This is a preliminary version of an article published by
Benjamin Guthier, Stephan Kopf, Wolfgang Effelsberg
Optimal Shutter Speed Sequences for Real-Time HDR Video
Proc. of IEEE International Conference on Imaging Systems and Techniques (IST), pp. 305-308, July 2012

Optimal Shutter Speed Sequences for Real-Time HDR Video

Benjamin Guthier, Stephan Kopf, Wolfgang Effelsberg
University of Mannheim, Germany
{guthier, kopf, effelsberg}@informatik.uni-mannheim.de

***Abstract*—A technique to create High Dynamic Range (HDR) video frames is to capture Low Dynamic Range (LDR) images at varying shutter speeds. They are then merged into a single image covering the entire brightness range of the scene. While shutter speeds are often chosen to vary by a constant factor, we propose an adaptive approach. The scene's histogram together with functions judging the contribution of an LDR exposure to the HDR result are used to compute a sequence of shutter speeds. This sequence allows for the estimation of the scene's radiance map with a high degree of accuracy. We show that, in comparison to the traditional approach, our algorithm achieves a higher quality of the HDR image for the same number of captured LDR exposures. Our algorithm is suited for creating HDR videos of scenes with varying brightness conditions in real-time, which applications like video surveillance benefit from.**

***Index Terms*—HDR Video, Shutter Speed, Real-Time**

I. Introduction

A recurring problem in video surveillance is the monitored scene having a range of brightness values that exceeds the capabilities of the capturing device. An example would be a video camera mounted in a bright outside area, directed at the entrance of a building. Because of the potentially big brightness difference, it may not be possible to capture details of the inside of the building and the outside simultaneously using just one shutter speed setting. This results in under- and overexposed pixels in the video footage, impeding the use of algorithms for face recognition and human tracking. See Figure 1 for an example. A low-cost solution to this problem is temporal exposure bracketing, i.e., using a set of LDR images captured in quick sequence at different shutter settings [1]. Each LDR image then captures one facet of the scene's brightness range. When fused together, an HDR video frame is created that reveals details in dark and bright regions simultaneously.

Fig. 1. The inside of the building is much darker than the outside. There is no shutter speed setting that exposes both correctly at the same time. A solution to this problem is using a sequence of shutter speeds and merging the images together.

In a video surveillance scenario, capturing and fusion must be performed in real-time. One way to speed up this process is to only capture as few LDR images as necessary, that is, to optimally choose shutter speeds at which to capture. In the surveillance example above, it may be a sensible choice to only use the two exposures shown in Figure 1. Such a choice can only be made if the scene's brightness histogram is considered.

Barakat et al. [2] focus entirely on minimizing the number of exposures while covering the entire dynamic range of the scene. Minimum and maximum of the scene's irradiance range are taken into account, and the least possible overlap of exposures is always chosen. The algorithm is a fast heuristic suitable for real-time use.

A very recent method to determine noise-optimal exposure settings uses varying gain levels [3]. For a given sum of exposure times, increasing gain also increases the SNR. The authors define SNR as a function over log radiance values. The extrema of the scene's brightness are considered.

The authors of [4] developed a theoretical model for photons arriving at a pixel by estimating the parameters of a Gamma distribution. From the model, exposure values are chosen that maximize a criterion for recoverability of the radiance map.

The focus lies on the impact of saturated pixels on the HDR result.

In [5], an algorithm for estimating optimal exposure parameters from a single image is presented. The brightness of saturated pixels is estimated from the unsaturated surrounding. Using this estimation, the expected quality of the rendered HDR image for a given exposure time is calculated. The exposures leading to the lowest rendering error are chosen.

In an HDR video, the histogram of scene brightness values is often a by-product of tone mapping the previous frames [6]. The novel approach we present in this paper thus uses the entire histogram to calculate a shutter speed sequence in real-time. The shutter speeds are chosen in a way, such that frequently occurring brightness values are well-exposed in at least one of the captured LDR images. This increases the average SNR for a given number of exposures or minimizes the number of exposures required to achieve a desired SNR. We also give our definition of *contribution functions* to specify precisely what we mean by "well-exposed". In order to be applicable to video, we consider bootstrapping and convergence to a stable shutter sequence. Additionally, we introduce a stability criterion for the shutter speeds to prevent flicker in the video. Due to space limitations of this paper, we have left out some of the details. A more extensive description of our work can be found in our technical report [7].

In the following section, we introduce weighting functions for LDR pixels and give our definition of *contribution functions*. Section III then defines log radiance histograms and demonstrates a useful relationship between them and contribution functions which is exploited by our algorithm. The algorithm for finding optimal shutter speed sequences itself is described in Section IV. The quality of the HDR images produced by our optimal shutter sequences and the computational cost are analyzed in Section V of this paper. Section VI concludes the paper.

II. Weighting Functions

An HDR image is a map of radiances contained in a scene. In order to reconstruct this radiance map from the pixel values of the captured LDR images, the camera's response function f must be known. For the duration Δt that the camera's shutter is open, a pixel on the CCD sensor integrates the scene radiance E, resulting in a total exposure of $E\Delta t$. The camera's response function then maps the exposure to a pixel value $I = f(E\Delta t)$, usually in the range of $[0, 255]$. When the shutter speeds Δt_i used to capture the LDR images are known, the inverse of the response function can be used to make an estimate $\tilde{E}_i$ of the original radiance from pixel value I_i in LDR image i:

$$\tilde{E}_i = \frac{f^{-1}(I_i)}{\Delta t_i}. \tag{1}$$

A good approximation of the radiance value at a pixel in the HDR image is then obtained by computing a weighted average over all estimates $\tilde{E}_i$:

$$E = \frac{\sum_i w(I_i)\tilde{E}_i}{\sum_i w(I_i)}. \tag{2}$$

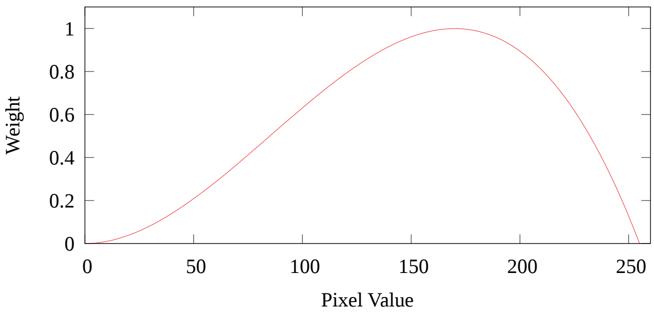

Fig. 2. The weighting function we use in our experiments. The weight of a pixel is its value multiplied by a hat function normalized to a maximum weight of 1.

The weighting function w determines how much the radiance estimate $\tilde{E}_i$ from a pixel I_i contributes to the corresponding HDR pixel E. In other words, it judges a pixel's usefulness for recovering a radiance value based on its brightness value. Note that without prior calibration, radiance values E computed like this only represent physical quantities up to an unknown scale factor. This is sufficient for our purpose. We thus use the terms *radiance* and *scaled radiance* interchangeably to denote the pixel values of an HDR frame.

Weighting functions are usually chosen to reflect noise characteristics of a camera, the derivative of its response function (i.e., the camera's sensitivity), and saturation effects. They are often found in the literature as parts of HDR creation techniques. Even though various weighting functions exist, they often share a few common properties. Most notably, the extremes of the pixel range are always assigned zero weight. This means that pixels with these values contain no useful information about the real radiance. Another common attribute of weighting functions is that pixels with a medium to high value are considered to be more faithful than dark pixels. This is due to the fact that a large portion of the image noise (e.g., quantization noise, fixed pattern noise) is independent of the amount of light falling onto the pixel. A bright pixel thus has a better signal-to-noise ratio than a dark one. Figure 2 shows an exemplary weighting function. In our experiments, we found that the function shown in the plot gives the best results, but our approach also works for any other choice.

For a given shutter speed Δt, we can calculate how well a radiance value E can be estimated from an image captured at Δt by combining the response and the weighting function. A radiance value E is mapped to a pixel value using the camera's response function f. The weighting function w then assigns a weighting to the pixel value. We define

$$c_{\Delta t}(E) = w(f(E\Delta t)) \tag{3}$$

as the *contribution* of an image captured at Δt to the estimation of a radiance value E. In the special case of a linear response function, $c_{\Delta t}$ looks like a shifted and scaled version of w. An example for a contribution function in the log domain is shown in Figure 3.

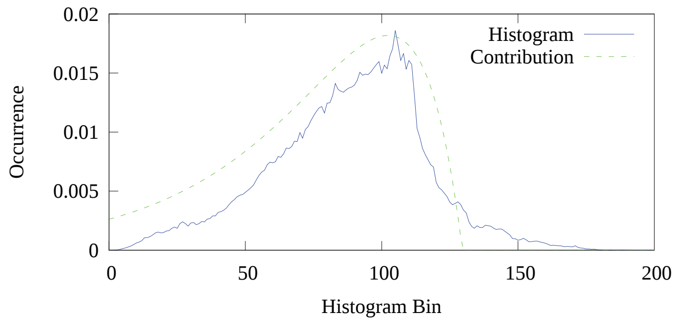

Fig. 3. Example of a log radiance histogram. The dashed line is the contribution function in the log domain of the first shutter speed chosen by our algorithm. The exposure was chosen such that it captures the most frequently occurring radiance values best.

III. Log Radiance Histograms

When creating HDR videos in real-time, the scene's brightness distribution is known from the previous frames. Additionally, some tone mapping operators create histograms of scene radiance values as a by-product or can be modified to create them with little extra effort. In this section, we describe how a log radiance histogram can be used to calculate a sequence of shutter speeds Δt_i which allows the most accurate estimation of the scene's radiance. We do this by choosing the Δt_i such that the peaks of the contribution functions $c_{\Delta t_i}(E)$ of the LDR images coincide with the peaks in the histogram. That is, radiance values that occur frequently in the scene lead to LDR images to be captured which measure these radiance values accurately. This is illustrated in Figure 3.

The histogram over the logarithm of scene radiance has M bins. Each bin with index $j = 1, ..., M$ corresponds to the logarithm of a discrete radiance value: $b_j = log(E_j)$. Bin j counts the number $H(j)$ of pixels in the HDR image having a log radiance of b_j. The bins have even spacing in the log domain, meaning that for any j, the log radiance values b_j and b_{j+1} of two neighboring bins differ by a constant $\Delta b = b_{j+1} - b_j$. The non-logarithmic radiance values corresponding to two neighboring bins thus differ by a constant factor $exp(\Delta b) = exp(b_{j+1})/exp(b_j) = E_{j+1}/E_j$.

Equation 3 states that, for a given shutter speed Δt and an LDR image captured using Δt, the value of $c_{\Delta t}(exp(b_j))$ indicates how accurately log radiance b_j is represented in the LDR image. When considering log radiance histograms, the continuous contribution function is reduced to a discrete vector of contribution values. It has one contribution value for each radiance interval of the histogram. We can now exploit a useful relationship between the log radiance histogram and our contribution vector: Shifting the contribution vector by a number of s bins leads to

$$\begin{aligned} & c_{\Delta t}(exp(b_j + s\Delta b)) \\ = \; & w(f(exp(b_j)exp(\Delta b)^s \Delta t)) \\ = \; & w(f(exp(b_j)\Delta t')) \\ = \; & c_{\Delta t'}(exp(b_j)), \end{aligned}$$

where

$$\Delta t' = exp(\Delta b)^s \Delta t. \tag{4}$$

This means that the contribution vector corresponding to shutter speed $\Delta t'$ is identical to a shifted version of the original vector. We thus easily obtain an entire series of contribution vectors for shutter speeds that differ by a factor of $exp(\Delta b)^s$. In other words, only the shift, but not the shape of the contribution function depends on the shutter speed in the log domain. This allows us to move the contribution function over a peak in the histogram and then derive the corresponding shutter speed using the above formula.

IV. Optimal Shutter Sequence

In order to compute an optimal shutter speed sequence, we first calculate an initial contribution vector from the known camera response and a chosen weighting function. Camera response functions can be estimated as described for example in [8], [9]. The initial shutter speed Δt to compute $c_{\Delta t}$ can be chosen arbitrarily. For ease of implementation, we choose Δt such that the first histogram bin is mapped to a pixel value of 1, that is $f(exp(b_1)\Delta t) = 1$. Note that $f^{-1}(0)$ is not uniquely defined in general. The size of the contribution vector depends on the dynamic range of the camera, reflected in its response function. Reaching a certain scene radiance $E_{N+1} = exp(b_{N+1})$, the camera's pixels will saturate, resulting in $f(exp(b_j)\Delta t) = 255$ for $j \geq N + 1$ in case of an 8 bit sensor. It is safe to assume that any reasonable weighting function assigns zero weight to this pixel value. Hence, the contribution vector $c_{\Delta t}(E_j) = w(f(exp(b_j)\Delta t))$ consists of N nonzero values. It can be shifted to $M + N - 1$ possible positions in the log radiance histogram. Each shift position s corresponds to a shutter speed Δt_i, which can be calculated using Equation 4: $\Delta t_i = exp(\Delta b)^s \Delta t$. This equivalence between shutter and shift is utilized later.

Here, we explain how a new shutter speed is added to an existing shutter sequence. The first shutter can be determined analogously. So we assume that the sequence already consists of a number of shutter speeds Δt_i. To each Δt_i belongs a contribution vector $c_{\Delta t_i}(E_j)$, with $E_j = exp(b_j)$ being the radiance values represented by the histogram bins. See Figure 3 for an example. We now need to decide whether to add another shutter to the sequence or not, and find out which new shutter brings the biggest gain in image quality. For this purpose, we define a *combined contribution vector* $C(E_j)$ that expresses how well the radiances E_j are captured in the determined exposures. We make the assumption, that the quality of the measurement of a radiance value only depends on the highest contribution value any of the exposures achieves for it. The combined contribution is thus defined as the maximum contribution for each histogram bin

$$C(E_j) = \max_i \left(c_{\Delta t_i}\left(E_j\right)\right). \tag{5}$$

This definition can now be used to calculate a single *coverage value* C to estimate how well-exposed the pixels in the scene are in the exposures. C is obtained by multiplying the frequency of occurrence of a radiance value $H(j)$ by the combined contribution $C(E_j)$ and summing up the products:

$$C = \sum_{j=1}^{M} C(E_j)H(j). \tag{6}$$

This is essentially the same as the cross correlation between the two. The algorithm tries out all possible shifts between a new contribution vector and the log histogram. The shutter speed corresponding to the shift that leads to the biggest increase of C is added to the sequence. If the histogram is normalized such that its bins sum up to 1 and the weighting function has a peak value of 1, then C is in the range of $[0..1]$ and can be expressed as a percentage. $C = 1$ then means that for each radiance value in the scene, there exists an exposure which captures it perfectly.

However, perfect coverage is not achievable in a realistic scenario. It is more practical to stop adding shutters to the sequence once a softer stop criterion is met. We came up with three different stop criteria: the total number of exposures, a threshold for C and a maximum sum of shutter speeds. The criterion that limits the *total number of exposures* is always active. It guarantees that the algorithm terminates after calculating a finite number of shutter speeds. We also use this criterion to manually choose the number of exposures for our evaluation for better comparability.

The *threshold for the coverage value C* is a quality criterion. A threshold closer to 1 allows for a better estimation of scene radiance, but requires to capture more exposures. We chose $C \geq 0.9$ for our running system.

For the type of camera we employ, the capture time of a frame is roughly proportional to the exposure time. And since we are interested in capturing real-time video at 25 frames per second, the sum of all shutter speeds must not exceed 40 milliseconds. Note that the camera exposes new frames in parallel to the processing of the previous ones. So we have indeed nearly the full HDR frame time available for capturing. Our third stop criterion is an adjustable threshold for the *sum of shutter speeds*. However, it should be made clear that the algorithm has little control over meeting this requirement. In the example shots we took, only two exceeded the threshold. But they in turn overshot it by a large factor. We argue that it is the camera operator's responsibility to adjust aperture and gain or to use a different lens to cope with particularly dark scenes.

The algorithm described here is greedy in that it does not reconsider the shutter speeds it already chose. We added a second iteration over the shutter sequence to allow for some hindsight refinement. Details on the refinement and its evaluation can be found in the technical report [7].

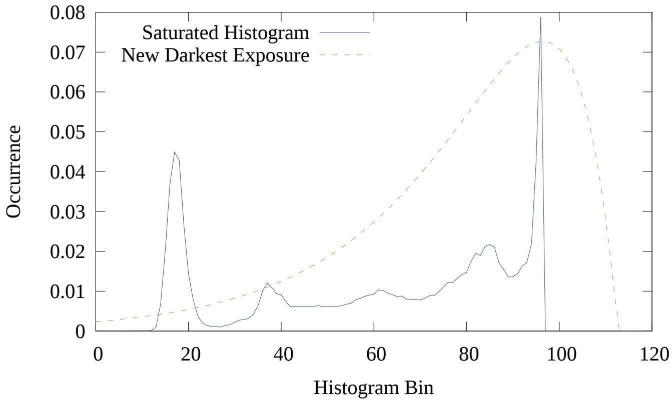

Fig. 4. Some areas of the scene are overexposed even in the darkest exposure. It shows up as a peak at the highest radiance value in the histogram. In the next frame, the algorithm chooses a shutter speed that covers the peak. By doing so, areas with a higher radiance than the previous maximum can still be captured faithfully.

So far, we described the algorithm to determine a sequence of shutter speeds for a single HDR frame based on a perfect histogram of the scene. However, there are two major problems that arise when applying this algorithm to HDR video directly: imperfect histograms and flicker.

Perfect histograms are not available in a real video. The available histograms are created from the previous frame which generally differs from the current one. Furthermore, the dynamic range covered by the histogram is only as high as the range covered by the previous exposure set. For example if the camera pans towards a window looking outside, the bright outdoor scene may be saturated even in the darkest exposure. This shows up as a thin peak at the end of the histogram of the previous frame (see Figure 4). How bright are these pixels really? To find out, the algorithm needs to produce a shutter sequence that covers a larger dynamic range than the histogram of the previous frame indicates. This allows the sequence to adapt to changes in the scene.

We accomplish this by treating the first shutter in the sequence differently. The special treatment is based on the observation that underexposed images contain more accurate information than overexposed ones. The dark pixels in an underexposed image are a noisy estimate of the radiance in the scene. However, this noise is unbiased. Saturated pixels on the other hand always have the maximum pixel value, no matter how bright the scene actually is. As a consequence of this observation, the first shutter is chosen such that its contribution peak covers the highest radiance bin of the histogram. The peak of a weighting function is usually not located at the highest possible pixel value. This means that radiances beyond the peak – if existing in the next frame – are still represented by a non-saturated pixel. See Figure 4 for an example. This allows to faithfully record radiance values that are a certain percentage higher than the previous frame's maximum, and the sequence can adapt to brighter scenes. Change towards a darker scene is less critical, because underexposed pixels still contain enough information about the real radiance to calculate a new longer shutter time. With adaptation enabled, bootstrapping becomes straightforward. We can start with any set of shutter speeds and arrive at the correct values after a few frames. The speed of adaptation is evaluated in the experimental results section of this paper.

The second problem to deal with when applying our algorithm to HDR video is flicker. It is a side effect of changing the shutter sequence over time. Consider the following scenario: A bright saturated area like a white wall leads to a peak at the highest histogram bin. This gives rise to a darker exposure taken in the next frame as shown in Figure 4. The darker exposure causes the histogram peak to spread out over several bins. It may now cause too little extra coverage to justify the darkest exposure. In this situation, the algorithm oscillates between including the lowest shutter speed and omitting it. In the resulting video, the white wall would alternate between having texture and being completely saturated. Stable shutter sequences are also desirable for a better use of camera buffers. We thus impose a stability criterion upon the shutter sequence. It is based on the definition of whether two given shutter speed sequences are *similar*. This similarity is defined in terms of a threshold over the averaged percentual difference between the shutters of the two sequences. The same shutter sequence is used for capturing images until a sufficiently high number of consecutive non-similar sequences have been calculated. Only then the new capture parameters are transmitted to the camera. For details on the stability criterion and an analysis of its behavior in a real-time HDR video system see [7].

V. Experimental Results

This section presents the evaluation of our algorithm for optimal shutter speed sequences. Section V-A describes a subjective user study we conducted to assess the HDR image quality our approach achieves compared to the traditional way of choosing evenly spread shutters. Section V-B contains an analysis of the algorithm's adaptation to changing brightness conditions and of its processing time.

A. Subjective User Study

A detailed description of the setup of our subjective user study is contained in our technical report. In this paper, we focus on presenting the results. 27 participants took part in the study. It was conducted over a website that allows to rate the quality of HDR images.[1] The subjects were shown twelve datasets of various HDR scenes. See Figures 5 and 6 for an example. Each dataset consisted of three HDR images: a reference image, an image created using shutter speeds from our approach and one where evenly spaced shutters were used. The two survey images were shown in random order to avoid subjective bias. Each of the two images had to be rated using the five scores (numerical value in parentheses): Very Good (5), Good (4), Average (3), Poor (2), Very Poor (1).

[1] http://pi4.informatik.uni-mannheim.de/∼bguthier/survey/

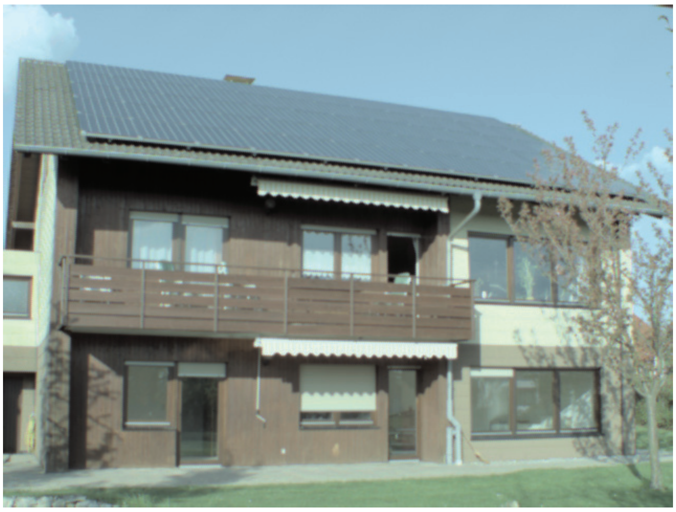

Fig. 5. Reference image of an exemplary scene used in our subjective evaluation.

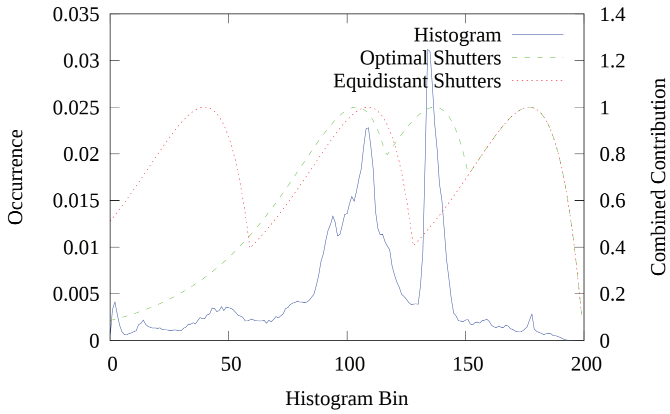

Fig. 6. Normalized log radiance histogram of Figure 5. The dashed lines show the two combined contribution functions. It can be seen, that the equidistant shutters disregard the histogram and an exposure is captured that adds little to the coverage value.

We used an AVT Pike F-032C FireWire camera capable of capturing 208 VGA frames per second with an aperture of f/2.8. Each scene was captured as a set of 79 LDR exposures covering the entire range of our camera's shutter settings. All 79 exposures were used to generate the reference image and the log radiance histogram of each scene. Only those exposures best matching the shutter speeds determined by the two algorithms were merged to create the survey images. The number of shutter speeds used was the same for both. It was manually chosen to be low enough for a discernible degradation of image quality to facilitate the rating process.

The main reason to use HDR still images instead of video for subjective quality assessment is the availability of a perfect reference image and with it the reproducibility of the results. Capturing 79 LDR exposures at varying shutter speeds allows to reconstruct the real scene radiance accurately. The shutter values are sufficiently close together to simulate arbitrary shutter sequences. Capturing the same amount of exposures for an HDR reference video is not feasible. Another reason is the difficulty to capture the optimal and the equidistant shutter video both at once. And lastly, HDR video may introduce various new artifacts like misalignment of the exposures or temporally inconsistent tone mapping. These additional artifacts may mask the difference between the two shutter speed choices.

The 27 participants rating 12 datasets resulted in a total of 317 valid pairs of scores – one for optimal and one for equidistant shutters. Averaging them yields a score of 3.73 for the optimal shutter algorithm and 2.83 for the equidistant approach. Note that the absolute value of the score is meaningless as the survey images were intended to be flawed. Our approach achieved a better score in 70%, the same in 16% and a worse score in 14% of the ratings.

B. Objective Measurements

In the experiment described in the following, we investigated the time it takes for our algorithm to adapt to changes in the scene. We did this by keeping the scene and the camera static, choosing extreme shutter speed sequences and measuring the number of frames it takes to stabilize. The scene and aperture of the camera were chosen such that the optimal shutter sequence consisted of four shutter values around the center of the camera's shutter range. By *center*, we mean the middle value in the log domain with the same *factor* to the lowest as to the highest shutter. For our camera, the shutter value of 1.74 ms is a factor of 47 higher than the minimum and lower than the maximum shutter. Three different starting sequences were set: the sequence consisting of only the shortest possible shutter, the longest shutter and a sequence covering the full shutter range with one stop between the shutters. We then measured the number of frames the algorithm took until the shutter sequence did not change anymore. The values are averaged over 375 runs for each of the three starting sequences.

As expected, the full coverage sequence adjusted the fastest. It took 2.07 frames to stabilize. This means that the stable sequence could be directly calculated from the first HDR frame in almost all of the iterations. From only the shortest shutter value, it took exactly 3 frames to stabilize. The algorithm already calculated three shutters in the second frame and reached the final sequence in the third. The worst adaptation speed was achieved when starting from only the longest shutter value, that is, from the brightest image. The lowest shutter in the sequence was approximately halved in every frame. In the average, the sequence was changing for 8.20 frames. This confirms our previous statement that convergence towards darker scenes (i.e., higher shutter values) is easier. It also justifies the special treatment of the first shutter in the sequence as described earlier.

Since it is our goal to perform shutter sequence computations in real-time to create HDR videos, we measured the processing time taken by our algorithm in a real-time HDR video system with an AMD Athlon II X2 250 dual-core CPU. The measurement was taken over a period of 15 seconds ($\approx$ 375 HDR frames). The monitored scene contained moving objects and many camera pans between dark indoor and very bright outdoor areas. As mentioned earlier, we assume that the histogram of the previous HDR frame was computed during tone mapping. Histogram creation is thus not included in these measurements. The experiment showed that 96.5% of our algorithm's processing time is spent for trying out all possible shifts between contribution vector and histogram to find the next shutter speed with the best coverage value. As a consequence, the processing time is roughly proportional to the number of shutters in the sequence. We measured 0.30 ms per shutter value including refinement. For comparison, the entire process of creating a displayable HDR frame from 2 to 8 base exposures takes 6 to 15 ms on a GPU. In a 25 fps real-time HDR video system, there are 40 ms available for processing each frame. Our algorithm is thus fast enough to be used in this application.

VI. Conclusions

We presented an approach to computing shutter speed sequences for temporally bracketed HDR videos. Our goal is to maximize the achieved HDR image quality for a given number of LDR exposures. This is done by consecutively adding shutters to the sequence that contribute to the image quality the most. It allows to save capturing and processing time over the traditional approach by being able to reduce the number of LDR exposures without impairing quality. Analysis of the algorithm's behavior in a real-time HDR video system showed that it is suitable for such a scenario and can be employed in video surveillance.

References

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

[2] N. Barakat, A. N. Hone, and T. E. Darcie, "Minimal-bracketing sets for high-dynamic-range image capture," *IEEE Trans. on Image Processing*, vol. 17, no. 10, 2008.

[3] S. Hasinoff, F. Durand, and W. Freeman, "Noise-Optimal Capture for High Dynamic Range Photography," in *Proc. of the 23rd IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[4] K. Hirakawa and P. Wolfe, "Optimal exposure control for high dynamic range imaging," in *Proc. of the 17th IEEE International Conference on Image Processing (ICIP)*, 2010.

[5] D. Ilstrup and R. Manduchi, "One-shot optimal exposure control," in *Proc. of the 11th European Conference on Computer Vision (ECCV)*, 2010.

[6] G. Ward, H. Rushmeier, and C. Piatko, "A visibility matching tone reproduction operator for high dynamic range scenes," *IEEE Transactions on Visualization and Computer Graphics*, vol. 3, no. 4, 1997.

[7] B. Guthier, S. Kopf, and W. Effelsberg, "Optimal shutter speed sequences for real-time hdr video," University of Mannheim, Tech. Rep., 2011, http://pi4.informatik.uni-mannheim.de/$\sim$bguthier/optshutter-TR.pdf.

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

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