# Perceptual Issues in Substituting Texture for Geometry

Holly Rushmeier[a], Bernice E. Rogowitz[a], Christine Piatko[b]

[a]IBM TJ Watson Research Center, P.O. Box 704, Yorktown, NY, USA
[b]Johns Hopkins Applied Physics Laboratory, 1100 Johns Hopkins Rd., Laurel MD, USA

## ABSTRACT

An important goal in interactive computer graphics is to allow the user to interact dynamically with three-dimensional objects. The computing resources required to represent, transmit and display a three dimensional object depends on the number of polygons used to represent it. Many geometric simplification algorithms have been developed to represent the geometry with as few polygons as possible, without substantially changing the appearance of the rendered object. A popular method for achieving geometric simplification is to replace fine scale geometric detail with texture images mapped onto the simplified geometry. However the effectiveness of replacing geometry with texture has not been explored experimentally.

In this paper we describe a visual experiment in which we examine the perceived quality of various representations of textured, geometric objects, viewed under direct and oblique illumination. We used a pair of simple large scale objects with different fine-scale geometric detail. For each object we generated many representations, varying the resources allocated to geometry and texture. The experimental results show that while replacing geometry with texture can be very effective, in some cases the addition of texture does not improve perceived quality, and can sometimes reduce the perceived quality.

**Keywords:** computer graphics, geometric simplification, texture, perception

## 1. INTRODUCTION

Graphics systems can make intensive use of available computational resources. Given the complexity and detail of geometric models available, trade-offs must be made in graphics rendering to balance between interactivity and perceptual quality. Advances in networking have also focused attention on the use of rendering approximations in order to enable the efficient use of bandwidth-constrained resources for distributed graphics applications. Different choices of rendering approximations will affect perception in different ways. Thus, it is desirable to develop perceptual measures to understand the impact of texture and geometry approximations. In this paper we study the implications of substituting texture for geometry.

Many geometric simplification algorithms have been developed to provide interactivity and reduce bandwidth requirements. The goal of these algorithms is to achieve a perceptually acceptable representation with minimal resource requirements. Ultimately, both the acceptable perceptual quality and resource requirements are dependent on the particular computing environment. The acceptable perceptual quality would be quite different for an e-commerce application where a customer is examining an object to make a decision whether to buy it, versus a game environment in which the player may be willing to accept a level of artificiality. The resource limitations would be quite different for an application in which transmission over the Internet is the bottleneck, versus an application where interactive frame rate on a specific workstation is required. Before we can address these issues, however, we need to enhance our understanding of the interplay of graphics resource allocation and perceptual quality.

Very little work has been done to explore the perceptual effects of different simplification schemes. For simple sprite representations, Horovitz and Lengyel[1] considered trade-offs in the perceptual and computational costs. Watson et al.[2] consider naming time as a predictor of the perceptual quality for various levels of geometric simplification. They considered only triangle reduction – not methods which replace geometry with texture. They found that both geometric and image metrics correlated poorly with the quality indicated by naming time.

The goal of our work is to provide a general framework for evaluating the perceptual effects of geometric simplification, and geometry-texture trade-offs. We do not specify a task or a particular network/workstation setting. Instead, we create simple, well-controlled stimuli, explicitly varying the geometry, texture and illumination. We introduce a simple scaling procedure as a method for exploring fundamental questions in the area of geometry/texture allocation. Does texture replacement always result in better perceptual quality for a given resource allocation? Are certain types of geometry more suitable for texture

Further author information: (Send correspondence to H.R.)
H.R.: holly@watson.ibm.com, B.R.: rogowtz@us.ibm.com, C.P.: Christine.Piatko@jhuapl.edu

**Figure 1.** In computer graphics, a solid object (leftmost image) is often represented as a dense mesh of triangles (second image from left.) To facilitate interactive display performance, objects are often simplified (third image from left), to reduce the number of triangles (rightmost image), while attempting to preserve the appearance of the object.

replacement? Are there different rules for different classes of objects, different viewing conditions? What is the impact of lighting?

To evaluate the degree to which adding texture could compensate for simplifications in geometry, we measured the perceived fidelity of two objects, a smooth sphere and a crinkly sphere, lit from the front or obliquely from the side. In each experiment, twelve test stimuli were created using three levels of geometric resource and four levels of texture resource (including a no-texture condition). In these experiments, psychophysical data were obtained from eight observers who rated the degree to which each of the test stimuli compared with a comparison stimulus with full geometry and full texture resource.

## 2. GEOMETRIC SIMPLIFICATION AND TEXTURE MAPPING

A wide range of representations can be used to model the geometry of an object – including tensor spline surfaces and quadric patches. For interactive display, typically all representations are converted to a set of polygons approximating the surface. For curved or complicated objects, thousands of polygons may be used. Usually networks of triangles are used, such as the example shown in Fig. 1 since triangles are guaranteed to be non-self-intersecting, facilitating hidden surface calculations. The time to display a surface depends directly on the number of triangles. Interactive display rates require that the number of triangles be limited. To maintain interactive rates, many algorithms have been developed to reduce the number of triangles in a given model while maintaining the visual appearance.

### 2.1. Simplification

Many different simplification methods have been proposed. Here we present just a brief overview. A comprehensive discussion can be found in an article by Cignoni et al.[3] Virtually all of the methods are associated with parameters that affect appearance, but none are based on perceptual data. One general class of methods creates simplified geometries by a series of small incremental simplifications. Popular incremental approaches are vertex removal and edge collapse. In vertex removal methods such as the one proposed by Schroeder et al.[4] one point in the mesh is removed, and new triangles are defined to fill the resulting hole. In edge collapse methods such as the one developed by Gueziec,[5] one edge is removed, and the two vertices at the ends of the edge are merged into one.

In any type of incremental method, a choice needs to be made to determine the next best vertex or edge to remove. The choice is generally made on the basis of a geometric metric. Common metrics include selecting the change that results in the smallest change in surface area, or selecting the change that results in a surface that is the closest to the original vertices.

Other methods work globally, such as clustering or volumetric methods.[6] These impose a maximum length scale on the simplified model by coalescing all vertices within a prescribed distance to one another into a single vertex. These methods have the advantage that they allow the object topology to change.

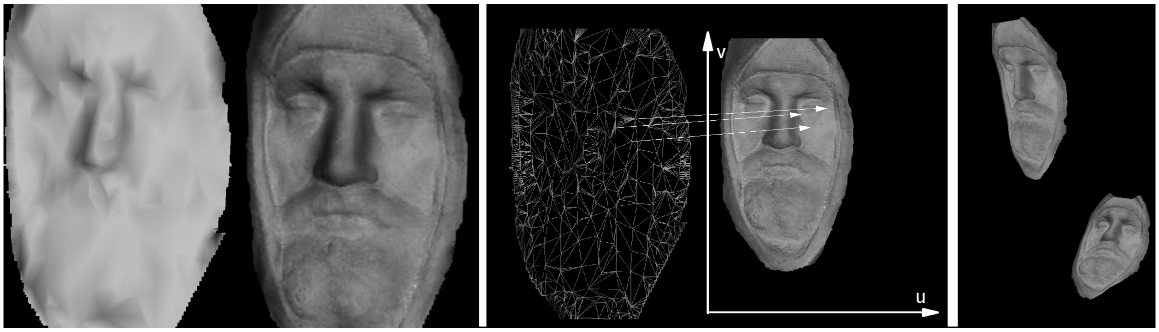


**Figure 2.** A simplified geometry can be given a detailed appearance using a texture map, as shown in the before and after images on the far left. At each vertex $(u, v)$ indices are stored , which are locations in an image, as shown in the center diagram. Details inside the triangle are "painted" on the triangle using the image pixels within the corresponding triangular area on the texture image. A texture-mapped object can be viewed from any direction (right-most image).

Recently more attention has been paid to visual impression rather than purely geometric measures. On one extreme some efforts seek to find the simplest symbolic representation of an object – for example representing a hand with a skeleton of five line segments. At the other extreme, some efforts attempt to produce a pixel by pixel identical image of the object by selecting the correct level from a hierarchical description of the object.[7] The object is rendered to the image plane progressively, at each step refining the representation until further refinement does not influence the image. This approach can account for both the particular view and lighting conditions. However, it does require the observer to wait while the object is progressively rendering on the screen.

## 2.2. Replacing geometry with texture

The most successful efforts to date to maintain visual appearance with a relatively small number of triangles make use of texture mapping. Texture maps provide the illusion of detail. Even though texture is a flat image, when they are attached to the geometry new views can be obtained. Texture maps that contain color and fine scale detail associated with lower resolution geometry have been used for decades in computer graphics,[8] [9] .

In texture mapping, each vertex in a model is associated with a $(u, v)$ coordinate pair, where $u$ and $v$ each vary from zero to one. As diagrammed in Fig. 2, the $(u, v)$ coordinates correspond to a location in an image. The detailed value for any point in the geometry is found by interpolating the $(u, v)$ coordinates of the vertices and looking up the corresponding location in the texture image. Storing detailed data in image maps is efficient because images do not require the explicit storage of positional or connectivity data. Images to represent detail may simply store colors. Bump maps are a type of texture image in which each pixel value represents a small height deviation of the detailed surface from the underlying surface. Normals maps store a vector quantity at each pixel representing surface normal.

There have been many successful methods for simplifying large sections of complex scenes by generating a texture map for a single polygon, or small number of polygons,[10] [11] . A major feature of the Talisman graphics architecture proposed by Torborg and Kajiya[12] was the representation of three dimensional objects as small polygons, or "sprites" that could be translated and warped. Soucy et al.[13] moved beyond mapping textures to single polygon by developing a systematic method for deriving color texture maps to represent color detail on a simplified version of a dense triangle mesh.

Display hardware for texture mapping colors is available at relatively low cost on personal computers, and is routinely used in computer games. A difficulty with using a color image for a texture map is lighting the object. There are two options: compute the lighting based on the simplified geometry, or use a color image that includes the effect of light. The problem with the first option is that the lighting variations make the simplified geometry visible, the problem with the second option is that the lighting cannot be changed.

Display hardware that performs dynamic lighting changes using bump or normals maps is rapidly becoming more widely available. The hardware interpolates the detailed value of the surface normal from maps and performs the dot product with the light direction in real time. The calculation performed in hardware to compute a detailed lit texture is diagrammed in Fig. 3. Cohen et al.[14] recognized the importance of dynamically relighting details and developed a method for computing both normals and color maps for a simplified version of a complex mesh.

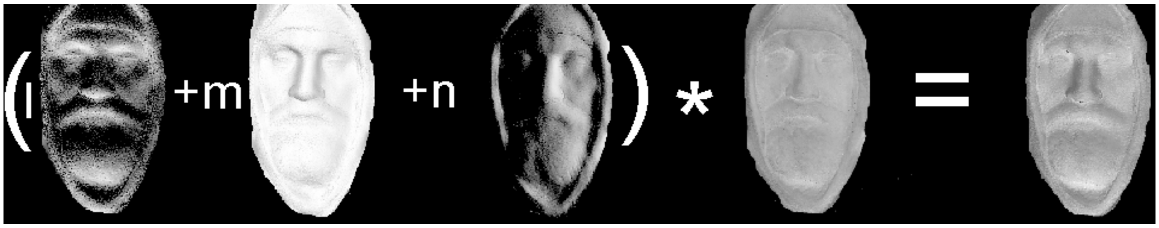


**Figure 3.** Texture maps can be dynamically relit by taking the dot product of light source direction $[l, m, n]$ (as shown in first three images from left), multiply by the image storing surface color (or simply albedo) and producing a new image (far right).

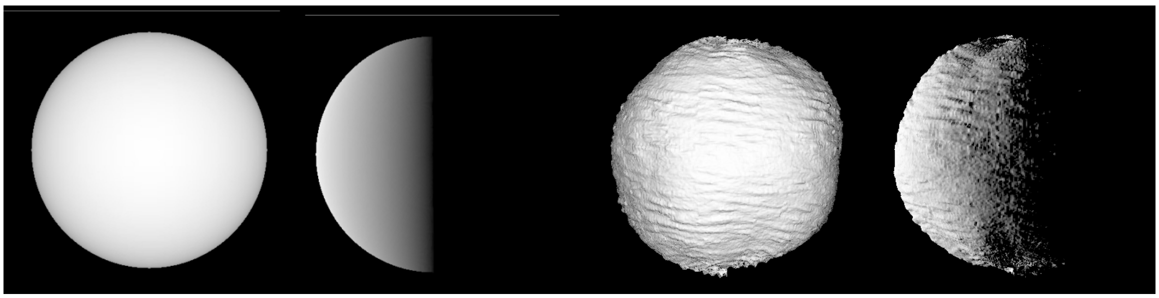

**Figure 4.** The objects used in our experiments were a sphere lit from the lef (leftmost image), a sphere lit from the right (image second from left), a crinkled sphere lit from the front (image thrid from left) and a crinkled sphere lit from the left (rightmost image.)

## 3. EXPERIMENTAL DESIGN

We examined perceptual trade-offs in replacing geometry with texture for objects of uniform color and surface finish. We examined abstract shapes, to avoid issues of semantic interpretation. We constructed our experimental stimuli in order to have object representations that allowed us to control the geometric and texture simplifications separately. Suitable numerical object representations were not readily available, since existing collections of test objects did not include the two-dimensional parameterization required for the texture maps. A few texture mapped objects were available on the Internet, but the textures were in the form of one texture per triangle, and so were not suitable for filtering to varying resolutions. For our experiment, test objects were constructed using IBM Visualization Data Explorer (DX). DX is a flexible visual programming system, and is available without cost as open source from http://www.opendx.org/

The stimuli used were a sphere and a sphere with a crinkled surface, each represented by varying levels of geometry and texture, viewed under direct or oblique lighting. We used a psychophysical scaling procedure to measure the perceived fidelity of each representation of each object, relative to a "perfect" reference representation.

### 3.1. Preparation of Stimuli

The two test objects are shown in Fig. 4. The objects were defined starting with a 512 x 512 grid of points, and warping the grid into a sphere with unit radius. Normals were computed at each grid vertex, a dot product was performed with lighting direction, and the results were stored in a 512 x 512 image to be used as a texture map. Texture $(u, v)$ coordinates were stored at each vertex, with a one-to-one correspondence between object vertices and texture image pixels for the full representation.

The geometry was simplified using the DX *Simplify* module which is an implementation of Gueziec's geometric simplification.[5] Gueziec's method produces a simplification that guarantees that the resulting surface is within a specified distance of all the original vertices. *Simplify* maintains the $(u, v)$ indices at the vertices remaining after each simplification, so all version of the sphere could be texture mapped with all versions of the texture image. Lower resolution versions of the lit normals maps were computed with box filtering to reduce aliasing artifacts.

The distribution of crinkles on the second object was obtained by painting intensity variations on a 512x512 image. This grey scale image was imported into DX, and the grey levels were interpreted as radial perturbations ranging between $0$ and $0.1$

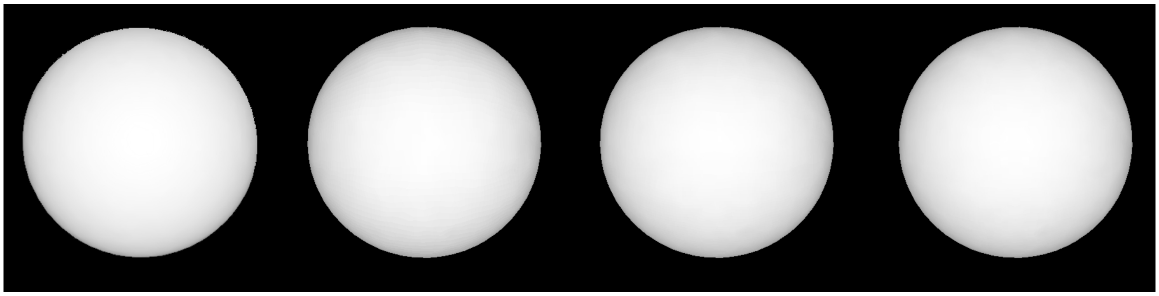

**Figure 5.** The medium geometry sphere lit from the front, with geometry only (far left), with simple texture (second from left), with medium texture (second from right) and with full texture (right).

on the unit sphere. After the full crinkled geometry was constructed, the normals maps and simplified geometries and images were computed as for the case of the sphere.

For each of the objects we generated two levels of simplification by experimenting with the distance error parameter in Gueziec's method to produce simplified objects that spanned a range of visual quality. For each object then we have representations we will refer to as full, medium and simple. We measure the resource required for each geometry as the compressed ascii file size of the DX representation. Clearly, this size would vary depending on the particular format used, and the particular geometry viewer used. The levels of geometric simplification were selected based on the complexity of the model. For the smooth sphere, the geometric resource was 4.4 Mb for the "full" geometry, .093 Mb (a reduction by a factor of 47) for the "medium" geometry and .047Mb (a further reduction by a factor of 2) for the "simple" geometry. For the crinkled sphere, the geometric resource was 14.4 Mb for the "full" geometry, 6.39 Mb (a reduction of a factor 2.3) for the "medium" geometry and .24 Mb (an additional reduction by a factor of 26.6) for the "small" geometry.

The same four levels of texture were selected for all four stimuli: the original 512x512 image as the "full" case, a 256x256 (factor of 2 reduction in resolution) image "medium" case, and 64x64 (additional factor of 4 reduction in resolution) as the "simple" case. The resource required is the uncompressed tif file for each of these images, since the texture memory required is the expanded image size. In terms of memory the "full" image is .787 Mb, the "medium" image is .197 Mb (a factor of 4 reduction), the "small image is .0122 Mb (an factor of 16 reduction) and 0 Mb for no texture. We use this size measure to illustrate the general framework for evaluation; the correct measures to use would depend on the specific task and computing environment.

We consider the effect of two lighting conditions, illustrated in Fig. 4. The first condition is with light parallel to the view direction. This is typical of the "headlight" lighting used in most computer graphics systems used to view individual objects. The second lighting condition was a light perpendicular to the view direction that causes a relatively abrupt, attached, shadow. Such shading could occur when an object is used as part of a virtual environment.

For each object and lighting condition, images were computed in which the object height corresponded to approximately 370 pixels on an image with black background. This size was chosen so that a few of the images could be displayed simultaneously on a 1280 x 1024 monitor. Twelve images were computed for each object/lighting combination, covering all combinations of simple/medium/full geometry and no/simple/medium/full texture. Samples of the various combinations are showin in Figs. 5 and 6. For the geometry only case, the image was formed by taking the dot product of vertex normal and light source direction at each vertex, and then using Gouraud shading for the smooth surface display. Note that for the full geometry case, the result of this technique is pixel-by-pixel identical to mapping the full texture on the geometry.

### 3.2. Procedure

Eight observers participated in the experiment. They were told that they would be asked to score images of various representations of an object relative to a "perfect" object (i.e. the image of the full geometry) using a scale of 0 to 100. A score of 100 indicated a perfect match. The observers were shown a variety of the comparisons they were to make from the four tests sets, and told to try to assign 0 to the worst match or matches. The observers were then asked to scale each of the twelve images in each of the 4 test sets, with the 4 test sets presented in a random order. The observers viewed a full-size, "perfect" image in one corner of the display screen, and a matrix of thumbnails of the objects to be rated. For each evaluation, the observer clicked on the thumbnail to expand it to full size. The observers were free to score each of the 12 objects in any order they liked, and were given no time limit. The time to complete all 48 comparisons was typically half an hour.

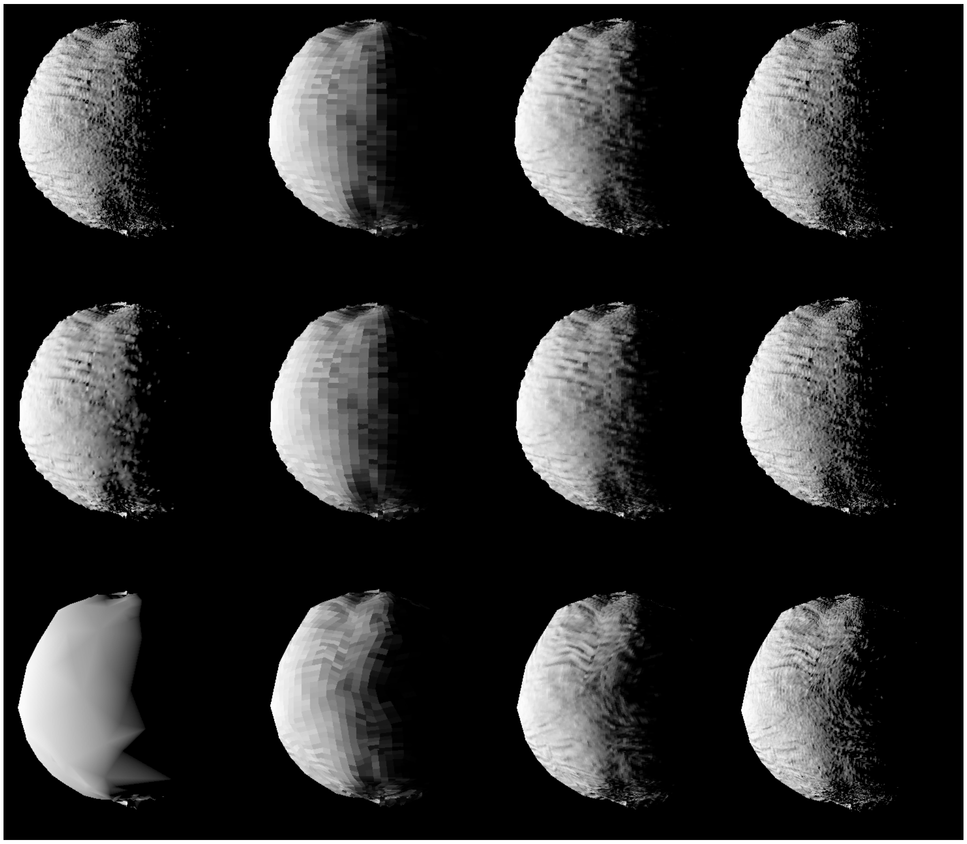

**Figure 6.** All of the representaions of the side lit crinkled sphere presented to viewers: geometry varies from full (top row) to simple (bottom row), texture varies from none (left column), to simple (second column from left) to full (right column).

## 4. RESULTS

The results of the observer scores are summarized in Figs. 7 to 9.

### 4.1. Validity of Results

Figure 7 shows the distribution of scores assigned by the eight observers in this experiment, across all four stimulus conditions. There was a high degree of intra-subject concordance. All observers used the whole range of scores from 0 to 100, and the responses are evenly distributed over that range. The number to the right of each distribution gives the value of $r^2$ for the correlation of that observer's scores with the mean scores for all observers. When each test set is examined individually, $r^2$ is systematically higher.

### 4.2. Perceptual Scaling Results

Figure 8 shows the mean rating scores for the four test stimuli, the smooth sphere viewed from the front (Sphere-Front), the smooth sphere viewed from the side (Sphere-Side), the crinkly sphere viewed from the front (Crinkle-Front) and the crinkly sphere viewed from the side (Crinkle-Side). For each stimulus the observer compared the full geometry original with 12 alternate representations varying in texture and geometry. The four texture levels are shown along the x axis, with the no-texture condition at the extreme left. The three geometry conditions are shown along the y axis.

*Sphere-Front.* The plot in the top left quadrant of Fig. 8 contains results for the smooth regular sphere, lit front on. Data for the control stimulus (full geometry and full texture) is shown by the top right column in the set. Since this is identical to the comparison stimulus, it should be judged as equal (100). For this "sphere-front" stimulus, there was a clear effect of geometry. Increasing the geometry by a factor of 2 (from .047 to .093) caused the quality ranking to roughly double. An additional factor of 47 increase in the geometric resource, however, did not produce a further increase in perceived fidelity. Said the other way

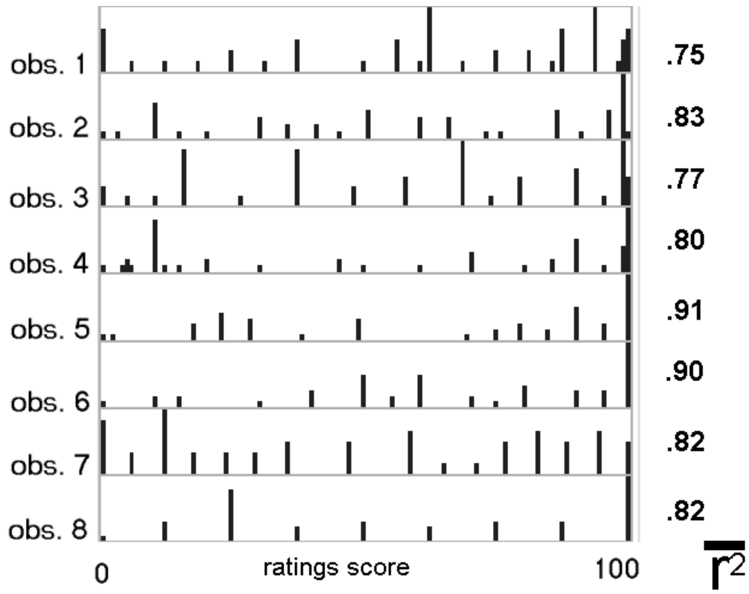


**Figure 7.** The distribution of scores assigned by each observer across the 4 test sets. The correlation coefficient $r^2$ shows the degree of correlation between each observer's score and the mean score for all observers. Each observer used the full range of scores. The individual scores correlated well with the mean scores.

around, this smooth shape, viewed front on, is impervious to geometric distortion. Reducing the geometric resource by a factor of 47 had no significant effect on the ratings. It was only perceived to be of reduced quality when the geometry was decreased by an additional factor of 2.

On this smooth sphere, variations in the texture resource had no effect on perceived image quality. The left-most set of columns shows the rating data for the three levels of geometric simplification with no texture added. For all three levels of geometric simplification, there was no change in the rating score with increases in the quality of the texture. For this low spatial-frequency texture, illuminated from the front, the rating score was driven entirely by the underlying geometry and was independent of texture.

*Sphere-Side.* When the sphere was viewed under oblique illumination (bottom left panel), the mean rating score dropped systematically for each decrease in geometry. This result occurred in the no-texture condition, and at all three levels of .

Under this illumination, adding texture had a significant effect on the perceived quality. Low resolution texture (.0122 Mb) degraded the perceived quality for all levels of geometric simplification. This may be due to the fact that under these lighting conditions, the pixel structure of the undersampled texture is very visible, producing an image which is distinctly different from the smoothly-shaded comparison stimulus. For the most simplified geometry (.047 Mb), additional texture did little to improve the perceived quality of the smooth sphere. For the less simplified models, using a less simplified texture significantly improved perceived quality, but with diminishing returns. The first factor of 16 (from .0122 to .197 Mb), produced a big improvement, but the next factor of four produced no additional effect. This may be because the factor of 16 jump in texture resolution was sufficient to eliminate most of the pixelation noise.

Can the degradation in perceived quality produced by the decrease in geometry be compensated for by adding texture? For the middle level (.093 Mb) geometry, the perceived quality of the geometry-plus-texture stimuli was always less than the perceived quality of geometry alone. Trying to compensate for reduced quality by adding texture would be a waste of resource.

*Crinkle-Front and Crinkle-Side.* The second model is the sphere with a highly textured surface, viewed under direct illumination (top right quadrant) and under oblique illumination (bottom right quadrant). Since the data for the two cases are quite similar, we will discuss them together. Looking first at the left-most columns, we see that reducing the geometry significantly reduced the perceived fidelity. For most simplifed geometry (.24 Mb), every increase in texture resource, even if that texture was highly subsampled, produced an increase in perceived fidelity. This can be seen clearly by looking at the bottom row of Fig. 6. The figure at the bottom left is the low-resolution geometric object without texture. The three images to its right

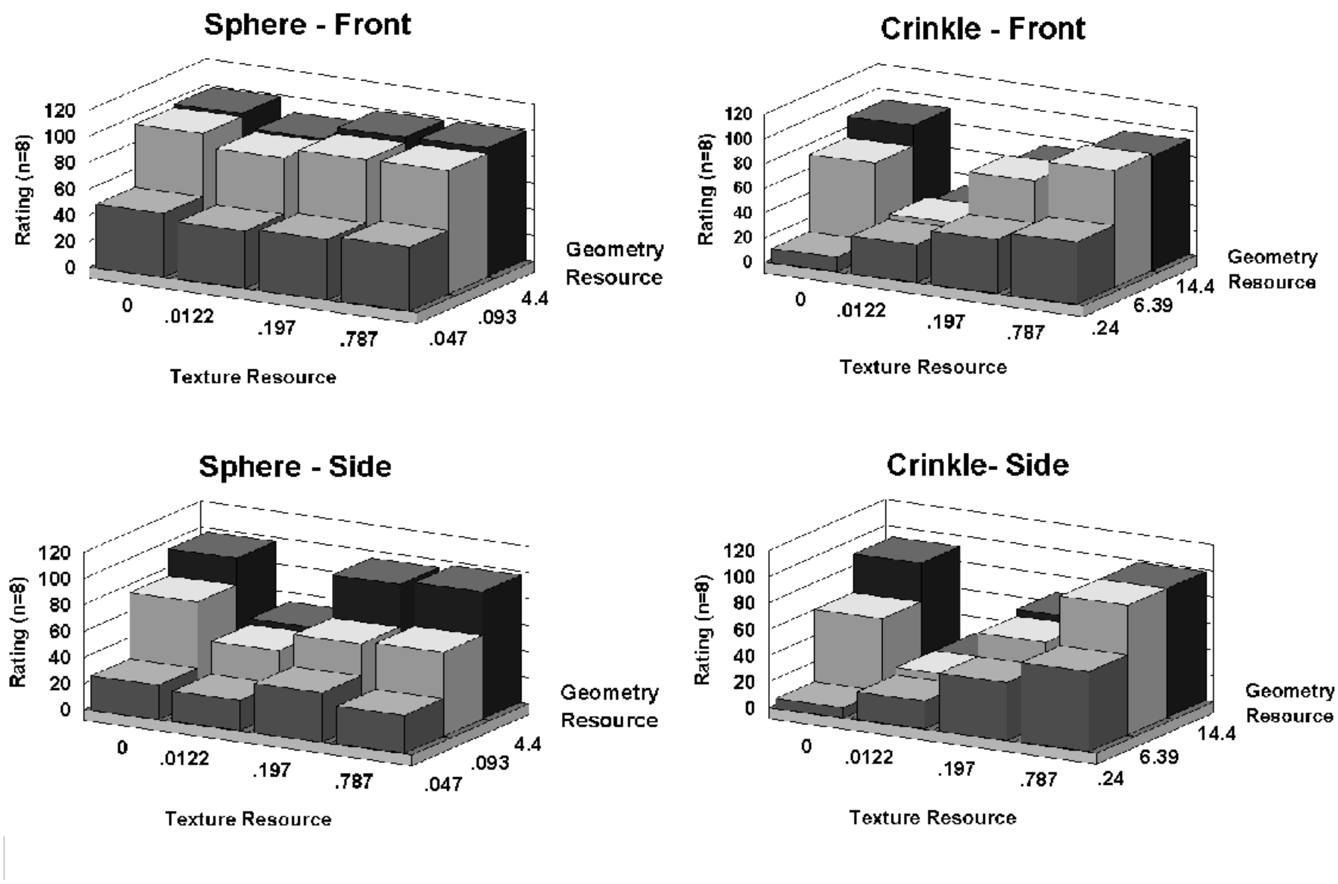


**Figure 8.** The mean scores for each of the four test sets are shown as bar charts. The bar in the back left of each chart corresponds to the full geometry. A comparison of the front- and side-lit sphere demonstrates that illumination can have a significant impact on perceived quality. A comparison of the sphere and crinkled sphere demonstrates that the effect of adding texture is different for these two objects. In particular, a comparison of the front row of bars (simplest geometry) shows that increasing texture does not improve perceived quality for the sphere, but does improve perceived quality for the crinkled sphere.

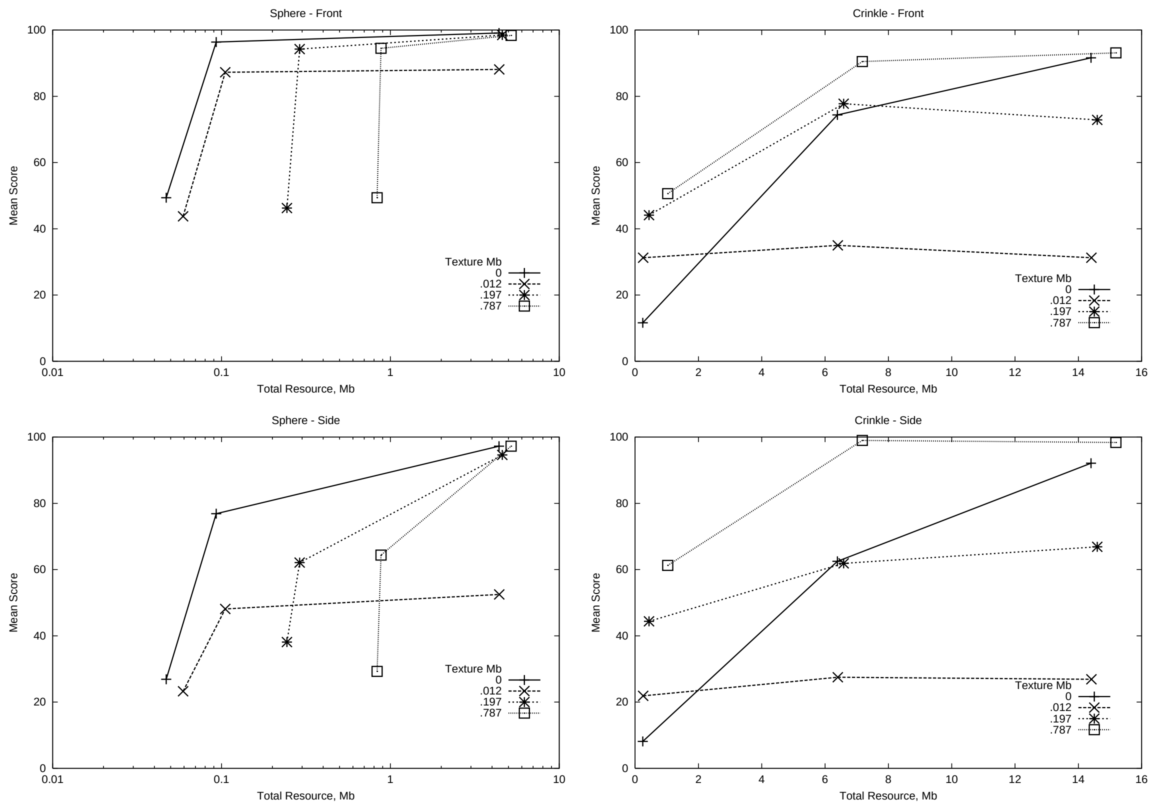


**Figure 9.** The mean scores for each of the four test sets are shown as line charts with mean score versus total memory resource. The 0 Mb texture case corresponds to geometry alone. For the sphere, the geometry alone always gives the best perceived quality at each resource level. For the crinkled sphere the full texture (.787 Mb) always gives the best perceived quality at each resource level.

show the effect of adding three resolution-levels of texture, respectively. Apparently, approximating the high spatial-frequency surface with an imperfect texture is better than no texture at all, for highly simplified geometries.

For the other geometries, which were much less simplified, adding a low-resolution texture (.0122 Mb) significantly decreased the perceived quality of the object. This may be because the simplified texture had a discriminably different (lower) spatial frequency than the perceived texture of the geometrically simplifed object. Adding additional texture resource increased the level of perceived quality.

### 4.3. Resource Trade-offs

Another question to ask of these data is what combination of texture and geometry gives the best perceptual effects? To answer this, we need to compare the perceptual ratings with the resources consumed in producing the stimuli. For example, in the above case, we see that for the crinkle spheres, the same perceived quality is obtained for low and high geometry, at .197 Mb texture resource. That is, a factor of 2 (7 Mb) is wasted by rendering the object with full geometry.

Figure 9 explores this issue more fully. Here the data for each of the four conditions are plotted in the same configuration as the plots in Fig. 8. Each quadrant plots the mean rating score across subjects as a function of the total resource, where total resource is the sum of the geometric and texture resource values, in megabytes. For the sphere front and sphere side stimuli, where there was a factor of 200 difference in resource, the values are plotted on a logarithmic scale. One approach to interpreting these data is to compare the perceived quality of the model when it is created with geometry alone (no texture), to the cases where texture has been added. The no texture case is shown as a full line. The horizontally shifted curves show the additional resource contributed when the three levels of texture are added. If the added texture enhances perceived quality, the geometry-plus-texture curve will lie above the dotted ”no-texture” curve. If adding the texture decreases perceived quality, then the geometry-plus-texture curve will lie below the no texture curve. The texture resources are indicated in the legend.

For the sphere-front stimulus (top left quadrant), the highest quality result is always obtained by geometry alone – for any resource level the dotted line is always at the top of the plot. That is, no expenditure of resource by adding texture improves the perceived quality of these stimuli. Furthermore, the poor quality texture (.0122 Mb) actually prevents any improved perceived quality when additional geometry is added, as indicated by the quality plot reaching a plateau. For the sphere-side stimulus (lower left quadrant) these general trends are repeated, with the gap between the quality of textured and untextured representations widening at each resource level.

For both the crinkle-front and crinkle-side stimuli (on the right) significant benefits are observed using the texture maps. Here the highest quality result is always obtained by the high-resolution texture mapped representation (.787 Mb). For the low-resolution geometry, applying any texture map results in an improved perceived quality, and the quality improves monotonically with the texture resolution. The resource required for the small geometry and full texture map is significantly less than the resource required for the medium geometry, without substantial loss in quality, particularly for the side lit case. The lines cross for the medium geometry case, reflecting the fact that only the high resolution texture is an improvement in this case. As in the sphere case, the flat line for the low resolution texture indicates that this level of texture mapping again actually prevents improvement by adding geometry.

## 5. DISCUSSION

In these experiments, we varied the texture and geometry of two models, a sphere with a low spatial frequency surface and a sphere with a crinkly, high spatial frequency surface. Each representation was compared with an original, "perfect" representation, providing a measure of the quality of the graphical representation.

4.1 Perceptual Discussion

In this section, we interpret the results in perceptual terms, exploring how our manipulations of geometry, texture, and lighting affected perceived fidelity for these two models. We consider two perceptual dimensions of these objects, their boundary contours and their texture. In this discussion, we discuss how our experimental manipulations affected these dimensions in a qualitative manner, but plan to make explicit measurements in the future.

*Boundary Contour.* Detecting small changes in an object's silhouette can be a very precise task for human observers, who can reliably discern variations on the order of seconds of arc, for example, seeing a broadcast spire against a sunset horizon. There has been a long tradition in the psychological literature regarding the role of boundary contours in the determination of object shape for objects with a countable number of contours. For complex silhouettes, the fractal dimension of the boundary contour has been shown to be important for object recognition (Rogowitz and Voss [15]). For simple shapes, Cortese and Dyre[16] have shown that shape discrimination depends on the frequency, amplitude and phase of the Fourier boundary contour. Reducing the geometry of an object makes the boundary contour more jagged, reflecting the fact that the object has been created with fewer polygons. In Fourier contour terms, the smooth sphere with full geometry has spatial frequency contours with zero amplitude. As the geometry is reduced, higher spatial-frequency contours are introduced, with increasing amplitude. The crinkly sphere has a high spatial-frequency contour. As the geometry is reduced, these high-spatial frequency components are replaced with successively lower spatial-frequency contours, with increasing amplitude.

The spatial resolution of the texture mapped onto the geometry can also affect an object's boundary contour. For the high-geometry crinkly sphere, the subsampled texture reduced the spatial-frequency of the boundary contour. For the low-geometry crinkly sphere, high-resolution texture increased the spatial frequency of the boundary contour.

*Texture Discrimination.* Another dimension along which these stimuli can be compared is the texture on the body of the object. Texture discrimination depends both on the frequency composition of the texture and the amplitude modulation of its components. For the smooth sphere, geometric simplification produces low spatial-frequency facets and contours. For the crinkly sphere, with a high spatial-frequency surface, geometric simplification reduces the spatial frequency of the object's surface, and can smooth out the crinkly surface altogether. Oblique illumination increases the contrast modulation of these crinkles and facets, which are made especially distinct as the object moves from full illumination to shadow, increasing texture discrimination.

In these experiments, several levels of texture were added to the geometric objects. In the case of the smooth sphere, this was a very low spatial-frequency texture, emulating the low-spatial frequency effect of light on a smooth sphere. Decreasing the resolution of the low spatial frequency texture produced some banding , contouring and worming, visible mostly in the oblique lighting condition. In the case of the crinkly sphere, this was a very high spatial-frequency texture, representing the effect of light on a crinkly sphere. At the lowest level of simplification, it produced a distinct, regular, pixellated pattern. With each increase in texture resolution, the more the texture map emulated the spatial frequency of the crinkly surface. At the highest level of texture resource, the majority of the high spatial-frequency detail was represented.

*Smooth Sphere.* For the sphere-front stimulus, ratings of perceived quality seem to be based only on boundary contour. A reduction in quality was only observed when the geometric resource was reduced from .093 Mb to .047, reflecting a perceptible change in the boundary contour. At all levels of geometric simplification, quality judgements were independent of texture resource, possibly because the texture was not visible under direct lighting. For the sphere-side stimulus, simplifying the geometry increased the amplitude and the spatial frequency of the boundary contour; increasing the texture slightly decreased the amplitude of the contour modulation. If the perceptual judgments were based solely on characteristics of the boundary contour, we would expect perceived quality to decrease for greater degrees of geometric simplification, which it does, and to increase slightly with added texture, which it does not. Adding a very simplified texture dramatically decreased perceived quality, presumably because the texture was discriminably different from the original. Adding additional texture resource improved the perceived quality, presumably reflecting a decrease in sampling artifacts.

*Crinkly Sphere.* If boundary contour were the major factor in determining the perceived quality of the crinkly sphere, perceived quality would be highest for the highest-geometry shape and decrease monotonically with reductions in geometry, correlated with the introduction of high amplitude, low spatial freqeuency jagged contours. This is certainly the effect obtained for geometry alone; under both illumination conditions, perceived quality increases with each increase in geometric resource. When texture is introduced, however, the medium-geometry object appeared to have the same quality as the high-geometry object. This suggests that the texture may mask the imperfections in the medium geometry's boundary contour.

Texture, however, can also decrease perceived quality. The most simplified texture, for example, brings perceived quality to its lowest levels, independent of the underlying geometry. This may be because this pixelated texture has a much lower spatial-frequency than the original, and is easily discriminated. Also, for all geometries, and under both illuminations, perceived quality increased monotonically with increased texture resolution, suggesting that successive approximations to the high spatial-frequency of the "original" stimulus may be driving these results.

*Illumination.* For the smooth sphere, the object was judged to be of systematically higher quality when viewed under direct illumination than when viewed under oblique illumination. For the crinkly sphere, representations based on the most simplified geometry and the most simplified texture appeared to have lower quality under oblique illumination, where the oblique lighting emphasized the jagged underlying geometry or the pixelated texture.

In interpreting these data, however, it is important to keep in mind that the goal of the experiment was not to study images of objects, but to understand the perception of objects. Thus, the front-view and the oblique-view are both descriptions of the same object, just viewed under two illuminations. In future experiments, we plan to ask observers to make judgments while dynamically varying the illumination. We suspect that when the observer is forced to view these stimuli as one object, that the judgments will be dominated by the lower perceived quality of the oblique view.

4.2 Graphics Discussion

When does texture successfully substitute for geometry? For the smooth sphere, texture was not able to compensate for the decrease in perceived quality produce by reducing geometry. For the crinkly sphere, however, the opportunity to use low-resource texture to substitute for geometry was clear. For highly simplified high spatial-frequency objects, viewed under either illumination, adding even small amounts of texture increase perceived quality substantially, and perceived quality increases monotonically with texture resource. Under both illumination conditions, adding less than 1 Mb of texture to the 6.39 Mb object improved the mean rating score $25\%$ giving it the same fidelity as the full-geometry object with 8 Mb more geometry. Clearly, texture can successfully trade for geometry when the geometry is complex and has been simplified by a factor of two or more. Since this is the arena of interest, these results are quite promising.

## 6. CONCLUSIONS

The framework and results presented here provide many useful insights into geometric simplification with texture substitution. The results of our experiment indicate that scaling experiments can produce consistent data regarding the perceived quality of object representations. The different results for front and side lighting for the smooth sphere indicate that lighting effects need to be accounted for in comparing objects, making this a fundamentally different problem from comparing images. The differing results for the smooth and crinkled sphere demonstrate that the benefit of expending resources substituting geometry with texture is object dependent. The results for varying texture resolution show that textures may be counterproductive if they are not of sufficiently high resolution. Practically speaking, if a system were forced to reduce texture map resolution because of a resource bottleneck, it may be preferable to not use texture at all.

These insights provide guidance for the development and testing of new simplification algorithms. Rather than using the same strategy for all objects, or all parts of a single object, it may be useful to analyze the nature of the object's geometry to test if it can be replaced by a texture map. We believe that our framework for measuring the perceptual consequences of different geometries, textures, and lighting conditions can be used to evaluate the success of such tests.

## REFERENCES


1. E. Horvitz and J. Lengyel, "Perception, attention, and resources: A decision theoretic approach to graphics rendering," in *Proceedings of the Thirteenth Conference on Uncertainty in Artificial Intelligence*, 1997.
2. B. Watson, A. Friedman, and A. McGaffey, "Using naming time to evaluate quality predictors for model simplification," in *Proceedings of SIG CHI 2000*, p. to appear.
3. P. Cignoni, C. Montani, and R. Scopigno, "A comparison of mesh simplification algorithms," *Computers and Graphics* , 2000.
4. W. J. Schroeder, J. A. Zarge, and W. E. Lorensen, "Decimation of triangle meshes," in *Proceedings of SIGGRAPH 92*, E. Catmull, ed., pp. 65–70, ACM, 1992.
5. A. Gueziec, "Locally toleranced surface simplification," *IEEE Transactions on Visualization and Computer Graphics* **5**, pp. 168–169, 1999.
6. J. Rossignac and P. Borrel, "Multi-resolution 3D approximations for rendering complex scenes," in *Geometric Modeling in Computer Graphics*, B. Falcidieno and T. Kunii, eds., pp. 455–465, Springer-Verlag, 1993.
7. R. Klein and A. Schilling, "Efficient rendering of multiresolution meshes with guaranteed image quality," *The Visual Computer* **15**, pp. 443–451, 1999.
8. P. Heckbert, "Survey of texture mapping," *IEEE Computer Graphics and Applications* **6**(11), pp. 56–67, 1986.
9. F. Weinhaus and V. Devarajan, "Texture mapping 3D models of real-world scenes," *ACM Computing Surveys* **29**(4), pp. 325–365, 1997.
10. J. Shade, D. Lischinski, T. DeRose, J. Snyder, and D. Salesin, "Hierarchical image caching for accelerated walkthroughs of complex environments," in *Proceedings of SIGGRAPH 96*, H. Rushmeier, ed., pp. 75–82, ACM, 1996.
11. P. Maciel and P. Shirley, "Visual navigation of large environments using textured clusters," in *SIGGRAPH Symposium on Interactive Graphics*, pp. 95–102, 1995.
12. J. Torborg and J. Kajiya, "Talisman: Commodity realtime 3D graphics for the PC," in *Proceedings of SIGGRAPH 96*, H. Rushmeier, ed., pp. 353–364, 1996.
13. M. Soucy, G. Goudin, and M. Rioux, "A texture-mapping approach for the compression of colored 3D triangulations," *The Visual Computer* **12**, pp. 503–514, 1996.
14. J. Cohen, M. Olano, and D. Manocha, "Appearance-preserving simplification," in *Proceedings of SIGGRAPH 98*, M. Cohen, ed., pp. 155–122, ACM, 1998.
15. B. Rogowitz and R. Voss, "Shape perception and low-dimension fractal boundary contours," in *Proc. SPIE Human Vision and Electronic Imaging: Models, Methods, and Applications*, vol. 1249, pp. 387–394.
16. J. Cortese and B. Dyre, "Perceptual similarity of shapes generated from Fourier descriptors," *Journal of Experimental Psychology: Human Perception and Performance* **22**, pp. 133–143.