



**HAL**  
open science

# A perceptual stopping condition for global illumination computations

Nawel Takouachet, Samuel Delepouille, Christophe Renaud

► **To cite this version:**

Nawel Takouachet, Samuel Delepouille, Christophe Renaud. A perceptual stopping condition for global illumination computations. Spring Conference on Computer Graphics, Apr 2007, Budmerice, Slovakia. hal-03479491

**HAL Id: hal-03479491**

**<https://hal.science/hal-03479491>**

Submitted on 14 Dec 2021

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# A perceptual stopping condition for global illumination computations

Nawel Takouachet\*

LIL

Université du Littoral Côte d'Opale

Samuel Delepoulle†

LIL

Université du Littoral Côte d'Opale

Christophe Renaud‡

LIL

Université du Littoral Côte d'Opale

## Abstract

The aim of realistic image synthesis is to produce high fidelity images that authentically represent real scenes. As these images are produced for human observers, we may exploit the fact that not everything is perceived when viewing scene with our eyes. Thus, it is clear that taking advantage of the limited capacity of the human visual system (HVS), can significantly contribute to optimize rendering software.

Global illumination methods are used to simulate realistic lighting in 3D scenes. They generally provide a progressive convergence to high-quality solution. One of the problem of such algorithms is to determine a stopping condition, for deciding if calculations reached a satisfactory convergence allowing the process to terminate.

In this paper, we propose and we discuss different solutions to this important problem. We show different techniques based on the Visual Difference Predictor (VDP) proposed by Daly [Daly 1993] to define a perceptual stopping condition for rendering computations. We use the VDP to measure the perceived differences between rendered images and to guide the Path Tracing rendering to satisfy a perceptual quality. Also, in a controlled experimental setting with real subjects, we validate our results.

**CR Categories:** I.3.3 [Picture image generation]: Display algorithms, Viewing algorithms—; J.4 [Social and Behavioral Sciences]: psychology

**Keywords:** global illumination, computer vision, image synthesis, perception-driven rendering, Visible Difference Predictor (VDP)

## 1 Introduction

In recent years, research in computer graphics has been directed toward new perception-driven techniques for rendering. The main purpose is to be able to replace human observer by a model that simulates the same behavior as his visual system.

Our perceptive performance is controlled by our threshold sensitivity to various features such as: contrast, spatial frequencies, orientations, shapes, colors... Therefore, when looking at our visual environment, we do not perceive all its components with the same performance. Some objects automatically and effortlessly "pop out" from their surroundings. Also, some details in image can not be perceived. Thus, limits of the human visual system (HVS) capacity

can be exploited to parameter and to monitor rendering algorithms based on perceived visual quality. In addition, such techniques can perform important improvements by focusing computations only on features judged as visible by the human visual system simulation [Mitchell 1987; Tumblin and Rushmeier 1993; Myszkowski 1998; Bolin and Meyer 1999; Volevich et al. 2000; Yee et al. 2001; Reddy 2001; Sundstedt et al. 2004; Farrugia and Péroche 2004; Longhurst et al. 2006]

Ideally, global illumination rendering process can be stopped when the current image quality becomes indistinguishable from the fully converged solution from a human observer.

However, in practice, this fully converged solution is unknown and it is exactly what we want to get at the end of rendering calculations. It is clear that in this stage of rendering process, the decision of a stopping condition appears as an important problem: How and what criteria must be taken into account to define a stopping condition of rendering calculations ?

In this paper, we suggest a perception-driven rendering technique to resolve this problem. We investigate, through different approaches, the application of the Visible Difference Predictor developed by [Daly 1993], to avoid unnecessary computations.

Our global illumination rendering algorithm is based on Monte Carlo Path Tracing (MCPT) method. The stochastic sampling used by this technique generates color variations perceived as noise [Szirmay-Kalos 1998]. Though, the fact of progressive adding samples reduces this noise. Consequently it increases the quality of the calculated image toward a final satisfactory quality solution.

We investigate the VDP to measure the quality of the rendering images. We decide to terminate computations when they attain a satisfactory perceptual quality even though they contain imperceptible MCPT noise.

The remainder of this paper is structured as follows. We will review previous work on perceptual models and perception-driven rendering techniques in Section 2. Then, in section 3, we will describe our experimental method used for psychophysical validation. Thereafter, in section 4, we will detail and discuss our approaches to decide a stopping condition of rendering computations based on the VDP. In addition, we will compare our results with Myszkowski's approach [Myszkowski 1998]. Finally, in section 5, we will suggest some ideas for future work.

## 2 Previous Work

Standard techniques of realistic rendering rely on physically accurate simulations of light reflection and surface interactions. At the final, the result of these rendering algorithms will be presented to human observers, who will examine images with they HVS limits

Recently, considerable efforts have been devoted to understanding and simulating the visual system behavior (HVS). These researches showed that HVS can fail to perceive some physics inaccuracy and be very sensitive to others [Koch and Ullman 1985; Daly 1993; Sarnoff Corporation 1997; Ferwerda et al. 1997; Niebur and Koch

\*e-mail: takouachet@lil.univ-littoral.fr

†e-mail: delepoulle@lil.univ-littoral.fr

‡e-mail: renaud@lil.univ-littoral.fr

1998; Sundstedt et al. 2004; Itti 2005; Navalpakkam and Itti 2006]. Therefore, new perceptually-based techniques of realistic rendering aim for perceptual accuracy rather than physical realism.

Various perceptual models have been proposed. One kind uses perceptual quality metrics which modulate the capacity of visual system to detect difference between images. The Visible Differences Predictor (VDP) [Daly 1993] predicts the probability of detection of differences between two images. It is based on frequency decompositions that extract some visual properties such as sensitivity to contrast, orientations ... The VDP operates only on the achromatic channel and is very costly to calculate. Thereafter, [Tolhurst et al. 2005] proposed a new multi-resolution visual difference prediction model that takes into account colors.

The Sarnoff Visual Discrimination Model (VDM) [Sarnoff Corporation 1997] is also a famous image comparison metric. It operates in the spatial domain. The key elements of the VDM include spatial resampling wavelets. It was later modified to the Sarnoff JND metric for color video. For the reason that the VDM operates in the spatial domain, it avoids the expensive Fourier transformations which take up to 40 % of the time consumed in the Daly VDP [Li 1997].

[Yee 2004] has proposed an abridged version of the VDP in the same way as Ramasubramanian [Ramasubramanian et al. 1999], in which they drop the orientation computation when calculating spatial frequencies. They also extend the VDP by including the color domain in computing the differences. This new version of the VDP increases speed over the full VDP, which is essential specially when testing a set of images sequence.

These perceptual metrics have shown much utility in Computer graphics. [Volevich et al. 2000; Myszkowski 1998] used the VDP to provide the quantitative measures of perceptual convergence by predicting and estimating the perceivable differences between the intermediate and final images. Also, the performance of different methods of global illumination rendering was measured and compared using the VDP.

[Farrugia and Péroche 2004] proposed a perceptually-based rendering method. The rendering accuracy needed per pixel is adjusted according to a perceptual adaptive metric based on the Multi-scale Model of Adaptation and Spatial Vision [Pattanaik et al. 1998]. Their goal is to improve rendering time without the viewer being aware of the difference between the refined image and the reference image computed with a standard global illumination method.

Another kind of perceptual rendering techniques is based on model of visual attention. [Yee et al. 2001; Sundstedt et al. 2004; Longhurst et al. 2006] adapted various visual attention models [Treisman and Gelade 1980; Niebur and Koch 1998; Koch and Ullman 1985; Niebur et al. 2001; Navalpakkam and Itti 2006; Itti 2005; Marendaz et al. 2005] in order to accelerate the global illumination computation in dynamic environments. These models are used to dictate where computational effort should be spent during the lighting solution. Rendering system then spent more time to calculate the observer's regions of interest.

### 3 Experimental study

In this section, we explain our experimental procedure used to evaluate and to validate our VDP application. The goal of this study is to measure the threshold of sampling from which observers do not see any differences between current image and the reference one (calculated with very high quality). The results of this experimen-

tation will be then compared with the values given by the VDP from different applications that we will detail after.

In all of our experiments, we used a Monte Carlo Path Tracing with the next event algorithm (MCPT) and the number of samples the we have used for each pixel is identical. The searched threshold will then be defined as the number of samples needed.

#### 3.1 First study

In a first stage of this experimental study, we need to get an interval of the perceptual quality threshold for our test scenes. This will allow us to reduce the set of values in which researching the threshold value.

We presented couples of images (reference, imageI),  $I$  in  $[50..4000]$  *samples/pixel* with a step of 50 *samples/pixel* and a reference image calculated with 10000 *samples/pixel*. In order to reinforce statistical results, we show the series of images in ascending and descending order. The results of these first experiments showed that the perceptual threshold for our test scenes is in  $[100..2500]$  *samples/pixel*. This interval corresponds to 50% of observers who do not detect any differences from reference images.

#### 3.2 Second study

The second stage consists in determining more precisely the perceptual threshold values. In addition, we would associate each value of the perceptual threshold in the previous interval to a percentage of observers.

The subjects of the experiment were 12 undergraduate students at the Institute of Technologies at Calais. The average age is 21.0 with a standard deviation of 3.35.

Subjects were placed at a distance of 0.5 m of the display (19 flat panel display at resolution 1280x1024, 300 cd/m<sup>2</sup>). First a calibration of the screen is done. Then subjects were asked to answer if they can perceive any difference between a reference and a target images. The reference and the target images were shown simultaneously as long as necessary for the subject to take the decision to answer. Subject gives his answer by pressing a button on the computer interface with the mouse.

The reference image is computed with 10000 *samples/pixel* and target images are computed successively with 100, 500, 900, 1300, 1700, 2100, 2500 *samples/pixel*. The same images with the same sampling density were shown two times to reduce errors.

The order of presentation is counterbalanced with a pseudo-randomized schedule, and the reference image is shown randomly at the left or at the right of the target image. This procedure is conform to the measure of the differential threshold with the method of constant stimuli.

Table 1 shows the probability of every subject to perceive a difference between target and reference images (see Figure 2 for a view of scenes). We used 4 different scenes, each scene being calculated with two intensities. This made it possible to study the effect of the intensity on subjects behavior and VDP results.

The probability to perceive a difference is supposed to follow a sigmoid law so we can approximate the results with a logit function [Wichmann and Hill 2001]. The general expression of this equation 1 is :

$$p(x) = \frac{c \cdot e^{a \cdot x + b}}{1 + e^{a \cdot x + b}} \quad (1)$$

Samples set	100	500	900	1300	1700	2100	2500
BAR 1	1	0,96	0,88	0,67	0,48	0,52	0,44
BAR 2	1	1	0,92	0,83	0,69	0,75	0,71
BOX 1	1	0,67	0,31	0,29	0,15	0,15	0,13
BOX 2	1	0,92	0,71	0,63	0,35	0,58	0,38
OCULIST 1	1	1	1	0,83	0,77	0,56	0,5
OCULIST 2	1	1	1	0,88	1	0,92	0,38
PLANTS 1	1	0,75	0,44	0,35	0,29	0,4	0,25
PLANTS 2	1	0,5	0,42	0,44	0,4	0,33	0,21
Global	1	0,85	0,71	0,61	0,52	0,53	0,37

Table 1: Probability to perceive differences.

Table 2 shows the value a, b and c computed by a logit regression, r being the regression coefficient.

	a	b	c	r
BAR 1	-0,00200	4,007	1,010	-0,940
BAR 2	-0,00178	4,558	1,010	-0,907
BOX 1	-0,00226	2,677	1,010	-0,841
BOX 2	-0,00186	3,489	1,010	-0,873
OCULIST 1	-0,00233	5,415	1,010	-0,940
OCULIST 2	-0,00178	5,474	1,010	-0,767
PLANTS 1	-0,00185	2,743	1,010	-0,797
PLANTS 2	-0,00173	2,457	1,010	-0,752
Global	-0,00173	3,276	1,010	-0,867

Table 2: The estimate coefficients of a logit regression for each scene used in tests.

Figure 1 shows plots from eq. 1 with values from table 2. We have calculated the value of the sampling perceptual threshold which corresponds to 50% and 10% of the observers which do not see any differences between reference image and target ones.

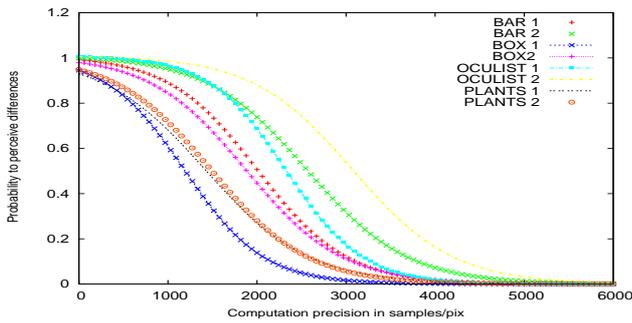


Figure 1: The probability to perceive differences for the different scenes used in tests.

These data will be used to evaluate our different results and to validate the approaches that we detail in the next section (see table 3).

	$p = 0,5$	$p = 0,1$
BAR1	2013,31	3104,74
BAR2	2578,98	3808,71
BOX 1	1193,52	2157,21
BOX 2	1882,12	3053,19
OCULIST 1	2333,57	3270,78
OCULIST 2	3093,18	4322,24
PLANTS 1	1490,05	2667,46
PLANTS 2	1435,82	2701,31
Global	1903,55	3166,91

Table 3: The Threshold of sampling which corresponds to 50% and 10% of the subjects who do not see differences between reference image (10000 *samples/pixel*) and target ones.

## 4 VDP: application for global illumination rendering

### Algorithm 4.1: ITERATRENDER()

*step* : Samples added per iteration  
*CS* : Current samples/pixel  
*MaxSampling* : Max number of Samples

**comment**: Initial rendering

*image2*  $\leftarrow$  Rendering(*step*)

*CS*  $\leftarrow$  *step*

**repeat**

$\left\{ \begin{array}{l} \textit{image1} \leftarrow \textit{image2} \\ \textit{image2} \leftarrow \textit{image1} + \textit{Rendering}(\textit{step}) \\ \textit{CS} \leftarrow \textit{CS} + \textit{step} \end{array} \right.$

**until** *CS* = *MaxSampling*

The VDP takes into account important aspects of the HVS that reduce the sensitivity to error. First, the sensitivity of HVS to contrast decreases with increasing light levels. Then, the sensitivity to perceive details changes with spatial frequency [Rushmeier et al. 1997; Rushmeier et al. 1995]. The last effect, masking [Ferwerda et al. 1997], takes into account the variations in sensitivity due to the signal content of the background.

We integrate the VDP to a MCPT global illumination algorithm. The VDP is applied to measure the quality of rendered images. It monitors the rendering process and decides if the calculated quality needs more perceptual accuracy.

However, the VDP requires two images for comparison. The best choice for the VDP comparison would be the final converged image and a partially converged one. However, the former is obviously not available. So what stage of computation and which image should be used for VDP comparison to define a stopping condition of rendering computations? In this section, we will describe and discuss some answers to this question. We have chosen to use a fast version of the VDP [Yee 2004] which operates in colors space.

All VDP applications that we will present are based on an MCPT iterative rendering algorithm. Before detailing them, we explain first the basic idea of an iterative rendering independently of perceptual techniques (algorithm 4.1). We set a maximum number of cast samples as a stopping condition. For every new iteration, we do not restart computations but we add some additional samples to the current samples set.



Figure 2: The different scenes of tests. The OCULIST scene is provided by courtesy of *Musée départemental d'archéologie de Bavay*. All images are calculated in resolution  $512 \times 512$ .

#### 4.1 Approach 1: successive comparison

In this first approach, we suggest to replace the maximum number of cast samples used as stopping condition by a perceptual threshold defined by VDP differences (algorithm 4.2). We evaluate a VDP comparison between two successive renderings and we stop the calculations whenever the measure of the perceptual error between the two images becomes smaller than some threshold.

**Algorithm 4.2:** SUCCOMPARE( $step$ )

$step$ : Samples added per iteration  
 $CS$ : Current samples/pixel

**comment:** Initial rendering  
 $image2 \leftarrow Rendering(step)$   
 $CS \leftarrow step$   
**repeat**  
 $\left\{ \begin{array}{l} image1 \leftarrow image2 \\ image2 \leftarrow image1 + Rendering(step) \\ CS \leftarrow CS + step \end{array} \right.$   
**until**  $VDP(image1, image2) \leq threshold$

**Discussion:** This method gives good results if the chosen sampling step is sufficiently large. Otherwise, the new additional samples do not sufficiently change the image. Consequently, the VDP do not detect any visible differences and the rendering calculations terminate before achieving a perceptually good quality.

Figure 3 shows an example of this problem. The first image is calculated with  $150 \text{ samples/pixel}$ . We add  $10 \text{ samples/pixel}$  to get the second image. The perceptual difference between these two images using the VDP is null. However after a number of iterations we get the third image calculated with  $800 \text{ samples/pixel}$  which is clearly very different from the two previous ones. This, can explain

the fluctuations and the irregularities observed in the curves of the figure 4.a and the figure 4.b.

We have evaluated the successive rendering with VDP comparison for the BAR 1 scene with different sampling steps. Curves presented in the figure 4 showed that the stopping rendering stage depends on sampling step. If the step is small, the rendering calculations terminate rapidly. With a  $50 \text{ samples/pixel}$  sampling step the differences become null at  $400 \text{ samples/pixel}$ . With a larger sampling step, *i.e.*  $100 \text{ samples/pixel}$ , the VDP continued to detect differences until  $1500 \text{ samples/pixel}$ . These results show the strong dependence that exists between the capability of the VDP to detect differences between successive rendering and the sampling step used.

Furthermore, the value of the 'good' sampling step is scene dependent. Tests made on all our scenes have showed that its value can change according to scene properties. As an example, the VDP fails to detect differences between successive renderings, at  $300 \text{ samples/pixels}$  for the OCULIST scene, at  $1500 \text{ samples/pixel}$  for BAR 1 and  $2200 \text{ samples/pixel}$  for BOX 1 (see figure 4.b). Choosing  $100 \text{ samples/pixel}$  as the value of sampling step, permits a good stopping stage only for the BOX 1 scene as compared to the experimental perceptual thresholds (Table 3). As a result, we can not define a general sampling step which gives a good stopping condition for all scenes.

#### 4.2 Approach 2: multiple comparisons

As already said, the successive rendering with VDP comparison approach is very dependent to the sampling step value and the scene properties. We have searched for another method less sensitive to the previous constraints.

The main operation in this approach is to compute multiple VDP comparisons rather than one. We thus calculate a window of  $n$  VDP measures.

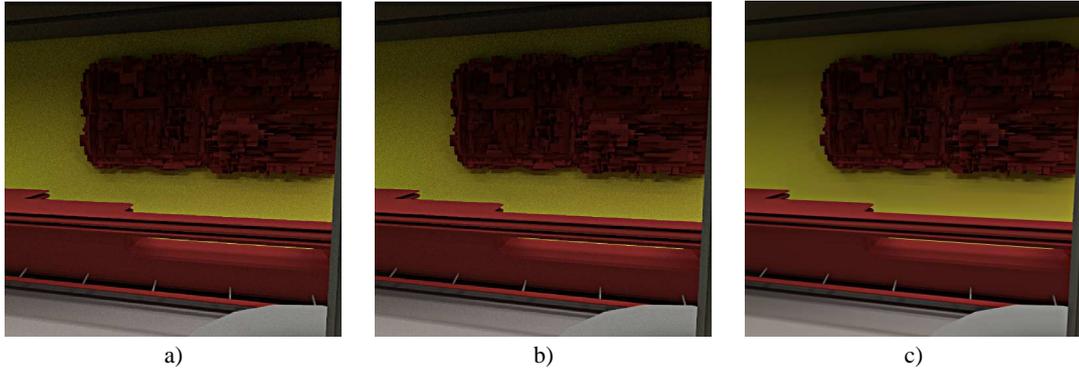


Figure 3: the image a) is calculated with 150 *samples/pixel*. The image b) is calculated with 160 *samples/pixel*. The image c) is calculated with 800 *samples/pixel*.

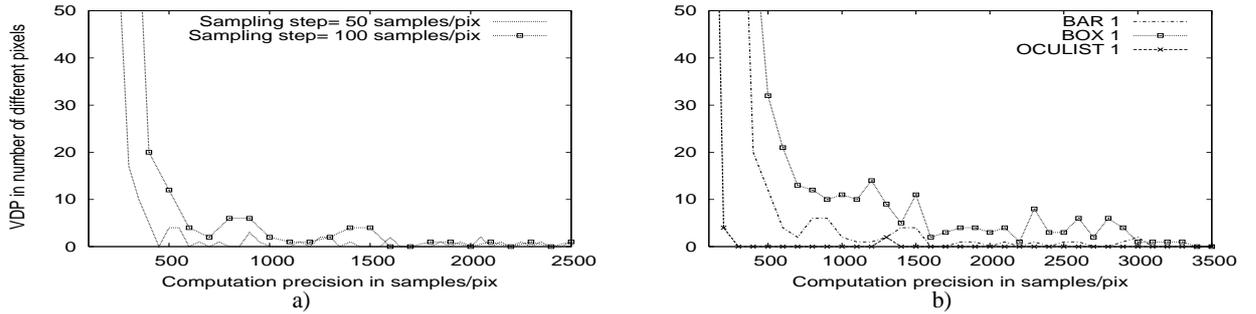


Figure 4: a) Successive rendering with VDP comparison using different sampling steps, for the BAR 1 scene. b) Successive rendering with VDP comparison for different scenes, using a sampling step = 100 *samples/pixel*.

For every window, images from images  $I_m, I_{m+1}, \dots, I_{m+n}$  are compared to the reference image  $I_{(m+n)+1}$ . The stopping condition of calculations can be decided when the  $n$  VDP values are lower than the perceptual VDP threshold. Otherwise, we displace the window of comparison and we repeat the same treatment (Algorithm 4.3).

**Discussion:** The iterative rendering with multiple VDP comparisons gives a better convergence as compared to the successive comparison. It permits to verify the variations of the visible perceptual differences during several rendering iterations (number of images in the comparison window). The rendering will be stopped when these variations are lower than some threshold. It means that the rendering computations attain a stable perceptual accuracy.

However, this technique is still dependent on the sampling step chosen (see figure 5.a). The larger the sampling steps, the better the accuracy of stopping condition. It attains 1300 *samples/pixel* with 50 *samples/pixels* sampling step and 2100 *samples/pixel* with 100 sampling step. Similarly the stopping rendering stage increases with the window comparison size: 1300 *samples/pixel* with 5 images in the comparison window and 2100 *samples/pixel* with 8 images (figure 5.b).

Figure 5.c shows that if we take a window comparison size = 5 images and the sampling step = 100 *samples/pixel*, we get a good stopping condition for our test scenes (BAR 1 : 2100, BOX 1 : 1900 and OCULIST 1 : 1700 *samples/pixel*). For the BAR 1 and the BOX 1 scenes, it corresponds to up 50% of the observer experimental results and to 30% for the OCULIST 1 scene (Table 3). We obtained similar results with the other scenes (up to 50% of ob-

server experimental results). The major drawback of this method is however its computation time. At every iteration we need  $n$  VDP executions which generates an important cost over the full computations rendering time.

**Algorithm 4.3:** MULTCOMPAR(*step*)

```

step : Samples added per iteration
CS : Current samples/pixel
n : window size
Images[n] : Array of rendering images
Diff[n] : Array of VDP differences
stop : Boolean to stop rendering

repeat
  RENDERWINDOW(Images[], CS, step)
  IRef ← Rendering(CS + step)
  CS ← CS + step
  VDPWINDOW(Images[], IRef, Diff[])
  stop ← true
  i ← 0
  while ((i < n) and (stop = true))
    if (Diff[i] > threshold)
      do {
        then stop ← false
        i ← i + 1
      }
  until stop = true

```

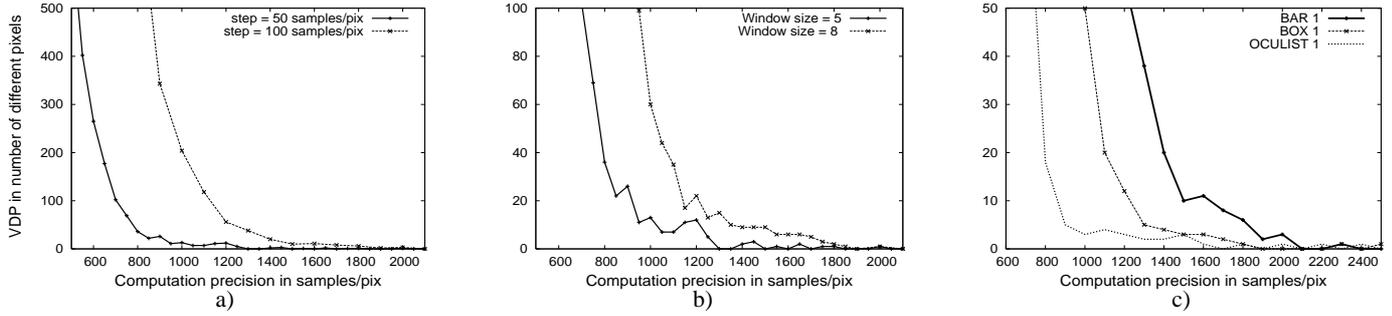


Figure 5: a) VDP multiple comparisons with the BAR 1 scene. a) Window comparison size =5, step sampling = (50,100 samples/pixel). b) Window comparison size =(5,8), step sampling = 50 samples/pixel. c) Rendering with multiple VDP comparisons using a sampling step = 100 samples/pixel and window comparison = 5 images for BAR1, BOX 1 and OCULIST 1 scenes.

### 4.3 Approach 3: reverse comparison process

#### Algorithm 4.4: REVERSECOMPAR(step)

```

step ← Samples added per iteration
CS ← Current samples/pixel
n ← window size
Images[n] ← Array of rendering images
Diff[n] ← Array of VDP differences
stop ← Boolean to stop rendering

stop ← false
comment: Initial rendering
IRef ← Rendering(CS + step)
CS ← CS + step
repeat
  { RENDERWINDOW(Images[], CS, step)
  VDPWINDOW(Images[], IRef, Diff[])
  Adjust ← REGRESSION(Diff[])
  if Adjust > ε
    then
      { image ← Rendering(CS + step)
      CS ← CS + step
      INSER(image, Images[])
    else stop ← true
until stop = true

```

When we start the calculation of the global illumination solution, we do not possess the fully converged image. This image is theoretically required in order to be compared to the current image and to decide to terminate the rendering process when they both become perceptually identical.

We propose in this approach to replace this image with an initial rendering (obtained after some calculation iterations). Generally, this rendering has a lower quality than the final image. The solution then consists to reverse the original comparison process (Algorithm 4.4). Thus, instead to stop the calculations when the difference between the reference and the current image is negligible, it will be terminated when this difference attains a peak and stabilizes. That means that the additional sampling does not change the perceptual quality of the calculated image. Thus, calculation is considered as fully converged.

**Discussion:** As shown in the figure 6.a, when the rendering calculations converge, the perceptual differences between the initial rendering and the current image increases. This difference attains a maximal and becomes then linear.

This approach is less sensitive to the sampling step than the previous ones. Figure 6.a shows that the two obtained curves with the BAR 1 scene and different sampling steps, have the same form and start being linear at the same sampling stage. The two curves do not describe the same VDP difference values since the reference image is not the same for the two measure sets. When adding 50 samples per pixel at each iteration, we computed the reference image with 50 samples per pixel. 100 samples per pixels were used for the second reference image, used when adding 100 samples per pixel at each iteration. If the same reference image was used the two curves would have been quite equals. This should indicate that this approach is relatively not sensitive to the initial sampling of the reference image.

The problem now is to be able to detect when the linear part of each curve starts and consequently when to stop the iterative rendering process. One of the statistical techniques which can be used when evaluating the linear relationship between two variables is the simple linear regression [Hidioglou and Patak 2004]. This methodology is widely used in business, the social and behavioral sciences, the biological sciences, and many other disciplines. It is used to estimate a model which relays two variables and predicts response of one of variable from the other. A linear regression line has an equation of the form  $Y = aX + b$ .

In our case, we would modulate the relationship between the sampling set and the VDP differences. We use the linear regression to estimate the director coefficient of the line which passes by the slope and to predict the function linearity.

We calculate the regression coefficient of the N VDP values computed between the initial rendering and the  $m$  to  $m+n$  images. At every iteration, we insert the new VDP value to recalculate the regression coefficient  $a$ . We decide to stop the rendering computations when the regression coefficient becomes less than some small value ( $\geq 0$  and  $\leq \epsilon$ ).

In the figure 6.b we show results of the linear regressions. The regressions values increase when the perceptual differences decrease. It becomes steady and near zero when the rendering calculations attain a peak of a perceptual accuracy.

[Myszkowski 1998] has proposed a method to estimate the perceptual differences between the current rendering ( $imageI$ ) and the

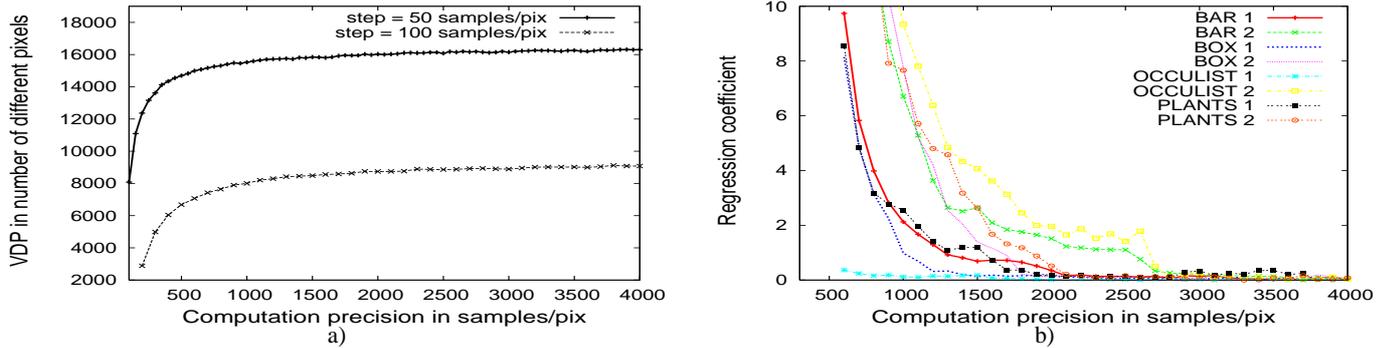


Figure 6: a) Results with the BAR scene using different sampling steps: 50 and 100 *samples/pixel*. b) Regression coefficient for test scenes, the number of VDP values using to calculate the regression coefficient = 5. The reference image is calculated with 100 *samples/pixel* and the sampling step for each next iteration is 100 *samples/pixel*.

	Experimental 50%	Experimental 10%	Myszkowski approach	Reverse comparison
BAR 1	2013,31	3104,74	3800	3300
BAR 2	2578,98	3808,71	4000	3700
BOX 1	1193,52	2157,21	2600	2300
BOX 2	1882,12	3053,19	3400	3000
OCCULIST 1	2333,57	3270,78	1800	1600
OCCULIST 2	3093,18	4322,24	3400	3100
PLANTS 1	1490,05	2667,46	3000	2700
PLANTS 2	1435,82	2701,31	3200	2900

Table 4: Comparison between experimental thresholds and the VDP reverse comparison results

image completely converged (*imageC*). He suggests that the VDP (*imageI, imageC*)  $\simeq$  VDP (*imageI, image $\alpha$ I*) with  $\alpha \simeq 0.5$ .

We have evaluated this method to obtain the value of  $I$  for which the VDP (*image, image0.5I*)  $\simeq 0$  using 100 *samples/pixel* for the sampling step. Table 4 shows the results obtained by our approach evaluated with a regression coefficient  $\leq 0.1$ . As compared to the experimental thresholds, it gives a good estimation of the perceptual threshold for the scenes : BAR 1, BAR 2, BOX 1, BOX 2, PLANTS 1 and PLANTS 2. Myszkowski’s approach overestimates the perceptual threshold for these scenes. For the OCCULIST 1 and OCCULIST 2 scenes, both methods fail to provide a good estimate of the sampling threshold. Note that this problem is not related to the two approaches but rather to the use of the VDP. Thus some new experiments should be performed in order to understand the reasons of this result.

From a memory consumption point of view, Myszkowski’s approach requires to store several images during computation for comparisons during the iterative rendering process. In our approach only the reference and current images have to be stored, reducing considerably the memory consumption. Furthermore our approach allows us to reduce from 8 to 14% the number of samples to be used. Because sampling is computationally demanding, our approach allows us to obtain converge image fastly.

## 5 Conclusion

Taking into account the limits of the human visual system (HVS) capacity can significantly improve realistic rendering algorithms by guiding the computations to achieve perceptual accuracy.

In this work, we have investigated the application of a fast version

of the VDP [Yee 2004] to monitor the Path Tracing global illumination rendering. We have proposed and we have discussed different approaches to define a perceptual stopping condition of Path Tracing computations. We have shown that the *successive comparison* method depends strongly to the sampling step and to the scene properties. The *multiple comparisons* approach can be parameterized to give a good stopping condition but it is costly to calculate. Then, we have demonstrated the *reverse comparison* method as less sensitive to the choice of the application parameters. In addition, the experimental results showed that our reverse comparison with linear regression provides a stopping rendering computations more efficiently and more accurately than Myszkowski’s approach [Myszkowski 1998].

However, we do not get an optimal result with all the test scenes. The calculated threshold for the OCCULIST scenes is lower than the experimental one both for our approach and Myszkowski’s one. This requires a validation of the VDP parameters for this kind of applications. This problem could be explained by the large number of textures that appear in this scene. Thus we think that the VDP could be more sensitive to masking than the HVS. Furthermore [Longhurst and Chalmers 2004] have shown through an experimental set that the VDP does not give the optimal responses.

In future work we thus plan to perform more systematic psychophysical experiments with more test scenes. This should further improve our experimental results and should be used to calibrate and to validate the VDP for realistic rendering applications. Additionally, we plan to combine a perceptual quality metric with a model of visual attention to guide a selective global illumination rendering in static and dynamic environments.

## References

- BOLIN, M., AND MEYER, G. 1999. A visual difference metric for realistic image synthesis. In *Human Vision and Electronic Imaging IV*, SPIE, B. E. Rogowitz and T. N. Pappas, Eds., vol. 3644.
- DALY, S. 1993. The visible differences predictor: an algorithm for the assessment of image fidelity. In *Digital images and human vision*, vol. 4, 124–125.
- FARRUGIA, J.-P., AND PÉROCHE, B. 2004. A progressive rendering algorithm using an adaptive perceptually based image metric. *Comput. Graph. Forum* 23, 3, 605–614.
- FERWERDA, J. A., PATTANAİK, S. N., SHIRLEY, P., AND GREENBERG, D. P. 1997. A model of visual masking for computer graphics. *Computer Graphics 31*, Annual Conference Series, 143–152.
- GIBSON, S., AND HUBBOLD, R. J. 2000. A perceptually-driven parallel algorithm for efficient radiosity simulation. *IEEE Transactions on Visualization and Computer Graphics* 6, 3, 220–235.
- HIDIROGLOU, M. A., AND PATAK, Z. 2004. Domain estimation using linear regression. *Survey Methodology* 30, 67–78.
- ITTI, L. 2005. Models of bottom-up attention and saliency. In *Neurobiology of Attention*, G. R. L. Itti and J. K. Tsotsos, Eds. Elsevier, San Diego, CA, Jan, 576–582.
- KOCH, C., AND ULLMAN, S. 1985. Shifts in selective visual attention: Towards the underlying neural circuitry. *Human Neurobiology* 4 (January), 219–227.
- LI, B., 1997. An analysis and comparison of two visual discrimination models.
- LONGHURST, P., AND CHALMERS, A. 2004. User validation of image quality assessment algorithms. In *TPCG '04: Proceedings of the Theory and Practice of Computer Graphics 2004 (TPCG'04)*, IEEE Computer Society, Washington, DC, USA, 196–202.
- LONGHURST, P., DEBATTISTA, K., AND CHALMERS, A. 2006. 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*, ACM SIGGRAPH, 21–29.
- MARENDAZ, C., CHAUVIN, A., AND HÉRAULT, J. 2005. A causal link between scene exploration, local saliency and scene context. *J. Vis.* 5, 8 (9), 919–919.
- MITCHELL, D. P. 1987. Generating antialiased images at low sampling densities. In *SIGGRAPH '87: Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, ACM Press, New York, NY, USA, 65–72.
- MYSZKOWSKI, K. 1998. The visible differences predictor: applications to global illumination problems. In *Eurographics Rendering Workshop*, 233–236.
- NAVALPAKKAM, V., AND ITTI, L. 2006. An integrated model of top-down and bottom-up attention for optimal object detection. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2049–2056.
- NIEBUR, E., AND KOCH, C. 1998. Computational architectures for attention. In *The Attentive Brain*, R. Parasuraman, Ed. MIT Press, Cambridge, Massachusetts, 164–186.
- NIEBUR, E., ITTI, L., AND KOCH, C. 2001. Controlling the focus of visual selective attention. In *Models of Neural Networks IV*, L. V. Hemmen, E. Domany, and J. Cowan, Eds. Springer Verlag, Aug.
- PATTANAİK, S. N., FERWERDA, J. A., FAIRCHILD, M. D., AND GREENBERG, D. P. 1998. A multiscale model of adaptation and spatial vision for realistic image display. *Computer Graphics 32*, Annual Conference Series, 287–298.
- RAMASUBRAMANIAN, M., PATTANAİK, S. N., AND GREENBERG, D. P. 1999. A perceptually based physical error metric for realistic image synthesis. In *Siggraph 1999, Computer Graphics Proceedings*, Addison Wesley Longman, Los Angeles, A. Rockwood, Ed., 73–82.
- REDDY, M. 2001. Perceptually optimized 3d graphics. *IEEE Comput. Graph. Appl.* 21, 5, 68–75.
- RUSHMEIER, H., WARD, G., PIATKO, C., SANDERS, P., AND RUST, B. 1995. Comparing real and synthetic images: Some ideas about metrics. In *Eurographics Rendering Workshop 1995*.
- RUSHMEIER, H., BARRETT, H., RHEINGANS, P., USELTON, S., AND WATSON, A. 1997. Perceptual measures for effective visualizations. In *VIS '97: Proceedings of the 8th conference on Visualization '97*, IEEE Computer Society Press, Los Alamitos, CA, USA, 515–517.
- SARNOFF CORPORATION, 1997. Sarnoff JND vision model algorithm description and testing, August. VQEG.
- SUNDSTEDT, V., CHALMERS, A., CATER, K., AND DEBATTISTA, K., 2004. Top-down visual attention for efficient rendering of task related scenes.
- SZIRMAY-KALOS, L. 1998. Stochastic methods in global illumination - state of the art report. Tech. Rep. TR-186-2-98-23, Institute of Computer Graphics and Algorithms, Vienna University of Technology, Favoritenstrasse 9-11/186, A-1040 Vienna, Austria, Aug. human contact: technical-report@cg.tuwien.ac.at.
- TOLHURST, D. J., RIPAMONTI, C., PÁRRAGA, C. A., LOVELL, P. G., AND TROSCIANKO, T. 2005. A multiresolution color model for visual difference prediction. In *APGV '05: Proceedings of the 2nd symposium on Applied perception in graphics and visualization*, ACM Press, New York, NY, USA, 135–138.
- TREISMAN, A. M., AND GELADE, G. 1980. A feature-integration theory of attention. *Cognit Psychol* 12, 1 (January), 97–136.
- TUMBLIN, J., AND RUSHMEIER, H. 1993. Tone reproduction for realistic images. *IEEE Comput. Graph. Appl.* 13, 6, 42–48.
- VOLEVICH, V., MYSZKOWSKI, K., KHODULEV, A., AND KOPYLOV, E. A. 2000. Using the visual differences predictor to improve performance of progressive global illumination computations. *ACM Transactions on Graphics* 19, 2 (April), 122–161.
- WICHMANN, F. A., AND HILL, N. J. 2001. The psychometric function: I. fitting, sampling, and goodness of fit. *Perception and Psychophysics* 63, 8 (November), 1293–1313.
- YEE, H., PATTANAİK, S., AND GREENBERG, D. P. 2001. Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. In *ACM Transactions on Graphics*. ACM Press, 39–65.
- YEE, H. 2004. A perceptual metric for production testing. *Journal of graphics tools* 9, 4, 33–40.