

Shading Models for Point and Linear Sources

TOMOYUKI NISHITA
Fukuyama University

ISAO OKAMURA and EIHACHIRO NAKAMAE
Hiroshima University

The degree of realism of the shaded image of a three-dimensional scene depends on the successful simulation of shading effects. The shading model has two main ingredients, properties of the surface and properties of the illumination falling on it. Most previous work has concentrated on the former rather than the latter.

This paper presents an improved version for generating scenes illuminated by point and linear light sources. The procedure can include intensity distributions for point light sources and output both umbrae and penumbrae for linear light sources, assuming the environment is composed of convex polyhedra. This paper generalizes Crow's procedure for computing shadows by using shadow volumes to compute the shading of umbrae and penumbrae. Using shadow volumes caused by the end points of the linear source results in an easy determination of the regions of penumbrae and umbrae on the face prior to shading calculation.

This paper also discusses a method for displaying illuminance distribution on a shaded image by using colored isolux contours.

Categories and Subject Descriptors: I.3.7 [**Computer Graphics**]: Three-Dimensional Graphics and Realism—*color, shading, shadowing, and texture*

General Terms: Algorithms

Additional Key Words and Phrases: Lighting simulation, luminous intensity distribution

1. INTRODUCTION

In order to display three-dimensional objects that look more realistic, researchers have developed techniques for simulating the properties of objects such as reflection, refraction, and transparency [8, 11, 14]. The light sources used in those techniques, however, have been limited to parallel light sources or point light sources.

In order to correctly describe the effects of real light sources, the spatial distribution of luminous intensity varying with direction and geometry of sources must be considered [12, 13]. In most previous works, however, all rays from a

This work was supported in part by Hosono-Bunka Foundation.

Authors addresses: T. Nishita, Faculty of Engineering, Fukuyama University, 985 Sanzo, Higashimura-cho, Fukuyama 729-02, Japan; I. Okamura and E. Nakamae, Faculty of Engineering, Hiroshima University, Saijo-cho, Higashihiroshima, 724, Japan.

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

© 1985 ACM 0730-0301/85/0400-0124 \$00.75

ACM Transactions on Graphics, Vol. 4, No. 2, April 1985, Pages 124-146.

point source were handled as uniform-intense, although the spatial distribution of the emittance usually varied with direction. In practical cases, the luminous intensity distribution is indispensable, especially for point sources to simulate the illumination of an environment. As for the geometry of the light sources, handling only parallel light sources and point light sources is insufficient because we have linear light sources, and area or volume light sources [10]. These types of sources create umbrae and penumbrae.

Shadows provide effective information concerning positional relationships between many objects and give the observer accurate comprehension of complex spatial environments. However, most previous algorithms have handled only umbrae. A method of fading the boundaries of shadows by means of dithering has been presented, but it is just an approximation [15]. Atherton et al. and Whitted have respectively pointed out the necessity of finding an algorithm which displays umbrae and penumbrae [2] and of handling distributed light sources [14]. Verbeck [12] recently presented methods for simulating the distributed light sources by using ray tracing and Brotman [4] presented methods by using depth buffer algorithms. In these methods, light sources are assumed as sets of point sources.

This paper proposes methods for displaying three-dimensional objects that are illuminated by point sources with luminous intensity distribution or by perfectly diffusing linear sources (Lambertian (cosine) distribution). We also present a display method of isolux contours depicted by color belts which are superimposed on a perspective image. By using this depiction, we can easily grasp the illuminance distributions.

The algorithms described in this paper apply to objects composed of convex polyhedra and the method of hidden surface removal is fundamentally based on [9] (see Appendix A1).

2. FUNDAMENTAL IDEA OF SHADOW PROCESSING

The algorithms developed so far for parallel light sources or point sources fall into three categories:

- (1) The removal of hidden surfaces and the detection of shadow boundaries are executed on each scan line when an image is produced [1, 3].
- (2) The shadowed areas on each polygon are detected prior to removal of hidden surfaces [2, 9].¹
- (3) Every volume of space swept out by the shadow of each object is obtained before removing hidden surfaces [6].

In this paper, the authors employ the second algorithm extended to treat point light sources with luminous intensity distribution. This algorithm uses the overlap test in the hidden surface algorithm (see Appendix 1) twice, once for the light source and once for the viewpoint. The algorithm detects shadow boundaries on a perspective plane observed from the light source (specified only by a direction

¹ The basic idea of [9] is introduced in [2].

for a parallel light source). For parallel sources and point sources existing outside the field of view, a suitable perspective plane can be determined. However, for point sources existing within the field of view, all vertices of objects can not always be projected onto the perspective plane. In this case, a procedure in object space is necessary. We propose a shadow detection method which operates on the perspective plane or on the object space as required (see Section 3.2).

None of the three algorithms mentioned above can be applied directly to shadow detection for linear light sources. Thus we are introducing the idea of detecting the shadowed areas for linear light sources prior to removal of hidden surfaces in the second algorithm and the concept of shadow volumes for linear light sources in the third algorithm. The proposed algorithm is executed by the following three steps. First, the volumes of umbrae and penumbrae are calculated. Second, the umbra and penumbra areas on each face are calculated by using the shadow volumes. Finally, the intensity of the faces is calculated on each scan line when the hidden surfaces are removed. The shading calculations are simplified by the prior detection of the shadow boundaries, at the expense of some additional storage.

3. POINT SOURCES WITH LUMINOUS INTENSITY DISTRIBUTION

3.1 Regions Needed for Shadow Detection and Luminous Intensity Distribution Curve

The distribution of the emittance of a point source varies with direction. In this paper, we limit our discussion to point sources having rotationally symmetric intensity distribution characteristics. In this case, the luminous intensity distribution is expressed by a curve as shown in Figure 1; it is called the luminous intensity distribution curve in illuminating engineering. The curve represents the variation of luminous intensity of a lamp on a plane including the light center and the illumination axis.

The intensity levels of display devices are finite. In some cases, the illuminance on the faces is very low because the illuminance decreases as the square of the distance from the light source. In addition, the reflector of a lamp often restricts the illuminating space of the light source. These facts suggest that some faces far from the source can be omitted from the shadow detection. As shown by a dotted line in Figure 2a and b, the size of shadow detection volume is proportional to the luminous intensity distribution curve. For simplicity of shadow detection, we use a partial sphere of lighting shown by the dot-dash lines in this figure. The partial spheres are defined as the part of a sphere bounding a space where the illuminance can not be neglected. In Figure 2a, θ_c shows the beam spread and r_a is the attainable distance of the light which depends on the maximum luminous intensity and the intensity levels (e.g., 8 bits) of the display device. That is, since the display intensity resolution is only 8 bits, at a certain distance from the light source, the effect of the lighting is insignificant (less than 1/256 of the maximum intensity). When θ_c is larger than ninety degrees (see Figure 2b) two spheres are required, upper and lower, with radii r_u and r_l , respectively, because in many cases the luminous intensity distribution characteristics are very different. The partial spheres of lighting are used only to reduce shadow detection computations (see Section 3.2).

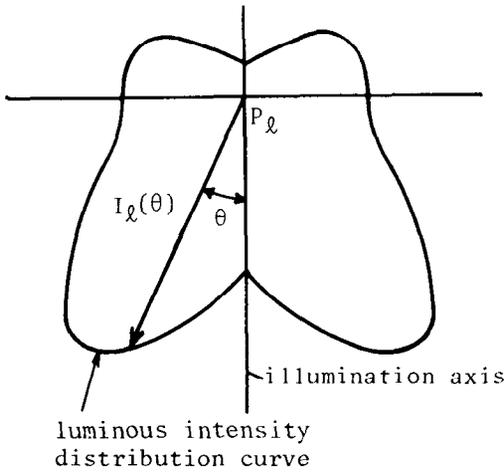


Fig. 1. Luminous intensity distribution curve (used for the street lamp in Fig. 11c).

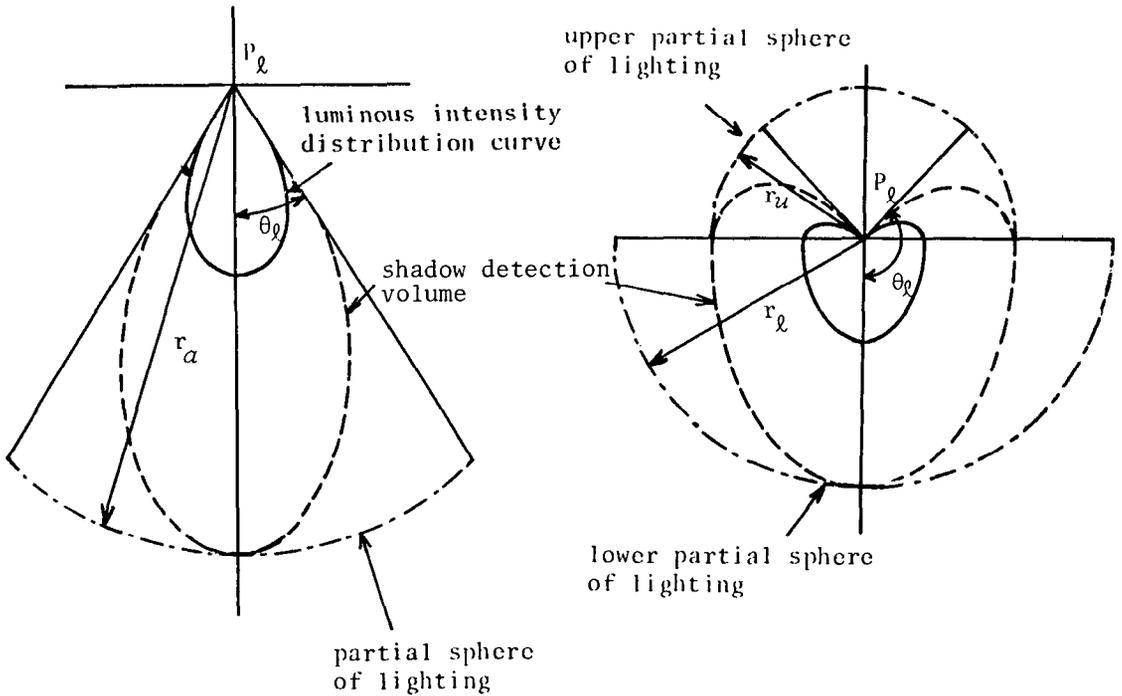


Fig. 2. Side view of luminous intensity distribution curves and partial spheres of lighting. (a) $|\theta_l| \leq \pi/2$. (b) $|\theta_l| > \pi/2$.

Actual luminous intensity distributions depend on a very large number of factors and are extremely complex. Simplifications can be made, however, that produce reasonably pleasing results. Two approximations are used in the images produced in this paper:

(1) Most luminous intensity distributions can be expressed by a function of $\cos\theta$ where θ is the angle between the illumination axis and the lighting ray. We

prepared some typical luminous intensity curves such as $I_r(\theta) = I(1 + \cos\theta)/2$ (I is given as the luminous intensity at $\theta = 0^\circ$).

(2) The luminous intensity data are given, for example, every ten degrees, ($\theta_i = 0^\circ, 10^\circ, \dots, 180^\circ$). Intervening luminous intensities are calculated by using linear interpolation, that is, the intensity $I_r(\theta)$ of angle θ , where $\theta_i \leq \theta < \theta_{i+1}$, is expressed by

$$I_r(\theta) = I_r(\theta_i) + (I_r(\theta_{i+1}) - I_r(\theta_i))t, \quad (1)$$

where

$$\begin{aligned} t &= (\theta - \theta_i)/(\theta_{i+1} - \theta_i) \\ &\approx (\cos\theta - \cos\theta_i)/(\cos\theta_{i+1} - \cos\theta_i), \end{aligned}$$

t is approximated because θ is obtained as the function of $\cos\theta$, as mentioned in Section 3.3. In order to avoid excessive cosine calculations, two look-up tables, $\cos\theta$, and $(\cos\theta_{i+1} - \cos\theta_i)^{-1}$, are used.

3.2 Shadow Detection

In deciding whether or not one convex polyhedron casts its shadows on another, we assume that the viewpoint is at the point source and that the perspective plane includes an arbitrary point on the illumination axis (see Q_c in Figure 3).

For a single point light source and convex polyhedra, shadow detection is performed for every pair of polyhedra, and each shadow boundary is stored as a convex polygon on a face. Then, the shadow areas on the face are obtained by the union of the shadows cast on it. This makes easy scanning of shadow boundaries because of convex shadows.

As shown in Figure 3a, when both convex polyhedra can be projected onto the perspective plane, shadow boundaries are calculated on the perspective plane by using the projected contour lines of the convex polyhedra (see [9]). Otherwise, the shadow detection is performed in object space using pyramids of infinite height that are formed by the point source and the contour edges of the convex polyhedra (see Figure 3b). If the pyramids for two convex polyhedra intersect each other, there is a shadow area on the farther polyhedron from the point source (in Figure 3b, V_1), and the shadow area is the part intersecting with the pyramid made from the closer convex polyhedron (in Figure 3b, V_2). This computation is easy because the pyramids are convex and always have a common vertex, the point source.

The reduction of shadow tests is significant when there are many objects and multiple light sources. Disregarding reflections, the polyhedra requiring shadow display are restricted within the field of view, while polyhedra outside the field of view can only act to cast their shadows on the polyhedra inside the field of view. By using the field of view and the partial sphere of lighting, the number of combinations for the shadow detection is reduced from $n_0(n_0 - 1)/2$ (i.e., n_0C_2) to $n_p(n_p - 1)/2 + n_p \cdot n_o$, where n_0 is the total number of convex polyhedra, n_p the number of the polyhedra within the field of view and the partial sphere of lighting, and n_o the number of polyhedra existing outside the field of view but within the partial sphere of lighting.

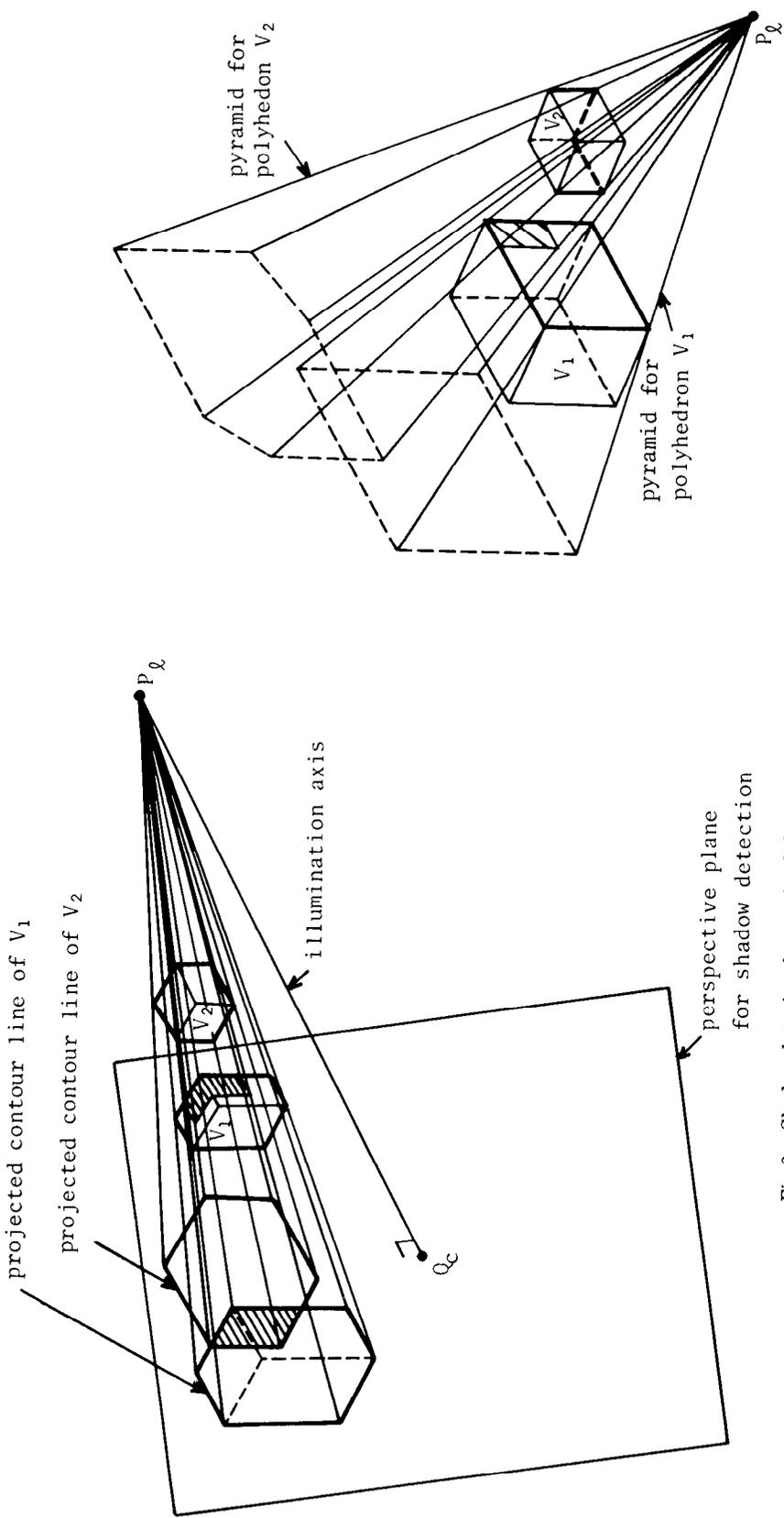


Fig. 3. Shadow detection for a point light source. (a) Shadow detection on perspective plane (b) Shadow detection object plane.

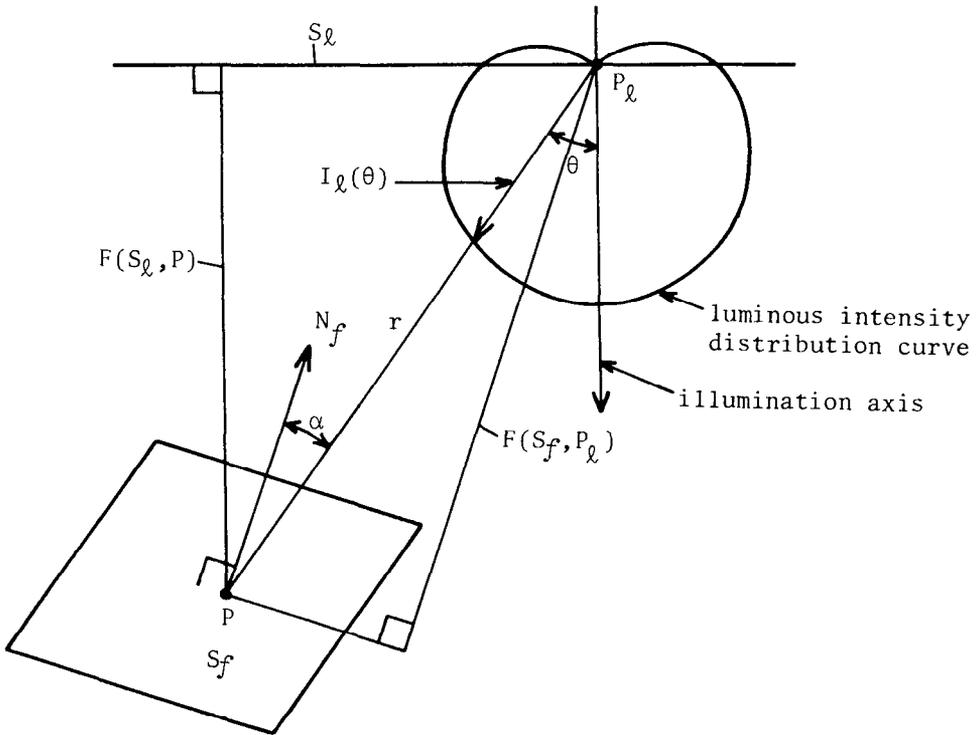


Fig. 4. Illuminance calculation for a point light source.

3.3 Shading

In order to save computation time, we use a technique similar to Gouraud shading [7]. The calculation of illuminance on faces along each scan is executed not only at the boundaries of the faces such as Gouraud shading but also at the boundaries of shadows and at regular intervals (three or five pixel intervals are used in this paper). The illuminance of pixels within each interval is computed by linear interpolation.

The illuminance, E_l , from a light source at P_l , at an arbitrary point P on a face S_f is expressed by

$$E_l = I_l(\theta) \frac{\cos\alpha}{r^2} \tag{2}$$

where r is the distance between P_l and P , α is the angle between the normal of a face S_f and the light ray from the source, and θ is the angle between the illumination axis and the light ray from P_l to P (see Figure 4).

Let $F(S, Q)$ be the perpendicular distance between the plane of polygon S and a point Q . Let the coefficients of the plane of S be (a, b, c, d) where (a, b, c) are the coefficients of the unit outward pointing normal to the plane of S and let $Q = (X, Y, Z)$. Then

$$F(S, Q) = aX + bY + cZ + d. \tag{3}$$

By using eq. (3), $\cos \alpha$ is easily obtained as follows:

$$\cos \alpha = \frac{F(S_f, P_r)}{r},$$

then eq. (2) is expressed by

$$E_r = \frac{I_r(\theta) F(S_f, P_r)}{r^3}. \quad (4)$$

$I_r(\theta)$ in eq. (4) is easily calculated by using the plane S_r that is perpendicular to the illumination axis and includes P_r (see Figure 4), because the luminous intensity of a point source is expressed by a function of $\cos \theta$; that is

$$\cos \theta = \frac{F(S_r, P)}{r}.$$

4. LINEAR SOURCES

The shape of a linear light source discussed in this section is assumed to be a line segment, and the luminous characteristics are Lambertian distribution and uniform brightness.

In this section, we discuss the shadow detection of penumbrae and umbrae and the shading calculation for the linear source.

4.1 Shadow Detection

4.1.1 Penumbra and Umbra Volumes for Shadow Detection. The shadows caused by a linear source consist of umbrae and penumbrae as mentioned before. Because objects are treated as sets of convex polyhedra, it is easy to determine the shadow volumes for shadow detection.

As shown in Figure 5, let us assume an arbitrary face S_f exists farther from a light source than a convex polyhedron V . We define the contour lines C_1 and C_2 , where C_1 is the silhouette contour of V when viewed from an end point Q_1 of the source and C_2 is the silhouette contour made by Q_2 . When these two contour lines are projected onto face S_f , the intersection of the polygonal area, as defined by the projected contour lines C'_1 and C'_2 , makes an umbra area (as shown by a cross-hatching in Figure 5). The minimum convex polygon surrounding C'_1 and C'_2 forms a penumbra area (as shown by a hatching and dotting in Figure 5).

Note that the penumbra should not be described as the union of the areas defined by C'_1 and C'_2 , but the convex hull of those areas. In the dotted regions of Figure 5, although both Q_1 and Q_2 are visible, a part of the segment is interrupted by the polyhedron.

Expanding this idea to a three-dimensional space, consider the shadow volumes U_r ($r = 1, 2$) which are formed by using the convex polyhedron V and Q_r ($r = 1, 2$). The volume U_r ($r = 1, 2$) is the open space surrounded by the following two types of planes: the planes consisting of the end point Q_r and all pairs of adjacent vertices of C_r ; and the planes of all faces of the polyhedron visible from Q_r . A penumbra volume is defined as the minimum convex volume surrounding U_1 and U_2 , and an umbra volume is defined as the intersection of U_1 and U_2 .

As shown in Figure 5 and Figure 6, there are two cases concerning the relationship between each pair of shadow volumes for a linear light source. That

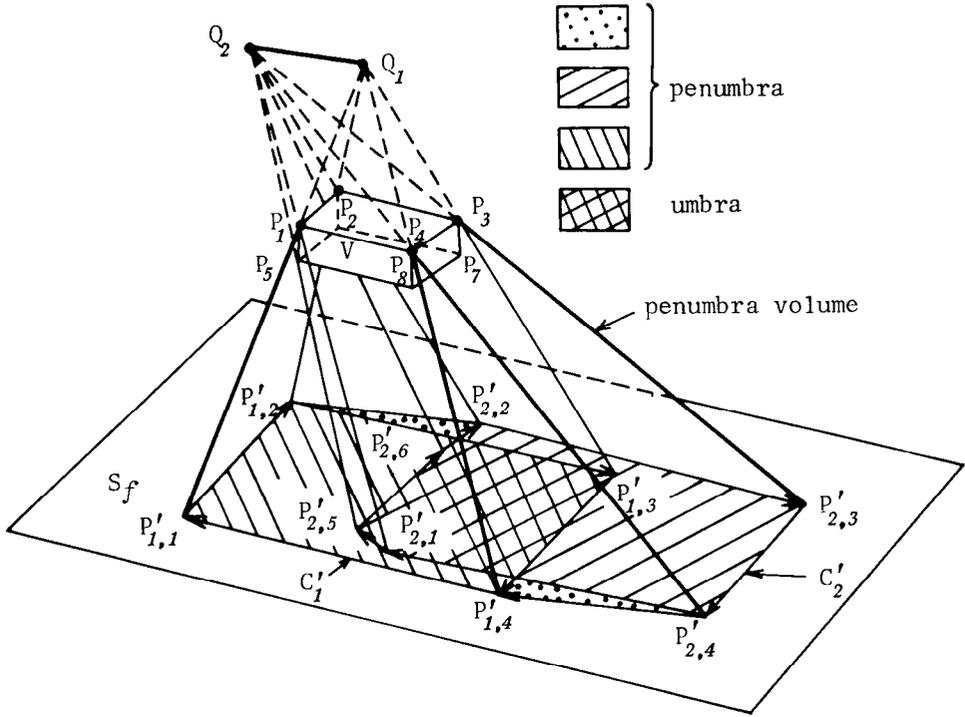


Fig. 5. Regions of penumbra and umbra (Case A for shadow volumes).

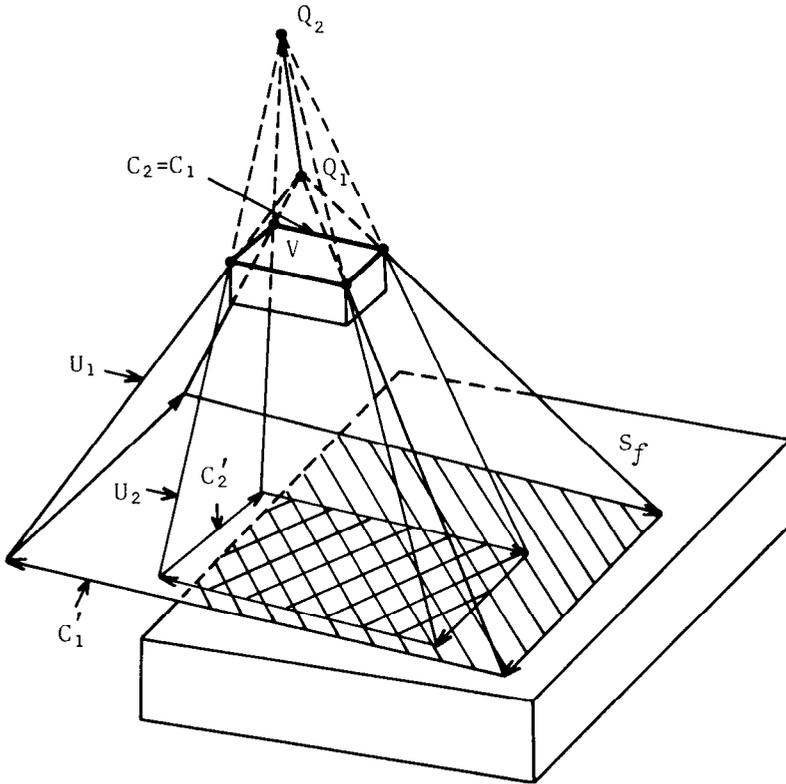


Fig. 6. Regions of shadows on face (Case B for shadow volumes).

is, U_1 and U_2 intersect each other (Case A) or one of U_ℓ ($\ell = 1, 2$) encloses the other one (Case B). The determining method of the penumbra and umbra volumes for Case A and Case B is described in Appendix 2.

Shadow detection is performed by using the relationships between the shadow volumes and each of the faces. This procedure is discussed in the following section.

4.1.2 Shadow Detection on a Face. Faces that are totally invisible from both end points of the linear source lie completely in the umbra and need not have further shadow detection performed. Faces that are visible from either end point of the linear light source are handled as described below.

The penumbra volume always includes the umbra volume. If the penumbra volume of the polyhedron V and the face S_f do not intersect, there is no shadow cast on S_f . Otherwise, the penumbra and umbra boundaries are obtained by projecting the contour lines C_ℓ ($\ell = 1, 2$) onto S_f from Q_ℓ . The penumbra and umbra boundaries are basically stored as loops on the face. We call the loops for penumbra and umbra, a penumbra loop and an umbra loop. We assume that contour lines C_ℓ ($\ell = 1, 2$), are defined in a clockwise direction when viewed from Q_ℓ . Then, the penumbra and umbra loops are determined by the following method. In Case A, let two common vertices of C_1 and C_2 be P_L and P_R (see the definition of P_L and P_R in Appendix A2; $P_L = P_4$ and $P_R = P_2$ in Figure 5). C'_1 and C'_2 are divided into two separate strings by using P_L and P_R , respectively (in Figure 5, the separated strings are $[P'_{1,4}, P'_{1,1}, P'_{1,2}]$, $[P'_{1,2}, P'_{1,3}, P'_{1,4}]$, $[P'_{2,2}, P'_{2,3}, P'_{2,4}]$, and $[P'_{2,4}, P'_{2,1}, P'_{2,5}, P'_{2,6}, P'_{2,2}]$). The penumbra loop is obtained by connecting two strings existing on the surfaces of the penumbra volume (in Figure 5, $[P'_{1,4}, P'_{1,1}, P'_{1,2}, P'_{2,2}, P'_{2,3}, P'_{2,4}]$). The umbra loop is obtained by connecting the other two strings. The umbra loop, however, twists (in Figure 5, $[P'_{1,2}, P'_{1,3}, P'_{1,4}, P'_{2,4}, P'_{2,1}, P'_{2,5}, P'_{2,6}, P'_{2,2}]$). In this case, the actual umbra area is surrounded by the loop segments in a clockwise direction. Note that the areas surrounded in a counter clockwise direction are part of the penumbra, as shown by dotted areas in Figure 5. In Case B, one of the projected contour lines is the penumbra loop, and the other one is the umbra loop. The obtained shadow loops are transformed to the coordinate system of the image space and are stored.

If some edges of the side surfaces of the penumbra volume have no intersection with the plane of the face S_f , the polyhedron V should be stored for the other process described in 4.2.2 because the penumbra loop can not be obtained.

4.2 Shading

Faces that are totally invisible from both end points of the linear source need not have further shading calculations (S_2 in Figure 7). Shading calculation is necessary for faces that are visible from either end point of the linear source (S_1 and S_3 in Figure 7). When the plane including a face intersects with the linear source, the face receives light from part of the linear source (S_3 in Figure 7). In this case, the linear source is divided into two parts with the plane. The portion of the visible side of this face is treated as a new light source because the light from the remaining portion is interrupted by the face itself.

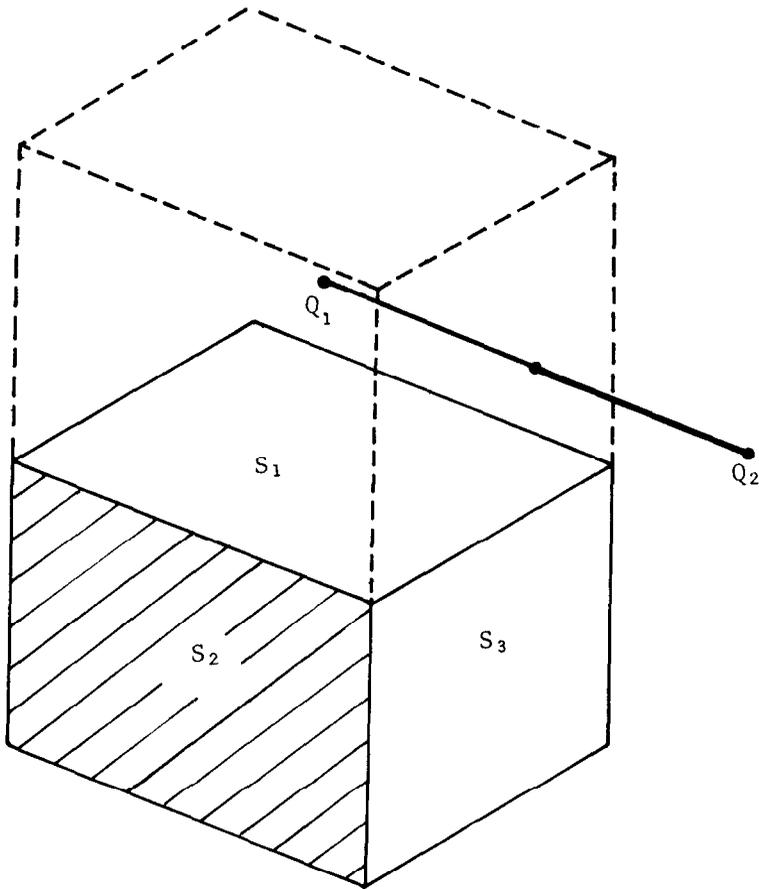


Fig. 7. Classification of faces for shading.

4.2.1 Illuminance Calculation of Unshadowed Portion. The method of calculating illuminance at an arbitrary unshadowed point P on the face S_f is described here. In Figure 8a, let L be the length of a linear source, Q an arbitrary point on the linear source, β the angle between the vector PQ and the normal N_f of S_f , θ the angle between the vector PQ and the linear source, ℓ the distance between Q_1 and Q , and r the distance between P and Q . When the luminous intensity per unit length of the source is expressed by I , the luminous intensity of the direction of angle θ is $I \sin \theta$ because of Lambertian distribution. Thus the illuminance E at P from a linear source is expressed by

$$E = I \int_0^L \frac{\sin \theta}{r^2} \cos \beta \, d\ell. \quad (5)$$

The calculation of the illuminance is divided into the following three cases for shortening the computation time. Let \mathbf{T} be the unit vector of the direction of

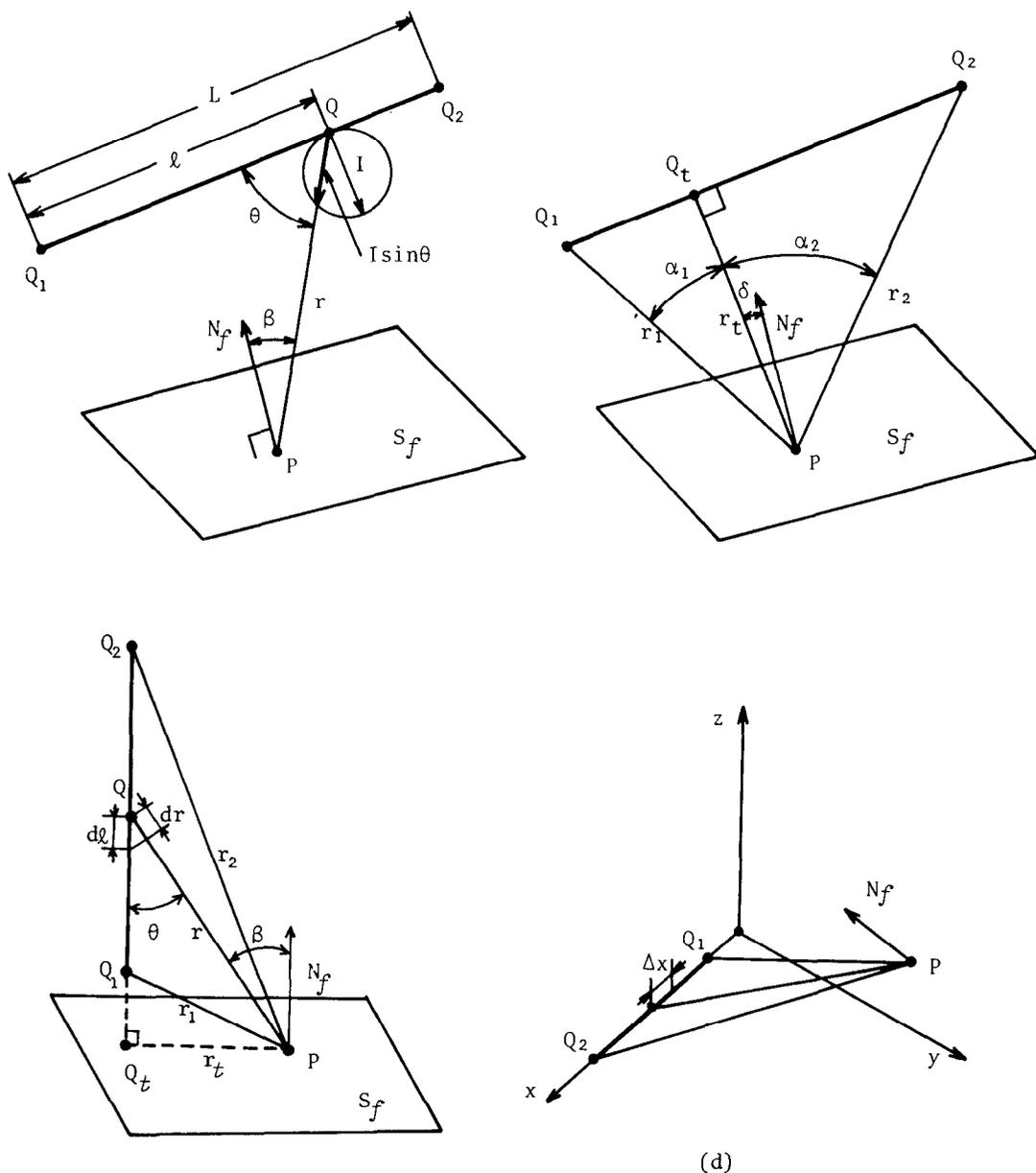


Fig. 8. Illuminance calculation for a linear light source.

linear source Q_1Q_2 , S_f the plane including P and perpendicular to τ , Q_t the intersection of S_f and the line including Q_1 and Q_2 , and r_1 , r_2 , and r , the distances PQ_1 , PQ_2 and PQ_t , respectively, as shown in Figure 8b.

Case A: The linear source is parallel to S_f . In this case θ is equal to $(\pi - \beta)$. The normal illuminance E_n (i.e., the illuminance of direction PQ_t) can be obtained by substituting $(\pi - \beta)$ for θ in eq. (5). Then the illuminance E can be obtained

as $E_n \cos \delta$ where δ is the angle between PQ_ℓ and the normal of S_ℓ . When Q_1 and Q_2 lie on the same side of S_ℓ , E is given by

$$E = (0.5I/r_\ell) |\alpha_1 + \sin \alpha_1 \cos \alpha_1 - (\alpha_2 + \sin \alpha_2 \cos \alpha_2)| \cos \delta. \quad (6)$$

If Q_1 and Q_2 exist on the opposite sides of S_ℓ , respectively, E is given by changing the sign of α_1 or α_2 in eq. (6), where α_1 and α_2 are the angles between PQ_ℓ and PQ_1 , and PQ_ℓ and PQ_2 , respectively. Let u_ℓ ($\ell = 1, 2$) be defined as $\tan \alpha_\ell$ in Figure 8b, then $u_\ell = F(S_\ell, Q_\ell)/r_\ell$. $F(S_\ell, Q_\ell)$ is obtained by the inner product of vectors, $(Q_\ell - P) \cdot \mathbf{T}$, because S_ℓ includes P and the normal of S_ℓ is \mathbf{T} . The terms $\alpha_\ell + \sin \alpha_\ell \cos \alpha_\ell$ ($\ell = 1, 2$) can be expressed by the functions of u_ℓ , and $\cos \delta$ by $F(S_\ell, Q_1)/r_\ell$. Thus, the illuminance E at P is given by

$$E = I |e_\ell(u_1) - e_\ell(u_2)| \frac{F(S_\ell, Q_1)}{r_\ell^2}, \quad (7)$$

where e_ℓ (i.e., $0.5(\alpha_\ell + \sin \alpha_\ell \cos \alpha_\ell)$) is formally defined as:

$$e_\ell(u) = 0.5 \left(\tan^{-1} u + \frac{u}{1 + u^2} \right).$$

In the numerical evaluation of $e_\ell(u)$, we want to insure some minimal error, such as, less than 1 percent. For small values of u ($|u| < 0.44$), the error can be most easily kept in bounds by using the following polynomial approximation:

$$e_\ell(u) = u(1 - 2/3 u^2 + 3/5 u^4 - 7/4 u^6).$$

Case B: The linear source is perpendicular to S_ℓ . In this case $\beta = \theta$, $\sin \theta = r_\ell/r$ and $d_\ell = \sec \theta dr$ (see Figure 8c), then eq. (5) is expressed as

$$E = I \int_{r_1}^{r_2} r_\ell/r^3 dr = I(r_1^{-2} - r_2^{-2}) \frac{r_\ell}{2} \quad (r_2 > r_1).$$

Since r_2 need not be greater than r_1 , the illuminance E is actually given as the absolute value of the above equation. That is

$$E = 0.5 I |r_1^{-2} - r_2^{-2}| r_\ell. \quad (8)$$

Case C: The cases excluding A and B. By transforming the world coordinates so that the linear source coincides with the x axis and the yz plane includes the point P , it is possible to simplify the numerical integration in eq. (5) (see Figure 8d). The computation time is reduced by giving the increment of numerical integration Δx as a function of $\min(r_1, r_2)/L$, because the farther point P is from the source, the larger Δx .

It is worth saving computing time in the first two cases (A and B), because fluorescent lamps on a ceiling are usually perpendicular or parallel to the walls and a floor.

In order to support specular reflection, it is necessary to adapt the Phong model [11] which was developed only for point sources. In our system, the

numerical integration method such as Case C can be available for the polyhedra having specular reflection characteristics because the numerical integration can be handled as a series of point sources.

4.2.2 Searching for the Shadowed Sections. On every scan line, a search for shadowed sections is made. Multiple polyhedra generally cast shadows on a face S_f , but for simplicity of explanation the shadow cast by a single polyhedron is described.

As an umbra always exists inside a penumbra, the intersection of the penumbra area and a scan line is searched for first. It is easy to test for this intersection because each penumbra loop (see Section 4.1.2) always forms a convex polygon in a clockwise direction. After the search for the penumbra, the search for the umbra area is executed. The umbra loop, however, is not always a convex polygon and the umbra area consists of only those areas surrounded in a clockwise direction by the umbra loop as mentioned in Section 4.1.2. As shown in Figure 9a, the umbra is a region surrounded by the edges of the umbra loop (except the edges common with the penumbra loop). Let the intersection points of the penumbra loop and the scan line be $P_{p,l}$ and $P_{p,r}$. Because the umbra area is always convex, the umbra on the scan line is obtained by cutting the segment $P_{p,l}P_{p,r}$ with the edges surrounding the umbra and crossing the scan line. In Figure 9a, the umbra on the scan line, the segment $P_{u,l}P_{u,r}$, is determined by clipping the segment $P_{p,l}P_{p,r}$ with the edges, P_1P_2 , P_2P_3 , P_3P_4 , P_5P_6 , and P_6P_7 .

On the other hand, if there are some edges of side surfaces of a penumbra volume having no intersection with the plane of the face S_f , the shadowed areas have not been obtained as mentioned in Section 4.1.2. In this case, the following process in object space is required. First, the segment which is the common part of the face S_f and the scan plane formed by the viewpoint as well as a scan line (in Figure 9b, segment $P_{f,l}P_{f,r}$) is obtained. The second step is executed by expanding the algorithm used in the image space mentioned above. That is, the polygons of the penumbra volume or umbra volume are used instead of the edges of the penumbra loop or the umbra loop in image space, respectively. The penumbra and umbra on the scan plane are obtained by clipping the segment $P_{f,l}P_{f,r}$ with those polygons. For simplicity, Figure 9b shows only a penumbra on the scan plane (the penumbra segment $P_{p,l}P_{p,r}$). Finally the segments of the penumbra and umbra are projected onto the perspective plane.

4.2.3 Illuminance Calculation in Penumbra Areas. A penumbra area is the region in which the light from the source is partially interrupted by several polyhedra. Therefore, the illuminance at an arbitrary point P in the penumbra is calculated by obtaining the visible parts of the linear source from P . These visible parts act as the new sources to P .

Quantitative invisibility [1], used for hidden line elimination, is available for determining the visible segments of the linear source, where the quantitative invisibility of a point on the linear source is defined as the number of front faces that lie between the point and the viewpoint (the calculation point P is assumed as the viewpoint). The segments whose quantitative invisibility equals zero are visible. In order to obtain the quantitative invisibility, all convex polyhedra casting shadows on point P are searched, and the contour lines of these polyhedra observed from P are obtained. When these contour lines and the linear source

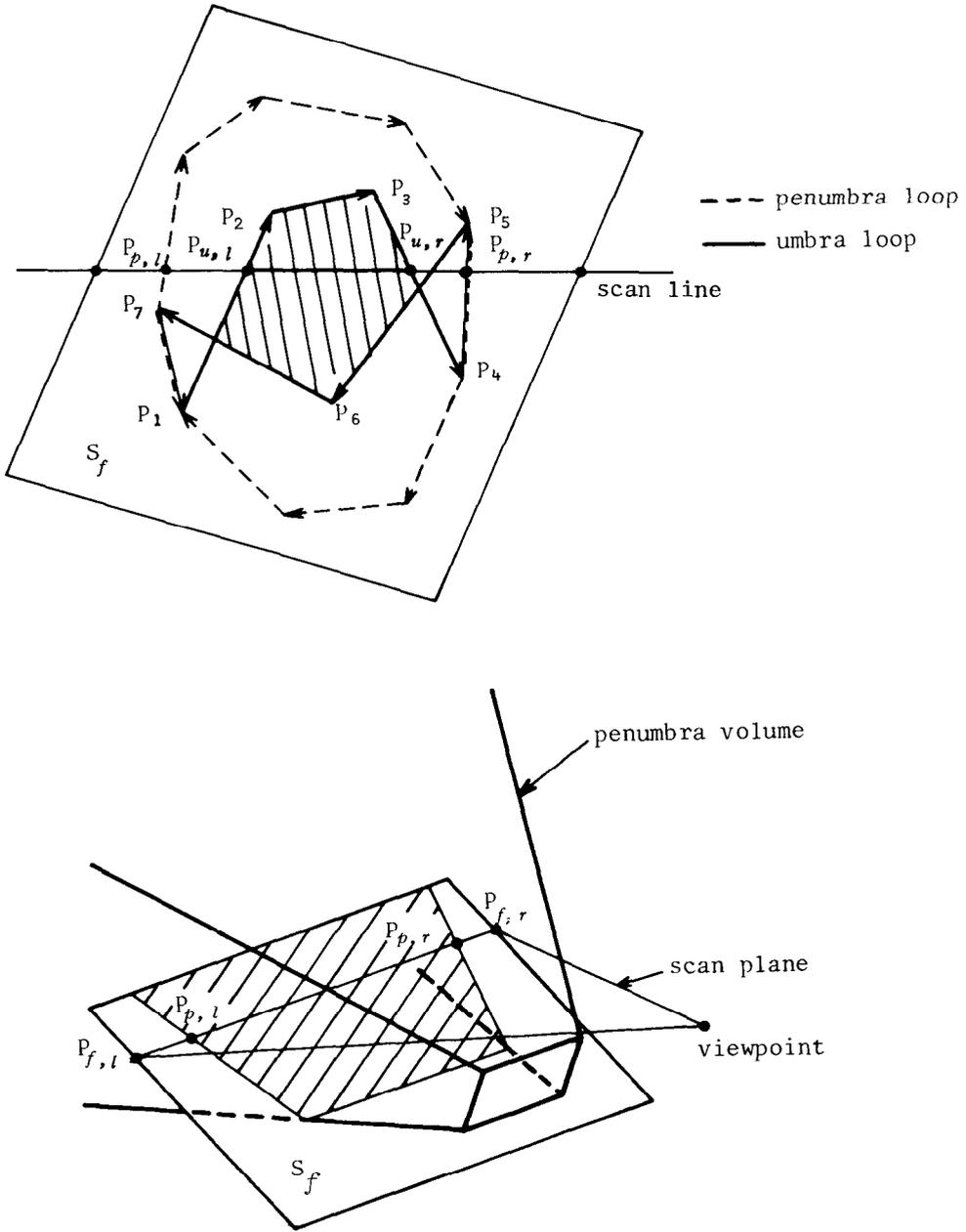


Fig. 9. Shadow on a scan line. (a) Shadow detection on image space. (b) Shadow detection on object space.

are viewed from P , the intersections between them are calculated because the quantitative invisibility on the linear source changes only at these intersections; the calculation is executed in the object space by using the linear source and the pyramids consisting of the contour lines and P .

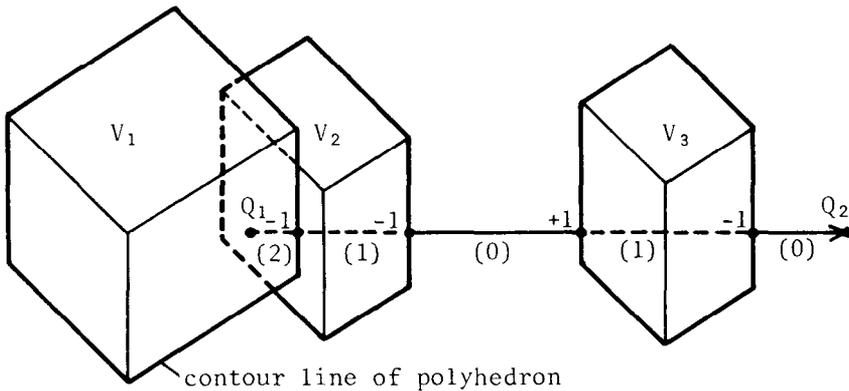


Fig. 10. A linear source viewed from calculation point for shading. (linear source Q_1Q_2 is partially hidden by polyhedra V_1 , V_2 and V_3).

Figure 10 shows a linear source viewed from the calculation point P (in Figure 10, each number in the parenthesis shows the quantitative invisibility).

5. DISPLAY OF ISOLUX CONTOURS

The halftone representations presented in this paper are realistic, but they do not depict the value of illuminance. This problem can be solved by displaying isolux contours on the shaded image. Superimposing the color spectrum on a shaded image is effective for grasping the numerical results. This method, as first pointed out by Christiansen [5], has been applied to display pressure, temperature, and so on.

A display method which superimposes the color belts of isolux contours on a shaded image is used here. These color belts and the perspective image are alternated, allowing observers to see the illuminance values easily and the perspective image in the same picture. The color of the isolux belt varies in proportion to the illuminance value. In this paper, the maximum value is assigned to red.

The representation proposed here offers a useful tool to illumination engineers and architects, even though the consideration of indirect illuminance components is neglected.

6. EXAMPLES

The examples for point sources with luminous intensity curves are shown in Figure 11 (a)–(f); (b), (d), and (f) depict the illuminance distributions of (a), (c) and (e), respectively. Pictures (a) and (b), are for interior lighting. The light's position coincides with the origin of the luminous intensity curve. Pictures (c) and (d), show outdoor lighting; the color of the lamps located at the front door, the side gate, and the street lamps are simulated by a fluorescent lamp, an incandescent lamp, and mercury arc lamps, respectively. The windows' lights are only simulated as colored windows.

Pictures (e) and (f), are depicted for comparison with the linear source, (g) and (h). The effect of penumbræ is very realistic.

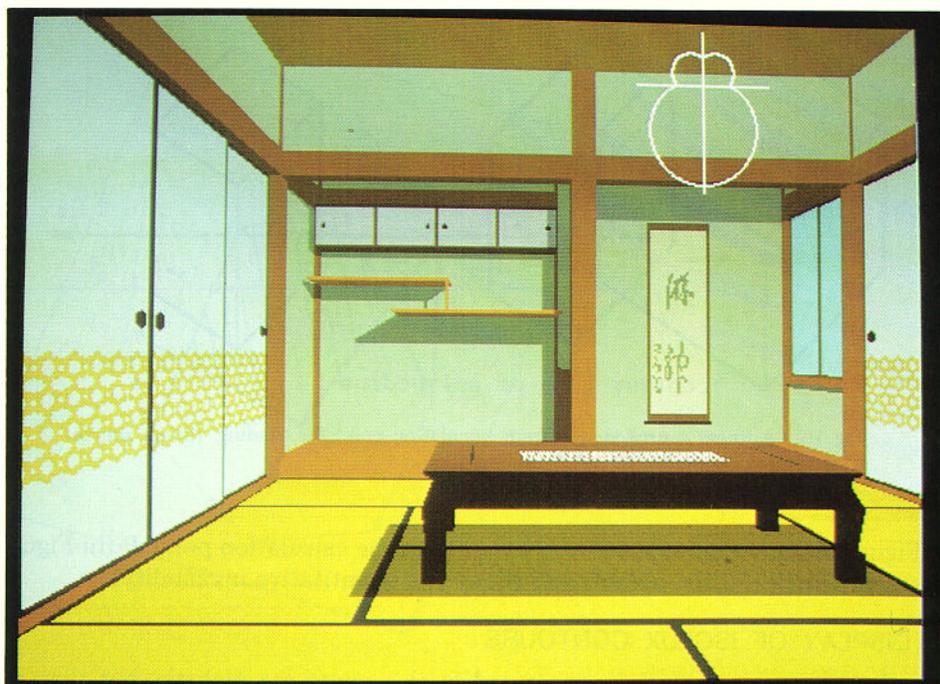


Fig. 11(a)



Fig. 11(b)

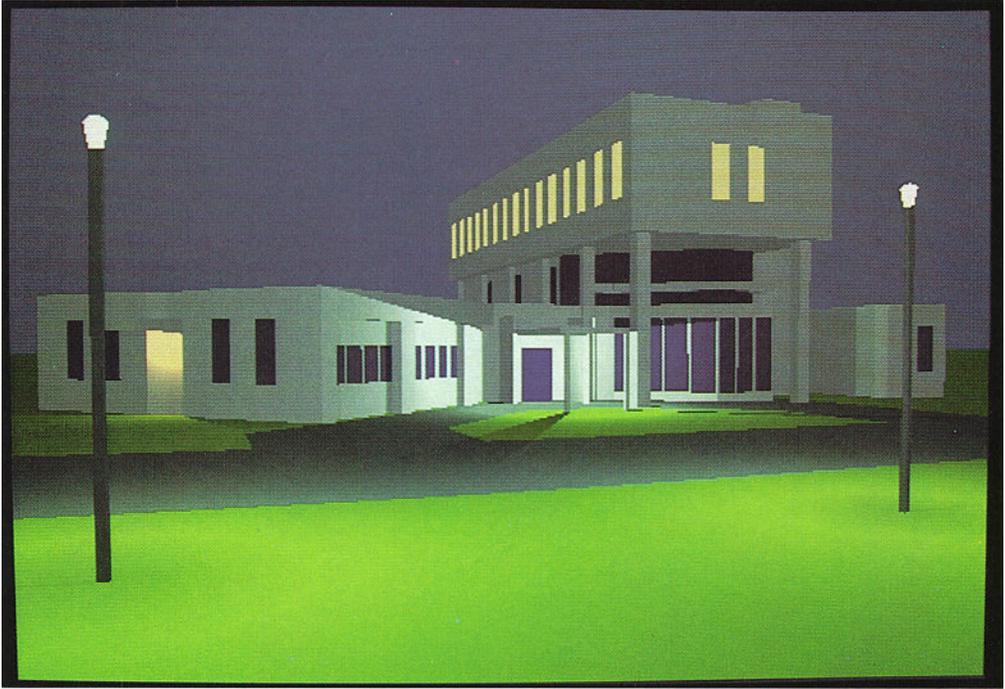


Fig. 11(c)



Fig. 11(d)



Fig. 11(e)

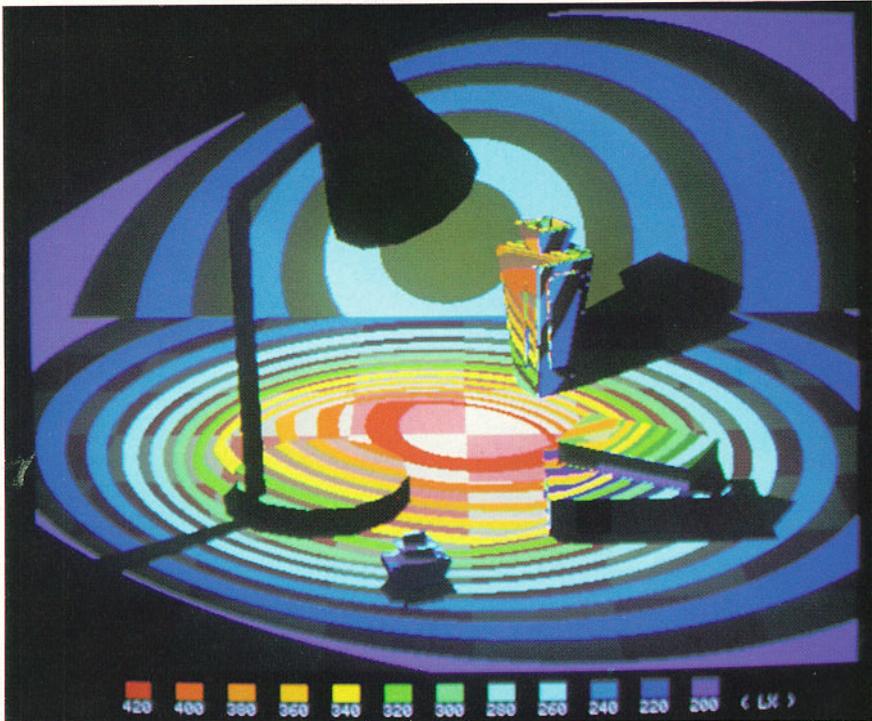


Fig. 11(f)

Fig. 11. continued

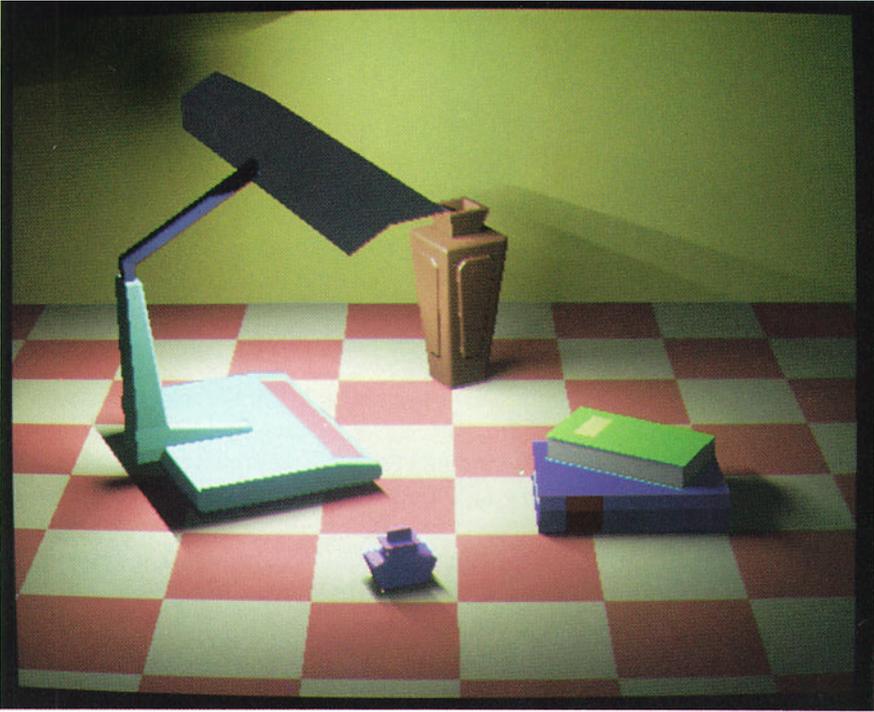


Fig. 11(g)



Fig. 11(h)

Fig. 11. Examples. (a)–(f): examples for point light sources, (g) and (h): examples for a linear light source. (b), (d), (f) and (h) depict the illuminance distributions of (a), (c), (e), and (g), respectively.

7. CONCLUSIONS

This paper has proposed two methods; one for shaded representation of three-dimensional objects illuminated by point sources with axi-symmetric luminous intensity distribution and the other for linear light sources with Lambertian luminous intensity distribution.

The following conclusions can be stated from the results obtained.

(1) The illuminance for point sources and linear sources is calculated precisely, and the reality of shaded representation is much improved. The procedure can be applied to lighting designs.

(2) The introduction of the partial sphere of lighting decreases the number of objects needed for shadow detection.

(3) Three-dimensional objects composed of a set of convex polyhedra simplify the determination of the volumes of penumbrae and umbrae needed for shadow detection.

(4) The illuminance calculation becomes simple because searching for the areas of penumbrae and umbrae on each face is done before scanning for hidden surface removal.

(5) Designers can easily grasp the global illuminance distribution by combining isolux color belts with the perspective image.

Appendix

A1. Hidden Surface Removal

The outline of the procedure is as follows:

- (i) Project the vertices of polyhedra onto the perspective plane, and extract the front faces of polyhedra.
- (ii) Calculate the priority of visibility for the overlapped polyhedra on the perspective plane. The overlap test is executed by means of the projected contour lines of convex polyhedra and the depth at the intersecting point in each contour is used to determine the priority.
- (iii) Scan the perspective plane from top to bottom. Hidden surface removal on each scan line is executed by using a technique similar to the painter's algorithm [8].

A2. Determining Penumbra and Umbra Volumes

Let's assume that the vertices P_{ℓ} ($\ell = 1, 2, \dots, n$) of a convex polyhedron V (n is the number of vertices of V), the contour lines $C_{\ell}[P_{\ell,1}, \dots, P_{\ell,i}, P_{\ell,i+1} \dots P_{\ell,n_{\ell}}]$ viewed from the end points of a linear source Q_{ℓ} ($\ell = 1, 2$), where n_{ℓ} is the number of the vertices of the contour line, and the faces $\bar{S}_{\ell,i}$ consisting of $[Q_1, Q_2, P_{\ell,i}]$ where the symbol $[]$ means the ordererd strings of vertices.

(1) *Case A*: This case occurs when both a vertex $P_{\ell,i}$ satisfying the eq. (9) and a vertex $P_{\ell,j}$ satisfying the eq. (10) exist.

$$F(\bar{S}_{\ell,i}, P_{\ell,i-1}) \geq 0, \quad F(\bar{S}_{\ell,i}, P_{\ell,i+1}) \geq 0, \quad (9)$$

$$F(\bar{S}_{\ell,j}, P_{\ell,j-1}) \leq 0, \quad F(\bar{S}_{\ell,j}, P_{\ell,j+1}) \leq 0, \quad (10)$$

where ℓ is 1 or 2, and eq. (9) (or (10)) means that all vertices of V lie on the

same half-space of the plane of \bar{S}_{C_i} (or \bar{S}_{C_j}). We define the vertex satisfying the eq. (9) as P_L and the vertex satisfying the eq. (10) as P_R ; P_L and P_R must be common vertices of C_1 and C_2 (in Figure 5, C_1 is $[P_1, P_2, P_3, P_4]$, C_2 is $[P_1, P_5, P_6, P_2, P_3, P_4]$, P_L is P_4 , and P_R is P_2). However, if the line segment Q_1Q_2 and the contour edges of V exist on the same plane, there are multiple vertices satisfying eq. (9) and (10). In this case, two arbitrary vertices are selected for P_L and P_R .

Using P_L , P_R , C_1 and C_2 mentioned above, the penumbra and umbra volumes are defined as below.

The penumbra volume is the space surrounded by the following four types of planes:

(i) The planes including Q_1 , and the adjacent vertices $P_{1,i}$ and $P_{1,i+1}$ which are the elements of contour lines C_1 and are counted in a clockwise direction from P_L to P_R (in Figure 5, $[Q_1, P_4, P_1]$ and $[Q_1, P_1, P_2]$ because $P_L = P_4$ and $P_R = P_2$ as mentioned above).

(ii) The planes including Q_2 , and the adjacent vertices $P_{2,j}$ and $P_{2,j+1}$ which are the elements of C_2 and are counted in a clockwise direction from P_R to P_L (in Figure 5, $[Q_2, P_2, P_3]$ and $[Q_2, P_3, P_4]$).

(iii) Two planes including the vertices $[Q_2, Q_1, P_L]$ and $[Q_1, Q_2, P_R]$ (in Figure 5, $[Q_2, Q_1, P_4]$ and $[Q_1, Q_2, P_2]$).

(iv) The faces of V which are completely visible from the light source.

The umbra volume is surrounded by the following three types of planes;

(i) The planes including Q_1 , and the adjacent vertices $P_{1,i}$ and $P_{1,i+1}$ which are the elements of C_1 and are counted in a clockwise direction with respect to the light source from P_R to P_L (in Figure 5, $[Q_1, P_2, P_3]$ and $[Q_1, P_3, P_4]$).

(ii) The planes including Q_2 , and the adjacent vertices $P_{2,j}$ and $P_{2,j+1}$ which are the elements of C_2 and are counted in a clockwise direction from P_L to P_R (in Figure 5, $[Q_2, P_4, P_1]$, $[Q_2, P_1, P_5]$, $[Q_2, P_5, P_6]$ and $[Q_2, P_6, P_2]$).

(iii) The faces of V which are visible from either one of the end points of a linear source.

(2) *Case B*: If Q_2 exists within the pyramid formed by Q_1 and C_1 , the penumbra and umbra volumes are U_2 and U_1 , respectively. If not, the penumbra and umbra volumes are U_1 and U_2 , respectively (in Figure 6, the penumbra volume is U_1).

In Case B, the projected contour lines, C'_1 and C'_2 , never intersect each other, as shown in Figure 6.

ACKNOWLEDGMENT

We gratefully appreciate the many helpful comments from the reviewers.

REFERENCES

1. APPEL, A. The notion of quantitative invisibility and the machine rendering of solids. In *Proceedings of the ACM National Conference* (1967), 387-393.
2. ATHERTON, P., WEILER, K., AND GREENBERG, D. Polygon shadow generation. *Comput. Graph.* 12, 3 (Aug. 1978), 275-281.
3. BOUKNIGHT, W. J. AND KELLY, K. C. An algorithm for producing half-tone computer graphics presentations with shadows and movable light sources. *SJCC, AFIPS*, 36 (1970), 1-10.

4. BROTMAN, L. S., AND BALDER, N. I. Generating soft shadows with a depth buffer algorithm, *IEEE Comput. Graph. Appl.* 4, 10 (Oct. 1984), 5-24.
5. CHRISTIANSEN, H. N. Application of Continuous Tone Computer Generated Images in Structural Mechanics. *Structural Mechanics Computer Programs—Surveys, Assessments, and Availability*, University Press of Virginia, Charlottesville (1974), 1003-1015.
6. CROW, F. C. Shadow algorithms for computer graphics. *Comput. Graph.*, 11, 2 (1977), 242-247.
7. GOURAUD, H. Continuous shading of curved surfaces. *IEEE Trans. Comput. C-20*, 6 (June 1971), 623-628.
8. NEWELL, M. E., NEWELL, R. G. AND SANCHA, T. L. A new approach to the shaded picture problem. In *Proceedings of the ACM National Conference* (Boston, Mass.) Aug. 1972, 443-450.
9. NISHITA, T., AND NAKAMAE, E. An algorithm for half-toned representation of three-dimensional objects. *Inf. Proc. Japan*, 14 (1974), 93-99.
10. NISHITA, T. AND NAKAMAE, E. Half-tone representation of 3-D objects illuminated by area sources or polyhedron sources. In *Proceedings of IEEE Computer Society's International Computer Software and Applications Conference (COMPSAC83)* (Chicago, Ill., Nov. 1983), 237-242.
11. PHONG, B. T. Illumination for computer-generated pictures. *Commun. ACM*, 18, 6 (June 1975), 311-317.
12. VERBECK C. AND GREENBERG, D. A comprehensive light-source description for computer graphics. *IEEE Comput. Graph. Appl.*, 4, 7 (July 1984), 66-75.
13. WARN, D. R. Lighting controls for synthetic images. *Comput. Graph.* 17, 3 (July 1983), 13-21.
14. WHITTED, T. An improved illumination model for shaded display. *Commun. ACM*, 23, 6 (June 1980), 343-349.
15. WILLIAMS, L. Casting curved shadows on curved surfaces. *Comput. Graph.* 12, 3 (Aug. 1978), 270-274.

Received October 1982; revised August 1983 and November 1984; accepted May 1985