DIFFUSION BASED PHOTON MAPPING

Lars Schjøth & Ole Fogh Olsen
IT University of Copenhagen
Rued Langgaards Vej 7, 2300 Copenhagen S

Jon Sporring
University of Copenhagen
Universitetsparken 1, 2100 Copenhagen Ø

Keywords: Ray-tracing, global illumination, photon mapping, caustics, density estimation, diffusion filtering.

Abstract: Density estimation employed in multi-pass global illumination algorithms give cause to a trade-off problem between bias and noise. The problem is seen most evident as blurring of strong illumination features. In particular this blurring erodes fine structures and sharp lines prominent in caustics. To address this problem we introduce a novel photon mapping algorithm based on nonlinear anisotropic diffusion. Our algorithm adapts according to the structure of the photon map such that smoothing occurs along edges and structures and not across. In this way we preserve the important illumination features, while eliminating noise. We call our method *diffusion based photon mapping*.

1 INTRODUCTION

Particle tracing is an important concept in global illumination. Particle tracing algorithms usually employ two passes. A first pass in which particles representing light are emitted from light sources and reflected around a scene, and a second pass which generates an image of the scene using the light transport information from the first pass. Common to all algorithms using particle tracing is that they trace light from the light sources. This generates information about lights propagation through the scene. In turn this information is used to reconstruct the illumination seen in the generated image. The advantage of particle tracing algorithms is that they effectively simulate all possible light paths. In particular they can simulate lighting phenomena such as color bleeding and caustics.

However, particle tracing algorithms are faced with a severe problem. In the particle tracing pass, particles are stochastically emitted from the light sources and furthermore often stochastically traced through possible light paths. This procedure induces noise, which has to be coped with in the reconstruction of the scene illumination. Unfortunately, the technique used to reduce noise also introduce a systematic error (bias) seen as a blurring of the reconstructed illumination. This is not necessarily a bad effect when concerned with slowly changing illumination, but it becomes an important problem when the illumination intensity changes quickly such as when concerned with caustics and shadows. This is a density estimation problem well-known in classical statistics.

In this paper we develop an algorithm which reduces noise and in addition preserve strong illumination features such as those seen in caustics. We have chosen to implement it in photon mapping. Photon mapping is a popular particle tracing algorithm developed by Henrik Wann Jensen (Jensen, 1996).

Our algorithm is inspired by a filtering method called *nonlinear anisotropic diffusion*. Nonlinear anisotropic diffusion is a popular method commonly used in image processing. It has the property of smoothing along edges in an image instead of across edges. Thus it preserves structures in an images while smoothing out noise. We call this novel algorithm *diffusion based photon mapping*.

Figure 1 illustrates two renderings; one using regular photon mapping and the other using diffusion based photon mapping. The images shows how diffusion based photon mapping reproduces caustics in higher detail than regular photon mapping.

Despite the fact that diffusion filtering is almost exclusively employed to process images, our method is not a post-processing step applied to photon mapping generated images. In diffusion based photon mapping we have adapted diffusion filtering in order to employ it on densities of photons during the illumination reconstruction.

Schjøth L., Fogh Olsen O. and Sporring J. (2006).
DIFFUSION BASED PHOTON MAPPING.
In *Proceedings of the First International Conference on Computer Graphics Theory and Applications*, pages 168-175
DOI: 10.5220/0001358101680175

Figure 1: [a b] Rendering of caustics created by a glass torus knot. Region zoom from Figure 3. a) Using regular photon mapping with cone filtering, b) using diffusion based photon mapping.

2 DENSITY ESTIMATION IN PHOTON MAPPING

In photon mapping indirect illumination is reconstruction through a series of queries to the photon maps. A photon map is a collection of photons created during a particle tracing phase in which photons are reflected around a scene using Monte Carlo ray tracing. Each query is used to estimate the reflected radiance at a surface point as the result of a local photon density estimate. This estimate is called *the radiance estimate*.

2.1 The Radiance Estimate

In his book (Jensen, 2001) Jensen derives an equation which approximates the reflected radiance using the photon map. This is done by letting the incoming radiance, L_i, at a point be represented by the incoming flux and letting the incoming flux at that point be approximated using the point's k nearest photons. In this way the equation for the reflected radiance becomes

$$L_r(\mathbf{x},\vec{\omega}) \approx \widehat{L}_r(\mathbf{x},\vec{\omega}) = \frac{1}{\pi r(\mathbf{x})^2} \sum_{i=1}^{k} f_r(\mathbf{x},\vec{\omega}_i,\vec{\omega})\Phi_i. \tag{1}$$

The equation sums over the k photons nearest the point $\mathbf{x}$. Φ_i is the flux represented by the i'th photon, f_r is the *bidirectional reflectance distribution function* (abbreviated BRDF), and $r(\mathbf{x})$ is the radius of a sphere encompassing the k nearest photons, where $\pi r(\mathbf{x})^2$ is the sphere's cross-sectional area through its center. The radius dependens on $\mathbf{x}$, as the its value is determined by the photon density in the proximity of $\mathbf{x}$. In density estimation $r(\mathbf{x})$ is the called the bandwidth, smoothing parameter or the windows width, we will use the terms bandwidth and support radius in this article.

The support radius is important because its size controls the trade-off between variance and bias. A small radius gives a limited support of photons in the estimate; it reduces the bias but increases the variance of the estimate. Inversely, estimating the radiance using a large radius results in an increase in bias and a decrease in variance.

Using a k'th nearest neighbor search to decide the support radius, Jensen helps limit bias and variance in the estimate by smoothing more where the photon density is sparse and less where the photon density is dense.

The radiance estimate in Equation 1 is simple insofar it weights each photon in the estimate equally. Jensen refined the radiance estimate in (Jensen, 1996) such that filtering was used to weight each photon according to its distance to the point of estimation.

It is possible to reformulate the radiance estimate to a general form such that it can be used with different filtering techniques. We formulate this general radiance estimate as

$$\widehat{L}_r(\mathbf{x},\vec{\omega}) = \frac{1}{r(\mathbf{x})^2} \cdot \sum_{i=1}^{k} K\left(\frac{(\mathbf{x}-\mathbf{x}_i)^T(\mathbf{x}-\mathbf{x}_i)}{r(\mathbf{x})^2}\right) \cdot f_r(\mathbf{x},\vec{\omega}_i,\vec{\omega})\Phi_i, \tag{2}$$

where $\mathbf{x}_i$ is the position of the i'th photon and $K(\mathbf{y})$ is a function that weights the photons according to their distance from $\mathbf{x}$. This function should be symmetric around $\mathbf{x}$ and it should be normalized such that it integrates to unity within the distance $r(\mathbf{x})$ to $\mathbf{x}$. In density estimation $K(\mathbf{y})$ is known as the kernel function. Usually, the kernel function decreases monotonically, weighting photons near $\mathbf{x}$ higher than those farther away. In this way the kernel function reduce bias where the change in density is significant.

In his PhD thesis (Jensen, 1996) Jensen presents the *cone filter*. This filter is used to reduce bias, such that edges and structure in the illumination are less blurred. As a kernel in the general radiance estimate the cone filter has the following form

$$K(\mathbf{y}) = \begin{cases} K(\mathbf{y}) = \frac{1-\frac{\sqrt{|\mathbf{y}|}}{k}}{(1-\frac{2}{3k})\pi} & \text{if } \sqrt{|\mathbf{y}|} < 1, \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

where $k \geq 1$ is a constant which controls the steepness of the filter slope.

Another useful kernel is the Epanechnikov kernel. The Epanechnikov kernel is known from statistics for its bias reducing properties and it is furthermore popular because it is computationally inexpensive. In computer graphics, Walter has employed it with good results in (Walter, 1998). In 2D the Epanechnikov kernel is given by

$$K(\mathbf{y}) = \begin{cases} \frac{2}{\pi}(1-\mathbf{y}) & \text{if } \mathbf{y} < 1, \\ 0 & \text{otherwise.} \end{cases} \quad (4)$$

In this paper we use the Epanechnikov kernel to examine our proposed method.

2.2 Bias Reduction

Bias reduction is a well examined subject, when concerned with density estimation both within the field of statistics and the field of computer graphics. Besides filtering, numerous methods addressing the issue has been presented.

The first method for reducing bias in photon mapping was suggested by Jensen (Jensen and Christensen, 1995). The method is called *differential checking* and it reduces bias by making sure that the support radius of the radiance estimate does not cross boundaries of distinct lighting features. This is done by expanding the support radius ensuring that the estimate does not increase or decrease, when more photons are included in the estimate.

Myszkowsky *et al.* (Myszkowski, 1997) suggested to solve the problem in much the same way as Jensen did with differential checking, however, they made the method easier to control and more robust with respect to noise. Myszkowsky *et al.* increase the support radius iteratively estimating the radiance in each step. If new estimates differ more from previous than what can be contributed variance, the iteration stops as the difference is then assumed to be caused by bias. More recently Schregle (Schregle, 2003) followed-up their work using the same strategy but optimizing speed and usability. Speed is optimized by using a binary search for the optimal support radius. This search starts in a range between a maximum and a minimum user-defined support radius. The range is split up, and the candidate, whose error is most likely to be caused by variance and not bias, is searched.

Shirley *et al.* (Shirley et al., 1995) introduced an algorithm for estimating global illumination. Like photon mapping this algorithm uses density estimation to approximate the illumination from particles generated during a Monte Carlo-based particle tracing step. However, unlike photon mapping the algorithm is gemoetry-dependent - the illumination is tied to the geometry. They called the algorithm *the density estimation framework* and they refined it in a series of papers.

The first edition of their framework did not try to control bias. In (Walter et al., 1997) they extended the framework to handle bias near polygonal boundaries. This was done by converting the density estimation problem into one of regression. In this way they could use common regression techniques[1] to eliminate boundary bias.

Later Walter in his PhD thesis (Walter, 1998), reduced bias by controlling the support radius of the estimate using statistics to recognize noise from bias. Benefiting from the field of human perception he used a measure for controlling the support radius such that noise in the estimate was imperceptible to the human eye.

Walter recognized that if bias was to be significantly reduced, using his method, perceptual noise had to be accepted in the vicinity of prominent edges and other strong lighting features. This is a common problem which also affects differential checking and both Schregle's and Myszkowsky's method. Hence, in the proximity of strong features such as the edges of a caustic the support radius stops expanding and the foundation on which the estimate is made is supported by few photons. This means that when estimates are made close to edges the support is limited and noise may occur.

In *diffusion based photon mapping* we employ the concept of nonlinear anisotropic diffusion in the radiance estimate of photon mapping. Nonlinear anisotropic diffusion is a well examined and popular technique within the field of image analysis. It is a filtering technique that adapts its smoothing according to the image structure. This means that it smoothes along edges and not across. It is known to be robust and effective (Weickert, 1998). To our knowledge the technique has not been employed in connection with photon mapping.

In contrast to Myszkowsky, Schregle and Walter's approach our method will smooth along edges and structures, it follows that its support will not be limited in the proximity of these.

[1] Specifically they used locally-weighted polynomial least-squares regression to eliminate boundary bias.

3 ANISOTROPIC FILTERING IN PHOTON MAPPING

To be able to use anisotropic filtering in photon mapping, we in some way have to be able describe the structure of the photon map, to get some guidance as how to adapt the filtering. It follows that it is necessary to modify the radiance estimate such that the kernel adapts according to the structure description and that we, furthermore, need to normalize this modified radiance estimate in order to preserve energy when the kernel changes shape.

3.1 Structure Description

The gradient of the illumination function denotes the orientation in which the illumination intensity changes and therefore describes the first order structure of the illumination. This information will be used to steer the filtering.

As the illumination function is estimated in the radiance estimate, the differentiated radiance estimate approximates the gradient of the photon map.

To differentiate the radiance estimate we combine the generalized radiance estimate from Equation 2, with a suitable kernel function. Furthermore, it is convenient to simplify the radiance estimate by assuming that all surfaces hit by photons are ideal diffuse reflectors. This means that the BRDF, f_r, is constant regardless of the incoming and outgoing direction of light. In this way the BRDF does not need to be differentiated as it does not depend on the position, $\mathbf{x}$, which is the variable in respect to which we differentiate.

This of course is a radical assumption as photons can be affected much by the type of surfaces they encounter. However, photons are only stored on diffuse surfaces, so the surfaces involved in the radiance estimate are most likely diffuse and need therefore not differ much from an ideal diffuse surface. Furthermore if we were to differentiate the BRDF then our algorithm would not be able to handle arbitrary BRDFs as we would have to know the BRDF in order to do so. In effect we would not retain the beneficial qualities of photon mapping. Another solution would of course be to do reverse engineering, to numerically estimate the BRDF in question, however, this approach is both cumbersome and computationally expensive.

Additionally, we have to make a constraint on the generalized radiance estimate. The estimate should use a fixed support radius for $r(\mathbf{x})$ such that the radius is independent of $\mathbf{x}$. Though this effectively reduces the radiance estimate to a common multivariate kernel estimator - rather than a k'th nearest neighbor estimator - this is not a severe constraint. The advantage of the k'th nearest neighbor search is its ability to reduce bias. This ability is important in the radiance estimate, however, when estimating the gradient, smoothing is an advantage as the gradient is perceptible to noise.

Combining a simplified version of the generalized radiance estimate with the two-dimensional Epanechnikov kernel we get

$$\widehat{L}_r(\mathbf{x},\vec{\omega}) = \frac{2f_r}{\pi r^2}\sum_{i=1}^{k}\left(1-\frac{(\mathbf{x}-\mathbf{x}_i)^T(\mathbf{x}-\mathbf{x}_i)}{r^2}\right)\Phi_i, \quad (5)$$

This equation can be differentiated giving us the gradient function of the estimated illumination function. Differentiating Equation 5 with respect to the j'th component of $\mathbf{x}$ gives the partial derivative

$$\frac{\partial\widehat{L}_r(\mathbf{x},\vec{\omega})}{\partial x_j} = \frac{4f_r}{\pi r^2}\sum_{i=1}^{k} -\frac{x_j - x_{ij}}{r^2}\Phi_i. \quad (6)$$

As seen from Figure 2, the gradient of the photon map is a plausible structure descriptor. Figure 2a is a distribution of photons and Figure 2b is a gradient field of the distribution. The gradient vectors are calculated using the photons nearest the center of each quadrant in the grid of the field. The gradient vectors along the edges of the distribution are those with greatest magnitude and the vectors are as expected perpendicular to edges and structures.

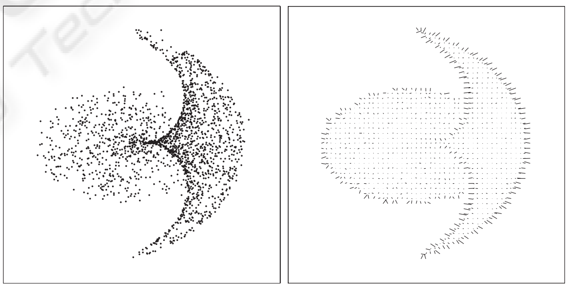

Figure 2: [a b] a) Cardioid shaped photon distribution created by light reflection within a metal ring, b) gradient field of the photon distribution in a).

We denote the gradient of the photon map $\nabla\mathbf{M}$, where $\mathbf{M}$ is the photon map.

A more advanced way is to describe the first order structure is with the *structure tensor*. The structure tensor was introduced to diffusion filtering by Weickert (Weickert, 1995). The advantage of the structure tensor is that even though it does not contain more information than the gradient descriptor, it is, unlike the gradient, possible to smooth it without losing important structure information. Being able to smooth the structure descriptor makes the orientation information less perceptible to noise.

The structure tensor is the tensor product of the gradient. In three dimensions it is given by

$$\nabla\mathbf{M}\otimes\nabla\mathbf{M} = \begin{pmatrix} M_x^2 & M_xM_y & M_xM_z \\ M_xM_y & M_y^2 & M_yM_z \\ M_xM_z & M_yM_z & M_z^2 \end{pmatrix}. \quad (7)$$

In diffusion based photon mapping we use the structure tensor to describe the structure of the photon map.

3.2 Diffusion Tensor

In diffusion filtering the filtering is controlled by a symmetric positive semidefinite matrix called the diffusion tensor (Weickert, 1998). This matrix can be constructed using information derived from a structure descriptor. One possible construction is for edge enhancing which will be apply in this paper to preserve the finer structures of the illumination.

For edge enhancing, smoothing should occur parallel to the edges not across them. The orientation of the local edge structure is derived from the structure tensor. The primary eigenvector of the structure tensor is simply the gradient which is perpendicular to the local edge orientation. The direction parallel to edges can be calculated as the cross-product of the surface normal and the vector representing the direction parallel to the structure. This will be the main direction of diffusion.

The eigenvectors and eigenvalues of the diffusion tensor describe respectively the main directions of diffusion and the amount of diffusion in the corresponding direction. Hence by constructing the diffusion tensor from the primary eigenvector of the structure tensor diffusion can be steered to enhance the edges.

The gradient of the illumination function (derived from the structure tensor) is only in the tangent plane to the surface if the surface is locally flat. Since the photon energy should stay on the surface we must insure that the main diffusion direction is the tangent plane.

Consequently, the diffusion directions are constructed in the following way.

The primary eigenvector of the structure tensor is projected to the plane perpendicular to the surface normal. This is our second diffusion direction $\vec{X}_2$. The third diffusion direction is the surface normal $\vec{X}_3$ and the primary diffusion direction is the cross product of the second and third diffusion direction $\vec{X}_1$. All vectors should be normalized.

The diffusion tensor is constructed as

$$\mathbf{D} = \mathbf{X}\ \mathrm{diag}(\lambda_1, \lambda_2, \lambda_3)\ \mathbf{X}^T, \quad (8)$$

where $\mathbf{X}$ is $[\vec{X}_1\vec{X}_2\vec{X}_3]$ and $\mathrm{diag}(\cdot)$ is the diagonal matrix containing the eigenvalues of $\mathbf{D}$ along the diagonal.

It remains to determine the amount of diffusion. That is the eigenvalues, λ:

$$\begin{aligned} \lambda_1 &= 1, \\ \lambda_2 &= \frac{1}{1+\left(\frac{\mu_1}{K}\right)^{1+\alpha}}, \quad \alpha > 0, \\ \lambda_3 &= 0.1, \end{aligned} \quad (9)$$

where the secondary eigenvalue, λ_2, is estimated using a function called the *diffusivity function*, introduced to diffusion filtering by Perona and Mallik (Perona and Malik, 1990). The diffusivity coefficient, K, decides when the function starts to monotonically decrease and α the steepness of the decline. In practice what it means is, that K is the threshold deciding what value of the primary eigenvalue of the structure tensor, μ_1, is considered an edge and what is considered noise, and α controls the smoothness of transition.

We suggest that the tertiary eigenvalue, λ_3 should be set to 0.1. The reason why the tertiary vector should have such a low eigenvalue is that it limits certain forms of bias. It is comparable to using a disc instead of a sphere to collect photons; a techniques to reduce bias in corners (Jensen, 2001).

We have now constructed a diffusion tensor which favors diffusion parallel to structures while limiting diffusion perpendicular to structures. We will utilize this tensor such that it controls the filtering of the photon map.

3.3 The Diffusion Based Radiance Estimate

The next step is to use the diffusion tensor to shape the kernel of the radiance estimate such that it smoothes along structures and edges and not across. To do this we have to shape our kernel in some way.

If we take a look at the multivariate kernel density estimator on which the radiance estimate is based,

$$\widehat{f}(\mathbf{x}) = \frac{1}{nh^d}\sum_{i=1}^{n} K\left(\frac{(\mathbf{x}-\mathbf{x}_i)^T(\mathbf{x}-\mathbf{x}_i)}{h^2}\right), \quad (10)$$

it is important to note that the kernel is uniform insofar that only a single parameter, h is used. Data points, $\mathbf{x}_i$, are scaled equally according to the bandwidth, h, and their distance to the center, $\mathbf{x}$. Smoothing occurs equally in all directions.

Now considering a simple two dimensional normal distribution:

$$f(\mathbf{x}) = \frac{1}{2\pi\sigma_1\sigma_2}\exp\left(-\frac{(x_1-\mu_1)^2}{\sqrt{2}\sigma_1} - \frac{(x_2-\mu_2)^2}{\sqrt{2}\sigma_2}\right), \quad (11)$$

where σ_1 and σ_2 are the standard deviations with respect to the axes and μ is the center of the distribution. Here we have a Gaussian kernel whose shape is specified by the two parameters for the standard deviation. Unfortunately, this equation only gives control in two directions.

However, generalizing the equation to d dimensions, we can use an inversed $d \times d$ covariance matrix, Σ^{-1}, to shape the normal distribution:

$$f(\mathbf{x}) = \frac{1}{(2\pi)^{d/2}\sqrt{\det \Sigma}} \cdot \exp\left(-\frac{(\mathbf{x}-\mu)^T \Sigma^{-1}(\mathbf{x}-\mu)}{2}\right). \quad (12)$$

In two dimensions using a diagonal covariance matrix with the variance values in the diagonal this equation is exactly the same as Equation 11. However, using a matrix we are not limited to control the shape of the Gaussian in only two direction. If we for example had shaped our Gaussian kernel to form an ellipse, we could rotate this kernel by rotating the covariance matrix. The equation will remain normalized as the determinant of a matrix is rotational invariant. So the shape of normal distribution in Equation 12 is controlled by the covariance matrix.

We can use Equation 12 to extend the generalized radiance estimate from Equation 2. To generalize the shape adapting properties we use the Mahalanobis distance from Equation 12 to shape the kernel. The Mahalanobis distance is a statistical distance. It is given by:

$$d(\mathbf{x},\mathbf{y}) = (\mathbf{x}-\mathbf{y})^T \Sigma^{-1}(\mathbf{x}-\mathbf{y}). \quad (13)$$

As the shape of the kernel should be controlled by the diffusion tensor we use the tensor in place of the covariance matrix. We can reformulate the generalized radiance estimate as:

$$\widehat{L}_r(\mathbf{x},\vec{\omega}) = \frac{1}{r^2\sqrt{\det \mathbf{D}}} \cdot \sum_{i=1}^{k} K\left(\frac{(\mathbf{x}-\mathbf{x}_i)^T \mathbf{D}^{-1}(\mathbf{x}-\mathbf{x}_i)}{r^2}\right) \cdot f_r(\mathbf{x},\vec{\omega}_i,\vec{\omega})\Phi_i. \quad (14)$$

We now have a general diffusion based radiance estimate, which filters the photon map adapting the shape of the kernel according to the diffusion tensor. Or to be even more general we have a radiance estimator which estimates the illumination function taking into consideration the structure of the photon map, such that edges and structures are preserved.

3.4 Implementation

Diffusion based photon mapping can be implemented in different ways depending on which structure descriptor is used, however, we propose to use the structure tensor and for this reason we need to estimate it or have it available during the radiance estimate in order to construct the diffusion tensor.

We do this using a preprocessing step that approximates the gradient of the photon map. The preprocessing step occurs between the photon tracing pass and the rendering pass. To approximate the gradient we sample it at all photon positions. The advantage of this procedure is that we can store the local gradient along with the photon and in this way does not need a separate gradient map. Additionally, we know the sampling positions to be located on a surface, as photons are only stored in connection with a surface. This is useful as the gradient is only relevant at surface positions.

During the radiance estimate we calculate the structure tensors at the photon positions near $\mathbf{x}$. In this way we can estimate the local structure tensor as the weighted average of the surrounding structure tensors. Smoothing the structure tensor reduces noise and furthermore gives a broader foundation from which to steer the filtering after.

Having calculated the local structure tensor we construct the diffusion tensor as described in the former section. This then is used in the general diffusion based radiance estimate together with a suitable kernel.

For a more through examination of diffusion based photon mapping refer to (Schjøth, 2005).

4 RESULTS

Figure 3 is a rendering of a scene consists of a simple glass torus knot positioned over a plane in space. A spherical light source above the torus knot creates the caustic on the plane.

Figure 3a illustrates the scene visualized employing the k'th nearest neighbor estimate in conjunction with the cone kernel whereas Figure 3b is visualized using diffusion based photon mapping.

The difference in the two images is seen in the caustics. The structure of the caustic in Figure 3b is much more detailed than the caustic in Figure 3a. Fine patterns are visible in the caustic in the image estimate using the diffusion based radiance estimate that are not visible in the image estimated using the cone kernel.

To further test diffusion based photon mapping we have constructed a photon distribution. The constructed distribution is rather simple yet it contains both edges and ridges and circular and rectangular shapes.

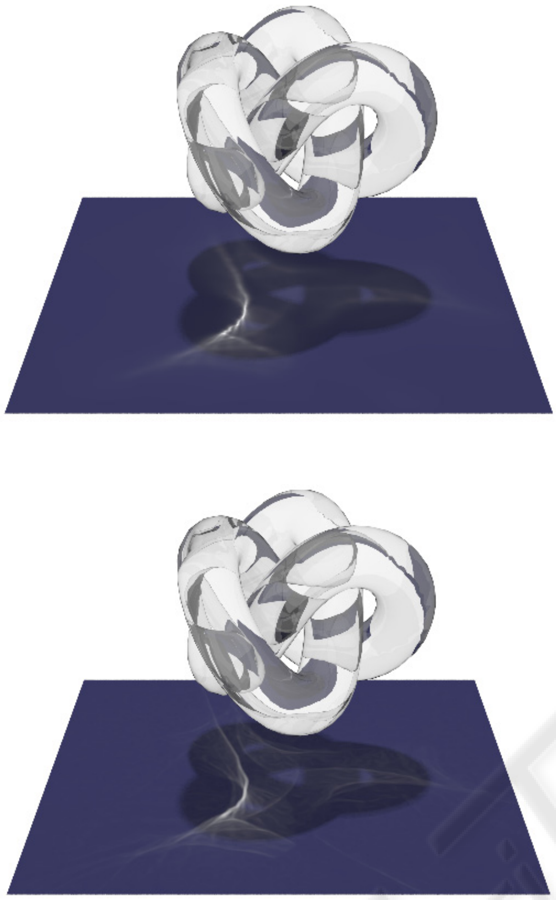

Figure 3: $\left[\begin{smallmatrix}a\\b\end{smallmatrix}\right]$ Rendering of a glass torus positioned above a plane using a) regular photon mapping and b) diffusion based photon mapping.

4.1 The Cone Kernel

We first test Jensen's cone kernel on the constructed distribution. This is done by first combining the cone kernel from Equation 3, with the general radiance estimate from Equation 2. We then estimate the radiance of the constructed photon distribution a number of times, iteratively expanding the support radius, allowing an increasing amount of photons in each estimation. This is done until the result contains an acceptable low noise level. The result of this procedure is illustrated in Figure 4. It is seen from the illustration that the noise level decreases slowly with respect to the number of photons per estimate. Bias is visible as a clearly identifiable blurring of shape edges. In addition boundary bias is seen along the boundaries of the images. It should be clear that the bias increases as the noise is reduced. This phenomenon

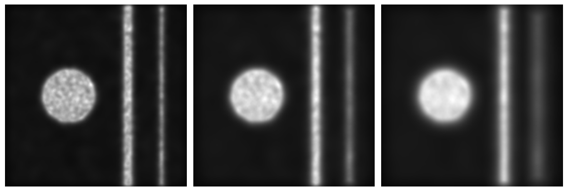

Figure 4: [a b c] A visualization of a constructed distribution estimated using the cone kernel. a) estimated using the 200 nearest photons, b) estimated using the 400 nearest photons and, c) estimated using the 800 nearest photons.

is directly related to the bias vs. variance trade-off accounted earlier. Another thing to notice is how the thin line losses intensity as the number of photons per estimate is increased. This happens because the energy of the line is spread out over a larger area as the smoothing increase.

4.2 The Diffusion Based Radiance Estimate

To test the applicability of the structure tensor as structure descriptor, we use the diffusion based radiance estimate together with the Epanechnikov kernel. In contrast to the cone kernel radiance estimate we will not us the k nearest neighbor method to reduce bias, instead we will use a fixed support radius letting the shape adaption reduce bias.

We first set the support radius low, and then we iteratively increase the support radius until the noise level is acceptable. This is done using a large value of the diffusivity coefficient K from Equation 9. In this way the kernel will stay uniform and will not adapt according to structure. Estimating the radiance with a uniform Epanechnikov kernel using different support radii we find a support radius which reduces noise to an acceptable level.

Using this support radius we test the diffusion based radiance estimate by iteratively decreasing the value of the diffusivity coefficient such that the kernel starts to adapt its shape according to the structure described by the structure tensor. The result of this procedure is illustrated in Figure 5. From the results of the diffusion based radiance estimate we see that edges are enhanced as the diffusivity coefficient is decreasing.

4.3 Summary of the Visual Results

We summarily compare the best results of the two radiance estimation methods. The results were chosen in the attempt to find the estimates with least noise and least bias. Figure 6 depicts the estimates. From

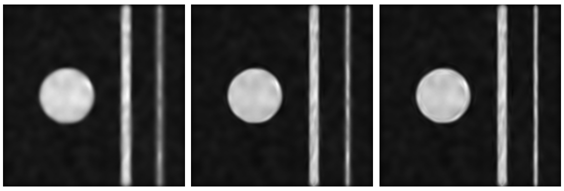

Figure 5: [a b c] A visualization of a constructed distribution estimated using diffusion based photon mapping. a) estimated using K=0.4, b) estimated using K=0.2 and, c) estimated using K=0.1.

the chosen results we see that the diffusion based radiance estimates reduces bias markedly better than the common k'th nearest neighbor radiance estimate. However, background noise is less pronounced in the common radiance estimate.

Another thing to notice is the thinnest line in the constructed distribution. We know that this line has photon distribution as dense as the two other shapes in the distribution. For this reason the thin line should be just as intense as the other shapes. However, as estimates are smoothed using a higher support radius and more photons per estimate, the energy is spread out. Comparing the three results it is seen that the structure based radiance estimate is most successful in preserving the energy of the thin line as it has almost the same intensity as the other shapes.

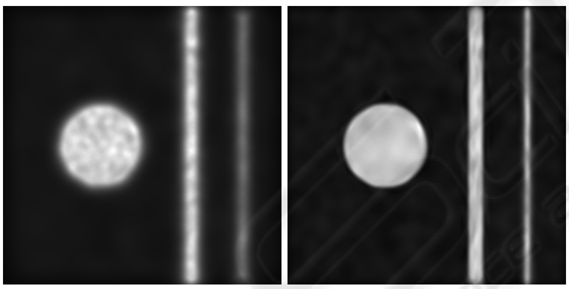

Figure 6: [a b] A visualization of a constructed distribution estimated using different radiance estimation methods. a) estimated using a uniform cone kernel with 400 photons per estimate and b) estimated using a gradient based shape adapting Gaussian radiance estimate with a diffusivity coefficient of K = 0.2.

Finally, it should be mentioned that it has not been our objective to test the computationally performance diffusion based photon mapping. Despite this we will say a few things about the running time. Considering the estimates of the constructed distribution. The image in Figure 6a was estimated using the cone kernel and 400 photons per estimate. It was computed in 45 seconds. In comparison the computation time for the tensor based radiance estimate, producing Figure 6b, was 1 minute and 3 seconds from which 12 seconds was used estimating the gradient map.

5 CONCLUSION

In this paper we have proposed a novel method for enhancing edges and structures of caustics in particle tracing algorithms. Our method is called diffusion based photon mapping.

The method is based on nonlinear anisotropic diffusion which is a filtering algorithm known from image processing. We have implemented it in photon mapping and we have shown that our algorithm is markedly better than regular photon mapping when simulating caustics. Specifically, does diffusion based photon mapping preserve edges and other prominent features of the illumination where as regular photon mapping blur these features.

REFERENCES

Jensen, H. W. (1996). *The Photon Map in Global Illumination*. PhD thesis, Technical University of Denmark, Lyngby.

Jensen, H. W. (2001). *Realistic image synthesis using photon mapping*. A. K. Peters, Ltd., Natick, MA, USA.

Jensen, H. W. and Christensen, N. J. (1995). Photon maps in bidirectional monte carlo ray tracing of complex objects. *Computers & Graphics*, 19(2):215–224.

Myszkowski, K. (1997). Lighting reconstruction using fast and adaptive density estimation techniques. In *Proceedings of the Eurographics Workshop on Rendering Techniques '97*, pages 251–262, London, UK. Springer-Verlag.

Perona, P. and Malik, J. (1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-12(7):629–639.

Schjøth, L. (2005). Diffusion based photon mapping. Technical report, IT University of Copenhagen, Copenhagen, Denmark.

Schregle, R. (2003). Bias compensation for photon maps. *Computer Graphics Forum*, 22(4):729–742.

Shirley, P., Wade, B., Hubbard, P. M., Zareski, D., Walter, B., and Greenberg, D. P. (1995). Global illumination via density-estimation. *Rendering Techniques '95*, pages 219–230.

Walter, B. (1998). *Density estimation techniques for global illumination*. PhD thesis, Cornell University.

Walter, B., Hubbard, P. M., Shirley, P., and Greenberg, D. P. (1997). Global illumination using local linear density estimation. *ACM Trans. Graph.*, 16(3):217–259.

Weickert, J. (1995). Multiscale texture enhancement. *Lecture Notes in Computer Science*, 970:230–237.

Weickert, J. (1998). *Anisotropic Diffusion in Image Processing*. B. G. Teubner, Stuttgart, Germany.