GLOBALLY OPTIMIZED MULTIVIEW VIDEO COLOR CORRECTION USING DENSE SPATIO-TEMPORAL MATCHING

Beerend Ceulemans, Shao-Ping Lu, Peter Schelkens, Adrian Munteanu

Department of Electronics and Informatics,
Vrije Universiteit Brussel
Pleinlaan 2, Brussels, Belgium

Department of Future Media and Imaging,
iMinds V. Z. W.
G. Crommenlaan 8, Ghent, Belgium

ABSTRACT

Multiview video is becoming increasingly popular as the format for 3D video systems that use autostereoscopic displays or free-viewpoint navigation capabilities. However, the algorithms that drive these applications are not yet mature and can suffer from subtle irregularities such as color imbalances inbetween different cameras. Regarding the problem of color correction, state-of-the-art methods directly apply some form of histogram matching in blocks of pixels between an input frame and a target frame containing the desired color distribution. These methods, however, typically suffer from artifacts in the gradient domain, as they do not take into account local texture information. This paper presents a novel method to correct color differences in multiview video sequences that uses a dense matching-based global optimization framework. The proposed energy function ensures preservation of local structures by regulating deviations from the original image gradients.

***Index Terms* —** multiview video, color-correction, optical flow, Laplacian matrix optimization, structure preservation

1. INTRODUCTION

Multiview video has the capability of delivering a much more immersive multimedia experience compared to simple 2D or stereoscopic 3D by offering both 3D through stereopsis and 3D by motion parallax. New and exciting autostereoscopic and free viewpoint technologies are being developed and will soon be ubiquitous in the modern multimedia landscape. The main problem that is hindering the acceptance of these technologies is the fact that content acquisition and delivery systems are lagging behind. The required 3D content is captured using multi-camera setups which calls for efficient synchronization, calibration and compression methods. Moreover, capturing a scene with eighty cameras or more (which is what is being done in current exploratory experiments) is overkill when having to transport and serve this data at remote locations. Even state-of-the-art compression such as 3D-HEVC has a hard time coping with the enormous amount of data produced by the cameras. A solution to this problem is to perform acquisition using only a sparse set of capturing cameras and to synthesize additional views based on a few originals. However, sparsifying the camera setup might be associated with substantial color differences between neighboring cameras, originating from complex scene materials or miscalibration. This in turn can further impede compression and view synthesis which are already challenging on their own. Moreover, studies have shown that subtle color differences in stereo or multiview content can cause visual fatigue and binocular rivalry [1, 2]. For this reason it is desirable that automated algorithms can deal with such color differences.

Popular techniques for color correction perform global or block-based histogram matching [3–5]. Histogram methods are however generic, do not take structural information into account and therefore, do not ensure its preservation after color correction is performed. In [6], in order to preserve structural information, sparse SIFT keypoints are extracted and color-correction is performed in an averaging manner per region of a segmentation of the original image. [6] shows that correspondence-based color correction can lead to good results but the method works locally and is not very robust to wrongly matched SIFT points. In this paper we present an extension of our previously proposed color correction method [7] which also performs color correction based on correspondences but in combination with a global optimization step. In [7], we compute sparse correspondences between an input and a reference camera view and between the input view and the previous color corrected frame. In order to have a one-to-one mapping between input pixels and a reference set, we complete the mapping by using a histogram corrected input as another reference. Based on these correspondences, new pixel values are assigned according to the minimization of an energy function that regulates deviations from the original image structures. In this paper, we demonstrate that our framework can further benefit from dense correspondence matching by means of optical flow-based correspondences. Our color correction performs dense matching in the temporal dimension as well as sparse SURF matching across cameras and the output is given by the minimization of an energy function which is efficiently computed due to the sparse nature of the resulting linear system.

2. PROPOSED METHOD

A schematic representation of the proposed framework is depicted in Figure 1. It is based on our previous work [7] with the addition of dense optical-flow based correspondence matching. Histogram matching is only used for the very first frame and for the remainder of the sequence, dense optical flow and sparse SURF (Speeded Up Robust Features) [8] matching are used. When depth maps are available, denser inter-camera matching is also possible by means of view synthesis.

2.1. Energy formulation

We aim to compute optimal colors for the pixels in the input view taking another view as reference. Additionally, we want to preserve the original textures and structures as much as possible. To achieve this, we pose the problem as an optimization problem, namely:

$$E = E_d + \beta_r E_r \tag{1}$$

$$\hat{I}_n = \arg\min_I(E) \tag{2}$$

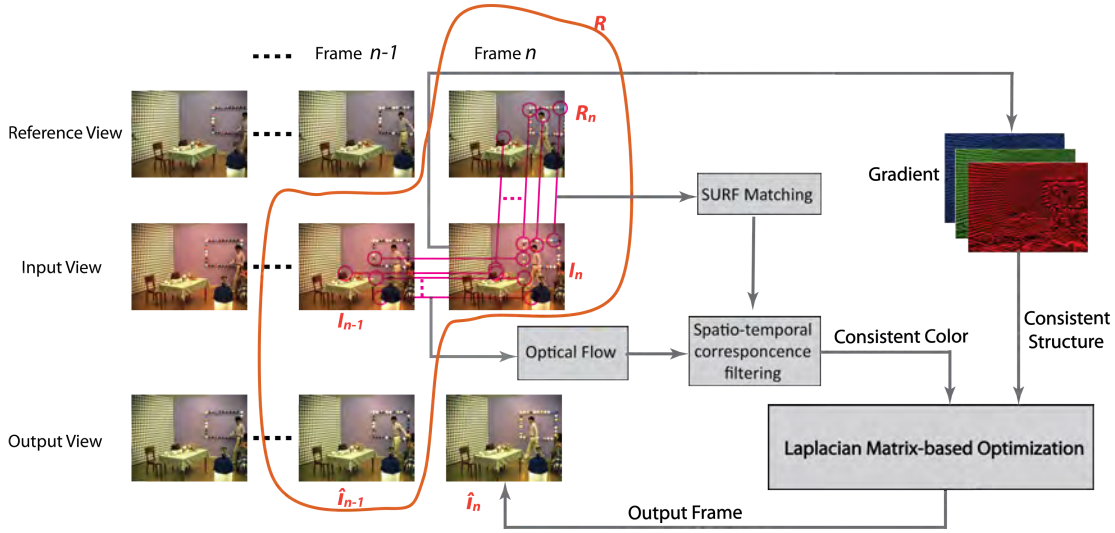


Figure 1. The generic framework describing the proposed multiview video color correction algorithm.

The total energy given by Equation (1) consists of a color consistency (*data*) term E_d and a structure preservation (*regularization*) term E_r which are balanced by the parameter β_r.

Color consistency. The data term E_d expresses that the color in the target frame $\hat{I}_n$ should be as consistent as possible to the color in an intermediate image $\tilde{I}$ which is constructed based on correspondences between the input image I_n and a reference set of known images $R = \{R_n, I_n, I_{n-1}, \hat{I}_{n-1}\}$, where R_n is the current reference frame, I_n is the current input frame, I_{n-1} is the previous input frame and $\hat{I}_{n-1}$ is the previous color-corrected frame. The construction of $\tilde{I}$ will be explained later. Using this image, we express the data term as:

$$E_d = \sum_{\mathbf{u}} \left(I(\mathbf{u}) - \tilde{I}(\mathbf{u}) \right)^2 \tag{3}$$

Structure consistency. In order to avoid distortions in the processed structure and textures, we impose a constraint on the gradient of the color correction result. We state that de gradient of this output should remain as close as possible to the original gradient:

$$E_r = ||\vec{\nabla} I - \vec{\nabla} I_n||^2 \tag{4}$$

Using finite differences and matrix-vector notations, we rewrite this energy as:

$$\begin{aligned} E_r = &(D_x I - D_x I_n)^T (D_x I - D_x I_n) \\ &+ (D_y I - D_y I_n)^T (D_y I - D_y I_n) \end{aligned} \tag{5}$$

where D_x and D_y denote the forward discrete differentiation operators and $(\cdot)^T$ is the transpose operator. Note that 2D-images are linearized to be represented as an N-dimensional vector rather than a matrix. All matrices are therefore of size $N \times N$. With this structure consistency term, we impose that correspondence matching between any image x and the input I_n should be the same as the matching between x and $\hat{I}_n$. Such matching is generally done based on gradient and luminance information and therefore invariant to the color correction.

2.2. Energy minimization

In order to minimize the proposed energy function in (1), we employ correspondence modeling and filtering followed by an efficient solver for Laplacian matrix inversion.

Correspondence modeling. The intermediate image $\tilde{I}$ is built from the reference set R using the following classes of image-wise correspondences:

1. temporal matching between consecutive frames in the output camera
2. spatial matching between the input/output and reference frames

The first mapping relation describes the color consistency in the temporal direction of the input view. The second class describes the color consistency between different camera views. In our previous work [7], we performed correspondence matching using SURF [8] and Fast Approximate Nearest Neighbor matching which only yields a sparse set of pairwise correspondences. In order to have a complete mapping between the input I_n and the intermediate image $\tilde{I}$, we also needed to map pixels of I_n to co-located pixels of a histogram-corrected version of I_n. We now remove this dependency by performing dense matching in the temporal direction by means of optical flow. We use the Simply Flow algorithm of [9] as implemented in the OpenCV library. In the very first frame we however still initialize our framework as in [7].

We now explain the construction of the intermediate correspondence image $\tilde{I}$. We assume that pixels $\mathbf{u} \in I_n$ have correspondences in either I_{n-1} or R_n. We say that $p(\mathbf{u}, \mathbf{v}) \in \psi_\mathbf{t}$ if pixel $\mathbf{u}$ corresponds to pixel $\mathbf{v}$ in the previous frame I_{n-1}. Similarly, we say that $p(\mathbf{u}, \mathbf{v}) \in \psi_\mathbf{c}$ if pixel $\mathbf{u}$ corresponds to pixel $\mathbf{v}$ in the other camera R_n. Built on these correspondences, we construct an image $\tilde{I}$ as follows:

$$\tilde{I}(\mathbf{u}) = \begin{cases} I_{n-1}(\mathbf{v}) + \alpha_\mathbf{n}(\mathbf{u}, \mathbf{v}), & \text{if } p(\mathbf{u}, \mathbf{v}) \in \psi_t \\ R_n(\mathbf{v}) + \tau_\mathbf{n}(\mathbf{u}, \mathbf{v}), & \text{if } p(\mathbf{u}, \mathbf{v}) \in \psi_c \end{cases} \tag{6}$$

where $\alpha_n(\mathbf{u}, \mathbf{v})$ is the original color difference between $I_n(\mathbf{u})$ and $I_{n-1}(\mathbf{v})$ and $\tau_n(\mathbf{u}, \mathbf{v})$ can be a color transistion model between the two cameras, which we put it to zero for simplicity. When also assigning weights β_t and β_c to these correspondences, we can write (3) as:

$$\begin{aligned} E_d = \beta_t \sum_{p(\mathbf{u},\mathbf{v}) \in \psi_t} &\left(I(\mathbf{u}) - \hat{I}_{n-1}(\mathbf{v}) - (I_n(\mathbf{u}) - I_{n-1}(\mathbf{v})) \right)^2 \\ &+ \beta_c \sum_{p(\mathbf{u},\mathbf{v}) \in \psi_c} (I(\mathbf{u}) - R_n(\mathbf{v}))^2 \end{aligned} \tag{7}$$

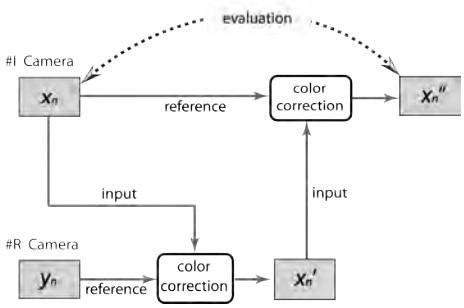


Figure 2. Proposed forward-reverse color correction evaluation.

Global optimization. We want to minimize the total energy given by equations (1), (5) and (7). We therefore introduce a diagonal matrix $W = \beta_r \mathbf{1}_N$ and another diagonal matrix Q defined as:

$$Q(\mathbf{u},\mathbf{u}) = \begin{cases} \beta_t & \text{if } p(\mathbf{u},\mathbf{v}) \in \psi_t \\ \beta_c & \text{if } p(\mathbf{u},\mathbf{v}) \in \psi_c \end{cases} \tag{8}$$

Using Q and $\tilde{I}$, we can write (7) more compactly in matrix-vector notation as:

$$E_d = \left(I - \tilde{I}\right)^T Q \left(I - \tilde{I}\right) \tag{9}$$

To obtain $\hat{I}_n$ which is the image I that minimizes the total energy function, we solve $dE/dI = 0$. The solution can be found by solving the following linear system:

$$\begin{aligned} \left[Q + D_x^T W D_x + D_y^T W D_y\right] \hat{I}_n \; = \\ Q\tilde{I} + (D_x^T W D_x + D_y^T W D_y) I_n, \end{aligned} \tag{10}$$

where D_x^T and D_y^T are backward discrete differentiation operators. Remember that all matrices are of size $N \times N$ where N is the number of pixels. At common video resolutions, the inhomogeneous Laplacian matrix on the left hand side easily becomes a few hundreds of gigabytes, which is impractical to fully hold in memory. The matrix however is sparse and symmetric positive definite so efficient solutions exist in the literature. We opt for Hierarchical Sparsification and Compensation [10] which has a low operation count and wall-clock time. The method greatly reduces the matrix' condition number to efficiently solve the problem.

3. EXPERIMENTAL EVALUATION

Because in the area of multiview color correction no ground truth data exists we employ a so-called forward-reverse evaluation [7]. This means that we color correct an image I to match a reference R and this corrected image $\hat{I}$, we further process in order to give it back the original colors of I. This way we can compute objective metrics such as Peak Signal to Noise Ration (PSNR) and Structure SIMilarity (SSIM) in order to evaluate the quality of our results.

We performed the experiments on the *Objects2* and *Flamenco2* multiview sequences. Both sequences consist of frames at resolution 640×480 stored in the YUV420 format and results on a frame of both are shown in Figure 3. As is clear from the figure, there is a significant color imbalance between the two camera views. We employ both our previous method [7] and the proposed method as well as histogram matching [3] in order to correct color differences between the selected camera views. In our implementation, we perform the computations separately on the three color channels in YUV444 space.

In Fig. 4 we show the SSIM metric computed between the color corrected frames and the original inputs as well as the PSNR values in the forward-reverse framework of Fig. 2. We can clearly see that the introduction of dense optical flow matching has not deteriorated the color correction algorithm. In fact, not much difference is visible in objective quality terms. However, it is important to point out that our previous method only relied on sparse feature matching and on histogram matching as a side information. By incorporating dense temporal matching by means of optical flow, we show that the proposed energy formulation and optimization method remain valid. In Fig. 5, the difference between the proposed method and our earlier work [7] is plotted. While the PSNR differences are inconclusive, the SSIM difference is overall positive however small. It is also worthy to point out that the SSIM values of both our methods that penalize large deviations from the original gradients remain more consistent over time compared to the classical histogram matching where there are much more fluctuations. Finally, the calculation of the optical flow does significantly increase the computational complexity. For a more detailed discussion about the complexity of our energy-based color correction framework, please refer to our previous work [7]. It is however important to point out that the proposed method is very flexible in terms of quality-complexity trade-offs as different components can be optimized individually and the choice for sparse of dense matching can be made in function of the target application.

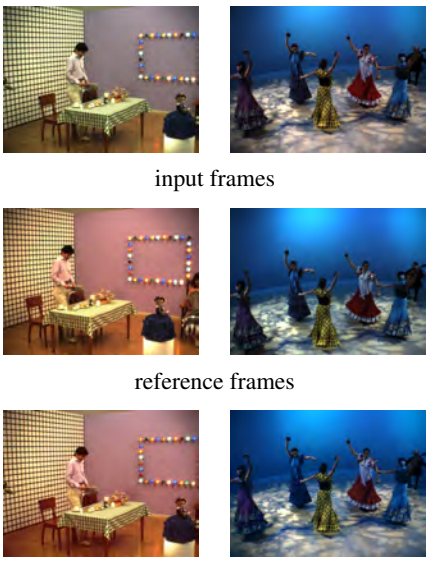
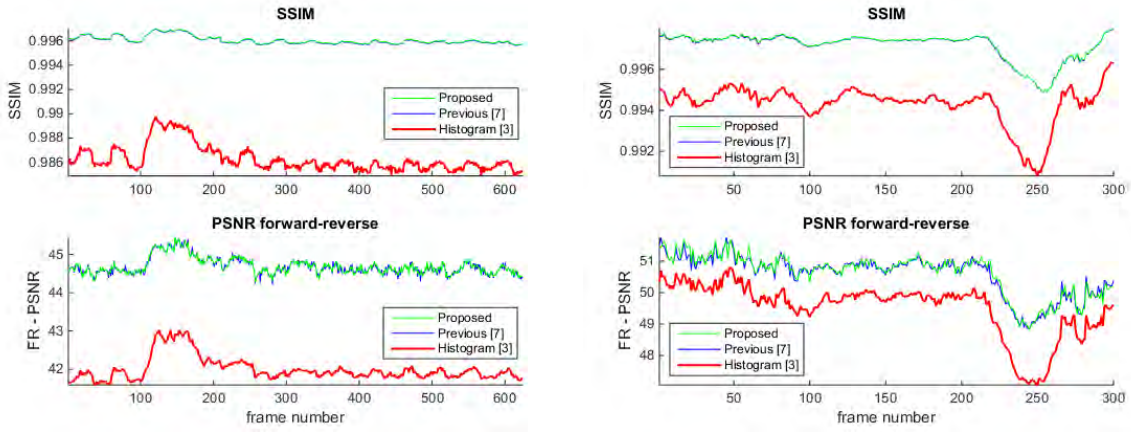


Figure 3. Visual results on the *Objects2* (left) and *Flamenco2* (right) multiview video sequences.

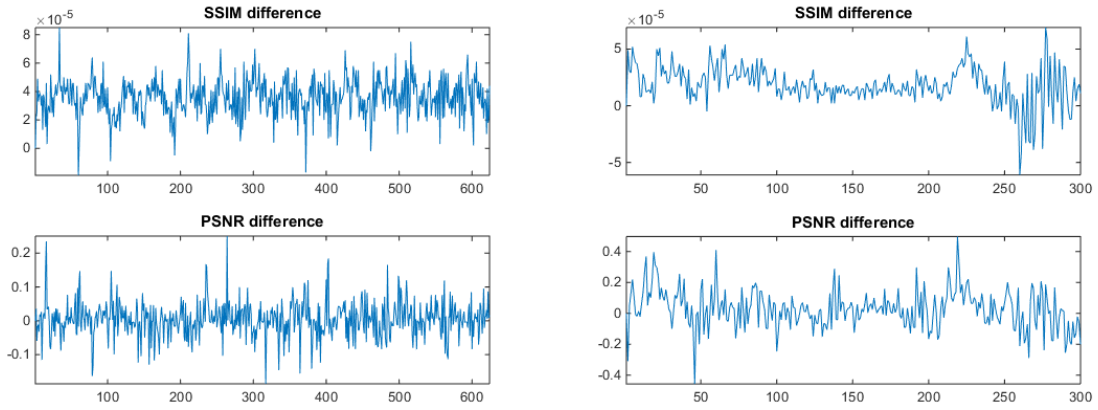
4. CONCLUSIONS

This paper introduces a novel method to correct for color differences in multi-camera setups. We demonstrate that the colors in an input video can be efficiently altered in order to match the colors of some reference camera while preserving the original local structure and texture information and without introducing artifacts. With this work we extend our previous method that performed sparse SURF matching followed by global optimization. We show that our framework can benefit from dense optical flow-based matching and in general that the quality of the color correction result can be further improved by providing as dense and as accurate correspondences as possible. Part of the research leading to this publication was performed in the High Tech Visualisation research program (HiViz) of iMinds.



(a) Objects2 (b) Flamenco2

Figure 4. Objective evaluation. The lower line (red) indicates the result from traditional histogram matching [3] while the two top lines (blue and green) are the results from [7] and the proposed method, respectively.



(a) Objects2 (b) Flamenco2

Figure 5. Difference plot.Because the lines in Figure (4) are too close to distinguish we also plot SSIM and PSNR differences between the proposed method with dense optical flow matching and our previous work [7].

5. REFERENCES

[1] J. Chen, J. Zhou, J. Sun, and A. C. Bovik, "Binocular mismatch induced by luminance discrepancies on stereoscopic images," in *International Conference on Multimedia & Expo*, 2014, pp. 1–6.

[2] M. Salmimaa, J. Hakala, M. Pölönen, T. Jävenpää, R. Bilcu, and J. Häkkinen, "Luminance asymmetry in stereoscopic content: Binocular rivalry or luster," in *SID Symposium Digest of Technical Papers*, vol. 45, 2014, pp. 801–804.

[3] U. Fecker, M. Barkowsky, and A. Kaup, "Histogram-based prefiltering for luminance and chrominance compensation of multiview video," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1258–1267, 2008.

[4] A. P. Hekstra, J. G. Beerends, D. Ledermann *et al.*, "PVQM - a perceptual video quality measure," *Signal Processing: Image Communication*, vol. 17, no. 10, pp. 781–798, 2002.

[5] S. A. Fezza, M.-C. Larabi, and K. M. Faraoun, "Feature-based color correction of multiview video for coding and rendering enhancement," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 9, pp. 1486–1498, 2014.

[6] Q. Wang, P. Yan, Y. Yuan, and X. Li, "Robust color correction in stereo vision," in *International Conference on Image Processing*, 2011, pp. 965–968.

[7] S.-P. Lu, B. Ceulemans, A. Munteanu, and P. Schelkens, "Spatio-temporally consistent color and structure optimization for multiview video color correction," *IEEE Transactions on Multimedia*, vol. 17, no. 5, 2015. doi: 10.1109/TMM.2015.2412879

[8] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *European Conference on Computer Vision*. Springer, 2006, pp. 404–417.

[9] M. Tao, J. Bai, P. Kohli, and S. Paris, "Simpleflow: A non-iterative, sublinear optical flow algorithm," in *Computer Graphics Forum*, vol. 31, no. 2.1. Wiley Online Library, 2012, pp. 345–353.

[10] D. Krishnan, R. Fattal, and R. Szeliski, "Efficient preconditioning of laplacian matrices for computer graphics," *ACM Transactions on Graphics*, vol. 32, no. 4, pp. 142:1–15, 2013.