EG UK Theory and Practice of Computer Graphics (2006)
M. McDerby, L. Lever (Editors)

# Structured image techniques for efficient high-fidelity graphics

X. Yang† and K. Debattista and A. Chalmers

Department of Computer Science , University of Bristol, United Kingdom

**Abstract**

*Global illumination rendering in real-time for high-fidelity graphics remains one of the biggest challenges for computer graphics in the foreseeable future. Recent work has shown that significant amounts of time can be saved by selectively rendering in high quality only those parts of the image that are considered perceptually more important. Regions of the final rendering that are deemed more perceptually important can be identified through lower quality, but rapid, rasterisation rendering. By exploiting this prior knowledge of the scene and taking advantage of image space based algorithms to concentrate rendering on the more salient areas higher performance rendering may be achieved. In this paper, we present a selective rendering framework based on ray tracing for global illumination which uses a rapid image preview of the scene to identify important image regions, structures these regions and uses this knowledge to direct a fraction of the rays traditionally shot. The undersampled image is then reconstructed using algorithms from image processing. We demonstrate that while this approach is able to significantly reduce the amount of computation it still maintains a high perceptual image quality.*

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

## 1. Introduction

Global illumination, using ray tracing techniques, is well known as being effective for creating accurate and high-fidelity representations of complex objects and space. However, existing global illumination applications suffer from expensive computational cost thus precluding them from rendering complex scenes in real time. Recently, a substantial amount of research has been done to find ways to improve this performance.

Sampling strategies identify important areas of the scene and focus computational effort on these reducing overall computational cost [Guo98, BWG03, LDC05a]. Typically, such algorithms generate some form of *map* before the actual rendering process and use this map to guide the sample allocation, for example [SDL*05, LDC06]. Prior knowledge of the scene can help produce a reasonable sample allocation. The problem of these algorithms is, however, that they mainly use linear interpolation which is unable to produce high quality images from very sparse samples. This can be improved by using smarter reconstruction algorithms [FJP02].

Calculations are typically computationally much less expensive compared to intersection tests in the object space. Thus many image processing based techniques have been used by global illumination algorithms with some good success [PP99, CSSD94, KW93]. However, these algorithms depend solely on image processing techniques and thus neglect the substantial amount of knowledge about the scene which can be obtained from the object space.

In this paper we present a novel framework which combines both the object space based rendering strategies, with image space based reconstruction algorithms. First of all we generate a quick scene preview on a GPU [LDC05b] and from this we produce an edge map as the guide for our sampling strategy. We have modified the lighting simulation sys-

† Email: yang@cs.bris.ac.uk

tem *Radiance* with a quadtree structure to organize the samples.

The rest of the paper is structured as follows. In Section 2 we discuss related previous work in both selective rendering and reconstruction algorithms. In Section 3 we describe our framework and the implementation of the algorithm. In Section 4 we evaluate our framework with both timing and visual perception metrics and present the results. Finally, in Section 5 we conclude and suggest possible future avenues of research.

## 2. Previous Work

In this section we discuss related work in sampling strategy for selective renderers and previous research on reconstruction algorithms.

### 2.1. Selective Rendering

Ray tracing based selective renderers are able to concentrate rays to the areas of importance and consequently save significant amount of computational resources.

Mitchell [Mit87] was the first to use perceptual techniques for their ray tracer. Painter and Sloan [PS89] used a Kd-tree for storing samples and identifying where the next samples should be shot for their progressive and adaptive ray-tracing based renderer. Myszkowski [Mys98] used a visual difference predictor [Dal93] to identify when to stop rendering in their progressive global illumination renderer. Guo *et al.* [Guo98] demonstrated a sparse sampling algorithm to reduce the number of rays needed to be shot to gain speed. However human visual system is highly sensitive to object silhouettes and shadow edges. This approach is prone to producing blurred and aliased outcomes because of lack of discontinuity information. An *Edge-and-Point(EPI)* based rendering algorithm was subsequently developed to exploit both advantages of sparse sampling and edge detection algorithms [BWG03]. They rasterised an *Edge-and-Point* image to illustrate the location of the discontinuities in the object space and interpolate the image from the information about edges in a sub pixel precision accuracy level. Yee *et al.* [YPG01] extended the saliency map [IKN98] to combine motion for their selective renderer. Cater *et al.* [CCW03] used task distractors for driving their selective renderer. Longhurst *et al.* in [LDC05a] and [LDC06] presented an approach to produce a quick preview of the scenes on a GPU, and then use this preview as the guideline for a selective renderer. They also exploited the edge information to overcome aliasing problems in global illumination. This approach still uses a relative large number of samples. We will compare this approach with ours in Section 5.

### 2.2. Reconstruction methods

Linear interpolation algorithms are the most widely used method to reconstruct high quality images from sparse samples. Many approaches such as [BWG03] use linear interpolation to produce the desired images. The cost of ray intersection tests for ray tracing is far more expensive compared with image space reconstruction. Therefore it's worth exploring optimal reconstruction algorithms for under-sampled images.

Wavelets have a great reputation for good quality of reconstruction, Piertrek [PP99] presented an approach to reduce variance in Monte Carlo integration based on the wavelet densities. Christensen *et al.* [CSSD94] used wavelet analysis to provide an efficient solution method for global illumination with glossy and diffuse reflections. Generally speaking, wavelets could be an ideal solution for sparsely sampled image reconstruction, however, the wavelet approach is not a suitable approach for our current renderer due to the high computational cost.

Knutsson *et al.* [KW93] presented a reconstruction algorithm by building a certainty map. This represents the certainty value of each pixel and interpolates surrounding pixel values from the information derived from this certainty map. This algorithm is termed as Normalised Convolution and is an extremely efficient approach for non-uniformed sparsely sampled image reconstruction. Furthermore, this algorithm is easy to implement and can easily be done in a separate pass from the rendered image.

## 3. Framework

An overview of our framework can be seen in Figure 1. The framework is composed of five stages. The first step uses a rapid image generation using rasterisation on fast graphics hardware to produce an image estimate. The subsequent step is to identify the more salient parts of the image using an image-space algorithm also performed on graphics hardware. The next step structures the important regions based on image-space subdivision of the saliency map. These regions are then used to direct where the selective renderer should shoot the rays. Finally an image reconstruction technique is used to produce a final image.

In the remainder of this section we provide a further overview of this pipeline for a general selective renderer. In Section 4 we describe our implementation of the framework.

### 3.1. Image preview and guideline image

Image preview produces basic scene descriptions such as direct lighting information, basic shadow, color and shape of the objects. The preview is calculated on modern graphics hardware therefore it is highly efficient. Once the image estimate is produced, a number of image space algorithms may be used to capture the salient features. For instance, an edge map can be used for antialiasing purposes [LDC05a] or a saliency map can be used for perceptual based selective renderer [LDC06]. This guideline information is pro-

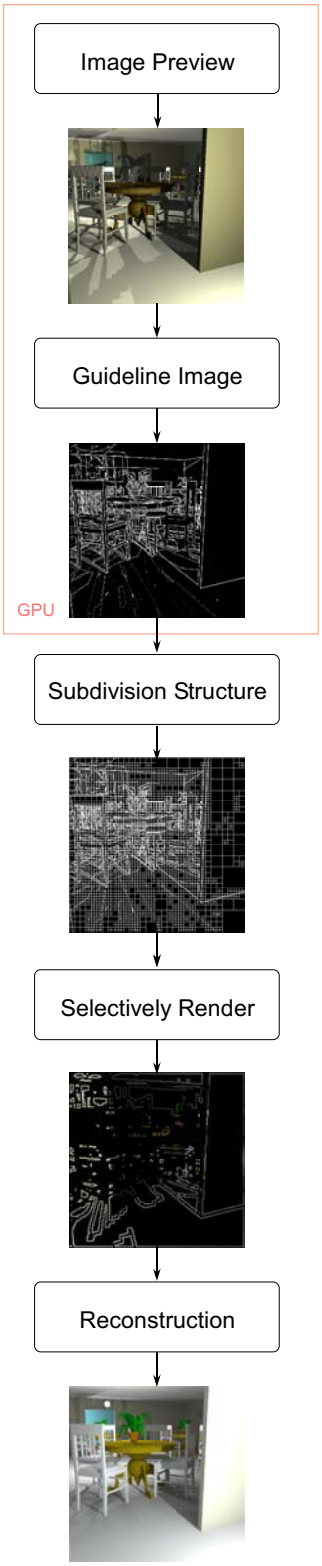


**Figure 1:** *The overview of our framework.*

duced from the previous scene preview also using fast graphics hardware. This information serves as a general guide for the rendering process and is able to save a significant amount of time over an equivalent algorithm performed on the CPU [LDC06].

### 3.2. Selective rendering

The guideline image is used as an input into the selective renderer. The selective renderer initially subdivides the salient features into a two-dimensional data structure using an image-space subdivision method. Subsequently this data structure is used to guide the selective renderer and focus rays onto the important areas.

### 3.3. reconstruction algorithm

Our framework uses a simple and effective normalized convolution for the final reconstruction step. The normalized convolution consists of two convolutions and one division, which is efficient and simple. The formula is shown in Figure 2 below.

$$S \otimes G / C \otimes G$$

S: Sparse samples
G: Gaussian filter
C: Certainty map

**Figure 2:** *The formula of Normalized Convolution.*

It first convolutes the sparsely sampled image with a Gaussian filter. One thing worth mentioning here is that the Gaussian filter needs to be large enough to recover the missing information in the image. The second step of the algorithm is to build a certainty map for the sparsely sampled image. It give a certainty value of one if there is a sample in this pixel/sub pixel and the certainty value will be zero if no sample exists for that pixel. Then, this algorithm will convolute the certainty map with the same size gaussian filter as the above to gain a convoluted certainty map. In the end, the convoluted sparse samples divide the convoluted certainty map to gain the normalized value for the interpolated image.

## 4. Implementation

In this section we will describe the implementation details of the framework. For this paper we used *Snapshot* [LDC05c] to generate the scene preview and then produced the guideline information using the edge detector technique from *Snapshot* [LDC05a]. Subsequently we implemented the subdivision structure using a quadtree data structure and as a selective renderer we use `srpict` [LDC06] based on a

modified version of the lighting simulation package *Radiance* [War94]. Finally, we programmed the normalised convolution algorithm in Matlab.

### 4.1. Snapshot and edge map generator

*Snapshot* is used for our rapid image generation using a rapid image preview from the scene description on a GPU using OpenGL. Images produced by *Snapshot* are generally composed of rasterised triangles using point light sources, Furthermore, *Snapshot* uses further techniques to attempt to simulate high-fidelity rendering. Shadow mapping [Wil78] via cubic texture maps are used to simulate shadows from point light sources. Cubic environment maps are used to provide approximate specular reflections, and stencil shadowing to accurately account for planar mirrors [Kil99]. Furthermore similar shaders as those used in *Radiance* and subsequently in our selective renderer have been implemented in *Snapshot* to generate images more similar to the final image. The complexity of *Snapshot* depends on the scene complexity in terms of light sources and geometry, however frame rates of upwards of 30fps may be achieved for relatively complex scenes [LDC06]. *Snapshot* is also capable of identifying salient features in an image in one of two ways: either using an edge detection algorithm or using more complex saliency maps [IKN98] and importance maps [SDL*05]. In this paper we only use edge detection based on the Sobel filter since edge areas are more prone to aliasing problems and more likely to be perceived by the human visual system. Compared with other edge detectors, Sobel filter is effective at producing clear and thick edges for both objects and shadows.

### 4.2. Selective renderer

In order to demonstrate the effectiveness of our framework we have extended the modified version of *Radiance* `srpict` to support the image space subdivision by means of a quadtree. We term our new renderer `qsrpict`. The original `srpict` renderer is a selective adaptive renderer based on stratified sampling [LDC06]. The `qsrpict` renderer inputs the edge map produced by *Snapshot* and adaptively divides the image in order to select locations where to shoot the rays. The edge map indicates the areas most likely to suffer from the aliasing problems. `qsrpict` replaces the stratified sampling algorithm with a quadtree data structure to distribute a small number of samples in the area where they are most needed and effective, in this case one for each leaf node of the quadtree. A series of detailed comparisons between the two renderer (`qsrpict` and `srpict`) are provided in Section 5. A view of quadtree of the Cornell Box is shown in Figure 3.

### 4.3. Reconstruction

The final step of this framework is to reconstruct a high fidelity image from the sparse samples. In our system we used

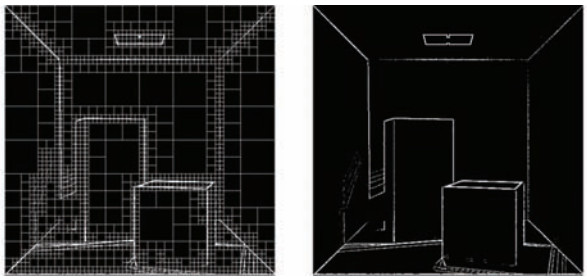

**Figure 3:** *The quadtree image(left) and the edge map(right) for the Cornell Box.*

normalized convolution to reconstruct the under-sampled images. We implemented this part of the framework in Matlab [HH00]. Matlab is an optimized mathematical package containing many built in image processing functions which are extremely fast. The computational time of the reconstruction is almost negligible in Matlab. In the future we plan to implement this part of process on a GPU to improve performance similar to that in [DWWL05].

## 5. Results

To evaluate the effectiveness of our approach, we tested the framework with two scenes. We produced three sets of still images for each scene. The first set are our baseline references that were computed using `srpict` running in non-selective mode, at one ray per pixel. The second group is the images we generated using our framework rendered in `qsrpict`. The third group where the images rendered in `srpict` using adaptive rendering that shot one ray at the corner of a $4 \times 4$ box and depending on the edge map either interpolates the result for the intermediate pixels or subdivides and recurs the operation until at most one ray per pixel is shot. Default *Radiance* settings for global illumination were used for all images. All three sets of images were calculated and rendered on the same machine, which has an Intel Pentium IV 2.8 GHZ with 1G RAM running on Linux.

**Scene1. Cornell Box:** This is a fairly straight forward and commonly used test scene for computer graphics. It demonstrates many effects in physically based rendering techniques, such as reflectance and shadows.

**Scene2. Table Test Room:** This is a complex scene containing quite a lot of edge information. We deliberately used this scene to test the effectiveness of the quadtree in terms of sample distribution and the quality of the reconstruction algorithm.

Table 1 shows the timing information and the number of rays used for the reference images, our `qsrpict` and the original `srpict`. The GPU based preview and edge map generation is very fast and computational time is negligible

(*ca.* 0.030 seconds for a size of 750*750 image). Furthermore, the reconstruction takes very little time in Matlab. For this reason, the timing information for the `qsrpict` only includes the time spent on the quadtree based selective renderer. For the same reason, the timing for the original `srpict` only includes the selective rendering time. The images rendered in our framework only used between 8%- 15% of pixels. Our framework gains significant speedups from the reference images and speedups from adaptive `srpict`.

Figure 4 illustrates the resultant images rendered in our framework and each of the steps.

### 5.1. Perceptual metrics validation

To further investigate the perceptual quality of the image produced by our framework, the Visual Difference Predictor (VDP) [Dal93] was used to compare the reference high quality image with the image from our framework. The VDP is a computer simulation of the human visual system and its ability to perceive differences in the images. Primarily each image is treated individually to remove frequencies that would not be witnessed by a human observer. The remaining differences are then weighted over both frequency and orientation channel, the metric is designed to highlight differences near and below a just perceivable threshold. To compare our framework with the original `srpict`, we performed two pairs of VDP comparison for each scene. The first pair is the comparison between the reference images and the images generated by adaptive `srpict`; the second one is the comparison between the reference images and the images generated by `qsrpict`. The VDP comparison results for both testing scenes are show in Table 2. The results show that the VDP values differ by less than 1%, and therefore the difference in the images rendered in our framework are almost imperceptible from the reference images.

## 6. Conclusion and Future Work

In this paper we have presented a framework which uses a rapid image estimate coupled with a subdivision structure-based selective renderer to produce sparse samples and reconstruct a high quality image from those sparse samples. The resulting constructed image created from the sparse samples using normalized convolution algorithm is of a high-quality and perceptually equivalent to the high quality reference image. Our framework only requires small amount of samples (10% samples) to produce this level of quality and significantly speeds up the rendering system. In the future, we plan to extend our framework to support saliency maps [IKN98] and importance maps [SDL*05], and will explore the possibility of using our framework for temporal coherence to reuse samples for animations similar to the work in [DWWL05]. Furthermore, we will offload more of the work on the GPU particularly the reconstruction algorithm on a GPU to gain further speedups.

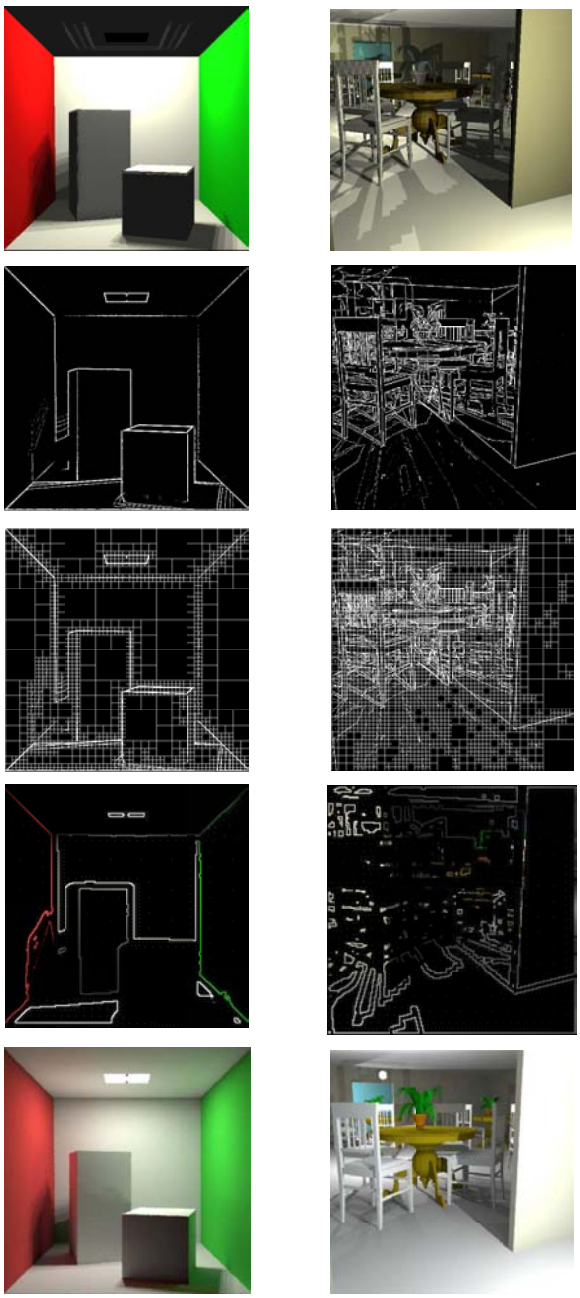

**Figure 4:** *The images for our test scenes. Left hand side demonstrates the Cornell Box and the right hand side the Room Scene. From top to bottom we show: the rapid image preview of the image, the edge map, a visualisation of the generated quadtree, the samples shot and finally the reconstructed image.*

| Scenes | Cornell Box | | Room Scene | |
|---|---|---|---|---|
| | Time | Number of Rays | Time | Number of Rays |
| Reference images | 5.76 | 262144 | 75.75 | 262144 |
| Adaptive srpict | 1.56 | 72947 | 54.17 | 149039 |
| qsrpict | 0.44 | 21445 | 10.6 | 33322 |

**Table 1:** *Timing comparison for the scenes.*

| Scenes | Cornell Box | | Room Scene | |
|---|---|---|---|---|
| | average VDP error | the number of error pixels | average VDP error | the number of error pixels |
| Reference images vs. Adaptive srpict | 0.22% | 0.28% | 0.17% | 0.62% |
| Reference images vs. qsrpict | 0.84% | 0.90% | 0.58% | 0.37% |

**Table 2:** *VDP comparison for the scenes.*

## 7. Acknowledgements

We would like to thank Peter Longhurst for the use of the Snapshot application and Room Test scene. We would also like to thank the RoD project and all those who have contributed ideas to this work.

## References

[BWG03] BALA K., WALTER B., GREENBERG D. P.: Combining edges and points for interactive high-quality rendering. *ACM Trans. Graph. 22*, 3 (2003), 631–640.

[CCW03] CATER K., CHALMERS A., WARD G.: Detail to Attention: Exploiting Visual Tasks for Selective Rendering. In *Proceedings of the Eurographics Symposium on Rendering* (2003), pp. 270–280.

[CSSD94] CHRISTENSEN P. H., STOLLNITZ E. J., SALESIN D. H., DEROSE T. D.: Wavelet Radiance. In *Fifth Eurographics Workshop on Rendering* (Darmstadt, Germany, 1994), pp. 287–302.

[Dal93] DALY S.: The visible differences predictor: an algorithm for the assessment of image fidelity. 179–206.

[DWWL05] DAYAL A., WOLLEEY C., WATSON B., LEUBKE D.: Adaptive frameless rendering. In *Proceedings of the Eurographics Symposium on Rendering* (2005), pp. 265–275.

[FJP02] FREEMAN W. T., JONES T. R., PASZTOR E. C.: Example-based super-resolution. *IEEE Computer Graphics and Applications 22*, 2 (2002), 56–65.

[Guo98] GUO B.: Progressive radiance evaluation using directional coherence maps. In *SIGGRAPH '98: Proceedings of the 25th annual conference on Computer graphics and interactive techniques* (New York, NY, USA, 1998), ACM Press, pp. 255–266.

[HH00] HIGHAM D. J., HIGHAM N. J.: *MATLAB Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 2000.

[IKN98] ITTI L., KOCH C., NIEBUR E.: A model of Saliency-Based Visual Attention for Rapid Scene Analysis. In *Pattern Analysis and Machine Intelligence* (1998), vol. 20, pp. 1254–1259.

[Kil99] KILGARD M.: Creating reflections and shadows using stencil buffers. In *GDC 99* (1999).

[KW93] KNUTSSON H., WESTIN C.-F.: Normalized and differential convolution: Methods for interpolation and filtering of incomplete and uncertain data. In *Proceedings of Computer Vision and Pattern Recognition (Proceedings of IEEE Computer Society Conference onComputer Vision and Pattern Recognition'93)* (New York City, USA, June 16–19 1993), pp. 515–523.

[LDC05a] LONGHURST P., DEBATTISTA K., CHALMERS A.: Analytic antialiasing for selective high fidelity rendering. In *SIBGRAPI 2005* (October 2005).

[LDC05b] LONGHURST P., DEBATTISTA K., CHALMERS A.: Snapshot: A rapid technique for driving a selective global illumination renderer. In *WSCG 2005 SHORT papers proceedings* (2005).

[LDC05c] LONGHURST P., DEBATTISTA K., CHALMERS A.: Snapshot: A rapid technique for driving a selective global illumination renderer. In *WSCG 2005 SHORT papers proceedings* (2005), pp. 81–84.

[LDC06] LONGHURST P., DEBATTISTA K., CHALMERS A.: A gpu based saliency map for high-fidelity selective rendering. In *AFRIGRAPH 2006 4th International Conference on Computer Graphics, Virtual Reality, Visualisation and Interaction in Africa* (January 2006), ACM SIGGRAPH, pp. 21–29.

[Mit87] MITCHELL D. P.: Generating antialiased images at low sampling densities. In *SIGGRAPH '87* (1987), ACM Press, pp. 65–72.

[Mys98] MYSZKOWSKI K.: The Visible Differences Predictor: Applications to global illumination problems. In

*Eurographics Workshop on Rendering* (1998), pp. 223–236.

[PP99] PIETREK G., PETER I.: Adaptive wavelet densities for monte carlo ray tracing. In *WSCG'99 Conference Proceedings* (1999), Skala V., (Ed.).

[PS89] PAINTER J., SLOAN K.: Antialiased ray tracing by adaptive progressive refinement. In *SIGGRAPH '89* (1989), ACM Press, pp. 281–288.

[SDL*05] SUNDSTEDT V., DEBATTISTA K., LONGHURST P., CHALMERS A., TROSCIANKO T.: Visual attention for efficient high-fidelity graphics. In *Spring Conference on Computer Graphics (SCCG 2005)* (May 2005).

[War94] WARD G. J.: The radiance lighting simulation and rendering system. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques* (1994), ACM Press, pp. 459–472.

[Wil78] WILLIAMS L.: Casting curved shadows on curved surfaces. In *SIGGRAPH '78: Proceedings of the 5th annual conference on Computer graphics and interactive techniques* (1978), ACM Press, pp. 270–274.

[YPG01] YEE H., PATTANAIK S., GREENBERG D.: Spatiotemporal sensitivity and Visual Attention for efficient rendering of dynamic Environments. In *ACM Transactions on Computer Graphics* (2001), vol. 20, pp. 39–65.