Sub-Geometry based Shadow Mapping

Hua Li, Huamin Yang, Xin Feng *, Zhanfang Chen, Jianping Zhao

Department of Computer Science and Technology
Changchun University of Science and Technology
Changchun, Jilin, China

Abstract — In this paper we propose a Sub-Geometry based Shadow Mapping (SGSM) algorithm for real-time sub-pixel accurate hard shadows rendering. The method focuses on addressing the aliasing artifacts due to the finite resolution of shadow maps. Inspired by the fact that geometry-based shadow mapping reconstructs the depth value of an occluding point corresponding to the triangle stored in the texel, in SGSM, a surface normal detection strategy is presented to recognize the overlapped triangles in one texel. Without increasing the overhead of system memory, SGSM can improve the accuracy of depth reconstruction. Experiments show that SGSM algorithm is an effective shadow map based method that eliminates the resolution issues and reduces the jagged shadow boundaries greatly.

Keywords - Rendering; geometry-based shadow maps; hard shadows; sub-pixel; anti-aliasing

I. INTRODUCTION

Shadows are an essential element in computer-generated scenes. Interactive rendering algorithms of hard shadows could roughly be divided into three categories: geometry-based techniques, image-based techniques, and ray-tracing [1]. Geometry-based techniques mostly concentrate on computing a boundary representation of the region occluded from a light source like shadow volume [2] algorithm. In general, shadow volume and its variants are accurate but they can’t be scaled well to complex scenes due to their large fill rate requirements. Ray Tracing [3] methods are greatly related to the complexity of the scene, when carry on the light and the triangle intersection. Shadow mapping [4] is a two-pass image-based method, which is mostly used in shadowing techniques in real-time scenarios currently with its efficiency and flexibility. Shadow mapping render the scene from the light’s point of view in the first pass to generate a z-buffer. In the second pass, it render the scene from eye’s view, each pixel is transformed into light view space, comparing its z value with the corresponding depth value stored in the shadow map to determine whether a view sample is shadowed or illuminated. Shadow mapping construction is reasonably fast and does not depend on the model’s geometric complexity. There are several surveys [5-8] in detail on shadow mapping technique.

On the other hand, shadow mapping suffers from discretization artifacts due to limited resolution. To avoid this, some efforts have been undertaken to overcome the aliasing artifacts. Perspective Shadow Map (PSM) [9] and Light Space Perspective Shadow Maps (LiSPSMs) [10] utilizing warped projection schemes to improve the virtual shadow map’s resolution according to the viewer’s perspective. Adaptive Shadow Map (ASM) [11] reduces aliasing by storing the shadow map as a hierarchical grid structure, it is refined to create higher resolution where contains shadow boundaries from the user’s viewpoint. Parallel Split Shadow Map (PPSM) [12] focuses on split the view frustum in smaller parts. Alias-free Shadow Map [13] utilizing more advanced GPU features gives a different view to generate shadow maps which transform visible pixels from screen space to light view which completely avoid jagged shadow boundaries, but it is inefficient and not support by current hardware efficiently. Fitted Virtual Shadow Map [14] samples the scene from the eye-point of view firstly to get the needed shadow map resolution in different parts of the scene, then traverse in kd-tree fashion to shadow the scene with shadow map tiles where needed. More approaches as Shadow Silhouette Maps [15] stores the point’s location on geometric silhouette map which reduces jagged shadow boundaries. Adaptive Depth Bias for Shadow Maps [16] proposed an adaptive depth bias method to estimate a tight bias bound for each fragment and provides minimal depth bias to eliminate false self-shadowing. Closest to our work, the Reconstructable Geometry Shadow Maps (RGSM) [17] proposes a geometry-based shadow map that stores triangle information within the shadow map, which helps to find the occluding triangle for shadow test. Sub-Pixel Shadow Mapping [18] proposes to store a fixed-size partial representation of the scene geometry using conservation rasterization to reduce aliasing and bias issues. But in these two methods, unless the triangle fully covers one texel, the storage of partial triangle will underestimate the depth for intersecting surface and result in false shadow and erroneous estimation. In this paper, we introduce a Sub-geometry based Shadow Map (SGSM), which can detect the overlapped triangles in one texel and reconstruct the depth faithfully with the exact triangle for a view sample. Our algorithm extracts and stores the triangles information visible to the light source from the light’s view firstly combining with the surface normal. As most hardware rasterizers generate fragments only if the center of the fragment is covered by the considered triangle, we use conservative rasterization [19] for all triangles those touched the texel. A normal detection algorithm is proposed to compare the four corners’ normal for sub-pixel exact shadow testing which
can faithfully represent the scene geometry. Our approach simply and effectively focuses on the related triangles to a view sample. Our contributions are:

To enforce a method for identifying overlapped triangles in one texel.

To develop a more precise hard shadows method with sub-pixel anti-aliasing.

II. SUB-GEOMETRY BASED SHADOW MAPPING

A. Overview

In the spirit of [17], we build and store a partial, approximate representation of the scene geometry within a Sub-geometry based shadow map (SGSM). Firstly, we enforce the generation of fragments using conservative rasterization for all the coverage triangles. We render the scene from light’s view and store geometric and pixel related information of the closest triangles. Secondly, a surface normal detection algorithm with threads and kernels is proposed to generate a SGSM using a screen-space buffer, in which the four corners’ normal are captured. We compare the four them to recognize overlapped triangles in one texel. At rendering step, our approach projects each view-sample into light space and estimate depth with the exact triangle plane. Standard rasterization may not capture the pixel centers for rasterizing thin geometry structures due to the triangles not covering them. Hasselgren et al. introduced conservative rasterization in 2005 [19] by an algorithm of slights forward motion of triangle edges in their normal direction. Conservative rasterization allows rasterization to generate fragments for every pixel touched by a primitive, which is a new feature in modern hardware of graphic. Fig. (1) (a) shows the normal rasterization, where fragments are not produced if the triangles’ coverage do not include pixel centers. Fig. (1) (b) shows the conservation rasterization, where all the fragments are produced as long as the pixel is being touched by triangles. Conservative rasterization is necessary as the precision is crucial for silhouette triangles and sub-pixel depth determination.

![Figure (1). (a) Standard rasterization, (b) Conservative rasterization.](image)

Our algorithm has the consideration to recognize different triangles in one texel due to the fact that in most scenes several triangles are rasterized at one given texel position and only the closest one was stored in [17][18]. A view-sample might be incorrectly classified when the projection is actually outside the triangle as shown in Fig. (2), where Texel1 is covered by triangle T2, T6, T7 and T8. But only T7 information is stored in this texel. When a view-sample \( \mathbf{P} \) in camera space is projected into the light space as \( \mathbf{P'} \) in Texel1, the depth would be reconstructed by triangle plane of T7 although the exact plan should be T6. Therefore, unless a triangle fully covers one texel, a partial triangle will underestimate the depth of the intersecting surface and results in false shadow and erroneous estimation.

B. Normal detection algorithm

Several of the view-samples falling in the same pixel are the source of aliasing in standard shadow mapping. Sub-pixel anti-aliasing requires reconstructing each view-sample faithfully to the depth where the point falls in. That means more detailed information about triangle’s distribution and connection in one texel is needed. But it would lead to intractable memory and computational costs to store the list of all overlapping triangles. Our normal detection algorithm identifies overlapped triangles by comparing the normal of the four corners of one texel. Using the property of continuity of the triangles, we do not need to store vertices information of additional triangles and that will not increase the computational complexity obviously. Thank works of Ville Timonen and his Line-Sweep Ambient Obscurance [20], we traverse through the geometric shadow map along lines in parallel.

Our approach scans the geometry-based shadow map in a dense set of parallel lines and incrementally tracks the triangle profiles along those lines. The parallel lines are spaced one texel width apart along the geometry-based shadow map and the lines are traversed one texel width step at a time. In a threaded implementation each thread processes one line through a geometry-based shadow map for extracting the vertices information and compare the normal at four corners by coordinating. As the consistency of triangle on surface of objects, all related triangles vertices information has been stored in adjacent texels, we detect all related triangles reading from SGSM. According to
algorithm1, a normal detection algorithm is enforced. Fig. (3) shows normal detecting in geometry-based shadow map by line-sweep method, where Texel1,ij is the central texel and Na, Nb, Nc, Nd are the related normal vectors.

![Figure (3). Line sweep of geometry-based shadow map.](image)

In algorithm1, our method sweep each of the texelline and compare the four normal of Na, Nb, Nc and Nd. If a view sample is projected into a fully covered texel, we reconstruct the depth by comparing with the light space depth: if reconstructed depth is greater or smaller than its depth, then the sample can be classified as lit or shadowed. Otherwise, we detect all the related triangles from adjacent texel. We have assume that different predetermined for fast computing. Fig. (4) (a) shows one texel covered by two triangles of T1 and T2, and (b) shows different predetermined patterns identified by normal with different colors of dots in corners, the number on the top of first line means the total number of triangles in one texel.

![Figure (4). Predetermined pattern sets of the sub-texel](image)

### C. Depth reconstruction

We use conservative depth in [19] to store the maximum depth within a texel. A view-sample $p$ is projected to light space with the slope of x and y direction, and the depth is reconstructed according to the equation (1) [18], where $d_p$ is the depth of triangle plane derived from a view sample $p$ (px,py)

$$d_p = z + (p_x - cx) \frac{d_z}{d_x} + (p_y - cy) \frac{d_z}{d_y}$$

(1)

cx and cy are the vertices coordinates of triangles stored in SGS, $\tilde{z}$ is the depth of texel center of the exactly triangle where a view sample $p$ projected in and it should be illuminated when $p$ is on the triangle plane or $d_p \leq \tilde{d}_p$ else $p$ is shadowed, where $\tilde{d}_p$ is the depth of $p$ in light space.

### III. IMPLEMENTATION

We implemented our method using the CUDA and GLSL and G3D. At geometry and fragment shader stages, the triangle information and a line-sweep method is enforced with the corresponding functions. This basic structure of our algorithm makes it a viable alternative to a large number of shadows mapping algorithms.

#### A. Rendering quality and anti-aliasing

The size of the shadow map influences the shadow's quality and rendering speed. The capabilities of the target hardware, the position of the shadow in relationship to the camera and the required quality are the factors that affect the size of the shadow map. We compared the SGS with the standard shadow mapping with different resolution using a single shadow map as shown in Fig. (5), where the shadow map resolution is 512×512 on the left and 4096×4096 on the right using global model. The rendering quality achieved by SGS is close to ray-tracing with pixel match of 99.81% using a single 4K×4K shadow map. Fig. (6) shows several screenshots to illustrate the different shadow qualities of standard shadow maps (SM), sub-pixel shadow mapping (SPSM) and ours (SGSM) at the resolution of 2048×2048 (top) and 1024×1024 (bottom) shadow maps. Multi-sampling anti-aliasing (MSAA) is a filtering approach based on super-sampling of depth. Fig. (7) shows shadow and self-shadow effect with SM and SGS using Multi-Sampling.
Anti-Aliasing (MSAA), where the shadow map resolution is 1024×1024. Our method generates good qualities and well scaled with MSAA. We also measured the frame rate of two test scenes using standard SM, sub-pixel shadow mapping (SPSM) and our algorithm (SGSM). For the memory occupation and fill-rate consumption, from about 11.4% to 19.6% computational overhead than standard shadow map measured by the Dragon and Global model using a single 1024×1024 shadow map, about from 8% to 10.9% overhead using a single 2048×2048 shadow map, but 12.8% and 12% lower than SPSM as shown in Table1 due to the effectively normal detection algorithm in parallel measured by Dragon model both perspective and projection anti-aliasing.

IV. LIMITATIONS

Although a conservative rasterization geometry shadow map has at least one texel being occupied by a thin primitive, but some triangles can’t be found out at special cases, e.g. a single thin triangle has not covered any of the corner of a texel. Our method has limitations due to the limited resolution of the shadow map especially at the shadow border, where depth is inconsistent; also errors occur if too many overlapped triangles fall in the same area in the SGSM (usually more than four triangles, like leaves). But more than 87% overlapped triangles can be detected by our method using global model.

V. CONCLUSION

SGSM algorithm recovers accurate reconstruction of depth and improves consistent depth estimations of partially covered texels. Line sweep algorithm in parallel reduces the overhead time consuming of identifying overlapped triangles. Both of the projective and perspective aliasing are removed, which are inherent in the shadow mapping. The image quality is close to ray-tracing solutions under certain models without increasing the amount of storage of triangles based on SPSM. Our method produces high quality hard shadows in current programmable graphics hardware and scales well with other anti-aliasing strategies.

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGMENT

This work was financially supported by development project of Jilin province science and technology (20140204009GX) and major scientific and technological plan of Changchun City(14KG008).

REFERENCES

HUA LI et al: SUB-GEOMETRY BASED SHADOW MAPPING