EUROGRAPHICS 2012 / P. Cignoni, T. Ertl
(Guest Editors)

Volume 31 (2012), Number 2

NoRM: No-Reference Image Quality Metric for Realistic Image Synthesis

Robert Herzog[1] and Martin Čadík[1] and Tunç O. Aydın[1,2] and Kwang In Kim[1] and Karol Myszkowski[1] and Hans-P. Seidel[1]

[1] MPI Informatik Saarbrücken, Germany, [2] Disney Research Zurich, Switzerland, http://www.mpi-inf.mpg.de/resources/hdr/norm/

Abstract

Synthetically generating images and video frames of complex 3D scenes using some photo-realistic rendering software is often prone to artifacts and requires expert knowledge to tune the parameters. The manual work required for detecting and preventing artifacts can be automated through objective quality evaluation of synthetic images. Most practical objective quality assessment methods of natural images rely on a ground-truth reference, which is often not available in rendering applications. While general purpose no-reference image quality assessment is a difficult problem, we show in a subjective study that the performance of a dedicated no-reference metric as presented in this paper can match the state-of-the-art metrics that do require a reference. This level of predictive power is achieved exploiting information about the underlying synthetic scene (e.g., 3D surfaces, textures) instead of merely considering color, and training our learning framework with typical rendering artifacts. We show that our method successfully detects various non-trivial types of artifacts such as noise and clamping bias due to insufficient virtual point light sources, and shadow map discretization artifacts. We also briefly discuss an inpainting method for automatic correction of detected artifacts.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Image Quality Assessment

1. Introduction

While photo-realistic rendering methods are getting more advanced over time, various rendering artifacts still appear as a problem in the results. These artifacts can be reduced or completely avoided by fine-tuning the rendering algorithm's parameters through trial and error. But this manual process is often time-consuming and requires some level of understanding about the inner machinery of the rendering method in consideration. Analogous to the field of objective image quality assessment where one can use computational *quality metrics* that predict the subjective quality evaluation, the objective quality assessment of synthetic images is highly beneficial because it eliminates the tedious manual labor required otherwise. Additionally, such a metric enables automatic detection and elimination of rendered images of unacceptable quality. To that end we propose an objective *image quality metric for realistic image synthesis* based on a machine learning system trained with various types of rendering artifacts.

Building a quality metric for synthetic images has additional challenges over a metric for natural images. The metrics for natural images are often *full-reference*, namely they rely on a non-distorted copy of the image for evaluating the distorted (test) image. Unlike in applications like compression and watermarking, in rendering such a reference image is often not available in practice and a metric for synthetic images should detect and predict the strength of rendering artifacts based solely on the test image. Although humans often detect distortions just as well without a reference, in contrast non-reference image quality metrics are usually inferior in performance to full-reference metrics [WR05]. Thus, the absence of a reference image is a significant constraint in metric design.

The central idea of this paper is to leverage 3D scene information to compensate for the lack of a reference image while detecting rendering artifacts. Any scene specific per pixel data data beyond color, such as depth, texture and material, is difficult, if at all possible, to obtain reliably in natu-

ral images. This is not the case for rendered scenes, and we show that taking full advantage of this additional information enables non-reference quality assessment of synthetic images with a prediction performance comparable to full-reference metrics.

- a fully automatic metric that detects rendering artifacts without a reference,
- a learning framework for common rendering artifacts that also guides our artifact removal,
- a human visual system-inspired model that predicts the perceived strength of rendering artifacts,
- a dataset of photo-realistic rendering artifacts including subjective artifact probability detection maps.

In a subjective study, we show that the performance of our metric matches the state-of-the-art in full-reference metrics. Our metric could be employed in rendering farms, as well as in controlling the rendering quality in client-server or cloud computing settings. One could also use it as a diagnostic tool for rendering quality, or in an optimization framework to find optimal parameters for a rendering method.

2. Related Work

In this section we review previous work on *non-reference* (NR) image/video quality assessment. First, we discuss the NR metrics for imaging applications, and then, we present rendering-specific solutions. For a detailed discussion of the *full-reference* (FR) and *reduced reference* (RR) quality metrics we refer the reader to the recent textbooks [Win05, WB06, WR05]. FR metrics tailored for computer graphics and HDR imaging applications are summarized in [RWD*10, Ch. 10] and [MKRH11].

NR metrics in imaging applications The key difficulty in developing NR metrics is the absence of a non-distorted reference image or some features representing it. Common approaches to compensate for this are (1) modeling distortion-specific characteristics, (2) using natural scene statistics, and (3) employing learning based classification methods.

Distortion-specific NR methods capitalize on the knowledge of artifact type and its unique characteristics [WR05, Ch. 3]. Examples include metrics for detecting blockiness due to lossy JPEG and MPEG compression and ringing at strong contrast edges [WB06], blurriness due to high frequency coefficients suppression [CCB11, LH11], banding (false contouring) at low gradient regions due to the excessive quantization [DF04]. There are some attempts of building more general NR quality metrics, which evaluate a combined contribution of individually estimated image features such as sharpness, contrast, noise, clipping, ringing, and blocking artifacts [WR05, Ch. 10]. The contribution of all features including their simple interactions is summed up with weights derived through fitting to subjective data.

Natural scene statistics [Sim05] derived from artifact-free images can be helpful in detecting artifacts. Sheikh et al. show that noise, blurriness, and quantization can be identified as deviations from these statistics [SBC05].

Image features extracted from distorted and non-distorted images are used for training machine learning techniques such as support vector machines (SVM) or neural networks. Moorthy and Bovik [MB10] use generalized Gaussian distribution (GGD) to parametrize wavelet subband coefficients and create 18-D feature vector (3 scales $\times$ 3 orientations $\times$ 2 GGD parameters), which is used to train an SVM classifier based on perceptually calibrated distortion examples from the LIVE IQA database. The classifier discriminates between five types of mostly compression-related distortions and estimates their magnitude. Saad et al. [SBC10] train a statistical model to detect distortions in DCT-based contrast and structure features.

Discussion: Our technique differs from previous work in three ways: (1) we use depth buffer and albedo information in addition to color, (2) our output is a distortion *map* rather than a scalar value, and thus, we show spatial distribution of distortions, and (3) our work specializes in rendering artifacts rather than compression/transmission related artifacts.

Rendering-specific NR metrics Some metrics in this category rely on *predicted* reference images. In image-based rendering the mis-registration error of pixels with respect to the ground truth reference image is a good measure of visual quality [KSGH09]. In 3DTV applications the lack of ground truth can be compensated by reprojecting (warping) images from different cameras to the mid-point view [KSGH09]. Also, when temporal frame replication is performed for reducing the rendering cost or display hold-type blur, similar reprojection in temporal domain is feasible [MRT99]. In contrast to these, our method is purely NR in that we do not need to predict a reference. This is also the case for the recent work of Berger et al. [BLL*10] where specialized ghosting detector explicitly works on an interpolated image.

Other work in computer graphics literature includes a model of the elevation of contrast discrimination threshold due to visual masking, which can be predicted based on the texture pattern only [RPG99, WPG02]. An estimator of bias, which mostly leads to blurred shading details, has been proposed within the progressive photon mapping framework [HJJ10]. This estimator relies strongly on intrinsic renderer information such as derivatives of estimated lighting function, which becomes feasible only for density estimation methods with smooth kernel functions. Our data-driven approach aims for using less rendering specific and easier to acquire data. Stokes et al. [SFWG04] introduced a perceptual NR metric, which predicts the contribution of the indirect illumination components towards perceived image quality. While the metric cannot detect local artifacts, similar to our metric it considers per pixel reflectance information.

Ramanarayanan et al. [RFWB07] proposed metrics that

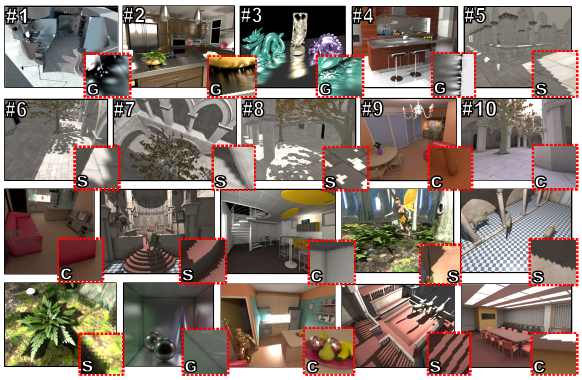

Figure 1: *Example images with artifacts used for our no-reference quality metric. Insets show magnified artifacts regions, letters indicate the type of artifact (C: VPL clampling, G: glossy VPL noise, S: shadow map aliasing). The numbered images correspond to the testset used in the user study.*

utilize per object reflectance and surface bumpiness information for training SVM classifier on subjective data. Their method measures the overall *visual equivalence* instead of identifying problematic image regions. Křivánek et al. [KFB10] investigated visual equivalence for instant radiosity (virtual point light) algorithms and proposed a number of useful rendering heuristics, which were difficult to formalize into a ready to use computational model.

3. Overview

In this work, we are interested in automatically detecting rendering artifacts, which are typical for global illumination solutions (Fig. 1), and that we briefly describe in Section 4. We achieve this via a machine learning approach (see Section 5) based on the discrimination of rendering-specific features (Section 5.2) trained on our generated database of synthetic image pairs (Section 5.1). The whole system is depicted in Fig. 2. Note that we do not intend to classify an image as a whole but rather predict the locations of artifacts in an image. Optionally, we can "clean" the image making effective use of inpainting techniques (Section 7) based on the same set of training image pairs and obtain a "pseudo-reference" image, which is then used to perceptually normalize the distortion map for the visibility of artifacts (Section 8). We present our results in Section 9 and demonstrate in a user study (Section 10) that our method is competitive with state-of-the-art reference methods (VDPs). Finally, we conclude with future prospects in Section 11.

4. Rendering-specific Artifacts

Photo-realistic rendering is still very time-consuming and rendering a high-resolution, globally-illuminated image may take several minutes to hours becoming even more critical in the case of an animation. Therefore, many rendering algorithms trade quality (bias) for speed and often leave it to the user to find the right parameters, eventually resulting in algorithm-specific artifacts, which are hard to control, i.e., the generated image might look fine partially but exhibits strong degradations in small areas.

In our experiments we focused on artifacts inherent to popular rendering algorithms, which comprise *Instant Radiosity* [Kel97] with glossy virtual point lights (VPLs), *Lightcuts* [WFA*05], and OpenGL rasterization using *PCF shadow maps* [RSC87], which produce VPL-based artifacts (i.e., low-frequency noise), clamping bias (darkened corners), and shadow map aliasing (jaggy shadow boundaries), respectively. Examples of images showing these artifacts are given in Fig. 1. We exclude stochastic noise (pixel-variance), e.g., anti-aliasing, path-tracing, from our study, which is much easier to handle and well-studied in the rendering community [BM98, RPG99, TJ97, KA91]. We also do not discuss temporal artifacts, which are beyond the scope of this work [YPG01].

5. Learning Rendering Errors from Examples

Computing the perceived image errors along with the final pixel colors during the rendering process can be very helpful for example to adapt the rendering. However, this is only feasible for easily analyzable errors in very specific algorithms, which often boils down to storing and evaluating lower order statistics (e.g., variance in path-tracing). In general, estimating the visual error without a reference is a difficult and ill-posed problem, which may easily become more demanding than the rendering process itself. Another issue is that we may not always have access to the renderer's source code or that we simply have not enough understanding of the underlying problem and in particular how to quantify the visible error. This could be because the rendering problem is hard to analyze or there are many hidden factors that have a large impact on the final, perceived rendering quality, like for example the shape of the geometry, local or global lighting distribution, scene material, rendering parameters, or even visual masking effects. All these thoughts have led to our data-driven non-reference quality metric (NoRM).

5.1. Image Data Collection

The problem of understanding and classifying rendering artifacts in general is too complex to be tackled analytically and we have chosen a data-driven approach that relies on machine learning. Since the space of artifacts even for one specific type is high-dimensional, we need many images with "right" and "wrong" examples to train a classifier initially. In general, while generating "clean" reference images may be time-consuming, producing various kind of artifacts in the rendered images is often trivial. Hence, we generated a database of rendered images with positive and negative example-images for each type of artifact (see some examples in Fig. 1). In contrast to image datasets used in computer vision tasks, our database comprises:

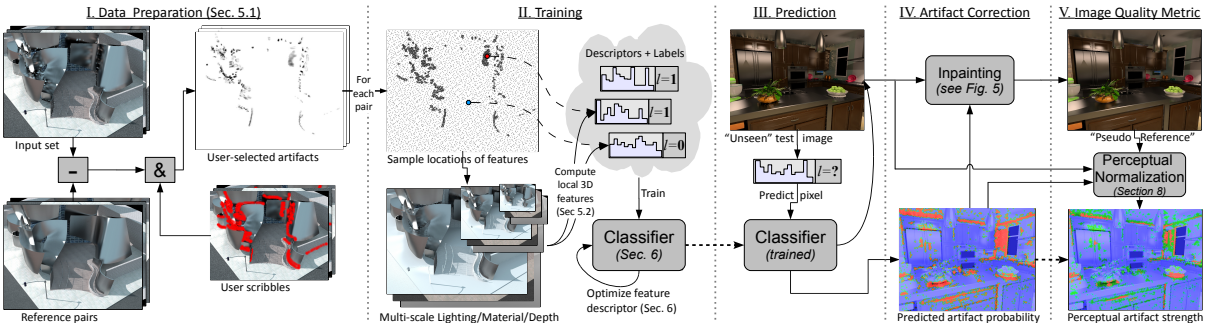

Figure 2: *Overview of the whole NoRM pipeline. Labels are semi-automatically extracted by differencing and thresholding the image with its reference and then masking this residual with a coarse user mask. For training the classifier, these labels are uniformly sampled with equal number of positive/negative samples at which we compute our multi-scale 3D features. The resulting high-dimensional descriptors are fed to the classifier (SVM). After optimizing parameters and feature dimension reduction via cross-validation, the classifier can predict artifacts in a new image. For artifact-prone pixels we inpaint reference patches from the same training image pairs to generate a pseudo-reference that is finally used in our perceptual normalization.*

- color image with reference,
- depth buffer,
- diffuse material buffer (textures),

which we refer to as a *frame* (see Fig. 3 for an example).

The reason why we restrict ourselves to this data – although we could in principle extract more – is that this data is relatively easy to dump and requires only little modification, if at all, of the rendering software. Specifically, these buffers are commonly stored in a *deferred renderer*. † Given a frame we generate other useful data, which we need for computing feature descriptors: screen-space normals from linearized depth, and approximate lighting (irradiance) using the color and material buffer.

In order to focus on artifacts which are above the threshold of visibility (and also on one specific type of artifacts) during the learning stage, initially, a coarse mask is manually painted over the tone mapped image. In the masked regions we compute the error between the pixels in the reference and the artifact-image via differencing. This way the user only needs to provide a rough mask in which we label those pixels for which the error really exists, see Fig. 2. Finally, we avoid that a few pixels are *not* assigned artifact labels (e.g., due to zero-crossings in the error signal) although neighboring pixels would indicate so. Therefore, we perform an additional dilation plus erosion (morphological closing) on the labels with a disc of pixel radius 2.

5.2. Features for Classification

Finding good descriptors or a combination of descriptors that discriminate the feature space well is crucial for any machine learning approach. We experimented with various standard techniques to classify and discriminate artifacts from the remaining "correct" pixels. Those feature descriptors comprised local histograms of color and depth, HoG (histogram of oriented gradients), multi-scale Hessian, and frequency domain descriptors based on discrete cosine transform (DCT). We applied these descriptors to compute features for our depth, color and material buffer pixels. It turned out that none of those techniques was discriminative enough to give satisfying results and we had to dig more into the rendering process itself exploiting scene and rendering-specific knowledge, which we will describe further.

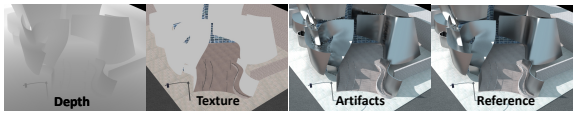

Figure 3: *Example of the data used as input for training.*

5.2.1. Texture Removal

Instead of computing features in the material buffer and increasing the dimension of the feature space, we partially remove the correlation of pixel color and texture to obtain the approximate lighting (compare the left image in Fig. 4 with Fig. 3). We only restrict ourselves to diffuse textures since these usually convey the most information about material structure in a synthetic scene. For diffuse surfaces in a scene this provides us with information about irradiance instead of pixel radiance, which is locally low-dimensional. Since the color buffer is given as HDR image, we simply divide the color pixels by the corresponding linearized material pixels. Care has to be taken with material values clamped to zero where the original lighting information in one or more color channels is essentially lost. For such rare cases we diffuse the lighting in the clamped color channels from spatially

† The depth buffer is always present in a rasterizer and the material buffer could be obtained by rendering the scene shot with only ambient lighting or using a simple "*eye-light* shader".

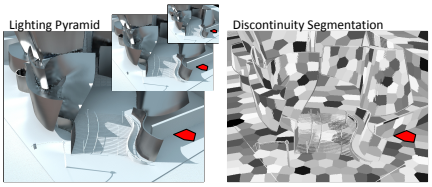

Figure 4: *Computing statistics of lighting in a local, contiguous neighborhood (red patch) at different scales.*

neighboring pixels, that are not affected by clamping. Entirely black material pixels are considered as light sources or specular surfaces and the corresponding color pixels are not altered. These heuristics worked well for our image database but are certainly not always satisfactory when dealing with complex glossy materials possibly consisting of several layered textures. For such a scenario the user can still provide the lighting image instead of relying on an ill-posed deconvolution of BRDF and lighting.

5.2.2. Screen-space ambient occlusion

Screen-space ambient occlusion (SSAO) has been developed for the GPU to efficiently compute an approximate scalar ambient-occlusion term $s_{ao}(x)$ solely based on the depth buffer. Essentially, ambient-occlusion computes the solid-angle covered by the non-occluded environment (far field) in the visible hemisphere of directions. Although SSAO is a crude approximation in screen-space, it can deliver good results for pixels where the surrounding occluders are all visible in the depth buffer. Ambient occlusion is highly correlated with the harmonic mean distance to the surrounding surfaces, which is often taken as an upper bound for the irradiance gradient of the indirect lighting [WRC88]. Since many lighting artifacts are due to large indirect gradients, the complement, $1 - s_{ao}(x)$, is also a good indicator for potential artifacts.

5.2.3. Rectified Tiles – Descriptors in Texture-space

In contrast to computer vision approaches, the presence of exact depth per pixel allows us to "unfold" a local image region from the surface captured by the depth buffer and transform it to its canonical view. This way, we are able to preserve depth discontinuities and perspective when computing local feature descriptors (e.g., histograms) and essentially reduce the dimensionality of the classification problem since we can operate in 2D texture space[‡]. For computing the local texture parametrization of the decals we use *discrete exponential maps* [SGW06] computed over the depth buffer, which is based on Dijkstra's graph-distance algorithm. An example of the computed decal parametrization is shown in Fig. 5. The so computed texture parametrization gives us the mapping from 2D texture space to projected 2D image space but we need the inverse mapping. Instead of directly "unwrapping" the surface colors via a splatting approach, we first compute the inverse texture mapping (the displacement field) via *splatting* and then use this (smooth) vector field for *gathering* the surface colors [Sze10]. Since splatting may lead to holes for overly stretched pixels, we fill the "deformation vector field" using a push-pull approach [GGSC96]. This two-stage approach better preserves high-frequencies in the colors and introduces only a small amount of blur due to (bilinear) resampling when gathering the color via the inverse texture mapping. To this end, we use the computed parametrization for computing histograms of oriented gradients (HoG) directly in texture space but also for the inpainting described in Section 7.

[‡] For 30.000 randomly extracted 16×16 pixel blocks from the glossy VPL images we run PCA on rectified, non-rectified blocks and captured 99.5% of the variance in 12 (10 for shadow map aliasing), 16 basis vectors out of 256, respectively.

5.2.4. Joint-Bilateral Filtering

To detect high-frequency artifacts in the image we perform frequency analysis. To eliminate the influence of edges and discontinuities in the depth buffer we blur the image with a joint-bilateral filter with weights steered by the depth and surface normal differences of the pixels under the filter footprint. The Gaussian variance of the depth and normal filter is automatically estimated from the 80-th and 91-th percentile of the depth and normal angle histogram, respectively. Next, we compute the residual as the filtered lighting subtracted from the original lighting image. For each feature sample we perform a local discrete cosine transform (8×8 DCT) in the residual image within a weighted Gaussian window (Gabor filter) at 2 different scales in a pyramid.

5.2.5. Local Statistics

Artifact image regions have different color distributions than the reference counterpart and we compute the first four central moments (mean, variance, skewness, kurtosis) in a local window of 16×16 pixels in 3 different scales. Similar to the joint-bilateral filtering, we only compute the statistics in a window over pixels that correspond to a contiguous surface in the depth buffer. In order to do so, we segment the depth buffer in piecewise continuous image segments via k-means clustering of pixels with respect to depth and surface normal (see Fig. 4).

6. Classification and Feature Optimization

We have proposed and tested several standard as well as specialized features described above. However, many of those features are not useful for our task or might be redundant. Certainly, using too many features, it is likely that the model overfits the training data and that we cannot provide enough examples to train the classifier efficiently. Hence, we have to select a subset from our feature pool such that the combined

feature is the most discriminative with respect to our artifact type.

Before the optimization, we linearly rescale the extracted feature vectors such that the 5th percentile of all data points maps to 0 and the 95th percentile to 1. This way, we estimate the feature bounds only from the "inlier" training data, i.e., we only account for samples within a standard deviation $\sigma \approx 1.5$ if we assume data samples are Gaussian distributed. Then, similar to [LSAR10], we use a greedy approach to select the "best" feature subset. The idea is to select the feature, one at a time, that minimizes the cross-validation error measure (*BER* see below) computed over the training set and add this feature to the current best feature set, which is initially empty. This procedure is continued till the cross-validation error of the classifier is increased when adding more features (i.e., increasing the feature dimension). The resulting features after this optimization starting with the entire feature pool are listed in Tab. 1. We use 10-fold cross-validation over the feature descriptors (computed from subimages) and split the randomly permuted training features into 90% training and 10% validation data and perform 5 iterations (i.e., evaluate the feature performance on half of the data set).

For the classifier we use a support vector machine (SVM) with a radial basis function (RBF) kernel. The two main parameters of the SVM, the regularization parameter C, and the RBF kernel width γ, are automatically computed by again minimizing the cross-validation error in a hierarchical manner (coarse-to-fine grid search). The best parameters for each type of artifact can be found in Table 1.

When testing the classifier on new (unseen) images, which may contain much fewer artifacts than non-artifact pixels, the classification may result in high recognition rates (>90%) even when every pixel is classified as "non-artifact". To get a more sensible error measure, we chose the balanced error rate (BER): $BER = \frac{1}{2}\left(\frac{|\{i \mid l(i)\in\Omega_+ \wedge p(i)\neq l(i)\}|}{|\Omega_+|} + \frac{|\{i \mid l(i)\in\Omega_- \wedge p(i)\neq l(i)\}|}{|\Omega_-|}\right)$, where $l(i)$ is the correct label, $p(i)$ is the predicted label of sample i and Ω_+, Ω_- is the set of positive, and negative labeled samples.

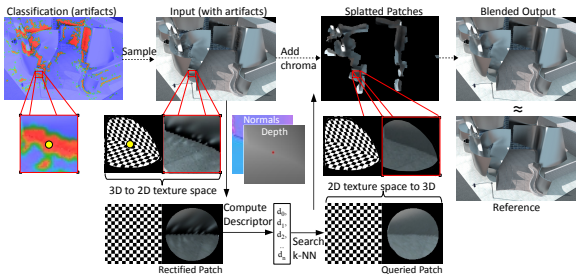

Figure 5: *Outline of the proposed inpainting algorithm (without the material decorrelation) illustrated for one patch including texture parametrization computed as described in Section 5.2.3.*

7. Artifact Correction via Inpainting

Certainly, detecting artifacts is appealing, but we would also like to remove the artifacts to obtain a higher quality, acceptable image, which we can utilize as a "pseudo-reference", see Section 8. In cases where the artifacts are minor and cover only a small fraction of the image, this is possible. We already described how to compute the likelihood of pixels to be prone to artifacts of certain types using SVM classification. An obvious approach to artifact elimination is to perform regression and learn the error function of the artifact training images. However, our tests using support vector regression were unsatisfactory, perhaps because the error function is often too noisy. Hence, we chose an approach based on context-sensitive inpainting.

First of all, during the correction phase we only touch those pixels that are classified as artifacts with a certain minimum strength. The main idea is to inpaint tiny images seamlessly into the detected artifact regions that match the local configuration of this region. Again, we exploit the additional information in the depth and material buffers to facilitate the inpainting process. First, we only inpaint rectified images that live in texture space, which we glue onto the contiguous surface as described in Section 5.2.3. Second, we remove the textures from the image before the inpainting process (see Section 5.2.1). Nevertheless, the inpainting procedure must still be able to preserve high-frequency edges (e.g., caustics, shadows) and must also hide the transition at the inpainting boundary. The later is achieved via linear blending of the splatting result with the original image, where the blending weights are computed from the binary artifact labels (red pixels in Fig. 5), which we blur with a Gaussian ($\sigma = 3$) after dilating them by a quarter of the patch size (i.e., 4 pixels). We also experimented with Poisson image blending [PGB03] but it produced sometimes unrealistically looking color bleedings.

Now, we need to find artifact-free image blocks to be painted into the local artifact region. Our inpainting operates in LDR *YCbCr* color space and we only inpaint tone mapped luminance (Y) while chroma ($CbCr$) is copied from a filtered version of the artifact image. We use a joint-bilateral filter as described in Section 5.2.4. For each artifact pixel a local image block (16×16 pixel) is extracted and rectified, which is then used to construct an index to query a database for the k-nearest neighbors (k-nn). This database is initially generated from our training image pairs and contains tens of thousands of rectified reference lighting patches together with the artifact descriptor index. As a descriptor we use the downsampled luminance (8×8) of the rectified artifact patch multiplied with a Gaussian envelope to penalize off-center pixels. In order to detect also large scale patterns, we use a multi-scale search and extract image blocks from the first l levels ($l = 2$) of a Gaussian pyramid. Therefore, the k retrieved reference patches are first upsampled (bicubic) to the corresponding scale of the search descriptor, then cropped

to our patch resolution, and blended according to their k-nn distance norm (L_1) before being warped to image space using the computed texture parametrization. Finally, after all pixels are sampled, the material is added back and the image is blended with the original image as described previously. The main algorithm steps are illustrated in Fig. 5.

8. Perceptual Normalization of Image Contrast

While a binary metric that detects the presence or absence of common rendering artifacts is useful, for most practical purposes it is also desirable to predict the perceived strength of these artifacts. The prediction of the perceived strength of artifacts in no-reference metrics involves additional challenges due to the absence of a reference image (I_{ref}). At a conceptual level, full-reference metrics often assume that the evaluated test image I_{tst} is simply I_{ref} plus some distortions D, and thus, D can be obtained by $I_{ref} - I_{dst}$. Without I_{ref}, obtaining D from I_{dst} is not trivial. To that end, we take advantage of the observation that the rendering artifacts we consider are of medium to high frequency, and approximate I_{ref} via inpainting I_{tst} (Section 7).

Given a rendered image, we employ a multiscale luminance contrast perception model [MDK08] to compute the hypothetical supra-threshold HVS response. The outcome of this computation is perceptually linearized local contrast of the input image. To do so, we first compute a 6-level Laplacian pyramid of image luminance L. Then, a Wilson's transducer [Wil80] function T is applied at each pyramid level L_k. The transducer function operates on HVS-referred values which take human spatial contrast and adaptation luminance sensitivity into account. The luminance adaptation map is approximated by the low-pass residue of the Laplacian pyramid.

The process above is repeated separately for I_{ref} and I_{tst}: given the luminance differences L_k and HVS sensitivities S, the transducers non-linearity models the contrast self-masking properties of the visual system at each pyramid level k. The differences of HVS responses scaled in Just Noticeable Difference (JND) units are then combined using a Minkowski summation with exponent 2. Formulae and implementation details are summarized in the supplementary material.

9. Results

We have tested our method on a set of 24 images generated from several 3D scenes (subset shown in Fig. 1), composed of 6 images containing glossy VPL artifacts, 12 images with shadow map artifacts, and 6 with VPL clamping artifacts. These images were rendered with different software: a GPU-based deferred renderer, an instant radiosity (VPL) renderer, a pathtracer and lightcuts [WFA*05] implementation, for producing the shadow map artifacts, glossy

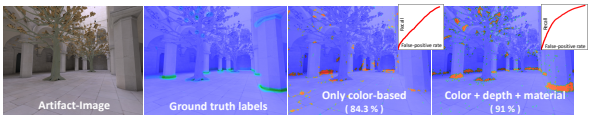

Figure 6: *Additional depth and material information improves the artifact detection significantly (4th image) compared to pure image-based classification (3rd image).*

VPL noise and clamping bias, and reference images, respectively. Each image has a corresponding depthbuffer, diffuse material buffer and a reference image. This small number of images may seem too low compared with pure image classification. However, remember that we train the classifier only on small multi-resolution subimages, which are also of low-dimension due to our decorrelation with geometry and material.

For training each artifact we extracted approximately 15.000 randomly sampled subimages (50% positive and 50% negative samples) from all images excluding the one for present testing. The most discriminative features we have found (which are also shown in Table 1) are: SSAO (Section 5.2.2), rectified depth histograms of oriented gradients (HoG) (see Section 5.2.3), rectified light HoG (i.e., color without textures as described in Section 5.2.1), multi-scale light, depth, and material statistics (i.e., variance, skewness, kurtosis, Section 5.2.5), and frequency analysis of the difference of bilateral-filtered images (bilateral DCT) (Section 5.2.4). SSAO is very effective in detecting clamping bias but only in combination with other features since isolated, it always predicts clamping even in reference images. The most important feature overall is the (rectified) HoG for color, which also slightly outperformed the bilateral DCT feature. In general, having more information behind the pixels clearly improves classification as shown in Fig. 6. We tested our descriptors on SVMs and approximate k-nearest neighbor (k-nn) classifiers (with 5 k-nn). The difference is quite diverse. For the shadow artifacts SVM clearly outperformed k-nn (approx. 10% smaller error) whereas for relatively fuzzy artifacts, clamping and VPL noise, both methods performed similar. Therefore, in our results we only provide results for SVM.

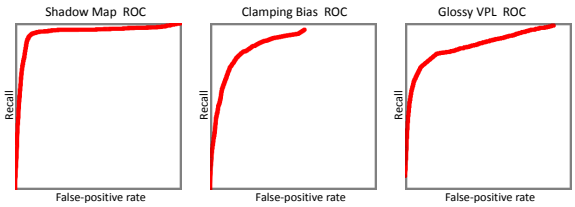

Figure 7: *Mean ROC curves for the shadow map (left), VPL clamping (center), and glossy VPL artifacts (right).*

A visualization of our detected artifacts versus ground truth user annotations is shown in Fig. 8. Further, numerical results and statistics can be found in Table 1 and in

the average *receiver operating charateristics* (ROC) curves (Fig. 7) for the different artifact classes. The classification works best for the shadow mapping artifacts. This is not surprising as shadow aliasing has usually a distinctive regular structure and high contrast, whereas the VPL clamping bias and glossy noise is difficult to address locally and without the global scene knowledge might be mistaken as shadow or highlights, respectively. Moreover, the initial user labeling is very subjective and any mistake (wrongly marked or missing artifact label) confuses the classifier rendering the problem much more complex and noisy, which also shows the downside of a data-driven approach. However, we highlight (in Sections 8,10) how to transform the initial noisy classification into a perceptualized output in form of a distortion map, which is comparable with reference-based VDPs. Besides, we should stress that some of our training examples (clamping bias) exhibit only subtle artifacts, which were even confused by human subjects in our user study.

Class	Features	SVC (C, γ)	Img. #	Acc. [%]	1-BER [%]
Shadow	Light-HoG-16×4×4, Light Bilateral DCT, Depth (Skew)	31.1, 0.036	#5	95.7	89.5
		31.1, 0.036	#6	96.2	84.7
		31.1, 0.036	#7	91.2	69.2
		31.1, 0.036	#8	86.6	90.4
Clamping	SSAO, Depth-HoG-16×2×2, Light-HoG-16×3×3, Light (Skew), Mat. (Var, Kurt)	10.3, 0.027	#9	92.0	74.2
		10.3, 0.027	#10	91.6	58.8
VPL noise	SSAO, Depth-HoG-16×2×2, Light-HoG-16×3×3, Light: (Var, Kurt), Depth: (Var, Skew), Mat.: (Var, Skew)	9.2, 0.02	#1	91.2	65.6
		9.2, 0.02	#2	85.0	68.2
		9.2, 0.02	#3	95.0	89.9
		9.2, 0.02	#4	75.8	71.6

Table 1: *The classification accuracy (Acc) and balanced error rate (BER) for different artifacts together with classification parameters (SVC) and the optimized feature set for each artifact type. The 3 dimensions of the HoG feature define the angular, spatial-X, and spatial-Y resolution of the histogram, respectively. The statistics over local light, depth, material (Mat.) regions are Variance (Var), Skewness (Skew), Kurtosis (Kurt). Corresponding predictions are shown in Fig. 8.*

The inpainting procedure works well for diffuse surfaces and even better for textured surfaces, which mask small inpainting errors (see Fig. 8, 2 last rows). On glossy surfaces with smooth low-frequency gradients and color bleedings inpainting seams may become visible but the overall quality is still improved. In particular, the shadow map artifacts are easy to cure and are perceptually hard to distinguish from the reference. However, there are also a few challenges. First, there is a tradeoff between patch size and reconstruction quality. If the patches are too small the artifact structure might be overlooked (e.g., for the shadow map aliasing we need a larger window to recognize the "jaggy" structure of the edge), whereas too large patches quickly increase the search space (curse of dimensionality) and produce overly blurred results. Besides, the larger the variety of image patches in the database, the better is the resulting inpainting quality. Currently, we extract in total around 50.000 16×16 patch pairs from the first 2 pyramid levels of the reference and artifact images. The patches also compress well and any dimension reduction (e.g., via PCA) would further speed up the inpainting and reduce the memory footprint considerably. Such improvements we leave as future work.

In general, the results of the inpainting procedure are subjectively better than the original distorted images, but they are not perfect and may still exhibit perceivable differences to artifact-free images. However, the main purpose of the inpainting step is to generate a pseudo-reference that makes perceptual normalization possible resulting in clearly improved quality of the distortion maps, as one may see in Fig. 8 (row 8) as well as in correlation values (Table 2).

10. User Study

We performed a subjective user study to validate the prediction performance of our metric. To our best knowledge this was the first attempt to subjectively label locations of visual artifacts caused by rendering techniques both in with- and without- the reference setups. Furthermore, we performed the comparison of existing full-reference metrics, which were not validated for the detection of rendering artifacts before. In this section, we summarize the obtained results, please refer to the supplementary material for a detailed discussion of the user study.

In the experiment, we displayed the set of 10 rendered test images (see Fig. 1) on a calibrated monitor to a group of 20 observers (15M/5F, aged 21–38, all of whom had normal or corrected vision). The observers were asked to mark the perceived artifact regions using a custom scribbling application. We performed two experiments: *with-the-reference*, where an image exhibiting rendering artifacts was presented along with the reference image; and *without-the-reference*, where subjects saw only the distorted image.

The marked regions for each trial were stored as distortion maps, which were then averaged over all subjects to find the mean subjective response. Next, the metric prediction for the corresponding stimulus was computed. Besides our proposed metric, we involved two full-reference metrics in the evaluation. Results of the experiment are visually summarized in Fig. 8. For the numerical analysis, we computed the 2D correlation between the mean subjective response and the metric prediction (for each test image and each experiment separately), as shown in Table 2.

Interestingly, the subjective distortion maps show apparent agreement between the artifact perception experiments in the presence and absence of the reference, which is corroborated by high correlation values (second column in Table 2). The exceptions are images 9 and 10, where the perceptual strength of clamping bias artifacts is rather low. The subjects are seemingly able to mark strong artifacts quite accurately without seeing the reference, while for perceptually weak artifacts, the reference is needed.

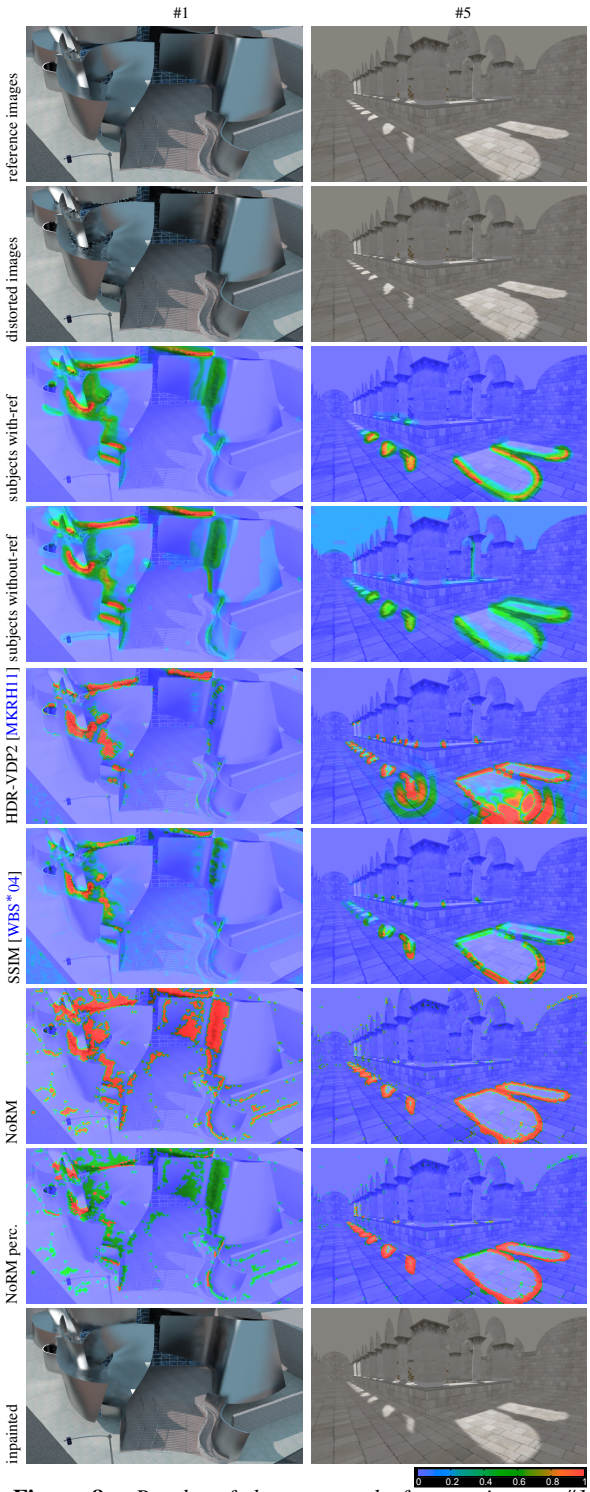

Figure 8: *Results of the user study for test images #1 and #5: average subjective artifact strenghts, and the comparison to predictions of current state-of-the art full reference metrics as well as the proposed no-reference technique. (Please refer to the supplementary material for all the images.)*

Distortion maps produced by classifier (NoRM) are binary, meaning the presence or absence of an artifact. These distortion maps sometimes tend to show too many locations, which may be correct, but the artifact severity is in reality obviously not uniform. However, thanks to the inpainting procedure, we are able to perform the perceptual normalization step (Section 8), which makes the strength of detected artifacts substantially closer to average subjective distortion maps. The prediction after the perceptual normalization (NoRMperc.) is a continuous supra-threshold distortion map calibrated in JND (just noticeable differences) units.

We compared the predictions of the proposed no-reference metric NoRM, with the state-of-the-art full-reference metrics HDR-VDP2 [MKRH11] and SSIM [WBS*04]. Neither HDR-VDP2 nor SSIM were designed or calibrated to predict the strength of rendering artifacts, but the distortion maps they produce are quite plausible. According to average correlations to the subjective ground truth distortion maps, SSIM slightly outperforms HDR-VDP2 (0.56 vs 0.535). The result of our metric (0.534) is qualitatively quite similar, making it competitive with current full-reference metrics in the targeted application. Finally, the perceptual normalization step makes predictions of NoRMperc. even closer to the experimental ground truth, resulting in the highest average correlation (0.586).

Image #	subj. no-ref.	HDR-VDP2	SSIM	NoRM	NoRM perc.
1	0.903	**0.725**	0.674	0.628	0.662
2	0.908	0.579	0.538	0.558	**0.590**
3	0.828	**0.778**	0.643	0.682	0.727
4	0.913	**0.495**	0.469	0.298	0.436
5	0.769	0.542	0.602	0.677	**0.748**
6	0.772	0.669	0.742	0.638	**0.767**
7	0.857	0.390	0.374	0.383	**0.479**
8	0.805	0.618	**0.692**	0.607	0.657
9	0.510	**0.418**	0.231	0.416	0.320
10	0.186	0.134	**0.637**	0.450	0.470
Average	0.745	0.535	0.560	0.534	**0.586**

Table 2: *Correlations of subjective responses in* with-the-reference *experiment with subjective responses in* no-reference experiment *and with the predictions of HDR-VDP2, SSIM, NoRM and NoRM after the perceptual normalization. The last row shows the average correlations over the test set. The best correlations (excluding the no-reference subjective experiment) for each stimulus are printed in bold.*

11. Conclusions and Future Work

In this paper, we proposed a novel learning based *no-reference* image quality metric for computer-generated images, which, as shown in our user study is competitive in performance with state-of-the-art visual difference predictors that *do* require a reference. Our work enables detecting and partially removing rendering artifacts. An important result of this work is that the depth and partial material information used in conjunction with color data drastically improves the classification, and even the inpainting procedure (see Fig. 6). We also present the first comparative subjective study of quality metrics on synthetic images.

Exploring further sources of information as well as classification techniques is a natural future direction. Also, a more challenging problem is quality assessment of the images with multiple types of artifacts. In the future we would like to investigate the classification of combined artifacts using a multi-class classifier and imposing a smoothness prior on the classified labels which could be facilitated by adopting e.g., Markov random fields using *belief propagation* in order to spatially smooth the labels while incorporating the correlations between different artifacts.

Acknowledgements

Many thanks to Tomáš Davidovič for VPL renderings, Mario Fritz, Chuong Nguyen, and Rafał Mantiuk for fruitful discussions, and to anonymous reviewers for suggestions on readability improvements. This work was partly supported by COST Action IC1005.

References

[BLL*10] BERGER K., LIPSKI C., LINZ C., SELLENT A., MAGNOR M.: A ghosting artifact detector for interpolated image quality assessment. In *Proc. IEEE International Symposium on Consumer Electronics (ISCE)* (2010), pp. 52–57.

[BM98] BOLIN M., MEYER G.: A perceptually based adaptive sampling algorithm. In *Proc. SIGGRAPH* (1998), pp. 299–309.

[CCB11] CHEN C., CHEN W., BLOOM J. A.: A universal reference-free blurriness measure. In *SPIE vol. 7867* (2011).

[DF04] DALY S., FENG X.: Decontouring: Prevention and removal of false contour artifacts. In *Proc. of Human Vision and Electronic Imaging IX* (2004), SPIE, vol. 5292, pp. 130–149.

[GGSC96] GORTLER S. J., GRZESZCZUK R., SZELISKI R., COHEN M. F.: The lumigraph. In *Proc. of SIGGRAPH* (1996), ACM, pp. 43–54.

[HJJ10] HACHISUKA T., JAROSZ W., JENSEN H. W.: A progressive error estimation framework for photon density estimation. In *ACM Transactions on Graphics (Proc. of SIGGRAPH Asia)* (2010), pp. 144:1–144:12.

[KA91] KIRK D., ARVO J.: Unbiased sampling techniques for image synthesis. In *Proc. of SIGGRAPH* (1991), pp. 153–156.

[Kel97] KELLER A.: Instant radiosity. In *Proceedings of SIGGRAPH* (1997), pp. 49–56.

[KFB10] KŘIVÁNEK J., FERWERDA J. A., BALA K.: Effects of global illumination approximations on material appearance. In *ACM Transactions on Graphics (Proc. of SIGGRAPH)* (2010), pp. 112:1–112:10.

[KSGH09] KILNER J., STARCK J., GUILLEMAUT J., HILTON A.: Objective quality assessment in free-viewpoint video production. *Signal Proc.: Image Comm. 24*, 1-2 (2009), 3–16.

[LH11] LIU H., HEYNDERICKX I.: Issues in the design of a no-reference metric for perceived blur. In *SPIE vol. 7867* (2011).

[LSAR10] LIU C., SHARAN L., ADELSON E., ROSENHOLTZ R.: Exploring features in a bayesian framework for material recognition. In *23rd IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2010), pp. 239–246.

[MB10] MOORTHY A., BOVIK A.: A two-step framework for constructing blind image quality indices. *IEEE Signal Processing Letters 17*, 5 (2010), 513 –516.

[MDK08] MANTIUK R., DALY S., KEROFSKY L.: Display adaptive tone mapping. In *ACM Transactions on Graphics (Proc. of SIGGRAPH)* (2008), vol. 27(3), pp. 68:1–68:10.

[MKRH11] MANTIUK R., KIM K. J., REMPEL A. G., HEIDRICH W.: HDR-VDP-2: a calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM Transactions on Graphics (Proc. of SIGGRAPH)* (2011), 40:1–40:14.

[MRT99] MYSZKOWSKI K., ROKITA P., TAWARA T.: Perceptually-informed accelerated rendering of high quality walkthrough sequences. In *EGSR* (1999), pp. 5–18.

[PGB03] PÉREZ P., GANGNET M., BLAKE A.: Poisson image editing. *ACM Trans. on Grap. (Proc. of SIGGRAPH)* (2003), 313–318.

[RFWB07] RAMANARAYANAN G., FERWERDA J., WALTER B., BALA K.: Visual equivalence: towards a new standard for image fidelity. In *ACM Trans. on Grap. (Proc. of SIGGRAPH)* (2007), pp. 76:1–76:11.

[RPG99] RAMASUBRAMANIAN M., PATTANAIK S. N., GREENBERG D. P.: A perceptually based physical error metric for realistic image synthesis. In *Proc. SIGGRAPH* (1999), pp. 73–82.

[RSC87] REEVES W. T., SALESIN D. H., COOK R. L.: Rendering antialiased shadows with depth maps. In *Proc. of SIGGRAPH* (1987), pp. 283–291.

[RWD*10] REINHARD E., WARD G., DEBEVEC P., PATTANAIK S., HEIDRICH W., MYSZKOWSKI K.: *High Dynamic Range Imaging*. Morgan Kaufmann Publishers, 2nd edition, 2010.

[SBC05] SHEIKH H., BOVIK A., CORMACK L.: No-reference quality assessment using natural scene statistics: JPEG2000. *IEEE Trans. on Image Processing 14*, 11 (2005), 1918 –1927.

[SBC10] SAAD M., BOVIK A., CHARRIER C.: A DCT statistics-based blind image quality index. *IEEE Signal Processing Letters 17*, 6 (2010), 583 –586.

[SFWG04] STOKES W. A., FERWERDA J. A., WALTER B., GREENBERG D. P.: Perceptual illumination components: a new approach to efficient, high quality global illumination rendering. In *Proc. of ACM SIGGRAPH* (2004), pp. 742–749.

[SGW06] SCHMIDT R., GRIMM C., WYVILL B.: Interactive decal compositing with discrete exponential maps. *ACM Transactions on Graphics (Proc. of SIGGRAPH) 25*, 3 (2006), 605–613.

[Sim05] SIMONCELLI E. P.: Statistical modeling of photographic images. In *Handbook of Image and Video Processing* (2005), Bovik A. C., (Ed.), Academic Press, Inc., pp. 431–441.

[Sze10] SZELISKI R.: *Computer Vision: Algorithms and Applications*. Springer, 2010.

[TJ97] TAMSTORF R., JENSEN H. W.: Adaptive sampling and bias estimation in path tracing. In *EGSR* (1997), pp. 285–295.

[WB06] WANG Z., BOVIK A. C.: *Modern Image Quality Assessment*. Morgan & Claypool Publishers, 2006.

[WBS*04] WANG Z., BOVIK A. C., SHEIKH H. R., MEMBER S., SIMONCELLI E. P., MEMBER S.: Image quality assessment: From error visibility to structural similarity. *IEEE Transactions on Image Processing 13* (2004), 600–612.

[WFA*05] WALTER B., FERNANDEZ S., ARBREE A., BALA K., DONIKIAN M., GREENBERG D.: Lightcuts: A scalable approach to illumination. *ACM Transactions on Graphics (Proc. of SIGGRAPH)* (2005), 1098–1107.

[Wil80] WILSON H.: A transducer function for threshold and suprathreshold human vision. *Biological Cybernetics 38* (1980), 171–178.

[Win05] WINKLER S.: *Digital Video Quality: Vision Models and Metrics*. Wiley, 2005.

[WPG02] WALTER B., PATTANAIK S. N., GREENBERG D. P.: Using perceptual texture masking for efficient image synthesis. *Computer Graphics Forum 21*, 3 (2002), 393–399.

[WR05] WU H., RAO K.: *Digital Video Image Quality and Perceptual Coding*. CRC Press, 2005.

[WRC88] WARD G. J., RUBINSTEIN F. M., CLEAR R. D.: A ray tracing solution for diffuse interreflection. In *Proceedings of ACM SIGGRAPH* (1988), pp. 85–92.

[YPG01] YEE H., PATTANAIK S., GREENBERG D. P.: Spatiotemporal sensitivity and visual attention for efficient rendering of dynamic environments. *ACM Transactions on Graphics 20* (2001), 39–65.