EG UK Theory and Practice of Computer Graphics (2009)
Wen Tang, John Collomosse (Editors)

An Aliasing Theory of Shadow Mapping

Fan Zhang†, Chong Zhao and Hanqiu Sun

Department of Computer Science and Engineering, The Chinese University of Hong Kong, China

Abstract

Shadow mapping is a popular image-based technique for real-time shadow rendering. Although numerous improvements have been made to help anti-aliasing in shadow mapping, there is a lack of mathematical tools that allow us to quantitatively analyze aliasing errors in its variants. In this paper, we establish an aliasing theory to achieve this goal. A generalized representation of aliasing errors is derived from a pure mathematical point of view. The major highlight of this representation is the ability of quantifying the aliasing error at any position for general view-light configurations. On the contrary, due to the geometric assumptions used in the computational model, previous work analyzes the aliasing only along the view direction in the simplest case where the light and view directions are orthogonal. Subsequently, as a direct application of our theory, we present a comparison of aliasing distributions in a few representative variants of perspective shadow maps. We believe that these theoretical results are useful to better understand shadow mapping, and thus inspire people to develop novel techniques in this area.

Categories and Subject Descriptors (according to ACM CCS): I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism – Shading, Shadowing

1. Introduction

The perception of shadowing effects is crucial for the immersion in virtual scenes in computer graphics. Although the concept of shadows is straightforward, high-fidelity and real-time shadow rendering is quite challenging due to the visibility complexity in global illumination. Most real-time shadow algorithms nowadays are based on *shadow volumes* [Cro77] and/or *shadow mapping* [Wil78]. Shadow mapping is more often used in rasterization-based render systems, because its image-based nature makes the performance mainly relate to the buffer resolution rather than the geometry complexity in object-based algorithms such as shadow volumes. Shadow mapping is basically two-pass rendering algorithm. The first pass is used to generate the depth texture when rendering the scene from the light's point of view, and in the second pass the scene is rendered from the eye's point of view again with the shadow information determined from the depth texture.

The main challenge when using shadow mapping is the annoying aliasing problem which is also from its image-based nature. Among countless ideas of doing anti-aliasing in shadow mapping, *perspective reparameterization* techniques are proposed based on the observation that the sampling density at the light in the post-perspective space better accommodates the requirements for the reconstruction of shadowed images, in which shadow maps are warped using a perspective transform to reduce aliasing errors for the objects near the viewer. Representative perspective reparameterization techniques include Perspective Shadow Maps (PSMs) [SD02] and its practical implementation [Koz04], Light Space Perspective Shadow Maps (LiSPSMs) [WSP04] and Trapezoidal Shadow Maps (TSMs) [MT04]. As explained in previous work [WSP04] [LTYM06] [ZSZW08], the essential difference among these representative reparameterizations is the selection for the distance of the projection reference point to the near plane of the warping frustum.

A key inspiration of perspective reparameterizations to the shadow mapping theory is that, the quantitative analysis of aliasing errors is doable by only considering *perspective aliasing errors* (subsection 2.2) because this kind of aliasing is scene independent. In this paper, we follow this research line to further investigate the mechanism of aliasing errors from a mathematical perspective and then establish a general aliasing theory in shadow mapping.

† Corresponding Author: fzhang@cse.cuhk.edu.hk

DOI: 10.2312/LocalChapterEvents/TPCG/TPCG09/093-100

Motivation: This paper is motivated by the following limitations in most of existing analysis results [WSP04] [LTYM06] [ZXTS06] [ZSZW08], which essentially result from the aliasing representation used (subsection 2.1).

- The aliasing errors are previously approximated by a geometrical intuition from an ideal scenario where the light and view directions are orthogonal. Because the angle between the light and view directions isn't taken into account by this direction-independent approximation, we're not able to compute the aliasing distribution for a general light-view configuration in dynamic scenes.
- Furthermore, even in this ideal view-light configuration, the representation of aliasing errors derived from the geometric intuition is no longer valid for points that do not locate at the view direction. In other words, such analysis is not able to measure the aliasing distribution within the whole 3D space.

Main Contributions: The main contributions of this paper are as follows:

- We establish a generalized representation of aliasing errors in perspective reparameterizations based on the solid mathematical analysis. Given appropriate assumptions, the aliasing errors in most perspective reparameterizations including standard shadow maps are unified by this representation.
- As a direct application of our theory, we compare the aliasing distributions of a few representative perspective parameterizations in the general case. The aliasing distributions in horizontal and vertical dimensions of the screen are visualized to guide people to better understand and further optimize these techniques.

1.1. Related Work

The literature of shadow rendering is vast and we thus only review the most relevant work in this paper. Refer to [WPF90] for a survey of research in this field.

An intuitive observation is that aliasing mainly occurs at shadow boundaries. This observation motivates hybrid techniques [CD04] [AA04] [AAM04] [GLY*03] and Shadow Silhouette Maps [SCH03]. In hybrid techniques, the shadows at shadow boundaries are generated by using shadow volumes instead. This mixture of image-based technique and object-based technique usually offers a better tradeoff between performance and quality. However, such hybrid techniques also inherit drawbacks of both methods at the same time. Instead of using shadow volumes, Shadow Silhouette Maps improve shadow qualities at shadow boundaries by using extra silhouette maps of the scene, but unexpected artifacts might be introduced when using silhouette maps as only one silhouette point is stored in each shadow map texel.

Partitioning algorithms [FFBG01] [Arv04] [TQJN01] [ZSXL06] nowadays becomes more popular in practice because of its simplicity and stability. The basic idea is straightforward: the scene is partitioned into multiple smaller parts and then generating a shadow map for each of them. The main cares in practice include the flickering of shadows and extra performance cost from multiple rendering passes. Another advantage of partitioning algorithms is that they are orthogonal with other shadow mapping techniques. For example, Lloyd et al. [LTYM06] proposed to integrate LiSPSMs and partitioning algorithms based on an analysis of perspective aliasing in the ideal case. A drawback of this method is that shadow map texels might be wrongly sheared when transforming the subdivisions to the canonical space. Considering the performance issue, in this paper, we only study the approaches using a single shadow map texture (although our theory is also valid for each shadow map in partitioning algorithms).

Perspective reparameterizations have been generalized in different ways. The perspective transform can be generalized to guarantees perfect sampling for the objects lie on a few user-specified planes [CG04]. Such generalized transform actually results in tilting the shadow map plane according to these planes. However such oblique shadow plane can not be globally optimal for the whole scene. Zhang et al. [ZXTS06] [ZSZW08] extend perspective reparameterizations to the general case by preserving the user-defined aliasing distribution (e.g. linear or minimum-norm) over the depth range when the viewer and/or light is moving. In comparison with our analysis in this paper, such generalizations do not take into account the aliasing distribution in the horizontal dimension and only work for points on the viewing line rather than the whole view frustum.

A lately remarkable improvement in shadow mapping is the emergence of pre-filtering techniques [DL06] [AMS*08] [Sal08] which make shadow maps filterable like normal textures. Previously, the only way of filtering shadow maps is Percentage Closer Filtering (PCF) [RSC87] in which the filtering occurs *after* the depth comparison. Such *post*-filtering prevent us from exploiting the built-in hardware filtering functionality. However, you need to carefully handle the light bleeding issue due to the incomplete approximation of the visibility step function.

1.2. Preliminaries

Consider an arbitrary point $\mathbf{c}(x,y,z)$ in the eye's coordinates system $(O;X,Y,Z)$, whose projections on the screen and shadow plane are $\mathbf{p}(q,p)$ and $\mathbf{s}(t,s)$ respectively. A small increment $\mathrm{d}\mathbf{c} = (\mathrm{d}x,\mathrm{d}y,\mathrm{d}z)$ in the view space causes a shift $\mathrm{d}\mathbf{p} = (\mathrm{d}q,\mathrm{d}p)$ on the screen, and an offset $\mathrm{d}\mathbf{s} = (\mathrm{d}t,\mathrm{d}s)$ on the shadow plane. The aliasing errors in horizontal and vertical dimensions of the screen, E_q and E_p respectively, are quantified as

$$(\mathrm{E}_q,\ \mathrm{E}_p) = \left(\left| \frac{\mathrm{d}q}{\mathrm{d}t} \right|,\ \left| \frac{\mathrm{d}p}{\mathrm{d}s} \right| \right). \tag{1}$$

In this paper, the above E_q and E_p are termed *aliasing functions* in shadow mapping.

Since q, p, t and s are all functions with respect to the view coordinates (x,y,z), the differentials involved in the aliasing functions can be explicitly calculated using the *total differential equation*. For instance,

$$\mathrm{d}q = \bigtriangledown q \cdot \mathbf{dc} \tag{2}$$

where $\bigtriangledown = (\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z})$ and $|\cdot|$ denotes the dot product of vectors. For given scene configuration, the normalized screen coordinates (q,p) are determined by the eye's projection transformation. Without loss of generality, in our analysis, we assume the field-of-views (FOVs) of the view frustum on both X and Y directions are 2ϕ. In Direct3D (similar in OpenGL), the normalized device coordinates (q,p) are transformed from the view coordinates (x,y,z) using the following perspective projection transform,

$$(q,\ p) = (\frac{1}{\tan\phi}\frac{x}{z},\ \frac{1}{\tan\phi}\frac{y}{z})$$

From Eq. (2), the differentials of screen coordinates are thus

$$\bigtriangledown q = \left(\frac{1}{\tan\phi}\frac{1}{z},\ 0,\ -\frac{1}{\tan\phi}\frac{x}{z^2}\right) \tag{3}$$

$$\bigtriangledown p = \left(0,\ \frac{1}{\tan\phi}\frac{1}{z},\ -\frac{1}{\tan\phi}\frac{y}{z^2}\right) \tag{4}$$

Substituting Eqs. (3) and (4) into Eq. (1),

$$(E_q,\ E_p) = \frac{1}{\tan\phi}\frac{1}{z}\left(\left|\frac{dx}{dt}\left(1-\frac{x}{z}\frac{dz}{dx}\right)\right|,\ \left|\frac{dy}{ds}\left(1-\frac{y}{z}\frac{dz}{dy}\right)\right|\right) \tag{5}$$

For the texture coordinates t and s in the above equation, the derivation of the explicit representations $t(x,y,z)$ and $s(x,y,z)$ is relatively complicated. The complexity mostly depends on how the texture transformation is designed when reparameterizing the shadow map.

1.3. Overview

To facilitate our analysis, the frequently used notations are listed in Table 1.

V	view frustum
W	warping frustum
Θ	light vector
θ	angle between view and light directions
n and f	near- and far- plane values for V
λ	near plane value for W
μ	depth range of W
ϕ and ϕ_w	half FOV of V and half FOV of W
$(O;X,Y,Z)$	eye's coordinates frame
$(O_w;X_w,Y_w,Z_w)$	W's coordinates frame
$(q,\ p)$	normalized screen coordinates in 2D
$(t,\ s)$	texture coordinates in 2D
$(E_q,\ E_p)$	aliasing functions

Table 1: *Notations.*

In this paper, we adopte the DirectX transformation matrices in our analysis: vectors are row-based and the coordinates system is left-handed. The coordinates for vectors (e.g. Θ) and 3D points (e.g. O_w) are defined in the eye's coordinates system $(O;X,Y,Z)$, which means the texture matrix in shadow mapping transforms points from the view space to the texture space.

The remainder of this paper is organized as follows: Section 2 details the derivations of the aliasing theory. As a direct application of our theory, Section 3 evaluates aliasing distributions in a few representative perspective reparameterizations. Finally, the conclusion goes to Section 4.

2. Perspective Reparameterizations

2.1. Local Representation of Perspective Aliasing

To approximate the aliasing functions represented by Eq. (5), the following assumptions are used in previous reparameterizations [SD02] [WSP04] [LTYM06] [ZXTS06] [ZSZW08].

$$1-\frac{x}{z}\frac{\mathrm{d}z}{\mathrm{d}x}\equiv\text{const.}\quad 1-\frac{y}{z}\frac{\mathrm{d}z}{\mathrm{d}y}\equiv\text{const.}\quad \frac{\mathrm{d}y}{\mathrm{d}z}\equiv\text{const.} \tag{6}$$

To satisfy the above assumptions, previous work only considers the points $\mathbf{c}(0,0,z)$ and restricts the small increment $\mathbf{dc}$ moving on small lines $\{\mathbf{l} : y = kz+b\}$, where k is a constant slope. With the assumptions, the aliasing functions are simplified as

$$E_q \sim \frac{1}{z}\left|\frac{\mathrm{d}x}{\mathrm{d}t}\right| \text{ and } E_p \sim \frac{1}{\tan\phi}\frac{1}{z}\left|\frac{\mathrm{d}z}{\mathrm{d}s}\frac{\mathrm{d}y}{\mathrm{d}z}\right| \sim \frac{1}{z}\left|\frac{\mathrm{d}z}{\mathrm{d}s}\right| \tag{7}$$

where $\sim$ stands for the linear dependence.

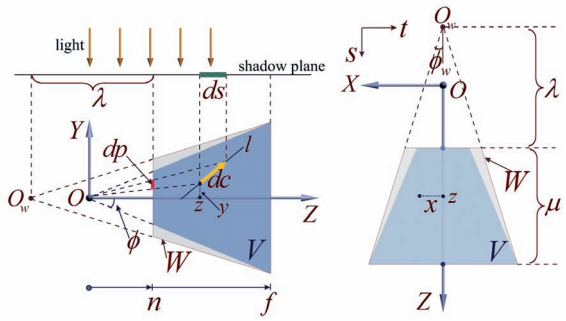

Figure 1: *Perspective reparameterizations in the ideal case. Left and right are the side view and the light's view respectively.*

For the overhead light on the YZ plane in Fig. 1, perspective shadow maps are generated in the post-perspective space of the warping transform W. The texture coordinates in standard shadow maps (SSMs) are computed by normalizing the projective coordinates into $[0,1]\times[0,1]$,

$$t(x,\ z) = \frac{1}{2}\frac{1}{\tan\phi_w}\frac{x}{z}+\frac{1}{2} \quad\text{and}\quad s(z) = \frac{\lambda+\mu}{\mu}(1-\frac{\lambda}{z})$$

A simple analysis of this scenario gives

$$\mu = f-n \quad\text{and}\quad \frac{\tan\phi_w}{\tan\phi} = \frac{f}{\mu+\lambda}$$

Combining them together and replacing dx/dt in Eq. (7) with $(\partial t/\partial x)^{-1}$, we get the following explicit representations of aliasing functions in the ideal case (note that $\approx$ occurs when approximating dx/dt with $(\partial t/\partial x)^{-1}$).

$$E_q \approx \frac{2f}{f-n+\lambda}\frac{z-n+\lambda}{z} \tag{8}$$

$$E_p = \frac{f-n}{(f-n+\lambda)\lambda\tan\phi}\frac{(z-n+\lambda)^2}{z} \tag{9}$$

Eqs. (8) and (9) are termed the *local representation* of aliasing functions. In variant reparameterizations, the free parameter λ is adjusted to control the aliasing distribution over the whole depth range. SSM is a special perspective reparameterization using a frustum with $\lambda_{\mathbf{SSM}} = \infty$. Hence,

$$E_q^{\mathbf{SSM}} \sim 1/z \quad \text{and} \quad E_p^{\mathbf{SSM}} \sim 1/z$$

It means the aliasing errors increase hyperbolically as the object moves closer to the view plane. On the contrary, PSM warps the scene and light using the view frustum, i.e. $\lambda_{\mathbf{PSM}} = n$, such that

$$E_q^{\mathbf{PSM}} \sim 1 \quad \text{and} \quad E_p^{\mathbf{PSM}} \sim z$$

The linear aliasing distribution of E_p dramatically improves the shadow qualities in the area near the viewer.

Limitations: The local representation clearly explained the motivation of perspective parameterizations. However, a few limitations still remaine due to the assumptions used in Eq. (6). Since the assumptions are satisfied only for the points on the view direction, there's no explicit way to quantitatively analyze the aliasing elsewhere. Furthermore, the local representation is only valid for the ideal case where $\theta = \pi/2$. When the light or viewer moves in the general case, the local representations can not guide us to adaptively select the appropriate warping transform according to the application-specific requirement.

2.2. Factorization of Aliasing Functions

To explicitly represent the aliasing functions in the general case, we consider the differential $d\mathbf{c} = (dx, dy, dz)$ on a small shaft $\mathbf{l}$. The small shaft $\mathbf{l}$ is parameterized with $x = k_x z - b_x$ and $y = k_y z - b_y$, where k_x and k_y denote the slopes of the line on the XZ and YZ planes respectively.

From Eqs. (2) and (5), the aliasing functions are parameterized into

$$(E_q,\, E_p) = \left(\frac{\frac{1}{\tan\phi}\frac{1}{z}\left(k_x - \frac{x}{z}\right)}{\frac{\partial t}{\partial x}k_x + \frac{\partial t}{\partial y}k_y + \frac{\partial t}{\partial z}},\ \frac{\frac{1}{\tan\phi}\frac{1}{z}\left(k_y - \frac{y}{z}\right)}{\frac{\partial s}{\partial x}k_x + \frac{\partial s}{\partial y}k_y + \frac{\partial s}{\partial z}} \right) \tag{10}$$

Perspective reparameterizations approximate the aliasing functions based on an important observation: the unknown coefficients (k_x, k_y) depend on local geometry details. Any reparameterization at a global scale cannot reduce this type of aliasing everywhere. For a light direction Θ, a practical solution is to factorize the aliasing functions in Eq. (10) into

$$E(\mathbf{c}, \Theta, \mathbf{l}) = E^{\text{pers}}(\mathbf{c}, \Theta) \times E^{\text{proj}}(\mathbf{c}, \Theta, \mathbf{l}) \tag{11}$$

where $E \in \{E_q, E_p\}$. Shadow map under-sampling can happen when *perspective aliasing errors* E^{pers} or *projection aliasing errors* E^{proj} becomes large. Projection aliasing usually happens for surfaces almost parallel to the light direction. Since projection aliasing depends on the geometrical details, the local increase of sampling densities on these surfaces is needed to reduce this type of aliasing. An inevitably expensive scene analysis at each frame is required such that using hardware-acceleration is impractical. On the other hand, perspective aliasing comes from the perspective foreshortening effect and can be reduced by warping the shadow-map. In this paper, we address the reduction of perspective aliasing errors.

2.3. Simplifications

As we mentioned before, the computation for the explicit representations of texture coordinates $s(x,y,z)$ and $t(x,y,z)$ is relatively complicated. The complexity is from two aspects:

Type of the light source: the texture matrix for point lights contains the light's perspective projection transform. It makes the mapping from view coordinates to texture coordinates not intuitive. Even a preliminary result was presented for point lights using the local representation [LGT*06], the comprehensive analysis in the general case remains challenging. In this paper, we only consider directional light sources.

Warping direction: the selection of the warping direction strongly influences the implementation complexity, because the type of light might be frequently switched between directional and point in the post-transformed space. When an inappropriate warping direction is selected, mapping singularities might be produced such that the aliasing analysis is difficult. Since any arbitrary perspective transformation can be used to warp the distribution of shadow map texels, it is sufficient to use a warping direction that's not perpendicular to the view direction like PSMs. This observation inspires Wimmer et al. [WSP04] to use a warping direction in parallel with the shadow plane, and construct the warping frustum in the light space. The main advantage of using this warping direction is that the direction of the light source doesn't change in the post-transformed space, thus no mapping singularities are generated. It greatly simplifies the implementation and aliasing analysis. In this paper, our computational model adopted the same warping direction.

2.4. Global Representation of Perspective Aliasing

With the simplifications in subsection 2.3, the warping frustum W is constructed as shown in Fig. 2. The Z_w and Y_w axes are parallel to the shadow plane and light vector respectively. The near and far planes of W bound the view frustum V. The projection reference point O_w of the warping frustum is determined by the λ selection in variant reparameterizations. A important assumption we use here is that the movement of the light is confined on the YZ plane. Without using this

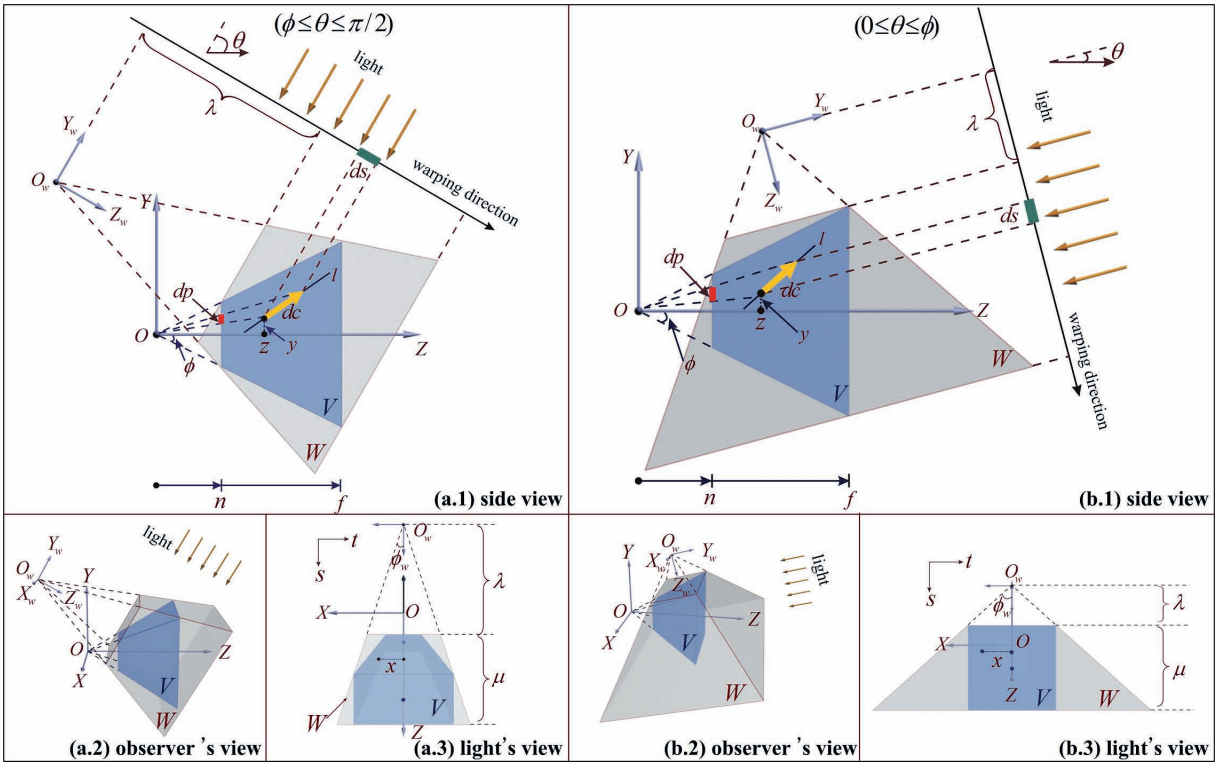

Figure 2: *Construction of the warping frustum W when* $\phi \le \theta \le \pi/2$ *(from **a.1** to **a.3**) and* $0 \le \theta \le \phi$ *(from **b.1** to **b.3**).*

assumption, representing texture coordinates (s,t) as a function of (x,y,z) seems impractical.

We only need to consider the light directions from the upper hemisphere over O due to symmetry. Since θ can be replaced by $\pi-\theta$ in our computation when $\theta > \pi/2$, we suppose $0 \le \theta \le \pi/2$ in our analysis from now on. The shadow-map reparameterization $(t,\ s)$ is induced by applying the perspective projection transform W to both the scene and light. Each point (x,y,z) in the view space is transformed to (x_w,y_w,z_w) in the coordinates system $(O_w;X_w,Y_w,Z_w)$, and then projected to (x_w^c,y_w^c,z_w^c) in W's post-perspective space. Finally, (x_w^c,z_w^c) is normalized into $[0,1]\times[0,1]$ to output the texture coordinates below.

$$\mathbf{s} = (t,\ s) = \left(\frac{1}{2}\frac{x_w}{z_w \tan\phi_w} + \frac{1}{2},\ \frac{\lambda+\mu}{\mu}\left(1-\frac{\lambda}{z_w}\right)\right) \tag{12}$$

Substituting Eq. (12) into Eq. (10) and using the supplementary analysis in **Appendix**, we obtained the following factorization of the aliasing functions.

$$E_q^{\text{pers}} = \frac{2\tan\phi_w}{\tan\phi}\frac{z_w}{z} \quad \text{and} \quad E_q^{\text{proj}} = \left|\frac{k_x - \frac{x}{z}}{k_x - \frac{x}{z_w}\left(\frac{\partial z_w}{\partial y}k_y + \frac{\partial z_w}{\partial z}\right)}\right|$$

$$E_p^{\text{pers}} = \frac{1}{\tan\phi}\frac{\mu}{(\lambda+\mu)\lambda}\frac{z_w^2}{z} \quad \text{and} \quad E_p^{\text{proj}} = \left|\frac{k_y - \frac{y}{z}}{\frac{\partial z_w}{\partial y}k_y + \frac{\partial z_w}{\partial z}}\right|$$

By ignoring projection aliasing errors, as shown in **Appendix**, the aliasing functions can be approximated by

$$E_q \approx E_q^{\text{pers}} = F_q(\lambda,\theta)\frac{z+G(y,\lambda,\theta)}{z} \tag{13}$$

$$E_p \approx E_p^{\text{pers}} = F_p(\lambda,\theta)\frac{\left(z+G(y,\lambda,\theta)\right)^2}{z} \tag{14}$$

Refer to **Appendix** for the meanings of the functions used above. Eqs. (13) and (14) are termed the *global representation* of aliasing errors in perspective reparameterizations. The global representation extends the local representation to all points within the view frustum without requiring the assumptions shown in Eq. (6). Given an arbitrary point (x,y,z), the global representation can quantitatively evaluate the aliasing error at this point. On the contrary, as we mentioned in subsection 2.1, the local representation is only able to quantify the aliasing errors at points with the form $(0,0,z)$. More importantly, the global representation is derived based on an analysis of the general case, such that the aliasing distribution remains analyzable for dynamic lights/viewer.

Aliasing metrics: We considered three kinds of aliasing metrics: $E_p(z,y,\lambda,\theta)$, $E_q(z,y,\lambda,\theta)$ and $E_{p\times q}(z,y,\lambda,\theta) = E_p \times E_q$. E_p and E_q quantify the aliasing in the vertical and horizontal dimensions of the screen independently, while $E_{p\times q}$ is an aggregate measure of the aliasing errors in both dimensions. In the factorization of each metric, the part of projection aliasing errors is ignored.

2.5. Constraints in perspective reparameterizations

Two major constraints are used in perspective reparameterizations.

Ignoring projection aliasing: The projection aliasing part $E^{\text{proj}}(\mathbf{c},\Theta,\mathbf{l})$ in Eq. (11) is ignored in perspective reparameterizations. This important constraint makes perspective reparameterizations scene-independent and quantitatively analyzable. It is impractical to completely eliminate projection aliasing E^{proj} because it is potentially unbounded. However, as pointed out by Lloyd et al [LTYM06], when reducing the perspective aliasing, the remained projection aliasing becomes less visible for at least two reasons. First, as shown in Fig. 2, when the projection aliasing stretches the texel size $|\mathrm{d}s|$ to $|\mathrm{d}\mathbf{c}|$ on the surface $\mathbf{l}$, the sampling density is reduced only in the stretched direction $\mathrm{d}\mathbf{c}$. Second, high projection aliasing occurs on the surfaces that are nearly parallel with the light direction. For these surfaces, few incoming light beams are reflected to the viewer such that the shadows are not perceptible.

λ **selection:** Given the light-view configuration, essentially the difference among variant perspective reparameterizations is the selection of the only free parameter λ. The applications need to adopt appropriate λ values to satisfying specific constraints. For example, the λ value in PSM, denoted by $\lambda_{\mathbf{PSM}}$, produces a linear aliasing distribution over the depth range in the ideal case [WSP04]. Such "*linear*" constraint usually provides better shadow quality at very near regions. The λ value in LiSPSM, denoted by $\lambda_{\mathbf{LiSPSM}}$, outputs an aliasing distribution "*minimal with respect to* L^∞ *norm*" [WSP04], which empirically works better in both practice and theory. The "*focus-driven*" constraint for the λ value in TSM (denoted by $\lambda_{\mathbf{TSM}}$) requires the user-specified region to occupy the majority of the shadow map resolution. An important note here is that we are *not* intended to claim which one of them is the best in this paper, because each of them can't be suitable for all applications.

3. Application

With the theory established in previous sections, we are now ready to analyze aliasing errors in variant reparameterizations. As a direct application of our theory, in this section three representative perspective reparameterizations including SSMs (keep in mind SSM is a special perspective reparameterization in which $\lambda=\infty$), LiSPSMs and TSMs are used to exemplify the applications of our aliasing theory.

Fig. 3 visualizes the distributions of errors within the view frustum at different θ values in these techniques. For each combination of θ and the algorithm in Fig. 3, a simple scene comprised of 25 even-spaced dragon models is used to illustrate the associated shadow map and shadow details. We emphasize an important note here again, we are *not* intended to draw a simple conclusion of which algorithm is the best, because obviously each algorithm has its own advantages and disadvantages in different types of scenes. The associated shadow map and shadow details are used for illustration only. The remainder of this section explains Fig. 3 in different aspects.

3.1. Extrema of Errors

As explained by Loyd et all in [LTYM06], the extrema of errors especially the maximum error give us a scene-independent measure of the "average" shadow quality. This subsection thus studies where the extrema of errors exhibit.

For given θ values, a simple analysis of Eqs. (13) and (14) gives several interesting observations. First, E_p, E_q and $E_{p\times q}$ increase as y becomes small, which means that the extrema of the three aliasing metrics are achieved on the side planes of the view frustum. Second, E_q increases as z decreases. E_p and $E_{p\times q}$ achieve the extrema when z gets to the boundaries of the depth range $[n,f]$. The λ value determines whether the maximum is achieved at $z=n$ or $z=f$. Therefore, for given θ and λ values, we have

$$\begin{aligned}
\arg\min_{(y,z)\in V} E &= (z=f,\ y=f\tan\phi)\\
\arg\max_{(y,z)\in V} E_q &= (z=n,\ y=-n\tan\phi)\\
\arg\max_{(y,z)\in V} E_p &= (z=n \text{ or } f,\ y=-z\tan\phi)\\
\arg\max_{(y,z)\in V} E_{p\times q} &= (z=n \text{ or } f,\ y=-z\tan\phi)
\end{aligned}$$

Where $E\in\{E_p,E_q,E_{p\times q}\}$. Notice that the maximum and minimum of E_q within the view frustum always occur at the bottom left and top right regardless of the λ selection.

A practical issue when implementing perspective reparameterizations is how to determine the appropriate warping strength (i.e. λ). Obviously, there's no such criteria that can suit all cases. The warping strength needs to be carefully tuned according to the scene model in practice. As an example, let's consider a scene in which most objects locate on or below the view plane (e.g. the scene used in our illustration). Since the maximum of the error $\max E_{p\times q}$ is achieved at the lower near or far corners in Fig. 3, minimizing such maximum errors requires a relatively smaller λ value. Since $\lambda_{\mathbf{TSM}}<\lambda_{\mathbf{LiSPSM}}$ at $\theta=90^\circ,60^\circ$, we thus know TSMs are more suitable for this type of scenes. This is clearly shown by the visualization of shadow maps in Fig. 3, in which more available shadow map resolution is used in TSMs at $\theta=90^\circ,60^\circ$.

3.2. Degenerate Case

As we can tell from Fig. 2, the view frustum is a quad from the light's point of view when $\theta\le\phi$. In this degenerate case, the warping is usually more effective over the eye's near/far plane than the eye's view plane (i.e. XZ plane in Fig. 3). Particularly, in the *dueling frusta* case $\theta=0$, all perspective reparamterizations do nothing with reducing the error E_p resulted from the eye's perspective projection. Even worse, the warping along the vertical direction in this case causes the

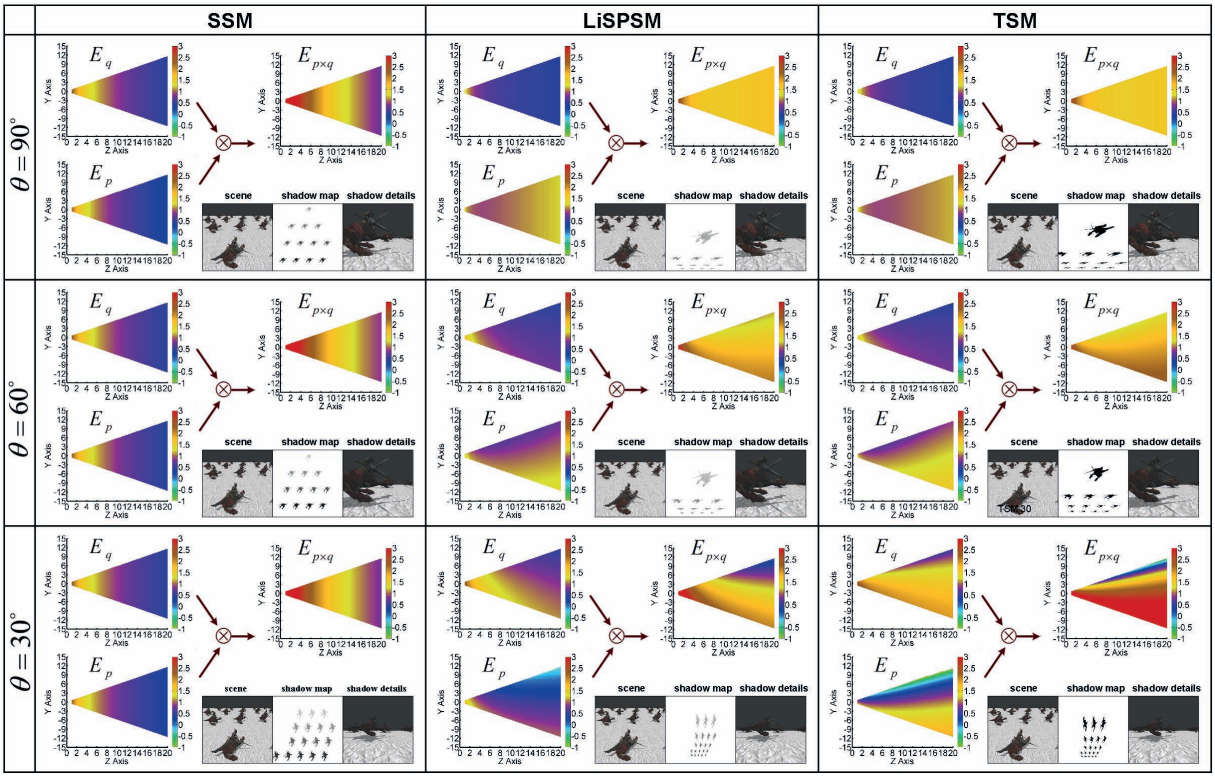

Figure 3: *From left to right: the side views of aliasing distributions within the view frustum in SSMs, LiSPSMs and TSMs respectively. From top to bottom: the side views of aliasing distributions within the view frustum at different* θ *values. A sample scene is used to illustrate the corresponding shadow map and shadow quality for each. The values of the aliasing errors are colorized using the same color-bar in which the red color corresponds the highest errors. The plots use a* $\log_5$ *scale.* $n = 1, f = 100, \phi = \pi/6$. *Refer to* ***http://www.cse.cuhk.edu.hk/∼fzhang/tpcg09*** *for a larger version of this image.*

unexpected stretching of shadow details on the image plane. Therefore, all perspective reparameterizations have no particular advantages over SSMs when $\theta \leq \phi$. This issue is also illustrated by Fig. 3.

We now consider the last row (i.e. $\theta = 30^\circ$) in Fig. 3. Redistributing E_p, E_q and $E_{p\times q}$ is more effective in the vertical direction in both LiSPSMs and TSMs. Therefore, errors are smaller in the upper half frustum and larger in the lower half frustum. This phenomena is particularly noticeable in $E_{p\times q}$ for TSMs. If most visible objects in the scene locate on or below the view plane (e.g. scene models used in typical driving games), in theory LiSPSMs can produce better quality in this case.

4. Conclusion

In this paper, we have presented an aliasing theory of shadow mapping. A generalized representation of aliasing errors are derived from solid mathematical derivations.Given the reparameterization, this aliasing representation explicitly measures how the aliasing errors are distributed within the view frustum in the general case. As a direct application of our aliasing theory, the aliasing distributions in a few representative perspective reparameterizations are evaluated. For the future work, two main direction will be investigated: 1) further improving our theory to adapt point lights; 2) developing direction-adaptive reparameterizations by using the aliasing theory.

References

[AA04] ARVO J., AILA T.: Optimized shadow mapping using the stencil buffer. *Journal of Graphics Tools 8*, 3 (2004), 23–32. 2

[AAM04] AILA T., AKENINE-MÖLLER T.: A hierarchical shadow volume algorithm. In *Proceedings of Graphics Hardware 2004* (2004), Eurographics Association, pp. 15–23. 2

[AMS*08] ANNEN T., MERTENS T., SEIDEL H.-P., FLERACKERS E., KAUTZ J.: Exponential shadow maps. In *Graphics Interface* (2008), 0002 C. S., Bartram L., (Eds.), ACM International Conference Proceeding Series, ACM Press, pp. 155–161. 2

[Arv04] ARVO J.: Tiled shadow maps. In *Proceedings of Computer Graphics International 2004* (2004), IEEE Computer Society, pp. 240–247. 2

[CD04] CHAN E., DURAND F.: An efficient hybrid shadow rendering algorithm. In *Proceedings of the Eurographics Symposium on Rendering* (2004), Eurographics Association, pp. 185–195. 2

[CG04] CHONG H. Y., GORTLER S. J.: A lixel for every pixel. In *the Eurographics Symposium on Rendering 2004* (2004), Eurographics, Eurographics Association. 2

[Cro77] CROW F. C.: Shadow algorithms for computer graphics. In *Proceedings of SIGGRAPH '77* (New York, NY, USA, 1977), ACM Press, pp. 242–248. 1

[DL06] DONNELLY W., LAURITZEN A.: Variance shadow maps. In *Proceedings of the 2006 Symposium on Interactive 3D Graphics, SI3D 2006,* (Redwood City, California, USA, march 2006), ACM, pp. 161–165. 2

[FFBG01] FERNANDO R., FERNANDEZ S., BALA K., GREENBERG D. P.: Adaptive shadow maps. In *Proceedings of SIGGRAPH '01* (New York, NY, USA, 2001), ACM Press, pp. 387–390. 2

[GLY*03] GOVINDARAJU N. K., LLOYD B., YOON S.-E., SUD A., MANOCHA D.: Interactive shadow generation in complex environments. *ACM Trans. Graph. 22*, 3 (2003), 501–510. 2

[Koz04] KOZLOV S.: Perspective shadow maps: care and feeding. *GPU Gems* (2004), 217–244. 1

[LGT*06] LLOYD B., GOVINDARAJU N. K., TUFT D., MOLNAR S., MANOCHA D.: Practical logarithmic shadow maps. In *SIGGRAPH '06: ACM SIGGRAPH 2006 Sketches* (New York, NY, USA, 2006), ACM Press, p. 103. 4

[LTYM06] LLOYD B., TUFT D., YOON S., MANOCHA D.: Warping and partitioning for low error shadow maps. In *Proceedings of the Eurographics Symposium on Rendering 2006* (2006), Eurographics Association, pp. 215–226. 1, 2, 3, 6

[MT04] MARTIN T., TAN T.-S.: Anti-aliasing and continuity with trapezoidal shadow maps. In *the Eurographics Symposium on Rendering 2004* (2004), Eurographics, Eurographics Association. 1

[RSC87] REEVES W. T., SALESIN D. H., COOK R. L.: Rendering antialiased shadows with depth maps. In *Proceedings of SIGGRAPH '87* (New York, NY, USA, 1987), ACM Press, pp. 283–291. 2

[Sal08] SALVI M.: Rendering filtered shadows with exponential shadow maps. *ShaderX6* (2008), 257–274. 2

[SCH03] SEN P., CAMMARANO M., HANRAHAN P.: Shadow silhouette maps. *ACM Trans. Graph. 22*, 3 (2003), 521–526. 2

[SD02] STAMMINGER M., DRETTAKIS G.: Perspective shadow maps. In *Proceedings of SIGGRAPH '02* (New York, NY, USA, 2002), ACM Press, pp. 557–562. 1, 3

[TQJN01] TADAMURA K., QIN X., JIAO G., NAKAMAE E.: Rendering optimal solar shadows with plural sunlight depth buffers. *The Visual Computer 17*, 2 (2001), 76–90. 2

[Wil78] WILLIAMS L.: Casting curved shadows on curved surfaces. In *Proceedings of SIGGRAPH '78* (New York, NY, USA, 1978), ACM Press, pp. 270–274. 1

[WPF90] WOO A., POULIN P., FOURNIER A.: A survey of shadow algorithms. *IEEE Comput. Graph. Appl. 10*, 6 (1990), 13–32. 2

[WSP04] WIMMER M., SCHERZER D., PURGATHOFER W.: Light space perspective shadow maps. In *the Eurographics Symposium on Rendering 2004* (2004), Eurographics, Eurographics Association. 1, 2, 3, 4, 6

[ZSXL06] ZHANG F., SUN H., XU L., LUN L. K.: Parallel-split shadow maps for large-scale virtual environments. In *VRCIA '06: Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications* (New York, NY, USA, 2006), ACM Press, pp. 311–318. 2

[ZSZW08] ZHANG F., SUN H., ZHAO C., WANG L.: Generalized minimum-norm perspective shadow maps. *Comput. Animat. Virtual Worlds 19*, 5 (2008), 553–567. 1, 2, 3

[ZXTS06] ZHANG F., XU L., TAO C., SUN H.: Generalized linear perspective shadow map reparameterization. In *VRCIA '06: Proceedings of the 2006 ACM international conference on Virtual reality continuum and its applications* (New York, NY, USA, 2006), ACM Press, pp. 339–342. 2, 3

Appendix

Derivation of Eqs. (13) and (14): In Fig. 2, the light views (right column) tells us

$$x_w = x \quad \text{and} \quad \tan\phi_w = \begin{cases} \frac{f\tan\phi}{\mu+\lambda} & \theta \in [\phi, \frac{\pi}{2}] \\ \frac{f\tan\phi}{\lambda} & \theta \in [0, \phi] \end{cases}$$

The side views (middle column) gives

$$\mu = \begin{cases} (n\frac{\tan\phi}{\tan\theta} + f - n + f\frac{\tan\phi}{\tan\theta})\sin\theta & \theta \in [\phi, \frac{\pi}{2}] \\ 2f\tan\phi\cos\theta & \theta \in [0, \phi] \end{cases}$$

and

$$z_w = \begin{cases} (z - n + n\frac{\tan\phi}{\tan\theta} - \frac{y}{\tan\theta})\sin\theta + \lambda & \theta \in [\phi, \frac{\pi}{2}] \\ \Big(f\tan\phi - (f-z)\tan\theta - y\Big)\cos\theta + \lambda & \theta \in [0, \phi] \end{cases}$$

The explicit expressions of (t,s) are then obtained by substituting the above x_w, μ and z_w into Eq. (12). Finally we derive the following factorizations of aliasing functions using Eq. (10).

$$E_q = E_q^{\text{pers}} \times E_q^{\text{proj}} = \left(\frac{2\tan\phi_w}{\tan\phi}\frac{z_w}{z}\right) \times \left(\frac{k_x - \frac{x}{z}}{k_x - \frac{x}{z_w}\frac{\partial z_w}{\partial y}k_y - \frac{x}{z_w}\frac{\partial z_w}{\partial z}}\right)$$

$$E_p = E_p^{\text{pers}} \times E_p^{\text{proj}} = \left(\frac{1}{\tan\phi}\frac{\mu}{(\lambda+\mu)\lambda}\frac{z_w^2}{z}\right) \times \left(\frac{k_y - \frac{y}{z}}{\frac{\partial z_w}{\partial y}k_y + \frac{\partial z_w}{\partial z}}\right)$$

To facilitate our analysis of perspective aliasing, we introduce two auxiliary functions F and G to further factorize E_q^{pers} and E_p^{pers} into

$$E_q^{\text{pers}} = F_q(\lambda,\theta)\frac{z + G(y,\lambda,\theta)}{z}$$

and

$$E_p^{\text{pers}} = F_p(\lambda,\theta)\frac{\big(z + G(y,\lambda,\theta)\big)^2}{z}$$

where

$$G(y,\lambda,\theta) = \begin{cases} \frac{\lambda}{\sin\theta} - n + \frac{n\tan\phi - y}{\tan\theta} & \theta \in [\phi, \frac{\pi}{2}] \\ \frac{\lambda}{\sin\theta} - f + \frac{f\tan\phi - y}{\tan\theta} & \theta \in [0, \phi] \end{cases}$$

$$F_q(\lambda,\theta) = \begin{cases} \frac{2f\sin\theta}{\lambda + (f-n)(1 - \frac{\tan\phi}{\tan\theta})\sin\theta} & \theta \in [\phi, \frac{\pi}{2}] \\ \frac{2f\sin\theta}{\lambda} & \theta \in [0, \phi] \end{cases}$$

$$F_p(\lambda,\theta) = \begin{cases} \frac{\sin^2\theta\Big(f - n + (f+n)\frac{\tan\phi}{\tan\theta}\Big)}{\tan\phi\lambda\Big(\lambda + \Big(f - n + (f+n)\frac{\tan\phi}{\tan\theta}\Big)\sin\theta\Big)} & \theta \in [\phi, \frac{\pi}{2}] \\ \frac{2f\sin^2\theta}{\lambda(\lambda + 2f\tan\phi\cos\theta)\tan\phi} & \theta \in [0, \phi] \end{cases}$$ ■