# Perceptually-informed accelerated rendering of high quality walkthrough sequences

Karol Myszkowski, Przemyslaw Rokita and Takehiro Tawara
University of Aizu, Aizu Wakamatsu 965-8580, Japan

**Abstract.** In this paper, we consider accelerated rendering of walkthrough animation sequences using combination of ray tracing and Image-Based Rendering (IBR) techniques. Our goal is to derive as many pixels as possible using inexpensive IBR techniques without affecting the animation quality. A perception-based spatio-temporal Animation Quality Metric (AQM) is used to automatically guide such a hybrid rendering. The Pixel Flow (PF) obtained as a by-product of the IBR computation is an integral part of the AQM. The final animation quality is enhanced by an efficient spatio-temporal antialiasing, which utilize the PF to perform a motion-compensated filtering.

## 1 Introduction

Rendering of animated sequences proves to be a very computation intensive task, which in professional production involves specialized rendering farms designed specifically for this purpose. While the progress in efficiency of rendering algorithms and increasing processor power are very impressive, with a similar pace the requirements imposed by the complexity of rendered scenes also increase. Effectively, rendering timings reported for the final antialiased frames are still counted in tens of minutes or hours.

It is well-known in the video community that the human eye is less sensitive to higher spatial frequencies than to lower frequencies, and this knowledge was used in designing video equipment [9]. It is also the conventional wisdom that the requirements imposed on the quality of still images must be higher than for images used in an animated sequence. Another intuition is that the quality of rendering can be usually relaxed as the velocity of a moving object (image pattern) increases. These observations are confirmed by systematic psychophysical experiments investigating the sensitivity of the human eye for various spatio-temporal patterns [16, 28]. For example, the perceived sharpness of moving low resolution (or blurred) patterns increases with velocity, which is attributed to the higher level processing in the visual system [30]. This means that all techniques that attempt to speed up rendering of every single frame separately cannot account for the eye sensitivity variations resulting from the temporal considerations. Effectively, computational efforts can be easily wasted on processing image details which cannot be perceived in the animated sequence. In this context, a global approach involving both spatial and temporal dimensions appears promising [23] and relatively unexplored research direction.

This research is an attempt to develop a framework for the perceptually-informed accelerated rendering of antialiased animated sequences. In our approach, computation is focused on those selected frames (keyframes) and frame fragments (inbetween frames), which strongly affect the whole animation appearance by depicting image details readily perceptible by the human observer. All pixels related to these frames and frame fragments are computed using a costly rendering method (we use ray tracing as the final pass of our global illumination solution), which provides images of high quality. The remaining pixels are derived using an inexpensive method (we use IBR techniques [21, 20, 25]). Ideally, the differences between pixels computed using the slower and faster methods should not be perceived in animated sequences, notwithstanding that such differences can be readily seen when the corresponding frames are observed as still images. The spatio-temporal perception-based quality metric for animated sequences is used to guide frames computation in the fully automatic and recursive manner. The special care is taken for efficient reduction of spatial and especially annoying temporal artifacts, which occasionally can be observed even in the professionally produced animated sequences.

In our approach, the Pixel Flow (PF) computed as the motion vector field is the key point of the overall animated sequence processing. It is computed using the IBR techniques, which guarantee its very good accuracy and high speed of processing for the synthetic images[1]. The PF is used in our technique in three ways:

- To reproject pixels from the ray traced keyframes to the image-based inbetweens.
- To improve temporal considerations of our perception-based animation quality metric.
- To enhance animation quality by performing antialiasing based on motion-compensated filtering.

Obviously, the best cost-performance is achieved when the PF is used in all three processing steps. However, since all these steps are only loosely coupled, and the costs of computing PF are very low, other scenarios are also possible e.g., fully ray traced animation can be filtered with motion compensation.

In this paper, we narrow our discussion to the production of high-quality walkthrough animations (only camera animation is considered), although some of solutions proposed by us can be used in a more general animation framework (refer to [23] for discussion of problems with global illumination in this more general case). We assume that walkthrough animation is of really high quality involving complex geometry and global illumination solutions, and thus it incurs significant costs for a single frame rendering (e.g., about 170 minutes in the example chosen as a case study in this research [1]). We make also other reasonable assumptions such as: animation path and all camera positions are known in advance, ray tracing (or other high quality rendering method) for selected pixels is available, depth (range) data for every pixel are inexpensive to derive for every frame (e.g., using z-buffer), and object identifiers for every pixel can be easily accessed for every frame (e.g., using item buffer).

In the following section, we discuss previous work on improving performance of animation rendering and perception-based video quality metrics. Then we describe efficient methods of inbetween frames computation we have used in our research. Section 4 describes our 3D antialiasing technique based on the motion-compensated filtering. In Section 5 we present our animation quality metric. Section 6 and the accompanying Web page [1] show results obtained using our approach. Finally, we conclude this work.

## 2 Previous work

In this research our objective is the reduction of time required for rendering frames, in particular, inbetween frames, which can be derived from the high-quality keyframes.

[1]For the natural image sequences the optical flow can be derived [27], which is more costly and usually far less accurate.

In this context, we discuss previous work on exploiting various forms of frame-to-frame coherence to speedup rendering and enhance image quality. To our knowledge, a method that automatically selects keyframes minimizing distortions visible by the human observers has not been presented yet. We review the perceptually-informed video quality metrics which could be used to guide rendering of inbetween frames.

### 2.1 In-between frame generation

Frame-to-frame coherence has been widely used in computer animation to speedup computations. Here we limit our discussion to techniques dealing with camera animation. Early research focused mostly on speeding up ray tracing by interpolating images for views similar to a given keyframe [2, 3]. These algorithms involved costly procedures for cleaning up image artifacts such as gaps between pixels (resulting from stretching samples reprojected from keyframes to inbetween frames), and occlusion (visibility) errors. For example, Adelson and Hodges [2] traced rays between the viewing position and the reprojected intersection point for every pixel to check for visibility errors. In this respect, recently developed IBR techniques are more efficient: the 3-D warping and warp ordering algorithms proposed by McMillan [21] efficiently solve the problem of visibility, and the "splatting" technique developed by Shade *et al.* [25] is fast and fills well gaps between resampled pixels.

Special treatment is required for objects occluded in the reference image (keyframe) and visible in the derived images (inbetween frames). Shade *et al.* [25] and Lischinski [18] warped a number of reference frames into the selected view (usually it is one of the reference views) and built the Layered Depth Image (LDI) structure in which the subsequent layers of occluded pixels were stored. The LDI structure is very compact because redundancies resulting from multiple reference images are removed, but rendering of these reference images can be costly. Mark *el al.* [20] and Darsa *et al.* [7] proposed techniques which fit well to the walkthrough applications. Compositing between just two warped frames computed along the animation path is performed. All algorithms discussed reduce significantly the problem of occlusions, however, some perceptible errors are still likely to appear when an object occluded in all reference frames becomes visible in desired views.

Processing of specular surfaces for inbetween frames using interpolation techniques is a hard problem. For a majority of ray tracing-based interpolation techniques all pixels depicting objects with specular properties are recomputed [2]. On the other hand, a vast majority of IBR techniques was developed for diffuse environments [21, 20, 25] and only few solutions handling more general reflectance functions were proposed. The light field [17] and Lumigraph [13] techniques are suitable for rendering of glossy objects. A dense grid of images is used to store lighting outgoing in all directions from a bounded region of space. It is not clear how to handle occlusions between objects if navigation is freely allowed within this bounded region [18]. This problem can be solved and more crisp images can be obtained using the surface light field approach [22] in which geometry is explicitly represented. View-dependent lighting is stored in a huge volume of textures attached to every glossy surface. In general, all light field-like techniques require a huge number of images that must be precomputed and stored to achieve a reasonable quality of derived images. This can be very costly, in particular, in applications dealing with synthetic images of high quality. Even with a huge number of precomputed images, sharp mirror reflections are hard to obtain in this framework.
The most promising in this context is technique proposed by Lischinski and Rappoport [18] who capture directional distribution of reflected lighting in multiple specialized

LDI-like structures which make possible recomputation of glossy and specular effects for changing views fully within the IBR framework. This solution seems to be especially suitable for interactive applications dealing with compact (localized in space) objects and requiring full freedom in selecting views.
In our application, scenes might be of substantial geometrical extent and visual complexity (textures and geometrical details) which would require LDIs of high resolution (this means high rendering cost to prepare such LDIs) to secure the high quality of rendering and to cover the full scene extent. On the other hand, since only the predefined set of views is processed during walkthroughs, these requirements can be relaxed for some LDIs storing the view-dependent light component, which are referenced less frequently or are not referenced at all. Automatic generation of an adaptive LDI representation accommodating for these requirements still remains an open research problem [18].

The Multiple Viewpoint Rendering technique [15] seems to be an interesting alternative to the traditional rendering, but to make it practical in our walkthrough application further research is required to enable less constrained camera motion within large environments.

### 2.2 Pixel flow applications in animation rendering

The Pixel Flow found many successful applications in video signals processing [27] and animated sequences compression [14]. Also, some applications in computer animation have been shown. Zeghers *el al.* [31] used the linear interpolation between densely placed keyframes, which was performed along the PF direction. To avoid visible image distortions only a limited number of inbetween frames could be derived (the authors showed examples for one or three consecutive inbetweens only). Shinya [26] proposed the motion-compensated filtering as the antialiasing tool for animation sequences.
Zeghers *el al.* and Shinya used animation information to compute the PF between images, and visibility computations were performed explicitly for every pixel. Using IBR techniques the PF computation is greatly simplified for walkthrough sequences, and the visibility is handled automatically.

### 2.3 Video quality metrics

In recent years, a number of video quality metrics based on the spatio-temporal vision models have been proposed. One of the main motivations driving development of such metrics was the need to evaluate the performance of digital video coding and compression techniques in terms of artifacts visible to the human observer [8, 29]. In this study, we are interested in general purpose metrics which are applicable for synthetic image sequences. Such an ideal metric should account for important characteristics of the Human Visual System (HVS) such as the multi-resolution structure of the early stages of human vision, spatio-temporal sensitivity to contrast, and visual masking [9]. One commonly used approach is to extend a still image quality metric into the time domain [19, 29]. A practical problem here is the lack of separability of spatio-temporal Contrast Sensitivity Function (CSF) [16] (it is separable only at high spatial and temporal frequencies [28]). In practice, spatial and temporal channels are modeled separately by a filterbank, and the spatio-temporal interaction is then modeled at the level of respective gains of the filters [8, 29].
Another practical problem is computational cost and memory requirements involved in processing in the time domain. Usually two temporal channels are considered [8, 19] to

account for transient (low-pass) and sustained (band-pass with a peak frequency around 8 Hz) mechanisms [28].

Lack of comparative studies makes difficult evaluation of the actual performance of discussed metrics. It seems that the Sarnoff Just-Noticeable Difference (JND) Model [19] is the most developed (the Tektronix, Inc. product PQA-200 Picture Quality Analyzer test instrument includes so called JNDmetrix which is based on this technology), while a DCT-based model proposed by Watson [29] is computationally efficient and retains many basic characteristics of the Sarnoff model [6]. In this research, we decided to use our own metric of animated sequence quality, which takes advantage of the PF that is readily available in our application.

## 3 Rendering of the animation

Rendering of animation sequence is one of the key factors affecting time required for the overall animation production. For rendering techniques relying on keyframing, the overall animation rendering time depends heavily upon the efficiency of inbetween frames computation, which usually significantly outnumber the keyframes. In this section, we outline briefly our approach to the generation of inbetween frames. Then, we describe our algorithm for managing computation of the complete animation.

### 3.1 Inbetween frames generation

When selecting appropriate walkthrough rendering methods, their overall costs should be taken into account. On this ground, we reject some fast rendering techniques which require very costly preprocessing, e.g., the light field and other similar techniques [17, 13, 22]. Also, since in walkthroughs the spatial range of camera motions may be quite substantial and the viewing directions may change significantly, we think that in this case the cost-performance of the LDI technique [25] is not so attractive. The LDI data structures provide information for a rather limited space of possible observer locations and viewing directions. Effectively, the ratio between the number of derived images and the number of images used to building LDI might be low.

This reasoning focused our attention on simpler solutions, which use very simple data structures and do not require intensive preparatory computations. To account for proper PF computation and occlusion relations we use the 3D warping and warp ordering algorithms developed by McMillan [21], which require just the reference image and the corresponding range data. The formulation of McMillan's warping equation fits very well to camera model used in ray tracing, which simplifies the compositing of IBR-based and ray traced pixels. To reduce gaps between stretched samples during image reprojection we use the adaptive "splatting" technique proposed by Shade *el al.* [25]. To remove holes resulting from occluded objects we blend two keyframes as proposed by Mark *et al.* [20]. Pixels depicting objects occluded in the two keyframes are computed using ray tracing.

Since specular effects are usually of high contrast and they attract the viewer attention when observing a video sequence [24], the special care is taken to process them properly. We use our perception-based animation quality metric to decide for which objects with strong glossy or transparent properties pixels must be recomputed using ray tracing.

### 3.2 Managing inbetween frames generation

In our approach, rendering of walkthrough sequences is designed as a recursive procedure. In the initialization step, the whole walkthrough is decomposed into segments $S$ of uniform length (a reasonable length is selected, e.g., 25 subsequent frames). Then every segment $S$ is processed separately.

The recursive procedure for processing segment $S$ is as follows. The first frame $k_0$ and the last frame $k_{2N}$ are generated using ray tracing (keyframes are shared by two neighboring segments and are computed only once). Then 3D warping [21] is performed, and we generate two frames corresponding to $k_N$ as follows: $k'_N = Warp(k_0)$ and $k''_N = Warp(k_{2N})$. Using the perception-based animation quality metric (AQM) we compute the map of perceptible differences between $k'_N$ and $k''_N$. This quality metric incorporates the PF between frames $k_{N-1}$ and $k_N$, and $k_N$ and $k_{N+1}$ to account for temporal sensitivity of the human observer.

In an analysis process, at first we search for perceptible differences in images of objects with strong specular, transparent and glossy properties, which we identify using the item buffer of frame $k_N$ (in Section 6 we provide details on setting the thresholds of AQM response, which are used by us to discriminate between the perceptible and imperceptible differences). All pixels depicting objects for which the significant differences are reported in the perceptible differences map will be recalculated using ray tracing. We mask out those pixels from the map. In the same manner, we mask out holes composed of pixels which could not be derived from the reference images using 3D warping. If the masked-out difference map still shows significant discrepancies between $k'_N$ and $k''_N$ then we split the segment $S$ in the middle and we process recursively two resulting sub-segments using the procedure described in the previous paragraph. Otherwise, we blend $k'_N$ and $k''_N$ (with correct processing of depth [25]), and ray trace pixels for remaining holes and masked out specular objects to derive the final frame $k_N$. In the same way, we generate all remaining frames in $S$. To avoid the image quality degradation resulting from multiple resamplings, we always warp the fully ray-traced reference frames $k_0$ and $k_{2N}$ to derive all inbetween frames in $S$.

We evaluate the animation quality metric only for frame $k_N$. We assume that derivation of $k_N$ applying the IBR techniques is the most error-prone in the whole segment $S$ because its minimal distance along the animation path to either the $k_0$ or $k_{2N}$ frames is the longest one. This assumption is a trade off between the time spent for rendering and for the control of its quality (we discuss the costs of AQM in Section 6), but in practice, it holds well for typical animation paths.

Figure 1 (see Appendix/Color Section) summarizes computation and compositing of an inbetween frame. We used the dotted line to mark those processing stages that are performed only once for segment $S$. The remaining processing stages are repeated for all inbetween frames.

As the final step, we perform our spatio-temporal antialiasing. To speedup rendering phase all pixels (including those that have been ray traced) are not antialiased until this last stage of processing.

## 4 Image enhancement

Composing still images of high quality into an animated sequence might not result in equally high quality of animation because of possible temporal artifacts. On the other hand, proper temporal processing of the sequence makes possible relaxing the quality of frames without perceptible degradation of the animation quality, which effectively means that simpler and faster rendering methods can be applied.

It is well-known that aliasing affects the quality of images generated using rendering techniques. This concerns as well images obtained using IBR methods which additionally may exhibit various kind of discontinuities (such as holes resulting from the visibility problems). These discontinuities can be significantly reduced using techniques like splatting and image compositing introduced above, but anyway in the resulting images in many places instead of smooth transitions - jagged unwanted edges and contours will be easily perceptible (refer to the enclosed animation samples [1]).

Aliasing is also inherent to all raster images with significant content. Images obtained in computer graphics, or in general - all digital images, are the sampled versions of their synthetic or real world continuous counterparts. Sampling theory states that a signal can be properly reconstructed from its samples if the original signal is sampled at the Nyquist rate. Due to limited resolution of output devices such as printers and especially CRTs the Nyquist rate criterion in computer graphics is rarely met - and the image signal cannot be represented properly with a restricted number of samples.

From the point of view of signal processing theory - discontinuities and aliasing artifacts described above are high frequency distortions. This suggests the possibility of replacing the traditional, computationally-expensive antialiasing techniques - like unweighted and weighted area sampling and super-sampling, by an appropriate image processing method. Such an approach was tried by Shinya [26], who derived the sub-pixel information improving efficiency of antialiasing from the image sequences by tracking a given sample point location along the PF trajectories. In his approach, Shinya emphasized temporal filtering (his filter has ideal antialiasing properties when its size is infinite), which lead to filters of very wide support (Shinya acquired temporal samples from 32 subsequent frames of animation). In our research, we have found that by treating both aspects - spatial and temporal in a balanced way, we were able to improve both quality and efficiency of antialiasing. We have obtained a very efficient and simple antialiasing and image quality enhancement method based on low pass filtering using spatial convolution. Spatial convolution is a neighborhood operation - i.e. a result at each output pixel is calculated using the corresponding input pixel and its neighboring pixels. For the convolution, this result is the sum of products of pixel intensities and corresponding weights from the convolution mask. The values in the convolution mask determine the effect of convolution by defining the filter to be applied. Those values are derived from the point spread function of the particular filter - in the case of low pass filtering, typically it will be the Gaussian function (for more details on convolution see, e.g., [12]). In our case - i.e., in the case of a sequence of images composing an animation, we have to consider not only the spatial but also temporal aspect of aliasing and discontinuities. The proper way of solving the problem is to filter the three dimensional intensity function (composed of a sequence of frames) not along the time axis - but along the pixel flow - i.e. the PF introduced earlier it this paper (results and differences between those approaches can be seen on the enclosed animation samples [1]). Such filtering technique, known also as the motion compensated filtering, was earlier used in video signals processing [27], image interpolation [31], and image compression [14].

In practice, we used a separable Gaussian filter with the maximum support size of $5 \times 5$ in spatial and 9 temporal domains. The main idea that enabled us to use it in a context of antialiasing in a sequence of computer generated images (coming both from ray tracer and IBR) was to use the IBR technique to obtain the PF in a computationally inexpensive way.

The drawback of such a motion-compensated filtering (as well as the other solutions [26, 31, 27]) is incorrect processing of directional lighting effects, which are especially

objectionable for crisp mirror reflections. Indeed, motion of the reflected/refracted patterns over specular surfaces as a function of camera motion does not correspond to motion of these surfaces in the image plane which is described by the PF. Since estimation of the optical flow for reflections and refractions is quite involved, we used the following trade-off which worked well in walkthroughs that we tested. We reduced the size of temporal filter support for objects with strong directional reflectance/refraction properties.
The PF obtained from IBR gives us the sub-pixel accuracy (coordinates are calculated using floating point arithmetic). We have modified the standard spatial convolution algorithm to accommodate this feature. For each input pixel of the temporal filter the value is calculated as a weighted average of 4 pixels in the proximity of the input point (weights are proportional to the distance between neighboring pixel center and this point position).
It is well-known that low pass filtering as a side effect is causing blurring and in fact - a loss of information in the processed signal or image. In our case we have to consider that the content and the final quality of the resulting animation is to be judged by a human observer. We were quite fortunate to find that with pixels velocity increase there is an increase of perceived sharpness (see also [30]) - for example, an animation perceived as sharp and of a good quality can be composed of relatively highly blurred frames. I.e., each still frame considered separately would be judged as blurred and unacceptable by the human observer. As a result, the case of animation excessive blurring introduced by our antialiasing technique is compensated by the perceptual phenomena. This fact was also confirmed by our AQM predictor.

The antialiasing technique we developed in the scope of this research proved to be efficient and computationally inexpensive. Achieved quality can be evaluated on the enclosed animation samples [1], and the comparison of timings between traditional methods and our antialiasing approach can be found in Section 6.

## 5 Animated sequence quality metric

Before we move to the description of our metric of the animated sequence quality, we recall some well-known relationships between sensitivity to temporal fluctuations and moving patterns [28], which lie at the foundation of our approach.

### 5.1 Spatio-velocity vs. spatio-temporal considerations

Let $f(x, y, t)$ denote the space-time distribution of an intensity function (image) $f$, and $r_x$ and $r_y$ denote the horizontal and vertical components of the velocity vector $\vec{\mathbf{r}}$, which is defined in the $xy$ plane of $f$. For simplicity we assume that the whole image $f$ moves with constant velocity $\vec{\mathbf{r}}$, and the same reasoning can be applied separately to any finite region of $f$ that moves with a homogeneous, constant velocity [31]. The intensity distribution function $f_{\vec{\mathbf{r}}}$ of the image moving with speed $\vec{\mathbf{r}}$ can be expressed as:

$$f_{\vec{\mathbf{r}}}(x, y, t) = f(x - r_x t, y - r_y t, 0) \tag{1}$$

Let $F(u, v, \omega)$ denote the 3D Fourier transform of $f(x, y, t)$, where $u$ and $v$ are spatial frequencies and $\omega$ is temporal frequency. Then the Fourier transform $F_{\vec{\mathbf{r}}}$ of the image moving with speed $\vec{\mathbf{r}}$ can be expressed as:

$$F_{\vec{\mathbf{r}}}(u, v, \omega) = F(u, v)\delta(r_x u + r_y v - \omega) \tag{2}$$

This equation shows the relation between the spatial frequencies and the temporal frequencies, resulting from the movement of the image along the image plane. For example, we can see that a given flickering pattern characterized by the spatial frequencies $u$ and $v$, and the temporal fluctuation $\omega$ is equivalent to the steady pattern of the same spatial frequencies, but moving along the image plane with speed $\vec{\mathbf{r}}$ such that

$$r_x u + r_y v = \omega \tag{3}$$

This relationship between the velocity of an image pattern and its temporal frequency was used by Kelly [16] in his experimental derivation of spatio-velocity CSF. Kelly measured contrast sensitivity at several fixed velocities of traveling waves of various spatial frequencies. Kelly found that the constant velocity CSF curves have very regular shape at any velocity greater than about 0.1 degree/second. This made easy fitting an analytical approximation to the contrast sensitivity data derived by Kelly in the psychophysical experiment. Obviously, equation (3) can be used to convert the analytical representation of the spatio-velocity CSF into the spatio-temporal CSF, which is commonly used in many applications including video quality metrics.

Kelly performed his psychophysical experiments with stabilization of the retinal image to eliminate the eye movements. Effectively in this case, the retinal image velocity depended exclusively on the velocity of the image pattern motion. However, in the natural observation conditions the spatial acuity of visual system is affected also by the eye movements of three types: smooth pursuit, saccadic, and natural drift. Tracking of moving image patterns with smooth-pursuit eye movements makes possible compensating for the motion of an object of interest, which leads to reducing of the retinal velocity and improving acuity. The smooth pursuit movements make also possible to keep the retinal image of an object of interest in the foveal region, in which the ability of resolving spatial details is the best. The smooth-pursuit eye movement is affected by saccades, which shift the eye's focus of attention and may occur every 100-500 milliseconds [24]. The saccadic eye movements are of very high velocity (160-300 deg/sec), and effectively the eye sensitivity is near zero during this motion [5]. During intentional gaze fixation the drift eye movements are present, and their velocity can be estimated as 0.15 deg/sec [16, 5].

Daly [5] pointed out that a direct use of spatio-temporal CSF as developed by Kelly leads to underestimating of the human vision sensitivity because of ignoring the target tracking by the eye movements. Daly extended the Kelly's spatio-velocity CSF to account for the eye movements, and showed the way to transform it into the spatio-temporal CSF.

We found that in our application it is more convenient to include directly the spatio-velocity CSF to our animation quality metric. The following reasons may justify our approach:

- The widely used spatio-temporal CSF was in fact derived from the Kelly's spatio-velocity CSF, which was measured for moving stimuli (the traveling wave).
- As Daly have shown [5] accounting for the eye movements is more straightforward for a spatio-velocity CSF than for a spatio-temporal CSF.
- It is not clear whether the vision channels are better described as spatiotemporal or spatiovelocity [16, 6].
- The PF provides us directly with velocity information for local image regions.

The following section describes the animation quality metric developed in this research.

### 5.2 Animation Quality Metric

As the framework of our Animation Quality Metric (AQM) we decided to expand the perception-based visible differences predictor for static images proposed by Eriksson *el al.* [11]. The architecture of this predictor was validated by Eriksson *el al.* through psychophysical experiments, and its integrity was shown for various contrast and visual masking models [11]. Also, we found that responses of this predictor are very robust, and its architecture was suitable for incorporation of the spatio-velocity CSF.

Figure 2 illustrates the processing flow of AQM. A pair of compared animation frames undergoes an identical initial processing. At first, the original pixel intensities are compressed by the amplitude non-linearity and normalized to the luminance levels of the CRT display (the maximum luminance of 100 $cd/m^2$ was assumed). Then decomposition into spatial and orientation channels is performed using the Cortex transform proposed by Daly [4], and contrast in every channel is computed (the global contrast definition [11] in respect to the mean luminance value of the whole image was assumed). In the next stage, the spatio-velocity CSF was computed according to the Kelly model. Then the visual masking is modeled using the threshold elevation approach [11]. The final stage is error pooling across all channels.

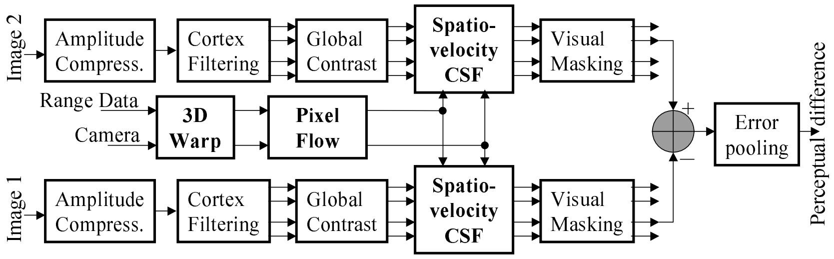



**Fig. 2.** Animation Quality Metric. The spatio-velocity CSF is based on velocity information for every pixel. For this purpose the Pixel Flow is computed for the next and previous frames along the animation path in respect to the input Image1 (or Image2 which should closely correspond to Image1). This requires the camera parameters for all three involved frames and the range data of Image1.

Since all stages of AQM are standard and well described in the provided references, we narrow further discussion to our extensions in respect to the predictor, which was originally proposed by Eriksson *el al.* We start by recalling the formula describing the Kelly spatio-velocity CSF model with its later extensions introduced by Daly [5]:

$$CSF(\rho, r) = c_0(6.1 + 7.3|\log(c_2 r/3)|^3)c_2 r(2\pi c_1 \rho)^2 \exp(-4\pi c_1 \rho(c_2 r + 2)/45.9) \tag{4}$$

where $\rho$ is spatial frequency in cycles per degree, $r$ is retinal velocity in degrees per second, and $c_0 = 1.14$, $c_1 = 0.67$, $c_2 = 1.7$ are coefficients introduced by Daly [5] to adapt the Kelly model to the typical levels of CRT display luminance (around 100 $cd/m^2$). The $CSF$ values are calculated for the center frequency $\rho$ of each Cortex frequency band. The resulting values are used to normalize contrasts in every frequency-orientation channel into the Just Noticeable Differences (JND) units [19, 4]. The retinal velocity is estimated as the difference of the image velocity $r_I$ and the eye movement velocity $r_E$ [5]:

$$r = r_I - r_E = r_I - \min(0.82 r_I + r_{Min}, r_{Max}) \tag{5}$$

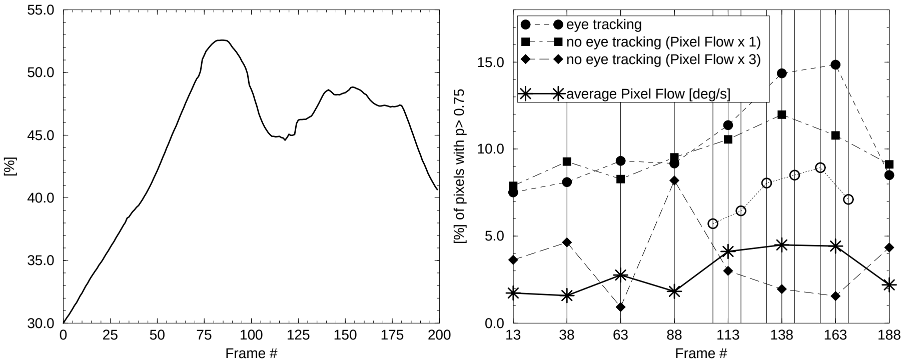
where $r_{Min} = 0.15$ is the estimated eye drift velocity, $r_{Max} = 80$ the maximum velocity of the smooth eye pursuit, and the coefficient 0.82 is experimentally derived efficiency of the eye tracking for a simple stimuli on the CRT display [5]. In general, the estimate of retinal velocity given by equation (5) is very conservative because it assumes that the eye is tracking all moving image elements at the same time. However, it cannot be considered as the upper bound of the eye sensitivity, because for the lower spatial frequencies the sensitivity may increase with the increasing retinal velocity [1], i.e., when the eye tracking efficiency is reduced. To account for this phenomena the eye movements can be ignored, in which case $r = r_I$. This assumption is actually made by many video quality metrics [8, 19, 29]. To get more conservative measure of the eye sensitivity these two estimates of the retinal velocity can be used and the maximum value of the sensitivity which depends on the image spatial contents can be selected.

The practical question arises how to estimate $r_I$. In our framework it becomes very easy because the PF derived using IBR techniques is available. For a given frame we use two estimates of the PF in respect to the previous and subsequent frames. We derive the retinal velocity vector for every pixel as the average of these estimates.
It is well-known that the image is maximally blurred in the direction of retinal motion, and the spatial acuity is retained in the direction orthogonal to the retinal motion direction [10]. To account for this characteristic of the visual system we project the retinal velocity vector to the direction of the filter band orientation.

# 6 Results

As the case study in this research we selected the walkthrough animation in an atrium of the University of Aizu [1]. The main motivation for this choice were interesting occlusion relationships between objects that are challenging for the IBR rendering. Also, a vast majority of surfaces in the atrium exhibits some view-dependent reflection properties including mirror-like and transparent surfaces, which made inbetween frames calculation more difficult. In such conditions, the AQM guided selection of keyframes and glossy objects within inbetween frames to be recomputed was more critical, and wrong decisions concerning these issues could be easy perceptible.
For our experiments we selected a walkthrough sequence of 200 frames. The resolution of each frame was $640 \times 480$ (to accommodate for the NTSC standard). At the initialization step, we split this walkthrough into eight segments $S$ of 25 frames each.

As described in Section 3.2 for every segment $S$ we run the AQM once to decide upon the specular objects which require recomputation. The AQM is calibrated in such way that 1 JND unit corresponds to a 75% probability that an observer can perceive the difference between the corresponding image regions (such a probability value is the standard threshold value for discrimination tasks [4]). If a group of connected pixels representing an object (or a part of an object) exhibits the differences bigger than 2 JND (93.75% probability of discrimination) we select such an object for recalculation. If for an object the differences below 2 JND are reported by the AQM then we estimate the ratio of pixels exhibiting such differences to all pixels depicting this object. We assume that if the ratio is bigger than 25% then we select such an object for recomputation - 25% is an experimentally selected trade-off value, which makes possible the reduction of the number of specular object requiring recomputation, at expense of some potentially perceptible image artifacts. These artifacts are usually hard to notice unless the observer attention is specifically directed to a given image region. The graph in Figure 3a depicts the percentage of pixels that are selected for recomputation in our walkthrough sequence. The percentage includes also pixels which cannot be properly derived using the IBR techniques, which usually are a small fraction of all recomputed pixels (in average 0.3% for our animation).




a) b)

**Fig. 3.** a) The percentage of pixels to be recalculated by ray tracing. b) The AQM prediction of the perceived differences between warped images of two neighboring reference frames taking into account various retinal image velocity. Also, the average Pixel Flow velocity expressed in [degree/second] units is shown. Lines connecting the symbols were added for the figure readability and they do not have any meaning for unmarked frames.

After masking out the pixels to be recomputed, the decision upon further splitting of $S$ is taken using the AQM predictions for the remaining pixels. The predictions are expressed as the percentage of unmasked pixels for which the probability $p$ of detecting the differences is greater than 0.75. Based on experiments that we conducted, we decided to split every segment $S$ when the percentage of such pixels is bigger than $10\%$. When computing the AQM predictions that we used to decide upon segment splitting, we assumed good tracking of moving image patterns with the smooth-pursuit eye movements (the retinal velocity is computed using equation (5)). The filled circles in Figure 3b show such predictions for the inbetween frames located in the middle of every initial segment $S$. Three segments with the AQM predictions over $10\%$ were split and the empty circles show the corresponding reduction of predicted perceptible differences. We performed also experiments assuming higher levels of the retinal velocity when observing our walkthrough animation. The filled squares in Figure 3b show the AQM predictions when the retinal velocity is equal to the PF (the eye movements are ignored). For all segments that we selected for splitting based on the smooth-pursuit eye movements assumption, the AQM predictions exceeded the threshold of $10\%$ as well when the eye movements were ignored. As we discussed in Section 5.2, although in general the eye sensitivity is improving when the eye tracking is enabled, however, for some image patterns the eye sensitivity can be better when the eye tracking is disabled (refer to the AQM predictions for the inbetween frame #38). The filled diamonds marks show the AQM prediction assuming that the original velocity of PF was multiplied by the factor three (the eye movements are ignored). Effectively, this corresponds to three times faster display of our animation. As expected, in general the perceivability of image artifacts decreases with the velocity. The graph shown with the thick line shows the average PF values in [degrees/second] units, which were measured for the selected inbetween frames.

To evaluate efficiency of our animation rendering system we compared the average time required for a single frame of the atrium walkthrough. All timings were measured

on the MIPS 195 MHz processor. For an antialiased frame (with adaptive supersampling) the required rendering time was about 170 minutes, and for the corresponding non-antialiased frame which was used as a keyframe for our image-based rendering took about 40 minutes. The average rendering time using our compositing of IBR and ray traced images was about 27 minutes (in average 43.9% of pixels were ray traced per frame). This included rendering keyframes, the AQM processing (which required 9 minutes to process a pair of frames, mostly because the Fast Fourier Transform of $1024 \times 512$ images was involved as the result of processing our $640 \times 480$ frames), IBR rendering (which requires about 12 seconds to warp and blend two reference frames). The motion-compensated 3D filtering added an overhead of 10 seconds per frame.
The most significant speedup was achieved by using our spatio-temporal antialiasing technique and avoiding the traditional adaptive supersampling. Our inbetween frames rendering technique added further 30% of speedup for the scene built mostly from surfaces exhibiting view-dependent reflectance properties. Even better performance can be expected for environments in which specular objects are depicted by a moderate percentage of pixels.

## 7 Conclusions

In this work, we proposed an efficient approach to rendering animated walkthrough sequences of high quality. Our contribution is in developing a fully automatic, perception-based guidance of the inbetween frames computation, which minimizes the number of pixels computed using costly ray tracing, and seamlessly (in terms of perception of animated sequences) replace them by pixels derived using inexpensive IBR techniques. Also, we have shown two useful applications of the Pixel Flow obtained as a by-product of IBR processing: (1) to estimate the spatio-velocity Contrast Sensitivity Function which made possible incorporation of temporal factors into our perceptually-informed image quality metric, (2) to perform the spatio-temporal antialiasing with motion-compensated filtering based on image processing principles (in contrast to traditional antialiasing techniques used in computer graphics). We integrated all these techniques into a balanced animation rendering system.

As the future work we plan to conduct validation of our AQM in psychophysical experiments. Also, we believe that our approach has some potential in automatic selection of reference frames used in IBR systems. As the future work we plan to investigate this issue.


## Acknowledgments

Special thanks to Scott Daly for his stimulating comments on video quality metrics.


## References


1. `www.u-aizu.ac.jp/labs/csel/aqm`. *The Web page accompanying to this paper*.
2. S.J. Adelson and L.F. Hodges. Generating exact ray-traced animation frames by reprojection. *IEEE Computer Graphics & Applications*, 15(3):43–52, 1995.
3. C. Chevrier. A view interpolation technique taking into account diffuse and specular inter-reflections. *The Visual Computer*, 13(7):330–341, 1997.
4. S. Daly. The Visible Differences Predictor: An algorithm for the assessment of image fidelity. In A.B. Watson, editor, *Digital Image and Human Vision*, pages 179–206. MIT Press, 1993.

5. S. Daly. Engineering observations from spatiovelocity and spatiotemporal visual models. In *Human Vision and Electronic Imaging III*, pages 180–191. SPIE Vol. 3299, 1998.
6. S. Daly. personal communication. 1999.
7. L. Darsa, B.C. Silva, and A. Varshney. Navigating static environments using image-space simplification and morphing. In *1997 Symposium on Interactive 3D Graphics*, pages 25–34. ACM SIGGRAPH, 1997.
8. C.J.van den Branden Lambrecht. *Perceptual models and architectures for video coding applications*. Ph.D. thesis, 1996.
9. C.J.van den Branden Lambrecht and O. Verscheure. Perceptual quality measure using a spatio-temporal model of the human visual system. pages 450–461. SPIE Vol. 2668, 1996.
10. M.P. Eckert and Buchsbaum G. The significance of eye movements and image acceleration for coding television image sequences. In A.B. Watson, editor, *Digital Image and Human Vision*, pages 89–98. Cambridge, MA: MIT Press, 1993.
11. R. Eriksson, B. Andren, and K. Brunnstrom. Modelling of perception of digital images: a performance study. pages 88–97. Proceedings of SPIE Vol. 3299.
12. R.C. Gonzalez and R.E. Woods. *Digital image processing*. Addison-Wesley, 1993.
13. S.J. Gortler, R. Grzeszczuk, R. Szeliski, and M.F. Cohen. The lumigraph. In *SIGGRAPH 96 Conference Proceedings*, Annual Conference Series, pages 43–54, 1996.
14. B.K. Guenter, H.C. Yun, and R.M. Mersereau. Motion compensated compression of computer animation frames. In *SIGGRAPH '93 Proceedings*, volume 27, pages 297–304, 1993.
15. Michael Halle. Multiple viewpoint rendering. In *SIGGRAPH 98 Conference Proceedings*, Annual Conference Series, pages 243–254, 1998.
16. D.H. Kelly. Motion and Vision 2. Stabilized spatio-temporal threshold surface. *Journal of the Optical Society of America*, 69(10):1340–1349, 1979.
17. M. Levoy and P. Hanrahan. Light field rendering. In *SIGGRAPH 96 Conference Proceedings*, Annual Conference Series, pages 31–42, 1996.
18. D. Lischinski and A. Rappoport. Image-based rendering for non-diffuse synthetic scenes. In *Proceedings of Eurographics Rendering Workshop '98*, pages 301–314, 1998.
19. J. Lubin. A human vision model for objective picture quality measurements. In *Conference Publication No. 447*, pages 498–503. IEE International Broadcasting Convention, 1997.
20. W.R. Mark, L. McMillan, and G. Bishop. Post-rendering 3D warping. In *1997 Symposium on Interactive 3D Graphics*, pages 7–16. ACM SIGGRAPH, 1997.
21. L. McMillan. *An Image-Based Approach to 3D Computer Graphics*. Ph.D. thesis, 1997.
22. G. Miller, S. Rubin, and D. Poncelen. Lazy decompression of surface light fields for precomputed global illumination. In *Rendering Techniques '98 (Proceedings of Eurographics Rendering Workshop '98)*, pages 281–292, 1998.
23. J. Nimeroff, J. Dorsey, and H. Rushmeier. Implementation and analysis of an image-based global illumination framework for animated environments. *IEEE Transactions on Visualization and Computer Graphics*, 2(4):283–298, 1996.
24. W. Osberger, A.J. Maeder, and N. Bergmann. A perceptually based quantization technique for MPEG encoding. pages 148–159. Proceedings of SPIE Vol. 3299, 1998.
25. J.W. Shade, S.J. Gortler, L. He, and R Szeliski. Layered depth images. In *SIGGRAPH 98 Conference Proceedings*, pages 231–242, 1998.
26. M. Shinya. Spatial anti-aliasing for animation sequences with spatio-temporal filtering. In *Computer Graphics (SIGGRAPH '93 Proceedings)*, volume 27, pages 289–296, 1993.
27. A. Murat Tekalp. *Digital video Processing*. Prentice Hall, 1995.
28. A.B. Watson. Temporal sensitivity. In *Handbook of Perception and Human Performance, Chapter 6*. John Wiley, New York, 1986.
29. A.B. Watson. Toward a perceptual video quality metric. In *Human Vision and Electronic Imaging III*, pages 139–147. Proceedings of SPIE Vol. 3299, 1998.
30. J.H.D.M. Westerink and C. Teunissen. Perceived sharpness in moving images. pages 78–87. Proceedings of SPIE Vol. 1249, 1990.
31. E. Zeghers, S. Carre, and K. Bouatouch. Faster image rendering in animation through motion compensated interpolation. In *Graphics, Design and Visualization*, pages 49–62, 1993.