Signal Processing: Image Communication 53 (2017) 40–50

Contents lists available at ScienceDirect

Signal Processing: Image Communication

journal homepage: www.elsevier.com/locate/image

Color correction for large-baseline multiview video

Siqi Ye, Shao-Ping Lu*, Adrian Munteanu*

Department of Electronics and Informatics, Vrije Universiteit Brussel, Pleinlaan 2, B-1050 Brussels, Belgium

ARTICLE INFO

Keywords:
Large-baseline cameras
Multiview video
Color correction
Local structure preservation

ABSTRACT

Color misalignment correction is an important, yet unsolved problem, especially for multiview video captured by large disparity camera setups. In this paper, we introduce a robust large-baseline color correction method that preserves the original manifold structure of the input video. The manifold structure is extracted by locally linear embedding (LLE), aimed at linearly representing each pixel based on its neighbors, assuming that they are all clustered in a high-dimensional feature space. Besides the proposed manifold structure preservation constraint, the proposed method enforces spatio-temporal color consistencies and gradient preservation. The multiview color correction solution is obtained by solving a global optimization problem. Thorough objective and subjective experimental results demonstrate that our proposed approach significantly and systematically outperforms the state-of-the-art color correction methods on large-baseline multiview video data.

1. Introduction

Multiview video acquisition becomes ubiquitous thanks to its ability of providing immersive information about the 3D scene and to the recent advances in visualization technologies (e.g. interactive free viewpoint television and autostereoscopic displays). Multiview video is the backbone for a vast amount of applications including virtual reality, digital entertainment, advertising, 3D graphics, broadcasting, video production, to name a few. In foreseeable future, multiview camera arrays will likely be small, *sparse*, and as easy to use as ordinary video cameras are nowadays. In this rapidly evolving area, research advances are needed in order to meet the severe demands concerning acquisition, representation, transmission and processing of multiview video.

While much work has been dedicated to this research domain, it remains very difficult to make large scale use of multiview video and to automatize production processes, as many steps still require the tedious manual work of skilled professionals.

One of the major problems encountered in multiview video acquisition systems is the color misalignment between sequences that are captured by different cameras. In addition to leading to visual fatigue and binocular rivalry [1,2], this problem does also further impede view synthesis, compression or other challenging multiview video processing techniques. Moreover, for *large-baseline* camera arrays, aligning the color of videos captured by such systems will be particularly difficult, due to the substantial variation of illumination conditions and surface properties in the scene.

Typical multiview color correction techniques are mainly based on histogram matching [3], with which both the style transfer and content-preservation are coarsely achieved. However, histogram-matching approaches suffer from introducing visual distortions: the color corrected results are much worse than using the original sequences as the color distribution of each view varies significantly in large-baseline multiview video.

Recent research focus, such as the work presented in [4], is shifting from processing globally accumulative mapping-based color transfer to constructing more complicated spatio-temporal mapping, which aims at preserving the gradient information to improve the color correction effect. However, as it will be demonstrated in this paper, the gradient information is far from sufficient to represent the detailed texture information in images. Furthermore, to the best of our knowledge, there is no solution in the literature specific for large-baseline multiview color correction, where the local structural information of each video should be well considered and preserved.

To address this problem, we present a manifold preservation approach for effectively correcting the color of large-baseline multiview videos. This paper is inspired by the recent progress on color correction [3,4] and manifold-based edit propagation [5,6]. In principle, we introduce the concept of robust color propagation in color-correction processing. Essentially, instead of requiring similar pixels to have similar results as usually enforced in histogram-based color correction methods, we aim at maintaining the manifold structure constructed by all pixels into a high-dimensional feature space. In order to better preserve the original structural information and propagate the aligned

* Corresponding authors.
E-mail addresses: siqi.ye@vub.ac.be (S. Ye), splu@etro.vub.ac.be (S.-P. Lu), acmuntea@etro.vub.ac.be (A. Munteanu).

http://dx.doi.org/10.1016/j.image.2017.01.004
Received 5 July 2016; Received in revised form 1 November 2016; Accepted 16 January 2017
Available online 24 January 2017

colors from the reference video, a manifold structure extraction and preservation constraint is proposed in our global optimization problem. Additionally, the proposed method offers spatio-temporal consistency and gradient preservation, similar to the approach in [4], effectively avoiding over-smoothing and visual artifacts, typical in manifold-based edit propagation.

In summary, the main contributions of this paper are as follows:

- We introduce a robust manifold preservation-based color correction method, specifically designed for large-baseline multiview video;
- We solve the color correction problem and yield a globally optimal solution;
- We provide objective and subjective evaluation results on various large-baseline multiview videos. Our manifold preserving color correction method outperforms all other existing approaches, demonstrating significant and systematic improvements over the state-of-the-art.

The paper is structured as follows. In the next section we relate our work to the current state-of-the-art in multiview video color correction and manifold-based edit propagation. Section 3 presents the proposed approach in detail. Experimental results, including critical implementation details, objective evaluation and a user study are given in Section 4. Finally, Section 5 summarizes the conclusions of this work.

2. Related work

Our work draws from two rich bodies of previous literature: multiview color correction and manifold modeling-based color processing. We thus review the related work in both areas.

Multiview color correction can be performed by hardware during acquisition (e.g. in [7,8]), as a prefiltering step after acquisition or even as part of the compression system. Here we focus on software-based techniques employed in the prefiltering or coding processes, which are most relevant to our work. Moreover, we review various key components involved in color correction, such as color compensation and correspondence matching.

One of pioneering methods of prefiltering is proposed by Fecker et al. [9]. In their method, histogram matching is applied by adapting the accumulative histogram of other camera views to that of the reference view. This approach was extended in [3] to improve the coding efficiency along with time-constant color conversion and global disparity compensation. Doutre et al. [10] calculate the average color of the matching points among all views and perform a least-squares regression for each view to set their color to be as close as possible to the average color. Kim et al. [11] consider focus correction in the color compensation processing. In [12], a block-based histogram matching method is proposed, whereby the blocks in reference frames are matched to those in target frames through spatial prediction. Yamamoto et al. [13] introduce lookup tables to dynamically correct the multiview color differences. Other lookup table designs include solutions based on dynamic programming [14] or polynomial basis [15,16].

Ilie et al. [7] construct a linear transformation matrix-based color modification solution. An over-determined linear system for color calibration is formulated by Li et al. in [17]. Other color correction solutions are based on linear scaling [18], pairwise basis functions [19] or high-order polynomials [10]. [20] also computes a local color transfer function by fixing a separate Gaussian distribution for each texture region. Similar processing making use of Gaussian filtering has been reported in [15].

However, most existing linear and nonlinear transformation methods easily suffer from artifacts, due to the difficulty in consistently mapping the color from the reference view to the target one in multiview video. To tackle this problem, Fezza et al. [21] applies histogram matching on a temporal sliding window and only common regions defined by an invariant feature detector are taken into account, where the temporal correlation is expected to be maintained. Nevertheless, local visual distortions in the textural structures may arise due to the inaccurate guidance of histogram matching. To address it, Lu et al. [4] proposed a color correction method enforcing spatio-temporally consistencies of color and structural information in multiview video. This method however cannot perfectly represent the local structure information in color correction processing, and, as shown in the experimental section, yields moderate results for large-baseline camera setups.

Correspondence matching is another key issue in multiview color correction. Typical sparse feature matching methods are employed to model this correspondence. For instance, Scale-invariant Feature Transform (SIFT) [22] is adapted in [15,17,21,23,24], while Speeded-up Robust Features (SURF) [25] has been used in [4,26]. Dense matching approaches were also introduced to improve the accuracy of frame-level correspondence matching. For example, a region matching method is used in [27], while disparity estimation is used to find corresponding points between views in [28]. In [23], the input image is segmented and corrected by locally-matched key points. We note that robust video segmentation is still an open issue in video processing. Block-level matching is also introduced in [29] by assuming a close similarity among spatio-temporally neighboring frames. However, block-matching correction methods would still result in outliers and color compensation based on them can easily suffer from blocking artefacts. To avoid the inaccuracies of sparse matching, Ceulemans et al. [26] further employed optical flow to construct spatio-temporal matching such that each pixel in the input image gets bijectively mapped to a reference pixel.

The proposed method is inspired from such full mapping-based nonlinear color correction. However, we focus more on preserving the high dimensional manifold structure originating from both the color and spatial texture of the input video, which makes our algorithm highly effective in aligning colors, particularly for large-baseline multiview videos.

The idea of preserving the local structure has also been well studied in a different context, namely color editing [30–32]. For instance, Qian et al. [33] utilized the manifold alignment concept to preserve the local geometries of color distribution and to align corresponding pixels for image stitching. Xu et al. [34] performed edit propagation on bidirectional texture functions in order to preserve the view independence features. SmartColor [35] is introduced to blend the target color to the background for head-mounted displays. More robust color distance metrics have great potentials in improving the visual perception of color editing. One example is geodesic distance based color harmonization [36]. Chen et al. [5] employed LLE [37] to represent the manifold structure and to propagate edits specified by the user. Ma et al. further proposed a manifold-preserving color image editing approach in [6]. In this method, the authors adaptively determine the size of pixel neighborhoods, which are used to represent the manifold structure of each pixel. We note that all these manifold-preserving methods focus on interactive *color editing* for single image or video, while our solution addresses adaptive *color correction* for multiview video.

3. Proposed algorithm

3.1. Overview

The proposed multiview color correction algorithm aims at automatically propagating the reference colors to the corresponding pixels in the target frames. To avoid inaccurate propagation that may result in texture distortion, the local structure information should be well preserved. Accordingly, in order to robustly perform color correction for *large-baseline* multiview video, the local manifold structure is of particular importance, and preserving this structure becomes a key constraint in our global optimization framework.

In essence, our global optimization framework comprises three constraints, namely, preserving the local manifold structure, spatio-temporal consistency and the original gradient information. We describe them in detail next.

3.2. Manifold structure preservation

Let I_t be the t*th* video frame whose color needs to be corrected, and denote by $I_t(x)$ the pixel with coordinate x in frame I_t. As introduced in [37], the data in a high dimensional space can be projected to a lower dimensional *manifold* where each sample can be represented by a linear combination of its nearest neighbors. Similarly, the color of each pixel in the image can also construct a local manifold structure with its neighbors in the feature space [5,6]. Thus, to obtain a robust manifold structure of I_t, a proper set of adjacent pixels should be selected for each pixel $I_t(x)$ from the original frame. In our solution, this is tackled by k-d tree clustering [38] operating in a five-dimensional feature space composed of three color channels, i.e. the red, green and blue values, and two spatial position indicators m and n. Then, the original manifold structure of each color frame is obtained by minimizing the following equation:

$$\min \sum_{x=0}^{N-1} \| I_t(x) - \sum_{k=0}^{K-1} \omega_k(x) I_{t,k}(x) \|^2 \text{ subject to } \sum_{k=0}^{K-1} \omega_k(x) = 1. \tag{1}$$

In this equation, $I_{t,k}(x)$ is the k*th* adjacent pixel of $I_t(x)$, N is the total number of pixels in the frame and K is the number of neighbors in the support region for the pixel $I_t(x)$. One notes that the weights $\omega_k(x)$ satisfy a normalization constraint. After establishing the appropriate neighborhood in the cluster, the weights $\omega_k(x)$ are computed using the LLE algorithm [39].

Thus, in the first phase, we get the original local manifold structure including each pixel's support neighbors and their corresponding weighting factors. In order to enforce manifold structure preservation in multiview color correction, the weighting factors and the local neighborhood of each pixel in the color corrected picture should be preserved after performing color correction. Hence, the following energy component should be minimized:

$$E_m = \sum_{x=0}^{N-1} \| O_t(x) - \sum_{k=0}^{K-1} \omega_k(x) O_{t,k}(x) \|^2, \tag{2}$$

where $O_t(x)$ represents the color at x in the *corrected* video frame, and $O_{t,k}(x)$ is the corrected color of k-*th* adjacent pixel in the same frame. Note that in this equation the local neighborhood and the weights $\omega_k(\cdot)$ are those obtained on the *original* frame from the minimization of Eq. (1).

Note that Eq. (2) can be further written into the matrix form:

$$E_m = O_t^T (I - W)^T (I - W) O_t, \tag{3}$$

where O_t is a $N \times 1$ vector composed of all pixels in the corrected video frame, I is a N-dimensional identity matrix and W is a $N \times N$ matrix composed of weights $\omega_k(\cdot)$.

3.3. Spatio-temporally consistent multiview color correction

Spatio-temporal consistency preservation. We follow the assumption, as introduced in [28,21,4], that similar content should share similar color in the scene. It is thus necessary to construct spatio-temporal consistency terms based on both the input and reference views in order to guide the color correction process.

Enforcing spatio-temporal consistencies takes into account the following three sub-constraints: (1) the spatial correspondence between the input video frame and the corresponding one in the reference video; (2) the temporal correspondence between the input video frame and its previous frame; and (3) the additional color correspondence by histogram matching. Therefore, to preserve spatio-temporal consistencies, the energy in the following equation should be minimized:

$$E_c = \beta_{ct} \sum_{x \in \Omega_r} \| O_t(x) - R_t(f_r(x)) \|^2 + \beta_{cp} \sum_{x \in \Omega_p} \| O_t(x) - O_{t-1}(f_p(x)) \|^2 + \beta_{ch} \sum_{x \in \Omega_h} \| O_t(x) - H_t(x) \|^2, \tag{4}$$

where β_{ct}, β_{cp} and β_{ch} are corresponding weighting factors of the above-mentioned sub-constraints. Ω_r is the domain composed of the matched feature points between the current frame in the input video and the corresponding frame in the reference video. Similarly, Ω_p represents the collection of matching points between the current frame and the previous (corrected) frame. All other pixels in the input frame that belong to neither Ω_r nor Ω_p are included then in Ω_h. $f_r(\cdot)$ and $f_p(\cdot)$ are the position mapping functions which indicate the positions of matching points in the reference frame and in the previous frame, respectively. R_t represents the t-*th* frame in the reference video and H_t is the histogram matching result of I_t with R_t. Unlike other existing color correction methods that usually employ SIFT or SURF to find the matching features, we use the detector introduced in [40,41] because of its more accurate and denser matching performance (more details are reported in their papers).

E_c can be represented in the matrix format by combining the three feature terms into a vector M and composing their corresponding weights into a diagonal matrix A:

$$E_c = (O_t - M)^T A (O_t - M), \tag{5}$$

Gradient preservation. Color correction should follow the principle that the structural information in the original scene should be preserved as much as possible. Similar to the modeling in [4], this is formulated by minimizing the gradient difference between the output frame and the input frame. Mathematically, this is expressed by minimizing:

$$E_r = \sum_{x=0}^{N-1} \| \nabla O_t(x) - \nabla I_t(x) \|^2, \tag{6}$$

where ∇ is the gradient operator for which the horizontal and vertical gradient components are computed using the Sobel operator. E_r can be further represented as the matrix form:

$$E_r = (D_x O_t - D_x I_t)^T (D_x O_t - D_x I_t) + (D_y O_t - D_y I_t)^T (D_y O_t - D_y I_t), \tag{7}$$

in which D_x and D_y denote the horizontal and vertical gradient components respectively.

Solution. Combining the above-mentioned three constraints, the optimized color correction result can be solved by minimizing the following total energy:

$$E = \beta_m E_m + \beta_c E_c + \beta_r E_r, \tag{8}$$

where β_m, β_c and β_r are the weighting factors for the corresponding energy terms.

The optimal color correction solution can be found by calculating derivatives and solving the following linear system of equations:

$$[\beta_m (I - W)^{\mathrm{T}} (I - W) + \beta_c A + \beta_r (D_x^{\mathrm{T}} D_x + D_y^{\mathrm{T}} D_y)] O_t = \beta_c A M + \beta_r (D_x^{\mathrm{T}} D_x + D_y^{\mathrm{T}} D_y) I_t. \tag{9}$$

This equation involves a sparse Laplacian matrix (the left-hand term multiplying O_t) which can be solved efficiently using the Hierarchical Sparsification and Compensation (HSC) method introduced in [42].

3.4. Analysis

In the proposed approach, the manifold preservation constraint includes the local structure preservation energy given by Eq. (2) together with the gradient preservation constraint, expressed by Eq. (6). This constraint is much stricter than just enforcing gradient preservation, as done in our previous method in [4]. The local manifold

structure intrinsically describes the local color correlation (weighted linear combination in this paper) between each pixel and its nearest pixels, while the original gradient captures just the color differences between each pixel and its four neighbors in the horizontal and vertical directions, respectively. In the extreme case, if the neighboring radius of the local manifold structure is equal to 1, the support domain in the manifold will be drastically regressed. In this case, the contribution of the manifold preservation constraint in the total energy given by Eq. (8) will be similar to that of the gradient preservation constraint.

On the other hand, the colors will not be well corrected when performing our solution with too little (or even without) the gradient preservation energy constraint (see the comparison in Section 4). One notes that, to capture the relation between each pixel and its neighbors in the manifold structure, not only the two spatial distances but also three color components are taken into account. Thus, around or on image contours, where each pixel's color may be much different from its spatial neighbors in the image, the local manifold structure will not be able to capture the pixel's relation with its spatial neighbors. In contrast, the simple, yet accurate gradient information can effectively build the bridges for pixels located on or very close to image contours. This is the reason why we introduce the combination of manifold preservation and gradient preservation constraints in our approach.

Note that in [4], the additional histogram matching-based sub-term is used to ensure a one-to-one color mapping for *all* input pixels in the spatio-temporal consistency term; in this way, [4] yields a *closed-form* solution to the color correction problem. In contrast, our algorithm can still yield a color corrected image without this additional sub-term. Nevertheless, if we do so, i.e, when only considering the manifold structure constraint E_m, together with gradient preservation E_r, and the spatio-temporal consistency constraint E_c but only based on the sparse feature points, the proposed solution turns out to be inadequate for propagating the corrected colors to the entire image, especially for flat areas where normally no feature points are detected (see the detailed comparison in Section 4). Therefore, the histogram matching component is still necessary as an additional reference for those pixels for which the sparse spatio-temporal feature matching fails to find appropriate correspondences.

4. Experimental results

The proposed method is experimentally evaluated using various multiview video sequences with large camera baselines. As reference techniques we report color correction results using three different algorithms, namely, the histogram matching method [3] (with the abbreviation of *Histogram*), the algorithm presented in [4] (denoted by *TMM*), and our proposed algorithm (denoted by *Proposed*). To compare the results of such different approaches, the objective evaluation is investigated in terms of the Peak Signal-to-noise Ratio (PSNR) and the Structural Similarity (SSIM) [43] metrics, which are normally used for image quality evaluation. We also perform a user study to subjectively assess the color correction results.

The proposed color correction algorithm has been applied to various standard multiview sequences, including the *Objects2*, *Race1* and *Crowd* as well as two uncalibrated sequences shot in our lab, namely *Etroshot* and *Beers*.

The input and reference views are chosen so as to have a large baseline between the view pairs. Specifically, for the standard sequences employed in this paper, view pairs with large baselines are selected according to the sequence description[1] and our visual perception. For the *Etroshot* and *Beers* sequences, the view angle difference is around 20 degrees between camera positions. Detailed parameters of the test view pairs are listed in Table 1. In this table, I_t and "R_t" represent the view numbers of the input and the reference cameras, respectively. "BL" is the distance between the input and the reference view, and "Frm" indicates the number of frames for each video sequence.

Table 1
Average numerical results of forward-reverse correction.

Video	Size	View Parameters ($I_t \rightarrow R_t$, BL, Frm)	Metrics	Histogram [3]	TMM [4]	Ours
Objects2	640×480	(0 → 7, 140 cm, 300)	PSNR	23.86	27.47	**28.50**
			SSIM	0.845	0.951	**0.971**
Crowd	640×480	(0→4, 40 cm,300)	PSNR	38.69	41.34	**43.50**
			SSIM	0.981	0.992	**0.996**
Race1	640×480	(0 → 7, 140 cm, 300)	PSNR	35.26	36.66	**41.96**
			SSIM	0.947	0.967	**0.997**
Etroshot	960×540	(−→−,20°, 150)	PSNR	35.53	37.27	**37.65**
			SSIM	0.975	0.989	**0.993**
Beers	960×540	(−→−,20°, 150)	PSNR	29.98	31.77	**33.05**
			SSIM	0.941	0.970	**0.981**

4.1. Implementation parameters

Our color correction system is implemented in Microsoft Visual Studio C++2013 combined with the implementation of Laplace matrix optimization in MATLAB 2012 on a PC equipped with an Intel Xeon CPU 3.5 GHz processor and 64 G RAM.

As described in Eq. (9), in our system there are three categories of weights associated with the manifold structure, spatio-temporal consistency and gradient preservation constraints, respectively. Since the spatio-temporal consistency constraint consists of three sub-terms, the weight for spatial sparse correspondence sub-term is empirically fixed as 600, while the weight assigned to temporal matching sub-term is attenuated from 600 by the factor of $\frac{N_{pre}}{N_{ref}}$, where N_{ref} and N_{pre} represent the numbers of the spatial feature points and of the temporal feature points respectively. The additional correspondence sub-term with histogram matching points is weighted by 300. By fixing all the above-mentioned parameters, more detailed investigations are then focused on selecting the weights of the manifold structure and gradient preservation terms. Fig. 1 shows the experimental results on the 2*nd* frame of the *Objects2*. In this figure, the manifold preservation energy is plotted in logarithm scale. For each chosen parameter assigned to the manifold preservation term, when the weight of the gradient preservation term is gradually increased, the PSNR firstly increases to a peak value and then goes down, while the SSIM keeps going up. One can tell from this figure that emphasizing too much on the gradient term would well preserve the original spatial structure of the image, but the reconstructed color may be distorted (see the decreased PSNR). On the contrary, too high weighting on the manifold preservation term will impair the performance of both the PSNR and SSIM. This phenomenon verifies the analysis in Section 3.4. Thus, based on extensive experiments, we set the final weights for the manifold structure term and the gradient preservation term to 100 and 5, respectively. We thus set all parameters to fixed values and use them for all test sequences in our experiments.

The execution time for correcting one frame is mainly spent on the following processing steps: 1) clustering using k-d tree and retrieving neighboring pixels under this tree; 2) calculating LLE for each color component, 3) performing feature matching; and 4) solving the final optimization problem. The total runtime cost of the proposed approach

[1] ftp://vqeg.its.bldrdoc.gov/Documents/Meetings/ITUT_SG9/May04/T01-SG09-040510-TD-GEN-0060!!MSW-E.doc

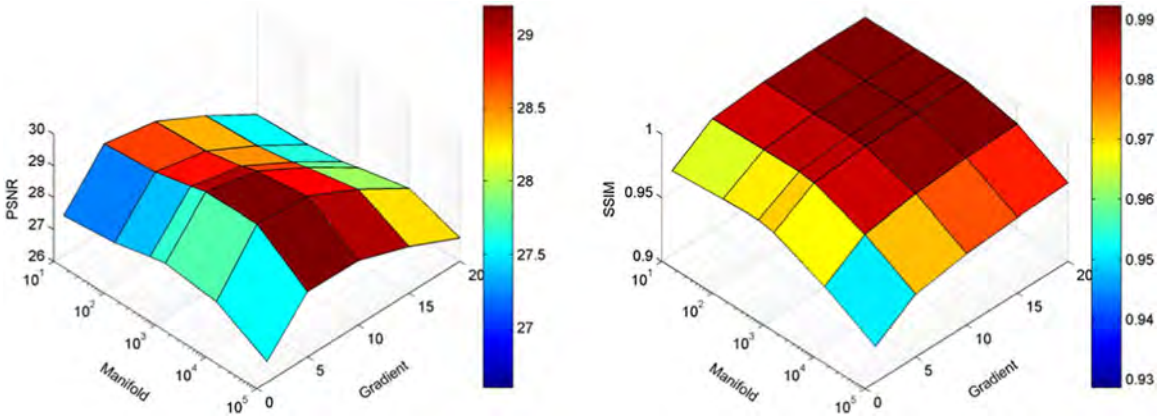

Fig. 1. Parameter selection for the proposed system.

for a video frame with HD resolution is about 20 s in release mode on a single CPU core. One points out that, in our approach, solving the sparse Laplacian linear system shares similar computational complexity with the method introduced in [4], where the runtime cost is about 8 s for a video frame with HD resolution. The computational consumption on the first step, i.e. pixel clustering and retrieving, is close to that of the final solver (i.e. about 8 s), while the second and third processing steps need much less time (about 4 s). As to the memory consumption, it takes 236MB for pixel clustering and retrieving, 600MB for solving the LLE problem, and around 860 MB to solve the sparse Laplacian matrix system for an HD frame.

4.2. Color correction results

The results obtained on the publicly available video sequences are shown in Figs. 2, 3 and 4. In Fig. 2, it is obvious that [3] easily produces visual artifacts on the blocks (see the 3rd row) and on the flowerpot (the 4th row). [4] eliminates part of the artifacts, but it still introduces a shadow at the right bottom of the blocks (the 3rd row) which does not appear in the source image, and the color transition on the flowerpot is also not smooth enough. Comparatively, the proposed result corrects the color through the whole image but still retains the original structure such as shadowing and transitions. The result using our proposed algorithm is more reddish which is more close to the reference compared to those obtained using [3,4], especially on the wall. Therefore, the proposed result can be considered as the best among the tested techniques on this sequence.

Similarly, in Fig. 3, one can see that the results of [3,4] are too yellowish compared to the reference as well as the source, while the color palette of proposed result looks much closer to that of the reference. In Fig. 4, the reference frame looks reddish but the source frame that needs to be color-corrected looks yellowish. However, the corrected results using [3,4] are too yellow which are again far from not only the source but also the reference. For the proposed result, it is obvious that it appears more reddish compared to the source, and it is the most similar to the reference in color compared with the results obtained using [3,4]. To show the color consistency between neighboring frames, we provide several successive video frames of the corrected sequence *Objects2*, their corresponding original frames as well as reference frames (see Fig. 5).

We also investigate the performance of the proposed method obtained when removing part of the proposed constraints (see the visual results in Fig. 6). First, we apply our solution when only using the manifold preservation and spatio-temporal consistency constraints that are based just on the sparse feature points, while for the manifold structure we perform clustering only considering each pixel's spatial distance (instead of with its 5D features). As shown in Fig. 6 (a), the colors are badly spatially propagated due to the lack of color correlation in the manifold construction. In contrast, in Fig. 6 (b) the output image is somewhat improved when the original RGB color information is involved in the manifold structure construction. However, it is noticeable that in some areas the textural information is not well preserved, and overall, the results are far from acceptable. As discussed in Section 3.4, if only the manifold structure is considered to represent the local structure information, the system always intends to generate an over smoothing effect. We also perform our algorithm by just removing the gradient preservation constraint, the results being shown in Fig. 6 (c). It is obvious that, while the color palette starts to become acceptable, there are some significant reconstruction errors that render the overall result as far from acceptable. The last image in this figure is generated when we remove the histogram matching-based sub-term in the spatio-temporal consistency constraint. Again, as described in Section 3.4, without this additional sub-term, it is still difficult for the system to propagate all the reference colors to the output image, although the contours and important structures are well reconstructed. Investigation has also been done involving Lab color space instead of RGB color space in the k-d tree clustering and retrieving step described in Section 3.2. The corresponding results will be discussed in Section 4.3.

4.3. Evaluation and comparison against the state-of-the-art

The purpose of this paper is to render the color of input view to be as similar as possible to that of the reference view. This turns out to be a particularly difficult problem when the disparity between the two views is large. The objective evaluation of the proposed method is based on the methodology proposed in [4]. In this methodology, the color distortion for the proposed and reference color-correction methods is determined. Additionally, we also follow a user study to subjectively assess the color correction results.

Objective evaluation. Objective metrics such as the PSNR and SSIM are commonly used to measure the quality of the reconstructed image, while they cannot be directly applied in multiview color correction due to the lack of ground truth. To address this problem, we employ the forward-reverse assessment methodology introduced in [4]. In this methodology, forward color correction is firstly performed, where the input color is corrected according to the reference view. Reverse color correction is subsequently performed with the same method as used in forward correction, but the reference is the original input view and the input is the result of forward correction [4]. After that, the PSNR and SSIM are computed between the output of reverse color correction and the original input frame which serves as ground truth.

This forward-reverse evaluation methodology is also performed for [3,4] to compare their results against those obtained with the proposed approach. Detailed average numerical results over Red, Blue and Green color components are shown in Figs. 7 and 8. These averages are then

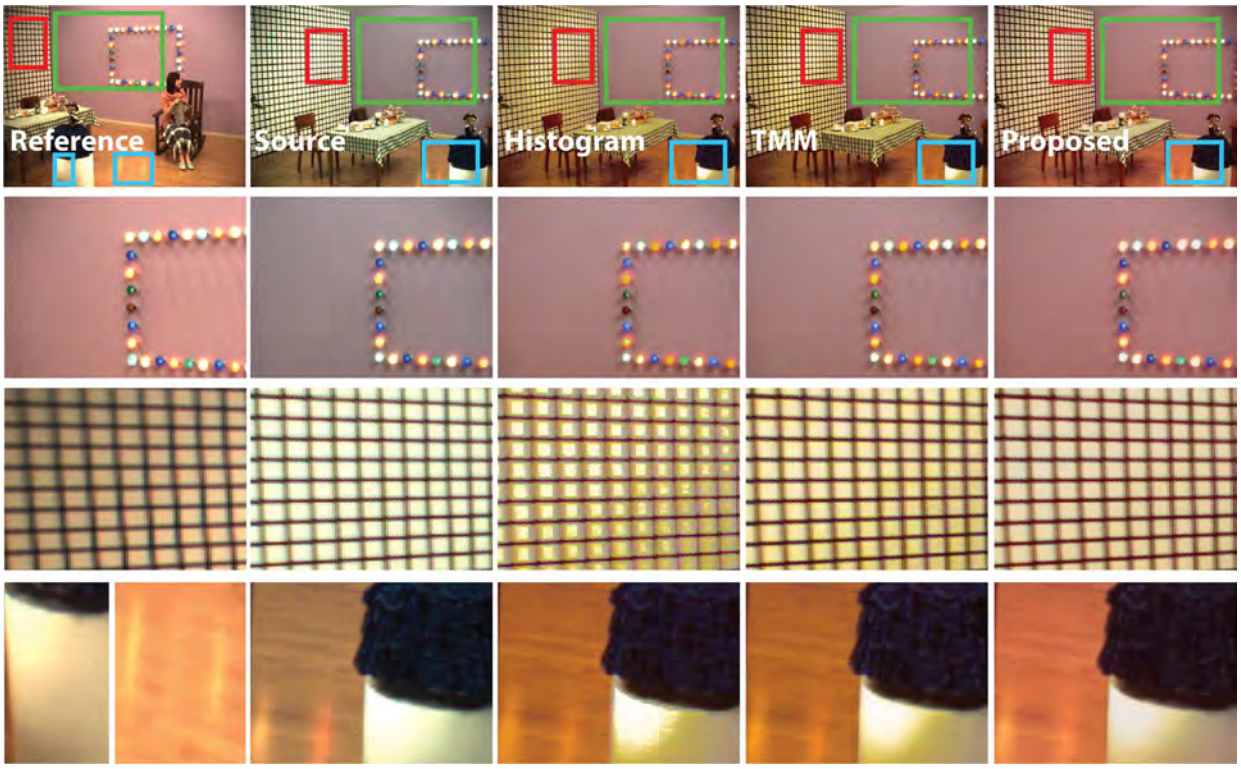

Fig. 2. Color corrected result of the *Objects2* sequence (7th frame). Source: the input view; Histogram: the result with [3]; TMM: the result with [4]; Proposed: the result with our method.

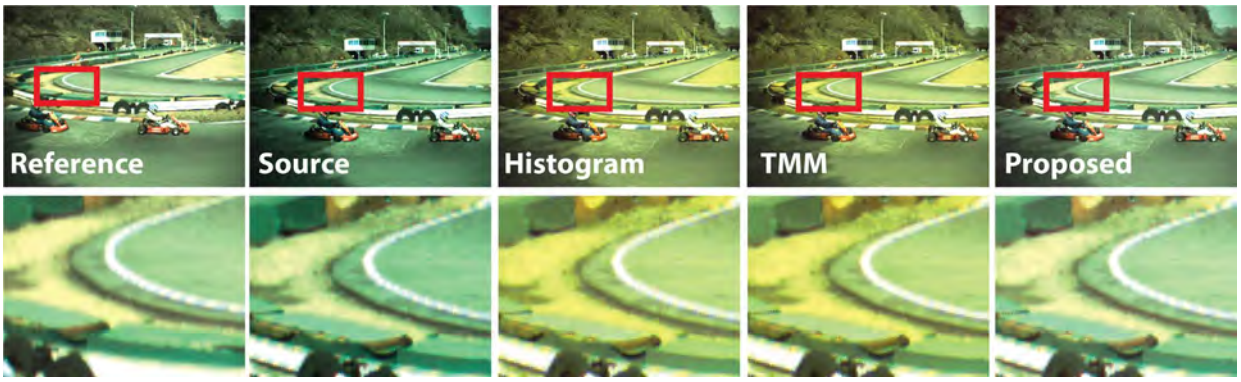

Fig. 3. Color corrected result of the *Race1* sequence (28th frame). Source: the input view; Histogram: the result with [3]; TMM: the result with [4]; Proposed: the result with our method.

Fig. 4. Color corrected result of the *Crowd* sequence (60th frame). Source: input view that color correction need to be performed; Histogram: result with [3]; TMM: result with [4]; Proposed: result with our method.

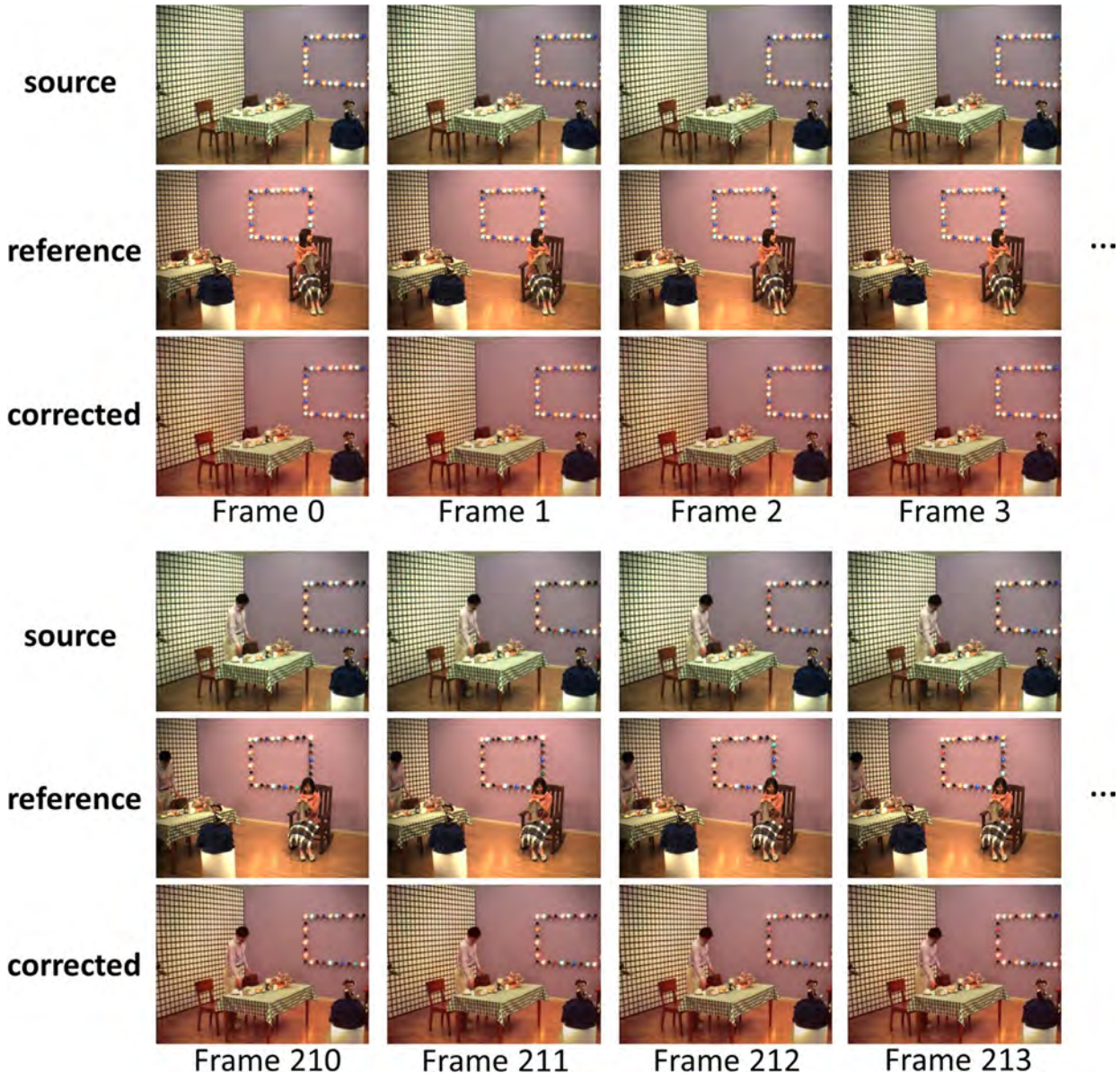

Fig. 5. Successive frames showing the color consistency of the source video and the corrected results of sequence *Objects2* using our proposed algorithm.

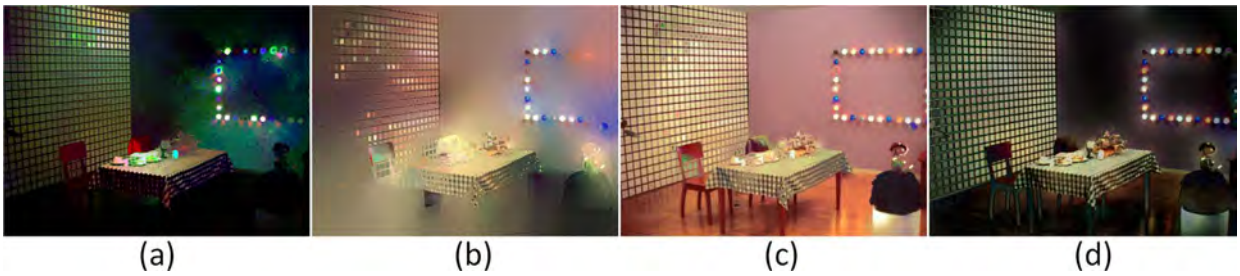

Fig. 6. Our color correction results obtained when removing part of the proposed constraints. (a): results when only considering the manifold preservation and two sparse matching-based consistency sub-terms, but in the manifold only each pixel's spatial information is considered (i.e. no color information is accounted for). (b): the same constraints as in (a), while the manifold is built using the proposed 5D feature space. Results obtained (c): without considering the gradient preservation constraint, and (d): without the additional histogram matching-based sub-term.

averaged over the number of corrected frames and the results for each sequence are shown in Table 1. With the proposed method, the minimal gain in terms of average PSNR over the number of frames is +2.12 dB compared to [3] and +0.38 dB compared to [4] (sequence *Etroshot*). The maximal gain can be up to +6.7 dB against [3] and +5.3 dB against [4] (sequence *Race1*).

In terms of the average SSIM over the number of frames, which is more appropriate than PSNR to evaluate texture structure information, the proposed algorithm gains range from +1.5% (*Crowd* sequence) to +12.6% (*Objects2* sequence) when compared to [3], and from +0.4% (*Crowd* and *Etroshot* sequences) to +3% (*Race1* sequence) in comparison to [4].

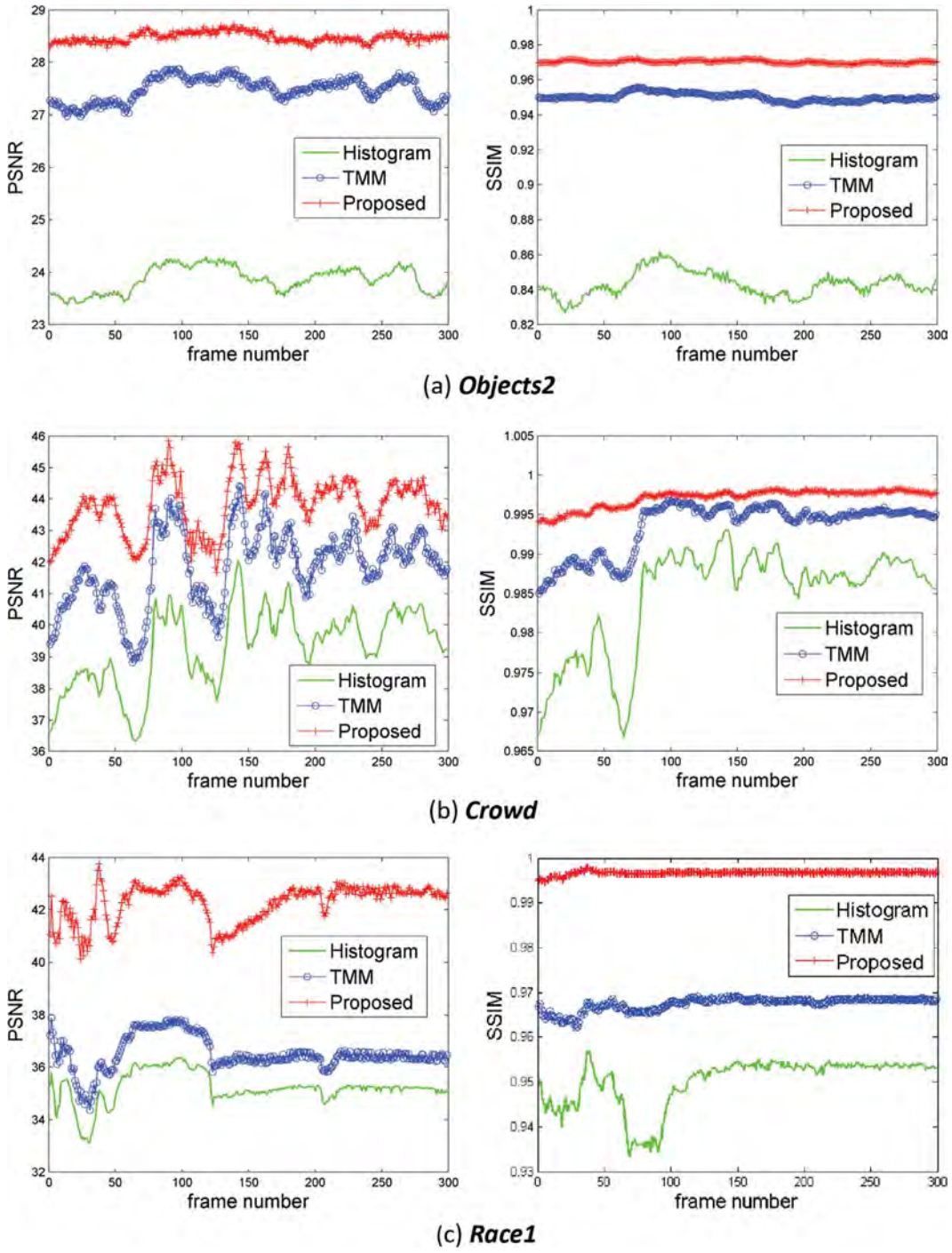

Fig. 7. Forward-reverse color correction results obtained on the *Objects2*, *Crowd* and *Race1* sequences. Histogram: the result with [3]; TMM: the result with [4]; Proposed: the result with our method.

To illustrate the effectiveness of our proposed algorithm for correcting colors of large-baseline multiview videos, experiments have also been performed under various baseline distances, and the examples are shown in Fig. 9. It is obvious that the color correction performance of [3] dramatically is degraded when the input and reference cameras are far away, while our proposed algorithm is able to achieve much better results under large baseline distances.

As mentioned in Section 4.2, different color spaces involved in the k-d tree clustering and retrieving step for preserving manifold structure have been investigated. Fig. 10 illustrates the results with Lab color space and with RGB color space, respectively. Experimental results using algorithms in [3,4] are also shown in this figure. Our results are surprisingly the best in the RGB color space instead of the Lab color space, although the latter is usually regarded as a linear space. The main reason would be that under RGB color space the linearly weighting of the correlated high-dimensional features are more suitable for the k-d tree clustering, local manifold extraction and structure preservation optimization in our solution.

User Study. In addition to quantifying the PSNR and SSIM using the forward-reverse methodology of [4], we have also performed a user study to evaluate the color correction obtained with the proposed system. The subjective evaluation has been performed in a standard-compliant Rec. ITU-R BT 500-13 visual-quality test facility, employed in classical still-image evaluations. The screen used in the subjective

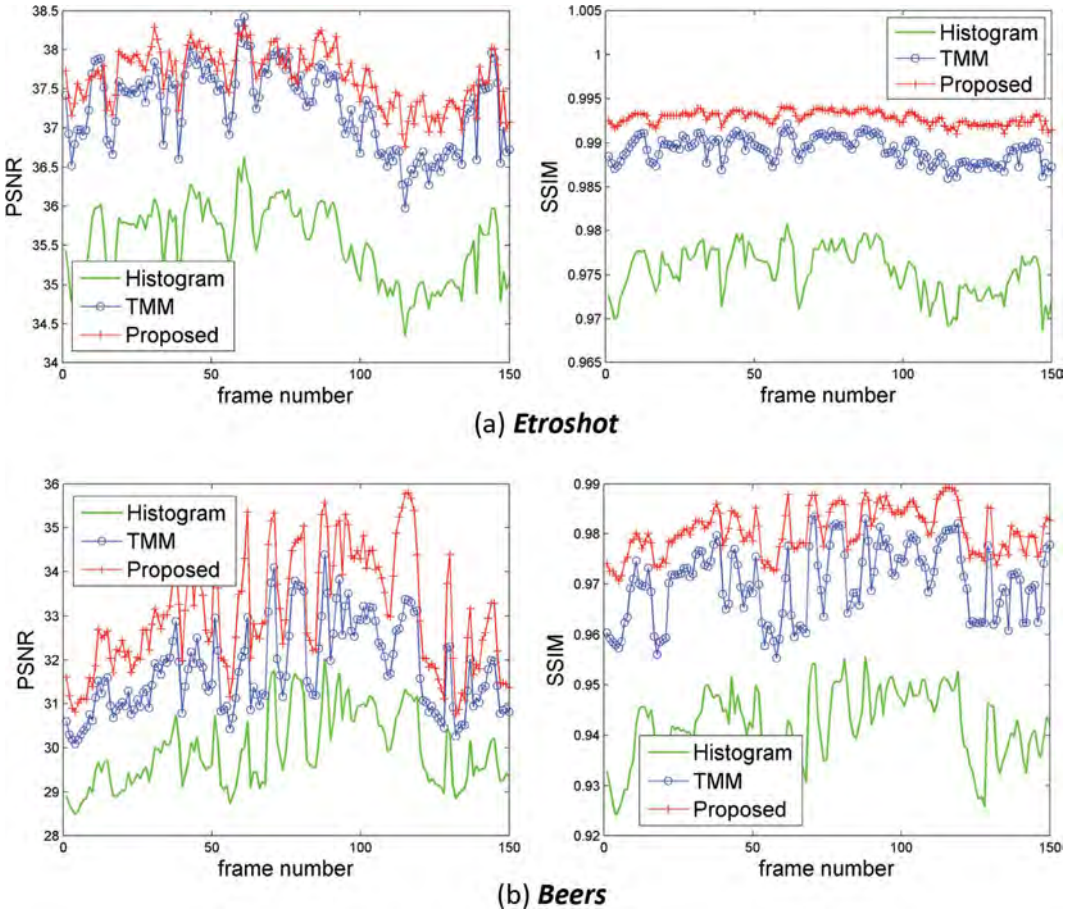

Fig. 8. Forward-reverse Correction of the *Etroshot* and *Beers* sequences. Histogram: the result with [3]; TMM: the result with [4]; Proposed: the result with the proposed method.

assessment is a color-calibrated HDTV with a 16/9 format ratio. The viewing distance was three meters and the maximum observation angle was around 30 degrees. We invited fifteen participants to the visual tests, five of them having professional background in computer vision or computer graphics and the remaining having little or no experience in these fields. The subjects presented no acuity or color deficiencies. Again, five sequences, i.e. *Objects2*, *Crowd*, *Race1*, *Etroshot* and *Beers*, were shown to each participant one by one. For each sequence, the reference view as well as three output views obtained using the three aforementioned color correction methods were shown at the same time. Participants were asked to vote which output has the most similar colors to that of the reference view.

The voting result is surprisingly convincing: the proposed method is always selected except of only two participants outside this field preferring the result of [4] in the *Objects2* sequence. This user study demonstrates that our proposed algorithm is substantially preferred over the state-of-the-art.

5. Conclusions

In this paper, we present a color correction method for large-baseline multiview video. Color correction is formulated as a global optimization problem whereby spatio-temporal color consistencies and gradient preservation are enforced. A major contribution of the proposed approach lies in incorporating a manifold-preservation constraint extracted from the original video frames into the global optimization problem. Both objective metrics and subjective evaluation results show that our proposed approach outperforms the state-of-the-art.

There are several avenues for future work and applications. One of the future research directions may focus on adapting our algorithm to various color spaces in order to achieve more robust color correction results. Improving computational performance by exploring multicore- and GPU-based implementations of the proposed method would be another research topic. In addition, it would be interesting to incorpo-

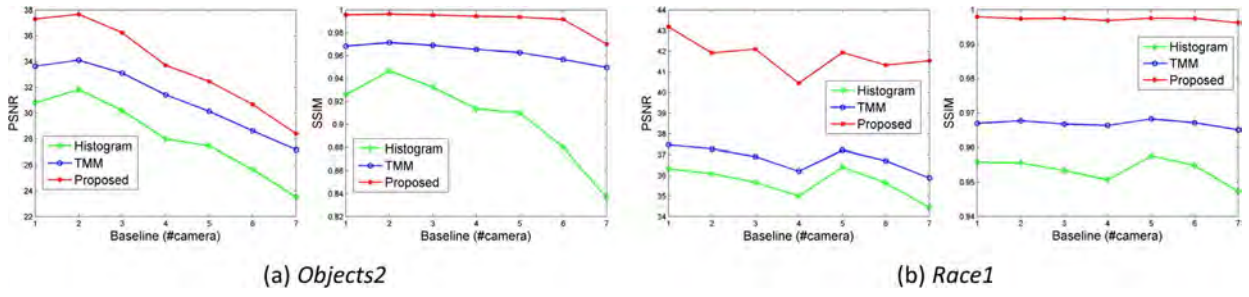

Fig. 9. Objective evaluation under different baseline distances. The *x*-axis indicated by the camera numbers represents the baseline distance and the *y*-axis indicates the values of PSNR and SSIM.

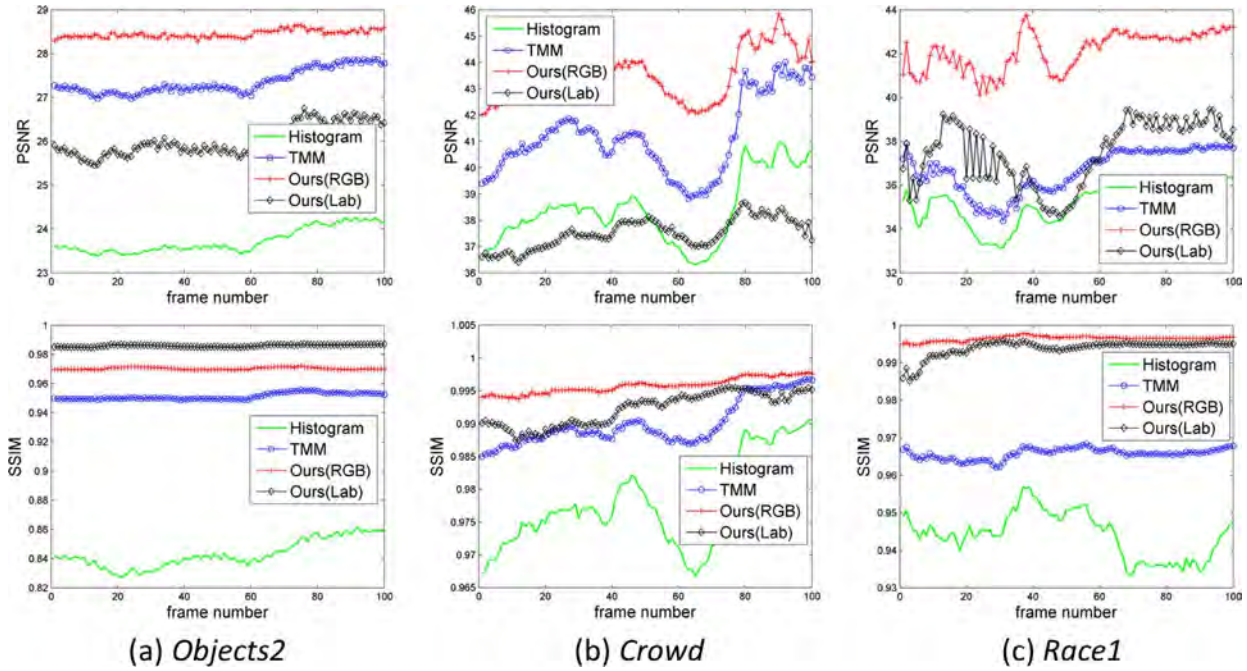

Fig. 10. Forward-reverse color correction results obtained on the *Objects2*, *Crowd* and *Race1* sequences. Histogram: the result with [3]; TMM: the result with [4]; Ours(RGB): the result with proposed algorithm with k-d tree clustering and optimally preserving the manifold structure in the RGB space; Ours(Lab): the result with proposed algorithm with k-d tree clustering and optimally preserving the manifold structure in the Lab space.

rate our color correction method into virtual view synthesis (e.g. [44–46]), multiview video compression and other immersive applications.

Acknowledgements

The authors would like to thank the anonymous reviewers and the editor for their insightful comments and suggestions which have improved this paper. The work in this paper has been supported by VUB (SRP), Innoviris (3DLicornea) and FWO (G.0256.15).

References

[1] J. Chen, J. Zhou, J. Sun, A.C. Bovik, Binocular mismatch induced by luminance discrepancies on stereoscopic images, in: Proceedings International Conference Multimed. Expo., IEEE, 2014, pp. 1–6.
[2] M. Salmimaa, J. Hakala, M. Polonen, T. Jarvenpaa, R. Bilcu, J. Hakkinen, Luminance asymmetry in stereoscopic content: Binocular rivalry or luster, in: SID Symposium Digest of Technical Papers, Vol. 45, Wiley Online Library, 2014, pp. 801–804.
[3] U. Fecker, M. Barkowsky, A. Kaup, Histogram-based prefiltering for luminance and chrominance compensation of multiview video, IEEE Trans. Circuits Syst. Video Technol. 18 (9) (2008) 1258–1267.
[4] S.-P. Lu, B. Ceulemans, A. Munteanu, P. Schelkens, Spatio-temporally consistent color and structure optimization for multiview video color correction, IEEE Trans. Multimed. 17 (5) (2015) 577–590.
[5] X. Chen, D. Zou, Q. Zhao, P. Tan, Manifold preserving edit propagation, ACM Trans. Graph 31 (6) (2012) 1–7.
[6] L.-Q. Ma, K. Xu, Efficient manifold preserving edit propagation with adaptive neighborhood size, Comput. Graph. 38 (2014) 167–173.
[7] A. Ilie, G. Welch, Ensuring color consistency across multiple cameras, in: Proceedings International Conference Comput. Vision, Vol. 2, Beijing, China, 2005, pp. 1268–1275.
[8] J. Zhong, B. Kleijn, X. Hu, Camera control in multi-camera systems for video quality enhancement, IEEE Sens. J. 14 (9) (2013) 2955–2966.
[9] U. Fecker, M. Barkowsky, A. Kaup, Improving the prediction efficiency for multi-view video coding using histogram matching, in: Picture Coding Symposium, 2006, pp. 1–5.
[10] C. Doutre, P. Nasiopoulos, Color correction preprocessing for multiview video coding, IEEE Trans. Circuits Syst. Video Technol. 19 (9) (2009) 1400–1406.
[11] W. Kim, J. Kim, M. Choi, I.-J. Chang, J. Kim, Low complexity image correction using color and focus matching for stereo video coding, in: Proceedings International Symp. Circuits Syst., Beijing, China, 2013, pp. 2912–2915.
[12] B. Shi, Y. Li, L. Liu, C. Xu, Block-based color correction algorithm for multi-view video coding, in: Proceedings International Conference Multimedia and Expo, IEEE, 2009, pp. 65–68.
[13] K. Yamamoto, M. Kitahara, H. Kimata, T. Yendo, T. Fujii, M. Tanimoto, S. Shimizu, K. Kamikura, Y. Yashima, Multiview video coding using view interpolation and color correction, IEEE Trans. Circuits Syst. Video Technol. 17 (11) (2007) 1436–1449.
[14] M. Panahpour Tehrani, A. Ishikawa, S. Sakazawa, A. Koike, Iterative colour correction of multicamera systems using corresponding feature points, J. Vis. Commun. Image Represent 21 (5) (2010) 377–391.
[15] K. Yamamoto, R. Oi, Color correction for multi-view video using energy minimization of view networks, Int. J. Autom. Comput. 5 (3) (2008) 234–245.
[16] C. Mouffranc, V. Nozick, Colorimetric correction for stereoscopic camera arrays, in: Proceedings Asian Conference Comput. Vision, Springer, Daejeon, Korea, 2012, pp. 206–217.
[17] K. Li, Q. Dai, W. Xu, Collaborative color calibration for multi-camera systems, Signal Process. Image Commun. 26 (1) (2011) 48–60.
[18] Y. Chen, C. Cai, J. Liu, Yuv correction for multi-view video compression, in: Proceedings International Conference Pattern Recognit., Vol. 3, Hong Kong, China, 2006, pp. 734–737.
[19] H. Jae-Ho, C. Sukhee, L. Yung-Lyul, Adaptive local illumination change compensation method for H.264/AVC-based multiview video coding, IEEE Trans. Circuits Syst. Video Technol. 17 (11) (2007) 1496–1505.
[20] M. Oliveira, A.D. Sappa, V. Santos, Color correction using 3D Gaussian mixture models, in: Proceedings International Conference Image Analysis and Recognition, Vol. 7324, Springer, 2012, pp. 97–106.
[21] S.A. Fezza, M.-C. Larabi, K.M. Faraoun, Feature-based color correction of multi-view video for coding and rendering enhancement, IEEE Trans. Circuits Syst. Video Technol. 19 (9) (2014) 1486–1498.
[22] D.G. Lowe, Object recognition from local scale-invariant features, in: Proceedings International Conference Comput. Vis., Vol. 2, IEEE, 1999, pp. 1150–1157.
[23] Q. Wang, P. Yan, Y. Yuan, X. Li, Robust color correction in stereo vision, in: Proceedings International Conference Image Proceedings, IEEE, 2011, pp. 965–968.
[24] H. Zeng, K.-K. Ma, C. Wang, C. Cai, Sift-flow-based color correction for multi-view video, Signal Process. Image Commun. 36 (2015) 53–62.
[25] H. Bay, A. Ess, T. Tuytelaars, L. Van Gool, Speeded-up robust features (SURF), Comput. Vis. Image Und. 110 (3) (2008) 346–359.
[26] B. Ceulemans, S.-P. Lu, P. Schelkens, A. Munteanu, Globally optimized multiview video color correction using dense spatio-temporal matching, in: Proceedings 3DTV-CON, IEEE, 2015, pp. 1–4.
[27] F. Shao, G. Jiang, M. Yu, K. Chen, A content-adaptive multi-view video color correction algorithm, in: Proceedings International Conference Acoust. Speech Signal Process., Vol. 1, Honolulu, Hawaii, 2007, pp. 969–972.
[28] F. Shao, G.-Y. Jiang, M. Yu, Y.-S. Ho, Fast color correction for multi-view video by modeling spatio-temporal variation, J. Vis. Commun. Image Represent 21 (5) (2010) 392–403.
[29] B. Shi, Y. Li, L. Liu, C. Xu, Color correction and compression for multi-view video using H.264 features, in: Proceedings International ASIA Conference Comput. Vis., Springer, 2009, pp. 43–52.
[30] E. Reinhard, M. Adhikhmin, B. Gooch, P. Shirley, Color transfer between images, IEEE Comput. Graph. Appl. 21 (5) (2001) 34–41.
[31] X. An, F. Pellacini, Appprop: all-pairs appearance-space edit propagation, ACM Trans. Graph 27 (3) (2008) 40:1–40:10.

[32] S.-P. Lu, G. Dauphin, G. Lafruit, A. Munteanu, Color retargeting: interactive time-varying color image composition from time-lapse sequences, Comput. Vis. Media 1 (4) (2015) 321–330.
[33] Y. Qian, D. Liao, J. Zhou, Manifold alignment based color transfer for multiview image stitching, in: Proceedings International Conference Image Process., IEEE, 2013, pp. 1341–1345.
[34] K. Xu, J. Wang, X. Tong, S.-M. Hu, B. Guo, Edit propagation on bidirectional texture functions, Comput. Graph. Forum 28 (7) (2009) 1871–1877.
[35] J.D. Hincapié-Ramos, L. Ivanchuk, S.K. Sridharan, P.P. Irani, Smartcolor: real-time color and contrast correction for optical see-through head-mounted displays, IEEE Trans. Vis. Comput. Graph. 21 (12) (2015) 1336–1348.
[36] X. Li, H. Zhao, G. Nie, H. Huang, Image recoloring using geodesic distance based color harmonization, Comput. Vis. Media 1 (2) (2015) 143–155.
[37] S.T. Roweis, L.K. Saul, Nonlinear dimensionality reduction by locally linear embedding, Science 290 (5500) (2000) 2323–2326.
[38] K. Xu, Y. Li, T. Ju, S.-M. Hu, T.-Q. Liu, Efficient affinity-based edit propagation using K-D tree, ACM Trans. Graph 28 (2009) 118:1–118:6.
[39] L.K. Saul, S.T. Roweis, An introduction to locally linear embedding, unpublished. Available at: ⟨http://www.cs.toronto.edu/roweis/lle/publications.html ⟩.
[40] P.F. Alcantarilla, A. Bartoli, A.J. Davison, Kaze features, in: Proceedings Euro. Conference Comput. Vis., Springer, 2012, pp. 214–227.
[41] P.F. Alcantarilla, T. Solutions, Fast explicit diffusion for accelerated features in nonlinear scale spaces, IEEE Trans. Pattern. Anal. Mach. Intell. 34 (7) (2011) 1281–1298.
[42] D. Krishnan, R. Fattal, R. Szeliski, Efficient preconditioning of laplacian matrices for computer graphics, ACM Trans. Graph 32 (4) (2013) 142:1–142:15.
[43] Z. Wang, A.C. Bovik, H.R. Sheikh, E.P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Image Process 13 (4) (2004) 600–612.
[44] S. Lu, J. Hanca, A. Munteanu, P. Schelkens, Depth-based view synthesis using pixel-level image inpainting, in: Proceedings International Conference Digital Signal Process., IEEE, 2013, pp. 1–6.
[45] S.-P. Lu, B. Ceulemans, A. Munteanu, P. Schelkens, Performance optimizations for PatchMatch-based pixel-level multiview inpainting, in: Proceedings International Conference 3D Imag., IEEE, 2013, pp. 1–7.
[46] B. Ceulemans, S.-P. Lu, P. Schelkens, G. Lafruit, A. Munteanu, Efficient MRF-based disocclusion inpainting in multiview video, in: International Conference Multimed. Expo. (Oral), IEEE, 2016, pp. 1–6.