Using Temporal Correlation for Fast and High-detailed Video Tone Mapping

Lino Coria, Panos Nasiopoulos
Department of Electrical and Computer Engineering
The University of British Columbia
Vancouver, BC, Canada
{linoc, panos}@ece.ubc.ca

Abstract— Tone Mapping Operators (TMOs) approximate the appearance of High-Dynamic-Range (HDR) content in standard displays. The few available video TMOs are computationally complex versions of existing schemes developed for still images. We introduce a fast method for video tone mapping that takes advantage of the strong temporal correlation that is usually found in video sequences. Each frame is divided into several blocks and we employ a motion estimation scheme that searches for similar blocks in a previously tone-mapped reference frame. The bilateral filter TMO is employed for the reference frames, but any other TMO can be used with our scheme. Experiments show that high-detailed tone-mapped content can be created without the need of employing a complex TMO on every frame.

Keywords- bilateral filter; high dynamic range; motion estimation; temporal correlation; tone mapping.

I. INTRODUCTION

High-Dynamic-Range (HDR) video is the next major development aiming at improving the video quality of experience. HDR content overcomes the visual limitations of traditional image/video media by capturing color data with much higher precision [1]. Although professional HDR capturing devices and displays have reached the prototype stage, their consumer versions are not expected to enter the market for a while yet. In fact, the vast majority of the consumer display devices available today can only offer a low dynamic range (LDR) of less than 100 to 1. Reproducing HDR content in LDR displays, a task known as tone mapping, is not a trivial process since there is a significant difference between the broad range of illumination values that were captured and the reduced amount of colors that can be reproduced. Tone Mapping Operators (TMOs) must be carefully designed so that the surplus of information available from HDR content can be employed to create more appealing images (i.e., with higher contrast) than the ones typically obtained with regular LDR cameras.

One of the most notable TMOs uses the bilateral filter to produce high-detailed LDR content [2]. In order to produce each output pixel, the bilateral filter takes into account both the spatial proximity and the magnitude similarities of the image’s pixels. Images created with the bilateral filter TMO are able to display high contrast regions similar to the original content but with a limited dynamic range.

Although a significant amount of work has been done around tone mapping schemes, research efforts have mainly concentrated on algorithms for still images. For the case of video, the straightforward solution is using TMOs on a frame-by-frame basis. However, better results in terms of complexity efficiency can be obtained by acknowledging and taking advantage of the temporal redundancies that are common in video sequences. In [3], for instance, a TMO for video content is presented which is an extension of [4], a scheme developed for static images. The approach for video takes into consideration the temporal luminance adaptation process which is modeled with an exponential decay function. Luminance values from previous frames contribute to the luminance value of the current frame. In [5], the gradient-domain tone mapping approach presented in [6] is extended towards video. The main idea of this method is to attenuate the magnitudes of HDR image gradients at each pixel. While [6] only considers the strengths of all the edges (from different scales) at every pixel location, the scheme proposed in [5] also employs the motion information of each pixel. Working on a pixel-by-pixel basis in a 3D array makes this scheme computationally intensive. Another video TMO is offered in [7]. This method is inspired by the retina model employed for image tone mapping [8]. Filters are employed in both time and space so that sudden changes in local luminance values do not occur among adjacent frames.

The aforementioned video tone mapping schemes are significantly more complex than the initial TMOs designed for still images and served as their foundation. With these new schemes, more operations need to be performed and larger data arrays are required in order to find the tone-mapped value of each pixel. In addition, each of these schemes cannot be easily adapted to incorporate any other tone mapping technique different than the one it is based on. Although the strong temporal correlation among adjacent frames is acknowledged throughout these methods, it would be desirable to exploit the temporal redundancies found in video sequences in order to simplify the tone mapping process, rather than to make it more complex. Of course, the high quality of the tone-mapped content should remain as the top priority.

To this end, we introduce a fast and efficient video tone mapping scheme that benefits from the high temporal correlation that is usually present in video content. Our method
is inspired by ideas behind video coding [9], in particular, the use of block-matching motion estimation. We employ this technique for identifying similarities among adjacent frames and use this information to reduce the amount of cumbersome tone mapping computations. The proposed algorithm relies on the bilateral filter TMO [2] which, as described in [1], provides a better reproduction of details at the edges than [4], since a large pixel region is used for the estimation of local adaptation. However, depending on the application and the requirements of the user, our approach may support several other TMOs. In our scheme, the video sequence is divided into groups of pictures (GOPs). Each GOP consists of an I-frame and several P-frames. The I-frame is used as reference and the bilateral filter scheme is applied to it, treating it as a standalone image. Every P-frame is divided into blocks and motion estimation is employed to find similar blocks in the I-frame. If an exact or similar match is found, we take advantage of the tone mapping information from the previous block. If not, tone mapping at a smaller scale is performed on this block. By working with blocks instead of pixels, our method is not as computationally demanding as other video tone mapping schemes and thus is suited for real-time applications.

The rest of the paper is organized as follows. Our video tone mapping scheme is described in detail in Section II. Section III presents an evaluation of the performance of our method. Finally, Section IV offers conclusions and ideas for future work.

II. PROPOSED VIDEO TONE MAPPING USING TEMPORAL CORRELATION

The temporal redundancy among adjacent video frames can be exploited so that tone mapping information that has been previously computed for one frame is employed for the next ones. By finding similarities between a new frame and a previous one, we could considerably speed up the video tone mapping process while still producing LDR content with high detail and quality. This is the goal of our scheme.

The scheme’s input is an HDR video sequence. This sequence is divided into groups of pictures (GOPs). Each GOP has the form IPP..., that is, an I-frame (or reference frame) followed by several P-frames (the frames where motion estimation will be employed). Different techniques are employed for tone-mapping I-frames and P-frames. We describe them next.

A. Tone-mapping of I-frames

I-frames are used as reference frames. In our scheme, the I-frame is treated as a standalone image during the tone mapping process. We have chosen to apply the bilateral filter TMO [2] to the I-frame, since it produces high quality images [1]. However, any other image tone mapping scheme can be chosen for this step. We briefly describe the bilateral filter tone mapping algorithm.

1) The color I-frame that we wish to tone-map is first transformed into a luminance component and two chroma components.

2) Using the bilateral filter, the HDR luminance component obtained from the I-frame is further decomposed into a base layer and a detail layer. This is achieved by using the HDR luminance component as the input of the bilateral filter. The base layer, $I_b$, which preserves the large-scale features of the image, is the output of the filter. The detail layer, $I_d$, is the division of the input intensity by the base layer.

3) Only the base layer is tone-mapped (i.e., has its contrast reduced). Details are preserved since the magnitude of the detail layer remains unaltered.

4) The LDR luma component is obtained by multiplying the tone-mapped base layer with the detail layer. The 8-bit LDR chroma components are linearly quantized from the HDR chroma arrays.

B. Tone-mapping of P-frames

P-frames are expected to resemble the I-frame from their GOP and thus, will rely on it as reference. Before the motion-estimation process begins, we first divide the color frame into a luminance component and two chroma components. We deal with the tone mapping of the chroma arrays the same way we did with the I-frame chroma components, that is, we linearly quantize them into 8-bit arrays. We then obtain a “rough copy” of both the luminance component for the P-frame and its corresponding I-frame luminance component. $P_{RC}$ and $I_{RC}$, respectively. $P_{RC}$ and $I_{RC}$ are scaled (downsized) and quantized versions (e.g., from 16 to 8 bits) of the original frames (see Fig. 1). These resulting images have fewer pixels with lower dynamic range than the originals. These “rough copies” will only be used to speed up the block-matching motion-estimation process. Once a good match is found, the original frames will be employed during the tone-mapping process.

Once $P_{RC}$ and $I_{RC}$ have been computed, the block-matching motion estimation process begins. This is performed on a block-by-block basis following a raster-scan pattern. Therefore, the first step of the motion estimation method is to divide $P_{RC}$ into non-overlapping blocks. For each block in $P_{RC}$, we employ the Adaptive Rood Pattern Search (ARPS) algorithm [10] for motion estimation in order to find a similar block in $I_{RC}$. We have chosen ARPS because it is a fast and simple block matching method that yields a fine performance (2 to 3 times faster than the traditional Diamond Search scheme with no reduction in PSNR). Occasionally, an exact match will be found. It is more likely, however, to find a similar match rather than an identical one. Because of this, we have divided the motion estimation process into three possible scenarios that depend on the similarity between the current block and the best match found on the reference frame. To this end, we employ a similarity threshold, which is a value that establishes which blocks are considered good matches. Different values were tested for this threshold and the results are shown in Section III. The three possible scenarios are described below:

1) Case 1: An exact match is found. If an exact match is found, then we use the information from the obtained motion vector to find the corresponding pixel block from $I_{LDR}$. Keep in mind that $P_{RC}$ and $I_{RC}$ are downsized versions of the actual video frames. This must be taken into account to compute the correct magnitude of the motion vector that corresponds to the chosen block on $I_{LDR}$. Once the desired reference block has been found, there are no extra computations needed for tone-
mapping the current block. We simply copy the corresponding \( I_{\text{LDR}} \) block into the corresponding location of \( P_{\text{LDR}} \) (see Fig. 1).

2) Case 2: A similar match is found. If there is not an exact match but the smallest Mean Absolute Difference (MAD) that was computed during the motion estimation process is below the similarity threshold, then we perform the following steps (which are illustrated in Fig. 2):

   a) We apply linear tone mapping to the current block (i.e., we reduce the bit-depth of the block’s pixels to 8 bits).

   b) We scale the magnitude of the motion vector so that it matches the correct size of the video frames. We use the corresponding block from the I-frame’s detail layer, \( I_{\text{D}} \) (which is the ratio of the HDR content and the base layer \( I_{\text{B}} \)).

   c) We multiply the values of the retrieved \( I_{\text{D}} \) block to those of the linearly tone-mapped block. The end result is a tone-mapped block with high contrast information obtained from \( I_{\text{D}} \).

   It is important to observe that, for Case 2, using block-matching motion estimation between adjacent frames might provide results that are not visually pleasing. Because we are tracking blocks instead of full objects, the final set of reference blocks might create a frame that doesn’t resemble the original content but clearly looks like a cluster of blocks put together. Performance evaluations have shown (see Section III) that in order to avoid this problem larger blocks (e.g., 40 \( \times \) 40-pixel blocks instead of the traditionally employed 8 \( \times \) 8-pixel blocks) should be used for our motion estimation process. By doing this, numerous pixels need to be matched and, therefore, it becomes very unlikely that the scheme will choose a block that does not belong to the object we are trying to track.

3) Case 3: A good match is not found. If the smallest MAD found during the motion estimation process is higher than the established similarity threshold, then information from the reference frame cannot be employed for the tone-mapping of the current block. In this case, the bilateral filter tone-mapping scheme is applied to the pixels of this block but, instead of using all the frame’s pixels for computing the output values, only a smaller region is employed. This region includes 1) the block to be tone-mapped and 2) some adjacent pixels from the top, bottom, left and right of this block. If these adjacent pixels were not included in the computations, the resulting tone-mapped block would show incorrect pixel representations around the block’s borders.

III. PERFORMANCE EVALUATION

We compare the visual fidelity of our tone-mapped video sequences to that of the LDR content created with the frame-by-frame tone-mapping approach. To this end, we use the Structural Similarity (SSIM) index [11] as an efficient way of assessing content fidelity. The goal of the following experiments is to select the most appropriate values for the various parameters of our algorithm so that our scheme works significantly faster than the straightforward frame-by-frame approach and the resulting LDR content is very similar to what can be obtained with the bilateral filter. We use the high-definition HDR video sequences “sunrise” and “library.”

For our tests, we vary the size of the blocks (from 5 \( \times \) 5 to 80 \( \times \) 80) and the value of the similarity threshold for Case 2 (from 4 to 20) while keeping the values of the other parameters fixed: an I-frame occurs every 4 frames, the downsampling factor is set to 4, the number of pixels that the motion estimation algorithm is allowed to move away from the center...
of the block in order to find the best match is set to 10, and the height and width of the region used for computing the bilateral filter for Case 3 is 1.2 times larger than the block’s height and width (i.e., for a $40 \times 40$ block the region is $48 \times 48$).

Fig. 3 illustrates the average SSIM indices that can be obtained with different parameter values. The SSIM index is higher than 0.9 when the threshold value is higher than 8. Thresholds higher than 20 are not shown since the SSIM values remain unchanged after that. It should also be observed that a better SSIM index can be obtained when using medium-size pixel blocks (i.e., $20 \times 20$ and $40 \times 40$).

Fig. 4 shows how many times faster our scheme is when compared to the frame-by-frame approach. Our method performs faster when high threshold values and large block sizes are employed. Based on this information, using $40 \times 40$ pixel blocks and a threshold value of 20 yields the best scenario in terms of speed (3.24 times faster than the frame-by-frame approach) and high image quality (0.92 SSIM index).

IV. CONCLUSION

In this paper we have introduced a fast method for performing video tone mapping which produces LDR content that is rich in visual details. Our method takes advantage of the strong temporal correlation that is common to virtually all video sequences. This is achieved by dividing each frame into several blocks and employing a fast block-matching motion estimation scheme that searches for similar blocks in a previously tone-mapped reference frame. Using information from the previously tone-mapped blocks, it is possible to create highly-detailed tone-mapped content without the need to employ a complex local tone-mapping operator on every frame. In our implementation, we have chosen to employ the bilateral filter TMO as the core of our scheme. However, one advantage of our method is that it can be extended for any other local tone-mapping operator. Our method is three times faster than the frame-by-frame tone mapping approach and produces LDR content that is as visually appealing.

REFERENCES


