광선추적 수행중 그림자의 빠른 검사를 위한 효과적인 알고리즘

어 길 수°, 최 훈 규, 경 종 민
한 국 과 학 기 술 원, 전 기 및 전 자 공 학 과

A Fast Shadow Testing Algorithm during Ray Tracing.

Kil Su Eo, Hun Kyu Choi, and Chong Min Kyung
Department of Electrical Engineering, KAIST

Abstract - This paper presents a new shadow testing acceleration scheme for ray tracing called hybrid shadow testing(HST), based on conditional switching between the conventional shadow testing method and Crow's shadow volume method, with the object polygons as well as the shadow polygons registered onto the corresponding cells under the 3-D space subdivision environment. Despite the preprocessing time for the generation and registration of the shadow polygons, the total shadow testing time of the new algorithm was approximately 50 % of that of the conventional shadow testing method for several examples while the total ray tracing time was typically reduced by 30 % from the conventional approach.

I. Introduction

While the principle of ray tracing has been widely used for realistic image synthesis in 3-dimensional computer graphics using the global illumination model instead of the local one used in the scan line algorithm, ray tracing still suffers from extremely long computation time which is mostly taken up by the shadow testings and ray/object intersection calculations.[1] Until now, more efforts have been made toward reducing the ray/object intersection calculation time than the shadow testing time. However, accelerating the shadow testing procedure is essential in reducing the overall ray tracing time, especially when there are multiple light sources, since it is not uncommon that the shadow testing time could account for more than 50 percent of the total rendering time in conventional ray tracing. The shadow testing in ray tracing corresponds to determining the visibility of a reflection point from the light source, that is, if any opaque object lies between the reflection point and the light source, the point is not visible from the light source, i.e., it is in shadow. In conventional shadow testing scheme [1], a ray called 'shadow ray', is fired from a reflection point toward each light source to determine the visibility of that reflection point from each light source. The shadow ray is advanced forward until it meets either a blocking opaque object (the point is in shadow) or a light source (the point is not in shadow). Although the program implementation is very easy since it is basically identical to the ray/object intersection calculation, the shadow testing time is very long because of the many intersection calculation steps involved.

Haines et al. [2] proposed another shadow testing acceleration scheme called 'light buffer' which is a cube surrounding each light source consisting of six (front, back, top, bottom, right, left) screens for each facet and is constructed in a preprocessing step. Each pixel of the buffer contains an ordered list of all objects which are intersected with the cone composed of the light source and the pixel rectangle. The corresponding pixel of the shadow buffer is referenced for every shadow testing during ray tracing.

Yet another shadow calculation scheme proposed by Crow is based on the concept of shadow volumes in the context of the scanline hidden surface algorithm.[3] Shadow volumes are composed of shadow polygons which are defined by projections of silhouette edges from the light sources. When the ray pierces a shadow polygon in ray tracing, the visibility from the light source may change. A number called, 'depth count' denoting the current shadow depth is updated for each intersection of the ray with shadow polygons. When the ray pierces a front-facing shadow polygon, the depth count is incremented, while it is decremented when the ray intersects a back-facing shadow polygon. (Shadow volume concept was further generalized by Bergeron.[4] If there are no shadow polygons in the trace path of a ray, the depth count doesn't change thereby leaving the end point of the trace path in the same shadow condition as the starting point at no additional cost. This is due to the shadow's spatial coherency which needs to be exploited in ray tracing to reduce the shadow testing time.

The so-called hybrid shadow testing (HST) which is a new shadow testing algorithm proposed in this paper is based on the conditional switching between shadow volume method and conventional shadow testing method under the uniform space subdivision environment where the shadow polygons as well as the object polygons are registered onto the relevant 3-D space subdivision cells.[5]

II. Cost Comparison between Shadow Volume Scheme and Conventional Scheme

In this section, we derive an analytic expression for the shadow testing time in the conventional shadow testing method and the shadow volume method. We assumed that the whole object space was divided into 3-D space uniform subdivision cells, and each primitive cell contains within it a list of all polygons partially or fully overlapping that primitive cell.[5]

A ray segment is defined as a line segment between two successive reflection points or between eye position and the first reflection point, which were shown as P_1 and P_2 in Fig.1, respectively. The ray visits each cell pierced by the ray segment one by one from P_1 to P_2. In our implementation of the ray tracing under uniform space subdivision, each ray segment is provided with its bucket for storing a polygon list. Even though some object polygons registered in a cell are not intersected by the ray or have an intersection point outside the cell, these polygons are still stored in the bucket such that they are not further checked for the intersection with the same ray segment at the following cells. This bucket is also used for avoiding the erroneous detection of the intersection of the ray with the polygon comprising the firing point of the same ray, which could occur due to the finite precision in representing the floating point numbers. Similar procedure was used for checking the intersection of the ray with the shadow polygon in our implementation of the ray tracer based on the shadow volume method.

Before we describe our proposed shadow testing algorithm called HST (Hybrid Shadow Testing) in the next section, a comparison of the conventional scheme and purely shadow volume scheme in terms of the CPU time requirement for shadow calculation is helpful and elaborated below. We first consider the shadow testing cost of the shadow volume method. The shadow testing cost of P_2 in Fig.1 using the shadow volume method is denoted as C_{sv}, which can be expressed as eq.(1);

$$C_{sv} = C_s N_s \qquad \text{eq.(1)}$$

where C_s is the cost of calculating the intersection point of a ray with a shadow polygon, and N_s is the total number of shadow polygons registered onto the cells which were pierced by the ray segment.

On the other hand, the shadow testing cost of the conventional shadow testing scheme, which is denoted as C_{cv}, can be represented as sum of the cost incurred by the advancement of the shadow ray and that incurred by the ray/object intersection calculations between the shadow ray and each object registered in the cells pierced by the shadow ray. The former cost is denoted as C_{cv1}, and the latter as C_{cv2}. Let's assume that object polygons in the scene are uniformly distributed. Then C_{cv1} can be expressed as the following equation;

$$C_{cv1} = C_i + C_c \frac{1-(1-p)^{\alpha N_c}}{1-(1-p)^{\alpha}} \qquad \text{eq.(2)}$$

where the meanings of C_i, C_c and N_c are as follows;

- C_i : the cost for initializing ray parameters for 3DDDA (3-Dimensional Differential Digital Analyzer) [5] such as direction vector, distance counters, and current cell index, etc,
- C_c : the cost incurred by the advancement of ray to the next cell in 3DDDA, that is, the cost incurred in updating the ray parameters to determine the next cell to be visited,
- N_c : the number of cells pierced by the shadow ray segment whose end points are the reflection point and the light source position,
- p : the average probability that a ray intersects an object polygon given that the cell onto which the object polygon is registered is pierced by the ray, and
- α : the average number of object polygons registered within a cell which must be tested for intersections with the ray visiting the cell (The polygons already tested for intersection with the current ray segment in the earlier cell are not further tested at the current cell due to the already-mentioned bucketing scheme).

On the other hand, C_{cv2}, which denotes the ray/object intersection calculation cost, can be expressed in a similar way.

$$C_{cv2} = C_o \frac{1-(1-p)^{N_o}}{p} \qquad \text{eq.(3)}$$

where C_o is the cost for the calculation of a ray/object intersection between shadow ray and object polygon, while N_o is the total number of object polygons registered in all the cells pierced by the shadow ray segment.

Hence, C_{cv} becomes

$$\begin{aligned} C_{cv} &= C_{cv1} + C_{cv2} \\ &= C_i + C_c \frac{1-(1-p)^{\alpha N_c}}{1-(1-p)^{\alpha}} + C_o \frac{1-(1-p)^{N_o}}{p} \end{aligned} \qquad \text{eq.(4)}$$

According to the definition of α,

$$N_o = \alpha N_c \qquad \text{eq.(5)}$$

Hence, eq.(4) becomes

$$C_{cv} = C_i + \{1-(1-p)^{N_o}\}\left\{\frac{C_c}{1-(1-p)^{\alpha}} + \frac{C_o}{p}\right\} \qquad \text{eq.(6)}$$

We used empirical approaches for deriving p and α as functions of various statistical parameters such as the number, average peripheral length and area of the object polygons, and the number of 3-D space subdivision cells are described. The cost parameters such as C_s, C_i, C_c, and C_o in eq.(1) and eq.(6) can be measured because they are dependent only upon the procedures and machines used.

III. Hybrid Shadow Testing (HST) Scheme

The shadow testing scheme using shadow volume concept becomes less attractive as the number of shadow polygons intersecting the ray is increased, due to the computational overhead of ray/shadow polygon intersection. In the other extreme case when the number of shadow polygons intersecting the ray is zero, shadow volume scheme is expected to be much faster than the conventional scheme since there's no need to fire a shadow ray at all. This observation made us consider a **hybrid shadow testing scheme** (HST) as a compromise between the conventional and shadow volume methods. In the proposed HST scheme, the value of N_s becomes a guide for determining the shadow testing scheme on the reflection point between two methods. When the total number of shadow polygons registered onto the cells pierced by the current ray segment (N_s) is less than a certain switching threshold denoted as N_{th}, the shadow volume scheme should be more effective than the conventional scheme for shadow testing. Equating eq.(1) and eq.(6) gives

$$N_{th} = \frac{1}{C_s}\left\{C_i + (1-(1-p)^{N_o})\left(\frac{C_c}{1-(1-p)^{\alpha}} + \frac{C_o}{p}\right)\right\} \qquad \text{eq.(7)}$$

Before resorting to actual shadow testing using conventional or shadow volume method, overall shadow testing time can be significantly reduced by utilizing the so-called ray segment shadow coherence. That is, if the the shadow bucket contains no shadow polygon (N_s=0), shadow condition doesn't change and the previous shadow condition is simply sustained. The first condition to be checked is self-shadow, which filters out all the polygons facing away from the light source. It can be simply tested by dot-producting the light vector (L) with the surface normal (N) on the reflection point. Although the shadow testing , i.e., checking whether $L \cdot N > 0$ or $L \cdot N < 0$, is very simple, it has a drawback that the shadow depth count can't be maintained, which makes it quite difficult to revert to the shadow volume method later. The pseudo code of the HST scheme is as follows;

```
For each light source
   if( N_s equals 0) /* Shadow bucket is empty. */
         (Null action);
   else if(L·N < 0) {
         shadow_condition = 1;
           /* Shadow_condition of 1 or 0 denotes
              that the reflection point is in shadow
              or not, respectively. */
         shadow_depth_flag = 0;
           /* Shadow_depth_flag is 1 when the shadow
              depth count is valid and 0 otherwise. */
   }
   else if( N_s <= N_th and (shadow_condition equals 0 or
         shadow_depth_flag equals 1)) {
         shadow_condition = shadow_volume_test();
         shadow_depth_flag = 1;
   }
   else   {
         shadow_condition = conventional_shadow_test();
         shadow_depth_flag = 0;
   }
```

IV. Experimental Results and Discussions

Three different schemes for shadow testing, i.e., shadow volume scheme, conventional scheme, and the proposed HST scheme were implemented on SUN 3/75 using C language. Fujimoto's uniform space subdivision scheme [5] was adopted which requires a preprocessing step for registering the object polygons and the shadow polygons onto the relevant cells. The image data files generated on the SUN system were transferred to IBM-PC/AT where a graphic system with 8-bit pixel depth was installed.

In our implementation of the HST scheme, N_{th} is determined only once in the preprocessing step and used over the whole shadow testings to avoid the evaluation of N_{th} at every reflection point. The measured values of C_s, C_i, C_c, and C_o used in eq.(1) through eq.(7) under the SUN 3/75 environment are shown in Table 1. Although our experiments were done for four example images which are "Glasses", "Desk", "Star", and "Cards", the image and data for "Desk" are shown becuase of the page limitation. Fig.(2) showes the image of "Desk". Fig.3 showes the measured CPU times for shadow testing using the three shadow testing schemes, i.e., conventional, shadow volume and the hybrid shadow testing (HST) methods for "Desk", where the values of N_{th} for HST scheme are externally supplied and varied. The predicted optimal values for N_{th} as calculated by eq.(7) and denoted by $N^o_{th,c}$ are in good agreement with their measured values denoted as $N^o_{th,m}$ for all examples. Table 2 present the CPU times consumed in preprocessing, shadow testing and other procedures of the three shadow testing schemes for "Desk". "Preprocessing" includes the file input, flattening the object hierarchy, object registrations, and viewing transformations. In the shadow volume and HST schemes, "Preprocessing" additionally includes the extraction of silhouette edges and registrations of the shadow polygons. "Others" includes jobs required for finding the reflection points, illumination calculations, and the file output, etc. Total shadow testing times consumed by the HST scheme are reduced by factors of 2.5, 2.2, 1.5, and 1.8, compared to the conventional scheme for the "Glasses", "Desk", "Stars", and "Cards" examples, respectively.

V. Conclusion

A new shadow testing algorithm during ray tracing, called **Hybrid Shadow Testing (HST)** is proposed and implemented under the 3-D space subdivision environment. The HST algorithm exploits various subschemes for fast determination of the shadow condition, i.e., 'sustain' for exploiting the ray segment shadow coherence, 'self-shadow' for fast screening of the back-facing polygon, 'shadow volume' and conventional shadow testing methods. An empirical formula for determining the optimal switching threshold (N_{th}) was proposed using analytically derived statistical parameters such as p and α. The experimentally obtained optimal values for N_{th} were generally in excellent agreement with those predicted from the analytic derivation, for various example images. The shadow testing time, itself, was reduced by approximately a factor of 2 from the conventional shadow testing schemes, while the total ray tracing time was reduced by about 30% for the four example images. Furthermore, by utilizing the pixel shadow coherence, the CPU times for shadow testing were shortened by additional 20-40%.

References

[1]. Whitted, T 'An Improved Illumination Model for Shaded Display' *Comm. of ACM*, Vol.23, No. 6, (June 1980) pp. 343-349

[2]. Haines, E A and Greenberg, D 'The Light Buffer : A Shadow-Testing Accelerator' IEEE CG&A Vol.6 No.9 (September 1986) pp. 6-16

[3]. Crow, F 'Shadow Algorithm for Computer Graphics' *Computer Graphics* (Proceedings SIGGRAPH 77), Vol.11, No. 2, (1977) pp. 242-248.

[4]. Bergeron, P 'A General Version of Crow's Shadow Volumes' IEEE CG&A Vol.6 No.9 (September 1986) pp. 17-28

[5]. Fujimoto, A, Tanaka, T, and Iwata, K 'ARTS: Accelerated Ray-Tracing System' IEEE CG&A Vol.6 No.4 (April 1986) pp. 16-26

Table 1 Experimental values (in msec) for the machine-dependent parameters, C_s, C_i, C_c and C_o

Items	C_s	C_i	C_c	C_o
Time	2.00	2.54	0.23	1.88

Table 2 CPU times (in minutes) comparison among three shadow testing methods; conventional, shadow volume and HST schemes for "Desk".

Algorithms	Preprocessings	Shadow testing	Others	Total
Conventional	0.42	170.83	158.00	329.25
Shadow volume	1.42	174.37	158.00	333.79
HST(N_{th}=4)	1.42	79.52	158.00	238.94

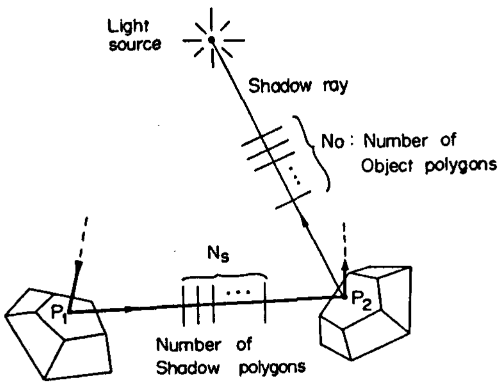

Fig.1 A schematic illustration of the two important numbers, N_s denoting the total number of shadow polygons registered onto the cells pierced by the ray segment between P_1 and P_2, and N_o denoting the total number of object polygons registered onto the cells pierced by the shadow ray shot from P_2 toward the light source.

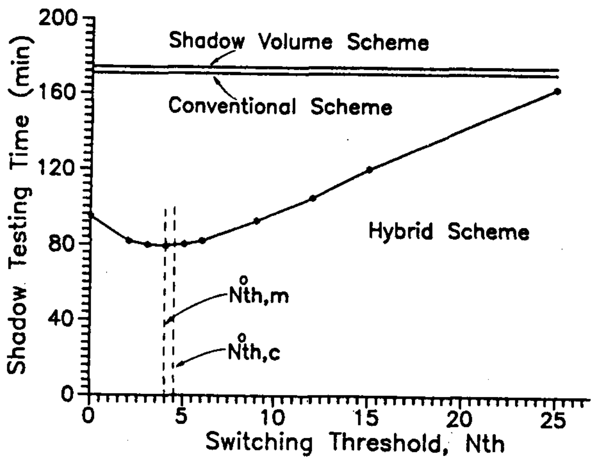

Fig.3 Shadow testing times by the HST scheme for various switching threshold values(N_{th}), compared with those of the conventional and shadow volume approaches for "Desk". $N^o_{th,c}$ denotes the calculated optimal value of N_{th}, and $N^o_{th,m}$ denotes the measured optimal value of N_{th}.

Fig.2 Image 'Desk' (three light sources)