Perception of Simplification Artifacts for Animated Characters

Michéal Larkin
Trinity College Dublin.

Carol O'Sullivan
Trinity College Dublin.

Figure 1: *Example scenes showing screen spaces used in our experiments*

Abstract

For real-time animation of characters, Level of Detail (LOD) techniques are usually deployed to achieve optimal efficiency and realism. Tools that can aid an application designer or developer to deploy simplification methods and LOD representations under different circumstances are therefore needed. In this paper we explore the perception of *texture*, *silhouette* and *lighting* artifacts on simplified character models, including the effect animation intensity has on the perceptibility of these artifacts. We propose novel methods to simulate each artifact separately, and test their effects in isolation and cumulatively. Our results provide useful insights and guidelines, including the importance of silhouette preservation. An adaptation of PerceptualDiff for predicting simplification detection has been developed, which can be used to guide automatic model placement and LOD selection in realtime character applications.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism—[Rendering]

Keywords: Level of Detail, Perception

1 Introduction

Realistic animated characters are needed for a wide variety of applications and often it is necessary to simulate high resolution meshes in order to meet user expectations. Even with the deployment of methods such as normal mapping, the number of triangles needed to animate such characters' faces and bodies is increasing rapidly. Therefore, Level of Detail (LOD) techniques, such as the replacement of complex geometry with a simplified substitute are particularly important to allow for real time computation and display in applications such as games and interactive virtual environments. Crowd simulation is another area where LOD creation is central to achieving optimal performance and realism. Tools that can aid an application designer or developer to deploy simplification methods and LOD representations under different circumstances are therefore needed.

In order to develop such a tool, in this paper we present a comprehensive study of the artifacts that are produced as an inevitable effect of LOD simplification of characters, focussing on the most common types: *texture*, *silhouette* and *lighting* anomalies. We also test the effect of animation on accentuating or masking simplifications. It is difficult to quantify the relative and cumulative effects of different errors, so the novelty of our study lies in the fact that we propose methods to simulate each artifact separately, and test their effects in isolation and cumulatively. We found that:

- Silhouette is the dominant artifact for simplification identification at smaller screen spaces
- Lighting and Silhouette artifacts are easily detected at larger screen spaces.
- Applying animations of different motion intensities has no effect on the perceptibility of any errors

Furthermore, in addition to providing useful guidelines for developers of games and other interactive applications, we also develop a perceptual metric that can be used to estimate the screen space at which a LOD simplification will go unnoticed when applied to a human character mesh. Development of this metric allows us to determine as a preprocess the distance in a virtual environment at which the replacement of a high level of detail mesh with a low resolution version will go unnoticed. This will be of particular use for automatic model placement and determination of LOD switching distances in realtime character applications such as crowd and group simulations.

1.1 Outline

We begin with a series of experiments designed to investigate the most salient artifacts introduced by a simplification operation. The seperation of these artifacts is detailed in Section 3. We then perform perceptual experiment designed to collect data on the perceptibility of silhouette simplifications, detailed in Sections 4 and 5. In Section 6 an image space metric is tested and adapted to three dimensional space. This metric is based on a model of the human visual system and requires no prior knowledge of the meshes to be tested. The developed metric is then tested and shown to effectively

APGV 2011, Toulouse, France, August 27 – 28, 2011.
© 2011 ACM 978-1-4503-0889-2/11/0008 $10.00

position meshes automatically in a 3D scene.

2 Related Work

Many algorithms exist that decimate meshes to various levels of detail, most of which use an object space metric to measure the error introduced by simplification. A comprehensive review of LOD techniques may be found in [Luebke 2003].

The difference between a high and low resolution mesh may be measured in object space with the Metro tool [Cignoni et al. 1998]. A more natural way to determine the error introduced by a simplification operation may be to measure the error introduced when rendering that mesh. Such an image space metric is described in [Lindstrom and Turk 2000].

A number of algorithms have taken into account human perception as a metric for level of detail. Luebke et al. [2000] use gaze data to simplify textured meshes, while Luebke and Hallen [2001] order simplifications according to their perceptual impact on a mesh to create a best-effort simplification. Williams et al. [2003] employ a contrast sensitivity function to simplify meshes based on sensitivity to changes in contrast, thereby retaining important details. An algorithm inspired by human perception was developed by Lee et al. [2005], which captures and retains detail in visually important areas of a mesh. These methods are based on models of the human visual system, closely relating simplifications to human perception. The Root Mean Squared (RMS) error has been used to measure the difference between rendered images [Lindstrom and Turk 2000]. By determining the RMS distance between images rendered from a selection of viewpoints sampled before and after a simplification operation, they can order the simplification operations so that the minimum error is introduced to the mesh. While this method works well, the RMS error has no basis in human perception.

Teo and Heeger [1994] note that Mean Squared Error (MSE) calculations are inaccurate at predicting perceptual distortion. They present a perceptual distortion error that is more accurate than MSE. The Visible Differences Predictor (VDP) has been proposed to determine the probability that the difference between two images will be detected [Daly 1993]. Comparisons between different image metrics have demonstrated that VDP performed well [Rushmeier et al. 1995]. Ramasubramanian et al. [1999] decoupled the computationally expensive spatially-dependent component from the luminance-dependent component in the VDP calculation to increase speed. They developed an error metric based on physical data that predicts the perceptual threshold at which artifacts may be detected. Two further methods that exploit human perception are *ltdiff* by Lindstrom [2000] and *PerceptualDiff* by Yee and Newman [2004]. The output of the PerceptualDiff algorithm is a number that the user may use to decide whether the images are below a threshold of perceptible difference. This threshold is simply defined as the number of pixels that are percieved to be different. Using this method to determine the perceptibility of simplification artifacts requires modification of the algorithm.

The popping artifact introduced by switching between discrete levels of detail may be disguised in both image space and object space. Switches may be disguised in image space by using alpha blending [Luebke 2003], by rendering both levels of detail simultaneously, and blending between them as the distance changes. Giegl and Wimmer [2007] improve on the errors visible when alpha blending is employed with minimal overhead, while the blending method can be further improved by rendering objects of different resolutions in separate passes and interpolating between the renderings using visibility textures [Scherzer and Wimmer 2008]. Switches may be disguised in object space by geomorphing between levels of detail, where between levels of detail, an object's vertices are interpolated between their positions in the low and high resolution representations [Hoppe 1996]. [Schwarz and Stamminger 2009] have presented a model that predicts the perceptibility of popping artifacts between level of detail representations during a level of detail switch. Our work concentrates on the perceptibility of the differences between two discrete level of detail models.

3 Artifact Separation

The artifacts introduced by simplification operations may be seperated into three categories: *Texture* artifacts are caused by stretching the texture applied to the mesh, due to fewer UV texture coordinates. *Lighting* artifacts are introduced by the reduction in the number of surface normals available for lighting calculations. *Silhouette* artifacts are introduced by the physical alteration of the surface of the mesh. The simplification methods discussed in Section 2 do not consider these artifacts as seperate entities.

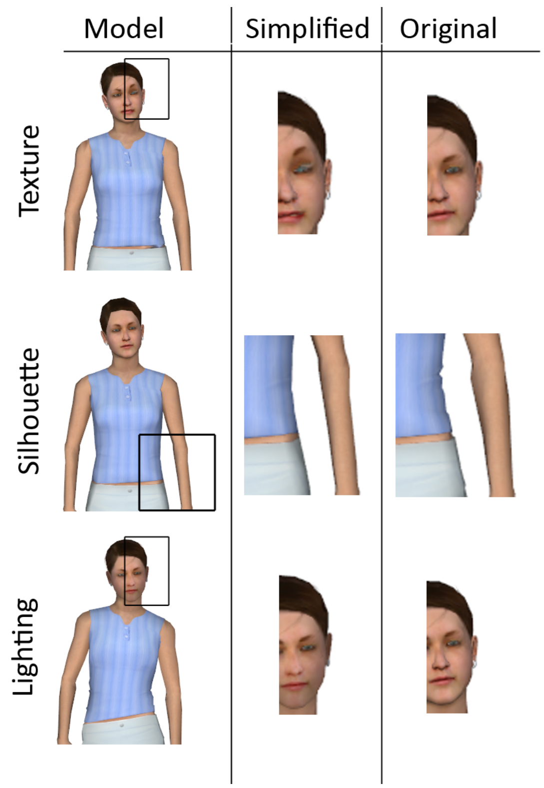

Figure 2: *Texture, Silhouette and Lighting Simplifications*

We hypothesize that different artifacts introduced by simplification will have different levels of perceptibility. We define the three major rendering artifacts introduced by mesh simplification to be *texture*, *silhouette* and *lighting*. For animated meshes, other artifacts may be introduced if detail is not maintained around areas of high deformation. Removing a large number of vertices around a deformable joint such as the elbow may introduce artifacts when the arm is bent. Artifacts such as this may be avoided by careful simplification, or the use of an automatic simplification method tailored towards animated meshes.

To independently assess the effect of each artifact; *texture*, *silhouette* and *lighting*, we have developed a method to generate meshes displaying only one of each of these artifacts. To ensure that we are not creating arbitrary simplifications, we base our introduced artifacts on data from *artist simplified* and *automatically simplified*

meshes. By separating each of the artifacts, we can test what individual effect they have on the perceptibility of a Level Of Detail (LOD) simplification. An example of the models used may be seen in Figure 1.

To recreate the texture errors on a high resolution mesh, we take samples of the UV coordinates from the low-res mesh and apply them to each vertex of the high-res mesh. The high-res mesh and its low resolution alternative are placed in the same object space. The space containing the low-res mesh is then voxelised. For each vertex on the high-res mesh we find the closest point on the low-res mesh by searching the voxelised space to find the closest triangle and then searching for the closest point on that triangle. Barycentric coordinates are then used to retrieve the exact UV coordinates on the low resolution mesh at that point. The UV coordinates on the high-res mesh are then replaced with the coordinates of the closest point on the surface of the low-res mesh. This results in a high-res mesh containing only the texture artifacts resulting from a LOD simplification (see Figure 2:top). To create a mesh containing only lighting artifacts, we sample the normals from the low-res mesh and apply those normals to the high-res mesh (see Figure 2:bottom).

To recreate the effects of a silhouette simplification, we must alter the high-res mesh to match the surface of the low-res mesh. Adjusting all vertices to match the low-res mesh may introduce texture and lighting artifacts as the surface changes. To eliminate these extraneous effects, we must only adjust the silhouette of the high-res mesh. As a preprocess, we calculate the "simplified" positions of all the vertices on the high-res mesh. These new vertex positions are stored in the character's vertex buffer. At runtime, we can determine the vertices that are on the silhouette of the character for the current frame of animation. Vertices near the silhouette have their position adjusted towards the simplified vertex position. As vertices reach the silhouette, their position will be as in the simplified mesh (see Figure 2:middle). This ensures that only the artifacts under scrutiny are introduced. A small amount of distortion between the original and simplified mesh is inevitable at the areas where the silhouette is altered, but in practice this distortion is outweighed by the distortion introduced to the silhouette. We used two types of meshes: artist simplified and meshes simplified with an automatic mesh decimation algorithm; Intel's MultiRes algorithm in 3D Studio Max, to ensure that our results are as generalizable as possible.

The experimental system was developed using an open-source renderer based on DirectX 9.0 and was displayed on a workstation with 4GB of RAM and an 8-series NVidia card. The characters were rendered using the Blinn-Phong model implemented as a Higher Level Shading Language (HLSL) program. The scene was rendered with 4x full-scene anti-aliasing (FSAA). All of the experiments were run at a maximum resolution of 1920x1200 pixels on a widescreen 24 inch flat screen LCD monitor with a dot pitch value of 0.27mm. Participants were from a range of different educational backgrounds, had normal or corrected to normal vision and were naïve as to the purpose of each experiment.

4 Artifact Detection

The first experiment was designed to investigate the factors that affected participants' ability to discrimininate between a typical high resolution game character mesh and a lower resolution LOD version of that mesh. We first hypothesized that *distance will have an effect on the perception of our four chosen artifacts*. Our second hypothesis was that *animation will reduce artifact detection*.

It is intended that the results of this experiment will inform a perceptual metric to estimate the screen space size at which a LOD mesh is considered equivalent to its original. This metric could then be used by artists/programmers to assist in the process of setting the display parameters for a LOD mesh, thereby making it perceptually indistinguishable from the original.

4.1 Experiment Design

Thirteen volunteers (7M,6F) took part in the experiment. We used a 2-Alternative Forced Choice (2AFC) experiment design, where the participants were asked to choose whether the character on the left or the right was the simplified one. We recorded both the accuracy of their responses and their reaction times. A gold standard high resolution mesh was always displayed in the centre. The first condition we tested was *screen space* (large, medium, small). Examples of the screen spaces may be seen in Figure 1. The gold standard mesh was always displayed at the large screen space, while the other two were displayed simultaneously at one of the other three screen spaces. The screen spaces were chosen by hand to give a reasonable range of perceptibility of artifacts, from obvious to subtle. We use screen space as a measurement, as distance in a 3D world is dependent on too many factors, such as resolution and render camera field of view.

The *artifact* (texture, silhouette, lighting, all three combined (TSL)) condition displayed the separated errors as described in Section 3. TSL here is actually the original simplified mesh, displaying a combination of all the simplification artifacts. We included TSL as a control in order to determine if the combination of all artifacts was worse than each one in isolation. The screen spaces were chosen based on pilot tests to ensure that the perceptibility of the artifact conditions ranged from obvious to subtle.

We also varied the *simplification method* (artist simplified, automatically simplified). The artifact being tested was applied to the simplified character only. We randomised whether the simplified character was displayed on the left or the right. The final condition was *motion intensity* (low, high), for which we used motion captured data. The low intensity motion was a simple idle animation, while the high intensity motion was a fast walking animation, which was displayed on the spot. We displayed the animation on the spot as we are testing screen space as a factor, and the inclusion of movement around the world would alter the screen space at which a character would be displayed.

We separated the experiment into two randomly presented blocks, so that the animations would be consistent in each. In the first block, the participants viewed the characters with the low intensity animation, with high intensity in the second block. Ninety-six trials were shown in each block: 4 artifact types * 2 simplification methods * 3 screen spaces * 2 template models * 2 repetitions. Trials were shown randomly to each participant to avoid ordering effects. Before running the experiment, participants were shown examples to familiarise them with the four artifact types, using a different template model.

4.2 Results

We averaged each participants responses over repetitions and template models. A three way repeated measures ANalysis Of VAriance (ANOVA) was conducted on the data. Post-hoc analysis was performed using Newman-Keuls comparisons of means, which is a conservative test for significance.

4.2.1 Accuracy

First, we considered the results for participant accuracy. A main effect of artifact was found ($F_{3,36} = 22.686, p < 0.00001$), see Figure 3. Overall, participants were significantly less accurate at

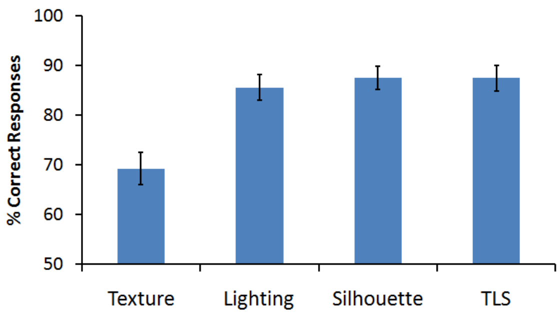

Figure 3: *Main effect of Artifact*

detecting texture artifacts than silhouette, lighting, or TSL. There was no significant effects for the other artifacts.

A main effect of screen space was also found, ($F_{2,24} = 67.95, p < 0.00001$). As expected, participants were significantly less accurate as screen space decreased ($p < 0.0005$ in all cases). At the largest screen space, participants were 95% accurate, which dropped to 68% at the smallest screen space. This shows that overall the artifacts were still quite perceptible at the smallest screen space (as 50% would be chance level). We found no main effect of simplification method, implying that there was no overall difference between the perception of the artifacts introduced, regardless of whether the mesh was simplified by an artist or automatically.

There was also no effect of animation intensity, meaning that the animations applied to the characters had neither an accentuating nor masking effect on the detection of any of the artifacts. We also found no interaction between simplification method or animation intensity and any of the other conditions. However, there was an interaction between artifact and screen space ($F_{6,27} = 6.657, p < 0.00002$), see Figure 4. All artifacts were detected with equal accuracy at the closest distance, but as screen space decreased, detection of texture artifacts became significantly more difficult than the other three ($p < 0.0005$ at distance 2 in all cases), ($p < 0.005$ at dist 3 in all cases). Notably, there was no significant difference in accuracy between TSL and silhouette or lighting alone at all three screen spaces.

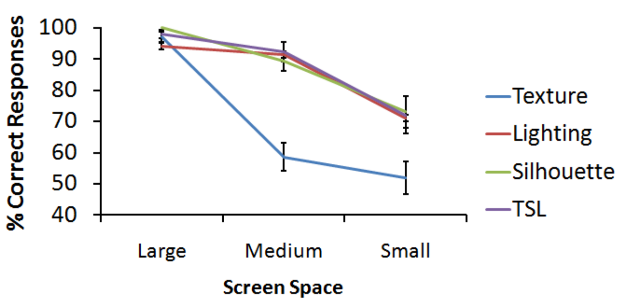

Figure 4: *Interaction effect between artifact and screen space*

4.2.2 Response Time

Response time for correct responses was also analyzed in order to determine if the time taken by participants to identify different artifacts varied. Analysis of response times showed a main effect of artifact ($F_{3,36} = 14.08, p < 0.00005$). Overall reaction times for silhouette and TSL combined were significantly faster than for texture and lighting ($p < 0.0005$). From the results shown in Figure 5 we can see that participants had a harder time identifying texture and lighting artifacts, and that the response times for silhouette and tsl were closely correlated. Though texture and lighting artifacts could be seen, it was only after the participant had searched for them, whereas the silhouette artifact was quickly detected.

A main effect of screen space on response time was also found, ($F_{2,24} = 12.299, p < 0.00021$), where participants were faster at answering for the largest screen space, as expected. Response time was also measured and analysed for motion intensity. Again, there was no effect of motion, or interaction between motion and other factors.

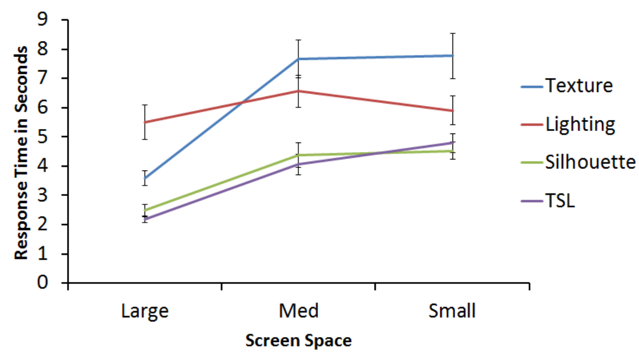

Figure 5: *Response times for each artifact across screen space.*

4.2.3 Discussion

We found that participants' *ability to detect texture artifacts decreases very quickly with screen space*. Participants were as good at detecting lighting artifacts as silhouette artifacts, but were *faster at detecting silhouette artifacts*. There was no significant difference between participants' ability to detect silhouette, lighting and TSL. Lighting is important, but it has no cumulative effect when combined with silhouette. These lighting errors can be masked with normal mapping to eliminate them completely. This provides support for our first hypothesis, i.e., that distance has a different effect on different artifacts, in that *silhouette is the dominant artifact for simplification detection, especially at larger distances*.

Not only did animation have no effect on artifact detection accuracy, it did not delay artifact detection to any significant degree. This contradicts our second hypothesis, as *applying animations of different intensities had no effect on the perceptibility of any artifacts*. For this reason we did not conduct any further experiments testing a range of motion intensities. Motion across the screen has been shown to have an effect on sensitivity to update rate [McDonnell et al. 2007], but our metric will run as a preprocess and may not have any prior knowledge of motion characteristics.

The remainder of our experiments concentrate on silhouette errors and their detection rates. As the usage of normal mapping for human characters is commonplace, lighting artifacts are not an issue. The use of normal mapping means that lower polygon counts can be used for meshes at larger screen spaces. This causes silhouette errors to be the most common artifact for current generation character rendering.

5 Silhouette Detection

To assist in the development of a metric to position LOD human character meshes in a scene, we must determine the perceptibility

of differences between the LOD mesh and its full resolution counterpart. This data will allow us to inform or test our developed metric. We have shown that silhouette is the most salient artifact for distinguishing simplification, so we use meshes with silhouette simplification alone in this experiment. Our hypothesis is that there will be a correlation between the difference in the silhouette of two meshes and the screen space at which that difference will be noticeable. The results from our experiments in Section 4 allow us to concentrate on silhouette artifacts alone. We are searching for the Point of Subjective Equality (PSE), i.e, chance level.

5.1 Experiment Design

A 2AFC experiment task was again used, where participants were asked to choose whether the character on the left or the right had been simplified. We recorded both participant responses and reaction times. As in the artifact experiment, a gold standard mesh was displayed in the centre with one mesh to the left and one to the right, either of which was simplified. Participants were asked to select the simplified character, indicating their choice with a mouse click, (left or right). The trial advanced after the participant made their selection.

The first condition tested was *screen space* (6), which were chosen to represent a full range of silhouette perceptibilities, from very obvious to extremly difficult to detect. At the lowest screen spaces, the participant should be guessing. We also tested a greater number of *model* (3), in order to capture as much data for a range of silhouette artifacts as possible. One female and two male character models were used.

Each model had a simple idle animation applied, to avoid using static models. We know from the initial experiment that animation has no effect on the perceptibility of silhouette artifacts. Participants viewed 144 trials, 3 models * 2 simplification types * 6 distances * 4 repetitions, in a randomised order to avoid ordering effects.

5.2 Results

Seventeen volunteers (10M,7F) took part in the experiment. We averaged all the participants' data over repetitions. A three way repeated measures ANOVA was conducted on the data set. All posthoc analysis was performed using Newman-Keuls comparisons of means.

5.2.1 Accuracy

First we examined the results for participant accuracy. A main effect of screen space was found, as expected ($F_{5,85} = 46.326, p < 0.00001$), see Figure 6. Participant accuracy decreased proportionately with screen space, with average accuracy as high as 96% at the largest screen space, dropping to 50% (i.e., chance) at the smallest screen space. This confirms that we chose a good range of screen spaces.

Overall, participants' average accuracy with different models varied from 65% to 75%. This confirms that we chose a good range of different LOD simplifications for our character models.

6 LOD Saliency Prediction

The data obtained in Section 5 has been used to evaluate the image comparison utility PerceptualDiff [Yee et al. 2001] as a basis for a predictor of simplification saliency. PerceptualDiff is an image comparison utility that provides saliency evaluations of two dimensional images based on a model of the human visual system.

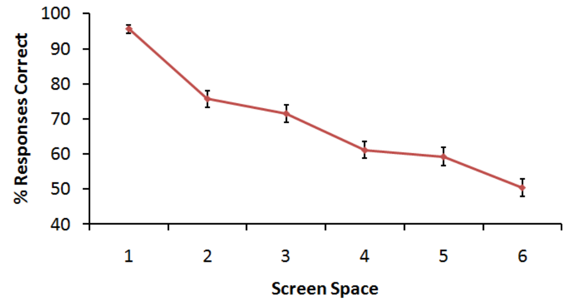

Figure 6: *Main effect of screen space in the silhouette detection experiment*

It takes into account how the sensitivity of the human visual system changes with increasing or decreasing light levels. It also takes into account spatial sensitivity, or how sensitivity to detail decreases with increasing frequency content. Finally, it models the masking effects introduced by the signal content of the background. Originally developed to detect differences in rendered images by testing if the number of pixels detected to be different are under a certain threshold, to make it suitable for use as a metric in three dimensions some alteration is required. Applying this method in three dimensions could provide us with a general metric with no need for experimental data as a seed.

6.0.2 Metric

To test the saliency of the differences between two meshes, we must use a non-view dependent method. Rendering a mesh from a number of angles and using PerceptualDiff to test the saliency of the differences would simply inform us of the number of pixel differences found between the renders. This is a method used in [Lindstrom and Turk 2000], which will tell the relative severity of a simplfication operation. However, to determine the distance at which this operation will go unnoticed we need a more accurate predictor.

A problem with using a view dependent metric to test a 3D model is that renders must be created from multiple angles, as in [Lindstrom and Turk 2000]. In this case, more rendered angles mean more accurate results. This can be very time consuming. Out method is to create a larger number of renders, but at a much smaller size, concentrating on the differences in small sections of the model.

We place two meshes, the original high resolution mesh and a simplified version, at the same position in our 3d scene, at a short distance from the camera without intersecting the clipping plane. The meshes are placed in the scene so that the scene details are available for the perceptibility calculation. Simply testing the meshes without the context of the surroundings would provide a less accurate measurement, as the masking effects of the background could not be taken into account.

Simply rendering the object to be tested from multiple angles and running an image difference calculation on this will not give a result of a sufficiently high granularity to be called accurate. We also wish only to test the silhouette area of the object. Our method is to test a much larger number of smaller areas, rendered so that the area lies on the silhouette of the mesh. This large number of smaller spaces is created by voxelising the space occupied by the mesh, with the voxel size covering one degree of visual angle. We choose this minimum measurement as this is the unit used by PerceptualDiff in its calculations. For each voxel that lies on the silhouette of the mesh, we create a render targeted at that voxel. To perform our PerceptualDiff test, the camera is rotated to look along the silhouette

at that voxel and the area rendered, first with the high resolution mesh, then with the low resolution mesh. The surrounding voxels are included in the render, as the PerceptualDiff calculation is based not on a single pixel, but includes data within one degree of each tested pixel.

For example, we use a display with a resolution width of 1920 pixels, covering a field of view of approximately 75 degrees, depending on the user's distance from the screen. Therefore, one degree of visual angle is covered by 25.6 pixels.

We then perform the PerceptualDiff calculation on the two rendered voxel areas. This calculation returns the number of pixels that would be perceived as different. The test is then repeated with increasing camera distance until the point is reached where no pixels are identified as perceptibly different.

We perform this test for each of the voxels lying on the mesh silhouette. From this we obtain a list of distances at which each tested part of the meshes will appear equivalent. To determine our final distance of imperceptibility, we have found that the average distance returned by all the voxels works well to determine the optimal placement of a mesh in the scene.

6.0.3 Usage

The number of renders and calculations performed by our method means that it takes a long time to run, so it is intended to run as a preprocess when used to compare a high resolution and low resolution version of a model in its entirety. Alternate uses for our method, such as a metric to drive a simplification method, would be localised to more specific areas of a mesh and therefore run at a much faster rate.

6.0.4 Results

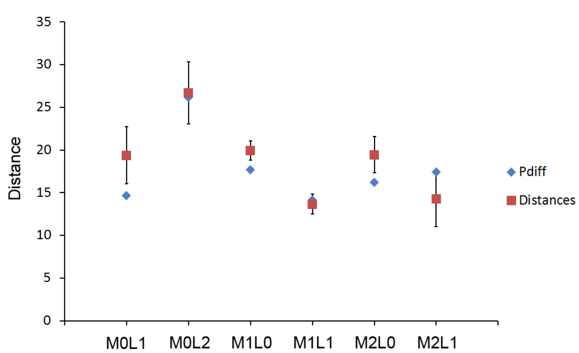

Figure 7: *Plot of observed PSE distances with standard error bars and distances computed by our metric.*

We have tested the models used in Section 5 with our metric. Our metric was initialised with the same viewing conditions as our perceptual experiments, a 1920 resolution width, a render camera field of view of 45 degrees, and a user field of view of 70 degrees. In each case, the models were placed within 3 world units (we use metres) of the PSE, showing that our method is a good indicator of the saliency of a simplified model.

In Figure 7 we can see that the results from our metric closely correlate with the observed PSE's for each comparison. These are the data points for our 6 mesh comparisons, between the high resolution mesh and its artist- and automatically-simplified counterparts. M corresponds to the model being tested, and L0 and L1 to artist- and automatically-simplified models. The metric calculation was performed under the same conditions as the perceptual experiment. The metric is based only on the human visual system and has no prior knowledge of the quality or detail of the meshes used in the perceptual experimentation. This data is used only to validate the results of our metric.

The data obtained from our participants was limited to a single point of view, whereas the data obtained from our perceptual metric is data averaged over all points of view. It is therefore a conservative estimate of the point at which a mesh should be placed. As with PerceptualDiff, parameters such as a user's field of view, or the signal content of the scene, can be adjusted to create a more or less conservative distance calculation. Further evaluation with a large number of meshes of varying detail and quality would be necessary to confirm the effectiveness of the metric.

7 Conclusions and Future Work

We have performed a series of perceptual experiments to determine the relative saliency of simplification artifacts. One of our most significant findings is that *silhouette* is the dominant artifact in mesh simplification, as it can be spotted more quickly and accurately than *texture* and *lighting* errors. This implies that the use of view dependent simplification would allow developers to make the most efficient use of their available polygon budget. We found that the intensity of the animation applied to a character did not have an effect on the perception of simplification errors. We know from [McDonnell et al. 2007] that movement across the display has an effect on the perception of update rate, but as we intend our results and metric to be used as a pre-process, this factor would need to be exploited as a run-time process.

Our developed metric applies not only to placing meshes in a scene, but could be used as a metric to simplify meshes to a specific polygon count, or to simplify a mesh as much as possible while ensuring the simplifications would not be visible beyond a certain distance. It also has the potential to be used as an aid to an artist while they simplify a model by hand, providing feedback during simplification operations.

A more in-depth alteration of the PerceptualDiff method to work in three dimensional space by inferring colour values and signal content from the surroundings of the simplified part of the mesh being tested would eliminate the need for expensive renders and provide significant speed advantages over our current method. This would also eliminate the necessity to place the model in a rendered scene, therefore allowing development of a utility external to the system in which the meshes are to be used.

Acknowledgements

We wish to thank the reviewers for their comments, and the participants in our experiments. This work was sponsored by Science Foundation Ireland as part of the Metropolis project.

References

CIGNONI, P., ROCCHINI, C., AND SCOPIGNO, R. 1998. Metro: measuring error on simplified surfaces. In *Computer Graphics Forum*, vol. 17, 167–174.

COHEN, J., VARSHNEY, A., MANOCHA, D., TURK, G., WEBER, H., AGARWAL, P., BROOKS, F., AND WRIGHT, W. 1996. Simplification envelopes. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, ACM, 119–128.

DALY, S. 1993. The visible differences predictor: an algorithm for the assessment of image fidelity. *Digital images and human vision 11*, 2.

GARLAND, M., AND HECKBERT, P. 1997. Surface simplification using quadric error metrics. In *Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., 216.

GIEGL, M., AND WIMMER, M. 2007. Unpopping: Solving the image-space blend problem for smooth discrete lod transitions. In *Computer Graphics Forum*, vol. 26, Citeseer, 46–49.

HOPPE, H. 1996. Progressive meshes. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, ACM, 99–108.

HOWLETT, S., HAMILL, J., AND O'SULLIVAN, C. 2004. An experimental approach to predicting saliency for simplified polygonal models. In *Proceedings of the 1st Symposium on Applied Perception in Graphics and Visualization*, ACM, 57–64.

LAND, M., AND HAYHOE, M. 2001. In what ways do eye movements contribute to everyday activities? *Vision research 41*, 3559–3565.

LEE, C., VARSHNEY, A., AND JACOBS, D. 2005. Mesh saliency. *ACM Transactions on Graphics 24*, 659–666.

LINDSTROM, P., AND TURK, G. 2000. Image-driven simplification. *ACM Transactions on Graphics (ToG) 19*, 3, 241.

LINDSTROM, P. 2000. *Model simplification using image and geometry-based metrics*. PhD thesis, Citeseer.

LUEBKE, D., AND HALLEN, B. 2001. Perceptually driven simplification for interactive rendering. In *Eurographics Workshop on Rendering Techniques*, 223.

LUEBKE, D., HALLEN, B., NEWFIELD, D., AND WATSON, B. 2000. Perceptually driven simplification using gaze-directed rendering. *University of Virginia Technical Report CS-2000-04*.

LUEBKE, D. 2003. *Level of detail for 3D graphics*. Morgan Kaufmann.

MACK, A. 2003. Inattentional blindness: Looking without seeing. *Current Directions in Psychological Science*, 180–184.

MCDONNELL, R., NEWELL, F., AND O'SULLIVAN, C. 2007. Smooth movers: perceptually guided human motion simulation. In *Proceedings of the 2007 Eurographics symposium on Computer animation*, Eurographics Association, 269.

RAMASUBRAMANIAN, M., PATTANAIK, S., AND GREENBERG, D. 1999. A perceptually based physical error metric for realistic image synthesis. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, 82.

RENSINK, R., O'REGAN, J., AND CLARK, J. 1997. To see or not to see: The need for attention to perceive changes in scenes. *Psychological Science 8*, 5, 368.

RUSHMEIER, H., WARD, G., PIATKO, C., SANDERS, P., AND RUST, B. 1995. Comparing real and synthetic images: Some ideas about metrics. In *Proceedings of the Sixth Eurographics Workshop on Rendering*, Citeseer, 213–222.

SCHERZER, D., AND WIMMER, M. 2008. Frame sequential interpolation for discrete level-of-detail rendering. In *Computer Graphics Forum*, vol. 27, 1175–1181.

SCHWARZ, M., AND STAMMINGER, M. 2009. On predicting visual popping in dynamic scenes. In *Proceedings of the 6th Symposium on Applied Perception in Graphics and Visualization*, ACM, 93–100.

TEO, P., AND HEEGER, D. 1994. Perceptual image distortion. In *International Symposium Digest of Technical Papers*, vol. 25, Citeseer, 209–209.

WATSON, B., FRIEDMAN, A., AND MCGAFFEY, A. 2001. Measuring and predicting visual fidelity. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM, 220.

WILLIAMS, N., LUEBKE, D., COHEN, J., KELLEY, M., AND SCHUBERT, B. 2003. Perceptually guided simplification of lit, textured meshes. In *Proceedings of the 2003 symposium on Interactive 3D graphics*, ACM, 121.

YARBUS, A., AND HAIGH, B. 1973. *Eye movements and vision*. Plenum press.

YEE, Y., AND NEWMAN, A. 2004. A perceptual metric for production testing. In *ACM SIGGRAPH 2004 Sketches*, ACM, 121.

YEE, H., PATTANAIK, S., AND GREENBERG, D. 2001. Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. *ACM Transactions on Graphics (TOG) 20*, 1, 39–65.