A Perceptually Based Adaptive Sampling Algorithm

Mark R. Bolin* **Gary W. Meyer†**

Department of Computer and Information Science
University of Oregon
Eugene, OR 97403

Abstract

A perceptually based approach for selecting image samples has been developed. An existing image processing vision model has been extended to handle color and has been simplified to run efficiently. The resulting new image quality model was inserted into an image synthesis program by first modifying the rendering algorithm so that it computed a wavelet representation. In addition to allowing image quality to be determined as the image was generated, the wavelet representation made it possible to use statistical information about the spatial frequency distribution of natural images to estimate values where samples were yet to be taken. Tests on the image synthesis algorithm showed that it correctly handled achromatic and chromatic spatial detail and that it was able predict and compensate for masking effects. The program was also shown to produce images of equivalent visual quality while using different rendering techniques.

CR Categories and Subject Descriptors: I.3.3 [Computer Graphics]: *Picture/Image Generation;* I.3.7 [Computer Graphics]: *Three-Dimensional Graphics and Realism;* I.4.0 [Image Processing and Computer Vision]: *General.*

Additional Key Words and Phrases: Adaptive Sampling, Perception, Masking, Vision Models.

1 Introduction

The synthesis of realistic images would be greatly facilitated by employing an algorithm that makes image quality judgements while the picture is being created instead of relying upon the user of the software to make these evaluations once the image is complete. In this way it would be possible to find the artifacts in a picture as it was being rendered and to invest additional effort on those areas. By targeting those parts of the picture where problems are visible, the overall time necessary to compute the picture could be reduced. It would also be possible to have the algorithm stop when the picture quality had reached a predetermined level. This would permit the use of radically different rendering algorithms but still have them produce an equivalent visual result.

Image quality evaluation programs have been developed by vision scientists and image processors to determine the differences between two pictures. Given a pair of input images, this software returns a visibility map of the variations between the two image arrays. While these programs are capable of making the visual judgements required by a perceptually based image synthesis algorithm, they are currently too expensive to execute every time a decision is necessary about where to cast a ray into an image or when the overall visual quality of the picture is acceptable. Their efficient evaluation also requires a frequency or a wavelet representation for the images instead of the usual pixel based scheme.

The objective of this paper is to integrate an existing image quality evaluation algorithm into a realistic image synthesis program. This is to be done in such a way that image quality judgements can be made as the image is produced without severely impacting the overall execution time of the rendering program. This will require that the image quality metric be made to run more efficiently without sacrificing its ability to detect visible artifacts. It will also necessitate that the coefficients of a frequency or a wavelet representation are computed by the image synthesis algorithm instead of the individual pixels of the final image. This will have the side benefit of allowing the algorithm to make use of statistical information about the frequency content of natural images when actual data from the scene being rendered is not available.

Including this introduction, the paper is divided into seven major sections. In the second section, previous work on vision based rendering algorithms is reviewed and existing image processing based vision models are described. A simplified version of a vision model is developed in the third section and is integrated into a rendering algorithm in the fourth section. In the fifth section the statistics of natural images are used to make guesses about unknown values as the image is computed. Finally, the results of the algorithm are discussed in the sixth section and some conclusions are drawn in the seventh section.

2 Background

A few attempts have been made to develop image synthesis algorithms that, as the picture is created, detect threshold visual differences and direct the algorithm to work on those parts of the image that are in most need of refinement. There have also been image processing algorithms invented, both inside and outside the field of computer graphics, that can be used to determine the visibility of differences between two images. In this section we review work in each of these areas in preparation for describing

*e-mail: mbolin@cs.uoregon.edu
†e-mail: gary@cs.uoregon.edu

how we have combined an image processing and image synthesis algorithm to create a new image rendering technique.

2.1 Vision Based Rendering Algorithms

Mitchell [21] was the first to develop a ray tracing algorithm that considered the perception of noise and attempted to operate near its threshold. He adopted a Poisson disk sampling pattern to concentrate aliasing noise in high frequencies where the artifact is less conspicuous. He also employed an adaptive sampling technique to vary the sampling rate according to frequency content. A contrast calculation was performed in order to obtain a perceptually based measure of the variation in the signal. Differential weighting was applied to the red, green, and blue contrasts to account for color variation in the eye's spatial sensitivity.

Meyer and Liu [20] developed an image synthesis algorithm that took full advantage of the visual system's limited color spatial acuity. To accomplish this they used an opponents color space with chromatic and achromatic color channels. They employed the Painter and Sloan [23] adaptive subdivision algorithm to compute a k-D tree representation for an image. Because lower levels of the tree contained the higher spatial frequency content of the picture, they descended the k-D tree to a lesser depth in order to determine the chromatic channels of the final image.

Bolin and Meyer [2] were the first to use a simple vision model to make decisions about where to cast rays into a scene and how to spawn rays that intersect objects in the environment. The model that they employed consisted of three stages: receptors with logarithmic response to light, opponents processing into achromatic and chromatic channels, and spatial frequency filtering that is stronger for the color channels. They computed a spatial frequency representation from the samples that they took. As higher image frequencies were determined the number of rays spawned was decreased. This allowed them to exploit the phenomena of masking in their algorithm.

Gibson and Hubbold [10] have used a tone reproduction operator to determine display colors during the rendering process. This made it possible to compute color differences in a perceptually uniform color space and control the production of radiosity solutions. They used this technique on the adaptive element refinement, shadow detection, and mesh optimization portions of the radiosity algorithm.

2.2 Image Processing Based Models of the Visual System

The architectures of image processing based models of the visual system share a number of common elements. The first stage of the models is usually a nonlinear intensity transformation. This is done to account for the visual system's difference in detection capability for information in light and dark areas of a scene. The second stage typically involves some spatial frequency processing. Most contemporary models break the spatial frequency spectrum into separate channels. The sensitivity of the individual channels is controlled so that the overall bandpass corresponds to the contrast sensitivity function. The spatial frequency hierarchy makes it possible to determine whether a signal will be masked or facilitated by the presence or absence of background information with a similar frequency composition. Finally the outputs of the separate frequency channels that are above threshold are summed to create a final representation.

Two important examples of image processing based models of the visual system are the Daly Visual Difference Predictor (VDP) [5] and the Sarnoff Visual Discrimination Model (VDM) [17]. The Daly VDP takes a more psychophysically based approach to vision modeling. As such it uses a power law representation for the initial nonlinearity and it transforms the image into the frequency domain in order to perform its filtering operations. The Sarnoff VDM focuses more attention on modeling the physiology of the visual pathway. It therefore operates in the spatial domain and does a careful simulation of such things as the optical point spread function.

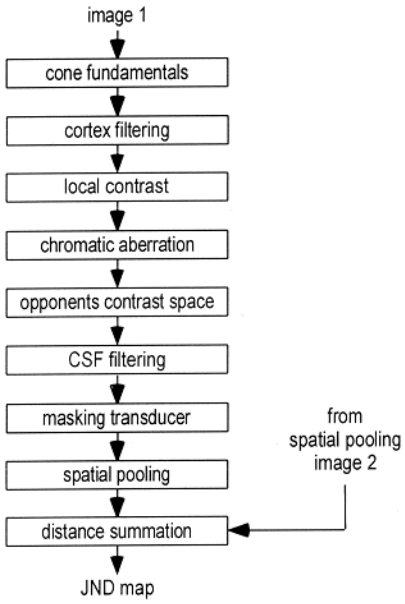

Figure 1: Block diagram of vision model.

In recent work, the Daly VDP and the Sarnoff VDM have been applied to precomputed computer graphic imagery. Rushmeier, *et. al* [25] used the initial stages of the Daly VDP (and other vision metrics) to compare a simulated and a measured image. Ferwerda, *et. al.* [7] extended the Daly VDP to include color and modified how it handles masking. The result was a new image processing based model of the visual system that they used to demonstrate how surface texture can mask polygonal tessellation. Li [15,16] has used computer graphic pictures to compare the Daly VDP and the Sarnoff VDM. She found that the two models performed comparably, but that the Sarnoff VDM gave better image difference maps and required less recalibration. The Sarnoff VDM was also determined to have better execution speed than the Daly VDP but required the use of significantly more memory. As a result of this comparison we have decided to use the Sarnoff VDM as the basis for our new vision based rendering algorithm.

3 Simplified Vision Model

The vision model that we have developed bears many similarities to the Sarnoff VDM discussed in the previous section. In creating a new model of visual perception we were motivated by two primary factors. The first and foremost criteria is the speed of the visual model. Modern visual difference predictors have gone to great lengths to accurately model the perceptual sensitivity of the human visual system. However, efficiency is seldom a design criteria in developing these systems. This fact limits the utility of these algorithms in applications where speed is a primary concern. The second factor that motivated our development of a new model is the correct handling of color. The majority of visual difference predictors have been designed only for gray scale images, and the ones that include color have neglected the significant effect of chromatic aberration.

The perceptual model that will be described has been imbed-

ded into a visual difference predictor. This difference predictor receives as input two images specified in CIE XYZ color space. It returns as output a map of the perceptual difference between the two images specified in terms of just noticeable differences (JND's). One JND corresponds to a 75% probability that an observer viewing the two images would be able to detect a difference, and the units correspond to a roughly linear magnitude of subjective visual differences [17].

A block diagram of our visual difference predictor is given in Figure 1. The steps *cone fundamentals* through *spatial pooling* are carried out independently on both input images. The differences between the two images are accumulated in the *distance summation* step.

In the first stage of the vision model entitled *cone fundamentals*, the pixels of the input image are encoded into the responses of the short (S), medium (M) and long (L) receptors found in the retina of the eye. This is accomplished using the transformation from CIE XYZ to SML space specified by Bolin and Meyer [2].

There is now abundant evidence for the existence of channels in the visual pathway that are tuned to a number of specific frequencies and orientations [17]. The visual processing that occurs on a channel is relatively independent of all other channels. In the Sarnoff VDM this *cortex filtering* stage is accomplished by transforming the image into a Laplacian pyramid and applying a set of oriented filters. The net result is a pyramidal image decomposition that is tuned to seven spatial frequencies and four angular directions. This transform is the primary source of expense in the Sarnoff VDM. In order to reduce the cost of this operation we decided to model the spatial frequency and orientation selectivity of the visual system through the use of a simple Haar wavelet transform. A number of other wavelet bases were considered, including Daubechies' family of wavelets [6] and the biorthogonal bases of Cohen, *et. al.* [4]. However, these transforms were discarded due to their expense. The two-dimensional non-standard Haar decomposition can be expressed as:

$$
\begin{aligned}
c_{l-1}[\frac{x}{2},\frac{y}{2}] &= (c_l[x,y]+c_l[x+1,y]+ \\
&\quad c_l[x,y+1]+c_l[x+1,y+1])/4 \\
d^1_{l-1}[\frac{x}{2},\frac{y}{2}] &= (c_l[x,y]-c_l[x+1,y]+ \\
&\quad c_l[x,y+1]-c_l[x+1,y+1])/4 \\
d^2_{l-1}[\frac{x}{2},\frac{y}{2}] &= (c_l[x,y]+c_l[x+1,y]- \\
&\quad c_l[x,y+1]-c_l[x+1,y+1])/4 \\
d^3_{l-1}[\frac{x}{2},\frac{y}{2}] &= (c_l[x,y]-c_l[x+1,y]- \\
&\quad c_l[x,y+1]+c_l[x+1,y+1])/4 \qquad (1)
\end{aligned}
$$

where c_l specifies the lowpass coefficients of the level l Haar basis, d^1_l, d^2_l and d^3_l are the detail coefficients of the three two-dimensional level l Haar wavelets, and $c_{levels-1}[x,y]$ corresponds to the response of either the small, medium or long receptors at a pixel location (where $levels$ represents the number of levels in the quad-tree). This decomposition is carried out for each of the S, M and L channels and is stored in a quad-tree representation with the highest frequency details at the bottom and lowest frequency at the top. The detail coefficients of the Haar transform constitute our cortex transform. These detail terms represent variations in the image that are localized in space, frequency and angular direction. The frequency selectivity of the detail terms at a given level of the tree is defined as the frequency in cycles per degree (cpd) to which the wavelet at that level is optimally tuned. The detail coefficients are tuned to three angular directions as illustrated in Figure 2. We acknowledge that the poor filtering and limited orientation tuning of the Haar wavelet is a limitation of this approach. However, the efficiency gains are substantial.

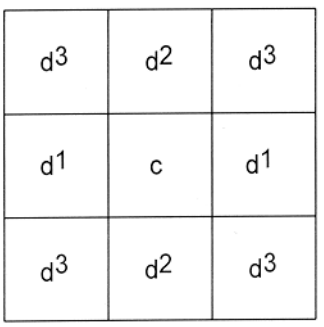

Figure 2: Angular tuning of Haar coefficients.

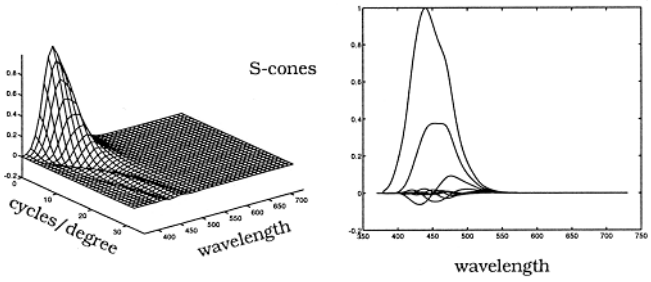

Figure 3: Effect of chromatic aberration on the S-cone photopigment sensitivity. Right diagram takes cross sections through left diagram at intervals of 4 cpd (from Marimont and Wandell [19]).

In the next stage labeled *local contrast* the eye's non-linear contrast response to light is modeled. This is accomplished by dividing the detail coefficients of each color channel by the associated lowpass coefficient one level up in the quad-tree. This operation produces a local contrast value which is functionally equivalent to the standard cone contrast calculation of $\frac{\Delta S}{S}$, $\frac{\Delta M}{M}$, and $\frac{\Delta L}{L}$. It additionally avoids the assumption, found in other models [5,7], that the eye can adapt at the resolution of a pixel.

The next step in the visual model incorporates the effect of *chromatic aberration*. Chromatic aberration describes the defocusing of light as a function of wavelength by the optics of the eye. The original chromatic contrast sensitivity experiments performed by Mullen [22] corrected for chromatic aberration. In order to accurately apply the results of her work it is necessary to reintroduce this effect. Chromatic aberration most strongly affects the sensitivity of the short wavelength receptors. The loss of sensitivity in the short wavelength receptors is demonstrated in Figure 3. This illustration shows that the sensitivity drops to less than half its original value at 4 cpd and is virtually non-existent at frequencies higher than 8 cpd. Chromatic aberration is simulated in our model by lowpass filtering the local contrasts of the S cone receptors as a function of spatial frequency. The lowpass filter used was generated by a fit to the data of Marimont and Wandell [19].

The following stage in the model consists of a transformation of the cone contrasts to an *opponents contrast space*. This space consists of a single achromatic (A) and two opponent color channels (C_1 and C_2). There is significant evidence that the signals produced by the cones undergo this type of transformation. The transformation matrix used to convert the cone contrasts is found in [2].

The sixth step of the vision model, labeled *CSF filtering*, in-

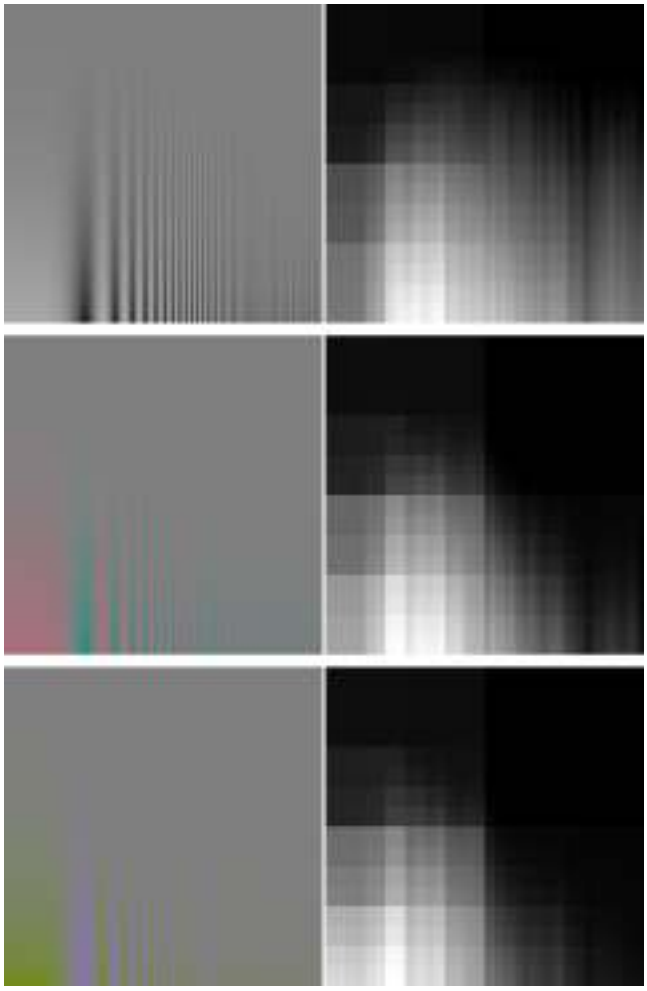

Figure 4: Achromatic and chromatic contrast sensitivity functions and comparison against a uniform gray field.

corporates variations in achromatic and chromatic contrast sensitivity as a function of spatial frequency. The gray scale contrast sensitivity illusion in the top left of Figure 4 demonstrates the sensitivity variation of the achromatic channel. In this demonstration contrast increases logarithmically from top to the bottom of the image and frequency increases logarithmically from left to right. The subjective contour in the shape of an inverted "U" that can be seen along the top of the image is generated by the points at which the contrast of the sinusoidal grating becomes just noticeably different from the gray background. This image demonstrates that achromatic sensitivity reaches its peak at around 4 cpd and drops off significantly at higher and lower spatial frequencies. The equation for the achromatic contrast sensitivity function that is used in our model is presented by Barten [1].

The middle and bottom images on the left side of Figure 4 contain contrast sensitivity illusions for the C_1 and C_2 color channels respectively. In these illustrations it should be observed that the peak sensitivity to chromatic contrast is less than that for achromatic contrast, and that the cutoff for the chromatic sensitivity function occurs at a much lower spatial frequency than in the achromatic illustration. The reader should also see that the shape of the subjective contour is strictly lowpass, with no drop-off at low spatial frequencies. The fact that the cutoff for the C_2 color channel is less than that for the C_1 is the result of axial chromatic aberration which was modeled at an earlier stage of the algorithm. In our algorithm the chromatic contrast sensitivity function is modeled with a Butterworth filter that has been fit to the chromatic contrast sensitivity data from Mullen [22].

At this stage in the algorithm the square of the contrast for each of the A, C_1 and C_2 channels is multiplied by the square of that channel's contrast sensitivity as a function of spatial frequency. The square of the contrast and contrast sensitivity function is used to model the energy response that occurs for complex cells, as described in the Sarnoff VDM. This transformation has the result of making the model less sensitive to the exact position of an edge, which is a property shared by the human visual system as well [17]. The illustrations on the right side of Figure 4 show the output of our visual difference predictor when comparing the contrast sensitivity illusions on the left side of this figure with a constant gray image. White indicates areas of large visual difference while black denotes regions of low visual difference. In these images we see that the algorithm is able to correctly predict the shape and cutoff of the subjective contour.

The next stage of the model labeled *masking transducer*, incorporates the effect of visual masking. Masking describes the phenomena where strong signals of a given color, frequency and orientation can reduce the visibility of similar signals. This property of the visual system is incorporated through the use of a non-linear, sigmoid transducer described in the Sarnoff VDM

$$T(A) = \frac{2A^{2.25/2}}{A^{2.05/2} + 1}, \tag{2}$$

where T(A) is the transducer output and A is the weighted contrast output from the previous stage of the model. This transducer is applied independently to the contrasts of each of the A, C_1, and C_2 color channels.

In computer graphic renderings, error primarily is manifested in the form of noise. Therefore, it is worthwhile to give special attention to the issue of noise masking. Noise in the achromatic channel is often the result of aliasing due to undersampling or can result from poor Monte Carlo light source integration. An illustration of the grayscale contrast sensitivity illusion perturbed by the introduction of random noise is given in the upper left of Figure 5. In this image the noise is readily apparent above and to the sides of the subjective contrast sensitivity contour, but is less perceptible in areas where the sinusoidal grating is visible. This result occurs because the strong visual sensitivity to these frequencies masks the presence of a portion of the frequency spectrum of the noise. The image in the upper right of this figure shows the output of our visual difference predictor when comparing the original contrast sensitivity illustration to the contrast sensitivity illustration with noise added. In this image we see that the visual model has correctly predicted that the error is less visible in the lower-center region where masking is strongest.

Noise in the chromatic channels can arise when Monte Carlo integration is performed with multiple colored lights or is used to compute diffuse inter-reflections. Fine grained noise is not masked significantly in the color channels due to the lower frequency cutoff for the chromatic contrast sensitivity function. However, masking can still have a strong affect on the visibility of coarse grained noise. In the middle left and bottom left images in Figure 5 we have overlaid the chromatic contrast sensitivity illusions with coarse grained noise. In these illustrations the noise is very apparent in regions where sensitivity to the chromatic grating is low (top and right of the images), but less visible in regions where the chromatic grating is very perceptible (lower left of the images). The images on the right once again show the output of the visual difference predictor when comparing the images with noise to the original chromatic contrast sensitivity illustrations. In these illustrations we see that the algorithm has correctly predicted that the coarse grained noise is less perceptible in the lower left region of

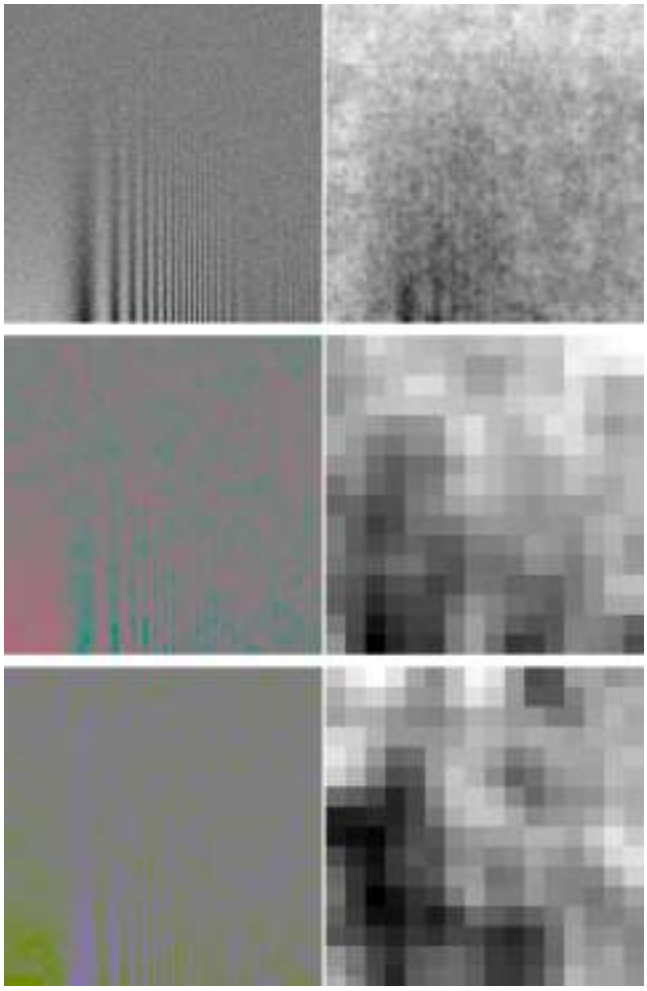

Figure 5: Achromatic and chromatic contrast sensitivity functions with noise, and comparison with noiseless contrast sensitivity functions.

the images.

In the next stage of the model labeled *spatial pooling*, the transducer outputs are filtered over a small neighborhood of surrounding nodes at each level of the quad-tree. This is similar to the pooling operation performed in the Sarnoff VDM. It captures the fact that foveal human sensitivity is at a maximum for sine wave gratings containing at least 5 cycles. The pooling filter that is used in our model is:

$$\begin{bmatrix} \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \\ \frac{1}{8} & \frac{1}{4} & \frac{1}{8} \\ \frac{1}{16} & \frac{1}{8} & \frac{1}{16} \end{bmatrix}. \quad (3)$$

The decision to use a 3x3 filter rather than the 5x5 filter specified in the Sarnoff VDM was made to improve the speed of the algorithm.

In the final *distance summation* stage the differences between the pooling stages of the two input images are computed and used to generate a visual difference map. The local visual difference at each node of the quad-tree is defined to be the sum across all orientations (θ) and color channels (c) of the differences of the pooling stages (P_1 and P_2) of the two images raised to the 2.4 power:

$$LD = \sum_{\theta=1}^{3} \sum_{c=1}^{3} (P_1[\theta, c] - P_2[\theta, c])^{2.4} \quad (4)$$

The final difference map is generated by accumulating visual differences across levels. This is accomplished by summing local difference down each path in the quad-tree and storing the result in the leaves. The visual difference map that is the output of the algorithm is given by the leaf differences raised to the 1/2.4 power.

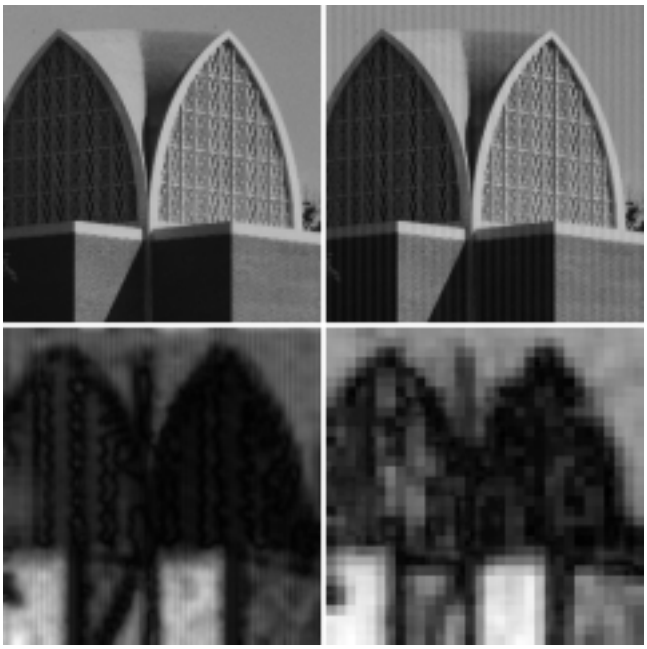

Figure 6: Top - Original chapel (left) and chapel with sinusoidal distortion (right). Bottom - Results of the Sarnoff VDM (left) and simplified vision model (right) visual difference predictions.

Figure 6 shows a comparison between the results of the original Sarnoff VDM and our simplified version for a set of complex images. The inputs are illustrated in the top row of the figure and consist of a chapel image and the chapel image perturbed by a sinusoidal grating. A visual comparison of these two images shows that the sinusoidal distortion is most evident in the dark regions at the base of the chapel. This is due to the eye's non-linear contrast response to light. Within the arches at the top of the chapel, there is no perceptible difference between the two images. This is because the lattice-work in these regions masks the presence of the sinusoidal grating. The visual difference map that is produced by the new algorithm contains a number of blocking artifacts that are caused by the Haar wavelet decomposition. However, the results of both algorithms are similar and correspond well with a subjective comparison of the input images. The Sarnoff VDM processed one channel in a gray-scale image representation and the new model processed three color channels. The new model executed in $1/60^{th}$ of the time of the original model.

4 Adaptive Sampling Algorithm

An adaptive sampling algorithm has been developed that is based on the visual model described in the preceding section. This algorithm receives sample values as input, and specifies the placement of samples at the image plane as output. The goal of

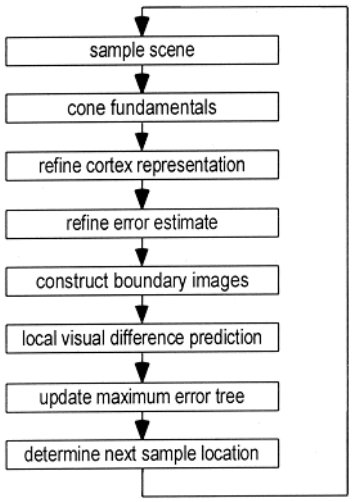

Figure 7: Block diagram of adaptive sampling algorithm.

the adaptive algorithm is to iteratively place each sample at the location that currently contains the largest perceptual error.

The key to developing this perceptually based adaptive sampling algorithm comes from two primary insights. The first is that an estimate of the image and its error can be used to construct two boundary images that may be used as input into a visual difference predictor. The output of this difference prediction can then be used to direct the placement of subsequent samples. The second insight is that a given sample only affects the value and accuracy of a very limited number of terms at each level of a Haar wavelet image representation. This fact makes the algorithm tractable because it implies that only a small number of operations are necessary to refine the image approximation, its error estimate and the visual difference prediction for any given sample.

The algorithm proceeds through a few basic steps. First, as samples are taken of the scene, a Haar wavelet image approximation is generated and refined. Next, a multi-resolution error estimate is developed and similarly refined. This error estimate is expressed in terms of the variance of the detail terms in the Haar representation. The image approximation and error estimate is then used to construct two boundary images which serve as input to the visual difference predictor. The output of the difference predictor is accumulated in a hierarchical tree. The nodes of this tree specify the maximum visual difference present at the current nodes and the children below it. This tree is traversed choosing the branch with the largest visual difference in order to determine the location on the image plane with the greatest perceptual error.

A block diagram of the algorithm is illustrated in Figure 7. As samples are taken their values are first transformed from CIE XYZ to SML space in the step labeled *cone fundamentals*. The Haar image representation and its error estimate are constructed in this space.

In the *refine cortex representation* stage the Haar image approximation is created and refined. This is done through a technique similar to the "splat and pull" method used by Gortler, *et. al.* [11]. The Haar image representation is stored in a quad-tree data structure. The leaves of this structure are defined to contain the intensity of single pixels in the image plane and the interior nodes contain the lower resolution lowpass and detail terms of the Haar representation. As a sample is passed into this stage it is "splatted" at the leaf containing the sample. The intensity at this leaf is simply the average of the samples taken within the pixel it is defined to cover. The lower resolution lowpass and detail terms are generated by "pulling" the updated leaf intensity up through the tree. During this process, if all children of a node contain at least a single sample, then the lowpass and detail terms are given by Equation 1. If only a single child contains a sample, then the detail terms are left undefined and the lowpass term is set equal to the lowpass of the child containing the sample. If only two or three children contain samples, then a simple scheme is used to fit the lowpass and one or two detail terms, respectively, to the values of the defined children. In this manner the image representation is gradually resolved as samples are taken of the scene. It is also worth noting that this process is very fast since the addition of a sample only requires the updating of a single path up the tree.

At the next step labeled *refine error estimate*, the error of the current Haar approximation is determined. This process is similar in some respects with the algorithm described by Painter and Sloan [23]. The error estimate is expressed in terms of the variance of the lowpass and detail coefficients. For leaf nodes containing at least two samples the variance of the pixel approximation is given by the variance of the samples in the leaf divided by the number of samples in the leaf [3,13]. The error of the lowpass and detail terms in the interior nodes is defined with respect to the error of their children. If the variance is defined for all children of a node, then the variance of the lowpass and detail terms at the node is equal to the sum of the variance of the four children divided by 16. This result comes from the rule

$$V[\sum^{i} a_i x_i] = \sum^{i} a_i^2 V[x_i] \qquad (5)$$

(where V denotes variance) and inspection of Equation 1. If the error is not defined for all children of a node and at least 2 samples have been taken, then the variance is given by the variance of the samples taken within the node divided by the number of samples in the node. As in the case of refining the Haar representation, updating the multi-resolution error estimate requires that only a single path in the tree be modified for the addition of each sample.

The next stage in the algorithm labeled *construct boundary images* is concerned with defining the two input images for the visual difference predictor. These input images are described by the magnitude of their detail coefficients which are used to determine the local contrast at an early stage in the vision model. Since the image approximation and error estimate has only changed along a single path in the tree, the detail terms for the boundary images only need to be updated along this path as well. The details for the two boundary images are derived from the details in the current image approximation and the variance of those details. The magnitude of the approximated detail specifies a mean value and the square root of the variance defines the spread of the standard deviation curve. The magnitudes of the details for the boundary images are taken from the 25% and 75% points on this curve. In this manner two boundary images are specified which should contain the true value 50% of the time. The boundary images are organized so that image 1 contains the detail of minimum energy contrast and image 2 contains the detail of maximum energy contrast.

A *local visual difference prediction* is performed at the updated nodes in the next step of the algorithm. The detail terms in the boundary images are passed through the *local contrast* to *spatial pooling* stages of the vision model. The transducer outputs at the current node is stored in the tree for fast re-use in the pooling stages of neighbor nodes.

In the step labeled *update maximum error tree* a value is stored at each updated node in the quad-tree which represents the maximum visual difference contained at the current node and the nodes below it. The local error at a node is defined as in the *distance summation* stage of the vision model (i.e. by the sum of visual distance between the boundary images across each detail and color

channel raised to the 2.4 power). The maximum error is defined to be the local error plus the largest maximum error contained in either of the four children. The maximum error of the root node is raised to the 1/2.4 power and represents the largest visual error contained at any location within the image plane. The maximum error in the interior nodes are used to determine which branch of the tree contains the greatest visual difference for the purpose of finding the next location to sample.

In the final stage labeled *determine next sample location* a sample location is selected at the point of maximum visual difference. The location is selected by traversing the quad-tree in a top-down fashion and, at each node, selecting the branch of maximum visual error. This traversal continues until a leaf node is encountered or an interior node is found which contains less than eight samples. If a leaf node is reached, a sample is randomly placed within it. If the traversal stops at an interior node, then a sample location is chosen randomly from a child's quadrant so that the number of samples in each child node is balanced to a tolerance of one sample.

The discussion thus far has assumed that only a single path in the quad-tree is affected by a given sample. However, this is not strictly the case. Due to the local contrast and spatial pooling stages of the vision model the modification of one node in the quad-tree can have an affect on the visual difference at neighboring nodes. One solution to this problem is to update multiple paths up the tree. However, this approach was deemed too expensive. Instead the problem can be effectively solved by adding a small amount of randomness to the traversal of the maximum error tree. As each node in the tree is visited, there is some likelihood a neighboring node will be chosen instead. In this manner, if a particular path is traversed often, there is a chance of selecting neighboring paths. This creates the opportunity to incorporate updated values into the local contrast and spatial pooling calculations for these paths.

The algorithm continues recursively until the maximum error of the root node drops below a specified tolerance. The output image is reconstructed by simply doing an inverse Haar transform of the image representation and converting pixel values from SML to the frame buffer space. This technique can also be used to construct an iterative display of the image during the progression of the algorithm.

5 Selecting Values for Unknown Quantities

A difficulty with adaptive sampling algorithms that are based on the sample variance is knowing when and to what extent to believe the error estimate obtained from the samples. This is especially true for the hierarchical variance estimation scheme described in this paper. If the first two samples obtained from the scene return exactly the same values and therefore have zero variance, can we conclude that the image has been computed exactly and stop? What if the image has been sampled densely and two samples from within a particular pixel of the image plane are the same, can we say that the intensity of the pixel has been computed correctly? A person analyzing these two situations would certainly believe that the scene has not been adequately sampled in the first case, but would probably be willing to stop sampling in the second case. The reason for this difference stems from the statistics of natural images.

A number of authors have analyzed the statistics of images commonly encountered in nature [8,9,24,26]. These authors have found that the frequency spectra of natural images is not random, but tends to be highly correlated and contains a 1/f drop-off in the magnitude of the frequency terms. Therefore, if only two samples have been taken of a scene, we have just begun to compute the low end of the frequency spectra. Based on our experience with images found in the natural world, we know that an average image contains higher frequency detail, and therefore believe that the scene has not been adequately sampled. Thus, we have some apriori knowledge about the error of an image approximation. If a portion of the frequency spectra has not been computed, then, on average, the approximation of the image will contain an amount of error that is equivalent to the 1/f magnitude of the uncomputed spectra.

We can also draw upon the statistics of natural scenes when we must choose unknown values for the chromatic channels. The frequency content of naturally occurring spectral reflectances is known to be very low [18]. This means that reflectances are more likely to be uniform across the spectrum than they are to be spikey. The result of this is that the average color in the natural world is quite desaturated. This implies that in the absence of other knowledge about the chromatic content of an object, setting the chromatic channels close to zero is as good a choice as one can make.

The statistics of natural images discussed in this section have been employed within our adaptive sampling algorithm. This is accomplished by initializing the two boundary images to a uniform gray for one, and a statistically average image for the other. The visual difference predictor is run on these two input images and the output is used to seed the visual difference at each node in the quad-tree. Initially, the estimated visual difference of the rendering is based on the comparison of the gray and statistically average image. As the algorithm progresses and the image approximation and error estimate is calculated at new nodes in the tree, the visual difference based on the average statistics is traded for the visual difference that is based on the variance and content of the scene samples.

6 Results

In this section we discuss the results of applying our image synthesis algorithm to three dimensional environments. Simple texture mapped disks are considered first followed by a scene with more complicated geometry and lighting. Two shading techniques will be used in these examples, direct and Monte Carlo light source sampling. The direct sampling method uses a simple shading algorithm in which point light sources are directly sampled each time a ray strikes a surface. The Monte Carlo method uses area light sources and blind Monte Carlo integration to evaluate the shading integral. In this approach the incident radiance at a surface point is evaluated by spawning a number of rays at random orientations across the positive hemisphere. We realize that blind integration is not the most efficient means of evaluating the shading integral. However, this technique provides a simple means of demonstrating a situation where noise is present within the illumination calculation.

Figure 8 shows three arrays of texture mapped disks in which the spatial frequency of the texture increases from left to right but the contrast of the texture decreases from bottom to top. In the top disk array the color of the texture varies along the A axis of AC_1C_2 space, in the middle disk array along the C_1 axis, and in the bottom array along the C_2 axis. The three arrays of texture mapped disks are rendered using direct light source sampling. In this case there is no noise generated and the spatial frequency content of the textures is the primary determinant of the sampling rate that is used. All of the disk arrays were rendered to the same visual tolerance. As can be seen in the figure, the sampling density decreases from high frequency to low frequency. Achromatic colors receive far more samples than chromatic colors due to the higher spatial frequency cut off of the achromatic contrast sensitivity function, while colors that vary in C_1 are sampled more often than colors that change in C_2. This difference in sampling between the two color channels is clear evidence of the filtering

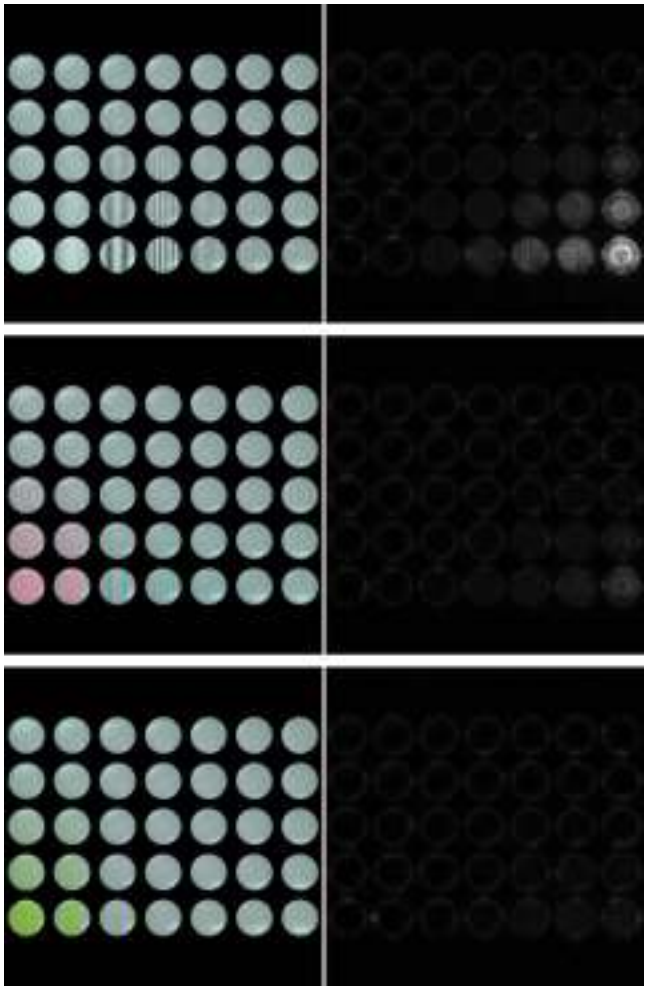

Figure 8: Sampling densities for direct light source sampling.

that is done by the visual model due to chromatic aberration in the eye.

In Figure 9 the achromatic disks from Figure 8 are rendered again using Monte Carlo light source sampling. In this case a significant amount of noise is generated and the effect of visual masking becomes important. As can be seen from the figure, the spatial sampling pattern is radically different from the direct light source sampling case in Figure 8. While disks with high frequency textures still receive the most samples, in this case the low frequency disks also get many samples because the noise can be seen on their surfaces. On the other hand, the middle of the spatial frequency range receives relatively few samples because the noise is less visible due to masking. In Figure 10 the environment and the lighting is made more complex but a similar result is obtained. When there is no noise, high achromatic spatial frequency transitions receive the most samples. When noise is present, more samples across the entire image are required, but fewer are necessary for frequencies where the noise is masked.

As a final example, identical scenes were synthesized to the same visual tolerance using two different rendering techniques. As can be seen in Figure 11, the images that resulted are comparable even though the sampling patterns and illumination calculations are very different. In the case where direct sampling of the light sources is performed, aliasing artifacts are the most prevalent defect; while for the scene where Monte Carlo sampling of the light sources was done, noise is the dominant problem. However, for a given perceptual tolerance, the algorithm holds each type of artifact to a similar level of visual impact.

The approach taken in this algorithm is to compute the perceptual metric for every ray that is cast into the scene. The cost to do this computation is 1 ms on a 100 MHz processor. Evaluation of the algorithm on a number of different test environments shows that it takes fewer samples than either a uniform sampling method or an adaptive approach with an objective error metric (90% less in certain cases). Timing tests reveal that the algorithm is able to provide the perceptual stopping criteria demonstrated in Figure 11 while remaining competitive with either the uniform sampling or standard adaptive sampling techniques. The method is faster than either uniform or standard adaptive sampling on every environment where it was tested, but it was not the overall winner in all cases. Additional work is necessary to exploit the algorithm's excellent spatial sampling rates and determine the optimal number of samples to be taken between evaluations of the perceptual metric.

7 Conclusion

An existing vision model has been incorporated into an image synthesis algorithm to control where samples are taken as a picture is created. The results obtained with the new algorithm on three dimensional scenes track the results obtained using the visual model by itself on two dimensional images. The impact on the execution time of the rendering program has been minimized while the amount of memory required has been increased. The contributions of this work can be summarized as follows:

1. A new image quality model has been developed. This new model is an efficient implementation of an existing algorithm. It executes in a fraction of the time of the original method without a significant sacrifice in accuracy. The model has also been extended for color including the effect of chromatic aberration in the optics of the eye.

2. An image synthesis system has been created that directly computes a wavelet representation for an image. This is a functional (instead of an explicit pixel based) scheme for describing a picture that facilitates the computation of a visual metric. It also permits the use of statistical information regarding the frequency distribution of natural images to estimate values in regions where samples have yet to be taken. In the same manner, guesses regarding unsampled colors were improved by using an opponents color space to store color.

3. A perceptually based approach to image synthesis has been produced. An image quality model was used to decide where to take the next sample in a picture. This can result in a savings in execution time because samples are only

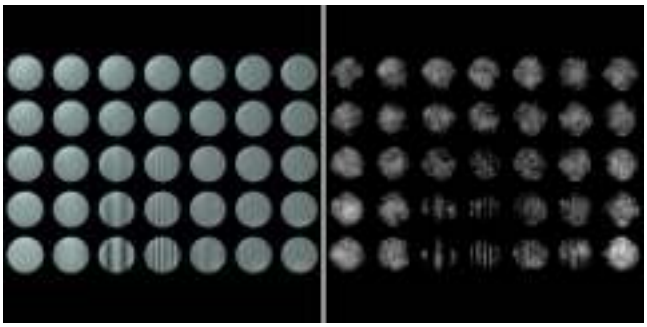

Figure 9: Sampling density for Monte Carlo light source sampling.

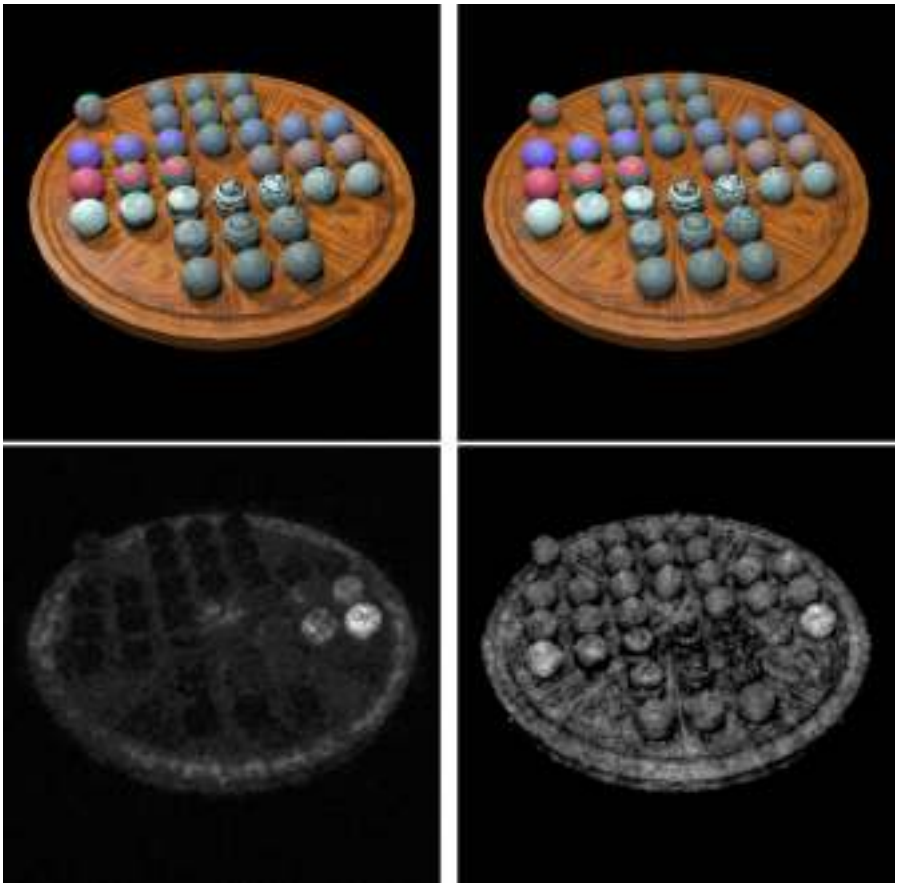

Figure 10: Sampling densities for direct (left) and Monte Carlo (right) light source sampling. Color varies in the middle three rows along the C_2, C_1, and A axes of AC_1C_2 space. Contrast of the middle three balls in the C_2 and A rows is decreased in the top two and bottom two rows respectively.

taken in areas where there are visible artifacts. The image quality model is also used to decide when enough samples have been taken across the entire image. This provides a visual stopping condition and makes it possible to employ different rendering algorithms but still produce equivalent pictures.

This work represents a first attempt to imbed a sophisticated image processing vision model into an image synthesis algorithm. While the results are encouraging it is clear that the approach taken here puts a certain amount of overhead onto every ray that is cast into the scene. An alternative tactic might be to initially sample the image at a low rate and compute the visual difference map from these values. The visual difference map can then be used to select regions of the image which require further sampling. The use of the imbedded version of the vision model might be saved until the image is more fully developed and the masking effects have become completely apparent.

8 Acknowledgements

The authors would like to thank Jae H. Kim for his help in creating Figures 8, 9, 10, and 11 and for his assistance in assembling all of the color figures in this paper. This research was funded by the National Science Foundation under grant number CCR 96-19967.

9 References

[1] Barten, P. G. J., "The Square Root Integral (SQRI): A New Metric to Describe the Effect of Various Display Parameters on Perceived Image Quality," *Human Vision, Visual Processing, and Digital Display*, Proc. SPIE, Vol. 1077, pp. 73-82, 1989.

[2] Bolin, M. R. and Meyer G. W., "A Frequency Based Ray Tracer," *Computer Graphics, Annual Conference Series*, ACM SIGGRAPH, pp. 409-418, 1995.

[3] Bolin, M. R. and Meyer G. W., "An Error Metric for Monte Carlo Ray Tracing," *Rendering Techniques '97*, J. Dorsey and P. Slusallek, Editors, Springer-Verlag, New York, pp. 57-68, 1997.

[4] Cohen, A., Daubechies, I., and Feauveau, J. C., "Biorthogonal Bases of Compactly Supported Wavelets," *Communications on Pure and Applied Mathematics*, Vol. 45, No. 5, pp. 485-500, 1992.

[5] Daly, S., "The Visible Differences Predictor: An Algorithm for the Assessment of Image Fidelity," *Digital Images and Human Vision*, A. B. Watson, Editor, MIT Press, Cambridge, MA, pp. 179-206, 1993.

[6] Daubechies, I., "Orthonormal Bases of Compactly Supported Wavelets," *Communications on Pure and Applied Mathematics*, Vol. 41, No. 7, pp. 909-996, 1988.

[7] Ferwerda, J. A., Shirley, P., Pattanaik, S. N., and Greenberg, D. P., "A Model of Visual Masking for Computer Graphics," *Computer Graphics, Annual Conference Series*, ACM SIGGRAPH, pp. 143-152, 1997.

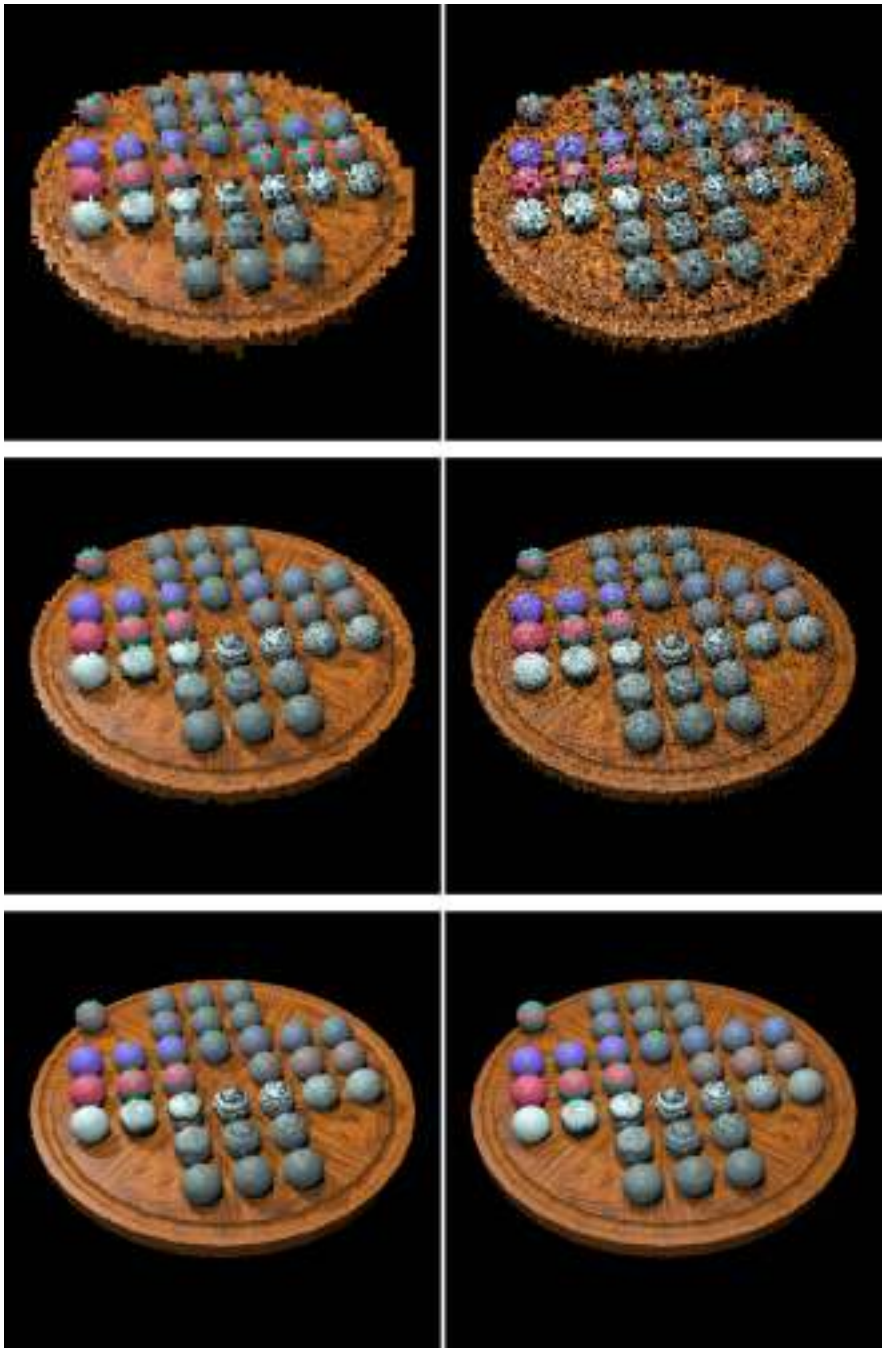

Figure 11: Image rendered at a visual tolerance of 7 (top), 5 (middle), and 3 (bottom) using direct light source sampling (left column) and Monte Carlo light source sampling (right column).

[8] Field, D. J., "Relations Between the Statistics of Natural Images and the Response Properties of Cortical Cells," *J. Opt. Soc. Am. A*, Vol. 4, pp. 2379-2394, 1987.

[9] Field, D. J., "What the Statistics of Natural Images Tell Us About Visual Coding," *Human Vision, Visual Processing, and Digital Display*, Proc. SPIE, Vol. 1077, pp. 269-276, 1989.

[10] Gibson, S. and Hubbold, R. J., "Perceptually-Driven Radiosity," *Computer Graphics Forum*, Vol. 16, pp. 129-140, 1997.

[11] Gortler, S. J., Grzeszczuk, R., Szeliski, R., and Cohen, M. F., "The Lumigraph," *Computer Graphics, Annual Conference Series*, ACM SIGGRAPH, pp. 43-54, 1996.

[12] Kirk, D. and Arvo, J., "Unbiased Sampling Techniques for Image Synthesis," *Computer Graphics, Annual Conference Series*, ACM SIGGRAPH, pp. 153-156, 1991.

[13] Lee, M. E., Redner, R. A., and Uselton, S. P., "Statistically Optimized Sampling for Distributed Ray Tracing," *Computer Graphics, Annual Conference Series*, ACM SIGGRAPH, pp. 61-67, 1985.

[14] Legge, G. E. and Foley, J. M., "Contrast Masking in human vision," *Journal of the Optical Society of America*, Vol. 70, pp. 1458-1470, 1980.

[15] Li, B., "An Analysis and Comparison of Two Visual Discrimination Models," *Master's Thesis, University of Oregon*, June 1997.

[16] Li, B., Meyer, G. W., and Klassen, R. V., "A Comparison of Two Image Quality Models," to appear in *Human Vision and Electronic Imaging III*, B. E. Rogowitz and T. N. Pappas, Editors, Proc. SPIE, Vol. 3299, 1998.

[17] Lubin, J., "A Visual Discrimination Model for Imaging System Design and Evaluation," *Vision Models for Target Detection and Recognition*, Eli Peli, Editor, World Scientific, New Jersey, pp. 245-283, 1995.

[18] Maloney, L. T., "Evaluation of linear models of surface spectral reflectance with small numbers of parameters," *J. Opt. Soc. Am. A*, Vol. 3, pp. 1673-1683. 1986.

[19] Marimont, D. H. and Wandell, B. A., "Matching Color Images: The Impact of Axial Chromatic Aberration," *J. Opt. Soc. Am. A*, Vol. 12, pp. 3113-3122, 1993.

[20] Meyer, G. W. and Liu, A., "Color Spatial Acuity Control of a Screen Subdivision Image Synthesis Algorithm," *Human Vision, Visual Processing, and Digital Display III*, Bernice E. Rogowitz, Editor, Proc. SPIE, Vol. 1666, pp. 387-399, 1992.

[21] Mitchell, D. P., "Generating Antialiased Images at Low Sampling Densities," *Computer Graphics, Annual Conference Series*, ACM SIGGRAPH, pp. 65-72, 1987.

[22] Mullen, K. T., "The Contrast Sensitivity of Human Colour Vision to Red-Green and Blue-Yellow Chromatic Gratings," *J. Physiol. (Lond.)*, Vol. 359, pp. 381-400, 1985.

[23] Painter, J. and Sloan, K. "Antialiased Ray Tracing by Adaptive Progressive Refinement," *Computer Graphics, Annual Conference Series*, ACM SIGGRAPH, pp. 281-288, 1989.

[24] Ruderman, D. L., "Origins of Scaling in Natural Images," *Human Vision, Visual Processing, and Digital Display*, Proc. SPIE, Vol. 2657, pp. 120-131, 1996.

[25] Rushmeier, H., Ward, G., Piatko, C., Sanders, P., and Rust, B., "Comparing Real and Synthetic Images: Some Ideas About Metrics," *Rendering Techniques '95*, P. M. Hanrahan and W. Purgathofer, Editors Springer-Verlag, New York, pp. 82-91, 1995.

[26] Schreiber, W. F., *Fundamentals of Electronic Imaging Systems*, Springer-Verlag: Berlin Heidelberg, 1993.