Flicker Reduction in Tone Mapped High Dynamic Range Video

Benjamin Guthier, Stephan Kopf, Marc Eble, Wolfgang Effelsberg

Department of Computer Science IV, University of Mannheim, Germany
{guthier, kopf, effelsberg}@informatik.uni-mannheim.de
meble@mail.uni-mannheim.de

ABSTRACT

In order to display a high dynamic range (HDR) video on a regular low dynamic range (LDR) screen, it needs to be tone mapped. A great number of tone mapping (TM) operators exist – most of them designed to tone map one image at a time. Using them on each frame of an HDR video individually leads to flicker in the resulting sequence. In our work, we analyze three tone mapping operators with respect to flicker. We propose a criterion for the automatic detection of image flicker by analyzing the log average pixel brightness of the tone mapped frame. Flicker is detected if the difference between the averages of two consecutive frames is larger than a threshold derived from Stevens' power law. Fine-tuning of the threshold is done in a subjective study. Additionally, we propose a generic method to reduce flicker as a post processing step. It is applicable to all tone mapping operators. We begin by tone mapping a frame with the chosen operator. If the flicker detection reports a visible variation in the frame's brightness, its brightness is adjusted. As a result, the brightness variation is smoothed over several frames, becoming less disturbing.

Keywords: HDR Video, Tone Mapping Operator, Flicker Detection, Flicker Reduction

1. INTRODUCTION

The term *dynamic range* is often used to describe the ratio between the brightest and darkest point in a given scene. Analogously, it describes the ratio between the highest and lowest luminance level that can still be captured faithfully by a camera. Being a ratio, the dynamic range is unitless*. As an example, the average luminance during a starless night is in the order of magnitude of 10^{-3}, and is 10^5 in the midday sun. This corresponds to an approximate dynamic range of 10^8 in nature. In a single scene at a single point in time, the dynamic range is usually much lower than that, but can still exceed the capabilities of traditional LDR capturing and display devices. This leads to under- and overexposed pixels in the captured images, and information on brightness differences between these pixels is lost.

There exist a number of techniques to capture high dynamic range scenes with regular imaging hardware. In order to display the created HDR image or video, its range of luminance values must be compressed to match the displayable range of the output device. This range compression step is called *tone mapping*.

A tone mapping operator typically considers the minimum, maximum or average luminance of a recorded HDR image and scales it down accordingly. However, most operators are designed with still images in mind. That is, the dynamic range of each frame of an HDR video sequence is scaled independently. In situations where the minimum, maximum or average brightness of a frame differs greatly from its predecessor, the scaling function changes rapidly from one frame to the next, leading to visible image flicker. Such a situation can easily arise when a bright object, such as the headlight of a car or a specular reflection, enters the field of view. The tone mapper attempts to map the greatly increased dynamic range onto the same output range, often changing the brightness of the entire image in the process.

In this paper, we alleviate the described problem in a way that is applicable to all still image tone mappers. In order to be independent of the tone mapper used, we remove image flicker as a post processing step. Essentially, the per-frame brightness difference (flicker) is smoothed over a number of frames to become less obtrusive. We first introduce a criterion to detect flickering frames. It is based on the difference of the average image brightness of two consecutive frames and on a threshold derived from Stevens' power law. Our assumptions are that:

*Although it is sometimes measured as a binary logarithmic value and denoted by *stops*.

- Image flicker is the most obtrusive artifact introduced by applying still image tone mappers to videos, and
- flicker can be detected sufficiently well by analyzing the average brightness of a tone mapped frame.

These assumptions are supported by our experimental results. To reduce the visibility of detected flicker, we adjust the average image brightness after tone mapping. This is done using image normalization and clamping, which is included as a last processing step in many tone mapping operators anyway. A frame that was mapped to a much darker average than its predecessor will be adjusted to a level closer, but still darker than the previous frame. We always keep the brightness variation within a range that is tolerable according to our flicker detection. After a few frames of convergence, the same average brightness the operator would maintain without our intervention is reached again.

The rest of this paper is structured as follows. Section 2 presents existing HDR capturing techniques and previous work in the field of tone mapping for still images and video. In Section 3, we present our method to detect flickering frames. This criterion is then used in Section 4 to reduce the visibility of image flicker. A description of our experiments to fine-tune the parameters of flicker detection in a subjective user study, the results of flicker reduction, and some processing cost considerations are then given in Section 5.

2. RELATED WORK

2.1 HDR Capturing Techniques

The creation of HDR still images using LDR hardware has been discussed in the literature. The most popular approach is temporal exposure bracketing, i.e., using a set of LDR images captured in quick sequence at different exposure settings. Each LDR image then captures one facet of the scene's dynamic range. When fused together, an HDR image is created that covers the full dynamic range of the scene. Most works in this field focus on the estimation of the inverse camera response function to map pixel values onto scene radiance,[1–4] or on image registration to compensate the camera motion between the shots.[5–8] Additionally, there exist a number of works that describe the capturing of HDR images by specialized hardware.[9–13]

More recently, these still image approaches have been extended to capturing HDR video. While the earlier works separated the capturing step from the offline HDR processing,[14] more and more papers are being published that aim at capturing HDR video in real-time.[15,16]

In our work, we use the real-time capable HDR capturing method proposed in[15] and save the created videos. All the processing described in this paper is then applied to saved videos.

2.2 Still Image Tone Mapping Operators

A variety of TM operators for still images already exists, and several works have been published to evaluate their quality.[17–20] There is a distinction between spatially invariant, global operators and spatially variant, local operators.[21]

Global operators are non-linear functions based on the content of an image as a whole, using statistical values such as average luminance to estimate optimal mapping parameters for a particular image. When the optimal function is found, the same transformation is applied to each pixel. Psychophysical models of brightness and contrast perception as well as retinal properties serve as a basis for most approaches.[22–25] These operators are simple and fast, but are limited in their ability to process very high dynamic ranges. A mapping of the full input range of luminance can be achieved by more engineering-oriented models such as logarithmic contrast compression[26] or histogram adjustment methods.[27]

Local operators consider a set of neighboring pixels for estimation of the parameters of a transformation function. Each pixel of an image is mapped differently, based on the local features of its neighborhood. Because the human visual system is sensitive to local contrast, high quality images spanning high dynamic ranges are possible with these methods. Due to the more complex nature of these operators, computing time increases and artifacts such as halo effects can occur.

Local operators can be divided into different classes. Center-surround methods increase local contrast by computing the difference between a pixel's value and a weighted set of its neighboring pixels. This procedure is inspired by receptor properties of the human visual system.[28–31]

Frequency-based operators separate the low and high frequency bands of an image. Assuming that an image is a product of light intensity and reflection, only the part relevant to light intensity is compressed. This is the low frequency band, while the image details contained in the high frequency bands are kept. Oppenheim et al.[32] published this technique known as the first TM operator, which attenuates low frequencies more than high frequencies.

Gradient-based operators alter the gradient of an image. High frequency regions cause significant differences between neighboring pixels, while low frequency regions contain rather small differences. Fattal et al.[33] observed, that drastic luminance changes greatly increase gradients. Thus the gradient field is compressed progressively.

For this work we chose three different TM operators. As a global operator the *Contrast-Based Scale Factor* by Ward[25] is used. The primary goal of this method is the preservation of contrast. A constant of proportionality between display luminance and world luminance has to be found, that yields a result with roughly the same contrast visibility as the actual scene. Calculation of the scale factor is based on the research of Blackwell,[34] who defined a relationship between adaptation to luminance changes and the just noticeable difference. It is then applied to pixel values to convert real luminance values into display values.

The *Photographic Operator* published by Reinhard et al.[28] is chosen from the set of local tone mappers. It is inspired by photographic development and printing techniques. First, a global tone mapping reduces the range of luminance to a displayable range. A center-surround function then finds low contrast regions. Their contrast and details are enhanced by analyzing the luminance of local regions and changing the contrast of a pixel related to its neighbors by dodging and burning.

The third operator considered is the *Histogram Adjustment* method proposed by Ward et al.[27] It globally applies a monotonic tone reproduction curve to all pixels. The idea is to allocate most of the displayable dynamic range to luminance ranges that are represented by many pixels. Thus pixels in less frequent brightness levels are compressed more strongly. Additionally, human visual limitations such as glare or visual acuity are regarded in further processing steps.

2.3 Tone Mapping Operators for Video

Only few TM operators specific to HDR video content have been proposed. Tone mappers encounter problems when dealing with strong illumination changes in adjacent frames, as current operators transform each frame independently.

Pattanaik et al.[35] presented an approach that models the time-dependent local adaptation of the human visual system and discussed temporal coherence to avoid flicker introduced by tone mapping.

Ferwerda[24] offers a time-dependent model, which is based on psychophysical experiments. However, the data used is not in accordance with psychophysical data, and the problems of local retina adaptation and eye movements influencing the state of adaptation persists.

Benoit et al.[36] propose a model based on properties of the human retina. HDR video content is enhanced by a non-separable spatio-temporal filter with added temporal constancy. This is done by imitating the retina's luminance compression and additional temporal information processing.

Another approach inspired by the non-linear adaptation of the retina by Meylan et al.[37] performs color rendering operations, such as tone mapping, directly on a color filter array (CFA) of a camera sensor. Their method works similarly to retinal processing of the human visual system. In traditional color processing, color reconstruction is done before color rendering to obtain the full resolution color image with three chromatic components per pixel, as a CFA retains only one chromatic component for each pixel.

An implementation of a TM operator using programmable graphics hardware was presented by Goodnight et al.[38] They extend the TM operator to include a time-dependent model, that is based on the adaptation model proposed by Durand and Dorsey.[39] It describes a multi-pass interactive rendering that computes the average luminance in a first pass and renders the scene with a TM operator in the second pass. Both multiplicative and subtractive light adaptation is simulated by applying a global multiplicative scale factor during the mapping from world luminance to display luminance.

Wang et al.[40] presented a modified version of the Photographic Operator for real-time operation. To achieve fast computation, only the global component of the operator is considered and pipelined with a video codec.

An approach using gradient domain tone mapping in HDR videos was introduced by Lee et al.[41] Temporal motion information is incorporated into the Poisson equation to obtain temporally coherent LDR videos.

A general model for temporal luminance adaptation was proposed by Krawczyk et al.[42] In accordance with the human visual system, that reacts to temporal changes in luminance conditions, a time constant for the speed of the adaptation is introduced. It is different for rods and for cones, so that adaptation speed depends on the level of scene illumination.

All methods for videos listed here were either implemented as a tone mapping operator, or specific to a certain operator, while our proposed method can be used in conjunction with any operator.

2.4 TM Quality Evaluation

Evaluation of TM operators is widely done in subjective experiments.[43, 44] Since there is no standardized method for tone mapping, our evaluations are performed in accordance with the Subjective Assessment Methodology for Video Quality (SAMVIQ).[44] It is a subjective video quality measurement for motion pictures generally used to evaluate video encoding techniques.

3. FLICKER DETECTION

It is our goal to perform tone mapping on high dynamic range videos using standard operators designed for still images. When doing so, temporal incoherence of the minimum, maximum or average scene luminance leads to image flicker in the tone mapped result. We argue that image flicker is the most disturbing artifact introduced in this process. This is supported by two experiments described in Section 5. We thus focus entirely on the detection and reduction of flicker.

The second assumption we make is that flicker is sufficiently well detected by computing the average image brightness of a tone mapped frame and comparing it to the average of the previous frame. The validity of this simple criterion is also backed up by our experiments described later in the paper.

Figure 1 shows the average brightness of the frames of a tone mapped video over time. It illustrates well the rapid drop and the rapid increase in average brightness between frames 52 and 53 and between frames 90 and 91. The respective frames are shown in Figure 2. They are taken from our HDR video *Turn*, which contains a camera turn from a dark indoor area towards a window showing a light outdoor scene. The discontinuities in Figure 1 occur exactly when the bright window first enters the camera's field of view or leaves it respectively.

To compute the average image brightness, we use the geometric mean given by

$$\tilde{I} = \prod_{j=1}^{n} (I(j) + \epsilon)^{1/n}, \tag{1}$$

where n is the number of pixels in the image and $I(j)$ is the value of the pixel at position j. ϵ is a small number to prevent the product from becoming zero for black pixels. This equation can be rewritten as

$$\tilde{I} = exp\left(\frac{1}{n}\sum_{j=1}^{n} log(I(j) + \epsilon)\right). \tag{2}$$

Over its arithmetic counterpart, the geometric mean has the advantage of being more resilient to outliers. Furthermore it more closely resembles the way average image brightness is perceived by the human eye. For this reason it is often used in tone mapping.

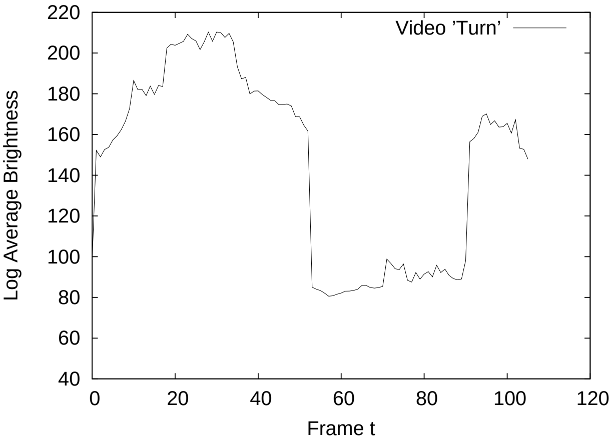

Figure 1. Log average pixel value of an HDR video tone mapped with the Photographic Operator. From frame 53 to 90, a window showing the light outside is visible. This leads to a large increase of the scene's dynamic range and a large drop in brightness of the tone mapped result.

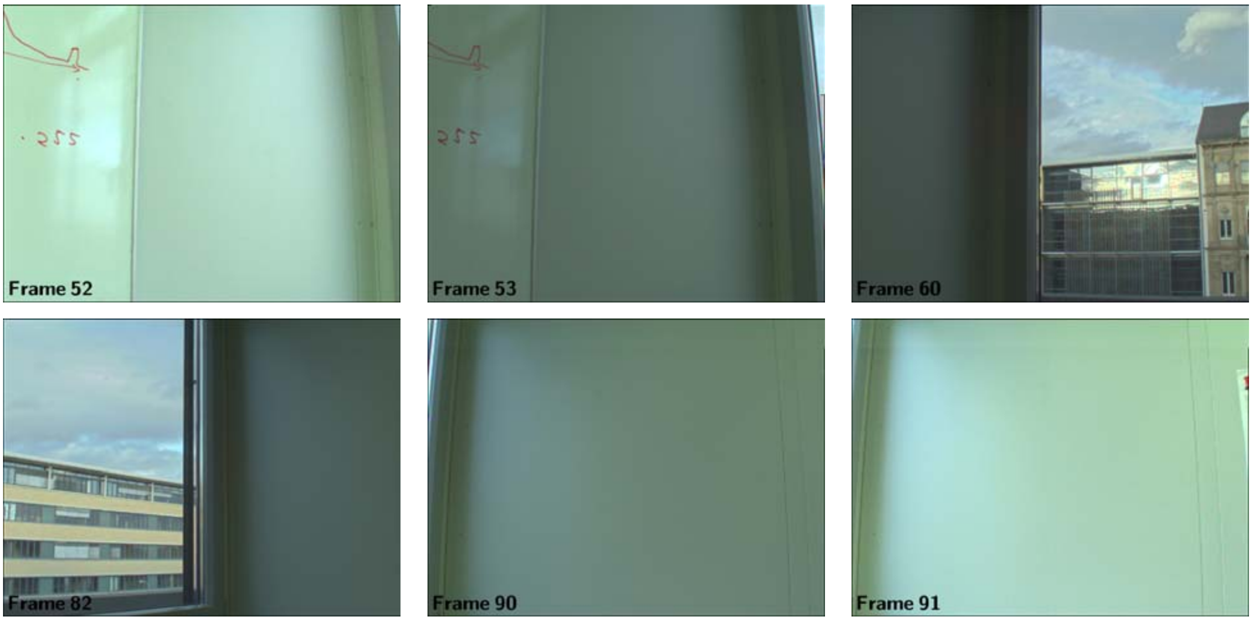

Figure 2. Six frames of the video *Turn* tone mapped with the Photographic Operator. As soon as the window enters the camera's field of view, the scene's minimum and maximum luminance changes greatly, leading to a visible brightness difference of the tone mapped frames. This can be seen between frames 52 and 53 and frames 90 and 91.

3.1 Calculating the Brightness Threshold

The biggest challenge in developing a flicker detection criterion using the log average pixel value is finding a suitable threshold for the difference of the averages of two consecutive frames. We experimented with two different models found in the literature on the human visual system. They are Weber's law[45] and Stevens' power law.[46] Both use the notion of a *just noticeable difference* ΔR, which depends on a given background luminance R, and both introduce an adjustable parameter k. For a given luminance level (in our case the log average of the previous frame), these models allow the computation of a maximum luminance change that will remain unnoticed to a human observer. Even though the setting for which these laws were developed slightly differs from ours, they serve as a perceptional basis for our criterion.

Weber's law is a mathematical model for the human visual perception.[45] It states that the ratio between the minimum incremental amount of luminance required to be perceptible and the background luminance is a constant:

$$k = \frac{\Delta R}{R} \tag{3}$$

In other words, the just noticeable difference is a fixed fraction of the predominant brightness. This equation can be solved for ΔR to yield the desired threshold.

Our second criterion is Stevens' power law.[47] It describes the relationship between the real magnitude of a general stimulus and the magnitude as perceived by a human. A number of TM operators use the power law with the stimulus being the sensation of brightness.[22,23] In its general form, it is given by

$$\Delta R = kR^{\alpha}, \tag{4}$$

where α is a constant specific to the type of stimulus considered. For brightness $\alpha \approx 0.33$.

A suitable value for the parameter k is determined experimentally for each criterion. This is presented in Section 5. In our experiments, we also found that Stevens' power law allows for the most accurate detection of flickering frames. It is therefore our criterion of choice.

4. FLICKER REDUCTION

In this section we demonstrate our flicker reduction algorithm, which makes use of the flicker detection introduced in the previous section. It will be seen that a robust detection makes flicker removal straightforward. If flicker occurs in a frame, we iteratively adjust its brightness until it is within the tolerable threshold. To illustrate our algorithm, we use frames 52 and 53 of the test video *Turn* as shown in Figures 1 and 2 as an example throughout this section: A part of the window showing the very light outside suddenly moves into the camera's field of view, making the maximum scene luminance go up rapidly. The TM operator maps the now increased dynamic range of the scene onto the same display range, resulting in an overall much darker image. The case where the image brightness *increases* rapidly is handled analogously and not explicitly described here.

The algorithm is implemented as a post-processing step and works with any tone mapper. We tested our approach with Ward's Contrast-Based Scale Factor,[25] the Histogram Adjustment by Ward et al.[27] and the Photographic Tone Reproduction by Reinhard et al.[28]

We start by tone mapping the current frame t with the chosen operator and settings. Next, the log average pixel value $\tilde{I}_t$ of the frame is computed using Equation 2. Then we calculate the maximum allowable brightness difference ΔR to the previous frame using Stevens' power law (Equation 4):

$$\Delta R = k(\tilde{I}_{t-1})^{0.33}, \tag{5}$$

where $\tilde{I}_{t-1}$ is the log average of the previous frame. Now we check whether $|\tilde{I}_{t-1} - \tilde{I}_t| > \Delta R$. If it is not, then the frame is likely not to be a flickering frame (see experimental results). In our example however, it is assumed that the current frame is much darker than the previous one ($\tilde{I}_{t-1} > \tilde{I}_t$). The goal is now to increase the frame's brightness so that it falls within a tolerable range. The lower end of this range is given by $\tilde{I}_{t-1} - \Delta R$, meaning that there shall be no detectable flicker. It is also desirable to maintain the original brightness produced by the TM operator as well as possible. After adjustment, $\tilde{I}_t$ should therefore be close to the lower end of the range. To accommodate this fact, we set the upper bound to $\tilde{I}_{t-1} - p\Delta R$, where p is a percentage we set to 50% in our implementation. The next step is to iteratively adjust the frame's brightness, producing a sequence $\tilde{I}_t^0, \tilde{I}_t^1, ..., \tilde{I}_t^i$, until it falls into the desired range of $[\tilde{I}_{t-1} - \Delta R, \tilde{I}_{t-1} - p\Delta R]$. As an explicit target value I^*, we aim for the range's center. The process of iteratively approaching the desired brightness is depicted in Figure 3.

4.1 Iterative Brightness Adjustment

Most TM operators perform a final normalization and clamping of the resulting floating point values. The tone mapper may produce scaled luminance values in an arbitrary range. In the normalization step, these values are shifted to begin from zero and are usually scaled to 255. Afterwards, the values may be clamped to $[0, 255]$ again for robustness against imprecision in the floating point operations. We modify these steps by introducing a scale parameter s. The values after tone mapping are then normalized to the range of $[0, 255s]$ instead. The

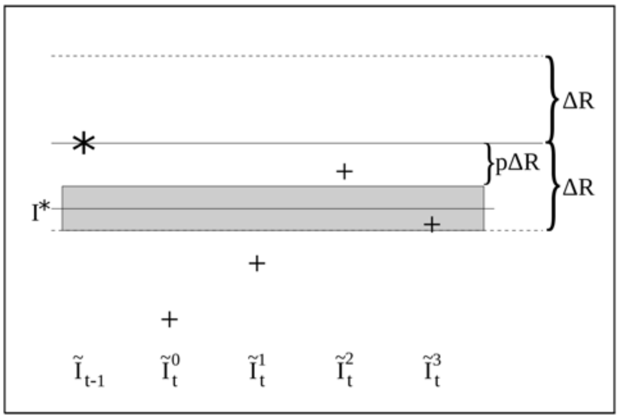

Figure 3. Frame t is too dark after tone mapping ($\tilde{I}_t^0 < \tilde{I}_{t-1} - \Delta R$). Its brightness is thus iteratively adjusted towards the target value I^*. After three iterations, it falls within the tolerable brightness range drawn in gray.

subsequent clamping to a byte remains unchanged. In doing so, the change of the average output brightness will be approximately linear to the change of s (only approximately because of shifting and clamping). That is, for a certain range of scale changes, there is a slope parameter m^0 for which

$$\tilde{I}_t^0 - \tilde{I}_t^1 = m^0(s^0 - s^1) \tag{6}$$

holds. In the following, we use the superscript to denote the iteration index in our iterative brightness adjustment scheme. $\tilde{I}_t^0$ is therefore the log average pixel value of frame I_t as determined earlier (called $\tilde{I}_t$ above). $s^0 = 1.0$ is the original scale used for normalization. $\tilde{I}_t^1$ and s^1 are the respective values after adjusting the scale once. The initial slope m^0 is set to an arbitrary value of 50. It is refined during the iterations and re-used for later frames. In the following, we omit the time parameter t to avoid clutter.

At this point, we know the frame's original average brightness $\tilde{I}^0$, the desired target brightness I^* to which we would like to adjust it, the original scale s^0 and a rough estimate of the coefficient m^0 to relate scale change to brightness change. From this, we can compute a new scale s^1 that maps the brightness closer to the target by rearranging Equation 6 and replacing $\tilde{I}_t^1$ with I^*:

$$s^1 = s^0 - \frac{I^0 - I^*}{m^0}. \tag{7}$$

The normalization step as described before is then re-done using the adjusted scale parameter s^1. This results in a new log average image brightness of $\tilde{I}^1$ in the first iteration. We now know that adjusting the scale from s^0 to s^1 changed the average from $\tilde{I}^0$ to $\tilde{I}^1$. Inserting these values into Equation 6 yields a more accurate approximation of the slope parameter which we denote by m^1.

In each iteration i, the scale parameter s^i is updated (Equation 7), and the original tone mapping output is normalized and clamped using s^i, resulting in $\tilde{I}^i$. A more accurate slope m^i is then estimated from

$$m^i = \frac{\tilde{I}^i - I^0}{s^i - s^0}. \tag{8}$$

The iteration ends as soon as $\tilde{I}^i$ falls within the tolerable target range given above, in which case the normalized image is the output of the post processing step. The iterative brightness adjustment is illustrated in Figure 3. See the next section for the mean number of iterations required. The final slope m^i from the last iteration is saved and re-used as an initial guess for the next flickering frame.

The adjusted brightness $\tilde{I}_t^i$ of frame t is now slightly lower than $\tilde{I}_{t-1}$ (more specifically: $\tilde{I}_t^i \approx I^* < \tilde{I}_{t-1}$). However, the difference is now within a range we consider to be unobtrusive. For the next frame $t+1$, we set the normalization scale to 1.0 again, i.e., we tone map with standard parameters. If there is no more rapid scene histogram change between frames t and $t+1$, $\tilde{I}_{t+1}$ is now closer to $\tilde{I}_t$ than $\tilde{I}_t$ was to $\tilde{I}_{t-1}$ and the amount of adjustment required is smaller. After a few frames, the difference approaches a value less than ΔR and no further adjustment is necessary. Figure 4 illustrates this convergence towards standard parameters.

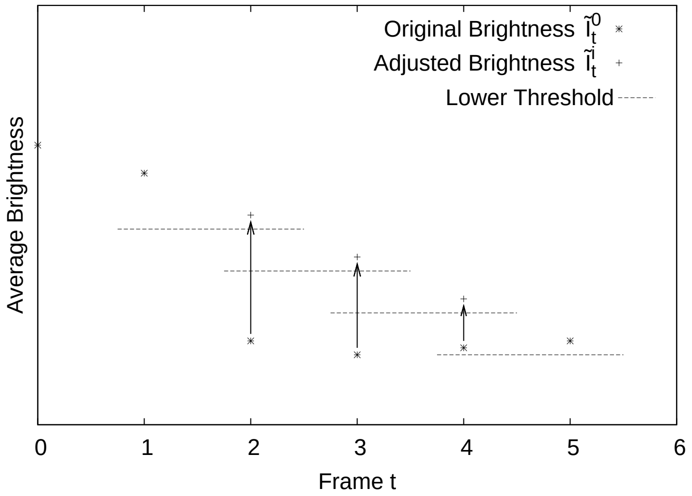

Figure 4. This plot shows a rapid decrease of average brightness between frames 1 and 2. Frame 2 is adjusted to reduce the brightness gap. The amount of adjustment needed decreases with each subsequent frame. In frame 5, the adjusted brightness has converged towards the value achieved by tone mapping with standard scale. No more adjustment is required.

5. EXPERIMENTAL RESULTS

5.1 Subjective Flicker Detection

Our experimental results are all based on two subjective user studies we performed. For both studies, we used the same five HDR videos, tone mapped without flicker reduction. The length of the video clips ranges from 14 to 44 seconds. Here are the names and a brief description of the videos:

Turn:	Mostly indoor scene with a window showing the bright outside. Smooth horizontal camera rotation with a sudden transition from dark to bright and back.
Handheld:	Handheld camera. Indoor with very dark areas and window. Random camera motion.
Outdoor:	Outdoor scene with a relatively narrow dynamic range. Horizontal and vertical camera rotation.
Outdoor South:	A street with moving cars and trees.
Linux Room:	Dark indoor scene with long exposure times and motion blur. Artificial light sources. In some frames, the window to the outside is visible.

Since different TM operators produce different flicker artifacts, we tone mapped these videos with the three operators: Ward's Contrast-Based Scale Factor[25] (referred to as *Scale* in the following), the Histogram Adjustment by Ward et al.[27] (*Histogram*) and the Photographic Tone Reproduction by Reinhard et al.[28] (*Photographic*). This resulted in 15 videos total. The URL where all our videos can be found is given in the *References* section.[48]

Both experiments included the same 10 test subjects. They had no prior experience with HDR video or tone mapping. The methodology of our experiments was based on SAMVIQ.[44] We only differ from the recommendation in that we do not show an explicit reference video. This was not possible due to the lack of an HDR display to show a non-tone-mapped reference video on.

In the first experiment, the subjects were told to mark the video frames in which they perceive a general distortion. They were not informed of our endeavor to remove flicker artifacts in particular. From the resulting list of marked frames, we removed three frames that were clearly due to problems in the capturing process and unrelated to tone mapping (e.g., motion blur or ghosting due to exposure bracketing). In the second experiment, we told the subjects to mark frames with visible image flicker. We observed that more frames were marked in the second experiment than in the first. Additionally, each frame that was marked in experiment 1 by anyone was marked by half or more of the users in experiment 2, i.e., the first results were a subset of the second. The fact that each reported distortion turned out to be classified as image flicker implies that flicker is the most obtrusive artifact introduced when still image tone mappers are applied to video.

To compile a final list of frames that are considered flickering, we used those frames that were marked by 50% or more of the subjects in the second experiment. This includes everything that was marked in the first one. Table 1 shows our list. Some clips exhibit a large brightness difference between frame 0 and 1. This is also an artifact of the capturing process, but was left in the list as it also constitutes flicker.

Video / Operator		**Reported Flickering Frames**
Turn	Scale	1, 53, 57, 71, 72, 90-92
	Photographic	1, 53, 91
	Histogram	19, 51-54, 91, 92
Handheld	Scale	12, 14, 34, 35
	Photographic	-
	Histogram	-
Outdoor	Scale	1, 13, 18
	Photographic	1, 13, 60
	Histogram	-
Outdoor South	Scale	6, 27, 28, 42, 54, 57
	Photographic	27, 28, 41, 42
	Histogram	-
Linux Room	Scale	10, 17, 50, 68
	Photographic	10, 18, 50, 78
	Histogram	14, 26, 78, 79

Table 1. List of flickering frames for each video tone mapped with each operator. We included all frames that were reported by 50% or more of the subjects in our second experiment. 50 frames were reported in total.

5.2 Setting the Parameter k

In Section 3, we introduced two criteria for the detection of flicker: Stevens' power law and Weber's law. Both had an adjustable parameter k to control their sensitivity to best match the subjective impression of flicker. We now describe how the marked frames shown in Table 1 are used to determine the optimal values for k and to judge the performance of the two criteria. For this purpose, a data set consisting of all 1311 frames of all video clips is created. For each frame, we compute the difference ΔR of its log average brightness and its predecessor's R. Knowing ΔR and R, solving Equations 3 and 4 for k yields a minimum sensitivity value for which the frame would just be detected as flickering. We thus also record k_{weber} and $k_{stevens}$ for each frame. Additionally, we mark the flickering frames according to Table 1. Figure 5 shows the number of frames having a certain value $k_{stevens}$ for the two classes of flicker and non-flicker frames.

For a given value of k, we can now count the number of correctly detected flickering frames (true positives, tp), frames incorrectly classified as flickering (false positives, fp), and missed frames (false negatives, fn). The metrics *precision* P and *recall* R are calculated as:[49]

$$P = \frac{tp}{tp + fp}, \tag{9}$$

$$R = \frac{tp}{tp + fn}. \tag{10}$$

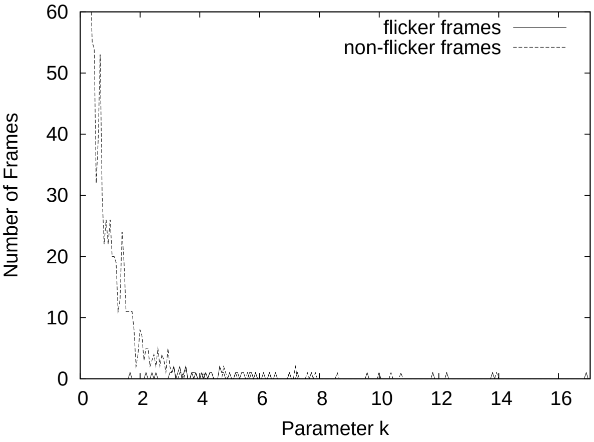

Figure 5. The histogram shows the number of flicker and non-flicker frames for each value of k for Stevens' power law.

Precision is defined as the fraction of correctly detected flicker frames among all detected frames, whereas recall is the fraction of correctly detected flicker frames among all actual flicker frames. Figure 6 plots precision and recall on our data set against the chosen value of k. A higher k generally increases the precision, but has a negative impact on recall.

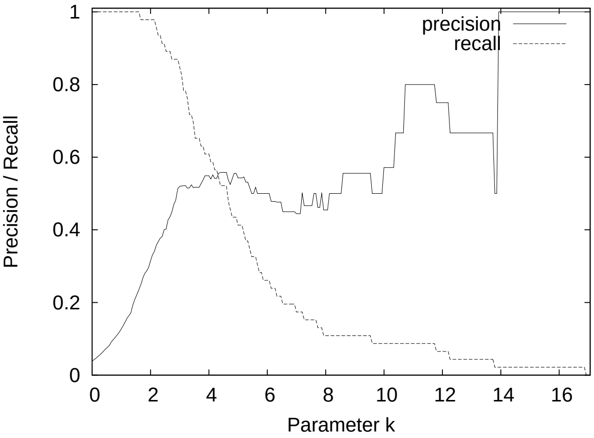

Figure 6. Precision and recall of our detector plotted against parameter $k_{stevens}$.

The F_β-score ($F_\beta \in [0, 1]$, a higher score is better) combines both precision and recall:

$$F_\beta = (1 + \beta^2) \cdot \frac{P \cdot R}{\beta^2 \cdot P + R}. \tag{11}$$

The parameter β expresses subjective preferences, e.g., whether the number of frames incorrectly classified as flickering is more relevant compared to flicker frames that are not detected (missed frames). In the case of $\beta = 1$, the F_1 score can be interpreted as the weighted average of precision and recall; $\beta = 2$ gives recall (missed frames) twice the weight of precision. Figure 7 shows the value of four different F-scores depending on the value of k. The global maximum of each F-score indicates the k which optimizes the subjective preferences. Higher β values lead to a lower k and reduce the number of missed frames at the expense of more false positives.

Our goal is to remove as much flicker as possible to generate high quality videos. Thus the number of missed flicker frames should be low. False positives are more acceptable, since additional tone mapping of a non-flicker frame merely increases the computational effort. Based on these considerations we chose the F_4-score as quality measurement criterion which assigns four times as much importance to recall as to precision.

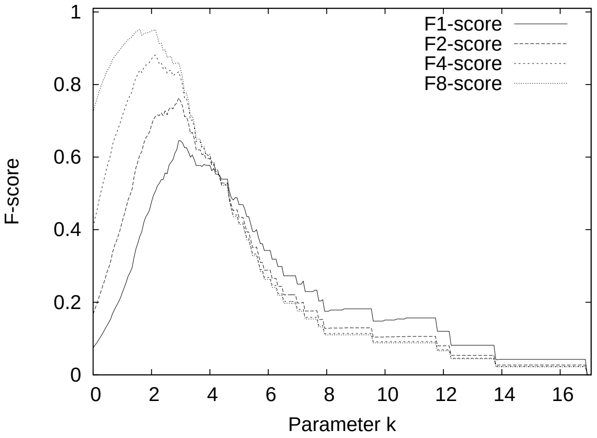

Figure 7. Comparison of different F-scores based on Stevens' power law.

To derive an optimal value for the parameter k, we consider the full data set. The optimal values for Stevens' power law and Weber's law are $k_{stevens} = 2.133$ and $k_{weber} = 0.0846$ respectively.

We use tenfold cross validation[50] to judge the performance of our flicker detection in practice. The idea is to partition the video frames into ten subsets: In one iteration, the maximum F_4 score (i.e., the optimal value of k) is calculated using nine training sets. The remaining subset is used to calculate a realistic F_4-score and to judge the performance of the detection. This is repeated ten times, such that all frames are used for validation exactly once. The average over all F_4-scores is used as the final score for the respective detection method. Table 2 summarizes the results.

Method	Avg. F_4-score	# false negatives	# true positives	# false positives
Stevens	0.8689	1	49	87
Weber	0.7774	2	48	149

Table 2. Comparison of cross-validation results.

It can be seen that the average F_4-score of Stevens' power law is higher, making it the better choice for our purpose. From the number of false negatives and false positives for our choice of k, we can conclude that a large difference of the log average pixel value between two frames is a necessary criterion for a flickering frame. That is, if our detector classifies a frame as non-flickering, it is very likely to actually be a non-flicker frame. This justifies our decision to adjust a frame's brightness until our detector stops reporting flicker. It is very likely that the flickering is now removed. In other words, we use flicker detection as an objective quality metric to prove the correctness of our flicker reduction algorithm.

Our results are presented as eight plots in Figure 8. They show the trend of the log average brightness before and after flicker reduction. We selected eight representatives out of the 15 videos we created. Corrected and original version of all 15 videos can be downloaded from the URL given in the *References* section.[48]

5.3 Computational Effort of Flicker Reduction

We assume that the log average brightness of a frame can be obtained from the TM operator as a by-product or calculated with little extra effort. The cost of flicker detection is thus negligible and so is the calculation of the new scale parameter in our iterative brightness adjustment scheme. Hence, the additional computational effort produced by our flicker reduction algorithm is mainly due to the repeated normalization of flickering frames. When processing the 1311 frames of our test videos, 274 additional normalizations were performed. They were caused by adjusting 208 flickering frames. This number differs from the 50 flickering frames in Table 1 because one flicker effect may have to be smoothed over an entire succession of frames, and we also process false positives. This gives an average number of 1.317 iterations per adjusted frame and 4.16 adjusted frames per subjective flicker artifact.

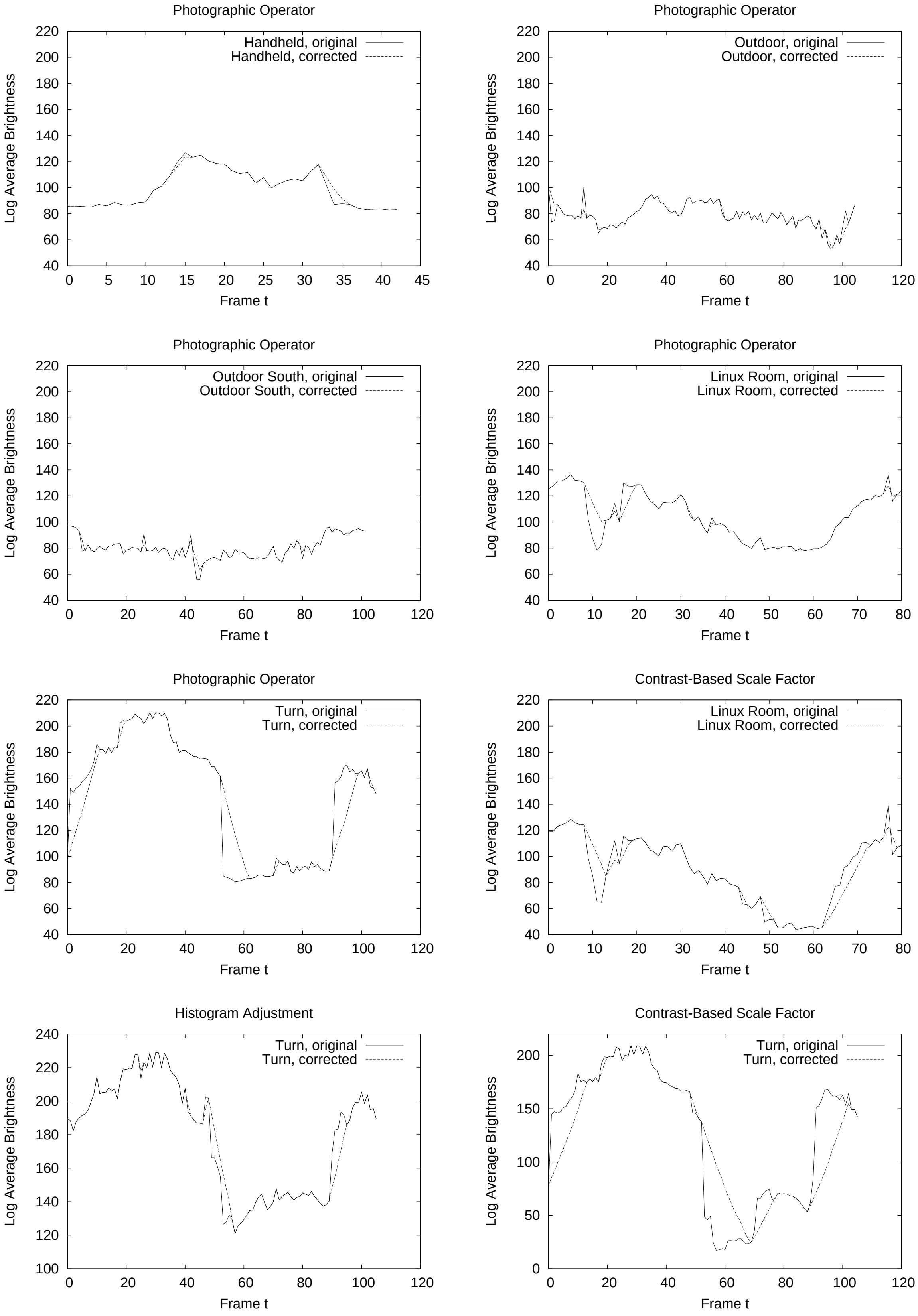

Figure 8. Average brightness before and after flicker reduction. We chose to show all five videos tone mapped with the Photographic Operator. Additionally, we show all three versions of *Turn*, since it has the most interesting properties.

6. CONCLUSIONS

We presented an algorithm for the automatic removal of flicker in tone mapped high dynamic range video. It is based on a threshold for the maximum allowable difference between the log average pixel value of two consecutive frames. When the threshold is exceeded, the frame's brightness is adjusted until it falls into a tolerable range. The flicker artifact is thus smoothed over several frames and becomes unobtrusive.

We evaluated our work by first fitting the threshold for flicker detection to the results of a subjective study. Flicker reduction adjusts the brightness until it is within the threshold. Assuming that we can detect flicker reliably, our flicker reduction algorithm is very likely to correctly remove flicker from the videos.

REFERENCES

[1] Debevec, P. E. and Malik, J., "Recovering high dynamic range radiance maps from photographs," in [*Proc. of the 24th Annual Conference on Computer Graphics and Interactive Techniques*], (1997).

[2] Mann, S. and Picard, R., "Being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures," in [*Proceedings of the IS&T 48th Annual Conference*], (1995).

[3] Mitsunaga, T. and Nayar, S. K., "Radiometric self calibration," in [*Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*], (1999).

[4] Robertson, M. A., Borman, S., and Stevenson, R. L., "Dynamic range improvement through multiple exposures," in [*Proceedings of the International Conference on Image Processing, 1999 (ICIP 99)*], **3**, 159–163 (Oct. 1999).

[5] Ward, G., "Fast, robust image registration for compositing high dynamic range photographs from hand-held exposures," *Journal of Graphics Tools: JGT* **8**(2), 17–30 (2003).

[6] Szeliski, R., "Image alignment and stitching: A tutorial," Tech. Rep. MSR-TR-2004-92, Microsoft Research (MSR) (Oct. 2004).

[7] Guthier, B., Kopf, S., and Effelsberg, W., "Histogram-based image registration for real-time high dynamic range videos," in [*Proc. of IEEE International Conference on Image Processing (ICIP2010)*], (Sept. 2010).

[8] Aguiar, P., "Unsupervised simultaneous registration and exposure correction," in [*IEEE International Conference on Image Processing, 2006*], 361–364 (2006).

[9] Brajovic, V. and Kanade, T., "A sorting image sensor: An example of massively parallel intensity-to-time processing for low-latency computational sensors," in [*IEEE International Conference on Robotics and Automation, 1996. Proceedings.*], **2**, 1638–1643 (Mar. 1996).

[10] Nayar, S. K. and Mitsunaga, T., "High dynamic range imaging: Spatially varying pixel exposures," in [*IEEE Conference on Computer Vision and Pattern Recognition, 2000. Proceedings.*], **1**, 472–479 (June 2000).

[11] Nayar, S. K. and Branzoi, V., "Adaptive dynamic range imaging: Optical control of pixel exposures over space and time," in [*Ninth IEEE International Conference on Computer Vision, 2003. Proceedings.*], **2**, 1168 – 1175 (Oct. 2003).

[12] Acosta-Serafini, P., *Predictive multiple sampling algorithm with overlapping integration intervals for linear wide dynamic range integrating image sensors*, PhD thesis, Massachusetts Institute of Technology (2004).

[13] Wang, H., Raskar, R., and Ahuja, N., "High dynamic range video using split aperture camera," in [*IEEE 6th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras (OMNIVIS, in conjunction with ICCV05)*], IEEE (2005).

[14] Kang, S. B., Uyttendaele, M., Winder, S., and Szeliski, R., "High dynamic range video," *ACM Transactions on Graphics (TOG)* **22**, 319 – 325 (July 2003).

[15] Guthier, B., Kopf, S., and Effelsberg, W., "Capturing high dynamic range images with partial re-exposures," in [*Proceedings of the IEEE 10th Workshop on Multimedia Signal Processing (MMSP)*], (2008).

[16] Unger, J., Gustavson, S., Ollila, M., and Johannesson, M., "A real time light probe," in [*Proceedings of the 25th Eurographics Annual Conference, vol. Short Papers and Interactive Demos*], 17–21 (2004).

[17] Yoshida, A., Blanz, V., Myszkowski, K., and Seidel, H.-P., "Perceptual evaluation of tone mapping operators with real-world scenes," in [*Human Vision and Electronic Imaging X, IS&T/SPIE's 17th Annual Symposium on Electronic Imaging (2005)*], Rogowitz, B. E., Pappas, T. N., and Daly, S. J., eds., *SPIE Proceedings Series* **5666**, 192–203, SPIE, San Jose, USA (Jan. 2005).

[18] Ashikhmin, M. and Goyal, J., "A reality check for tone-mapping operators," *ACM Trans. Appl. Percept.* **3**(4), 399–411 (2006).

[19] Cadik, M., Wimmer, M., Neumann, L., and Artusi, A., "Evaluation of hdr tone mapping methods using essential perceptual attributes," *Computers & Graphics* **32**(3), 330–349 (2008).

[20] Ledda, P., Chalmers, A., Troscianko, T., and Seetzen, H., "Evaluation of tone mapping operators using a High Dynamic Range display," *ACM Transactions on Graphics* **24**, 640–648 (July 2005).

[21] Devlin, K., Chalmers, A., Wilkie, A., and Purgathofer, W., "Tone reproduction and physically based spectral rendering," in [*State of the Art Reports, Eurographics 2002*], Fellner, D. and Scopignio, R., eds., 101–123, The Eurographics Association (Sept. 2002).

[22] Miller, G. S. and Hoffman, C. R., "Illumination and reflection maps: Simulated objects in simulated and real environments," *SIGGRAPH '84: Course Notes for Advanced Computer Graphics Animation* , 1–12 (July 1984).

[23] Tumblin, J. and Rushmeier, H., "Tone reproduction for realistic images," **13**, 42–48 (Nov. 1993).

[24] Ferwerda, J. A., Pattanaik, S. N., Shirley, P., and Greenberg, D. P., "A model of visual adaptation for realistic image synthesis," in [*SIGGRAPH 96 Conference Proceedings*], Rushmeier, H., ed., *Annual Conference Series*, 249–258, ACM SIGGRAPH, Addison Wesley (Aug. 1996).

[25] Ward, G., "A contrast-based scalefactor for luminance display," *Graphics Gems IV* , 415–421 (1994).

[26] Drago, F., Myszkowski, K., Annen, T., and Chiba, N., "Adaptive logarithmic mapping for displaying high contrast scenes," in [*Computer Graphics Forum*], **22**(3), 419–426 (2003).

[27] Larson, G. W., Rushmeier, H., and Piatko, C., "A visibility matching tone reproduction operator for high dynamic range scenes," *IEEE Transactions on Visualization and Computer Graphics* **3**(4) (1997).

[28] Reinhard, E., Stark, M., Shirley, P., and Ferwerda, J., "Photographic tone reproduction for digital images," *ACM Transactions on Graphics* **21**(3), 267–276 (2002).

[29] Ashikhmin, M., "A tone mapping algorithm for high contrast images," in [*Rendering Techniques 2002 (Proceedings of the Thirteenth Eurographics Workshop on Rendering)*], 145–156, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland (June 2002).

[30] Meylan, L. and Süsstrunk, S., "High dynamic range image rendering with a retinex-based adaptive filter," *IEEE Transactions on Image Processing* **15**(9), 2820–2830 (2006).

[31] Rahman, Z.-U., Jobson, D. J., and Woodell, G. A., "Retinex processing for automatic image enhancement," *Journal of Electronic Imaging* **13**(1), 100–110 (2004).

[32] Oppenheim, A., Schafer, R., and Stockham, T.G., J., "Nonlinear filtering of multiplied and convolved signals," *Proceedings of the IEEE* **56**, 1264–1291 (Aug. 1968).

[33] Fattal, R., Lischinski, D., and Werman, M., "Gradient domain high dynamic range compression," *ACM Transactions on Graphics* **21**(3), 249–256 (2002).

[34] Technical Committee 3.1, "An analytic model for describing the influence of lighting parameters on visual performance, Vol. 1: Technical foundations," *CIE 19/2.1* (1981).

[35] Pattanaik, S., Tumblin, J., Yee, H., and Greenberg, D., "Time-dependent visual adaptation for fast realistic image display," in [*Proceedings of the 27th annual conference on Computer graphics and interactive techniques*], 47–54, ACM Press/Addison-Wesley Publishing Co. New York, NY, USA (2000).

[36] Benoit, A., Alleysson, D., Herault, J., and Callet, P., [*Spatio-temporal Tone Mapping Operator Based on a Retina Model*], 12–22, Springer-Verlag, Berlin, Heidelberg (2009).

[37] Meylan, L., Alleysson, D., and Süsstrunk, S., "Model of retinal local adaptation for the tone mapping of color filter array images," *J. Opt. Soc. Am. A* **24**(9), 2807–2816 (2007).

[38] Goodnight, N., Wang, R., Woolley, C., and Humphreys, G., "Interactive time-dependent tone mapping using programmable graphics hardware," in [*ACM SIGGRAPH 2005 Courses*], *SIGGRAPH '05*, ACM, New York, NY, USA (2005).

[39] Durand, F. and Dorsey, J., "Interactive tone mapping," in [*Proceedings of the Eurographics Workshop on Rendering Techniques 2000*], 219–230, Springer-Verlag, London, UK (2000).

[40] Wang, T.-H., Wong, W.-S., Chen, F.-C., and Chiu, C.-T., "Design and implementation of a real-time global tone mapping processor for high dynamic range video," in [*IEEE International Conference on Image Processing, 2007 (ICIP 2007)*], **6**, 209–212 (2007).

[41] Lee, C. and Kim, C.-S., "Gradient domain tone mapping of high dynamic range videos," in [*IEEE International Conference on Image Processing, 2007 (ICIP 2007)*], **3**, 461–464 (2007).

[42] Krawczyk, G., Myszkowski, K., and Brosch, D., [*HDR Tone Mapping*], 147–178, Springer Series in Advanced Microelectronics (26), Springer, Heidelberg (2007).

[43] International Telecommunication Union, "Methodology for the subjective assessment of the quality of television pictures." ITU-R BT.500 Recommendation (Jan. 2002).

[44] European Broadcasting Union, "SAMVIQ - Subjective assessment methodology for video quality." Report by the EBU Project Group B/VIM (May 2003).

[45] Ferwerda, J. A., "Elements of early vision for computer graphics," *IEEE Comput. Graph. Appl.* **21**(5), 22–33 (2001).

[46] Stevens, J. C. and Stevens, S. S., "Brightness function: Effects of adaptation," *J. Opt. Soc. Am.* **53**, 375–385 (Mar 1963).

[47] Stevens, S. S., "The surprising simplicity of sensory metrics," *American Psychologist* **17**(1), 29–39 (1962).

[48] http://ls.fmi.uni-mannheim.de/en/pi4/department/people/benjamin-guthier/projekte/.

[49] Olson, D. L. and Delen, D., [*Advanced Data Mining Techniques*], Springer, Berlin Heidelberg, 1 ed. (2008).

[50] Kohavi, R., "A study of cross-validation and bootstrap for accuracy estimation and model selection," in [*Proc. of the 14th Intl. Conf. on Artificial intelligence - Volume 2*], 1137–1143 (1995).