Final Gathering using Ray Differentials

Mark Gjøl
IMM DTU
DTU - bygning 321
2800 Lyngby

cyrax@b0rken.dk

Bent Dalgaard Larsen
IMM DTU
DTU - bygning 321
2800 Lyngby

bdl@imm.dtu.dk

Niels-Jørgen Christensen
IMM DTU
DTU - bygning 321
2800 Lyngby

njc@imm.dtu.dk

ABSTRACT

We have developped a method for attaining better quality from the calculations of indirect illumination in photon mapping, used for global illumination.

During the photon mapping calculations, the final gathering step has previously sampled the world over areas that have largely been dependant on the distribution of photons. This strategy gives a series of problems that are usually solved by increasing the sampling rate. To reduce these problems, we suggest sampling over the real area, given by the area the ray ought to cover. We calculate the footprint by using ray differentials, and then look up in a precalculated mipmap of estimated irradiance values. By doing this we get a better estimate of the incoming diffuse lighting, which means a more correct image with the same amount of samples. By using a mipmap of irradiance estimates, this enhancement is reached faster than by using regular final gathering.

Keywords

Ray Tracing, Global Illumination, Photon Mapping, Ray Differentials, Final Gathering.

1.INTRODUCTION

In ray tracing it is common to separate the rendering of an image into direct and indirect lighting. Determining the amount of direct light reaching a point is usually fairly straightforward, whereas computing the indirect light might require a considerable effort. A popular technique for estimating the indirect lighting is called photon mapping [Jen95a], where a database of light information is extensively used to find out how much light is reflected towards a given point.

In the final gathering step of photon mapping, it is attempted to estimate the indirect irradiance by sampling the world in the visible hemisphere, and collecting photons where visible. Since the photons are scattered relatively sparsely over the scene, it is necessary to collect them over a small area rather than at the point the ray hits. This might pose a problem, if the direct illumination on a surface varies with a high frequency: You might either miss some spots with a different irradiance because the sampling area is too small, or you might sample areas too much, because the sampling area is too big. With too few samples, which may be a moderate amount, classical final gathering will inevitably generate a lot of noise for this kind of scene. Furthermore, since the sampling size is not dependant on the distance the ray has travelled, rays travelling far will in most instances be sampling over a too small area, leaving large areas unsampled, while rays travelling only short distances will be over-sampling.

We suggest letting the sampling area be dependant on the distance the ray has travelled, and the amount of rays used for sampling the indirect irradiance. By making the sampling area dependant on the distance, a ray travelling a long distance will be sampling a large area, while a ray travelling a short distance only sample a small area. Letting the sampling area be dependant on the journey of the ray will help eliminate noise in the final result, since more visible surface will be sampled, while details in the indirect illumination will remain.

By using a mipmap of irradiance estimates the time it takes to estimate the irradiance is kept low, even samples covering lots of photons. Furthermore the mipmap ensures that the irradiance estimates remain correct, even for large sampling areas, that may cover empty space – see figure 1.

2.RELATED WORK

In 1986 Kajiya [Kaj86a] introduced path tracing as a means of solving the rendering equation that was proposed at the same time. By shooting rays from the eye into the scene, and letting them bounce around until a light source is hit, direct as well as indirect lighting is taken into account, and the method will

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
Copyright UNION Agency – Science Press, Plzen, Czech Republic.

asymptotically converge towards a correct image. In order to accelerate the convergence, the direct irradiance can be taken into account at each bounce. However, as this method only traces rays from the eye, the importance of the light position is somewhat neglected, and images will only be rid of noise after a considerable number of samples.

In 1985 Arvo suggested tracing light from the light sources into the scene in a step before rendering in what he called Backward Ray Tracing [Arv86a]. By storing the points the rays hit in illumination maps he is able to better calculate diffuse and specular reflection of light.

In 1993 Lafortune and Willems suggested Bi-Directional path tracing [Laf93a], in which a ray is simultaneously traced from the light and the camera, thus rendering both positions with equal weight. Visibility tests are then performed from the points the eye ray hits to the points the light ray has hit to determine the irradiance from these points.

In 1995 Jensen and Christensen suggested photon mapping [Jen95a] where a step is introduced before the rendering pass, where rays are sent in all directions from the light sources. The intensities of these rays are saved in a Kd-tree – a photon map, which can be used to determine the indirect irradiance in the rendering pass.

To save expensive lookups in the photon map and to reuse calculations determining the irradiance at a point, Christensen suggested calculating irradiance estimates [Chr99a] for all or a fraction of all photons in the photon map. This way it is only necessary to search for a single irradiance estimate in a data structure, rather than for several photons and calculating their irradiance.

In order to manage scenes with huge photons maps, Christensen [Chr04a] suggested making an irradiance atlas where irradiance estimates are stored in a collection of maps that cover different sampling areas. Ray differentials are then used to determine which map to sample. By doing this Christensen is able to use the harddrive as efficient storage for the irradiance values. Furthermore he gets an effect, where rays with large footprints will sample the irradiance in large areas and rays with small footprints will sample it small areas.

Igehy proposed the use of ray differentials in 1999 [Ige99a] as a way to sample the footprint of the ray, such that textures can be sampled using a fitting kernel. By keeping track of the ray's footprint along two independant axes Igehy is able to get an anisotropic estimate of the ray's footprint using only few extra calculations compared to a strictly one-dimensional ray – that is, the footprint can be ellipse or rectangular shaped. It thus requires that you keep track of four extra vectors; the partially differentiated position regarding x and y: $\frac{\partial \boldsymbol{P}}{\partial x}, \frac{\partial \boldsymbol{P}}{\partial y}$, and the differentiated direction regarding x and y: $\frac{\partial \boldsymbol{D}}{\partial x}, \frac{\partial \boldsymbol{D}}{\partial y}$. $\boldsymbol{D}$ is here the direction of the ray, while $\boldsymbol{P}$ is the position where the ray has hit.

3.FINAL GATHERING WITH RAY DIFFERENTIALS

In order to get a better estimate of the indirect irradiance at a given point in the scene we suggest sampling irradiance estimates over areas that correspond to the areas the sampling rays ought to cover.

After the photons have been emitted from the light sources, an irradiance mipmap must be built for looking up in during final gathering. When the mipmap has been built, the source photons can be deleted from the memory such that less memory will usually be used.

At final gathering several new rays are then sent out into the world from the point we are inspecting. When they hit an object, their footprint is determined by using ray differentials, and irradiance estimates are averaged from the mipmap to determine the average irradiance over that area.

Irradiance estimate mipmap

At the moment the photons have been distributed in the scene, an initial map of irradiance estimates must be created – this will be the most detailed layer of the mipmap. The map is created as described by Christensen [Chr99a], where an irradiance estimate is created at the position of each photon. A larger sampling area is then chosen – for instance twice as big – and a new layer of the mipmap is created, based on the initial map. This is done simply by averaging over the estimates in the initial map. If the photons were used directly artifacts would be generated when sampling at the edges of objects. This is illustrated in figure 1. Just as Christensen suggests, you should only average over estimates, where the dot product of the normal of the surface they inhabit and the current surface normal is more than or equal 0.9. This is to ensure that the irradiance estimates are related to the surface in question. If the surface normal deviates more than about 25 degrees the surface is considered unrelated, and a new estimate is created. More maps are generated until a satisfactory sampling size has been achieved, the irradiance estimates of the previous map should be used, rather than the initial map every time, to speed up the process.

Note that an irradiance estimate should only contribute to the average, if the entire estimate is completely inside the new estimate. This means that it should only be used if $\|\boldsymbol{x_i}-\boldsymbol{x_j}\|<r_i-r_j$, where $\boldsymbol{x_i}$ is the position of the new estimate, $\boldsymbol{x_j}$ is the position of the estimate being considered, r_i is the radius of the new estimate and r_j is the radius of the estimate being considered.

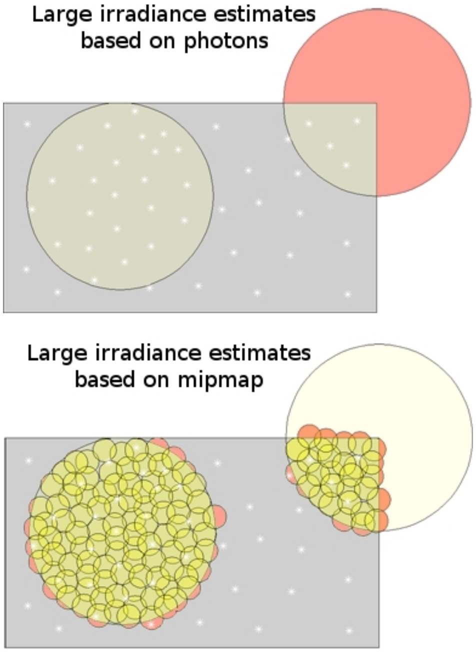

Figure 1: The difference between sampling estimates and photons. The red areas show the error in the estimate.

3.1.1Irradiance estimate removal

In order to avoid too many irradiance estimates in the mipmap, some values must be removed. This is both important because the memory usage would become overwhelming, but also because it does not make sense to have that many overlapping irradiance estimates. When removing estimates in this fashion one will have to search a smaller map and collect fewer estimates in the final gathering step without having to compromise image quality.

Before creating the initial irradiance estimate map, the photons must be inspected. Inspecting each photon, nearby photons must be marked as being not required, thus leaving evenly spaced photons all over the scene, where irradiance estimates are to be calculated. Obviously marked photons should not be inspected, as this would mark all photons, and no photon positions would be used. Also, as when creating the map, photons – and later other estimates – should only be marked if the dot product of the normal of the surface they are on, and the current normal is more than or equal 0.9. This is again only a suggested value that is used to ensure that the geometry is more or less coherent. When the photons have been marked, irradiance estimates are calculated for the positions of all the photons that have not been marked and saved in the initial irradiance estimate map. A certain value must be chosen for dertermining which photons are too close. We suggest a quarter of the initial sampling radius, since this will leave a fairly detailed layer.

When the initial layer has been created, the irradiance estimates of this layer are inspected. Just as with the photons, when inspecting one estimate, nearby estimates are marked as not having to be re-estimated at larger maps. A good value for marking estimates is about 0.75x the radius of the current map. This will give a fairly sparse distribution with only a small overlap in estimates. The method is illustrated in figure 2.

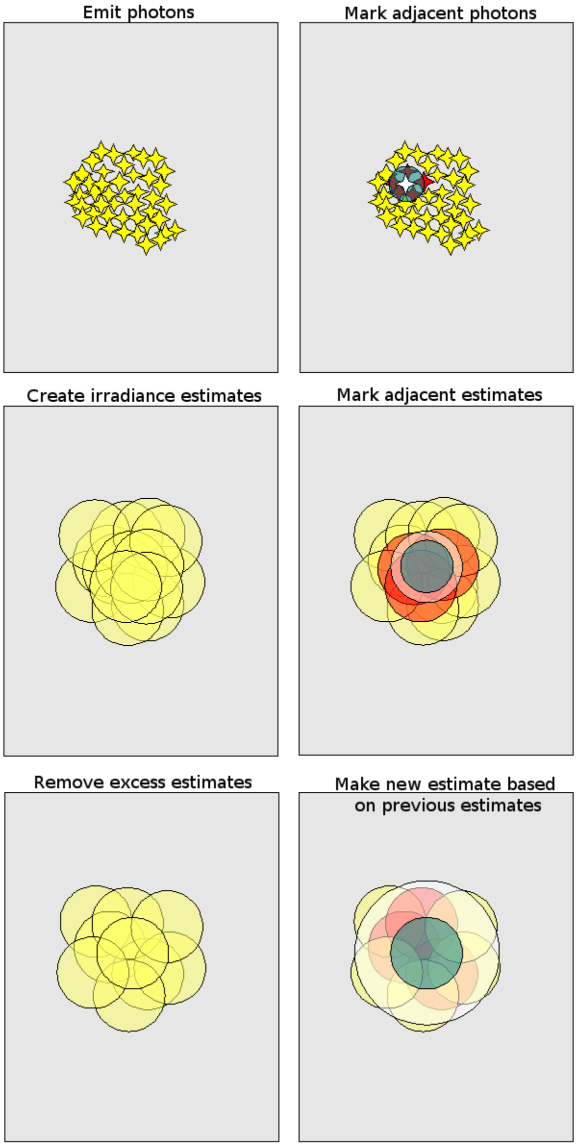

Figure 2: After the photons have been distributed in the scene adjacent photons are marked. A new map is built in the places of the unmarked photons based on all photons. Adjacent estimates are now marked and a new layer of the mipmap is built based on the first one.

Initial values in final gathering

When the rays are initially sent out, the differentiated position is initialized to zero. Ray differentials are therefore not used prior to the final gathering step. Since the indirect irradiance rarely changes much over small areas it is not necessary to sample the exact area the ray covers.

The differentiated direction: $\frac{\partial \boldsymbol{D}}{\partial x}, \frac{\partial \boldsymbol{D}}{\partial y}$ must be initialized in a way, such that the rays' footprints cover the entire visible hemisphere. When using a unit hemisphere, the total area that needs to be covered on the hemisphere is $A=2\pi r^2=2\pi$. If we are sampling with *N* rays each ray should cover an area of $\frac{2\pi}{N}$. Since the amount of rays commonly used for final gathering is in the thousands thus having thousands of rays each covering a small area on the visible hemisphere, the area for each ray can be assumed to be locally flat. The radius of the footprint of each ray has to be:

$$A=\pi r_f^2 \Leftrightarrow \frac{2\pi}{N}=\pi r_f^2 \Leftrightarrow r_f=\sqrt{\frac{2\pi}{N}}$$

If the hemisphere is being sampled in a cosine lobe the differentiated directions should be:

$$\frac{\partial \boldsymbol{D}}{\partial x}=\frac{\partial \boldsymbol{D}}{\partial y}=\frac{r_f}{\cos(\theta)}=\frac{\sqrt{\frac{2\pi}{N}}}{\cos(\theta)}$$

θ is here the angle between the normal, and the direction of the ray. Figure 3 illustrates how the rays are now distributed evenly over the visible hemisphere.

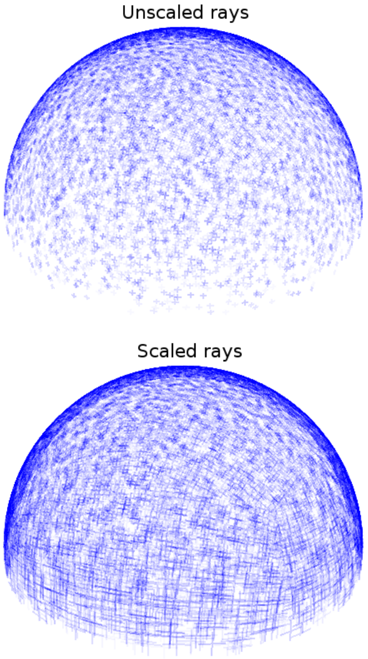

Figure 3: Ray footprints drawn on the surrounding hemisphere. Footprints are visualized as the a cross showing the differentiated position and are averaged over ten pictures. When not scaling the rays gaps appear along the edge of the hemisphere.

Sampling

When a ray from final gathering has hit an object, the footprint must be sampled. Since the partially differentiated direction was calculated as a circle, the footprint is now an ellipse. Since the irradiance estimates are circular, it is necessary to use a smaller layer in the mipmap, such that several estimates are averaged, rather than just the closest one. This means that the layer that is one size smaller than the closest (in size) in the mipmap should be used. To find the size of the footprint, you can just take the longest partially differentiated position. Find the size in the mipmap that is the closest to this, and choose the layer that is based on half that radius.

Note, that if the radius is smaller than the initial irradiance estimate map, the closest irradiance estimate should simply be chosen. This is not a problem, since the area will be very small, and the photon map might not be detailed enough to give a usable estimate of the irradiance for sizes smaller than those saved in the mipmap.

If the irradiance estimates are stored in a Kd-tree, a common way to access them is by getting all estimates within a certain radius in a spherical search. This can be used for collecting the irradiance estimates. Collect all photons within the radius corresponding to the largest partially differentiated position, project the positions onto the plane of the differentiated positions and remove points that lie outside the ellipse.

Since the areas that are sampled might become very big, it is important to make sure that the collected irradiance estimates are somehow related to where the ray comes from. To ensure this, the irradiance estimate should only be used, if it is facing the direction of the ray – that is, if the dot product between the ray's direction and the normal of the object that the irradiance estimate is less than zero.

4.RESULTS

Scenes with lights pointing directly at a surface are difficult to model using photon mapping, since the light on the wall will have sharp edges and in practice work as a direct light. For the images in figures 4 and 5 the larger part of the room is illuminated by indirect lighting, while only some of the ceiling and parts of the walls are illuminated directly. A setup like this makes it very important where the final gathering samples are made, since the indirect lighting changes radically over very small distances. With few rays for sampling in final gathering, the advantage of using ray differentials should be more apparent since it will cover larger areas, and get a better estimate of the lighting.

ISBN 978-80-86943-16-9

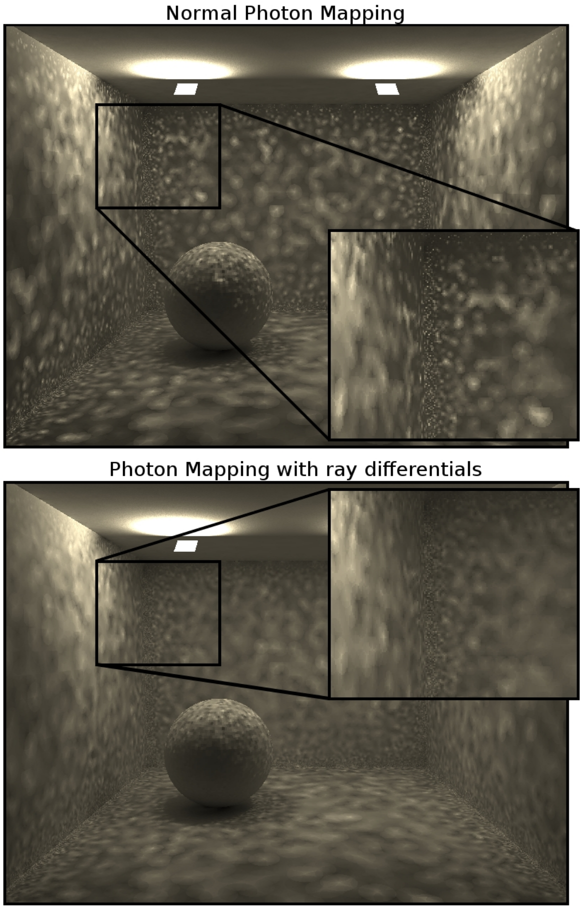

Figure 4: A photon mapped room without and with ray differentials. 200 samples have been used for final gathering.

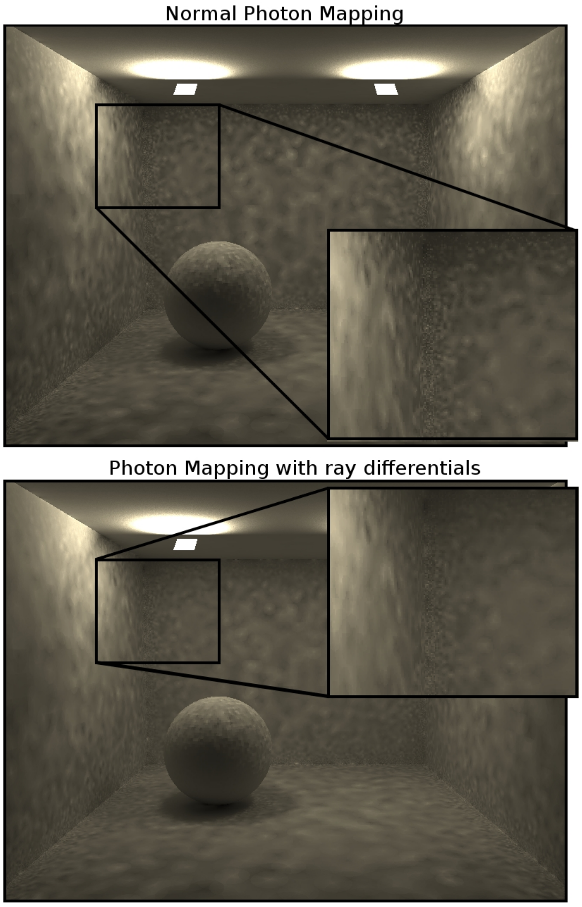

Figure 5: A photon mapped room without and with ray differentials. 800 samples have been used for final gathering.

5.BENCHMARK

To test the impact in speed of doing final gathering using ray differentials, with an irradiance estimate mipmap, the scene from figure 4 has been used – though with a single large light, rather than two small ones. The benchmark has been performed on a 2 Ghz Intel core 2 duo with 2GB RAM.

A test has been performed with 1,000,000 photons, 500,000 photons with an initial search radius of 0.05, and with 1,000,000 photons with an initial search radius of 0.1. Changing the amount of photons and the search radius will impact on how long it takes to find them in the Kd-tree – using a unsuitably low search radius will force more searches in the Kd-tree in order to obtain a sufficient amount of photons. The values for rendering time include the total time for creating the image, excluding emitting the photons, since this is the same for both techniques.

The difference in the tables note the deviation in time between normal photon mapping and our method.

Search radius of 0.05, it takes 10.1 seconds to emit photons. The creation of the mipmap takes a total of 15.6 seconds for this setup.

1,000,000 photons	Normal Photon mapping	Using ray differentials	
Samples	Total	Total	Difference
200	**124.7 s**	**113.9 s**	**8.7 %**
800	**476.3 s**	**361.5 s**	**24.1 %**
1800	**970.8 s**	**759.4 s**	**21.8 %**

Search radius of 0.05, it takes 5.0 seconds to emit photons. The creation of the mipmap takes a total of 10.2 seconds for this setup.

500,000 photons	Normal Photon mapping	Using ray differentials	
Samples	Total	Total	Difference
200	**133.8 s**	**107.0 s**	**27.7 %**
800	**488.6 s**	**347.1 s**	**31.1 %**
1800	**1056.6 s**	**730.6 s**	**31.6 %**

Search radius of 0.1, it takes 5.0 seconds to emit photons. The creation of the mipmap takes a total of 5.7 seconds for this setup.

500,000 photons	Normal Photon mapping	Using ray differentials	
Samples	Total	Total	Difference
200	**117.9 s**	**103.3 s**	**17.2 %**
800	**420.4 s**	**353.2 s**	**17.3 %**
1800	**904.6 s**	**775.4 s**	**14.9 %**

6.DISCUSSION

Using ray differentials to determine the sampling size reduces the noise that appears in final gathering, when using only few samples. When more samples are used, the results converge. This is partially because, as more samples are used, the scene is sampled more thoroughly, but also because, as more samples are used, the rays' footprints will become smaller than the minimum sampling radius and the advantage of using ray differentials disappears.

The usage of ray differentials, and the creation of a mipmap introduces some extra necessary calculations, but the usage of the mipmap ensures that the total rendering time is actually shorter than when doing normal final gathering. While there will be a short peak in memory usage when creating the mipmap, this will hardly ever be a problem. When creating the mipmap it is possible to save each layer on disk sequentially, since the data of each layer is mutually independent. When the first layer of the mipmap has been created, the photon map can be discarded. Depending on the distribution of the photons, the total memory usage will almost always be smaller than that of the photon map, due to the removal of superfluous irradiance estimates.

The results show that there is still quite a bit of noise, even when using ray differentials for final gathering, even in simple scenes. This is because the entire visible area is not sampled, as intended. The visible hemisphere is sampled using a stratified random [Jen03a], which is a common way to ensure an even yet random distribution. This, however, means that some areas will still be sampled multiple times, while others will stay unsampled. It is however generally a bad idea to sample the world using a static pattern, as these are bound to produce unwanted patterns. A stratification that also produced a suitable initial spread for the rays would most likely eliminate most of the noise.

While this method is good for fairly homogenous objects, more complex objects may not yield as good a result. This is because ray differentials only take the triangle that is hit into account, so even if the footprint spans several triangles only the one the 1D ray hits is used for determining the orientation of the surface. A different method that took more triangles into account might yield a better result.

Since the proposed method only changes the irradiance representation and adds some information to the ray it is compatible with other recent optimizations such as metropolis photon sampling [Fan05a] or other similar importance sampling techniques [Chr03a].

While this method bares a resemblance to irradiance atlases [Chr04a], the irradiance estimated by our method is more accurate, since it uses several irradiance estimates in an ellipse, rather than selecting the nearest estimate covering the same general area.

7.CONCLUSION

In this paper we have introduced a method for improving the indirect irradiance estimate produced by the final gathering step of photon mapping, by keeping track of each ray's footprint with ray differentials. By using an irradiance estimate mipmap the speed as well as image quality is not only preserved but improved. The use of ray differentials ensures that the method remains versatile for use with objects of varying representations.

8.REFERENCES

[Arv86a] Arvo, J, Backward Ray Tracing, Developments in Ray Tracing, pp. 259-263,1985. ACM Siggraph ’85 Course Notes, 1986

[Chr03a] Christensen, P. H., Adjoints and Importance in Rendering: an Overview, IEEE Transactions on Visualization and Computer Graphics (TVCG), volume 9, number 3, pp. 329-340, 2003

[Chr04a] Christensen, P. H., and Batali, D., An Irradiance Atlas for Global Illumination in Complex Production Scenes, Rendering Techniques 2004 (Proceedings of the Eurographics Symposium on Rendering 2004), pp. 133-141. Eurographics / ACM, 2004

[Chr99a] Christensen, P. H., Faster Photon Map Global Illumination, Journal of Graphics Tools,volume 4, number 3, pp. 1-10. ACM, 1999

[Fan05a] Fan, S., Chenney, S., and Lai, Y., etropolis Photon Sampling with Optional User Guidance, Eurographics Symposium on Rendering, 2005

[Ige99a] Igehy, H., Tracing Ray Differentials, Computer Graphics (Proc.SIGGRAPH 99), s. 179-186, 1999

[Jen03a] Jensen, H. W., Monte Carlo Raytracing, Siggraph'03 course 44, pp. 149-161, 2003

[Jen95a] Jensen, H. W., and Christensen, N. J., Photon Maps in Bidirectional Monte Carlo Ray Tracing of Complex Objects, Computers & Graphics vol. 19 (2), s. 215-224, 1995

[Kaj86a] Kajiya, J. T., The Rendering Equation, Vol. 20, no. 4, pp. 143-150, 1986

[Laf93a] Lafortune E. P., and Willems, Y. D., Bi-Directional Path Tracing, Proceedings of Third International Conference onComputational Graphics and VisualisationTechniques, pp. 145-153, 1993