Pareto-Based Perceptual Metric for Imperceptible Simplification on Mobile Displays

Fan Wu† Emmanuel Agu † and Clifford Lindsay †

Department of Computer Science
Worcester Polytechnic Institute, Worecester, MA, 01609, USA

Abstract

Graphics on mobile devices has become popular because untethered computing is convenient and increases productivity. Mobile displays come in different resolutions that affect the scene Level-of-Detail (LoD) that users can perceive: smaller displays show less detail, making lower resolution meshes and textures acceptable. Mobile devices frequently have limited battery energy, low memory and disk space. To minimize wasting system resources, we try to render mobile graphics scenes at the lowest LoD at which users do not perceive distortion due to simplification. We call this LoD the Point of Imperceptibility (PoI). The PoI LoD depends on several factors including screen size, scene geometry and lighting levels. We propose a perceptual metric that identifies the PoI LoD of a target mobile display and accounts for object geometry, lighting and shading. Our perceptual metric generates a screen-dependent pareto distribution with a knee point that corresponds to the PoI. We employ wavelets for simplification, which gives direct access to the mesh undulation frequency that we then use to parametrize the perceptual CSF curve.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Mobile Graphics, Perceptual Simplification Metrics, Wavelets, Multiresolution analysis

1. Introduction

Graphics on mobile devices has become popular because untethered computing is convenient and increases productivity. On-site consultants, such as architects, can use drawings on cell phones and PDAs to discuss preliminary designs with clients. Other mobile graphics applications include educational animations, virtual product catalogs, and mobile games. Our work focusses on the client-server scenario shown in figure 1. High resolution graphics meshes and textures are stored on the server, and then simplified when requested by a mobile client. Simplification is crucial to reduce the consumption of limited battery energy and memory on mobile devices during rendering. Additionally, increasing mesh and texture resolution beyond a certain Level-of-Detail (LoD), users cannot perceive improved visual realism. We call this LoD the *Point of Imperceptibility (PoI)*.

† {wuf,emmanuel,clindsay}@cs.wpi.edu

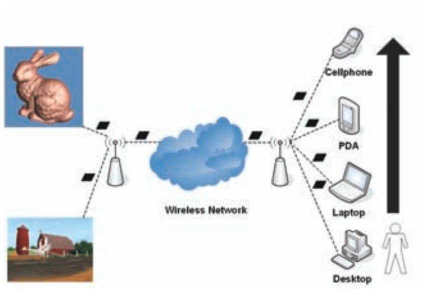

Figure 1: *Mobile graphics scenario*

Our goal is to render meshes and textures that are close to their PoIs. Our experiments show that the LoD users can perceive depends on the screen resolution; smaller screens show less detail (lower PoI). For instance, we found that for a given mesh, a laptop's display had a PoI of 20K faces, while a cell phone's PoI was 5K faces for the same mesh. This represents a 4x change in the acceptable LoD level based on screen resolution. Previous work has neglected to directly relate selected LoD levels with target mobile screen resolution. Other factors that influence PoI such as mesh features and object distance from the screen have been studied in the literature. This paper focusses primarily on how PoI changes with screen resolution. We develop an analytic metric that computes the PoI of both meshes and textures (images). Due to the large variety of mobile screen resolutions available, analytically computing the PoI is preferred to manual calibration for each mobile screen resolution. Our metric determines the mesh LoD corresponding to the PoI and takes as input 1) the original mesh LoD 2) mobile screen resolution and 3) mesh lighting information.

2. Background and Related Work

Multiresolution using wavelets: Our mobile graphics framework uses wavelets to represent meshes and textures at various LoDs *(called multiresolution analysis)* [Lou94]. During wavelet decomposition, a mesh is iteratively subdivided to approximate a curved surface. Decomposition generates a coarse base mesh, along with wavelet coefficients that refine the base mesh. During reconstruction, the LoD of the base mesh is increased by iteratively adding wavelet coefficients. Our mesh simplification implementation is based on the Loop wavelet transform. We also consider multiresolution of textures using wavelets using a 2D Haar wavelet decomposition. Alternate vertex-based surface-to-surface simplification metrics such as Quadric Error Metrics [GH97], have also been proposed. We simplify using wavelets for two main reasons. First, wavelet decomposition aggressively compresses (over 100x) meshes for fast transmission over low-bandwidth wireless links. Secondly, wavelet decomposition yields mesh and texture undulation frequencies that we use in the perceptual component of our PoI metric. In fact, using the wavelet filter frequency to directly parametrize the perceptual *Contrast Sensitivity Function (CSF)* curves is one of our main contributions.

Perceptual simplification metrics: Surface-to-surface geometric metrics express the deviation of a simplified mesh surface from the original mesh. To account for how lighting shading, texturing and other visual effects affect the perceptibility of simplification artifacts, perceptual metrics have been developed [Red01], [LT00], [LH01]. Rather than measure geometric error in object space, perceptual metrics focus on how mesh and image LoDs affect the contrast and frequency of pixel color changes. Most perceptual simplification metrics are based on the CSF, which is a plot of contrast and color perceptibility thresholds of human vision. The highest contrast and lowest spatial frequency exhibited by a rendered image at the pixel level determines its location on the CSF curve. As a major contribution of our work, we use the frequency of the wavelet decomposition filters directly as the frequency of the CSF curves.

Our work is unique because it integrates the target screen resolution as a parameter for LoD selection. Previously published simplification metrics that did not explicitly consider target screen size would erroneously select the same PoI LoD for a cell phone as it would select for a laptop. Image-driven simplification proposed by Lindstrom and Turk [LT00] is an exception. However, image-driven simplification compares pixel-level differences between pre-rendered versions of the original mesh and its simplified version on a target screen. Pre-rendering takes time, making their algorithm inappropriate for fast LoD selection.

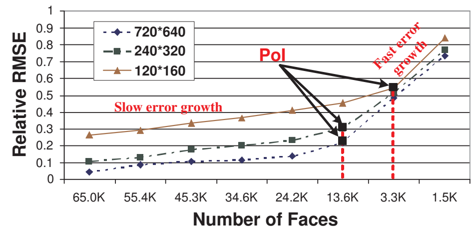

Figure 2: *Screen-dependent pareto plots using PoI metric*

3. Overview of Our Approach

Our proposed metric for imperceptible simplification extends the work of Tack *et al* [TLCL05]. Tack *et al* expressed the surface-to-surface L_p norm error due to mesh simplification but did not explicitly address the perceptibility of errors on diverse screen resolutions, or consider the effects of lighting on the final rendered mesh. We develop our PoI in two distinct phases. First, in section 4.1 we develop a geometry-only component without considering lighting effects. Next, in section 4.2, we integrate perceptual components that account for scene lighting. A preview of our final results is now used to build intuition. Figure 2 shows pareto plots generated using our metric. Three plots are shown corresponding to three different mobile screen resolutions (laptop:720x640, PDA:320x240, cellphone:160x120). The points on the curves are screen-dependent simplification errors calculated using our PoI metric. Thus, our metric generates a family of plots, one for each target screen resolution and we shall show that:

Hypothesis 1: Each of the curves in figure 2 follows a pareto distribution. Starting with the original mesh on the left of the plots, relatively low errors are generated as LoD is reduced up until a *knee point*. Beyond the knee point, reducing LoD levels results in sharp increases in error. We conjecture that a) users cannot perceive simplification errors to the left of the knee point b) the knee point is the Point of Imperceptibility (PoI); and c) To the right of the PoI (knee point), users can perceive simplification errors.

Hypothesis 2: Based on the results of Luebke and Hallan, we conjecture that lighting will further reduce the perceptibility of errors, essentially lowering the PoI.

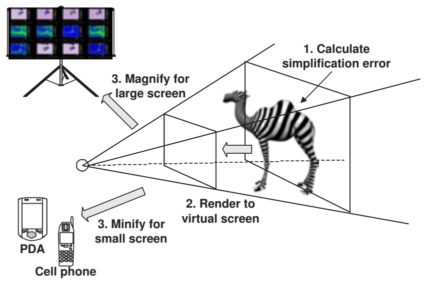

Figure 3: *Steps in deriving our PoI metric*

4. PoI Error Metrics

4.1. Geometry-only PoI Metric

This section derives the first part of our metric that considers only the distortion of mesh geometry without factoring in the effects of lighting. Our derivation has three steps: 1) Calculate mesh distortion due to simplification; 2) Render the simplified mesh to a large virtual screen M_1; 3) Minify blocks of pixels of M_1 to a pixel of the mobile display M_2. We can magnify if $M_2 > M_1$ as in a large tiled display. For screen-aligned images, only step 3 is performed. Figure 3 summarizes the steps to derive our metric. Equation 1 is our PoI metric for geometry only.

$$l^p(S_1,S_2) = \underbrace{(1-\frac{F_2}{F_1})\frac{\sum_{i=0}^{F} A(T_i) l^p(T_i,S_2)}{\sum_{i=0}^{F} A(T_i)}}_{Object-space} + \underbrace{E_p}_{Screen-space} \quad (1)$$

where F_1 is the number of triangles in surface S_1, F_2 is the number of triangles in surface S_2. If $F_1 < F_2$, we can rewrite $1-\frac{F_2}{F_1}$ as $1-\frac{F_1}{F_2}$. The first part of Equation 1 deals with surface-to-surface LoD simplification errors in object space and the second term (E_p) deals with pixel-level minification errors caused by rendering to different screen resolutions. Rendering a high-resolution mesh to a small screen generates errors in both terms. A screen-aligned texture incurs errors only due to the second (E_p) term. Likewise, if a given mesh LoD (no surface simplification) is rendered to two different screen sizes, errors are only generated in the second term. For a target mobile display width, W (in pixels) and height H (in pixels), the term E_p is defined as:

$$E_p = \sqrt[p]{\frac{1}{W_2 \times H_2} \sum_{i=1}^{W_2 \times H_2} (\frac{W_2 \times H_2}{W_1 \times H_1} \sum_{j=1}^{\frac{W_1 \times H_1}{W_2 \times H_2}} \sqrt[p]{S_p})^p} \text{ where}$$

$$S_p = \frac{1}{3}[(\frac{R_{i2}-R_{j1}}{256})^p + (\frac{G_{i2}-G_{j1}}{256})^p + (\frac{B_{i2}-B_{j1}}{256})^p] \quad (2)$$

where W_1 and H_1 are the width and height of screen M_1 and W_2 and H_2 are the width and height of screen M_2. We assume that $W_1 > W_2$. Otherwise, W_1 and W_2 should be interchanged. In our system, we use relative Root Mean Square Error (RMSE) ($p = 2$). In Equation 2, we calculate the screen space RGB error pixel by pixel and normalize it. S_p calculates the average RMSE of RGB values between one pixel on the smaller screen and the corresponding group of pixels on the larger screen. Figure 2 shows plots of our PoI metric considering three screen sizes.

4.2. Perceptual Metric

In this section we extend our PoI metric to account for lit meshes. First, we note that effects such as lighting and shading can reduce the perceptibility (sharpness) of mesh edges, making distortion less visible. We model this reduction in error perceptibility as passing the original mesh (sharp) through a filter that removes some distortion. To account for the error masking caused by lighting, we multiply our geometry-only expression (equation 1) by a factor $M_p(S_1,S_2)$. Our new PoI expression is:

$$l^p(S_1,S_2) = [(1-\frac{F_2}{F_1})\frac{\sum_{i=0}^{F} A(T_i) l^p(T_i,S_2)}{\sum_{i=0}^{F} A(T_i)} + E_p] \times M_p(S_1,S_2) \quad (3)$$

Next we derive an expression for $M_p(S_1,S_2)$. The response of the human visual system can be expressed as a convolution of the input stimulus with the visual cortex's impulse response. To determine the perceptibility of a lit mesh, we determine the eye's response by multiplying the wavelet transform of the mesh by the CSF. Mannos and Sakrison [JD74] experimentally modeled the CSF as $C_s(f_s) = [0.0499 + 0.2964 f_s] \times exp[-(0.114 f_s)^{1.1}]$ where f_s is spatial frequency in cycles per degree.

During wavelet decomposition we determine the filter frequency ranges that correspond to each LoD, and then multiply the corresponding wavelet coefficients with the CSF. For each frequency range a *sensitivity weight,* C_m is computed by integrating the CSF curve over that frequency band. Wavelet transformation involves the iterative application of L, a lowpass filter and H, a high-pass filter. Thus, by applying H to a discrete input with bandwidth $(0,\pi)$, a level of coefficients with bandwidth $(\pi/2,\pi)$ is acquired. Thus, after m iterations, the weight for level m is $C_m = \int_{F_m} CSF(\omega) d\omega / A(F_m)$ where F_m is the frequency subband $\left(\frac{\pi}{2^m}, \frac{\pi}{2^{m-1}}\right)$ and $A(F_m)$ is the width of the band. During the rendering step, for each LoD, we track which group of screen pixels are modified by the wavelet coefficients at that level. Thus, for each pixel (i,j) on the mobile display, we multiply the wavelet coefficients in a given frequency band with the contrast sensitivity weight corresponding to that frequency band giving $D_1(m,i,j) = C_m W(1,m,i,j)$. C_m is the contrast sensitivity weight and $W(m,i,j)$ is the wavelet coefficient at level m and pixel location (i,j). Thus, our perceptual metric is:

$$M_p(S_1,S_2) = \frac{\sum_{m,i,j} |D_1(m,i,j) - D_2(m,i,j)|^2}{N_h \times N_v} \quad (4)$$

where D_1 and D_2 are error values of pixel i, j, when considering level m of the wavelet coefficients. N_h and N_v are

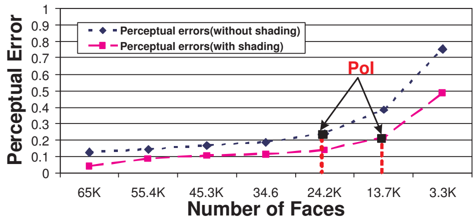

Figure 4: *Curves with shading and without shading*

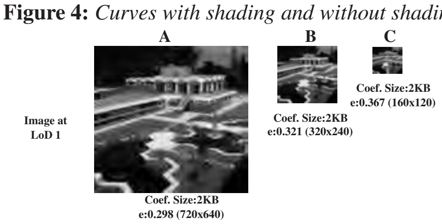

Figure 5: *Coefficients file size and Relative RMSE*

the number of pixels in horizontal and vertical directions on the small screen. Figure 4 shows our plots using equation 3. The errors with lighting and shading are clearly smaller than the errors without lighting and shading.

4.3. Texture simplification using our PoI metric

Our proposed metric can also compute the PoI of a high resolution texture. To compute texture PoI, we note that since there is no surface-to-surface error for a 2D texture, we only calculate the screen-space term in equation 3, and the E_p and $M_p(S_1,S_2)$ terms. Figure 5 shows images generated using different sizes of coefficients files.

5. Metric Validation and Analysis

User Studies: Having derived our PoI metric, we need to validate that it accurately selects the LoD at which real users stop perceiving increases in mesh or image resolution. Our approach was to 1) generate a series of mesh and image LoDs 2)Use our PoI metric to compute the mesh or texture LoD corresponding to the PoI and 3) Experimentally determine the PoI by showing the rendered LoDs to real users. Our results showed that PoI metric indeed worked correctly. Our user studies involved 84 participants. In our study, several LoDs of several models were rendered at three different screen sizes (laptop:720x640, PDA:320x240 and cellphone:160x120). Figure 6 shows one set of resolutions of a rendered bunny for screen size 240x320 pixels.

Performance Gains using PoI: Using a mesh or texture at the PoI instead of its original resolution improves usage of battery power, CPU cycles, memory and other mobile resources. We measure encoding, transmission and decoding times, and quantify potential battery energy savings by using a lower resolution mesh. We use our tool for measuring energy usage of mobile applications. Table 1 presents sample resource savings from using our metric.

Figure 6: *Rendered meshes at different LoDs in user study*

$NumofFaces$	13K (PoI)	65K (orig.)	Saved
$T_{Trans.}$	1.23ms	7.03ms	82.5%
$T_{Decoder}$	463ms	832ms	44.4%
$Power\ Cons.$	12865mw	33298mw	61.4%

Table 1: *Saved Resources for mesh*

6. Conclusion and Future Work

We presented a wavelet-based framework for scalable graphics transmission and rendering. We derived a Point of Imperceptibility (PoI) error metric that accurately selects the lowest acceptable mesh (or image) resolution for a mobile display's resolution. Our PoI metric considers mesh geometry and also has a perceptual component that accounts for the effects of lighting and shading. We performed user studies to validate our metric, and demonstrated resource savings. As future work, we are considering several directions. In creating our PoI pareto plots, the meshes were only from one view direction, and most meshes have different features from different angles. To account for the view dependence of our PoI metric, we propose calculating our PoI metric for each object from multiple view points around the original mesh, and then combining these values. Finally, we would like to consider error perception on texture-mapped objects.

References

[GH97] GARLAND M., HECKBERT P.: Surface simplification using quadric error metrics. *Proceedings of Siggraph 97* (1997), 209–216.

[JD74] J.L.MANNOS, D.J.SAKRISON: The effects of a visual fidelity criterion on the encoding of images. *IEEE Trans. Info Theory, 20(3)* (July 1974), 525–536.

[LH01] LUEBKE D., HALLEN B.: Perceptually driven simplification for interactive rendering. *Proc. Eurographics Rendering Workshop* (2001), 7–18.

[Lou94] LOUNSBERY M.: Multiresolution analysis for surfaces of arbitrary topological type. *PhD diss., U. Washington* (1994).

[LT00] LINDSTROM P., TURK G.: Image-driven simplification. *ACM Trans Graphics 19, 3* (2000), 204–241.

[Red01] REDDY. M.: Perceptually optimized 3d graphics. *IEEE Comp. Graph. & Appl. 21(5)* (2001), 68–75.

[TLCL05] TACK N., LAFRUIT G., CATTHOOR F., LAUWEREINS R.: Pareto based optimization of multiresolution geometry for real time rendering. *Proc. ACM Web 3D* (2005), 19–27.