisons of line thicknesses.



Figure 3.4. This screenshot shows Controlled-Flythrough Mode. The upper-right area contains the instructions displayed to subjects.

Once a subject had explored enough to meet these minima, the program would advance to the final phase (see Figures 3.6 and 3.7), in which the user would express his or her preferences between the techniques via two pairs of radio buttons (each pair offered the choices “Bottom Left” and “Bottom Right”). At this point, the subject would still have the option to explore the scene in the two interactive modes as before, but the interface for the final selections would continue to appear in the upper right quarter of the screen in Controlled-Flythrough Mode. Finally, the



Figure 3.5. This screenshot shows Free-Camera Mode. The upper-right area contains the instructions displayed to subjects.

subject would click a “Submit” button to enter his or her selections, and the program would write the logged exploration information mentioned above, along with the user’s final feedback (expressed as “DYNAMIC” or “BAKED” for each variable of interest rather than as “Bottom Left” or “Bottom Right”), to a text file named with a random number (e.g., “98897121.txt”). See Figure 3.8 for an abridged example of the content of one of the text files.

3.6 Statistical Analysis

For each variable of interest, the researcher employed a hypothesis test for a single proportion of a single sample in order to make the necessary inference. A normal distribution was assumed for the data, as such an assumption is required for a test of this type. In a prior stage of the project, the researcher conducted power



Figure 3.6. This screenshot shows the final phase when the user is in Free-Camera Mode. The instructions tell the subject to switch to Controlled-Flythrough Mode to make the final selections, as there is no onscreen mouse cursor in Free-Camera Mode.

analyses corresponding to this type of test; each used a power of 0.80 and a significance level (α) of 0.05 in order to determine an appropriate sample size and the threshold for the sample proportion associated with that size. As stated previously, the targeted minimum sample size became 47 in the end. Thus, if one sample proportion would exceed a threshold of 0.70, the corresponding population proportion would be taken to be larger than 0.5.

After the collection of data, hypothesis tests were conducted, each of which utilized the sample size, a sample proportion, and the hypothesized population proportion of 0.5 to calculate a test statistic (z) and subsequently a p-value. If the chosen significance level of 0.05 would exceed this p-value, the corresponding null hypothesis would be rejected. Although the threshold to be exceeded was already known from the power analyses, the researcher would still report the p-value for

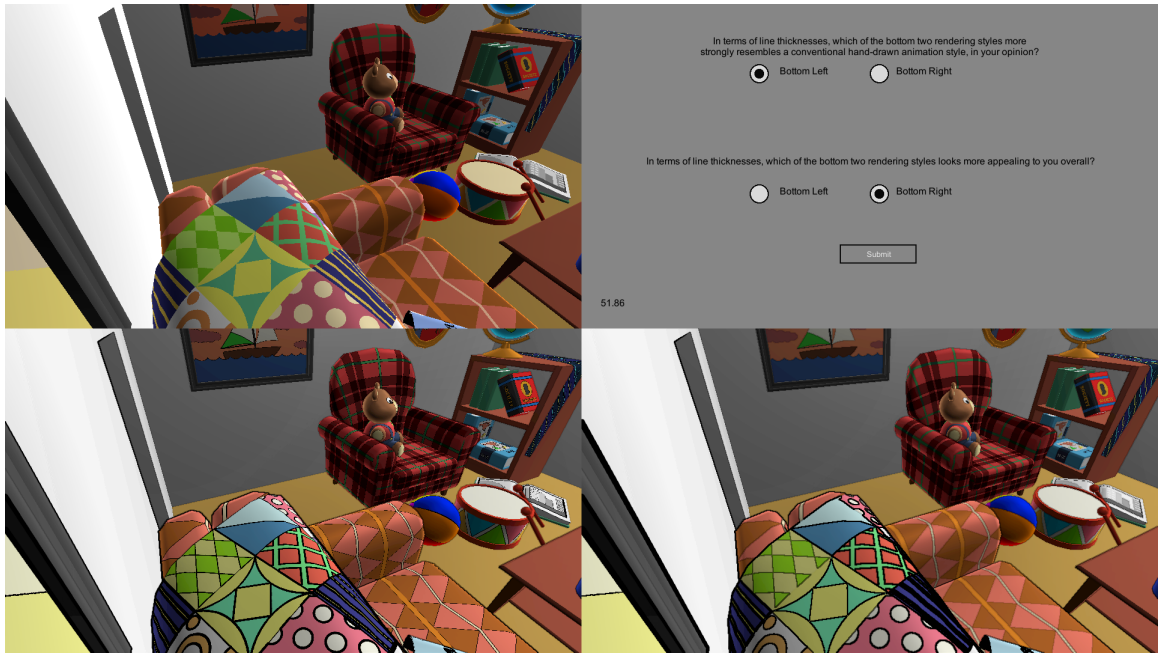


Figure 3.7. This screenshot shows the final phase when the user is in Controlled-Flythrough Mode. The instructional area now contains the final questions, the radio buttons, and the “Submit” button.

each variable so as to provide further insight into the degrees of significance of the results. These tests, like the prior power analyses, were carried out via the MiniTab statistical tool.

The logged camera-movement data from the subjects was not analyzed in this study; it was simply used for measuring overall times for interactive phases of sessions, for reasons explained in Section 4.4. However, these details were preserved for use in potential future research. Analysis of this largely qualitative data was not regarded as critical in the present study, given the particulars of the methodology. In a study involving remote testing locations at which the researcher would not have been physically present, such details would have been critical in the analysis. However, in the context of the study at hand, in which the researcher was present in person for the sessions and able to observe subject progression and interaction

```

1 1.844017 2.205484 -0.864455
2 3.975474 173.2424 -0.0001969218
3 -----
4 -----
5 -----
6 -----
7 1.832503 2.209773 -0.7614555
8 1.241929 173.9215 -0.0001289168
9 -----
10 -----
11 -----
12 -----
13 -----
14 -----
15 -----
16 -----
17 -----
18 -----
19 1.823198 2.208533 -0.6637232
20 357.8814 175.0498 -4.548788E-05
21 -----
22 -----
23 -----
24 -----
25 -----
26 -----
27 -----
28 -----
29 -----
30 -----
31 -----
32 -----
33 -----
34 1.837971 2.208307 -0.8118322
35 2.62067 173.5352 -0.0001645156
36 -----
37 -----
38 -----
39 -----

```

```

879 -----
880 -----
881 -----
882 -----
883 1.046003 0.5667237 -1.75396
884 17.84999 303.3 0
885 -----
886 -----
887 4.949993 296.4 -2.142425E-07
888 -----
889 1.045994 0.5667236 -1.753969
890 8.309994 289.05 -4.31516E-07
891 -----
892 -----
893 7.799995 276.75 -8.617466E-07
894 -----
895 0.7510519 0.5178356 -1.729874
896 10.49999 273.6 0.683136E-07
897 -----
898 0.5545775 0.4806678 -1.72818
899 10.34999 263.55 -4.339478E-07
900 -----
901 -----
902 11.24999 265.6501 4.3525E-07
903 -----
904 0.4568526 0.4606451 -1.735365
905 11.55001 265.7998 -2.483546E-05
906 -----
907 0.2027937 0.4085866 -1.754028
908 11.55003 265.7995 -5.010663E-05
909 -----
910 0.1050736 0.388564 -1.761209
911 11.55004 265.7993 -6.448506E-05
912 -----
913 -0.09033144 0.3468796 -1.766542
914 12.45 273.3 -2.185835E-06
915 -----
916 -0.0903405 0.3468794 -1.766547
917 13.64999 273.6 0.78589E-07

```

```

2159 -----
2160 -----
2161 -----
2162 -----
2163 -----
2164 -----
2165 -----
2166 -----
2167 -----
2168 -----
2169 -----
2170 -----
2171 -----
2172 -----
2173 -----
2174 -----
2175 -----
2176 -----
2177 -----
2178 -----
2179 -----
2180 -----
2181 -----
2182 -----
2183 -----
2184 -----
2185 -----
2186 -----
2187 -----
2188 -----
2189 -----
2190 -----
2191 -----
2192 -----
2193 -----
2194 Convention: DYNAMIC
2195 Aesthetics: BAKED
2196 -----
2197 -----

```

Figure 3.8. An example results file, “98897121.txt”, is shown. In each two-line block, the upper set of values, if present, are a set of position values for the camera, and the lower set, if present, are a set of Euler rotation values. These were recorded every 0.5 second during the interactive phase of a session. A line of dashes in place of a set of three values indicates that the position/rotation had not changed from the prior corresponding value in the last 0.5 second.

strategies, the detailed exploration data arguably constituted surplus information in many respects.

3.7 Threats to Validity

Some internal threats to validity were apparent. For practical reasons, the sample was not truly random. Persons younger than 18 were absent from it, even though they were included in the population of inference. In addition, because people in the Purdue Polytechnic Institute were inevitably overrepresented given the recruitment methods, the sample may have contained a disproportionate number of

individuals who had artistic experience and thus were more likely than the average person to evaluate line thicknesses in strict accordance with artistic conventions. People with artistic experience may also have been more likely to recognize the relationship between perspective foreshortening and output line thicknesses, and they may have made more educated or reliable judgments regarding the aesthetic merits of each technique.

It was possible that certain testers may have been inclined to make rushed final decisions regarding their technique preferences, or they may have devoted insufficient attention to the observation and comparison of line thicknesses during their sessions. Moreover, it was not guaranteed that all of them would explore the scene sufficiently to be able to make a properly informed judgment. While the aforementioned input “quotas” (a minimum of 100 key presses and 1000 frames of mouse movement in the interactive phase) were established to combat this threat by forcing a certain amount of exploration, participants were able to partially skip these requirements by moving the mouse and tapping the keys aimlessly. Additionally, even if participants did pass the input requirements normally, it was possible that those relatively unfamiliar with the mouse/keyboard control scheme would spend a significant portion of the intended exploration time simply mastering the controls and inducing the camera to cooperate.

Certain technical issues with the testing application appeared to threaten the validity of the study as well. In the first place, the only differences in line thicknesses between the NPR techniques were intended to stem from the presence or lack of perspective foreshortening, as this variable was intended to be the only one affecting each style’s perceived adherence to artistic convention and aesthetic appeal in the context of this study. However, in reality, not all line thicknesses appeared the same between the NPR techniques, even from view angles from which they were intended to appear identical. The most likely primary explanation for this flaw was that the rendered texture colors on certain objects, even prior to lighting, were not always identical to those found in the source image files, on account of color-depth

limitations associated with texture storage in Unity 3D. If the rendered base colors were not identical to the source colors, the dynamic method would not calculate color differences (and thus line thicknesses) of the same magnitude as those calculated externally for the baked-lines textures, as the latter calculations had been performed directly upon the source images. Figure 3.9 displays a particularly notable example of this phenomenon that involved grayscale colors.

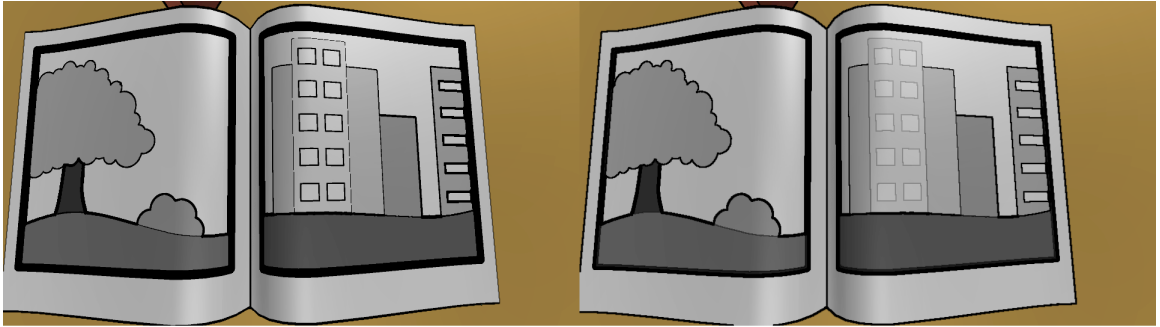


Figure 3.9. Some of the line thicknesses differ between the NPR techniques in this image, even though they were meant to appear equal. The grayscale color values in this book’s texture may not have been accurately preserved when the texture was loaded into Unity 3D.

A similar issue involving unwanted influences on line thicknesses was the stretching of certain baked lines in textures that were not uniformly UV-mapped onto objects. As shown in Figure 3.10, a ball object in the scene used a rectangular texture that was drawn together horizontally at the top and bottom, so that the baked lines on the left were thinner near the purple “end caps” than elsewhere. In contrast, the dynamically-rendered lines on the right exhibited no such effect. Figure 3.10 also shows a stuffed bear object of irregular shape that required complex UV unwrapping. Certain lines in the left (baked-lines) render had slightly incorrect or varying widths due to inconsistent UV spacing on different areas of the bear model’s surface; such inconsistency of spacing was not matched by

corresponding variation in the auto-generated lines baked into the texture files, which only varied with color-difference values.

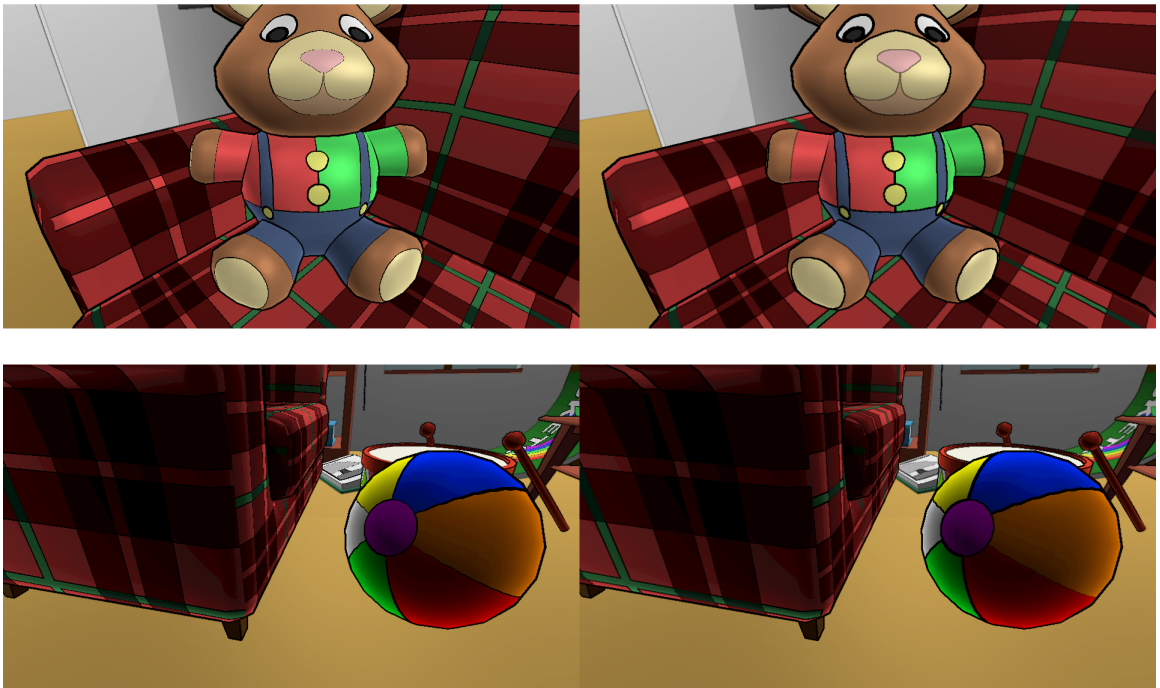


Figure 3.10. In the baked-lines renders on the left side, some unwanted line-thickness variations were present on account of inconsistent UV spacing.

Another potential obstacle to subject interpretation of line thicknesses as intended lay in the manner in which the dynamic method handled the rendering of thin lines (i.e., either distant lines or lines between similar colors). As explained and illustrated in the Appendix, the dynamic method used partially-darkened pixels to represent lines less than two pixels thick, because it could not actually generate inter-color lines that thin. While this partial darkening arguably created the illusion of thinner lines, at least at a distance, it was possible that viewers would not interpret the effect in this manner, perhaps instead seeing such lines as thicker than intended (i.e., as possessing their actual thickness of two pixels), or even not viewing

them as full-fledged lines. Such interpretations had the potential to confound the results.

A final concern regarding subject interpretation involved not a technical issue but the omission of a feature for controlling global line-thickness scaling. With its provision of only one fixed thickness for any given line viewed from a given distance and angle, the testing application had the potential to make certain dynamically-generated lines look excessively and unavoidably thick from certain views (see Figure 3.11), to the point where subjects could potentially just regard the dynamic-method lines as thick in an absolute sense. Thus there was a risk that viewers would merely see a “thin versus thick” difference between techniques (at least if they generalized from certain camera views) and thus conclude that the dynamic method was either more or less conventional or appealing purely because it used thicker lines. In contrast, with a global scaling option, extremes of thickness could have been less of a distraction, as testers could have opted to reduce overall line-thickness scaling for views such as those in Figure 3.11. However, such a feature would have required the program to include and switch between even more textures, as multiple textures with differently-scaled baked-line thicknesses would have been required for each object.

3.8 Summary

This chapter has explained the methodology and framework for this study in terms of its goal, hypotheses, variables, population, sampling, testing application, testing procedures, data collection methods, statistical analysis, and threats to validity. The next chapter will present data collected from the subjects in the sample via the above procedures, the results of the statistical analysis of said data, observations regarding the actual sample’s demographics, and notes regarding issues or disparities that arose in the testing phase.

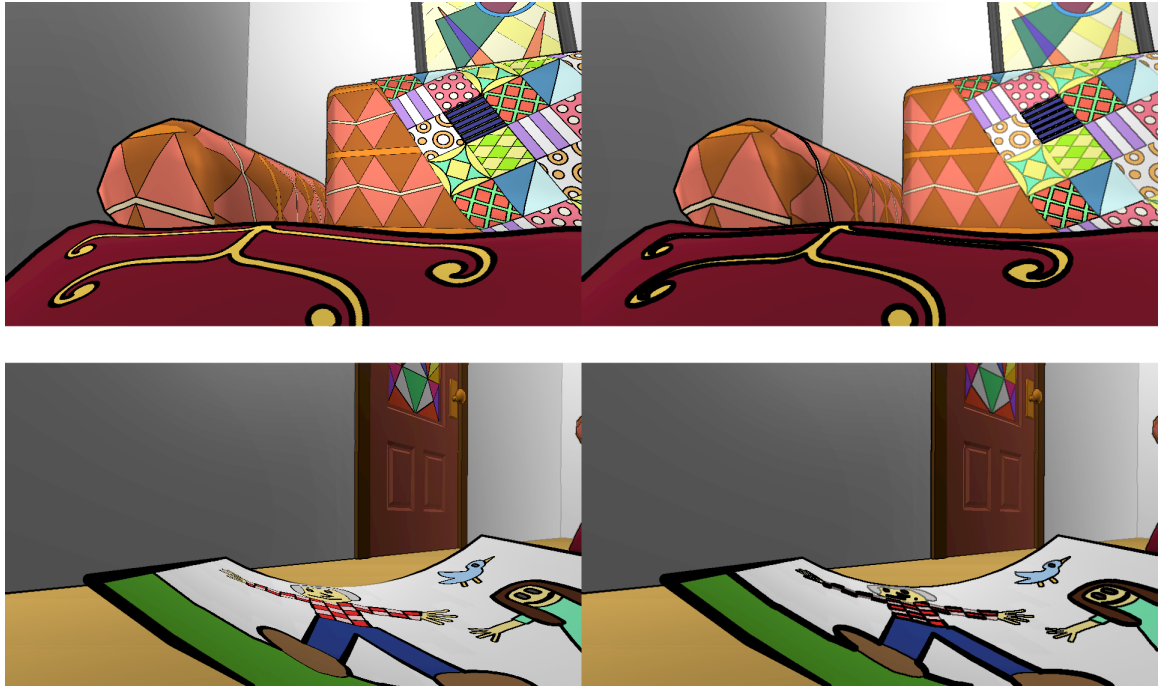


Figure 3.11. These renders were generated from a camera view on the default flythrough path. In the dynamic-method renders on the right, certain lines appear notably thick, particularly in view of the extreme view angles on the corresponding surfaces. It was possible that this could unintentionally mislead viewers into thinking that the dynamic technique was merely distinguished by overall thicker lines.

CHAPTER 4. RESULTS

This chapter will lay out the details and outcomes of the testing phase. First the numbers will be presented – namely, the sample size and the sample proportions associated with each technique in terms of each variable. Next the results of the statistical analysis will be presented, and information will follow regarding the sample demographics, the problems encountered, and the steps taken to mitigate those problems.

4.1 Statistics

The testing phase of the study took place over ten days. 49 subjects participated in the study – slightly more than the targeted minimum of 47 that would have permitted a threshold of 0.70 for the two-tailed, single-proportion hypothesis tests. In the end, given the sample size of 49, a power of 0.80, and an α of 0.05, a power analysis revealed that 0.695336 was the actual threshold for a sample proportion beyond which the corresponding population proportion would have been inferred to be larger than 0.5. This is indicated in the MiniTab output in Figure 4.1.

Of the 49 subjects, only 19 regarded the dynamic method as more adherent to artistic conventions. However, 28 of them regarded the dynamic method as more visually appealing. See the “ALL RESULTS” section of Table 4.1 for the overall totals. For each variable of interest, a two-tailed, single-proportion hypothesis test was carried out on the results. Figure 4.2 provides MiniTab output displaying the results of both tests. In terms of perceived adherence to artistic convention, 38.7755% of subjects favored the dynamic method and 61.2245% favored the baked-lines method; neither of these proportions exceeded the 69.5336% required for null-hypothesis rejection. The p-value from the z-test – 0.152, well above the α of

Power and Sample Size

Test for One Proportion

Testing $p = 0.5$ (versus $\neq 0.5$)
 $\alpha = 0.05$

Sample			
Size	Power	Comparison	p
49	0.8		0.695336

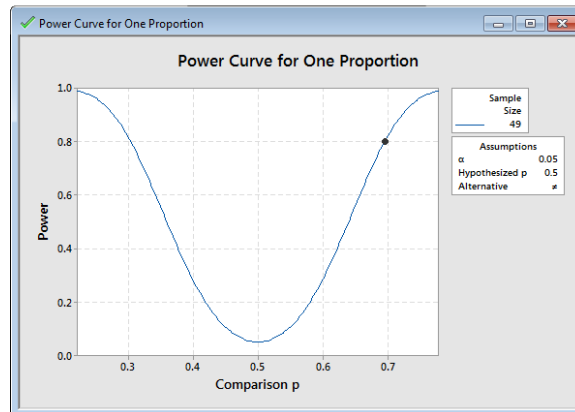


Figure 4.1. This MiniTab output, from a two-tailed power analysis starting from an α of 0.05, a power of 0.80, and a sample size of 49, indicates a threshold value of 0.695336.

0.05 – confirmed that the null hypothesis for the first variable could not be rejected. In terms of aesthetic appeal, 57.1429% of subjects favored the dynamic method and 42.8571% favored the baked-lines method. As each of these proportions also failed to exceed the required 69.5336%, and as the p-value of 0.392 from this second z-test greatly exceeded the α , the null hypothesis for the second variable could not be rejected either.

Therefore, it could not be inferred that the population proportions were significantly different with regard to either perceived adherence to artistic conventions or aesthetic appeal.

As mentioned previously, the logged camera-movement information was not subjected to any statistical analysis. However, it remains available to future

ALL RESULTS		
	Dynamic	Baked
Convention	19	30
Aesthetics	28	21
TEN LONGEST TIMES		
	Dynamic	Baked
Convention	3	7
Aesthetics	6	4
TEN SHORTEST TIMES		
	Dynamic	Baked
Convention	3	7
Aesthetics	5	5

Table 4.1

This table sums up the choices of subjects between the NPR techniques with respect to each variable of interest. The middle and bottom sections tally the choices provided by the ten subjects with the longest interactive-phase times and the ten with the shortest times, respectively.

Test and CI for One Proportion: Convention

Test of $p = 0.5$ vs $p \neq 0.5$

Event = DYNAMIC

Variable	X	N	Sample p	95% CI	Exact P-Value
Convention	19	49	0.387755	(0.251974, 0.537615)	0.152

Test and CI for One Proportion: Aesthetics

Test of $p = 0.5$ vs $p \neq 0.5$

Event = DYNAMIC

Variable	X	N	Sample p	95% CI	Exact P-Value
Aesthetics	28	49	0.571429	(0.422104, 0.711805)	0.392

Figure 4.2. This MiniTab output displays the results of two-tailed, single-proportion hypothesis tests that were carried out on each variable of interest. Each used an α of 0.05 and a power of 0.80.

researchers upon request, along with the associated user feedback regarding preferences.

4.2 Demographics

As expected, given the methods of recruitment, a large majority of the subjects were students at Purdue University; many were graduate students and many were undergraduates. Because they were primarily recruited on the third floor of Knoy Hall, the central location of the Purdue Polytechnic Institute, most of them were likely students in this college, and many were students in the Department of Computer Graphics Technology. No faculty participated as subjects. However, at least three testers were non-students whom the researcher initially invited to the

building for testing. While all testers were over 18 years of age, with a vast majority seeming to be college-aged, the actual range of ages appeared to span multiple decades.

4.3 Problems Encountered

In the course of the study’s testing phase, a number of issues, both expected and unexpected, manifested themselves. For instance, a small minority of subjects struggled to master the camera controls for a time, although many others immediately understood them. It is possible that prior experience in playing first-person computer games aided some subjects in adjusting to the controls, while a lack of experience in playing such games, or in playing video games in general, may have been linked to greater difficulty in adjustment.

Some subjects offered voluntary and unsolicited verbal explanations of their choices (not formally recorded, as no mechanism existed for doing so that had been approved for the study), which helped to reveal or highlight some of the other methodological concerns laid out in Section 3.7. Such concerns included those involving the partial darkening of line pixels in the dynamic method, those involving the lack of an option for global line-thickness scaling, and those involving line-thickness differences between techniques that arose from factors other than perspective foreshortening (e.g., non-uniform UV mapping or inaccurate texture-color storage in the program).

At certain points, in the course of verbally describing expectations for subjects, the researcher used the adjective “perceived” to describe the line-thickness differences that subjects were intended to analyze. This term was primarily intended to hint to subjects that partially-blended lines in the output of the dynamic method (see Section 3.7 and the Appendix) were meant to be considered “thin” for comparison purposes, as the partial darkening of their pixels was meant to simulate thinness to human perception. However, the researcher failed to

consistently apply this term; indeed, the idea of using it only arose at an intermediate point in the testing phase of the study.

Notably, some testers bypassed some of the minimum exploration requirements for advancement to the final phase. Near the end of the interactive phases of their sessions, they moved the mouse and pressed the keys aimlessly until they reached the minimum input counts. However, this was known at the time to the researcher, who personally observed the sessions and judged this action to be warranted in all of those particular cases, even encouraging it at times. For example, on the first day of testing, an earlier build of the application instead required 2000 frames containing mouse movement instead of 1000 in order for testers to progress to the final phase; this proved to be an unexpectedly excessive requirement, given that progress toward it was limited by framerate and given that the framerate on the testing machine was much lower than that on the development machine. In order to prevent certain sessions on the first day from growing too long, the researcher permitted the bypasses. Additionally, on various days of testing some subjects were permitted to bypass the requirements because they had spent much of their interactive session time in Controlled-Flythrough Mode and had thus accumulated less average recorded mouse movement per unit time than other subjects had (mouse movement was only recorded in Free-Camera Mode). In short, when the researcher subjectively judged that testers had sufficiently examined the various line-thickness differences but were being held back by unnecessarily burdensome input requirements in their cases, they were allowed to thus skip the remaining requirements.

For the purpose of gaining insight into whether this permitting of input-requirement bypasses affected the study's results, the times taken by the various subjects in testing were examined via the camera-movement logs in the results files, and the results of the ten testing sessions with the longest interactive phases (ranging from 6.625 minutes to 12.058 minutes) were isolated, as were those of the ten sessions with the shortest times (ranging from 2.692 minutes to 4.075

minutes). It was assumed to be likely that longer interactive-phase times – technically measured from the first recorded user input until the clicking of “Submit” – corresponded to a greater degree of exploration, while shorter times corresponded to a lesser degree. However, the proportions exhibited in the longest-session results and in the shortest-session results were reasonably similar to those in the overall results as well as to each other, as the second two sections of Table 4.1 indicate. 30% of the ten users with the longest interactive sessions considered the dynamic method to more closely imitate conventional hand-drawn styles, and the same percentage of the ten with the shortest sessions did as well. 60% of those with the longest sessions found the dynamic method more visually appealing, whereas 50% of those with the shortest sessions did. These results did not conflict with the overall findings – that is, that the differences in proportions were not statistically significant but that larger sample proportions preferred the baked-lines method for adherence to convention and the dynamic method for aesthetic appeal.

Another disparity between testing sessions lay in the fact that some subjects asked many more questions than others did, mainly in order to receive clarifications. However, any information that was provided to any given tester would have been offered to any other testers upon request as well. Likewise, any information withheld from any testers prior to the conclusions of their sessions – for instance, an early revelation of the fact that one method had lines baked into textures and that the other did not, or an explicit mention of the influence of perspective foreshortening on line thicknesses in the baked-lines method – was consistently withheld from all other testers as well.

4.4 Summary

This chapter revealed the figures from testing and analyzed them statistically, with no commentary upon their broader implications, and it described the makeup

of the sample and the issues encountered. In contrast, the next and final chapter will feature conclusions and speculation concerning said analysis, expand upon the nature and relevance of the problems encountered in testing, and recommend paths for future research that were inspired in part by the existence of those issues.

CHAPTER 5. CONCLUSIONS AND RECOMMENDATIONS

This chapter will contain the general inferences drawn from this study's results as well as the speculations that they prompted – particularly the researcher's ideas regarding the potential causes of any surprising outcomes. It will then delve into the nature and likely importance of certain problems encountered during testing, and it will conclude with recommendations of future work motivated by these thoughts.

5.1 Interpretations of Findings

The statistical analysis failed to provide evidence that the general population preferred either NPR technique in terms of either perceived adherence to artistic conventions or perceived aesthetic appeal. While this did not positively imply that the techniques were regarded equally in those respects, it permitted that as a clear possibility and did not provide any reason for belief that a significant computation/aesthetics trade-off existed between the techniques. Whereas the baked-lines method had been known to be more computationally efficient from the start, the dynamic method could not be expected, given the results of this study, to counter that with superiority of its own in another respect. Thus, the outcome of this study appeared to suggest that the baked-lines technique was the more recommendable of the two after all.

Such an outcome appeared reasonable even in view of the researcher's initial expectations. In theory, as explained in the ponderings within this study's literature review, the lack of influence of perspective foreshortening on dynamic inter-color lines would seem to conform more accurately to artistic convention than the presence of such an influence would, and it appears reasonable to consider perceived

adherence to artistic convention as a factor directly influencing perceived aesthetic appeal, albeit obviously not fully determining it. However, the possibility had always clearly existed that observers – particularly ones who had not developed a keen eye for certain aspects of graphical detail – would simply not have taken notice of the perspective-based nature of the key line-thickness differences between techniques, as had the possibility that they would not have recognized the “decal effect” of perspective foreshortening on baked inter-color lines to be an element theoretically alien to traditional line-art styles. It is possible that such viewer oversights contributed to the ambiguous comparative standings of the two methods with regard to the variables of interest.

However, the proportions within the sample, while not significantly different in a statistical sense, were rather puzzling in terms of their relative sizes. In terms of aesthetic appeal, testers overall preferred the technique that they had been expected to prefer – that is, the dynamic one. However, in terms of adherence to artistic conventions, they unexpectedly tended to prefer the baked-lines method. The researcher had initially guessed that the dynamic method would be found more appealing, for the primary reason that it would be considered more adherent to artistic conventions, but clearly the notion of such a causation was undermined by the unexpected figures concerning the latter variable. Thus it was unclear what factors, if any, primarily motivated the overall aesthetic-appeal preference, particularly in view of the inherent subjectivity of such a variable. It was possible that the unwanted factors that influenced line-thickness differences in the testing program had some effect on this. It was also conceivable that some testers had ignored the directions that they only devote attention to line thicknesses and had instead expressed a preference for the relatively “smooth” appearance of the dynamic lines, which featured the antialiasing-like effects of partial pixel darkening. However, a highly likely explanation is simply that the difference in the sample proportions was random, particularly given the established lack of statistical significance.

The preference among subjects for the baked-lines method in terms of perceived adherence to artistic conventions was perhaps even harder to explain, although again randomness was distinctly probable as a cause. Another potential factor was the aforementioned risk that some subjects would just regard the two styles as a “thinner-lined style” and a “thicker-lined style”, the latter of which had certain lines that appeared too thick from certain angles to fit their mental image of a conventional hand-drawn style. Moreover, some individuals may have mentally associated hand-drawn line art with a high degree of thickness variation between lines in the same render, as well as with variation along individual lines (i.e., tapering). Given that the baked lines tended to exhibit more of such variations (ones bound to curved surfaces were particularly subject to smooth variation of perspective foreshortening), subjects may have judged the baked-lines method to look more “conventional” for that reason.

5.2 Discussion of Problems Encountered

The problems that occurred during testing, documented in Section 4.4 above, either confirmed existing theoretical concerns regarding methodological validity or initially drew the researcher’s attention to ones not clearly foreseen. In the case of the input-requirement bypasses, however, the concern that some users may have been suboptimally informed on the differences between techniques due to insufficient scene exploration was arguably satisfactorily addressed by the outcome of the examinations of the ten longest and ten shortest sessions. The similarities of the proportions between both extremes, as well as between each extreme and the aggregate, suggested a lack of a clear cause for worry that variations in scene-exploration times had noticeably confounded the study’s outcomes.

On the other hand, the fact that some subjects had asked a relatively high number of questions did indicate a possibility that they had achieved superior understanding and insights in the end, relative to subjects who had simply followed

instructions and kept any uncertainties regarding the researcher's expectations to themselves. This fact, combined with the researcher's inconsistent application of the "perceived" descriptor to the relative line-thickness differences intended for observation (see Section 4.3), may have contributed to an unintentionally uneven distribution of information among subjects. Such a disparity can be difficult to fully avoid, but its possibility is critical to document, particularly given that future research would benefit from said documentation.

5.3 Future Research

Multiple varieties of future studies could use this one as a launching point. Given the unexpected sample proportions related to adherence to artistic conventions, searches for alternate theories regarding linkages between the inter-technique differences and the variables of interest may be in order. One potential method for gaining new insights into the true nature of these linkages would be to provide test subjects with open-ended survey questions regarding their specific mental conceptions of conventional hand-drawn line art and regarding the essential visual elements thereof. Such information, coupled with the results of this study or of a future modification of it, may assist in explaining the surprising sample proportions, if indeed they are not simply explained by randomness in view of their statistical insignificance.

Related studies could also be carried out that would have areas of focus that were excluded by this project's delimitations. For instance, a future study could focus on performance optimization of the baked-lines technique presented here, or even of both techniques (perhaps with the purpose of closing the computational gap as much as possible and thereby reducing the relevance of any trade-off that may exist). Comparative studies similar to this one could be carried out on NPR techniques that use light-blended colors to generate inter-color lines rather than base colors (while the latter seem more generally desirable for color-difference

calculations, techniques incorporating the former would still expand the pool of styles available to artists).

If researchers should desire in the future to conduct similar studies in order to re-examine the same research questions from this one, they might be well advised to incorporate stronger controls for confounding variables than those offered in this study. As explained in Section 3.7, options for adjustment of global line-thickness scaling could mitigate the concern that subjects would simply view the line-thickness differences in terms of thinness versus thickness, although implementation of such scaling would require additional baked-lines textures and would probably be strictly quantized due to the nature of line precomputation. Future researchers could also seek an improved technique for color-edge detection that does not force NPR lines to be at least two pixels in width; such a change may permit the elimination of distracting and irrelevant differences in the general sharpness and opacity of lines. In addition, they may wish to correct the bug in this study's application that apparently involved stored texture colors, so that colors would not be inaccurately displayed onscreen and so that dynamically-calculated color differences would be consistent with the precomputed ones.

In terms of methodology and execution of testing, investigators conducting a similar study could exercise greater care to provide complete and identical information and guidance to every tester, eliminating as much as possible the need for clarifying questions. They may likewise desire to implement checks for sufficient scene exploration that are both more methodologically robust and better tailored to the needs of individual users (e.g., not so demanding in terms of time that testers would wish to skip the checks). Finally, they could consider opening testing to individuals younger than 18 years of age, in order to more accurately represent the general population with their sample.

5.4 Summary

This chapter provided conclusions derived from the study's quantitative results, brought up possible explanations of unexpected developments, further examined the problems encountered in the study, and suggested avenues of future research.

LIST OF REFERENCES

LIST OF REFERENCES

- Ao-yu, W., Min, T., & Jin-Xiang, D. (2004). A survey of silhouette detection techniques for non-photorealistic rendering. In *2004 IEEE First Symposium on Multi-Agent Security and Survivability* (pp. 434–437). doi: 10.1109/ICIG.2004.28
- Bratkova, M., Boulos, S., & Shirley, P. (2009). oRGB: A practical opponent color space for computer graphics. *IEEE Computer Graphics and Applications*, *29*(1), 42–55. doi: 10.1109/MCG.2009.13
- Cardona, L., & Saito, S. (2014). Gaussian curvature based localized stylization method for view-dependent lines extracted from 3D models. In M. L. Gavrilova, C. J. K. Tan, X. Mao, & L. Hong (Eds.), *Transactions on Computational Science XXIII* (pp. 40–57). Springer Berlin Heidelberg. doi: 10.1007/978-3-662-43790-2_3
- Cole, F., Golovinskiy, A., Limpaecher, A., Barros, H. S., Finkelstein, A., Funkhouser, T., & Rusinkiewicz, S. (2012). Where do people draw lines? *Communications of the ACM*, *55*(1), 107–115. doi: 10.1145/2063176.2063202
- Cole, F., Sanik, K., DeCarlo, D., Finkelstein, A., Funkhouser, T., Rusinkiewicz, S., & Singh, M. (2009). How well do line drawings depict shape? In H. Hoppe (Ed.), *ACM SIGGRAPH 2009 Papers* (pp. 28:1–28:9). New York: ACM. doi: 10.1145/1576246.1531334
- Constable, M. (2011). Artist-led suggestions towards an approach in content aware 3D non-photorealistic rendering. In R. Koch & F. Huang (Eds.), *Computer Vision – ACCV 2010 Workshops* (pp. 142–151). Springer Berlin Heidelberg. doi: 10.1007/978-3-642-22819-3_15
- DeCarlo, D., Finkelstein, A., Rusinkiewicz, S., & Santella, A. (2003). Suggestive contours for conveying shape. In A. P. Rockwood (Ed.), *ACM SIGGRAPH 2003 Papers* (pp. 848–855). New York: ACM. doi: 10.1145/1201775.882354
- Duke, D., Barnard, P., Halper, N., & Mellin, M. (2003). Rendering and affect. *Computer Graphics Forum*, *22*(3), 359–368. doi: 10.1111/1467-8659.00683
- Feldman, U. (1993). *Quantifying the dimensions of color experience*. Thesis, Massachusetts Institute of Technology. Retrieved 2015-04-07, from <http://dspace.mit.edu/handle/1721.1/12558>
- Geng, P. W. (2010). Artistic rendering for 3D object. In *The algorithms and principles of non-photorealistic graphics* (pp. 161–228). Springer Berlin Heidelberg. doi: 10.1007/978-3-642-04891-3_5

- Gooch, B., Sloan, P.-P. J., Gooch, A., Shirley, P., & Riesenfeld, R. (1999). Interactive technical illustration. In J. Rossignac, J. Hodgins, & J. D. Foley (Eds.), *Proceedings of the 1999 Symposium on Interactive 3D Graphics* (pp. 31–38). New York: ACM. doi: 10.1145/300523.300526
- Goodwin, T., Vollick, I., & Hertzmann, A. (2007). Isophote distance: A shading approach to artistic stroke thickness. In B. Gooch, M. Agrawala, & O. Deussen (Eds.), *Proceedings of the 5th International Symposium on Non-photorealistic Animation and Rendering* (pp. 53–62). New York: ACM. doi: 10.1145/1274871.1274880
- Hajagos, B., Szécsi, L., & Csébfalvi, B. (2012). Fast silhouette and crease edge synthesis with geometry shaders. In C. O’Sullivan (Ed.), *Proceedings of the 28th Spring Conference on Computer Graphics* (pp. 71–76). New York: ACM. doi: 10.1145/2448531.2448540
- Isenberg, T., Neumann, P., Carpendale, S., Sousa, M. C., & Jorge, J. A. (2006). Non-photorealistic rendering in context: An observational study. In *Proceedings of the 4th International Symposium on Non-photorealistic Animation and Rendering* (pp. 115–126). New York, NY, USA: ACM. doi: 10.1145/1124728.1124747
- Lin, J., Yang, H., Wang, D., Li, G., Gao, X., Wu, Q., & Liao, M. (2015). Fast and high quality suggestive contour generation with L0 gradient minimization. In Y. Tan, Y. Shi, F. Buarque, A. Gelbukh, S. Das, & A. Engelbrecht (Eds.), *Advances in Swarm and Computational Intelligence* (pp. 484–491). Springer International Publishing. doi: 10.1007/978-3-319-20472-7_52
- Magdics, M., Sauvaget, C., García, R. J., & Sbert, M. (2013). Post-processing NPR effects for video games. In C. C. L. Wang, N. Magnenat-Thalmann, & Z. Pan (Eds.), *Proceedings of the 12th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry* (pp. 147–156). New York: ACM. doi: 10.1145/2534329.2534348
- Masuch, M., Schlechtweg, S., Schönwälder, B., & Magdeburg, D. (1997). daLi! - Drawing Animated Lines! In O. Deussen & P. Lorenz (Eds.), *Proceedings of Simulation und Animation 97, SCS Europe* (pp. 87–96). Retrieved from <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.8.4605>
- Matsuo, T., Mikami, K., Watanabe, T., & Kondo, K. (2011). Shape oriented line drawing in real-time 3DCG. In Z. Pan (Ed.), *SIGGRAPH Asia 2011 Posters* (pp. 45:1–45:1). New York: ACM. doi: 10.1145/2073304.2073355
- Mitchell, J. L., Brennan, C., & Card, D. (2002). Real-time image-space outlining for non-photorealistic rendering. In T. Appolloni (Ed.), *ACM SIGGRAPH 2002 Conference Abstracts and Applications* (pp. 239–239). New York: ACM. doi: 10.1145/1242073.1242252
- Mokrzycki, W. S., & Tatol, M. (2011). Color difference Delta E - A survey. *Machine Graphics and Vision*, 20(4), 383–411. Retrieved from <http://www.researchgate.net/publication/236023905>

- Nienhaus, M., & Döllner, J. (2004). Sketchy drawings. In L. van Zijl & P. Marais (Eds.), *Proceedings of the 3rd International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa* (pp. 73–81). New York: ACM. doi: 10.1145/1029949.1029963
- Ohtake, Y., Belyaev, A., & Seidel, H.-P. (2004). Ridge-valley lines on meshes via implicit surface fitting. In J. Marks (Ed.), *ACM SIGGRAPH 2004 Papers* (pp. 609–612). New York: ACM. doi: 10.1145/1186562.1015768
- Perrot, M., Habrard, A., Muselet, D., & Sebban, M. (2014). Modeling perceptual color differences by local metric learning. In D. Fleet, T. Pajdla, B. Schiele, & T. Tuytelaars (Eds.), *Computer Vision – ECCV 2014* (pp. 96–111). Springer International Publishing. doi: 10.1007/978-3-319-10602-1_7
- Saito, T., & Takahashi, T. (1990). Comprehensible rendering of 3-D shapes. In F. Baskett (Ed.), *Proceedings of the 17th Annual Conference on Computer Graphics and Interactive Techniques* (pp. 197–206). New York: ACM. doi: 10.1145/97879.97901
- Sanocki, T., Bowyer, K. W., Heath, M. D., & Sarkar, S. (1998). Are edges sufficient for object recognition? *Journal of Experimental Psychology: Human Perception and Performance*, 24(1), 340–349. doi: 10.1037/0096-1523.24.1.340
- Son, M., Kang, H., Lee, S., & Lee, S. (2007). Abstract line drawings from 2D images. In *15th Pacific Conference on Computer Graphics and Applications, 2007. PG '07* (pp. 333–342). doi: 10.1109/PG.2007.63
- Wang, S., Ma, Z., Liu, X., Chen, Y., & Wu, E. (2012). Coherence-enhancing line drawing for color images. *Science China Information Sciences*, 56(11), 1–11. doi: 10.1007/s11432-012-4685-5

APPENDIX

APPENDIX

This appendix provides descriptions of the shader algorithms used in the testing application. It also provides the complete code for the first and second image-processing shaders, in Figures A.1 and A.2 respectively.

A.1 Lighting

Lighting was calculated normally for the default render, but it was deferred to the first image-processing pass in both NPR algorithms, so that in the dynamic method the base texture colors could be utilized for color-difference calculations prior to that point, and so that the shading would look identical between the two NPR algorithms. The deferred lighting values were placed in the alpha channel of the initial color map used as input for the first image-processing pass, and the pre-lit color values were placed in the RGB channels. (As detailed in Section , this strategy unfortunately ended up creating quantization issues with lighting in the NPR renders, presumably on account of the limited range of values possible for a pixel's alpha between shader passes relative to those possible for a floating-point variable utilized within a single shader.)

A.2 Image Processing for Dynamic Method

In the first post-processing pass for the dynamic method, Sobel edge detection was performed on depth and normal maps so that silhouette and contour lines would be included in the final renders (Figure A.1, Lines 93-130). Both horizontal and vertical 3x3 Sobel kernels were used, and edge detection individually took place in each RGB channel of the normal map. The per-pixel scalars resulting

from the edge detection process – positive for pixels on normal edges or depth edges and zero for others – were multiplied by a factor that varied linearly with depth – specifically, a fractional constant divided by the non-linear depth-map value at the current pixel (Lines 107-108); this acceptably approximated depth-based scaling on account of the extreme nearness of the near clipping plane of the cameras to the eye (“zero”) point. The products of the Sobel-edge scalars and the “depth factors” were then stored in the alpha channel of the new output map. Next, around all pixels not on a Sobel edge, a 3x3 neighborhood of pixels was sampled from the initial, non-light-blended color render, and each of the four opposing pairs of pixels around the center had its color values compared (Lines 156-162, Lines 51-68) according to the $\Delta E''_{\text{RGB}}$ color-difference metric (see Section 2.4). The largest of the four resulting color-difference scalars was placed in the alpha channel of the output map at the current pixel (Lines 164-167); higher values indicated higher color contrasts. Like the per-pixel Sobel edge scalars, this color-contrast value was multiplied by the “depth factor” mentioned above. At the end of this post-processing pass, the output map would have newly light-blended colors in its RGB channels (i.e., the products of the initial color map’s RGB values and its alpha values) and the scalar edge and color-contrast values in its alpha channel.

For the second post-processing pass, the output map from the prior step was the new input. This step simply superimposed dark NPR lines of varying thickness over the colors by blending pixel values in accordance with edge and color-contrast values in the alpha channels of nearby pixels. In effect, a black circle was intended to be generated around each pixel that had non-zero alpha; overlapping circles around neighboring pixels would form lines in the final render. The higher a pixel’s alpha value would be, the larger its circle would be, and thus the thicker the line would be at that point. To that end, around each pixel in this post-processing pass, a constant-sized square neighborhood of pixels was sampled (Figure A.2, Lines 36-40). Each pixel in this neighborhood had its alpha value examined. If the alpha value was high enough for a correspondingly large imaginary circle around it to

```

1 Shader "PostProcessEdges"
2 {
3   Properties
4   {
5     _MainTex ("", 2D) = "white" {}
6     _HKV ("High Kernel Value", Float) = 2
7     _LKV ("Low Kernel Value", Float) = 1
8     _ShapeThreshold ("Shape Threshold", Float) = 0.25
9     _ColorThreshold ("Color Threshold", Float) = 0.01
10    _NormMod ("Normal Modifier", Range(0.0, 1.0)) = 0.5
11    _ICLOn ("ICL On", Float) = 1.0
12    _ThicknessDivisor ("Thickness Divisor", Float) = 0.0
13  }
14
15  SubShader
16  {
17    Pass
18    {
19      ZTest Always
20      Cull Off
21      ZWrite Off
22      Fog { Mode off }
23
24      CGPROGRAM
25      #pragma vertex vert_img
26      #pragma fragment frag
27      #pragma target 3.0
28      #include "UnityCG.cginc"
29
30      uniform sampler2D _MainTex;
31      uniform float4 _MainTex_TexelSize;
32      uniform sampler2D _CameraDepthNormalsTexture;
33      uniform float _NormMod;
34      uniform float _HKV;
35      uniform float _LKV;
36      uniform float _ShapeThreshold;
37      uniform float _ColorThreshold;
38      uniform float _AlgType;
39      uniform float _ICLOn;
40      uniform float _ThicknessDivisor;
41
42      int getRGBEqual(float3 col1, float3 col2)
43      {
44        if (col1.r == col2.r && col1.g == col2.g &&
45            col1.b == col2.b)
46          return 1;
47
48        return 0;
49      }
50
51      float getRGBDifference(float3 col1, float3 col2)
52      {
53        float3 cDiff = float3(col1.r - col2.r,
54                              col1.g - col2.g,
55                              col1.b - col2.b);
56
57        float rAvg = (col1.r + col2.r) / 2.0;
58        float DE = sqrt(((2.0 + rAvg) * cDiff.r * cDiff.r +
59                       (4.0 - rAvg) * cDiff.g * cDiff.g +
60                       (3.0 - rAvg) * cDiff.b * cDiff.b));
61        DE = DE / 3.0; // normalize max value
62                       // (black vs. white = 3.0) to 1.0
63
64        if (DE < _ColorThreshold)
65          return 0;
66
67        return max(0.0f, min(1.0f, (DE - _ColorThreshold) /
68                            (1 - _ColorThreshold)));
69      }
70
71      float4 frag (v2f_img i) : COLOR
72      {
73        float s[9]; // Sobel kernel
74        float cnvRedX = 0;
75        float cnvRedY = 0;
76        float cnvGreenX = 0;
77        float cnvGreenY = 0;
78        float cnvBlueX = 0;
79        float cnvBlueY = 0;
80        float cnvDepthX = 0;
81        float cnvDepthY = 0;
82        float3 normMaskValue;
83        float4 inputCol;
84        float finalCol;
85        float3 norm;
86        float depth;
87        float depthFactor = 1;
88        int indX;
89        int indY;
90
91        inputCol = tex2D(_MainTex, i.uv.xy);
92
93        // silhouette/contour edge detection
94
95        s[0] = _LKV; s[1] = _HKV; s[2] = _LKV;
96        s[3] = 0.0; s[4] = 0.0; s[5] = 0.0;
97        s[6] = _LKV; s[7] = _HKV; s[8] = _LKV;
98
99        for (int k = 0; k < 3; k++)
100         for (int j = 0; j < 3; j++)
101         {
102           indX = k * 3 + j;
103           indY = j * 3 + k;
104           DecodeDepthNormal(tex2D(_CameraDepthNormalsTexture, i.uv.xy +
105                                 _MainTex_TexelSize * float2(j-1, k-1)),
106                             depth, norm);
107
108           if (j == 1 && k == 1)
109             depthFactor = 1.0f / depth / _ThicknessDivisor;
110           norm = normalize(norm * (UNITY_MATRIX_T_MW,
111                               float4(norm, 0.0)).xyz);
112           norm = float3(_NormMod, _NormMod, _NormMod) +
113                 (1.0 - _NormMod) * norm;
114           cnvRedX += (norm.r * s[indX]);
115           cnvRedY += (norm.r * s[indY]);
116           cnvGreenX += (norm.g * s[indX]);
117           cnvGreenY += (norm.g * s[indY]);
118           cnvBlueX += (norm.b * s[indX]);
119           cnvBlueY += (norm.b * s[indY]);
120           cnvDepthX += (depth * s[indX]);
121           cnvDepthY += (depth * s[indY]);
122         }
123
124         normMaskValue.r = sqrt(pow(cnvRedX, 2) + pow(cnvRedY, 2));
125         normMaskValue.g = sqrt(pow(cnvGreenX, 2) + pow(cnvGreenY, 2));
126         normMaskValue.b = sqrt(pow(cnvBlueX, 2) + pow(cnvBlueY, 2));
127         depthMaskValue = sqrt(pow(cnvDepthX, 2) + pow(cnvDepthY, 2));
128         finalCol = 0.1 * (normMaskValue.r + normMaskValue.g + normMaskValue.b);
129         if (finalCol < _ShapeThreshold)
130           finalCol = 0;
131         finalCol = max(finalCol, min(1.0, 10 * depthMaskValue));
132         if (_ICLOn == 0.0)
133           return float4(inputCol.r * inputCol.a, inputCol.g * inputCol.a,
134                        inputCol.b * inputCol.a, finalCol * depthFactor);
135
136         float3 color1[4];
137         float3 color2[4];
138         float tempOutput[4];
139
140         color1[0] = tex2D(_MainTex, i.uv.xy + _MainTex_TexelSize *
141                           float2(0, -1)).rgb;
142         color1[1] = tex2D(_MainTex, i.uv.xy + _MainTex_TexelSize *
143                           float2(1, -1)).rgb;
144         color1[2] = tex2D(_MainTex, i.uv.xy + _MainTex_TexelSize *
145                           float2(1, 0)).rgb;
146         color1[3] = tex2D(_MainTex, i.uv.xy + _MainTex_TexelSize *
147                           float2(1, 1)).rgb;
148         color2[0] = tex2D(_MainTex, i.uv.xy + _MainTex_TexelSize *
149                           float2(0, 1)).rgb;
150         color2[1] = tex2D(_MainTex, i.uv.xy + _MainTex_TexelSize *
151                           float2(-1, 1)).rgb;
152         color2[2] = tex2D(_MainTex, i.uv.xy + _MainTex_TexelSize *
153                           float2(-1, 0)).rgb;
154         color2[3] = tex2D(_MainTex, i.uv.xy + _MainTex_TexelSize *
155                           float2(-1, -1)).rgb;
156
157         for (int c = 0; c < 4; c++)
158         {
159           if (getRGBEqual(color1[c], color2[c]) == 1)
160             tempOutput[c] = 0;
161           else
162             tempOutput[c] = getRGBDifference(color1[c], color2[c]);
163         }
164
165         finalCol = max(max(max(max(tempOutput[0], tempOutput[1]),
166                                 tempOutput[2]), tempOutput[3]), finalCol);
167         return float4(inputCol.r * inputCol.a, inputCol.g * inputCol.a,
168                      inputCol.b * inputCol.a, finalCol * depthFactor);
169       }
170     }
171   }
172 }
173 }
174 }

```

Figure A.1. The code for the first post-processing shader, which performs edge detection and comparison of colors across inter-color edges, is displayed here in full.

encompass the center pixel, the center pixel would be made black (Lines 41-42). If the center pixel was outside the circle's edge by a mere sub-pixel distance, it would receive an intermediate level of darkening. Otherwise, the pixel colors would remain

unchanged. This process would repeat for all pixels in the neighborhood, with the darkest blending value being used in the end for the center pixel.

```

1 Shader "PostProcessFinal"
2 {
3   Properties
4   {
5     _MainTex ("", 2D) = "white" {}
6     _BrushR ("Brush Radius", Float) = 11.0
7     _Tint ("Tint", Float) = 0.0
8   }
9
10  SubShader
11  {
12    Pass
13    {
14      ZTest Always
15      Cull Off
16      ZWrite Off
17      Fog { Mode off }
18
19      CGPROGRAM
20      #pragma target 4.0
21      #pragma vertex vert_img
22      #pragma fragment frag
23      #include "UnityCG.cginc"
24
25      uniform sampler2D _MainTex;
26      uniform float4 _MainTex_TexelSize;
27      uniform float _BrushR;
28      uniform float _Tint;
29
30      float4 frag (v2f_img i) : COLOR
31      {
32        float4 inputCol = tex2D(_MainTex, i.uv.xy);
33        float curA;
34        float tempMult = 1;
35
36        for (float k = -_BrushR; k <= _BrushR; k += 1.0)
37          for (float j = -_BrushR; j <= _BrushR; j += 1.0)
38            {
39              curA = tex2D(_MainTex, i.uv.xy +
40                _MainTex_TexelSize.xy * float2(j, k)).a;
41              tempMult = min(tempMult, clamp(1 + sqrt(j*j + k*k) -
42                curA * _BrushR, 0, 1));
43            }
44
45        return float4((_Tint + (1 - _Tint) * inputCol.r) * tempMult,
46          (_Tint + (1 - _Tint) * inputCol.g) * tempMult,
47          (_Tint + (1 - _Tint) * inputCol.b) * tempMult, 1);
48      }
49    }
50  }
51  FallBack off
52 }
53
54

```

Figure A.2. The code for the second post-processing shader, which generates NPR lines of varying thickness on the rendered image based on the output map from the first, is displayed here in full.

Given the method of checking inter-color edges, the first post-processing pass could not in fact generate inter-color edge representations in its output edge map that were less than two pixels in width. For this reason, the second pass could not output inter-color lines less than two pixels thick either. Thus, if output lines were purely black, lines intended to appear one pixel wide would appear too thick. For this reason, the aforementioned partial darkening of pixels (Figure A.2, Lines 41-42) was implemented to handle subpixel offsets of pixels from true edges. Thus, output lines meant to be less than two pixels thick would appear lighter in weight from a distance than their actual thickness implied. This also had the side effect of creating an antialiasing-like effect at the outer edges of normal, thick, black lines in the final

renders. Figure A.3 demonstrates the use of partially-darkened pixels to represent lines that are meant to be approximately one pixel thick.



Figure A.3. These cropped portions of side-by-side NPR renders illustrate the partial-darkening effect for thin lines. On the right, the baked-lines method displays lines less than two pixels thick; the dynamic method draws thicker lines, but it dims them to compensate.

A.3 Image Processing for Baked-Lines Method

The two post-processing steps were the same for the baked-lines method, except that the color-difference checks around the current pixel in the first pass were omitted. In other words, only the Sobel edge values for generating silhouettes and contours, modified by the “depth factor”, were placed in the alpha channel in that pass. The baked-lines method utilized the same post-processing shader files, but the “_ICLON” (i.e., “[Dynamic] Inter-Color Lines On”) variable in the first post-processing pass was set to 0 instead of 1 for this method, so that the early return statement on Lines 132-133 would be executed.