Variance Shadow Maps.(方差阴影图。 )

William Donnelly[[1]](#footnote-1) Andrew Lauritzen†.(威廉姆·唐纳利·安德鲁·劳里岑 )

Computer Graphics Lab, School of Computer Science, University of Waterloo.(滑铁卢大学计算机科学学院计算机图形学实验室。 )

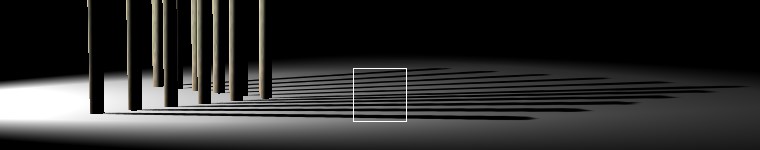
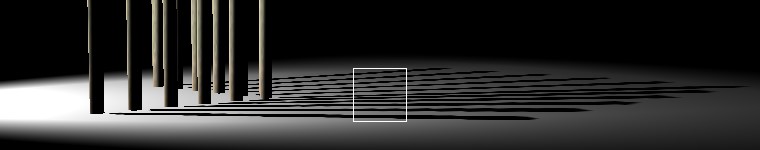


Figure 1: Comparison of anisotropic filtering vs.(图1:各向异性滤波与 ) no anisotropic filtering.(无各向异性滤波。 ) Top: Regular shadow map with bilinear percentage closer filtering.(上图:规则阴影图，双线性百分比更接近过滤。 ) Bottom: Variance shadow map with mipmapping and 16x anisotropic filtering.(下图:方差阴影图，带MIP映射和16x各向异性滤波。 )

# Abstract.(摘要。 )

Shadow maps are a widely used shadowing technique in real time graphics.(阴影映射是实时图形中广泛使用的一种阴影技术。 ) One major drawback of their use is that they cannot be filtered in the same way as color textures, typically leading to severe aliasing.(它们使用的一个主要缺点是不能以与颜色纹理相同的方式对它们进行过滤，这通常会导致严重的混叠。 ) This paper introduces *variance shadow maps*, a new real time shadowing algorithm.(在实时图形中IDELY使用阴影技术。 它们使用的一个主要缺点是不能以与颜色纹理相同的方式对它们进行过滤，这通常会导致严重的混叠。 本文介绍了一种新的实时阴影算法——方差阴影图。 ) Instead of storing a single depth value, we store the mean and mean squared of a distribution of depths, from which we can efficiently compute the variance over any filter region.(不是存储单个深度值，而是存储深度分布的均值和均值平方，从中我们可以有效地计算任何滤波区域上的方差。 ) Using the variance, we derive an upper bound on the fraction of a shaded fragment that is occluded.(机翼算法 不是存储单个深度值，而是存储深度分布的均值和均值平方，从中我们可以有效地计算任何滤波区域上的方差。 利用方差，我们推导出遮挡的阴影碎片分数的上界。 ) We show that this bound often provides a good approximation to the true occlusion, and can be used as an approximate value for rendering.(我们证明，这个界限通常提供了一个很好的近似真实遮挡，并可以作为一个近似值用于渲染。 ) Our algorithm is simple to implement on current graphics processors and solves the problem of shadow map aliasing with minimal additional storage and computation.(指遮挡的阴影碎片。 我们证明，这个界限通常提供了一个很好的近似真实遮挡，并可以作为一个近似值用于渲染。 该算法在现有的图形处理器上实现简单，并以最少的额外存储和计算量解决了阴影映射的混叠问题。 )

**CR Categories:** I.(CR类别:I。 )3.(3。 )7 [Computer Graphics]: ThreeDimensional Graphics and Realism—Color, shading, shadowing, and texture.([计算机图形学]:三维图形和真实感-颜色，阴影，阴影和纹理。 )

**Keywords:** real-time rendering, shadow maps, shader programming, graphics hardware.(关键词:实时渲染，阴影图，着色器编程，图形硬件。 )

# Introduction.(导言。 )

Shadow maps [Williams 1978] and shadow volumes [Crow 1977] are the two most common shadowing algorithms used in real time applications.(阴影映射[Williams1978]和阴影体积[Crow1977]是实时应用中最常用的两种阴影算法。 ) Shadow maps have many advantages compared to shadow volumes; for example they are easy to implement, their cost is less sensitive to geometric complexity and they can be queried at arbitrary locations.(阴影映射[Williams1978]和阴影体积[Crow1977]是实时应用中最常用的两种阴影算法。 与阴影卷相比，阴影映射有许多优点； 例如，它们易于实现，成本对几何复杂性不太敏感，并且可以在任意位置进行查询。 )

Unfortunately, like most textures, shadow maps suffer from aliasing if not filtered properly.(不幸的是，像大多数纹理，阴影地图遭受别名，如果没有适当的过滤。 ) Modern graphics hardware provides built-in methods to reduce texture aliasing on color textures: namely mipmapping and anisotropic filtering.(现代图形硬件提供了内置的方法来减少纹理在彩色纹理上的混叠:即MIP映射和各向异性滤波。 ) These techniques are inapplicable to standard shadow maps, since they will simply interpolate the depths of neighboring pixels.(不幸的是，像大多数纹理，阴影地图遭受别名，如果没有适当的过滤。 现代图形硬件提供了内置的方法来减少纹理在彩色纹理上的混叠:即MIP映射和各向异性滤波。 这些技术不适用于标准阴影图，因为它们只是内插相邻像素的深度。 ) Typically real time implementations use nearest-neighbor sampling of shadow maps, or take several samples and average the results.(通常，实时实现使用阴影图的最近邻采样，或者采取几个采样并对结果进行平均。 ) This method is expensive, causes aliasing and does not take full advantage of graphics hardware’s aforementioned fast, built-in filtering capabilities.(不适用于标准阴影图，因为它们只是内插相邻像素的深度。 通常，实时实现使用阴影图的最近邻采样，或者采取几个采样并对结果进行平均。 这种方法代价昂贵，造成混叠，并且没有充分利用图形硬件的上述快速、内置的过滤功能。 )

To address the problem of efficiently filtering shadow maps, we note that each texel of a standard shadow map can only represent the depth of a single point.(为了解决有效过滤阴影映射的问题，我们注意到标准阴影映射的每个纹理只能表示单个点的深度。 ) Variance shadow maps improve on this scheme by representing a *distribution* of depths at each texel.(呈现单个点的深度。 方差阴影图通过表示每个纹理的深度分布来改进该方案。 ) To approximate such a distribution using a small amount of data, we store the first and second moments: the mean depth and mean squared depth.(每个纹理的深度。 为了使用少量数据来近似这样的分布，我们存储第一和第二矩:平均深度和平均平方深度。 ) One major advantage of this representation is that we can approximate the average of two distributions by averaging the moments.(这种表示的一个主要优点是，我们可以通过求矩的平均值来近似两个分布的平均值。 )

When querying the variance shadow map, we use the moments to compute a bound on the fraction of the distribution that is more distant than the surface being shaded.(当查询方差阴影映射时，我们使用矩来计算比被阴影化的表面更远的分布的分数的界。 ) We show that this bound provides a good approximation for the amount of light reaching any given surface, and therefore can be used for rendering correctly anti-aliased shadows.(我们证明，这个界限提供了一个很好的接近量的光到达任何给定的表面，因此可以用于渲染正确的反锯齿阴影。 )

Because the moments can be interpolated, we can make use of the wide range of filtering techniques that are available for color textures, effectively eliminating aliasing.(由于矩可以插值，我们可以利用广泛的过滤技术，可用于彩色纹理，有效地消除混叠。 )

* Reduces aliasing on shadow maps by enabling the use of filtering techniques such as mipmapping and anisotropic filtering,.(通过使用MIP映射和各向异性滤波等滤波技术，减少阴影图上的混叠。 )
* Allows shadow maps to be pre-filtered for percentage closer filtering, and.(允许对阴影映射进行预筛选，以便进行百分比更接近的筛选，以及。 )
* Can be implemented on current graphics hardware at a cost comparable to that of ordinary shadow maps.(可以在当前图形硬件上以与普通阴影图相当的成本实现。 )

# Related Work.(相关工作。 )

Williams introduced shadow maps [Williams 1978] as an efficient algorithm for computing shadows in general scenes.(Williams提出了阴影映射[Williams1978]，它是一种计算一般场景中阴影的有效算法。 ) However, he points out that the usual filtering techniques for color textures cannot be applied to depth values.(然而，他指出，通常的颜色纹理过滤技术不能应用于深度值。 )

Percentage closer filtering [Reeves et al.(百分比更接近过滤[Reeves et al。 ) 1987] provides a solution to the problem of shadow map aliasing.(百分比逼近滤波[Reeves et al.1987]为阴影图混叠问题提供了一种解决方案。 ) The key insight is that a correct filtering algorithm needs to filter the *results* of the depth comparisons, instead of filtering the depths.(阴影映射别名问题。 关键的洞察是，正确的过滤算法需要过滤深度比较的结果，而不是过滤深度。 ) This is accomplished by randomly sampling the shadow map, so many samples are required to eliminate noise.(而不是过滤深度。 这是通过对阴影图进行随机采样来实现的，因此需要许多采样来消除噪声。 )

Deep shadow maps [Lokovic and Veach 2000] store a distribution of depths instead of a single depth at each pixel.(深阴影地图[Lokovic和Veach2000]存储深度分布，而不是每个像素的单个深度。 ) As a result, percentage closer filtering can be done as a pre-process.(因此，百分比更接近过滤可以做为一个预处理。 ) Subsequently, each query requires a constant amount of work independent of the filter size.(深阴影地图[Lokovic和Veach2000]存储深度分布，而不是每个像素的单个深度。 因此，百分比更接近过滤可以做为一个预处理。 随后，每个查询都需要与筛选器大小无关的固定工作量。 ) However, deep shadow maps are not particularly amenable to implementation on graphics hardware for two reasons: each pixel must encode a piecewise linear function using a large amount of data, and the procedure for averaging two distributions is non-trivial.(t与过滤器大小无关的工作量。 然而，由于两个原因，深度阴影图不特别适合在图形硬件上实现:每个像素必须使用大量数据来编码分段线性函数，并且平均两个分布的过程是不平凡的。 )

Opacity shadow maps [Kim and Neumann 2001] encode a.(不透明度阴影图[Kim和Neumann2001]编码a。 )

distribution of depths, but use a fixed amount of storage per pixel.(深度分布，但每个像素使用固定的存储量。 ) Each pixel stores a function just as in deep shadow maps, but instead the function is constructed from its values at a fixed set of points.(深度分布，但每个像素使用固定的存储量。 每个像素都存储一个函数，就像在深度阴影图中一样，但该函数是由固定点集的值构造而成的。 ) Thus the algorithm can take advantage of the speed of graphics hardware.(因此，该算法可以充分利用图形硬件的速度优势。 ) The price to pay for this flexibility is extreme quantization of depths, so the technique is mostly suitable for rendering of dense volumetric objects such as hair and fur.(由其在固定点集处的值构造而成。 因此，该算法可以充分利用图形硬件的速度优势。 这种灵活性的代价是深度的极端量化，因此该技术最适合于呈现稠密的体积对象，如头发和毛皮。 )

An alternative approach to the problem of shadow map aliasing is to alter the shadow map projection.(另一种解决阴影图混叠问题的方法是改变阴影图投影。 ) This avenue was pursued in adaptive shadow maps [Fernando et al.(这一途径在自适应阴影图中得到了应用[Fernando等人。 ) 2001], perspective shadow maps [Stamminger and Drettakis 2002], light space perspective shadow maps [Wimmer et al.(另一种解决阴影图混叠问题的方法是改变阴影图投影。 这一途径在自适应阴影图[Fernando等人，2001年]，透视阴影图[Stamminger和Drettakis，2002年]，光空间透视阴影图[Wimmer等人，2002年]中得到了推广。 ) 2004] and trapezoidal shadow maps [Martin and Tan 2004].(2004年]和梯形阴影图[Martin和Tan，2004年]。 ) Because we make no assumptions about the shadow projection, these approaches are compatible with variance shadow maps; in fact, the two approaches are complementary.(因为我们不对阴影投影做任何假设，这些方法与方差阴影映射是兼容的； 事实上，这两种方法是相辅相成的。 )

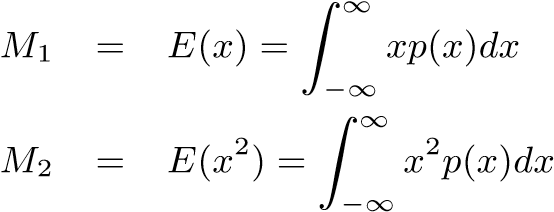
Some graphics hardware has native support for shadow maps, for example through the OpenGL ”GL ARB shadow” extension.(一些图形硬件具有对阴影映射的本机支持，例如通过OpenGL“GL ARB Shadow”扩展。 ) However, the extension does not specify behaviour with respect to interpolation or mipmapping.(但是，扩展并没有指定与插值或MIP映射有关的行为。 ) Some NVIDIA graphics processors support bilinear filtering [Everitt et al.(PHICS硬件具有对阴影映射的本机支持，例如通过OpenGL“GL ARB Shadow”扩展。 但是，扩展并没有指定与插值或MIP映射有关的行为。 一些NVIDIA图形处理器支持双线性滤波[Everitt et al。 ) 2000], but to our knowledge, no graphics hardware supports trilinear or anisotropic filtering on shadow maps.(2000]，但据我们所知，没有图形硬件支持阴影图上的三线性或各向异性滤波。 )

# Algorithm Overview.(算法概述。 )

As with conventional shadow mapping, we first render the scene from the light’s point of view.(与传统的阴影映射一样，我们首先从光线的角度渲染场景。 ) For shadow mapping, we would render the depth as seen from the light; for variance shadow maps we render into a two-channel buffer, rendering both the depth and the square of the depth.(与传统的阴影映射一样，我们首先从光线的角度渲染场景。 对于阴影映射，我们将呈现从光中看到的深度； 对于方差阴影映射，我们渲染成一个双通道缓冲区，渲染深度和深度的平方。 ) Although in regular shadow mapping we would not want to use any type of anti-aliasing when rendering from the light’s point of view, anti-aliasing will actually be of benefit when rendering variance shadow maps.(W映射，我们将渲染从光中看到的深度； 对于方差阴影映射，我们渲染成一个双通道缓冲区，渲染深度和深度的平方。 虽然在常规阴影映射中，我们不希望在从光的角度呈现时使用任何类型的抗锯齿，但抗锯齿实际上在呈现方差阴影映射时是有益的。 )

Once we have created the shadow map, we can do preprocessing on the texture to facilitate filtering.(一旦我们创建了阴影图，我们就可以对纹理进行预处理，以便于过滤。 ) This can include generating mipmaps [Williams 1983] or computing summed area tables [Crow 1984].(这可以包括生成MIPMAP[Williams1983]或计算总和面积表[Crow1984]。 ) To further reduce aliasing and soften shadow edges we can also blur the variance shadow map.(一旦我们创建了阴影图，我们就可以对纹理进行预处理，以便于过滤。 这可以包括生成MIPMAP[Williams1983]或计算总和面积表[Crow1984]。 为了进一步减少混叠和软化阴影边缘，我们还可以模糊方差阴影映射。 )

Since we have rendered depth and squared depth in the texture, the result of filtering our texture will be to recover the moments *M*1 and *M*2 over that filter region, defined as follows:.(由于我们已经渲染了纹理中的深度和平方深度，过滤我们的纹理的结果将是恢复该过滤区域上的矩M1和M2，定义如下:。 )

(1).(（1）。 )

(2).(（2）。 )

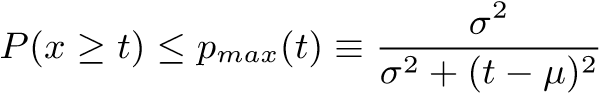
From these we compute the mean *µ* and variance *σ*2:.(由此计算出均值µ、方差σ2:。 )

*µ* = *E*(*x*) = *M*1 (3).(µ=E(x)=M1(3)。 )

 (4).(（4）。 )

The variance can be interpreted as a quantitative measure of the width of a distribution.(方差可以解释为分布宽度的定量度量。 ) As a result, it should place a bound on how much of the distribution can be concentrated far away from the mean.(因此，它应该对分布中有多少可以集中在远离平均值的地方设定一个界限。 ) This bound is stated precisely in Chebyshev’s inequality:.(方差可以解释为分布宽度的定量度量。 因此，它应该对分布中有多少可以集中在远离平均值的地方设定一个界限。 这一界在切比雪夫不等式中得到了精确的表述:。 )

**Theorem 1 (Chebychev’s inequality, one-tailed version)** *Let x be a random variable drawn from a distribution with mean µ and variance σ*2*.(定理1（Chebychev不等式，单尾形式）设X是取自均值µ、方差σ2分布的随机变量。 ) Then for t > µ.(则对于t>µ. )*

 (5).(（5）。 )

The quantity *P*(*x* ≥ *t*) in equation 5 is exactly the quantity we wish to compute in order to perform percentage closer filtering, since it represents the fraction of pixels over a filter region that will fail the depth comparison with a fixed depth *t*.(公式5中的量P（x≥t）正是我们希望计算的量，以便执行百分比逼近滤波，因为它表示滤波器区域上无法与固定深度t进行深度比较的像素比例。 )

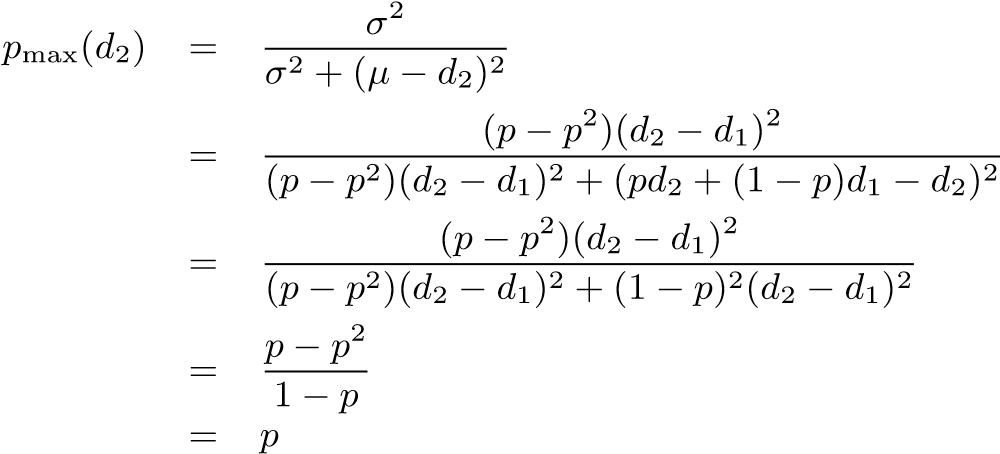
However, equation 5 is only an upper bound; a priori there is no reason to assume it will allow us to compute the true value *P*(*x* ≥ *t*).(然而，等式5只是一个上限； 先验地，没有理由假定它将允许我们计算真实值p(x≥t)。（ ) It can, however, provide a good approximation, as we show in the following example.(但是，它可以提供一个很好的近似值，如下例所示。 )

## Planar Occluders and Receivers.(平面封堵器和接收器。 )

Consider the case of a single planar occluder at depth *d*1, casting a shadow onto a planar surface at depth *d*2.(考虑在深度D1处的单个平面遮挡器的情况，在深度D2处在平面表面上投射阴影。 ) Suppose we have a fixed filter, where *p* is the percentage of the filter that is unoccluded.(假设我们有一个固定的滤波器，其中p是未遮挡滤波器的百分比。 ) Then we have:.(那么我们有:。 )

|  |  |  |
| --- | --- | --- |
| *µ* = *E*(*x*).(µ=E(x)。 ) | =.(=。 ) | *pd*2 + (1 −*p*)*d*1.(PD2(1−P)D1。 ) |
| *E*(*x*2).(E(X2)。 ) | =.(=。 ) |  |
| *σ*2.(σ2。 ) | =.(=。 ) |  |
|  | =.(=。 ) | (*p*−*p*2)(*d*2−*d*1)2.(（p−p2）（d2−d1）2。 ) |

Using these values, we can compute *p*max according to equation 5:.(使用这些值，我们可以根据公式5计算Pmax: )



Thus in this simple situation, we see that Chebyshev’s inequality is an equality, and gives the *exact* result of percentage closer filtering.(因此，在这种简单的情况下，我们看到切比雪夫不等式是一个等式，并给出了百分比逼近滤波的精确结果。 )

Although this is a very particular situation, it actually provides a reasonable approximation to a common situation in many real scenes.(虽然这是一种非常特殊的情况，但它实际上提供了许多真实场景中常见情况的合理近似值。 ) In the case of a single occluder and single receiver, we can take a small neighbourhood in which the depth of the occluder and receiver are approximately constant.(虽然这是一种非常特殊的情况，但它实际上提供了许多真实场景中常见情况的合理近似值。 在单个遮挡器和单个接收器的情况下，我们可以取其中遮挡器和接收器的深度近似恒定的小邻域。 ) In this case, equation 5 will not provide an exact value, but a close approximation.(在这种情况下，等式5将不提供精确值，而是提供近似值。 ) Thus we use *p*max in rendering as an approximation to the true value *p*.(DER和接收器近似为常数。 在这种情况下，等式5将不提供精确值，而是提供近似值。 因此，我们在渲染中使用pmax作为真实值p的近似。 )

# Implementation.(执行情况。 )

We implemented variance shadow maps on a GeForce 6800GT using Sh [McCool and Du Toit 2004] and OpenGL.(我们使用SH[McCool和Du Toit2004]和OpenGL在GeForce6800GT上实现了方差阴影映射。 ) Sh allows us to build the variance shadow mapping shader on top of existing light shaders, and automatically combine it with any of the surface shaders in the scene.(SH允许我们在现有的光着色器之上构建方差阴影映射着色器，并自动将其与场景中的任何表面着色器组合。 )

The GeForce 6 series supports filtering of 16-bit per component floating point textures (fp16), so we use hardware mipmapping and anisotropic filtering.(GeForce6系列支持每个组件16位浮点纹理(FP16)的过滤，因此我们使用硬件MIP映射和各向异性过滤。 ) Our implementation is roughly as follows:.(我们的实现大致如下:。 )

1. Render to a four component fp16 framebuffer object from the light’s point of view[[2]](#footnote-2).(从光线的角度渲染到一个四组件的FP16帧缓冲对象。 ) In the fragment program, output the depth and squared depth of the current fragment to the framebuffer.(。在片段程序中，将当前片段的深度和平方深度输出到framebuffer。 ) We scale the depths to be in the range [0,1] to avoid overflowing the fp16 numeric boundaries.(我们将深度缩放到范围[0，1]内，以避免溢出FP16数值边界。 )
2. Optionally prefilter the shadow map using a two-pass separable gaussian blur.(可选地，使用两次可分离的高斯模糊对阴影映射进行预滤波。 )
3. Have OpenGL automatically generate mipmaps.(让OpenGL自动生成MIPMAP。 )
4. Render the scene normally from the camera’s point of view.(从相机的角度正常渲染场景。 ) In the fragment shader, read the shadow map to get the moments *M*1 and *M*2.(从相机的角度正常渲染场景。 在片段着色器中，读取阴影图以获得时刻M1和M2。 ) If the current fragment has depth *< µ*, then the surface is unshadowed.(。如果当前碎片的深度小于µs，则表面无阴影。 ) Otherwise, compute the variance from the first two moments (equation 4) and scale the light intensity by *p*max (equation 5).(欠的。 否则，计算前两个矩的方差（等式4），并用Pmax缩放光强（等式5）。 )

In addition, we implemented a version using a 32-bit per component shadow map.(此外，我们实现了一个版本使用32位每个组件的阴影映射。 ) Since our hardware does not support filtering at that precision, we implemented bilinear filtering in the fragment shader.(此外，我们实现了一个版本使用32位每个组件的阴影映射。 由于我们的硬件不支持这种精度的过滤，我们在片段着色器中实现了双线性过滤。 ) In principle we could also.(原则上我们也可以。 )

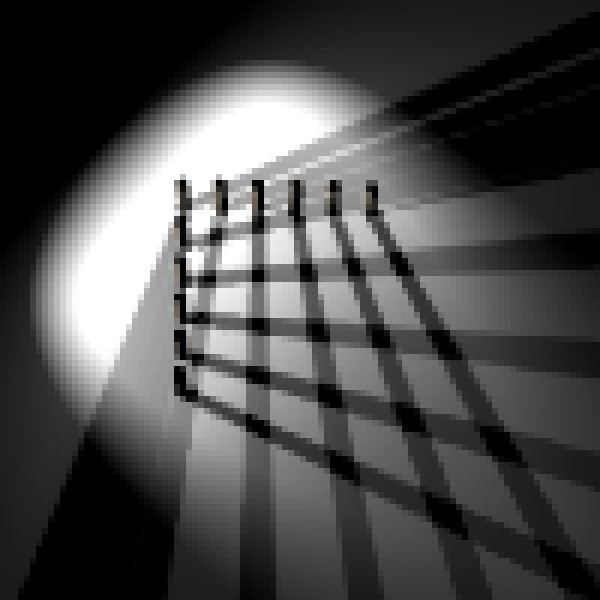
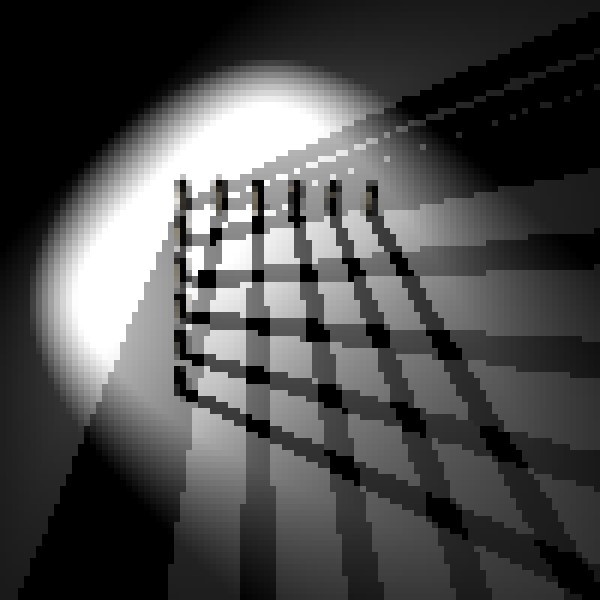


Figure 2: Comparison of variance shadow mapping without mipmapping (left) and with mipmapping (right).(图2:没有MIPmapping的方差阴影映射（左）和有MIPmapping的方差阴影映射（右）的比较。 )



Figure 3: Left to right: 1) standard shadow mapping, 2) 5x5 percentage closer filtering, 3) 5x5 bilinear percentage closer filtering, 4) variance shadow maps with 5x5 separable gaussian blur.(图3:从左到右:1）标准阴影映射，2）5×5百分比更近滤波，3）5×5双线性百分比更近滤波，4）具有5×5可分离高斯模糊的方差阴影映射。 )

emulate mipmapping and anisotropic filtering with shader code; in practice, this would run too slowly on current hardware.(用着色器代码模拟MIP映射和各向异性滤波 实际上，这在当前的硬件上运行得太慢。 )

We also implemented the following alternate shadowing methods for comparison.(我们还实现了以下用于比较的备用阴影方法。 ) Each of them uses a single unfiltered fp16 component to store the occluder depth.(它们中的每一个都使用单个未过滤的FP16组件来存储封堵器深度。 )

1. Simple shadow mapping using a single nearest neighbor depth comparison.(使用单个最近邻深度比较的简单阴影映射。 )
2. Bilinear filtering of the neighboring four depth comparisons.(相邻四个深度比较的双线性滤波。 )
3. Percentage closer filtering with gaussian weights, each sample being a nearest neighbor depth comparison.(使用高斯权重的百分比更接近滤波，每个样本是最近邻深度比较。 )
4. Percentage closer filtering with gaussian weights, each sample being the bilinear filtered result of the neighboring four depth comparisons.(使用高斯权重的百分比逼近滤波，每个样本是相邻四个深度比较的双线性滤波结果。 )

# Results.(结果。 )

Figure 2 shows the result of using mipmapping with variance shadow maps.(图2显示了使用MIPMapping和方差阴影映射的结果。 ) As with color textures, mipmapping effec-.(与彩色纹理一样，MIP映射也会影响。 )

tively reduces the aliasing of the shadow map when viewed from a distance.(当从远处观察时，可以减少阴影图的混叠。 ) Anisotropic filtering can also be used with variance shadow maps, eliminating aliasing that occurs when viewing surfaces at shallow angles (see Figure 1).(各向异性滤波也可用于方差阴影图，消除浅角度观察表面时出现的混叠（见图1）。 )



Figure 4: When the variance over a filter region is high, light bleeding artifacts can occur.(图4:当滤波器区域上的方差较高时，会出现轻微的出血伪影。 ) The circled region of the car’s shadow should be solid black.(汽车阴影的圆圈区域应该是实心黑色的。 ) Note that the contrast has been increased here so that the artifact can be seen more easily.(请注意，这里的对比度增加了，因此可以更容易地看到伪影。 )

Figure 3 shows a side-by-side comparison of the results of using a prefiltered variance shadow map, and an equivalent percentage closer filter.(图3显示了使用预滤波方差阴影图和等效的百分比贴近滤波器的结果的并排比较。 ) As the figure demonstrates, the output of these two methods is almost identical, and vastly superier to simple shadow mapping and standard nearest neighbor percentage closer filtering.(图3显示了使用预滤波方差阴影图和等效的百分比贴近滤波器的结果的并排比较。 如图所示，这两种方法的输出几乎相同，远远优于简单的阴影映射和标准的最近邻百分比逼近滤波。 )

## Light Bleeding.(轻微出血。 )

Our formula for *p*max was derived as a lower bound on the brightness, and although it works well in many situations, it is not guaranteed to be an accurate approximation.(我们导出的Pmax公式是作为亮度的一个下界，虽然它在许多情况下都能很好地工作，但并不能保证它是一个精确的近似值。 ) We can see from equation 5 that whenever the variance *σ*[[3]](#footnote-3) is nonzero, *p*max(*d*) *>* 0 for all *d*.(作为亮度的一个下界，虽然它在许多情况下都能很好地工作，但并不能保证它是一个精确的近似值。 从等式5可以看出，当方差σ为非零时，对所有d，pmax(d)>0。 ) When *σ*2 is small, *p*max goes to zero quickly, and so the effect is not very noticeable.(当σ2较小时，Pmax迅速趋近于零，影响不明显。 ) However, when *σ*2 is large the results can be noticeable as seen in figure 4.(很快就会变成零，所以效果不是很明显。 然而，当σ2较大时，结果会非常明显，如图4所示。 )

In practice, this is a only a problem for scenes with a high depth complexity relative to the shadowed light source.(在实践中，这仅是相对于阴影光源具有高深度复杂性的场景的问题。 ) For scenes with a low depth complexity (eg.(对于具有低深度复杂性的场景（例如， ) many scenes lit by the sun), the artifacts are non-existent or negligible.(许多场景被太阳照亮），这些神器是不存在的或可以忽略不计的。 )

## Performance.(表演。 )

To compare the performance of several shadow mapping implementations we chose a very simple scene with a single spotlight, shown in Figure 1.(为了比较几个阴影映射实现的性能，我们选择了一个非常简单的场景和一个聚光灯，如图1所示。 ) Because all of the methods examined operate in image space, their performance is independent of geometric complexity.(由于所研究的所有方法都是在图像空间中操作的，所以它们的性能与几何复杂性无关。 )

The variance shadow map results used hardware mipmapping, trilinear and 16x anisotropic filtering.(方差阴影图结果使用硬件MIPMAPP、三线性和16x各向异性滤波。 ) Hardware texture filtering is inapplicable to the other algorithms2, and thus any required filtering was implemented in the shader.(方差阴影图结果使用硬件MIPMAPP、三线性和16x各向异性滤波。 硬件纹理过滤不适用于其他算法2，因此任何所需的过滤都在着色器中实现。 )

All measurements are in frames per second, taken on a GeForce 6800GT.(在GeForce6800>上进行的所有测量都以帧/秒为单位。 ) Rendering at a resolution of 1024x768 and varying the shadow map size, the results are as follows:.(在GeForce6800>上进行的所有测量都以帧/秒为单位。 以1024x768的分辨率进行渲染并改变阴影图的大小，结果如下:。 )

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 128x128.(128x128。 ) | 256x256.(256x256。 ) | 512x512.(512x512。 ) | 1024x1024.(1024x1024。 ) |
| Shadow Map.(阴影地图。 ) | 340.(340。 ) | 334.(334。 ) | 314.(314。 ) | 243.(243。 ) |
| Bil.(比尔。 ) PCF 1x1.(PCF1x1。 ) | 224.(224。 ) | 219.(219。 ) | 209.(209。 ) | 175.(175。 ) |
| VSM.(VSM。 ) | 265.(265。 ) | 255.(255。 ) | 228.(228。 ) | 154.(154。 ) |
| PCF 3x3.(PCF3x3。 ) | 153.(153。 ) | 151.(151。 ) | 149.(149。 ) | 130.(130。 ) |
| Bil.(比尔。 ) PCF 3x3.(PCF3x3。 ) | 22.(22。 ) | 22.(22。 ) | 22.(22。 ) | 21.(21。 ) |
| VSM 3x3.(VSM3x3。 ) | 254.(254。 ) | 225.(225。 ) | 154.(154。 ) | 63.(63。 ) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 640x480.(640x480。 ) | 800x600.(800x600。 ) | 1024x768.(1024x768。 ) | 1280x960.(1280x960。 ) |
| Shadow Map.(阴影地图。 ) | 470.(470。 ) | 470.(470。 ) | 314.(314。 ) | 210.(210。 ) |
| Bil.(比尔。 ) PCF 1x1.(PCF1x1。 ) | 457.(457。 ) | 309.(309。 ) | 209.(209。 ) | 133.(133。 ) |
| VSM.(VSM。 ) | 450.(450。 ) | 334.(334。 ) | 228.(228。 ) | 159.(159。 ) |
| PCF 3x3.(PCF3x3。 ) | 336.(336。 ) | 229.(229。 ) | 149.(149。 ) | 92.(92。 ) |
| Bil.(比尔。 ) PCF 3x3.(PCF3x3。 ) | 53.(53。 ) | 33.(33。 ) | 22.(22。 ) | 15.(15。 ) |
| VSM 3x3.(VSM3x3。 ) | 230.(230。 ) | 195.(195。 ) | 154.(154。 ) | 119.(119。 ) |

We now fix the shadow map size at 512x512 and vary the framebuffer resolution:.(现在，我们将阴影映射大小固定为512x512，并改变帧缓冲区分辨率:。 )

Notice that the PCF results do not vary greatly with shadow map size, but rather by the number of onscreen pixels.(请注意，PCF结果并不随阴影映射大小而有很大变化，而是随屏幕上像素的数量而变化。 ) By contrast, the VSM results scale with the framebuffer resolution at the same rate as standard shadow mapping.(相反，VSM结果以与标准阴影映射相同的速率以帧缓冲区分辨率缩放。 ) However, when using a prefilter such as in the VSM 3x3 case, the variance shadow mapping performance scales with the shadow map size.(请注意，PCF结果并不随阴影映射大小而有很大变化，而是随屏幕上像素的数量而变化。 相反，VSM结果以与标准阴影映射相同的速率以帧缓冲区分辨率缩放。 然而，当使用诸如在VSM3x3情况下的预滤波器时，方差阴影映射性能与阴影映射大小成比例。 )

## Numerical Stability.(数值稳定性 )

Equation 4 for computing the variance is known to be numerically unstable when using floating point arithmetic.(当使用浮点算术时，计算方差的公式4在数值上是不稳定的。 ) This is because when the variance is small compared to the average depths we have *E*(*x*2) ≈ *E*(*x*)2 and so we are subtracting two approximately equal large numbers, potentially resulting in loss of precision.(当使用浮点算术时，计算方差的公式4在数值上是不稳定的。 这是因为当方差与平均深度相比较小时，E(x2)≈E(x)2，因此减去两个大致相等的大数，可能会导致精度损失。（实习编辑:顾萍）（完） ) In practice, we found that artifacts were sometimes visible when using a 16-bit floating point shadow map, but never with 32-bit floating point values.(因此我们减去两个大致相等的大数，可能导致精度损失。 实际上，我们发现在使用16位浮点阴影映射时，工件有时是可见的，但从不使用32位浮点值。 ) The fp16 artifacts can be largely eliminated by splitting each 32-bit float into two 16-bit floats for storage (since we are currently forced by the hardware to use a four component texture anyways) and recombining them after the texture lookup.(16位浮点阴影映射，但从不使用32位浮点值。 通过将每个32位浮点拆分为两个16位浮点以进行存储（因为硬件当前强制我们无论如何都要使用四个组件纹理）并在纹理查找之后重新组合它们，可以在很大程度上消除FP16伪影。 ) Moreover, we expect future hardware to support mipmapping and anisotropic filtering of 32-bit floating point textures and thus resolve this issue.(此外，我们希望未来的硬件能够支持32位浮点纹理的MIP映射和各向异性滤波，从而解决这个问题。 )

# Conclusions.(结论。 )

We have introduced variance shadow maps as a simple and effective solution to the problem of aliasing in shadow maps.(我们引入方差阴影映射作为阴影映射中混叠问题的一种简单而有效的解决方案。 ) Our results are based on an upper bound on the result of percentage closer filtering based on the mean and variance of a distribution of depths, which we showed provides a good approximation to percentage closer filtering.(我们引入方差阴影映射作为阴影映射中混叠问题的一种简单而有效的解决方案。 我们的结果是基于基于深度分布均值和方差的百分比更接近滤波结果的上界，这为百分比更接近滤波提供了很好的近似。 ) Finally, we have shown that variance shadow maps can be implemented easily on modern graphics hardware and compare favorably in both performance and quality to existing real time techniques for shadow map filtering.(EPTHS，我们展示了一个很好的近似百分比更接近的过滤。 最后，我们证明了方差阴影图可以很容易地在现代图形硬件上实现，并且在性能和质量上都优于现有的阴影图实时滤波技术。 )

# Acknowledgements.(致谢。 )

We would like to thank Michael McCool and Stefanus Du Toit for their helpful suggestions, and Chris Iacobucci for the car model.(我们要感谢迈克尔麦库尔和斯蒂芬纳斯杜托伊特的有益建议，以及克里斯亚科布奇的汽车模型。 )

# References.(推荐信。 )

Crow, F.(乌鸦，F。 ) 1977.(乌鸦，1977年成立。 ) Shadow algorithms for computer graphics.(1977年。 计算机图形学中的阴影算法 ) In *Computer Graphics (Proc.(。在计算机图形学中（过程。 ) SIGGRAPH)*, vol.(计算机图形学（程序签名图），第卷。 ) 11, 242–.(11,242-。 )

248.(248。 )

Crow, F.(乌鸦，F。 ) C.(C。 ) 1984.(克罗，F.C。1984年。 ) Summed-area tables for texture mapping.(纹理映射的求和区域表。 ) H.(H。 ) Christiansen, Ed.(克里斯蒂安森。 ), vol.(，第卷。 ) 18, 207–212.(18,207-212。 ) Held in Minneapolis, Minnesota.(在明尼苏达州明尼阿波利斯举行。 )

Everitt, C.(Everitt，C。 ), Rege, A.(，Rege，A。 ), and Cebenoyan, C.(，Cebenoyan，C。 ) 2000.(Everitt，C。，Rege，A。和Cebenoyan，C。2000。 ) Hardware shadow mapping.(2000年。 硬件阴影映射。 ) Tech.(技术人员。 ) rep.(代表。 ), NVIDIA Corp.(，Nvidia公司。 ), Feb.(，2月。 ) Available at http://www.(网址:http://www。 )nvidia.(英伟达。 )com/.(com/。 )

Fernando, R.(费尔南多 ), Fernandez, S.(，Fernandez，S。 ), Bala, K.(，巴拉，K。 ), and Greenberg, D.(，Greenberg，D。 ) P.(P。 ) 2001.(Fernando，R。，Fernandez，S。，Bala，K。和Greenberg，D。P。2001年。 ) Adaptive shadow maps.(自适应阴影映射。 ) In *SIGGRAPH ’01: Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 387–390.(2001年。 自适应阴影映射。 在SIGGRAPH'01:Proceedings of the28th Annual Conference on Computer Graphics and Interactive Technologies，ACM Press，纽约，NY，USA，387-390。 )

Kim, T.(金，T。 )-Y.(-Y。 ), and Neumann, U.(，Neumann，U。 ) 2001.(Kim，T-Y。和Neumann，U。2001。 ) Opacity shadow maps.(不透明阴影图。 ) In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, Springer-Verlag, London, UK, 177–182.(2001年。 不透明阴影图。 在第12届欧洲制图技术研讨会论文集，Springer-Verlag，伦敦，英国，177-182。 )

Lokovic, T.(洛科维奇 ), and Veach, E.(Veach，E。 ) 2000.(Lokovic，T。和Veach，E。2000年。 ) Deep shadow maps.(2000年。 很深的阴影地图。 ) In *Computer Graphics (Proc.(阴影图。 在计算机图形学中（过程） ) SIGGRAPH)*, 385–392.(计算机图形学（Proc.Siggraph），385-382。 )

Martin, T.(马丁 ), and Tan, T.(，Tan，T。 )-S.(-S。 ) 2004.(Martin，T。和Tan，T-S。 2004年。 ) Anti-aliasing and continuity with trapezoidal shadow maps.(梯形阴影图的抗锯齿性和连续性 ) In *Proceedings of the 2nd EG Symposium on Rendering*, Eurographics Association, Springer Computer Science, Eurographics.(2004年。 梯形阴影图的抗锯齿性和连续性 在第二届EG渲染研讨会论文集，欧洲制图协会，斯普林格计算机科学，欧洲制图。 )

McCool, M.(McCool，M。 ), and Du Toit, S.(，杜托瓦，S。 ) 2004.(McCool，M。和Du Toit，S。2004。 ) *Metaprogramming GPUs with Sh*.(2004年。 用SH。 ) AK Peters.(阿克·彼得斯。 )

Reeves, W.(里夫斯 ), Salesin, D.(，Salesin，D。 ), and Cook, R.(，和库克，R。 ) 1987.(里夫斯，W。，Salesin，D。，and Cook，R。1987。 ) Rendering antialiased shadows with depth maps.(使用深度图渲染抗锯齿阴影。 ) In *Proc.(1987年。 使用深度图渲染抗锯齿阴影。 继续。 ) SIGGRAPH*, vol.(继续。 Siggraph，vol。 ) 21, 283–291.(21,283-291。 )

Stamminger, M.(Stamminger，M。 ), and Drettakis, G.(，Drettakis，G。 ) 2002.(Stamminger，M。和Drettakis，G。2002年。 ) Perspective shadow maps.(透视阴影地图。 ) In *SIGGRAPH ’02: Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 557–562.(2002年。 透视阴影地图。 在Siggraph'02:第29届计算机图形学和交互技术年会论文集，ACM出版社，纽约，纽约，美国，557-562。 )

Williams, L.(威廉斯 ) 1978.(Williams，L.1978年。 ) Casting curved shadows on curved surfaces.(在曲面上投射曲面阴影。 ) In *Proc.(1978年。 在曲面上投射曲面阴影。 继续。 ) SIGGRAPH*, vol.(继续。 Siggraph，vol。 ) 12, 270–274.(12,270-274。 )

Williams, L.(威廉斯 ) 1983.(Williams，L.1983年。 ) Pyramidal parametrics.(金字塔参数学 ) In *Computer Graphics (SIGGRAPH ’83 Proceedings)*, 1–11.(1983年。 金字塔参数学 计算机图形学（Siggraph'83会议录），第1-11页。 )

Wimmer, M.(维默，M。 ), Scherzer, D.(、Scherzer、D。 ), and Purgathofer, W.(，Purgathofer，W。 )

2004.(2004年。 ) Light space perspective shadow maps.(光空间透视阴影图。 ) In *Proceedings of the 2nd EG Symposium on Rendering*, Eurographics Association, Springer Computer Science, Eurographics.(2004年。 光空间透视阴影图。 在第二届EG渲染研讨会论文集，欧洲制图协会，斯普林格计算机科学，欧洲制图。 )

1. e-mail: wdonnelly@math.(电子邮件:wdonnelly@math。 )uwaterloo.(滑铁卢。 )ca †e-mail: atlaurit@cgl.(电子邮件:wdonnelly@math.uwaterloo.ca？电子邮件:atlaurit@cgl。 )uwaterloo.(滑铁卢。 )ca.(大约。 ) [↑](#footnote-ref-1)
2. We only need two components, but current hardware does not support rendering to two component fp16 textures.(我们只需要两个组件，但目前的硬件不支持渲染到两个组件FP16纹理。 ) Optionally one could use multiple render targets instead of multiple components.(可选地，一个可以使用多个呈现目标而不是多个组件。 ) [↑](#footnote-ref-2)
3. NVIDIA provides an extension to perform post-depthcomparison bilinear filtering of shadow map lookups [Everitt et al.(NVIDIA提供了一个扩展来执行阴影地图查找的深度比较后双线性过滤[Everitt et al。 ) 2000].(2000年]。 ) We have not used this extension in our implementation, but it should be noted that its usage may improve performance on some platforms.(我们还没有在我们的实现中使用这个扩展，但是应该注意，它的使用可能会提高某些平台上的性能。 ) [↑](#footnote-ref-3)