EUROGRAPHICS 2001 / A. Chalmers and T.-M. Rhyne
(Guest Editors)

*Volume 20* (*2001*), *Number 3*

# Automatic Lighting Design using a Perceptual Quality Metric

Ram Shacked Dani Lischinski

School of Computer Science and Engineering, The Hebrew University of Jerusalem, Jerusalem, Israel

**Abstract**

*Lighting has a crucial impact on the appearance of 3D objects and on the ability of an image to communicate information about a 3D scene to a human observer. This paper presents a new automatic lighting design approach for comprehensible rendering of 3D objects. Given a geometric model of a 3D object or scene, the material properties of the surfaces in the model, and the desired viewing parameters, our approach automatically determines the values of various lighting parameters by optimizing a perception-based image quality objective function. This objective function is designed to quantify the extent to which an image of a 3D scene succeeds in communicating scene information, such as the 3D shapes of the objects, fine geometric details, and the spatial relationships between the objects.*

*Our results demonstrate that the proposed approach is an effective lighting design tool, suitable for users without expertise or knowledge in visual perception or in lighting design.*

## 1. Introduction

Lighting design for image synthesis involves specifying values for lighting parameters, such as position, color, and intensity, for each of the light sources in a 3D scene model. Once the scene geometry, the material properties, and the viewing parameters have been specified, the appearance of the scene in a rendered image depends exclusively on the lighting. Poorly designed lighting may result in incomprehensible images, containing under- and over-illuminated regions, exhibiting poor contrast, and failing to effectively communicate the scene to a human observer.

The traditional approach towards lighting design for image synthesis typically uses a *direct design* paradigm, where the user iteratively specifies all of the required lighting parameters, renders the scene, evaluates the results, makes modifications in the design, and so forth. This is essentially a trial-and-error approach, with the obvious drawback that the user must actively participate in each iteration. Thus, the design process is time-consuming and tedious, and it requires expertise in lighting design as well as an understanding of visual perception issues.

An alternative approach is based on an *inverse design* paradigm. The user is presented with some interface that enables him to specify a set of objectives and/or constraints that the lighting design should satisfy, and the parameters are then solved for in an automatic fashion[6, 11, 18, 23]. These methods, reviewed in Section 2, are certainly less tedious, but still requires users to know and to be able to articulate *a priori* what is the appearance that they desire to achieve. Thus, this approach might still be difficult to use for a non-expert user, whose goal is merely to render a comprehensible image of the scene at hand.

This paper presents a novel fully automatic approach to lighting design, geared towards generation of comprehensible, communicative images of 3D objects. More specifically, given a geometric model of a 3D object or scene, the material properties of the surfaces in the model, and the desired viewing parameters, our approach automatically determines the values of various lighting parameters. This is done by optimizing a *perception-based image quality objective function* designed to quantify the extent to which an image of a 3D scene succeeds in communicating scene information, such as the 3D shape of each object, fine geometric details, and the spatial relationships between the objects in the scene.

Our image quality function, described in Sections 4 and 5, measures in an image several perceptually significant qualities that are directly affected by changes in the illumination conditions. It then computes the differences between these measured values and a set of automatically determined target values that should be measured under "ideal" lighting conditions. A weighted sum of these differences constitutes our assessment of the quality of a particular lighting design. The resulting quality value can be used to compare different lighting designs (for the same scene and viewpoint).

In Section 6 we utilize our image quality function as an objective function for lighting design: we present a system that searches the space of lighting designs spanned by several free lighting parameters for a locally optimal lighting design (corresponding to a local minimum in the objective function). In conjunction with some heuristics for automatic setting of the initial lighting specifications, our system provides a *fully automatic* tool for lighting design.

Our method is mostly suitable for ordinary users, with no expertise in lighting design or visual perception, who simply wish to synthesize a comprehensible image of their scene model. It is also suitable for incorporation into various modeling tools for CAD and animation. The experiments reported in Section 7 demonstrate that our approach is able to quickly and automatically generate lighting designs that are significantly superior to commonly used default lighting configurations.

## 2. Related work

Most previous automatic lighting design systems use the inverse design paradigm. From the user's point of view such systems are primarily characterized by the set of design goals the user is free to specify, and the design space that they search. Most systems search a very limited portion of the lighting parameters space, and still require considerable knowledge and expertise from the user.

Schoeneman *et al.*[23] control colors and intensities of light sources by "painting" desired colors onto the scene's surfaces. Kawai *et al.*[11] control light emissions and directions, as well as surface reflectances by requesting the user to specify various constraints and objectives for the illumination. Both of these techniques work for mostly diffuse scenes, and do not change the positions of lights. The design goals are achieved using optimization. Poulin and Fournier[17] and Poulin *et al.*[18] let the user specify shadows and highlights as design goals, from which they infer the light source position and surface roughness.

Costa *et al.*[6] present a methodology in which fictitious luminaires can be defined and placed in the scene to describe desired radiance distribution. Free design variables are then chosen (e.g. light location and direction), and optimization is used to determine their values. This is a powerful approach, capable of handling a wide range of design variables and constraints; however, specifying the design goals and the constraints in this system appears to be a difficult task even for expert lighting designers. Furthermore, the objective function for the optimization process must be also specified by the user by programming it using a supplied scripting language.

An entirely different approach for exploring the space of lighting designs is presented in the Design Galleries framework of Marks *et al.*[14]. Given a set of lighting parameters, they try to optimally disperse the space of solution images in terms of perceptual quality, and allow the user to browse these possible results and linearly combine them to try and compose a desired solution. However, the metric that they use to measure the perceptual quality distance between two given images is a simple pixel intensity distance.

Several other relevant works belong to the area of non-photorealistic rendering. These works are concerned with generating visually comprehensible renderings of 3D objects. Having the privilege of using non-photorealistic enhancement techniques, these methods usually draw the edges of the objects in black, and enhance the appearance of surfaces by techniques such as adding cool-to-warm tone gradations[10] or drawing contour lines and curved hatching[22]. Our work has similar goals, but we limit ourselves to photorealistic computer graphics techniques, and enhance visual comprehensibility by manipulating only the lighting parameters.

## 3. Visual Perception

In order to design a perceptual quality metric for automatic lighting design, we must first define what visual information we would like our images to communicate, and then find practical computational ways to *quantify* the effectiveness with which this information is communicated in specific images.

Our approach is based on a fundamental assumption that the perceptual quality of computer-generated images is determined by several distinguishable aspects of visual information:

**Shape** Since computer-generated photorealistic images are normally concerned with displaying 3D scenes, the most fundamental requirement from such an image is that it should clearly convey the 3D shape of the visible surfaces and objects, as well as the spatial relationships between them.

**Details** A photorealistic image of a 3D scene should capture, as much as possible, the fine geometric details present in the model, and display them conspicuously.

**Surface properties** An image should communicate surface properties, such as color, reflectance, and roughness.

**Realism** Beyond communicating shape information a photorealistic image should convey a realistic impression. For our purposes, we are concerned with the *presence* of visual information that gives a sense of realism and not with the degree of accuracy of the image in terms of similarity to a real world scene[21].

The next step is to address the question of how these types of information are represented in the image, and in particular, how they might be affected by changes in lighting.

The process of recovering the 3D structure from an image of a scene is performed by the human visual system (HVS)[2, 15]; The input for this process is the spatial and spatiotemporal patterns of light arriving from the scene (or from an image of the scene) and falling on the retinas. The HVS analyzes these patterns of light to retrieve information about surfaces and objects in the environment being viewed. The questions of interest to us are: (a) *what* features in the retinal image convey the information leading to three dimensional perception; (b) *how* can they be detected; and (c) *how much* each feature contributes to visual perception.

Space limitations prevent us from presenting a survey of the vast amount of relevant research in the fields of visual perception and computer vision (but see Shacked[24]). Unfortunately, this research is able to provide us with only limited answers to question (a), even fewer answers to question (b), and hardly any useful answers to question (c). In other words, the current results in these fields are insufficient for our purpose of quantifying the perceptual quality of a given image. Nevertheless, some of the visual perception theories and psychophysical research did provide us with useful information about features that affect image perception, and in particular three-dimensional perception and the recovery of depth and local surface orientation. These features are:

- Edges: occluding (profile) edges, boundary (internal) edges[10, 12, 15, 20, 22], and reflectance edges[1].
- Shading gradients: luminance gradients created by the reflection of light on surfaces[3, 10, 16, 20].
- Shadows[3]: *attached* shadows (self shadows) and *cast* shadows. In this paper we take into account only attached shadows.
- Highlights: specular highlights can also help in perceiving the shape of objects[7, 25], in addition to providing information about the materials they are made of.
- Luminance information is much more important for 3D shape perception than color information[3, 8, 20].
- Brightness adaptation and contrast sensitivity: the HVS adapts itself to the current light intensity level. When viewing an image, that intensity is determined by the average image intensity. Contrast sensitivity is highest around the adaptation level, and decreases away from it[2, 4, 9].
- Lightness constancy: there is no unique illumination intensity for the reflectance of an object to be perceived correctly by the HVS[4, 5].
- Light source attributes: (i) light direction: the HVS tends to assume that the light illuminating the scene is coming from above[20]. (ii) spectral distribution: from the perceptual standpoint, there does not appear to be any benefit in chromatic illumination, while achromatic light preserves the colors of the materials[3, 8, 20]. (iii) number of lights: the HVS tends to assume that the scene is illuminated by a single light source[1, 20]. Furthermore, multiple lights can cause effects that may be confusing and contradictory. Therefore, increasing the number of light sources in the scene should generally only be done when necessary for resolving particular deficiencies in the illumination.

## 4. Perceptual Image Quality Function: Principles

In this section we construct a perceptual image quality function for lighting design. More specifically, guided by our knowledge about the human visual perception of 3D shape and spatial relationships, we define a mapping $f_Q$ that receives as input a 3D scene model $M$ and an image $I$ that was rendered from this model, and maps its input to a single non-negative scalar value. This value attempts to quantify the extent to which the image $I$ contains features and exhibits properties that make it easy for a human observer to comprehend the 3D shape and structure of the scene $M$. The smaller the value of $f_Q(I)$, the higher the estimated perceptual quality of the image. Naturally, $f_Q$ is designed in such a way that it is strongly dependent upon the lighting in the scene. Changes in the lighting design cause changes in $I$, which are in turn reflected in the value of $f_Q(I)$. Thus, the problem of finding an optimal lighting design for the scene is cast as an optimization problem — finding a local minimum of $f_Q$.

It should be noted that the images $I$ are luminance images, rendered from the model $M$ without applying any surface textures that might be present in the model. The reason is that we operate on pixel intensities when quantifying the perceptual quality of an image. Surface textures perturb these intensities, making it difficult to isolate the effects of changes in lighting on the perceptual image quality. Therefore, if If the scene does contain surface textures, the lighting design process is performed ignoring them, but they can be integrated back into the scene once the lighting has been determined.

The function $f_Q$ is defined as a linear combination of six *target terms*, each responsible for measuring a different feature or property in the image:

$$f_Q = f_{grad} + f_{edge} + f_{var} + f_{mean} + f_{hist} + f_{dir}, \qquad (1)$$

More specifically, these six terms have been designed to respond to:

1. *Local luminance patterns*: The term $f_{grad}$ measures the magnitudes of the shading gradients present in the image. The term $f_{edge}$ detects edges in the image and measures their prominence.
2. *Pixel luminance statistics*: $f_{var}$ measures the distance of the luminance variance from a target value. $f_{mean}$ measures the distance of the mean luminance from a target value. $f_{hist}$ measures the distance of the luminance histogram shape from an ideal equalized histogram.
3. *Illumination direction*: $f_{dir}$ measures the elevations of the light sources with respect to the viewing direction.

In the remainder of this section we describe each of the six target terms in more detail, and illustrate their impact on the lighting of a scene using Figure 1*. The order in which the functions are presented is similar to the order in which the terms were added into the quality function in the course of our research: each new term was added in order to overcome deficiencies unresolved by the previous terms. For clarity of presentation, the terms are described at the level of principles. A precise definition for each term will be given in Section 5.

### 4.1. The shading gradients term $f_{grad}$

This term measures the average shading gradient magnitude in the image. Only shading gradients in geometrically smooth regions of the scene are taken into account here, since gradients at the edges are accounted for by a different term ($f_{edge}$). The value computed by this term is the difference between the measured average gradient and a target value representing the maximum average gradient that could potentially be measured in an image of that particular scene with the given viewing parameters.

**Example:** Figure 1a was rendered using lighting parameters obtained by optimizing only the $f_{grad}$ term. It is obvious that this term alone is unable to produce satisfactory results: the resulting image is too bright, and much of the detail is washed-out by the strong illumination.

### 4.2. The detected edges term $f_{edge}$

Given a 3D scene model it is easy to establish which edges could be visible in an image of the scene, rendered from a given viewpoint[22, 13]. The extent to which these potential edges are in fact perceived by a human observer, depends mostly on the lighting. The $f_{edge}$ term measures this extent by applying an edge detection operator at pixels located on potential edges, and summing the responses.

**Example:** Figure 1b was rendered using lighting parameters determined by optimizing the function $f_Q = f_{grad} + f_{edge}$. As expected, several edges that were not visible in 1b become clearly visible now. A secondary result of adding $f_{edge}$ is that it reduces the overall excessive intensity caused by using only a gradient component. The result is still not satisfactory, however, since the image tends to contain extremely dark (under-illuminated) regions alongside with extremely bright (over-illuminated) ones. Important details are often lost in both types of these extreme regions. It is difficult to perceive fine detail in such regions, because of the global brightness adaptation of the visual system, which causes poor contrast sensitivity in dark and bright areas.

### 4.3. The variance term $f_{var}$

In order to overcome the problem of large extreme dark and bright regions, we introduce a term that inhibits extreme variations in intensity, by measuring the distance between the variance in the pixel luminances and a target variance value. This target variance must be chosen carefully, so as to reduce the extreme variations in intensity, but still allow a sufficient dynamic range in which shading gradations can take place.

It should be noted that our use of a variance reducing term is consistent with the low dynamic range principle used in (non-photorealistic) technical illustration[10].

**Example:** Figure 1c was rendered using lighting parameters determined by optimizing the function $f_Q = f_{grad} + f_{edge} + f_{var}$, using a target standard deviation value of 42 (on a 0 to 255 scale). The histogram of the image, shown in Figure 2b, has a standard deviation of 41.8 that is very close to the target value (instead of 77.6 in Figure 2a). Figure 1c shows a major improvement with respect to the high intensity variance found in 1b; However, since there was no constraint on the mean image intensity it still appears too bright.

### 4.4. The mean term $f_{mean}$

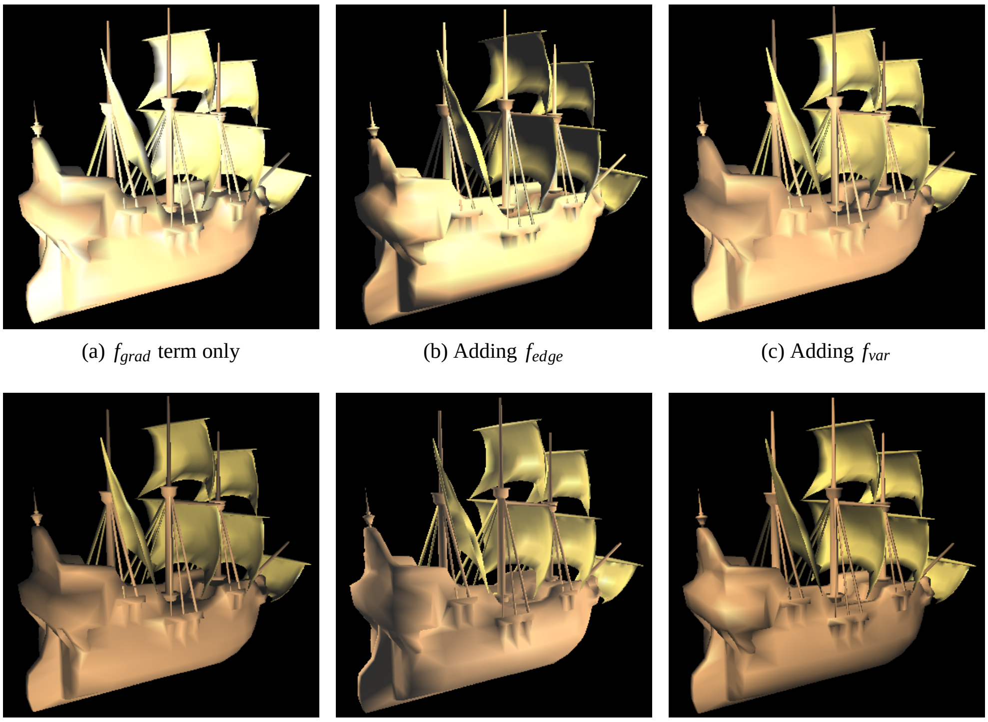
The overall brightness of an image is an important factor in its appearance. From perceptual quality point of view, it is undesirable for the image to appear too dark or too bright, since this tends to weaken the effect of the shading and may hide various features and detail in the scene. Even if no loss of detail occurs, there is still some subjective notion of an "appropriate" brightness for the image. For instance, the ship in Figure 1c might appear too bright to most observers. In order to control the overall brightness of the image, we add another target term $f_{mean}$ in order to pull the mean luminance in the image towards a desired target value.

**Example:** Figure 1d was rendered using lighting parameters determined by optimizing the function $f_Q = f_{grad} + f_{edge} + f_{var} + f_{mean}$, using a target mean value of 124. The optimization achieved a mean value of 130 (instead of 185 in Figure 1c), along with a standard deviation of 40.5. The histogram is shown in Figure 2c. The resulting image 1d is correspondingly darker than 1c.

### 4.5. The histogram equalization term $f_{hist}$

Lighting design optimized using the terms introduced so far has the tendency to inhibit shadows and highlights, often producing large areas with very uniform shading, such as the side of the ship's body in Figure 1d. This is manifested by the histogram in Figure 2c, which shows that most of the pixels have luminances in a rather narrow range around 160–170. In order to address these deficiencies we introduce another term $f_{hist}$ designed to make the image histogram closer in shape to a more equalized histogram. This term tends to increase the variance, conflicting with the $f_{var}$ term, but in practice it turns out that it is the proper balance between these two terms that generates the best results.

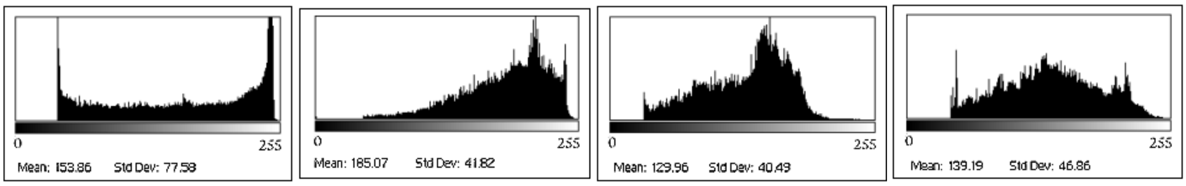
**Example:** Figure 1e was rendered using lighting parameters determined by optimizing the function $f_Q = f_{grad} + f_{edge} + f_{var} + f_{mean} + f_{hist}$. Comparing this image with Figure 1d we see improved shading on the side of the ship's body, as well as more highlights (on the sails) and shadows.



(a) $f_{grad}$ term only (b) Adding $f_{edge}$ (c) Adding $f_{var}$

(d) Adding $f_{mean}$ (e) Adding $f_{hist}$ (f) Adding $f_{dir}$

**Figure 1:** *Quality function: cumulative effect of target terms on the image.*




(a) $f_{grad} + f_{edge}$ (b) Adding $f_{var}$ (c) Adding $f_{mean}$ (d) Adding $f_{hist}$

**Figure 2:** *Quality function: cumulative effect of target terms on the image histogram.*

Note the corresponding change in the shape of the histogram in Figure 2d.

### 4.6. Light direction term $f_{dir}$

Psychophysical tests indicate that it is sometimes easier for the human visual system to correctly interpret a 3D shape when it is illuminated from above. However, it is not clear how general and fundamental this phenomenon is. Furthermore, it is not clear what is the necessary elevation above the horizon, and in many scenes a horizontal illumination direction appears to yield the best results. Therefore, we decided to add to the quality function a term that constrains the light source to illuminate the scene from above, but we treat this term as *optional*, as opposed to the first five terms, which are considered *fundamental*. This term simply measures the difference between the elevation of the light sources and some target elevation angle.

**Example:** Figure 1f was rendered using two light sources: the parameters of the first light were determined by optimizing the complete quality function $f_Q = f_{grad} + f_{edge} + f_{var} + f_{mean} + f_{hist} + f_{dir}$. A secondary light source was fixed at the viewpoint, and could not be modified by the optimization process. Note that both images 1e and 1f, are quite satisfactory in terms of the lighting, despite significant differences in appearance due to the additional constraint on the illumination direction.

### 4.7. Summary

Above we have introduced the six target terms comprising our overall perceptual image quality function. Each of these terms measures the "quality" of a certain feature in the input image by computing a distance from an ideal target value. The target terms were designed so as to encourage the following desired features: The image should conspicuously show the edges of the scene (feature and reflectance edges), and introduce shading gradients on scene surfaces, such that these gradients are sufficiently strong to be noticeable. The global appearance of the image should be such that most regions are displayed by intensities within a limited range, centered at some mid-intensity level, and yet some shadowed and highlighted regions are still allowed to exist in relatively small regions, or in regions that do not contain important details and shape information.

Although the resulting quality function may appear to be overconstrained, our experiments have shown that each and every one of the five fundamantal target components is necessary in order to ensure satisfactory results, as demonstrated by the five examples in Figure 3.

## 5. Perceptual Image Quality Function: Practice

While the previous section described the principles around which our perceptual quality function was designed, in this section we give the precise definitions of the target terms, while addressing some of the issues that must be resolved in order to use this function in practice, in the context of automatic lighting design. These issues are:

**Normalization:** each of the target terms of $f_Q$ must be normalized such that the values they generate all lie in the same range, e.g., $[0,1]$. This is a critical requirement, because if the values produced by one term are significantly smaller than those produced by the rest, the effect of that term on the behavior of $f_Q$ is minor, and in practice it is as if this component was totally excluded, leading to deficiencies such as those shown by fig. 3. On the other hand, if a term's values are much larger than the rest, the quality function will be dominated by this term, and the desired balance will not be achieved.

**Weights:** rather than simply taking the sum of the normalized target terms as the quality function $f_Q$, we assign each target term $f_i$ a weight $w_i$, and define

$$f_Q = \sum w_i f_i \qquad i \in \{grad, edge, var, mean, hist, dir\}$$

The weights $w_i$ can be used to manipulate the dominance of the different target terms, and thus to control the sensitivity of the quality function to different features in the image. In our system we use the following empirically determined weights: $w_{grad} = 0.6$, $w_{edge} = 0.6$, $w_{var} = 0.5$, $w_{mean} = 0.4$, $w_{hist} = 0.55$, and $w_{dir} = 0.5$ (or 0, when $f_{dir}$ is disabled).

**Target values:** most of the target terms in the previous section were defined as a difference between some measured quantity in the image, and a target value for that quantity. The target values typically depend on the particular scene and viewing parameters, and we will discuss how to determine them for each target term.

**Auxiliary data structures:** in order to efficiently evaluate the target terms we will need an auxiliary data structure, which will be precomputed before the lighting design optimization process begins. We refer to this data structure as the *precomputed image map* (PIM). The PIM is essentially a classification of the pixels in the image into three categories: background pixels, edge pixels, and surface pixels.

### 5.1. The detected edges term $f_{edge}$

Recall that the goal of this term is to make sure that the lighting parameters are chosen so as to accentuate, as much as possible, all of the feature and reflectance edges in the scene that can potentially be visible from the specified viewing position. In other words, if an image pixel has been marked as an edge pixel in the PIM, we would like the lighting to create an easily perceived edge at that pixel. To that end, the $f_{edge}$ term is defined as

$$f_{edge}(I) = \frac{1}{N}\left(N - \sum_{(i,j)\in E} O_{edge}(p_{ij})\right), \tag{2}$$

where $I$ is a luminance image, $E$ is the set of edge pixels in the PIM, $N$ is the total number of edge pixels in the PIM, $p_{ij}$ is the $(i,j)$-th pixel of $I$, and $O_{edge}$ is an edge detection operator, defined as follows:

$$O_{edge} = O_{thresh} \cdot O_{zc},$$

where $O_{zc}$ is 1 over pixels where a Laplacian zero-crossing occurs and zero otherwise, and

$$O_{thresh}(p_{ij}) = \begin{cases} 1 & t_{max} < |\nabla p_{ij}| \\ \frac{|\nabla p_{ij}| - t_{min}}{t_{max} - t_{min}} & t_{min} < |\nabla p_{ij}| < t_{max} \\ 0 & |\nabla p_{ij}| < t_{min} \end{cases} \tag{3}$$

where $t_{min}$ and $t_{max}$ are edge detection thresholds, and $|\nabla p_{ij}|$ is the magnitude of the gradient at location $p_{ij}$ in the luminance image $I$.

$f_{edge}$ produces values in the range $[0,1]$. It achieves the optimal value of 0 only if at all edge pixel locations the luminance gradient exceeds a threshold of $t_{max}$ and a Laplacian zero crossing occurs. The worst case value of 1 is obtained if none of these pixels exhibit a gradient exceeding at least $t_{min}$.

### 5.2. The shading gradient term $f_{grad}$

The goal of the $f_{grad}$ term is to encourage the presence of shading gradients at surface pixels, providing important perceptual cues regarding the shape and the orientation of object surfaces in the scene. The term measures the average shading over all surface pixels $g(I)$, and computes the difference between this average and a target value $g_t$, which represents the largest average shading gradient that can possibly be measured for this scene. Clearly, this target value strongly depends on the particular scene and viewing parameters: scenes containing shiny curved objects will naturally exhibit much stronger shading gradients than a scene consisting of matte polyhedra.

Accurate calculation of the target value does not appear

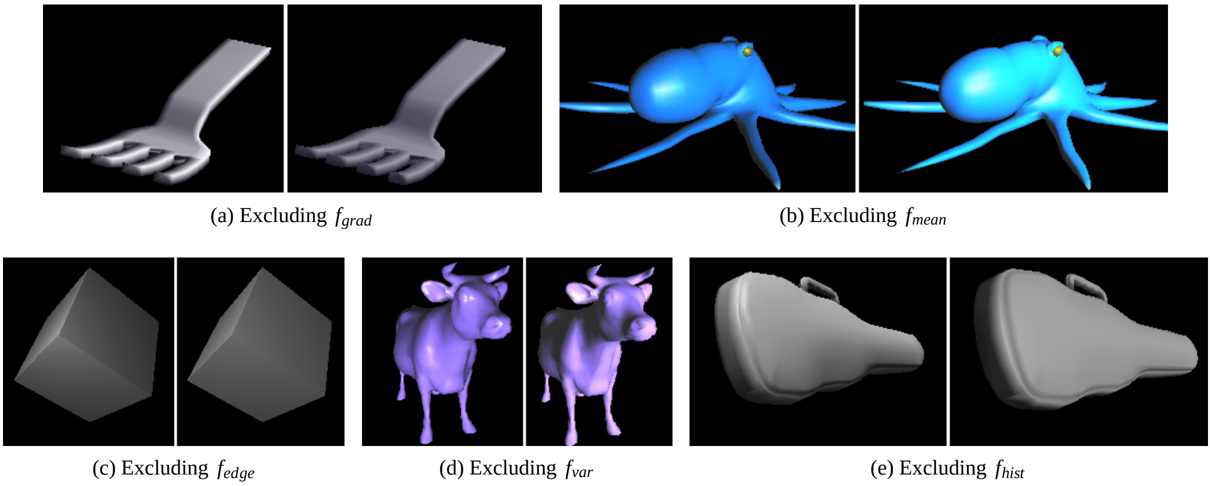


**Figure 3:** *Each of the five examples (a)–(e) in this figure demonstrated the effect of omitting a single target term from $f_Q$. The left image in each example was rendered with lighting parameters obtained using the complete quality function, while the right image shows the effect of excludinga single component. (a) Excluding the $f_{grad}$ term causes the right image to appear substantially duller than the left. (b) Excluding $f_{mean}$ results in a partial loss of highlights and an undesirable shift in material color. (c) Excluding the $f_{edge}$ term causes one of the cube edges to almost disappear. (d) Exclusion of $f_{var}$ results in undesireable under- and over-illuminated regions on the cow. (e) Exclusion of $f_{hist}$ results in a flatter appearance of the curved side of the violin case.*

to be analytically solvable for general scenes, and an accurate numeric solution would be similar in terms of computational expense to the entire lighting design optimization. Fortunately, our experience has shown that an educated guess of the target value is sufficient for satisfactory results. In our implementation we base this guess on a single rendering of the scene: the idea is to predict what illumination conditions would generate the strongest shading gradients[24], set the lighting parameters accordingly, render the scene and compute the average shading gradient in the resulting image.

The average shading gradient $g(I)$ is defined as:

$$g(I) = \sqrt{\frac{1}{N} \sum_{(i,j)\in S} \left|\nabla p_{ij}\right|^2}$$

where $S$ is the set of surface pixels in the PIM and $N$ is the total number of these pixels, and the shading gradient target term is defined as

$$f_{grad}(I) = \begin{cases} \frac{g_t - g(I)}{(1-\alpha)g_t} & \text{if } g(I) \geq \alpha g_t \\ 1 & \text{otherwise} \end{cases}$$

The parameter $\alpha$ is used to achieve proper scaling for the values generated by this term. The value $\alpha = 0.3$ was found to work well in practice.

### 5.3. The variance term $f_{var}$

The $f_{var}$ term measures the difference between the standard deviation $\sigma(I)$ of the surface pixel luminances and a target standard deviation value, denoted by $\sigma_t$. The result is normalized by the target value:

$$f_{var}(I) = \min\left(\frac{|\sigma(I) - \sigma_t|}{\sigma_t}, 1\right) \tag{4}$$

Our experiments have shown that a target standard deviation between 40 and 45 generally produces satisfactory results. This value is generally valid when all displayed surfaces have uniform reflectance. If the scene contains a wide range of reflectances, the target value is corrected by a factor proportional to the variance in the reflectances of the visible surfaces[24].

### 5.4. The mean term $f_{mean}$

The $f_{mean}$ term measures the difference between the mean intensity of the surface pixels $m(I)$ and a target mean-intensity, denoted by $m_t$. The result is scaled to the interval $[0,1]$:

$$f_{mean}(I) = \frac{|m(I) - m_t|}{\max(m_t, 255 - m_t)} \tag{5}$$

Our experiments have shown that a reasonable target value for $m_t$ is obtained by setting it to around 150. If the scene contains many dark surfaces, the target value is reduced accordingly[24].

### 5.5. The histogram equalization term $f_{hist}$

The $f_{hist}$ term measures the distance between the histogram $h(I)$ of the luminance image and a target histogram $h_t$. In an ideal equalized histogram each luminance values occurs an equal number of times $n_t = N/255$, where $N$ is the number of

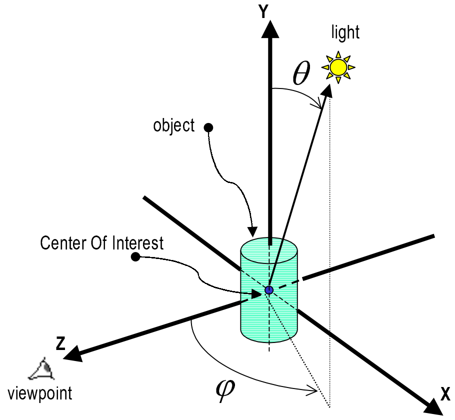



**Figure 4:** *Light source direction is measured in an object centered spherical coordinate system. The polar angle θ is used to measure the elevation of the light source.*

surface pixels. Denoting by $n_i$ the height of the $i$-th column in the actual histogram $h(I)$, the distance between $h(I)$ and $h_t$ is defined as

$$d = \sqrt{\frac{1}{255}\sum_{i=0}^{255}(n_i - n_t)^2} \qquad (6)$$

While searching for appropriate scaling for this term, we found that neither the ideal histogram, nor the worst case histogram (where all pixels have the same luminance) occur in practice. Therefore, the term is scaled using two scaling parameters, $\alpha_1$ and $\alpha_2$, such that $0 < \alpha_1 < \alpha_2 < 1$. The parameter $\alpha_1$ ($\alpha_2$) define a more practical worst (best) case histogram, with all pixels equally distributed in $\alpha_1$ ($\alpha_2$) percent of the columns. Denoting the distance between these two histograms and the ideal histogram by $d_{\alpha_1}$ and $d_{\alpha_2}$, the properly scaled version of $f_{hist}$ is defined as

$$f_{hist}(I) = \begin{cases} 1 & \text{if} \quad d > d_{\alpha_1} \\ \frac{d - d_{\alpha_2}}{d_{\alpha_1} - d_{\alpha_2}} & \text{if} \quad d_{\alpha_2} < d < d_{\alpha_1} \\ 0 & \text{if} \quad d < d_{\alpha_2} \end{cases}$$

In our experiments we set $\alpha_1 = 0.1$ and $\alpha_2 = 0.8$.

### 5.6. The light direction term $f_{dir}$

This optional target term constrains the light direction to come from above. The input to this term are the elevations of the light sources, each given by the polar angle θ, as shown in Figure 4, The term simply measures the differences between each θ and a target polar angle $\theta_t$. As suggested by Curran[7], a target angle $\theta_t = 45°$ was used in our experiments. The properly scaled $f_{dir}$ term is defined as

$$f_{dir} = \begin{cases} \frac{d}{65} & \text{if} \quad d < 65 \\ 1 & \text{otherwise} \end{cases} \qquad (7)$$

where $d$ is the RMS difference between the elevations of the light sources and the target elevation $\theta_t$:

$$d = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\theta_i - \theta_t)^2} \qquad (8)$$

## 6. A Lighting Design System

Using the perception-based quality function developed in the previous sections, we have implemented an automatic lighting design system. A detailed description of this system appears in Shacked's thesis[24]; here we summarize its main features. The input to our system is a model of a 3D scene (including geometry and material properties), a set of viewing parameters for a desired image, and a set of lighting parameters, such as the number of light sources, their positions and/or directions, and their intensities. Each lighting parameter is declared as either free or fixed. The free parameters are those whose values we wish to determine automatically in an optimal fashion using our system. Fixed parameters are assigned initial values by the system. These values are taken into account by the quality metric, but the system is not free to modify them in the course of optimization.

In our current implementation we use a straightforward OpenGL-based rendering tool: Z-buffering for hidden-surface removal, and OpenGL's default shading model[26]. The objects in our scenes are represented as polygon meshes, with per-vertex normals. Each polygon can have its own material properties. As explained earlier in the paper, texture maps are ignored by the optimization process. Viewing parameters are specified using OpenGL's camera model.

The system first performs a self-calibration stage, during which the appropriate target values are established for the different target terms of $f_Q$, taking into account the input scene and viewing parameters, as explained in the previous section. Next, the free and fixed lighting parameters are assigned with initial values. Using the free parameters as optimization variables and the quality function as an objective function, an optimization process is then performed to find the optimal settings for the lighting parameters. The optimization process is an iterative search in the multidimensional space of the free variables, looking for a local minimum of the quality function.

In our current implementation we use a simple steepest descent optimization scheme[19]. To approximate the gradient of the quality function, partial derivatives for the free variables must be computed. We approximate partial derivatives numerically, by taking differential steps at the direction of each free variable, rendering, and evaluating the quality function. Once an optimization direction is chosen, the solution advances in that direction until reaching a minimum, and then a new direction is calculated. When optimizing over more than one free light source, the optimization process alternates its steps between the different light sources.

In our experiments we found that the quality function may have several local minima, and the solution typically converges to the minimum nearest to the initial guess. Searching for a global minimum, in our opinion, is not necessary: local minima provide quite satisfactory lighting designs, if the initial values are selected in an intelligent fashion[24].

### 6.1. Performance and resolution

Unsurprisingly, most of the computation cost is incurred by the optimization loop. In each iteration the scene model must be rendered one or more times in order to evaluate the quality function $f_Q$, and (in some iterations) estimate its partial derivatives, and the resulting images must be converted from RGB to luminance. If the rendering is hardware-assisted, we must also read the resulting image from the frame buffer to the main memory. Thus, the computation cost is directly affected by two main factors: the resolution of the image and the performance of the rendering tool.

Consequently, the lighting design should be performed at the lowest resolution at which the main features in the scene are still visible. Determining such *threshold resolution* for a given model is a topic for future work. Having examined all of our test models, we were able to choose some common resolution, which is the smallest resolution that still yielded a satisfactory solution for all test cases. This resolution was found to be of around 62,000 pixels (e.g., $250^2$). Therefore, in all our experiments we reduced the desired image to 62,000 while preserving its aspect ratio, and performed the optimization at that resolution.

### 6.2. Reducing the number of free parameters

The number of free lighting parameters determines the dimension of the design space and has a significant effect on the computation time of the optimization stage, because of the need to compute the partial derivative of $f_Q$ with respect to each free variable. Therefore, it is desirable to keep the number of free parameters as low as possible.

In our current implementation this reduction is achieved as follows. First, we only consider *positional* light sources located at a *fixed* distance from the scene's center of interest. Positional lights were empirically found to generally yield illumination that better matches the requirements of our quality function, compare to directional lights. We also found that good results are obtained when the distance is set to twice the radius of the scene.

Furthermore, we only consider white light sources (with equal intensity in the *R*, *G*, and *B* components). This guarantees that the colors of the surfaces in the scene are determined by the corresponding material properties, which is part of the information that we would like the image to communicate to an observer. In any case, since our quality function operates in the luminance domain of the image, rendering with colored lights is practically meaningless, as far as the lighting design optimization is concerned.

In OpenGL each light source is specified by means of its diffuse, specular, and ambient intensity. We fix the ambient intensity at 15 percent of the diffuse intensity. Consequently, each free light source specified to our system involves optimizing over 4 free parameters: the two direction parameters $(\theta, \varphi)$, its diffuse intensity, and its specular intensity. The remaining parameters are considered fixed and their values are assigned as described earlier.

### 6.3. Lighting configurations

In order to fully configure the system for performing the automatic optimization process, two further decisions should be made:

1. The number of free light sources that should participate the process.
2. Whether or not the $f_{dir}$ term should be enabled.

In practice, for all the test cases examined, we found that choosing one of the following three configurations is sufficient to obtain satisfactory results:

**Configuration1** $f_{dir}$ is disabled, and a single free light source is optimized.
**Configuration2** $f_{dir}$ is enabled, one free light source is optimized, and a secondary fixed light source is positioned at the viewpoint.
**Configuration3** $f_{dir}$ is disabled, and two free light sources are optimized.

Hence, the only decision that should be made by the user of our system is which of these three configurations should be used. We found that cases where $f_{dir}$ substantially improves the illumination of the scene are rare, and its effect is more to direct the solution to a different image appearance, which may be considered preferable by some perceptual considerations, even if not resolving any crucial visual issues[20]. Using two free lights is often not required, and in some cases may even lead to less satisfactory results than the two other configurations. As a rule of thumb, two free lights should be used only when illumination by a single free light fails to produce satisfactory results.

## 7. Results

We used the system described in the previous section to automatically determine the lighting design in a large number of test scenes, with a variety of different materials and different viewing parameters. In the vast majority of cases the lighting designs produced by the system were found satisfactory in terms of the visual quality goals we set to ourselves in Section 3. We also found these designs to favorably compare with the best designs we were able to generate using manual manipulations of the lighting parameters (the direct design paradigm mentioned in Section 1).

To illustrate the effectiveness of our lighting design system Figure 5 shows a comparison of several different lighting designs on four different models. For each model a row of four images is shown. The images in columns (a) and (b) of each row were rendered using naive lighting settings, which are nevertheless often used in practice as default lighting settings:

**Default1** A single directional light illuminating the model from the direction of the viewpoint.

**Default4** Four directional light sources: a top-left source with spherical coordinates $(\theta,\varphi) = (45^\circ,-45^\circ)$; a top-right source at $(45^\circ,45^\circ)$; a bottom-left source at $(135^\circ,-45^\circ)$; and a bottom-right source at $(135^\circ,45^\circ)$.

The images in columns (c) and (d) are rendered with optimized lighting settings, generated by our system using **Configuration1** and **Configuration2**, as explained in Section 6.3.

The improvements in visual appearance and comprehensibility of the images in columns (c) and (d) over those produced with default lighting are quite apparent, and at times even dramatic. Additional lighting design results and comparisons are included in the electronic archive accompanying this submission, including examples with two free light sources.

Table 1 contains some statistics regarding the four examples shown in Figure 5, as well as the galleon model from Figure 1. In all cases the optimization process, which accounts for 85 percent of the total execution time, takes only a few seconds. Around three quarters of this time were spent rendering images and reading the frame buffer. The remaining quarter was spent computing $f_Q$.

## 8. Summary and Future Work

We have presented a fully automatic lighting design system for traditional (photorealistic) rendering of 3D models. The lighting designs are obtained by optimizing a perception-based image quality function, yielding comprehensible images of 3D object, which effectively communicate information about shapes, materials, and spatial relationships.

There are many promising directions for future research, some of which are outlined in the remainder of this Section.

**System automation:** the automation of the system can be further enhanced by developing methods that based on analysis of the scene, viewing-parameters and perhaps the rendering algorithm, can automatically determine the preferred number of free light sources, and recommend whether the optional $f_{dir}$ term should be enabled. An automatic method for establishing the appropriate *threshold resolution* for each scene would also prove very useful for speeding up the optimization process.

**Optimization:** more sophisticated optimization methods should be considered and implemented. Using techniques that reduce the number of optimization iterations performed, as well as the total number of function evaluations will significantly enhance system performance.

**View-independent solutions:** extending the system to find view-independent lighting solutions can be very useful. However, this is a difficult task, due to the fact that the core of the system is an analysis performed on a 2D image created for a particular viewpoint.

**Global illumination:** extending the system to work with global illumination models requires proper handling of additional effects such as cast shadows, inter-reflections and refraction, as well as a more efficient optimization process — since rendering costs will be much higher. The technique described in this paper could be used to provide an initial guess for the more expensive global illumination based lighting design system.

**Quality function:** clearly, the formulation and implementation given for the quality function is not unique and definitive. Although several methods of measuring and evaluating the visual and geometrical information were examined, applying different techniques may be proven beneficial. One example is the use of more accurate edges evaluation technique, which takes into account a scale of expected prominence for various edges. Using other visual information, such as spatial frequencies or colors for perceptual quality evaluation may be considered. Various features of the HVS, which affects the response of human to the visual information may be integrated into the quality metric. For instance, the non-linear response and threshold sensitivity of the HVS to luminance contrast can be used to enhance the accuracy of measuring the effect of shading gradients.

## References

1. E. Adelson and A. Pentland. The perception of shading and reflectance. In D. Knill and W. Richards, editors, *Perception as Bayesian Inference*, chapter 11. Cambridge University Press, 1996. 3

2. V. Bruce and P. R. Green. *Visual Perception: Physiology, Psychology and Ecology*, chapter 5–9. Lawrence Erlbaum Associates Ltd., second edition, 1991. 3

3. P. Cavanagh and Y. G. Leclerc. Shape from shadows. *Journal of Experimental Psychology: Human Perception and Performance*, 15(1):3–27, 1989. 3

4. A. Chalmers, S. Daly, A. McNamara, K. Myszkowski, and T. Troscianko. Image quality metrics. Siggraph 2000 course notes, July 2000. 3

5. S. Coren and L. M. Ward. *Sensation and Perception.* Harcourt Brace Jovanovich, third edition, 1989. 3

6. A. C. Costa, A. A. de Sousa, and F. N. Ferreira. Lighting design: A goal based approach using optimization. In *Rendering Techniques '99*, pages 317–328. Springer-Verlag, 1999. 1, 2

7. W. Curran and A. Johnston. The effect of illuminant position on perceived curvature. *Vision Research*, 36(10):1399–1410, 1996. 3, 8

8. K. K. De Valois and F. Kooi. The role of color in spatial vision. In L. Harris and M. Jenkin, editors, *Proc. 1991 conference on spatial vision in humans and robots.*, pages 149–159. Cambridge University Press, 1993. 3

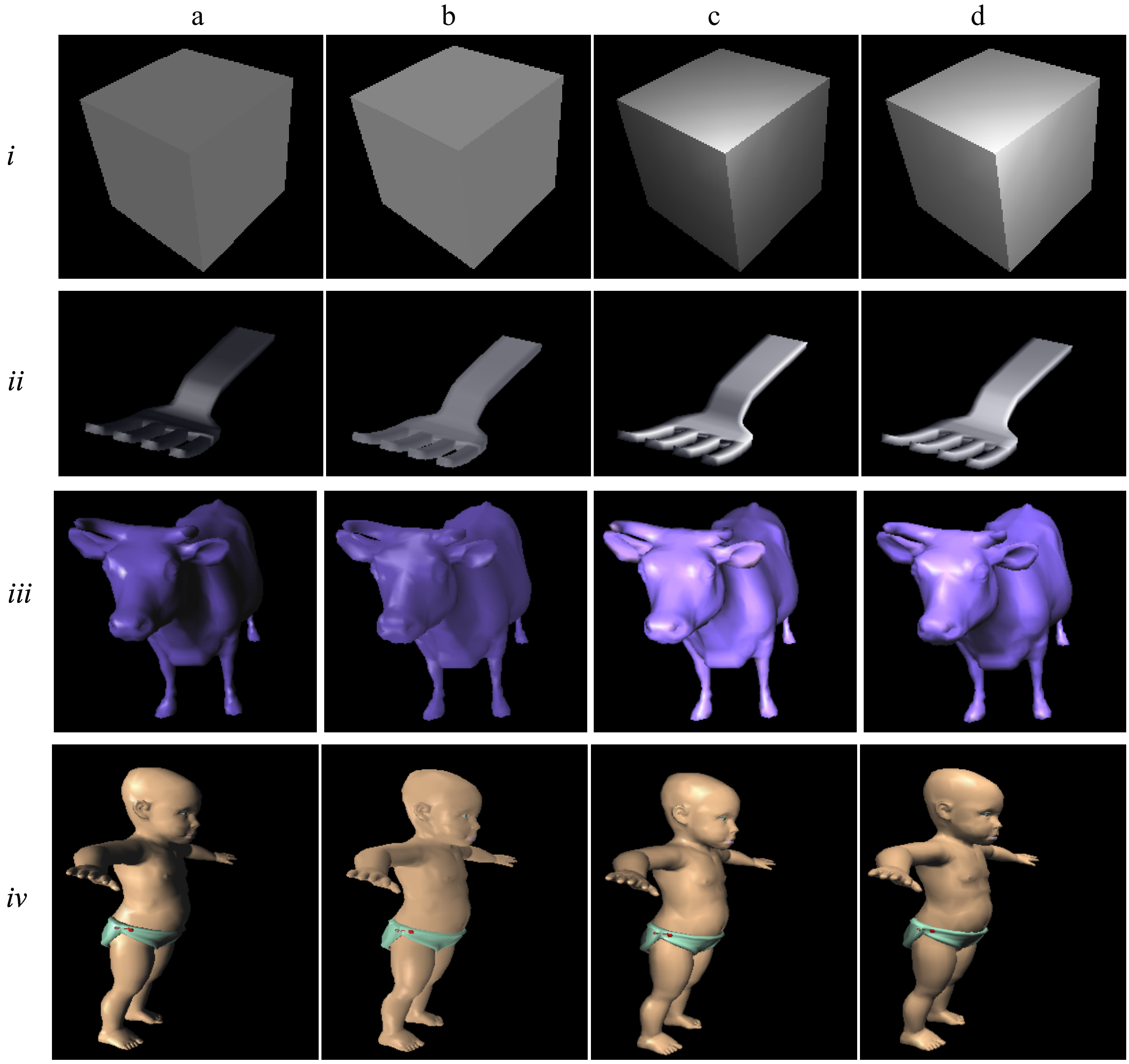



**Figure 5:** *Lighting design results on four different models. Images in columns (a) and (b) were rendered using the Default1 and Default4 lighting settings. Columns (c) and (d) were rendered using two lighting designs obtained by our optimization process, using Configuration1 and Configuration2, respectively, as described in Section 6.3.*
***Row** i **(Cube):** Default lighting in (a) and (b) results in totally uniform shading on the cube's faces and fails to display all edges. Optimized lighting: in (c) and (d) all edges are prominent; shading gradients on the cube's faces convey enhanced 3D impression.*
***Row** ii **(Fork):** Default lighting in (a) and (b) fails to properly convey material and shape information in several locations, mainly due to lack of illumination in (a) and lack of shading gradients in (b). Optimized lighting: (c) major improvement are introduced by both highlights and shadows; (d) similar improvements with illumination coming from above.*
***Row** iii **(Cow):** Default lighting: (a) shadowed regions hide shape information, image is generally too dark; (b) flat illumination lacking shading gradients causes loss of shape and edge information. Optimized lighting: (c) a major improvement in overall brightness; shape features are much more visible; (d) similar improvements with illumination coming from above.*
***Row** iv **(Baby):** Default lighting: (a) body shape and details are lost in the shadowed far side and in the near arm and hand; (b) too much illumination yields a washed-out, low-contrast appearance. Optimized lighting: in (c) and (d) all body parts are properly illuminated, the contrast is just right, and many details become visible.*

| Model name | # of polygons | Resolution | Figure | # of optimization iterations | # of renderings performed | Optimization time (sec) |
|---|---|---|---|---|---|---|
| Cube | 54 | 200 x 200 | 5 *i*-c | 9 | 38 | 2.0 |
| | | | 5 *i*-d | 13 | 47 | 2.4 |
| Fork | 812 | 167 x 117 | 5 *ii*-c | 12 | 41 | 2.2 |
| | | | 5 *ii*-d | 7 | 36 | 2.1 |
| Cow | 5,805 | 186 x 208 | 5 *iii*-c | 11 | 39 | 3.5 |
| | | | 5 *iii*-d | 4 | 20 | 2.3 |
| Baby | 12,712 | 193 x 318 | 5 *iv*-c | 5 | 21 | 3.2 |
| | | | 5 *iv*-d | 5 | 25 | 3.6 |
| Galleon | 4,699 | 362 x 389 | 1 (e) | 8 | 35 | 4.4 |
| | | | 1 (f) | 9 | 57 | 6.5 |

**Table 1:** *Lighting design process statistics for the five test models shown in this paper. Times were measured on a 866 MHz Pentium III PC with Nvidia GeForce2 graphics accelerator.*


9. A. S. Glassner. *Principles of Digital Image Synthesis*, volume 1, chapter 1: The human visual system. Morgan Kaufmann, 1995. 3
10. A. Gooch, B. Gooch, P. Shirley, and E. Cohen. A non-photorealistic lighting model for automatic technical illustration. In *Computer Graphics* Proceedings, *Annual Conference Series*, pages 447–452, Aug. 1998. 2, 3, 4
11. J. K. Kawai, J. S. Painter, and M. F. Cohen. Radioptimization — goal-based rendering. In *Computer Graphics* Proceedings, *Annual Conference Series*, pages 147–154, 1993. 1, 2
12. J. J. Koenderink. What does the occluding contour tell us about solid shape? *Perception*, 13:321–330, 1984. 3
13. L. Markosian, M. A. Kowalski, S. J. Trychin, L. D. Bourdev, D. Goldstein, and J. F. Hughes. Real-time nonphotorealistic rendering. In T. Whitted, editor, *SIGGRAPH '97 Conference Proceedings*, pages 415–420. Addison Wesley, Aug. 1997. 4
14. J. Marks, B. Andalman, P. A. Beardsley, W. Freeman, S. Gibson, J. Hodgins, T. Kang, B. Mirtich, H. Pfister, W. Ruml, K. Ryall, J. Seims, and S. Shieber. Design Galleries: A general approach to setting parameters for computer graphics and animation. In *SIGGRAPH '97 Conference Proceedings*, pages 389–400. Addison Wesley, Aug. 1997. 2
15. D. Marr. *Vision. A Computational Investigation into the Human Representation of Visual Information.* Freeman, New York, 1982. 3
16. E. Mingolla and J. T. Todd. Perception of solid shape from shading. *Biological Cybernetics*, 53:137–151, 1986. 3
17. P. Poulin and A. Fournier. Lights from highlights and shadows. In D. Zeltzer, editor, *Proceedings of the 1992 Symposium on Interactive 3D Graphics*, pages 31–38, Mar. 1992. 2
18. P. Poulin, K. Ratib, and M. Jacques. Sketching shadows and highlights to position lights. In *Proceedings of Computer Graphics International 97*, pages 56–63, 1997. 1, 2
19. W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing.* Cambridge University Press, second edition, 1992. 8
20. V. S. Ramachandran. Perception of shape from shading. *Nature*, 331(14):163–166, 1988. 3, 9
21. H. Rushmeier, G. J. Ward, C. Piatko, P. Sanders, and B. Rust. Comparing real and synthetic images: Some ideas about metrics. In P. M. Hanrahan and W. Purgathofer, editors, *Rendering Techniques '95*, pages 82–91. Springer-Verlag, 1995. 2
22. T. Saito and T. Takahashi. Comprehensible rendering of 3-D shapes. *Computer Graphics*, 24(4):197–206, 1990. 2, 3, 4
23. C. Schoeneman, J. Dorsey, B. Smits, J. Arvo, and D. Greenberg. Painting with light. In *Computer Graphics* Proceedings, *Annual Conference Series*, pages 143–146, 1993. 1, 2
24. R. Shacked. Automatic lighting design using a perceptual quality metric. Master's thesis, The Hebrew University of Jerusalem, Israel, 2001. http://www.cs.huji.ac.il/~danix/ldesign. 3, 7, 8, 9
25. J. T. Todd and E. Mingolla. Perception of surface curvature and direction of illumination from patterns of shading. *Journal of Experimental Psychology: Human Perception and Performance*, 9(4):583–595, 1983. 3
26. M. Woo, J. Neider, and T. Davis. *OpenGL Programming Guide.* Addison-Wesley Developers Press, second edition, 1997. 8