



# High Dynamic Range Image Rendering with a Retinex-Based Adaptive Filter

Laurence Meylan, *Student member, IEEE*, Sabine Süssstrunk, *Member, IEEE*

**Abstract**— We propose a new method to render high dynamic range images that models global and local adaptation of the human visual system. Our method is based on the center-surround Retinex model. The novelties of our method is first to use an adaptive filter, whose shape follows the image high contrast edges, thus reducing halo artifacts common to other methods. Secondly, only the luminance channel is processed, which is defined by the first component of a principal component analysis. Principal component analysis provides orthogonality between channels and thus reduces the chromatic changes caused by the modification of luminance. We show that our method efficiently renders high dynamic range images and we compare our results with the current state of the art.

## I. INTRODUCTION

**R**EPRODUCING an image that corresponds to direct observation of a scene is a non-trivial task. Indeed, the scene irradiance captured by the human visual system (HVS) is processed in a highly non-linear manner. Digital cameras also capture the irradiance of the scene, but process it in a way that differs from the treatment of the HVS. Consequently, the captured image often changes from the human perception of the original scene. This difference is especially visible for high dynamic range (HDR) scenes. Dynamic range is defined by the luminance ratio between the brightest and the darkest object in a scene. A scene or an image is said to be HDR when its dynamic range exceeds by far the one of the capture or display device. Scenes can have three or more orders of magnitude of luminance variations (1000:1), which can be perceived by our visual system by means of adaptation [1]. A typical CRT monitor can approximatively display 2 orders of magnitude (100:1). A compression of the dynamic range is required when the scene dynamic range exceeds the one of the display device.

The purpose of our method is to reproduce HDR images on low dynamic range (LDR) display devices. This is commonly called tone mapping or tone reproduction. We do not address the problem of capturing HDR images; we use existing HDR image databases.

Simple tone mapping algorithms globally compress the dynamic range using a logarithmic function, gamma function or sigmoidal function. These point-to-point matching functions are fast and suitable for LDR scenes but are not always sufficient to render HDR scenes, whose dynamic range exceeds by far the dynamic range of display devices. Processing an HDR image only globally can cause a loss of contrast, which



Fig. 1. Example of HDR scene that requires local processing. **Left:** Image rendered using a gamma correction. **Right:** Image rendered with the Retinex-based adaptive filter proposed in this article that combines global compression and local processing.

is apparent in the loss of detail visibility. A local processing is thus necessary in addition to global compression for the reproduction to be visually appealing. Local processing allows to increase the local contrast, which increases the visibility of some parts of the image while the global compression scales the image's dynamic range to the output device's dynamic range. A local processing is also a better imitation of the HVS, which adapts locally to each part of a scene in order to form a percept where all details are visible [2], [3].

Figure 1 shows an example of an HDR scene that requires local processing. The left image was rendered using just a global tone mapping (gamma correction) and the right image was rendered using the method presented in this article that combines global compression and local processing. We observe that our method retrieves details in the central, shadowed part of the scene while just applying a gamma correction leaves the central part too dark.

We have developed a local tone mapping algorithm that solves the problems encountered by previously developed methods, namely it does not produce halo artifacts nor local “graying-out” artifacts, and provides good color rendition. Halo artifacts are due to the proximity of two areas of very different intensity. For example, if a dim area is close to a bright window, the bright pixels can influence the processing of the dim area and can cause a black halo around the bright area. Moreover, local filtering tends to make pure black and pure white low contrast areas turn gray. These phenomena are illustrated in Figure 2. The shadow on the face is a halo artifact due to the background window. The black t-shirt looks washed out due to the local filtering. Figure 2 was computed with our previous algorithm that combines several Gaussian constants into a single filter [4]. The filter had a circular shape for every pixel.

Our method belongs to the surround-based Retinex algorithms [4], [5], [6] but differs in the way the surround is



Fig. 2. Example of halo artifacts and graying-out. The shadow on the face is a halo artifact due to the background window. The black t-shirt looks washed out due to the local filtering.

defined as well as in the way the color channels are treated. Rather than applying Retinex independently to R,G,B color channels, we perform a principal component analysis (PCA) on the input image and define an image-dependent linear transform. Then, applying this transform to the R,G,B image, we obtain an image whose first channel represents the luminance and the other two channels represent chrominance. Retinex is applied to the luminance channel only. The PCA provides orthogonality among components and allows the color to remain relatively stable despite the processing of the luminance. Concerning the surround definition, we replace the traditional circular shape by an adaptive surround, whose shape follows the high contrast edges and thus prevents halo artifacts. Such spatial processing requires a greater computational time than global methods. We address this by applying spatial adaptive filtering on a downsampled version of the image.

This article is organized as follows: Section II presents background information and related algorithms. The Retinex-based adaptive filter method is described in Section III while the color processing is detailed in Section IV. Section V analyzes the computational complexity of our method and proposes a way to reduce it by computing the mask on a downsampled version of the image. Section VI provides comparisons with other methods that are state of the art; we discuss the different properties of these methods and show our results. Section VII concludes this article.

## II. BACKGROUND

Tone mapping algorithms intend to reproduce images that are similar to what a person perceived when observing the corresponding scene. The reader is referred to [7] for a general review of tone mapping algorithms. Here, we only discuss the methods relevant to our research. We first introduce human local adaptation and models. Then, we present the Retinex theory of color vision that also models human local adaptation, followed by computational models based on it. We show that most of these models cause halo artifacts. Third, we present algorithms that propose a solution to reduce halo artifacts. Finally, we present some existing solutions for color image processing in the presence of local tone mappings.

### A. Models of HVS local adaptation

The HVS deals with the high dynamic range of natural scenes by means of adaptation [2], [3]. We consider two

kinds of adaptations: global and local adaptation. Global adaptation is the adaptation to the average light level while local adaptation is a local gain control that takes place as we visually scan a scene.

HVS models of global and local adaptation have been developed and applied to images. Spitzer and Semo [8] present a model for color contrast based on retinal mechanisms and adaptation. It simulates the properties of opponent cells and double opponent cells to predict the perceived image. Pattanaik et al.'s model [3] is recognized as one of the most comprehensive computational model of adaptation and spatial vision. It addresses the problem of realistic tone reproduction based on multi-scale processing of the retinal image and adaptation processes. Alleysson and Süssstrunk [9] propose a three layer model of the retina that includes a non-linear encoding and two adaptation levels. The first adaptation is provided by the photoreceptors' adaptive sensitivity, which forms the first layer of the model. The second layer is responsible for opponent encoding, and thus provides normalization and contrast gain control. Then, the third layer is formed by the ganglion cells that also provides an adaptive non-linearity.

These algorithms were shown to accurately model psychovisual data. However, they are not convenient for image processing due to complexity and poor visual results. What we aim to develop is an image processing method that is biologically plausible and that renders visually pleasing images. For that purpose, we take inspiration from Retinex, a model of color vision that has been widely used for image processing tasks.

### B. Retinex computational models for local tone mapping

Retinex is a theory of color vision developed by Land that intends to explain how the visual system extracts reliable information from the world despite changes of illumination [10], [11]. It is based on a series of experiments carried out with a flat surface composed of color patches and three controllable independent light sources. Land showed in his experiments that there is little correlation between the amount of radiation falling on the retina and the apparent lightness of an object. He concludes that the perceived color of a unit area is determined by the relationship between this unit area and the rest of unit areas in the image, independently in each wave-band, and does not depend on the absolute value of light.

Retinex is a simplified model of the HVS and can conveniently be adapted for computational image rendering algorithms. A variety of computational models have been implemented. Their common principle is to assign a new value to each pixel in the image based on spatial comparisons of light intensities. Their differences are the order in which the pixels are addressed, as well as the distance weighting functions. The primary goal of Retinex is to decompose the image into the reflectance image and the illuminant image to remove illumination effect.

A first version of Retinex was developed by Land [10]. It computes subsequent additions of pixel differences along a set of one-dimensional random paths contained in the image. The new value of each pixel, which represents the reflectance image, is determined by the average over all paths. Horn

reformulated Land's Retinex and showed that the illuminant can be estimated using a two-dimensional Laplacien [12]. Hurlbert formalized the Retinex theory mathematically and showed that it is equivalent to solving a Poisson equation [13]. Another theoretical study of Retinex is provided by Brainard and Wandell [14]. They study the convergence properties of Land's Retinex and show that, as the number of paths and their lengths increases, the result converges to a simple normalization. A further development introduces randomly distributed paths using Brownian motion [15]. Practically, the problems with these Retinex *path-based* methods are their high computational complexity and the free parameters, such as the number of paths, their trajectories, and their lengths.

The *iterative* version of Retinex is a two-dimensional extension of the path version. It computes a new value for each pixel by iteratively comparing pixels in the image [16]. The drawback of this implementation is that the number of iterations is not defined and has a critical effect on the final result. Although a method exists that automatically defines the number of iterations using an early stopping technique [17], it remains an important issue. Sobol [18] contributes to the improvement of the Retinex iterative version by introducing a ratio modification operator. This operator allows better compression in high contrast areas while increasing the visibility in low contrast areas.

*Surround-based* Retinex computational models are non-iterative. Each pixel is selected sequentially and treated only once. New pixel values are given by the ratio between each treated pixel and a weighted average of its surround. This technique was first proposed by Land [5] and then used by Rahman et al. [6], [19] for their multi-scale Retinex with color restoration (MSRCR). The multi-scale version is an extension of the single-scale Retinex that aims to reduce halo artifacts induced by the single-scale method. It is obtained by averaging three single-scale Retinex using three different spatial constants. The single-scale Retinex computes the new value of each pixel by taking the ratio between the treated pixel and a weighted average of its surround, whose weights are given by a Gaussian function. The color restoration factor is introduced to compensate for the loss of color saturation inherently present in their method. This color correction greatly enhances the saturation but does not ensure a correct rendition of colors [20]. In a previous article, we propose a method based on MSRCR [4]. Instead of using three single-scale Retinex, we include all spatial constants into a single filter. The algorithm is applied to the luminance channel and no color restoration is applied. iCAM [21] is another rendering algorithm based on spatial properties of vision (local adaptation and spatial filtering). Unlike other surround-based methods, it is a complete model of image appearance and quality. Moreover, it was specifically developed to render HDR images.

An interesting approach to Retinex is provided by Kimmel et al. [22], [23]. They reformulate the Retinex theory, which is equivalent to illumination estimation, and show that it can be formulated as a quadratic programming optimization problem. As decomposing the image into reflectance image and illumination image is mathematically ill-posed

[24], they redefine the problem using physically motivated considerations such as illumination smoothness and limited dynamic range of the reflectances. Adding these constraints allows their algorithm to converge to a unique solution, the optimal illumination. However, the computational complexity of quadratic programming optimization is high since each pixel is an unknown to the minimization formula. In a later article [23], they propose several methods for reducing the complexity of the above approach by restricting the solution to have a pre-defined structure using either a look-up table, linear or non-linear filters, or a truncated set of basis functions. These simplifications involve less free parameters and yield reasonable yet sub-optimal results. An application of their algorithm for gamut mapping was recently published [25].

A common drawback of most local tone mapping methods mentioned above is the possible apparition of halo artifacts around light sources. Indeed, using the assumption that the illuminant is spatially smooth leads to halo artifacts in the presence of high contrast edges. Kimmel et al. [22], [23] added a halo reduction term in their quadratic programming optimization formulation to account for this problem. Similarly, the ratio modification operator of Sobol [18] also contributes to reducing halo artifacts.

### C. Solving the halo artifact problem using other methods than Retinex

The presence of halo artifacts is a well-known issue when rendering HDR images. Using a local operator involves a trade-off between the compression of dynamic range and the rendition of the image. A strong compression leads to halo artifacts while a weak compression does not provide the expected improvement of detail visibility. This problem has been addressed by Tumblin and Turk [26] who propose a method called Low Curvature Image Simplifiers (LCIS) to increase the local contrast while avoiding halo artifacts. LCIS uses a form of anisotropic diffusion to enhance boundaries while smoothing non-significant intensity variations. This method does not take into account whether details are visibly significant. Consequently, the resulting images tend to look unnatural due to excessive detail visibility. Fattal et al. compress the dynamic range using a gradient attenuation function defined by a multiresolution edge detection scheme [27]. A new low dynamic range image is obtained by solving a Poisson equation on the modified gradient field. This approach provides good results but requires parameter tuning. Reinhard et al. developed a local method based on the photographic dodging and burning technique [28]. They use a circular filter, whose size is adapted for each pixel by computing a measure of local contrast. A related method was proposed by Ashikmin [29], which computes a measure of the surround luminance for each pixel. This measure is then used for the definition of the tone mapping operator. Both methods provide an efficient way of compressing the dynamic range while reducing halo artifacts. However, the restriction to a circular surround limits their performance. DiCarlo and Wandell investigated tone mapping algorithms and the creation of halo artifacts [30]. They suggest the use of robust operators to avoid these.

A robust Gaussian includes a second weight that depends on the intensity difference between the current pixel and its spatial neighbors. This technique preserves the sharpness of large transitions. A recent approach based on LCIS and robust operators was proposed by Durand and Dorsey [31]. Their method renders HDR images using bilinear filtering, an alternative for anisotropic diffusion. Their method is not a strict tone reproduction in the sense that it does not attempt to imitate human vision. We will compare our method with theirs (see Section VI-C).

A first attempt to compare tone mapping algorithms is published in [32], but assessing the quality of existing tone mapping algorithms is still a major concern.

#### D. Color processing

The way the color is processed by tone mapping methods has also been extensively discussed in the literature. Funt and Barnard [20], [33] investigate the MSRCR of Rahman et al. [6], [19]. They argue that MSRCR tends to desaturate the colors, due to the averaging operation on small neighborhoods that have a graying-out effect on the image. Moreover, the color restoration step added to compensate for the loss of saturation can at best approximate the color that was removed and acts in a unpredictable way. Funt and Barnard propose a color preserving multi-scale Retinex that is applied to the luminance channel. A post-processing is also added to enhance the image saturation. Monobe et al. propose a method to preserve local contrast under different viewing conditions [34]. Their method treats luminance only. They found also that the choice of the color space greatly influences the final image. Yang and Rodriguez propose two methods to process luminance while minimizing the chromatic changes [35]. They use shifting and scaling properties of the LHS and YIQ spaces.

Kimmel et al. developed their quadratic programming optimization algorithm for a monochrome image [22], [23]. Then, they apply it separately either to the three color channels of an RGB image or to the V channel of an HSV-encoded image. They found that the first approach could lead to exaggerate color shifts or to a loss of saturation. As previously found [20], [33], applying it only to the V channel yields better results. In [18], Sobol proposes to apply its iterative Retinex-based method to the luminance channel only. Unlike previously mentioned methods that define the luminance as a weighted sum of R,G,B color channels, his luminance definition is given by the maximum between these three channels. The final color image is obtained by adding the new luminance to the log-encoded RGB image.

In this article, we propose a surround-based Retinex method to render HDR images that uses an adaptive filter whose shape follows the image contours. By adapting both the shape and the size, it is more flexible than previous surround-based methods and thus better prevents halo artifacts. The local processing is applied to the luminance channel only. We use a color transformation based on PCA to ensure good color rendition.

### III. THE RETINEX-BASED ADAPTIVE FILTER METHOD: LUMINANCE PROCESSING

The global framework of our method is illustrated in Figure 3. Luminance and chrominance are processed in parallel but only the luminance is treated by the Retinex-based adaptive filter method (right part of Figure 3). The luminance  $\Psi$  is given by the first principal component of the input image, which is linear with respect to scene radiance. A first global compression is applied to both the luminance image  $\Psi$  and the linear RGB image  $I$ . Then, we apply the Retinex-based adaptive filter method in the log-domain to the globally corrected luminance  $\Psi'$  (see III-B) while a logarithm is applied to the globally corrected RGB image  $I'$ .  $I'$  is then transformed into a luminance chrominance encoding through PCA. Its first component  $C_1$  is replaced by the treated luminance  $\Psi_{new}$  and the image thus obtained is transformed back to RGB.

This section presents first the global tone mapping and then the Retinex-based adaptive filter method that is applied to the luminance channel only. The color image processing including PCA transformation and saturation enhancement is described in Section IV.

#### A. Step 1: global tone mapping

Our method consists of two parts: a preliminary global tone mapping followed by Retinex-based local processing. The global tone mapping that is applied to the linear image performs a first compression of the dynamic range. It can be compared to the early stage of the visual system where a global adaptation takes place [9], [36]. We design our global tone mapping function to be similar to the adaptation of photoreceptors, which can be approximated by a power function. The curvature of the function that determines the adaptation state depends on the mean luminance of the total field of view [9]. Consequently, we compute the exponent of the power function from the average luminance of the image.

We define the luminance image as the first principal component of the linear image  $I$ . Let  $\Psi$  be the luminance image encoded linearly, whose maximum value is 1. The non-linear luminance  $\Psi'$  is given by

$$\Psi' = \Psi^{\frac{1}{\gamma}}, \quad (1)$$

where the value of  $\frac{1}{\gamma}$  is an affine function of the average luminance in the image,  $\overline{\Psi_{al}}$  (3):

$$\frac{1}{\gamma} = \min\left(1, \frac{1}{6}\overline{\Psi_{al}} + \frac{2}{3}\right). \quad (2)$$

The coefficient of the affine function were defined experimentally as follows: a high or average key image is not globally compressed and is therefore assigned  $\frac{1}{\gamma} = 1$ . As the average luminance decreases, the exponent  $\frac{1}{\gamma}$  decreases, increasing the sensitivity for dark areas. The average luminance  $\overline{\Psi_{al}}$  is computed by taking the average of the log-encoded pixels:

$$\overline{\Psi_{al}} = \frac{\sum_{p \in \Psi} \log(\Psi(p))}{N}, \quad (3)$$

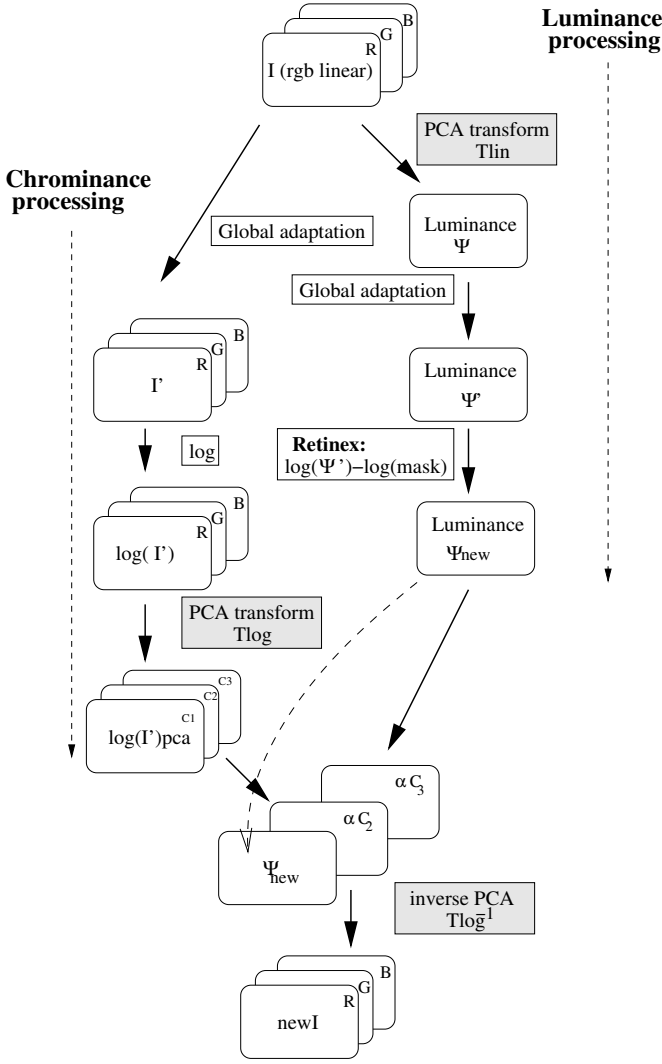


Fig. 3. Global framework. Our method uses parallel processing; one for luminance and one for chrominance. See text for explanation.

where  $N$  is the number of pixels in the luminance image  $\Psi$ , and  $p$  is a pixel value in  $\Psi$ .

### B. Step 2: local adaptation

After global processing, local adaptation is performed using a surround-based Retinex method. Traditionally, surround-based Retinex methods [4], [5], [6] compute a new value for each pixel by taking the difference between the log-encoded treated pixel and the log-encoded value of a mask, as described in (4). The mask represents a weighted average of the treated pixel's surrounding area.

$$\Psi_{new}(x, y) = \log(\Psi'(x, y)) - \log(mask(x, y)), \quad (4)$$

where  $\Psi'$  is the luminance image and  $mask$  is computed by convolving the luminance  $\Psi'$  with a surround function.

A drawback of the surround-based methods is that small filters tend to make pure black or pure white low contrast areas turn gray. This is due to local normalization. We overcome this

problem by introducing a weighting factor  $\beta(x, y)$  that ensures the conservation of white and black areas.

Each pixel value of the luminance image  $\Psi_{new}$  is computed as follows:

$$\Psi_{new}(x, y) = \log(\Psi'(x, y)) - \beta(x, y) \cdot \log(mask(x, y)), \quad (5)$$

where  $(x, y)$  is the pixel coordinate. The image  $\Psi'$  is encoded in floating points in the range  $[0, 1]$ . We temporarily scale it to a larger range and clip it to a minimum value of 0.01 in order to perform the logarithm operation in the range  $[0.01, 100]$ . The log-encoded image is then scaled back to the range  $[0, 1]$  without loss of information.

$$\log(\Psi'(x, y)) = \frac{\log(\max(0.01, \Psi'(x, y) * 100))}{\log(100)}. \quad (6)$$

As for  $\beta$ , it weighs the mask depending on the pixel values at coordinate  $(x, y)$ :

$$\beta(x, y) = 1 - \frac{1}{1 + e^{-7 \cdot (\Psi'(x, y) - 0.5)}}. \quad (7)$$

The  $\beta$  factor is based on a sigmoid function and maps white to white and black to black, which is necessary to obtain visually pleasing images. For a pixel of high intensity, the mask is weighted by a value close to 0. Since the mask is subtracted from the log-encoded luminance, it effectively keeps the pixel bright. Similarly, a pixel of low intensity is weighted by a value close to 1, which has the effect of maintaining black. This function lets the middle gray values change without constraint while restricting the black to remain black and the white to remain white.

Another important drawback of surround-based Retinex methods is that there is a trade-off between the increase in local contrast and a good rendition of the image. A small surround allows a significant increase in local contrast but induces halo artifacts along high contrast edges. Using a larger surround reduces the artifacts, but provides less increase in local contrast. This trade-off has already been mentioned by Jobson et al. [19] and Barnard and Funt [20], remarking that MSRCR caused halo artifacts along high contrast edges.

Our adaptive filter method prevents halo artifacts by adapting the shape of the filter to the high contrast edges in the image. Thus, the filter follows image contours. In this way, a bright area has less influence on the treatment of a neighboring dim area. This modification does not change the principle of Retinex surround-based methods, that is, to compute the difference between each pixel value and a weighted average of its surround. What changes with the adaptive filter method is the way the mask is computed. Since the filter is different for each pixel, it is not possible to use a convolution anymore. The mask is thus computed specifically for each pixel using

$$mask(x, y) = \sum_{\theta=0}^{360} \sum_{r=0}^{r_{max}} \Psi'(x + r \cdot \cos(\theta), y + r \cdot \sin(\theta)) \cdot e^{-\frac{r^2}{\sigma_{\theta, r}^2}}, \quad (8)$$

where  $\theta$  is the angle of the radial direction,  $r$  is the distance to the central pixel and  $\sigma_{\theta, r}$  is defined as follows:

$$\sigma_{\theta,r} = \begin{cases} \sigma_0 & \text{no high contrast edge was crossed along } \theta \\ \sigma_1 & \text{a high contrast edge was crossed along } \theta \end{cases}$$

The value of the mask at coordinate  $(x, y)$  is given by a weighted average of pixels surrounding the position  $(x, y)$ . The weights of surrounding pixels are given by a Gaussian function, whose spatial constant varies according to the image high contrast edges. Practically, it is done by selecting one pixel after the other in a radial manner. The first pixel to be selected is the central pixel. Then, all pixels along a radial direction are added, weighted by a Gaussian function with spatial constant  $\sigma_0$ . If an edge is crossed along the radial direction,  $\sigma_{\theta,r}$  is assigned a smaller value  $\sigma_1$  and keeps the same value until  $r = r_{max}$  with  $r_{max} = 3 \cdot \sigma_0$ . The weighted sum of pixels continues for each direction until the surround is completed. For each new radial direction,  $\sigma_{\theta,r}$  is reset to its initial value  $\sigma_0$ . The weighted sum of pixels is normalized by the sum of weights so that each pixel has an equal contribution to the mask even if it is surrounded by edges. The numerical values for  $\sigma_0$  and  $\sigma_1$  are chosen to be fractions of the image size. Experimentally, we have found that  $\sigma_1$  needs to be at least  $\frac{1}{2}\sigma_0$  to avoid halos in most of our images. We did not use  $\sigma_1 = 0$  to ensure that no artifacts will be introduced by the hard threshold even when the edge is very close to the treated pixel. Some limitations of precision arise from the fact that a pixel has only 8 neighbors. Therefore, there are only 8 possible radial directions to explore from the central pixel. The surround is covered recursively starting from the center. Each pixel is used only once in the weighted sum.

### C. Edge detection

We use a Canny edge detector to detect high contrast edges [37]. The Canny method finds edges by looking for global maxima of the image gradient. It detects strong and weak edges. Weak edges appear in the output only if they are connected to strong edges. The thresholds for strong and weak edges are fixed values chosen experimentally and kept the same for all images. Fixed thresholds are desirable since we only want to detect high contrast edges. It is thus possible to obtain no edges for an image that has no high contrast edge and where circular, non-adaptive surrounds are sufficient.

The construction of the filter according to a segmented image is illustrated in Figure 4. The top and bottom left images represent the original image and its corresponding edge map, respectively. The filter was computed for the pixel indicated by the cross. The bottom right image shows the corresponding adaptive filter.

### D. Post-processing: Histogram scaling

A final processing is needed to remove outliers and to scale the luminance before it is integrated back into the color image processing. This is done using histogram scaling and clipping. 1% of the pixels are clipped at both extremities of the histogram. It is also possible to improve the final result with a gamma correction depending on the output device characteristics.

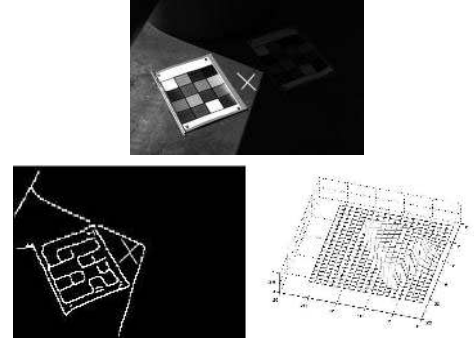


Fig. 4. Construction of the adaptive filter for the pixel indicated by the cross. **Top:** Luminance image  $\Psi'$ . **Bottom left:** Luminance image segmented with a Canny edge detector. **Bottom right:** 3D representation of the filter corresponding to the pixel indicated by the cross.

## IV. THE RETINEX-BASED ADAPTIVE FILTER METHOD: COLOR PROCESSING

Our method takes inspiration from the HVS, which treats chromatic and achromatic data independently. In the retinal pathways, the LMS signals captured by the cones are decorrelated. After being processed by subsequent neural stages, they form two major parallel circuits. One is achromatic and non-opponent. The other is chromatic and opponent [38], [39].

Based on this knowledge, our method uses a principal component analysis (PCA) to decorrelate the RGB representation of the input image into three principal components. Our motivation is that PCA has properties that intrinsically leads to an opponent representation of colors. In [40], Buchsbaum and Gottschalk describe the relation between PCA and the HVS. They show that optimum transformations in terms of information processing, such as PCA, result in one component that is all positive and has the largest share of signal energy. It is the achromatic channel, carrying luminance information. The second and the third components have one zero-crossing and two zero-crossing, respectively. They represent the opponent channels for chrominance: R-G and Y-B.

In the previous section, we described the treatment applied to the luminance channel only. The result of the luminance processing is inserted back into the parallel color image processing as illustrated in Figure 3. After the global power function, we take the logarithm of the RGB image  $I'$  in order to follow the same steps that were applied to the luminance. Then, the log-encoded image  $\log(I')$  is transformed into a decorrelated space to obtain  $\log(I')_{pca}$ . The principal component is replaced by the treated luminance  $\Psi_{new}$  that was computed in parallel and recomposed with the chrominance channels. The chrominance channels are weighted by a factor  $\alpha$ , in order to compensate for the loss of saturation induced by the increase in luminance. The increase in luminance is partly due to the logarithm operation applied to  $\Psi'$ . Since this operation is similar to all images, we use a constant factor  $\alpha$ . We found experimentally that  $\alpha = 1.6$  is a suitable value.

As mentioned before, applying a local processing separately to R,G,B results in color shifts and graying-out of color. The solution is to transform the image into a luminance chrominance encoding. However, with most transforms, some

luminance information remain in the chrominance and vice versa, due to the non-orthogonality of the color space basis vectors. Unlike most transforms, PCA provides an orthogonal representation of the luminance and chrominance components, which results in good color rendition when the luminance is processed. A linear transformation such as YUV [36] provides good color rendition as well, but the resulting images are slightly different. Figure 7 shows an image treated by our algorithm using PCA transform and one using YUV transform. The image computed using YUV looks slightly green but it is hard to justify which one is more visually appealing. The advantages of PCA is its perfect decorrelation between luminance and chrominance. It works well for natural images, which contain a reasonable diversity of colors. However, particular cases such as a singular color image would lead to an ill-conditioned transformation matrix and thus to the failure of the PCA algorithm. This does not happen when treating natural images even in the presence of a color cast, but is more likely to happen with synthetic images.

## V. REDUCING COMPUTATIONAL COMPLEXITY

### A. Luminance processing

The use of an adaptive filter instead of a fixed surround shape involves a significant increase in computational complexity. Indeed, when the filter is the same for each pixel, the mask can efficiently be computed by a multiplication in the Fourier domain. The introduction of the adaptive filter prevents the use of a convolution and therefore makes the method computationally very expensive. The adaptive filter method before simplification has an order of complexity of  $O(N + N^2)$ , where  $N$  is the number of pixels in the image. The first term  $N$  is due to the difference operation and the  $N^2$  term is due to the mask computation.

We propose two solutions to reduce the computational time. The first solution consists in limiting the size of the surround by taking the decay of the Gaussian weighting function into account. Our default value for the radius surround size is  $r_{max} = 3\sigma_0$ . This reduces the computational complexity to  $O(N + N \cdot (2r_{max})^2)$ .

The second solution is to use a downsampled version of the image to compute the mask. The mask is then upsampled before being subtracted from the high resolution image (9). The upsampling and downsampling operations are performed using bilinear interpolation. A similar method to reduce the computational complexity was introduced by Moroney [41].

Let us use the symbol  $\bullet_{\downarrow n}$  for downsampling by  $n$  and  $\bullet_{\uparrow n}$  for upsampling by  $n$ . Equations (5) and (8) for computing the treated luminance become

$$\Psi_{new}(x, y) = \log(\Psi'(x, y)) - \beta(x, y) \cdot \log(mask_{\uparrow n}(x, y)) \quad (9)$$

$$mask(x, y) = \sum_{\theta=0}^{360} \sum_{r=0}^{r_{max}} \Psi'_{\downarrow n}(x, y)(x + r \cdot \cos(\theta), y + r \cdot \sin(\theta)) \cdot e^{-\frac{r^2}{\sigma_{\theta, r}^2}}, \quad (10)$$

where  $n$  is chosen such that the larger dimension of the downsampled image  $\Psi_{\downarrow n}$  equals a constant, whose default value is 200.

Considering this second simplification, the computation time of the mask is fixed and is bounded by  $\frac{N}{n^2} \cdot \frac{(2r_{max})^2}{n^2}$ . That makes the computational complexity of order:

$$O(N + \frac{N}{n^2} \cdot \frac{(2r_{max})^2}{n^2}) \quad (11)$$

### B. Color processing

Section V-A analyzes the computational complexity to process the luminance channel only. In fact, the RGB input image first has to be transformed to a luminance chrominance encoding, which requires  $O(N)$  operations. Processing only the luminance not only provides good color rendition but also requires less computational time. Indeed, treating separately the R,G,B channels would multiply by three the complexity of (11), which is significantly more time consuming than adding an  $O(N)$  operation. The PCA transform costs more in term of computational time than a fixed transform such as YUV. However, the additional time spent to compute the PCA is neglectable compared to the time spent to compute the mask.

## VI. DISCUSSION, COMPARISONS AND RESULTS

In this section we justify the need for an adaptive filter by showing an example where it helps to prevent halo artifacts. Then we compare our method to other local tone mapping methods: the MSRCR of Rahman et al. [6], the gradient attenuation method of Fattal et al. [27] and the fast bilateral filtering method of Durand and Dorsey [31], currently recognized as one of the best published methods [32].

### A. Importance of the adaptive filter

Figure 8 illustrates the difference between using an adaptive filter that follows the high contrast edges in the image and a non-adaptive filter, whose shape is circular for every pixel. The two images were computed with exactly the same method except for the filter's shape. The non-adaptive case was computed with an edge map set to zero everywhere, such that the surround's shape is always circular and does not follow high contrast edges.

The benefit of the adaptive filter is clearly shown in Figure 8: the detail of the tower and in the forest are more visible using the adaptive filter method.

This is due to the edge-preserving properties of the mask as illustrated in Figure 5. The use of the adaptive filter method prevents the areas of different intensity to influence areas beyond high contrast edges.

### B. Comparison with other methods

Our algorithm finds its basis in the MSRCR algorithm of Rahman et al [6]. It is therefore natural to make a comparison with their performances. The MSRCR images were obtained with the free version of the software "PhotoFlair" using the





Fig. 5. The edge-preserving properties of the mask prevents areas of different intensity to influence areas beyond high contrast edges. **Top:** Input image  $\Psi'$ . **Bottom left:** Mask with adaptive filter. **Bottom right:** Mask without adaptive filter.

default settings<sup>1</sup>, which puts “demo” tags across the image. Figure 9 shows a comparison between MSRCR and our adaptive filter method. The benefit of the adaptive filter is clearly seen on both images. With MSRCR, there is a shadow on the person near the window and on the red dog. Moreover, the black t-shirt tends to become gray. Due to the adaptive filter, our method does not generate halos on the face of the person and on the t-shirt. The  $\beta$  factor (5) prevents the t-shirt to turn gray. Similarly on the bottom image, the detail of the tower is more visible on the image treated by the adaptive filter method.

The presence of halo artifacts comes from the fact that MSRCR is based on the assumption that the illuminant is spatially smooth. This results in a mask similar to the bottom right panel of Figure 5, which leads to halos when subtracted from the log-encoded luminance. Other the methods that are based on the same smooth illuminant assumption ([5], [6], [12], [16]) suffer from the same drawback. Nevertheless, they are good at rendering images of lower dynamic range or in the absence of large intensity ratios.

Fattal et al. [27] treat HDR images with a gradient attenuation method. Figure 10 shows their result. Their method is very good at increasing local contrast without creating halo artifacts but the effect tends to be exaggerated. A border effect appear on the left of the image and the colors seem unnatural.

### C. Comparison with fast bilateral filtering

We chose to compare our method to the fast bilateral filtering method developed by Durand and Dorsey [31] for two reasons. First, it is recognized as one of the best algorithms tested on HDR images that has been published so far [32]. Second, although the initial approach is different, the actual treatment of pixels is comparable to that of our method. Fast bilateral filtering is based on an alternative of anisotropic diffusion to enhance boundaries while smoothing non-significant intensity variations. The new pixel values are computed by weighting surrounding pixels as a function of their spatial position as well as their intensity difference.

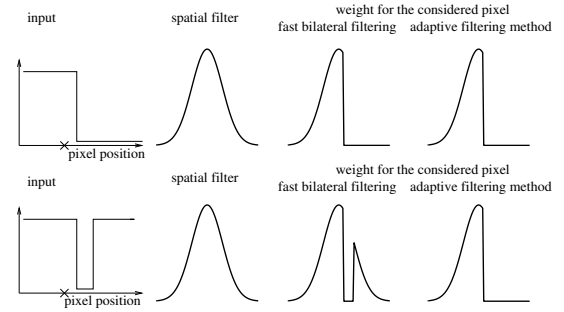


Fig. 6. In the case of fast bilateral filtering, the weight of one pixel is decreased if its intensity is different from that of the intensity of the treated pixel. However, the weight increases again if the intensity of the next pixel is similar to the currently treated pixel. With our method, as soon as a high contrast edge is detected, the weight of the current pixel is decreased as well as that of pixels located after the edge.

Our method computes the new pixel values by weighting surrounding pixels as a function of their spatial position and their spatial relation to high contrast edges. Our method gives more importance to spatial information. Figure 6 illustrates this difference. In the case of fast bilateral filtering, the weight of one pixel is decreased if its intensity is different from that of the intensity of the treated pixel. However, the weight increases again if the intensity of the next pixel is similar to the currently treated pixel. With our method, as soon as a high contrast edge is detected, the weight of the current pixel is decreased as well as that of pixels located after the edge.

Figure 11 compares the images obtained with the two methods. We observe that Durand and Dorsey’s method is better at increasing the local contrast in bright areas while our method provides a better result in dim areas. The head of the person is rendered better by our algorithm. This is due to the adaptive shape of the filter that prevents the sky to influence the color of the face and thus avoids the usual backlight effect. The way the color is rendered also influences the judgment of images. Fast bilateral filtering algorithm renders images that are more saturated than our method. It is suitable for some images but lead to unnatural impression in other cases, such as the reddish skin in Figure 11.

### D. Image acquisition and results

Producing an HDR image that is an accurate representation of the scene radiances is no longer an issue. Recently developed methods allow any capturing devices to virtually produce HDR images using a multiple exposure technique [42], [43], or by simultaneously sampling the spatial and exposure dimensions of image radiances [44]. HDR images are usually represented in floating point. They are stored in a special format called rgbe [45]. Our algorithm takes rgbe images as input.

For this article, we use radiance maps obtained with multiple exposure techniques [42], [43] as well as raw images taken with a Canon EOS30 and a Canon Powershot G2 digital camera. We assume that input images use sRGB primaries [46]. No color transformation is applied prior to processing. For the images that were generated by the multiple exposure

<sup>1</sup>PhotoFlair was developed by TruView Imaging Company (<http://trueview.com>)

technique [42], there is no guarantee that the images are in sRGB color space. As many of them are used in the literature, we still use them for comparison and assume they are encoded in sRGB. The output of our algorithm are 24 bits/pixel images rendered for standard displays, i.e. the color image encoding is sRGB [46].

Our method provides a solution for rendering HDR images using a spatial method. By combining spatial filtering with a segmentation to detect high contrast boundaries, we provide a way to reduce halo artifacts. Moreover, the introduction of a PCA to compute the luminance and chrominance channels allows good color rendition. Figure 12 shows HDR images treated by our algorithm. The high resolution images and the code are available for download on our web page [47].

## VII. CONCLUSION

The problem of rendering HDR images has been widely studied and a large number of methods exists. Although they enhance the quality of rendered images, these methods still suffer from some problems. Common drawbacks are the apparition of halo artifacts when increasing the local contrast, graying-out of low contrast areas and bad color rendition.

We provide a method to render HDR images taking inspiration from the Retinex model of color vision. In particular, our method is based on surround-based Retinex but uses an adaptive filter whose shape follows the high contrast edges of the image. In this way, the influence of a bright area on a neighboring dim area is decreased, thus preventing halo artifacts. We also include a sigmoid function that weighs the mask in order to prevent the graying-out of pure white or pure black low contrast areas. The Retinex-based adaptive filter is applied to the luminance channel only, which is defined by the first component of a PCA. Using PCA provides an image-dependent color-space transformation that guarantees orthogonality between channels. It minimizes the chromatic changes induced by the processing of luminance.

We tested our method on various HDR images, compared it with other algorithms and showed that it efficiently increases the local contrast while preventing halo artifacts and provides a good rendition of colors.

## ACKNOWLEDGMENT

We would like to thank Frédo Durand and Raanan Fattal to have made their images available for our comparison.

This work was supported by the Swiss National Science Foundation under grant number 21-101681.

## REFERENCES

- [1] D. Hood and M. Finkelstein, *Sensitivity to light*. In *Handbook of Perception and Human Performance*, K. Boff, L. Kaufman, and J. Thomas, Eds. New York: Sensory Processes and Perception, Wiley, 1986, vol. 1.
- [2] M. A. Webster, "Human colour perception and its adaptation," *Network: Computation in Neural Systems*, vol. 7, no. 4, pp. 587–634, November 1996.
- [3] S. N. Pattanaik, J. A. Ferwerda, M. D. Fairchild, and D. P. Greenberg, "A multiscale model of adaptation and spatial vision for realistic image display," in *SIGGRAPH 98 Annual Conference on Computer Graphics*, Orlando, July 1998, pp. 287–298.
- [4] L. Meylan and S. Süssstrunk, "Bio-inspired image enhancement for natural color images," in *IS&T/SPIE Electronic Imaging 2004. The Human Vision and Electronic Imaging IX*, vol. 5292, San Jose, CA, January 2004, pp. 46–56.
- [5] E. H. Land, "An alternative technique for the computation of the designator in the Retinex theory of color vision," in *National Academy of Sciences of the United States of America*, vol. 83, no. 10, May 1986, pp. 3078–3080.
- [6] Z. Rahman, D. D. Jobson, and G. A. Woodell, "Retinex processing for automatic image enhancement," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 100–110, January 2004.
- [7] K. Devlin, "A review of tone reproduction techniques," Department of Computer Science, University of Bristol, Tech. Rep. CSTR-02-005, November 2002.
- [8] H. Spitzer and S. Semo, "Color constancy: a biological model and its application for still and video images," *Pattern Recognition*, vol. 35, pp. 1645–1659, 2002.
- [9] D. Alleysson and S. Süssstrunk, "On adaptive non-linearity for color discrimination and chromatic adaptation," in *IS&T First European Conference on Color in Graphics, Image, and Vision*, Poitiers, France, April 2-5 2002, pp. 190–195.
- [10] E. Land, "The Retinex," *American Scientist*, vol. 52, no. 2, pp. 247–264, June 1964.
- [11] E. Land and J. McCann, "Lightness and Retinex theory," *Journal of the Optical Society of America*, vol. 61, no. 1, pp. 1–11, January 1971.
- [12] B. K. P. Horn, "Determining lightness from an image," *Computer Graphics and Image Processing*, vol. 3, pp. 277–299, 1974.
- [13] A. Hurlbert, "Formal connections between lightness algorithms," *Journal of the Optical Society of America A*, vol. 3, pp. 1684–1692, 1986.
- [14] D. Brainard and B. Wandell, "Analysis of the Retinex theory of color vision," *Journal of the Optical Society of America A*, vol. 3, no. 10, pp. 1651–1661, October 1986.
- [15] A. Rizzi, C. Gatta, and D. Marini, "From Retinex to automatic color equalization issues in developing a new algorithm for unsupervised color equalization," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 15–28, January 2004.
- [16] B. Funt, F. Ciurea, and J. McCann, "Retinex in matlab," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 48–57, January 2004.
- [17] F. Ciurea and B. Funt, "Tuning Retinex parameters," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 58–64, January 2004.
- [18] R. Sobol, "Improving the Retinex algorithm for rendering wide dynamic range photographs," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 65–74, January 2004.
- [19] D. Jobson, Z. Rahman, and G. Woodell, "A multiscale Retinex for bridging the gap between color images and the human observation of scenes," *IEEE Transactions on Image Processing*, vol. 6, no. 7, pp. 965–976, July 1997.
- [20] K. Barnard and B. Funt, "Investigations into multi-scale Retinex," in *Colour Imaging in Multimedia '98*, Derby, UK, March 1998, pp. 9–17.
- [21] M. D. Fairchild and G. M. Johnson, "ICAM framework for image appearance differences and quality," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 126–138, January 2004.
- [22] R. Kimmel, M. Elad, D. Shaked, R. Keshet, and I. Sobel, "A variational framework for Retinex," *International Journal of Computer Vision*, vol. 52, no. 1, pp. 7–23, 2003.
- [23] M. Elad, R. Kimmel, D. Shaked, and R. Keshet, "Reduced complexity Retinex algorithm via the variational approach," *Journal of visual communication and image representation*, vol. 14, no. 1, pp. 369–388, 2003.
- [24] G. West and M. Brill, "Necessary and sufficient conditions for von Kries chromatic adaptation to give color constancy," *Journal of Mathematical Biology*, vol. 15, pp. 249–258, 1982.
- [25] R. Kimmel, D. Shaked, M. Elad, and I. Sobel, "Space-dependent color gamut mapping: a variational approach," *IEEE Transactions on Image Processing*, vol. 14, no. 6, pp. 796–803, June 2005.
- [26] J. Tumblin and G. Turk, "LCIS: A boundary hierarchy for detail-preserving contrast reduction," in *SIGGRAPH 99 Annual Conference on Computer Graphics*, Los Angeles, CA, August 1999, pp. 83–90.
- [27] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," in *SIGGRAPH 02 Annual Conference on Computer Graphics*, 2002, pp. 249–256.
- [28] E. Reinhard, M. Stark, P. Shirley, and J. Ferwerda, "Photographic tone reproduction for digital images," in *SIGGRAPH 02 Annual Conference on Computer Graphics*, Los Angeles, CA, July 2002, pp. 267–276.
- [29] M. Ashikhmin, "A tone mapping algorithm for high contrast images," in *EUROGRAPHICS 2002*, P. Debevec and S. Gibson, Eds., Pisa, Italy, June 2002, pp. 1–11.

- [30] J. M. DiCarlo and B. Wandell, "Rendering high dynamic images," in *IS&T/SPIE Electronic Imaging 2000. Image Sensors*, vol. 3965, San Jose, CA, January 2000, pp. 392–401.
- [31] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," in *SIGGRAPH 02 Annual Conference on Computer Graphics*, Los Angeles, CA, July 2002, pp. 257–266.
- [32] J. Kuang, H. Yamaguchi, G. M. Johnson, and M. D. Fairchild, "Testing HDR image rendering algorithm," in *IS&T/SID Twelfth Color Imaging Conference: Color Science, Systems, and Application*, Scottsdale, AR, November 2004, pp. 315–320.
- [33] B. Funt, K. Barnard, M. Brockington, and V. Cardei, "Luminance-based multi-scale Retinex," in *AIC Colour 97 Eighth Congress of the International Colour Association*, vol. I, Kyoto, Japan, May 1997, pp. 330–333.
- [34] Y. Monobe, H. Yamashita, T. Kurosawa, and H. Kotera, "Fadeless image projection preserving local contrast under ambient light," in *IS&T/SID Twelfth Color Imaging Conference: Color Science, Systems, and Application*, Scottsdale, AR, November 2004, pp. 130–135.
- [35] C. C. Yang and J. J. Rodriguez, "Efficient luminance and saturation processing techniques for color images," *Journal of visual communication and image representation*, vol. 8, no. 3, pp. 263–277, September 1997.
- [36] C. Poynton, *Digital video and HDTV, algorithms and interfaces*. San Francisco, California: Morgan Kaufmann publishers, 2003.
- [37] J. Canny, "A computational approach to edge detection," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 8, no. 6, pp. 679–698, 1986.
- [38] B. Wandell, *Foundations of Vision*. Sunderland Massachusetts: Sinauer Associates, Inc, 1995.
- [39] P. K. Kaiser, *The Joy of Visual Perception: A Web Book*, 1996.
- [40] G. Buchsbaum and A. Gottschalk, "Trichromacy, opponent colours coding and optimum colour information transmission in the retina," in *Royal Society of London, Series B, biological Sciences*, vol. 220, no. 1218, November 1983, pp. 89–113.
- [41] N. Moroney, "Method and system of local color correction using background luminance mask," U.S. Patent 6,741,753, May 25, 2004.
- [42] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *SIGGRAPH 97 Annual Conference on Computer Graphics*, Los Angeles, CA, August 1997, pp. 369–378.
- [43] F. Xiao, J. DiCarlo, P. Catrysse, and B. Wandell, "High dynamic range imaging of natural scenes," in *S&T/SID Tenth Color Imaging Conference: Color Science, Systems, and Application*, Scottsdale, AZ, November 2002, pp. 337–442.
- [44] S. K. Nayar and T. Mitsunaga, "High dynamic range imaging: Spatially varying pixel exposures," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2000)*, vol. 1, Head Island, South Carolina, June 2000, pp. 472–479.
- [45] G. Ward, "High dynamic range image encodings," [www.anywhere.com/gward/hdrenc/hdr\\_encodings.html](http://www.anywhere.com/gward/hdrenc/hdr_encodings.html).
- [46] "ISO 22028-1:2004 photography and graphic technology – extended colour encodings for digital image storage, manipulation and interchange – part 1: Architecture and requirements."
- [47] "Miscellaneous," [ivrgwww.epfl.ch/index.php?name=Miscellaneous](http://ivrgwww.epfl.ch/index.php?name=Miscellaneous).



**Sabine Süssstrunk** (M'02) is Professor for Images and Visual Representation in the School of Information and Communication Sciences (I&C) at the Ecole Polytechnique Fédérale de Lausanne (EPFL), Switzerland. She received the B.S. degree in Scientific Photography from the Swiss Federal Institute of Technology (ETHZ), Switzerland, the M.S. degree in Graphic Arts Publishing from the Rochester Institute of Technology, Rochester, USA, and the Ph.D. degree from the School of Computing Sciences at the University of East Anglia, Norwich, UK. Her main research areas are digital photography, color image processing, image quality metrics, and digital archiving. From 2003-2004, she was a Visiting Scholar at Hewlett-Packard Laboratories in Palo Alto, USA. Prior to EPFL, she was Principle Imaging Researcher at Corbis Corporation, Seattle, USA, and Assistant Professor in the School of Photographic Arts and Sciences at the Rochester Institute of Technology. Prof. Süssstrunk is a member of IS&T, OSA, and ACM.



**Laurence Meylan** received the M.S. degree in computer sciences from the Ecole Polytechnique Fédérale de Lausanne (EPFL, Switzerland) in 2002. Since then, she has been a PhD candidate in the Laboratory of Audiovisual Communications in Lausanne (LCAV-EPFL), Switzerland. Her work has focused on the problem of high dynamic range imaging for both the creation and the rendering of high dynamic range images. In particular, she studied the color and luminance rendition of HDR images in the presence of local tone mapping algorithms.



Fig. 7. Difference between using a PCA and a YUV transform to compute the luminance. The image computed using YUV looks slightly green. **Left:** Image computed using PCA. **Right:** Image computed using YUV.



Fig. 8. The adaptive filter method allows to preserve detail visibility even along high contrast edges. **Left:** Non-adaptive filter method. **Right:** Adaptive filter method.



Fig. 9. **Left:** Image treated with MSRCR. **Right:** Image treated with the adaptive filter method.



Fig. 10. **Left:** Image treated with Fattal's gradient attenuation method. **Right:** Image treated with the adaptive filter method.



Fig. 11. **Top:** Gamma-encoded image. **Middle:** Image treated with the adaptive filter method. **Bottom:** Image treated with the fast bilateral filtering method.





Fig. 12. Results of the Retinex-based adaptive filter method. **Left:** Gamma-encoded image. **Right:** Image treated with our method.