Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.12785/amis/090325

Shadow Mapping Algorithms: Applications and Limitations

Hoshang Kolivand[1,*], *Mohd Shahrizal Sunar*[1,*], *Ayman Altameem*[2], *Amjad Rehman*[3] *and Mueen Uddin*[4]

[1] MaGIC-X (Media and Games Innovation Centre of Excellence), UTM-IRDA Digital Media Centre, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia
[2] College of Applied Studies and Community Services, King Saud University, Riyadh, KSA
[3] MIS Department College of Business Administration Salman bin Abdul Aziz University Alkharj, KSA
[4] Kulliah of Information and Communication Technology International Islamic University, Malaysia
[5] Department of ICT Faculty of Computing Asia Pacific University KL Malaysia

Received: 26 Jul. 2014, Revised: 27 Oct. 2014, Accepted: 28 Oct. 2014
Published online: 1 May 2015

Abstract: This study provides an overview of popular and famous algorithms and techniques in shadow maps generation. Well- known techniques in shadow maps generation is described detail, along with a discussion of the advantages and drawbacks of each. Basic ideas, improvements and future works of the techniques are also comprehensively summarized and analyzed in depth. Often, programmers have difficulty selecting an appropriate shadow generation algorithm that is specific to their purpose. We have classified and systemized these techniques. The main goal of this paper is to provide researchers with background on a variety of shadow mapping techniques so as make it easier for them to choose the method best suited to their aims. It is al-so hoped that our analysis will help researchers find solutions to the shortcomings of each technique.

Keywords: Shadow techniques, hard shadows, shadow rendering, shadow mapping

1 Introduction

Shadows are one of the prominent effects to make a realistic virtual environment as, the distance between objects will be observed through the shadows with respect to the light sources. Although shadows are the main factor of realist virtual environments, they are expensive in order to rendering in real-time [19, 17]. Video games equipped with shadows look realistic as gamers feel playing in real environments and enjoy as much as possible [16].Games without shadows are not pleasurable to gamers. Nowadays, video games supported with soft shadows or even hard shadows are in order to enhance the realism of virtuality.

Rendering of realistic virtual environment in real-time has always been a main issue in game engines. In the real life, natural scenes include a huge number of small and big materials that are difficult to model and take a lot of time to render and require a substantial amount of memory. Victory on this problem has been an attractive topic and challenging problem for many researchers during the couple of current centuries.

Many algorithms are proposed for shadow generation. Shadow Maps and Shadow Volumes are the well-studied algorithms for casting shadows on other virtual objects. Geometrically-based of shadow volumes causes high number of calculations [20].

In this paper the basic ideas, improvements and future works of the techniques are presented. The main goal of this paper is to provide researchers with background on a variety of techniques so as make it easier for them to choose the method best suited to their aims. It is also hoped that our analysis will help researchers find solutions to the shortcomings of each technique. Compared to previous surveys, the main strengths of this research include the following four points: First, it includes all widely used algorithms and shadow generation techniques and describes them in detail, with a main focus on current algorithms. Secondly, we have tried to reveal all the advantages and shortcomings of each algorithm and technique. Thirdly, a hierarchy of the development of each algorithm based on existing drawbacks is expressed. Finally, the current work

* Corresponding author e-mail: hoshang@utm.my, shahrizal@utm.my

proposes a series of suggestions which are aimed to remove the drawbacks and improve the short-comings of each algorithm.

1.1 Shadow Maps

Shadow maps are significant milestone in the evolution of shadow generation in computer graphics [39]. They are image-based, therefore there is no need more calculation compare to shadow volumes which are expensive in the case of rendering. Shadow maps are fast enough in rendering then they are employed in different fields of computer graphics such as computer games, animations [14] and motion vision [18]. Shadow mapping algorithms include two phases. At the first stage, the whole scene must be rendered from the light position. Second, the whole scene must be rendered again but from camera point of view. If any pixels can be seen from second rendering but not from first rendering it means located in shadows else in lit. Shadow maps are convenient for generating soft shadows because it has aliasing and to omit aliasing, most of proposed algorithm creates a soft outline automatically. The variety of different algorithms in shadow mapping is one of the problems faced by researchers by complicating the selection of the appropriate algorithm. In the following the most popular algorithms are introduced.

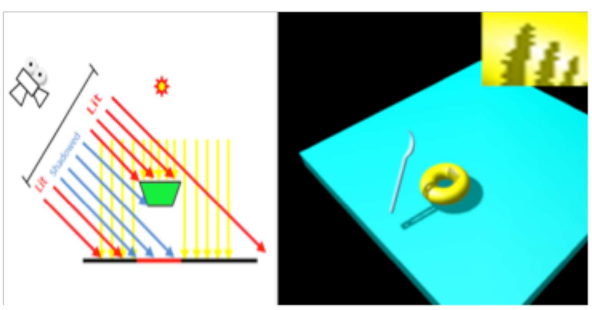

Fig. 1: Left: theory of shadow mapping. Right: standard shadow mapping

Generating shadows using shadow maps algorithms are easy, fast and widely used. The efficient use of shadow maps create soft shadows of high quality and higher frames per second compared to geometrical based algorithms. In comparison with shadow volumes, they have some advantages and disadvantages [35]. In this regard several researchers explored and reported in state of art [8,41,33,7,42,1,2,26,17].

1.2 Image Based Algorithms

Many techniques based on shadow mapping, have been proposed to alleviate aliasing. For instance, super-sampling, adaptive sampling and stochastic samplings are three unique examples. Super-sampling uses more than ocitene regular spaced sample per pixel. It reduces the aliasing without regard to the number of samples. Adaptive sampling uses additional rays to trace the edges between the previous rays [15]. This method suffers from the huge number of required rays. Although, both techniques do not eliminate aliasing, adaptive sampling is better for anti-aliasing.

Aliasing is a critical drawback of shadow maps [35]. Many researchers are involved in solving aliasing problems of shadow maps [8,41,33,7,16,42,26,32,12]. Some of the interesting algorithms that have been proposed are introduced in the next section.

Yang et al. [41] improved VSM and published a paper entitled" Variance Soft Shadow Mapping" (VSSM). VSSM is based on PCF [33] to reduce the aliasing by filtering. Dimitrov [7] with the NVIDIA Corporation proposed cascade shadow to create soft shadows. This new technique was based on a type of shadow mapping called" cascaded shadow map" (CSM). CSMs are usually useful for wide scenes such as large terrain or forest [24]. Generating some layers and set the appropriate resolution enhance the quality of shadows for wide environments. CSMs are suitable for semi-soft shadowing which are used in outdoor rendering.

1.3 Filtering

Reeves et al. citeReeves1987proposed a technique to reduce aliasing by filtering, a technique which they called Percentage-Closer Filtering (PCF). This algorithm, by filtering the depth map with interpolation of binary data, attempted to produce monotonic data in the outline of a shadow. This method involved binary values. If applied to depth maps, it would cause the shadow edges to be sharp, and soft anti-aliasing would not be possible. In original shadow maps, if the depth map is less than the texture average the pixel is in shadow. The opposite is true in lit regions. Traditional shadow mapping features sharp outlines and a lot of aliasing. In contrast to traditional shadow mapping, PCF follows these steps in reverse, which means that the data must be filtered first to produce new binary data. The next step is to produce the filtered data average. The new data can range between 0 and 1 without the requirement that the data be exactly 0 or 1. Figure 2 illustrates this as well. Some games and game engines use PCF directly to implement soft shadows. Fernando [9] proposed a technique known as Percentage Closer Soft Shadow (PCSS) to create soft shadow. It includes two steps - occluder search and PCF filtering. Some drawbacks of PCSS are:

–A single planer near the shadow map that does not have a convenient approximation must be assumed
–No appropriate self-shadowing in some situations and requires more bias
–A fraction of light which is blocked by occluder must be considered to equal the PCF filter weight
–Incorrect estimation of average depth, therefore, results in the calculation of incorrect visibility values
–Searching shadow maps and performing percentage closer filtering requires the computation of more shadow maps, leading to non-achievable performance.

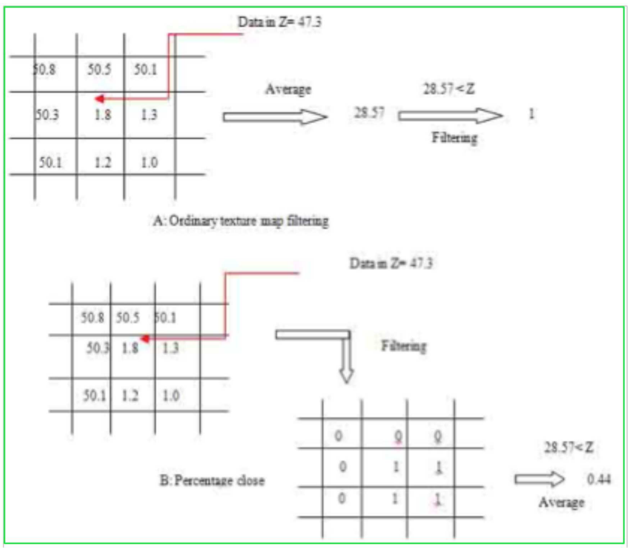

Fig. 2: Comparison between ordinary texture map filtering (A) and percentage closer filtering (B)

1.4 Transparent Supporting

Lokovic et al. [31] proposed a technique entitled Deep Shadow Maps to reduce aliasing. Deep shadow maps need to store fractional visibility functions. As a result, percentage closer filtering can be used as a pre-processing calculation. Deep shadow maps produce high quality and fast shadows for very complex objects such as hair. This algorithm has some advantages and shortcomings. They are faster in lookup and more efficient in memory use. Another benefit of deep shadow maps is the ability to produce a transparent shadow for transparent objects. They also support self-shadowing. A further advantage of these algorithms are using for shadow of motion objects. This algorithm can implement shadows of non-rigid objects such as glasses and clouds. Finally, the last advantage of deep mapping is mip-mapping support. Deep shadow mapping reduces lookup costs especially when different scales are used in viewing.

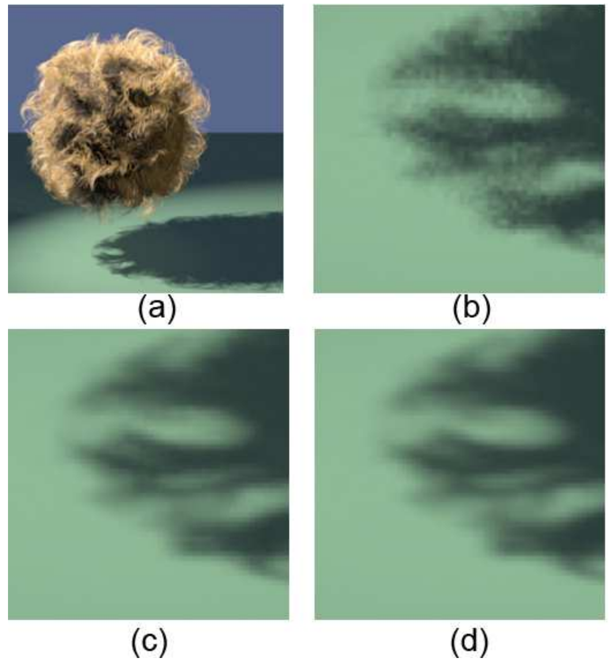

Fig. 3: (a) A ball of hair with 50,000 hairs. (b) Normal shadow map (512*512), (c) Normal shadow map (4096*4096), (d) Deep Shadow Maps (512*512)

Expensive rendering is the biggest shortcomings of these kinds of algorithms compare to traditional shadow maps with the same pixel resolution.

1.5 Synchronizing and Matching the Resolution

Pixel size mismatches between two renderings of environment in shadow maps causes aliasing. The pixel size in the view point and the light point are different because of two different points of view. Fernando et al. [9] resolved shadow mapping aliasing by using a technique to synchronize the size of a pixel in both view-points. This was accomplished by saving the depth map in a different structure on be half of the normal flat two-dimensional view. A hierarchical structure is taken into account to enhance the image quality. In creasing the resolution is another technique which is employed to increase the quality. ASMs store depth map in a quad-tree. Each node of the quad-tree is supported by a fixed resolution. Each shadow map is split by a fixed number.

[23] proposed a technique based on adaptive shadow map algorithms called ”Resolution Matched Shadow Maps (RMSMs)”. In this method depth map must be stored in a quadtree with a desirable resolution. RMSMs

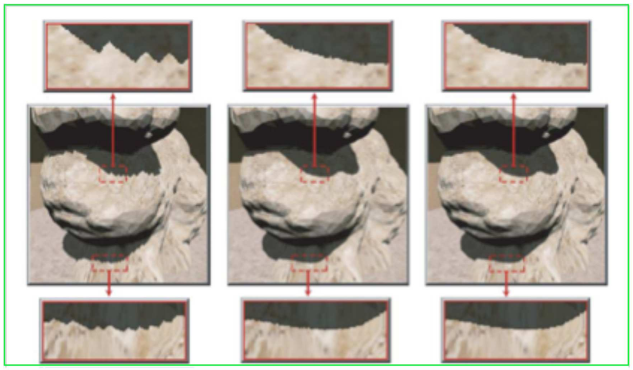

Fig. 4: A conventional 20482048 pixel shadow map

generated shadows in a single step, but ASMs generate shadows using iterative refinement, reducing the rendering time. Unlike ASMs, RMSMs require a little geometry because of the approximation of sample positions.

[34] proposed ”Adaptive Volumetric Shadow Maps (AVSM)”, which was an adaptive algorithm that was constructed on ASM using deep maps. It is technique that offers good-quality non rigid and complex data such as smoke.

[37] proposed perspective shadow maps. They implemented PSM in a normalized device coordinate space. PSMs work as standard shadow maps independent of what is captured in the camera ”Post Perspective Space (PPS)” instead of the word spaces.

At the first stage, the whole scene, including the light source was transformed to the PPS, and then the scene is render from light position. During this rendering, the depth buffer was enabled. This buffer stored an image called the ”Perspective Shadow Map”. PSM reduced the aliasing and created sharp outlines as well as shadow volumes. Assigning gradient resolution from high to low for near and far objects respectively is the main idea of this technique.

1.6 Self- Shadowing and Soft-Shadowing

[8] introduce an algorithm which called Variance Shadow Maps (VSM). The algorithm is based on shadow mapping for enhancing anti-aliasing which is the main issue of shadow maps. They stored mean (μ) and squared of depths distribution (σ^2). on be half of a single depth value (z-value). VSMs generated soft shadows as well as PCFs. The variance of depth value helps to approximate the intensity of penumbra. Another advantage of VSMs was that it supported self-shadowing. By applying a blur filtering in a separate step, soft shadow was achieved.

$$\sigma^2 = \int_{-\infty}^{+\infty} x^2 p(x)dx, \mu = \int_{-\infty}^{+\infty} xp(x)dx \qquad (1)$$

By solving the one-tailed version of Chebyshev's inequality [45] to compute the intensity for $d_{fragment} > \mu$:

$$P(d_{map} \geq d_{fragment}) \leq P_{max}(d_{fragment}) \equiv \frac{\sigma^2}{\sigma^2+(d_{fragment}-\mu)^2} \qquad (2)$$

where d_{map} is a variable of a distribution with mean and variance and $d_{fragment}$ is the fragment's depth.

To calculate the $M1$ regular shadow mapping is used. For $M2$ shadow mapping with depth squared is used. However, because of non-planarity, incorrectly-lit is still a problem with VSM.

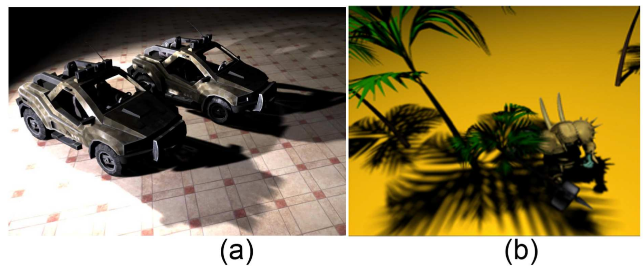

Fig. 5: (a) Variance Shadow Maps (b) Result of VSSM

[21] proposed a beneficial algorithm to solve both VSM problems. They called the algorithm Layered Variance Shadow Maps (LVSM). LVSM divided the depth shadow into layers. This technique has some advantages and disadvantages.

Although it precisely reduced the texture, it was also applied to solve other problems in the VSSM method. By managing the layers it eliminated leakage between layers. More layers had less bleeding resulting in higher quality.

1.7 Partitioning

[7] proposed a new technique which could generate semi-soft shadows entitled ”Cascaded Shadow Maps” (CSM). Cascaded shadow maps tried to reduce aliasing by increasing high resolution for areas in the scene which were close to the view point and by decreasing the resolution for areas of the scene that were far away. CSM splits the view frustum into some segments and fills the z-buffer for each segment separately. CSM was used for wide scenes such as large terrain. CSM algorithm is done in two steps. Rendering from light source position and camera position are the two different data which store in z-depth respectively. Allocating an appropriate resolution is the next step of this algorithm which gives good enough quality.

Recently, [24] worked on a ”Cascade Shadow Map” for the cases that direction of camera view and direction of the light not perpendicular. They presented a Light

Space Cascade Shadow Map (LiSCSM) and tried to solve the shortcomings of cascade shadow map by splitting the scene into non-intersection layers and creating one depth for each of the layers in the light space. This allowed for the existence of different shadow maps without any intersection sample points. To have soft enough shadows and avoid shadow flicking, the algorithm was combined with a stable cascade shadow map.

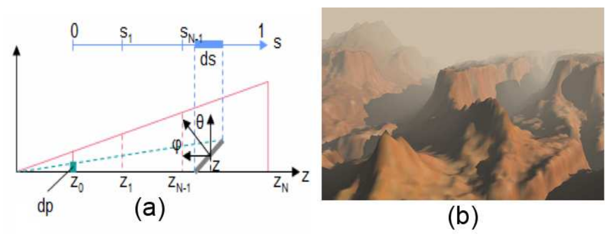

Fig. 6: (a) Theory of view frustum splitting, (b) Result of CSM with 3 partitions in 1024*1024 resolution

[3] proposed a new technique named Convolution Shadow Maps that became an effective arbitrary linear shadow filter. Convolution Shadow Maps replaced storing depth values at each pixel by encoding a binary visibility function.

The main difference between Convolution Shadow Maps and PCF is that the Convolution Shadow Maps took into account Fourier's expansion instead of using statistical estimates of the depth distribution. Converging toward the PCF solution is another difference with respect to VSM, when more Fourier coefficients were added. PCF equation was trans-formed to Convolution Shadow Maps by exchanging square wave function to with the combination of basic functions when no shadow testing is required for filtering.

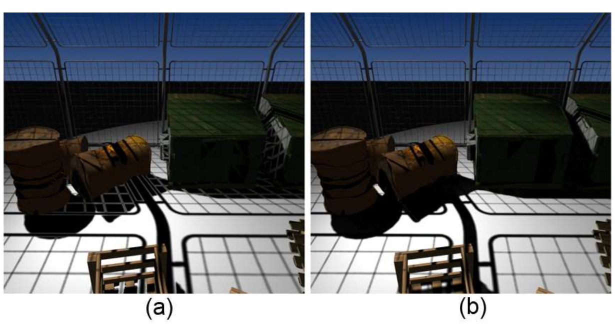

Fig. 7: A comparison between VSM and Convolution Shadow Maps

Recently Bavoil [5], proposed a new technique to create soft shadows by averaging hard shadows from multiple points of light evenly distributed on a lit area. Multi-View Soft Shadows (MVSSs) worked as accumulation-buffer rendering [18]. A feature unique to this technique was a single rendering for MVSS from point of view. The algorithm was implemented in two accurate steps. Firstly, the scene geometry was rasterized into multiple shadow maps. Each point of the light source data must be rasterized by a single shadow map. In the second step, soft shadow was rendered by averaging the hard shadows of each shadow map per pixel shade.

To avoid self-shadowing artifacts, MVSSs like PCSSs use a depth bias. This was done by pushing shadow map fragments away from the light sources. MVSSs required smaller kernels compared to PCSSs; therefore, they can use smaller depth bias without introduc-ing artifacts. This makes MVSS technique more robust than PCSS to support self-shadowing.

2 Discussion

Shadow mapping requires an extension to be implemented in current graphics hardware. It uses *GL_ARB_SHADOW* for the comparison between depth values. A problem with shadow mapping sampling occurs when surfaces are too close together. Offsetting the Z-value using a small bias is a solution to this problem [40, 13, 27].

2.1 Culling

[6] used a culling method to reduce parts of the shadow receiver that made no contribution to the visible shadows. Many researchers have used culling techniques to reduce the part of the rendering scene to speed up the rendering process by using a mask of potential shadow receivers to cull the shadow casters. Many different receiver masks have been proposed and compared in terms of rendering times and computation costs. The greatest benefit is that this method speeds up by 3x-10x for wide scenes and by 1.5x-2x for regular scenes. The algorithm can be summarized in four steps. The first two steps are to determine the shadow receivers and to generate a mask for shadow receivers. Then, using the mask, cull the shadow casters in the rendering and finally, compute the shading. The main difference between these methods and previous culling algorithms are steps 2 and 3.

The mask in the light view represents the shadow receivers. A very simple approximation of this mask can be calculated by rendering bounding boxes of shadow receivers that can be seen from view-point in the stencil buffer. Four types of receiver masks were investi-gated. They are Bounding Volume Mask (MVOL), Geometry Mask (GEOM), Combined GEOM and MVOL (GEOM+MVOL) and Fragment Mask (FRAG).

[36] used [6] in Alan Wake. To recognize visibility from a view point, a ierarchical hardware-based query

occlusion algorithm was used. They used a prototypical implementation of a shadow caster-culling algorithm [6] to obtain acceptable levels of shadow mapping. Although using a hardware-based occlusion system has some advantages, such as not requiring any modification to the pipeline or game content, an application will be required for a GPU assisted solution to synchronize the GPU and the CPU. Another drawback is that valuable GPU time is consumed, which could be a problem for some platforms.

Shadow mapping is sometimes unable to project the whole environment in a single shadow map. In this case, all light sources could be combined in one, but this solution requires several shadow maps and consequently, increases the time of rendering.

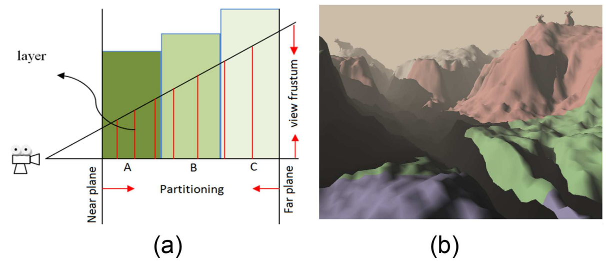

Fig. 8: Left: z-partitioning by capturing four different shadow maps for each partition. Right: result of cascade shadow mapping with four partitions.

2.2 Partitioning

Partitioning is another solution used to alleviate under-sampling errors. Z-partitioning is very well-known and widely used in current game engines. In Z-partitioning, the view frustum is split along the z-axis. Shadow map will be stored separately for each partition. Some famous algorithms which used this technique are Plural Sunlight Buffers [38], Z-Partitioning [28], Parallel Split Shadow Maps (PSSM) [42] and Cascaded Shadow Maps (CSM) [7].

There are two methods which can be used to improve z-partition algorithms. First, in all algorithms, n renderings are necessary for$n$ partitions. [43] proposed a technique to reduce the number of renderings. The second aspect is to find the best position for each split plane. Lloyd [29] [30] proposed the following formula for split position:

$$C_i = z_n \left(\frac{z_f}{z_n}\right)^{\left(\frac{i}{m}\right)} \tag{3}$$

where z_n and z_f are the near and far plane distances in the view frustum and m is partition numbers. Logarithm function changes the uniform distribution of partitioning. In this case, the resolution of near partitions could be as high as 128 or 64 bits, the middle partitions can have medium resolution of 32 or 16 bits and finally, the third part that is far away from the camera can have lesser resolution of around 8 bits or less.

Z-partitioning is much more robust than warping schemes because z-partitioning works independent of the axes of shadow maps. [44] discussed the optimization of z-partition with respect to flickering artifacts, split section, strategies of shadow map storage, and filtering across splits. [22] also proposed a probabilistic extension for Cascaded Shadow Maps based on sample distribution but this method requires special hardware to calculate the depth histogram for the clustering approach.

In frustum face partitioning, shadow map must be calculated for each face. This scheme is important for LogPSM [30]. [22] presented Sample Distribution Shadow Maps (SDSM), an automatic enhancement for anti-aliasing of shadow mapping. SDSMs greatly reduced oversampling, under sampling and geometric aliasing errors. The idea behind this method was to find the fixed size and placement of z-partitions using the distribution of the camera-space depth buffer for a current frame with a light projection matrix. The algorithm looked for actual light distribution samples of space shadow maps and found an efficient place for each partition [4]. The world space position for each depth buffer sample was reconstructed and then projected into the light space in order to generate shadow samples.

In basic logarithmic distributions, more partitions are located near the camera. As a result, there may be no objects on some partitions, whereas the SDSM logarithm is an automatic recognition technique used to find efficient partitions. Adaptive partitioning, which reduces under-sampling errors, is the last kind of partition-ing technique to be discussed here. Many techniques had been proposed to allocate samples in an optimal way before storing the shadow maps. The more prominent techniques are ”Adaptive Shadow Maps” [9]. ”Tiled Shadow Maps” [4], ”Resolution Matched Shadow Maps” [23], ”Queried Virtual Shadow Maps” [11] and ”Fitted Virtual Shadow Maps” [10,25].

3 Advantages and Shortcomings of Shadow Maps

3.1 Advantages

A hardware-based occlusion system is fully free of any geometrical calculations because of image basing. Stencil buffer is not required. Moreover, it can easily support primitives instead of supporting only polygons. The ability of GPU is one of the advantages of each algorithm in computer graphics as they can fully accelerate on GPU.

Shadow mapping can be implemented entirely on GPU. Rendering speed is a critical subject in real-time rendering. Using current hardware, shadow maps can be generated faster than shadow volumes can be generated. Of course, speed depends on the com-plexity of the scene. Self-shadowing is an important factor for shadow algorithms; shadow maps can support it as well as shadow volumes. Finally, shadow mapping can create soft shadows.

3.2 Shortcomings

Aliasing is the biggest problem of shadow maps. Aliasing is due to synchronization of the two depth maps. The scene must be rendered separately for each light source. This will take more time for an omnidirectional light point. It requires a 180 degree shadow frustum that must be handled by buffering. A problem occurs when a light source is located inside the scene as this situation requires six buffers to handle all shadow cases. Another shortcoming of shadow maps is that they are view-dependent and not robust.

Incorrect self-shadowing is another problem of shadow maps. This is due to imprecision and under-sampling of the depth values that store a shadow map per texel. Biasing is needed in order to obtain robust results.

4 Conclusion

In this survey, we have discussed a variety of famous image based shadow algorithms. An overview on each of them was presented. Identifying the main advantages of each algorithm was a prominent characteristic of this paper. The shortcomings of the earlier algorithms are the main reason new algorithms were devised. Most importantly, how each algorithm works was discussed. The figures illustrate the properties, advantages and drawbacks of the algorithms. Finally, it was suggested that it would be beneficial to know which algorithm is best to use in different situations. Image based algorithms are convenient when the cost of rendering is an important consideration. In a situation that calls for a single shadow map, LiSPSMs is a good choice as the algorithm consumes as much as a traditional shadow mapping technique, especially for outdoor rendering. Modification of LiSPSMs using the method proposed by Lloyd et al. [27] is recommended to keep the quality of shadow maps.

Z-partitioning is the most appropriate technique to employ when multi shadow map are needed. Most of the z-partitioning based algorithms are appropriate for this situation, especially CSMs and PSSMs. Z-partitioning based algorithm is also convenient in situations where there are large scenes, such as most of the outdoor rendering games. Adaptive partitioning is the best algorithm when high quality is desired.

PCF is less expensive and alleviates some errors and aliasing when filtering is required, especially for small filtering kernels. For large filtering kernels, LVSMs perform better. PCF is best used in cases requiring both anti-aliasing and soft shadow generation and where rendering time is a concern.

As we have shown, none of the algorithms are suitable for all situations nor can they solve all the problems simultaneously. We hope this review can help researchers choose the best algorithm to meet their goals.

Acknowledgement

The research paper was supported by Universiti Teknologi Malaysia (UTM), Institut Jantung Negara (IJN) and King Saud University (KSU). Our deepest thanks and appreciation to the Deanship of Scientific Research at King Saud University (KSU) Riyadh Kingdom of Saudi Arabia for funding this research. We are also thankful to our colleague researchers for their assistance in neural network training and testing phases. The authors are grateful to the anonymous referee for a careful checking of the details and for helpful comments that improved this paper.

References

[1] R. Amin, R. Gould, W. Hou, R. Arnone, and Z. Lee. Optical algorithm for cloud shadow detection over water. 2013. IEEE.

[2] T. Annen, Z. Dong, T. Mertens, Hans-Peter Bekaert, P. andSeidel, and Jan Kautz. Real-time, all-frequency shadows in dynamic scenes. *ACM Transactions on Graphics (Proceedings of ACM SIGGRAPH 2008)*, 27(3):1–34, 2008.

[3] T. Annen, T. Mertens, P. Bekaert, H.p. Seidel, and J. Kautz. Convolution shadow maps. *Proceedings of the 18th Eurographics conference on Rendering Techniques*, pages 51–60, 2007. Eurographics Association.

[4] J. Arvo. Tiled shadow maps. *Computer Graphics International, 2004. Proceedings*, pages 240–246, 2004. IEEE.

[5] L. Bavoil. Multi-view soft shadows. *NVIDIA technical report*, 2011.

[6] J. Bittner, O. Mattausch, A. Silvennoinen, and M. Wimmer. Shadow caster culling for efficient shadow mapping. *In Symposium on Interactive 3D Graphics and Games, ACM, New York, NY, USA, I3D 11*, pages 81–88, 2011.

[7] R. Dimitrov. Cascaded shadow maps. *NVIDIA, Technical Report*, 2007.

[8] W. Donnelly and A. Lauritzen. Variance shadow maps. *In Proceedings of the 2006 ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, pages 161–165, 2006.

[9] R. Fernando, S. Fernadez, K. Bala, and D.P. Greenberh. Adaptive shadow maps. *In Proceedings of ACM SIGGRAPH ACM Press / ACM SIGGRAPH*, pages 387–390, 2001.

[10] M. Giegl and M. Wimmer. Fitted virtual shadow maps. *In Proceedings of GI (Graphics Interface) C. G. Healey and E. Lank (Eds.). Canadian Human-Computer Communications Society, Montreal, Canada, (2007)*, pages 159–168, 2007.

[11] M. Giegl and M. Wimmer. Queried virtual shadow maps. *In Proceedings of I3D (ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games) ACM Press*, pages 65–72, 2007.

[12] J. Gumbau, M. Sbert, L. Szirmay-Kalos, M. Chover, and C. Gonzlez. Smooth shadow boundaries with exponentially warped gaussian filtering. *Computers and Graphics (Pergamon)*, 37(3):214–224, 2013.
[13] J.M. Hasenfratz, M. Lapierre, N. Holzschuch, and F. Sillion. A survey of real-time soft shadows algorithms. *Computer Graphics Forum*, 22(4):753–774, 2003.
[14] I. Ismail, H. Kolivand, M.S. Sunar, and A. H. Basori. An overview on dynamic 3d character motion techniques in virtual environments. *Life Science Journal*, 10(3), 2013.
[15] K. Jiang, A.-H. Li, and Y.-Z. Su. Adaptive shadow detection based on global texture and sampling deduction. *Guangdianzi Jiguang/Journal of Optoelectronics Laser*, 23(11):2174–2179, 2012.
[16] H. Kolivand, Z. Noh, and M. S. Sunar. A quadratic spline approximation using detail multi-layer for soft shadow generation in augmented reality. *Multimedia Tools and Applications*, 73(3):1225–1245, 2014.
[17] H. Kolivand and M. S. Sunar. Realistic real-time outdoor rendering in augmented reality. *PloS one*, 9(9):e108334, 2014.
[18] H. Kolivand and M.S. Sunar. An overview on based real-time shadow techniques in virtual environment. *TELKOMNIKA*, 10(1):171–178, 2012.
[19] H. Kolivand and M.S. Sunar. Real-time outdoor rendering using hybrid shadow maps. *International Journal of Innovative Computing, Information and Control (IJICIC)*, 8(10B):7169–7184, 2012.
[20] H. Kolivand and M.S. Sunar. A survey of shadow volume algorithms in computer graphics. *IETE Technical Review*, 30(1):38, 2013. Medknow Publications.
[21] A. Lauritzen and M. McCool. Layered variance shadow maps. *In GI '08: Proceedings of Graphics Interface 2008 Toronto, Ontario, Canada), Canadian Information Processing Society*, pages 139–146, 2008.
[22] A. Lauritzen, M. Salviy, and A. Lefohnz. Sample distribution shadow maps. *In Advances in Real-Time Rendering in 3D Graphics and Games*, 2010.
[23] A. E. Lefohn, S. Sengupta, and J. Owens. Resolution-matched shadow maps. *ACM Trans. Graph.*, 26(4):1–17, 2007.
[24] X. H. Liang, S. Ma, L. X. Cen, and Z. Yu. Light space cascaded shadow maps algorithm for real time rendering. *Journal of Computer Science and Technology*, 26(1):176–186, 2011. Springer.
[25] L. Liu, W. Zhou, and H. Li. Soft shadow mapping by line fitting. *Jisuanji Fuzhu Sheji Yu Tuxingxue Xuebao/Journal of Computer-Aided Design and Computer Graphics*, 24(12):1533–1541, 2012.
[26] Y. Liu and X. Granier. Online tracking of outdoor lighting for augmented reality with moving cameras. *IEEE Transactions on Visualization and Computer Graphics 18(4)*, pages 573–580, 2012.
[27] B. Lloyd. Logarithmic perspective shadow maps. *Ph.D Thesis*, 2007. University of North Carolina at Chapel Hill.
[28] D. B. Lloyd, D. Tuft, S. Yoon, and D. Manocha. Warping and partitioning for low error shadow maps. *Proceedings of the 17th Eurographics conference on Rendering Techniques*, pages 215–226, 2006. Eurographics Association.
[29] D. B. Lloyd, J. Wendt, N. K Govindaraju, and D. Manocha. Cc shadow volumes. *Proceedings of the Fifteenth Eurographics conference on Rendering Techniques*, pages 197–205, 2004. Eurographics Association.
[30] D.B. Lloyd, N.K. Govindaraju, C. Quammen, S.E. Molnar, and D. Manocha. Logarithmic perspective shadow maps. *ACM Transactions on Graphics*, 27(4):1–32, 2008.
[31] T. Lokovic and E. Veach. Deep shadow maps. *In Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, pages 385–392, 2000.
[32] Ajay K Mandava, Emma E Regentova, and George Bebis. Lfad-locally and feature-adaptive diffusion based image denoising. *Applied Mathematics and Information Sciences*, 8(1), 2014.
[33] W.T. Reeves, D.H. Salesin, and P. L. Cook. Rendering antialiased shadows with depth maps. *Computer Graphics (Proceedings of SIGGRAPH 87*, 21(4):557–562, 1987.
[34] M. Salvi, K.l. Vidimce, A. Lauritzen, and A. Lefohn. Adaptive volumetric shadow maps. *Computer Graphics Forum (Proceedings of EGSR 2010)*, 29(4):1289–1296, 2010.
[35] D. Scherzer, M. Wimmer, and W. Purgathofer. A survey of real-time hard shadow mapping methods. *Computer Graphics Forum*, 30(1):169–186, 2011.
[36] A. Silvennoinen, T. Soininen, M. Mäki, and O. Tervo. Occlusion culling in alan wake. *ACM SIGGRAPH 2011 Talks*, page 47, 2011. ACM.
[37] Stamminger, Marc, and G. Drettakis. Perspective shadow maps. *In Proceedings of SIGGRAPH*, 2002.
[38] K. Tadamura, X. Qin, G. Jiao, and E. Nakamae. Rendering optimal solar shadows using plural sunlight depth buffers. *In CGI 99: Proceedings of the International Conference on Computer Graphics (Washington, DC, USA, 1999), IEEE Computer Society*, page 66, 1999.
[39] L. Williams. Casting curved shadows on curved surfaces. *SIGGRAPH '78*, 12(3), 270-274 1978.
[40] A. Woo, P. Poulin, and A. Fournier. A survey of shadow algorithms. *Computer Graphics and Applications, IEEE*, pages 13–32, 1990.
[41] B. Yang, Z. Dong, J. Feng, H.P. Seidel, and J. Kautz. Variance soft shadow mapping. *Computer Graphics Forum*, 29(7):2127–2134, 2010.
[42] F. Zhang, H. Sun, and O. Nyman. Parallel-split shadow maps on programmable gpus. *GPU Gems*, 3:203–237, 2007.
[43] F. Zhang, H. Sun, L. Xu, and L. K. Lun. Parallel-split shadow maps for large-scale virtual environments. *Proceedings of the 2006 ACM International Conference on Virtual Reality Continuum and its Applications*, pages 311–318, 2006. ACM.
[44] F. Zhang, A. Zaprjagaev, and A. Bentham. Practical cascaded shadow maps. *ShaderX*, 7:305–330, 2009.
[45] L. Zhou and Z. C. Hu. Chebyshevs inequality for banach-space-valued random elements. *Statistics & Probability Letters*, 82(5):925–931, 2012.

Hoshang Kolivand has obtained his Ph.D in UTM ViCubelab in Universiti Teknologi Malaysia in 2013. His research interests include Computer Graphics and Augmented Reality. Received the M.S degree in Applied Mathematics and computer from Amirkabir University, Iran in 1999. Previously he was a lecturer in Shahid Beheshti University Iran. He has published enormous articles in international journals, conference proceedings and technical papers including article in books. Dr. Hoshang Kolivand is an active reviewer of many conferences and international journals. He has published many books in object-oriented programming and mathematics.

Mohd SHahrizal Sunar obtained his PhD from National University of Malaysia in 2008. His major field of study is real-time and interactive computer graphics and virtual environment. He received his MSc in Computer Graphics and Virtual Environment (2001) from The University of Hull, UK and BSc degree in Computer Science majoring in Computer Graphics (1999) from Universiti Teknologi Malaysia. He served as academic member at Computer Graphics and Multimedia Department, Faculty of Computer Science and Information System, Universiti Teknologi Malaysia since 1999. Since 2009, he had been given responsiblity to lead the department. The current research program that he lead are Driving Simulator, Augmented Reality, Natural Interaction and Creative Content Technology. He had published numerous articles in international as well as national journals, conference proceedings and technical papers including article in magazines. Assoc. Prof. Dr. Shahrizal is an active professional member of ACM SIGGRAPH. He is also a member Malaysian Society of Mathematics and Science.

Ayman Altameem is vice dean in college of applied studies and community services King Saud University Riyadh KSA. He received his PhD in Information Technology, Computing, University of Bradford, Bradford, UK. and M.Sc. Information Systems, Computing, London South Bank University, London, UK. His keen interests are E-commerce, Information Systems and Artificial Intelligence.

Amjad Rehman is an assistant prof. in MIS department CBA Salman Abdul Aziz University Alkharj KSA. He received his PhD from Faculty of Computing Universiti Teknologi Malaysia with specialization in Intelligent Data Mining, Forensic Documents Analysis, OCR & Visulization. Dr. Rehman is author of more than 30 international journal papers indexed with ISI/SCIE .

Mueen Uddin Mueen Uddin is Post-Doctoral Research Fellow at International Islamic University Malaysia. He has his PhD from Universiti Teknologi Malaysia UTM in 2012. His research interests include Green IT, Energy Efficient data centers, Green Metrics, Global Warming Effects, Virtualization, Cloud Computing, Digital content protection and deep packet inspection, intrusion detection and prevention systems, MANET routing protocols. Dr. Mueen has BS & MS in Computer Science from Isra University Pakistan with specialization in Information Networks. Dr. Mueen has published more than 35 international Journal Papers in indexed and reputed journals.