

Spatio-temporally Consistent Color and Structure Optimization for Multiview Video Color Correction

Shao-Ping Lu, *Member, IEEE*, Beerend Ceulemans, *Student Member, IEEE*, Adrian Munteanu, *Member, IEEE*, and Peter Schelkens, *Member, IEEE*

Abstract—When compared to conventional 2D video, multiview video can significantly enhance the visual 3D experience in 3D applications by offering horizontal parallax. However, when processing images originating from different views, it is common that the colors between the different cameras are not well calibrated. To solve this problem, a novel energy function-based color correction method for multiview camera setups is proposed to enforce that colors are as close as possible to those in the reference image but also that the overall structural information is well preserved. The proposed system introduces a spatio-temporal correspondence matching method to ensure that each pixel in the input image gets bijectively mapped to a reference pixel. By combining this mapping with the original structural information, we construct a global optimization algorithm in a Laplacian matrix formulation and solve it using a sparse matrix solver. We further introduce a novel forward-reverse objective evaluation model to overcome the problem of lack of ground truth in this field. The visual comparisons are shown to outperform state-of-the-art multiview color correction methods, while the objective evaluation reports PSNR gains of up to 1.34 dB and SSIM gains of up to 3.2% respectively.

Index Terms—Multiview video, color correction, energy minimization, Laplacian matrices, spatio-temporal matching, structure preservation.

I. INTRODUCTION

MULTIVIEW video significantly improves the viewing experience by providing an immersive perception of 3D scenes. A plethora of multiview video applications are being targeted nowadays, including multiview-based three-dimensional television (3DTV), free-viewpoint television (FTV), telepresence, content creation, advertising, entertainment, gaming, or virtual reality. Due to the continuous progress of image-based rendering and view interpolation technologies, these applications have received a steadily increasing attention over the past years both from the academic and industrial communities [1], [2].

Although multiview video intrinsically contains more information about the 3D scene compared to conventional single-view video, its broad-scale deployment in practice remains a distant objective, and a lot of research is still necessary in order

to make it a reality. Among the open issues, one challenging topic is multiview video color correction. Because multiview video is recorded by different cameras, significant inter-view variations in the recorded color channels may arise either globally, over an entire video frame, or locally at certain points of interest. Global color misalignment could be introduced by different color temperatures and other parameter miscalibrations between the cameras. Local color discrepancies are usually caused by different lighting conditions because of complex surface material and reflective properties of objects in the scene. Consequently, when generating novel views in the scene using image-based rendering (IBR) techniques [3], the misaligned colors between the input views lead to color inconsistencies in the synthetic views. These artifacts severely affect the visual quality in the synthesized views. Such color misalignment (or color asymmetries) would also result in visual fatigue, binocular rivalry and other negative 3D viewing effects [4]–[6]. Moreover, multiview compression performance is affected by global or local illumination variations between different views.

To account for color differences in multiview video, the scene content can be modified directly, which is referred to as *prefiltering* [7], or it can be changed during the encoding process [8]. In contrast to a prefiltering method, color correction carried out within a compression system can benefit from information computed by the encoder (for example the motion vectors). Obtaining this information in a prefiltering method would lead to a significant increase in computational complexity. On the other hand, prefiltering techniques are not limited by the constraints imposed in a coding system, whose main goal is to optimize the rate-distortion performance, and which does not take into account the constraints that should be satisfied by a color correction scheme.

The color correction method proposed in this work can be considered as a prefiltering technique. In this family of algorithms, multiview color correction is performed according to some *target* criteria [9] which needs to be optimized by taking the other views into account. The reference technique in the literature in this category is the traditional histogram matching method which attempts to correct the color between the input and reference images according to a global cumulative histogram matching criterion. Despite of their popularity, histogram methods [7], [10], [11] are agnostic about structural information in the input images, which may cause local distortions in the textural structures when color correction is performed. The proposed method solves this problem by making use of local texture information in the

Copyright (c) 2013 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org.

The authors are with the Department of Electronics and Informatics (ETRO), Vrije Universiteit Brussel, iMinds, B-1050 Brussels, Belgium (e-mail: splu@etro.vub.ac.be; bceulema@etro.vub.ac.be; acmuntea@etro.vub.ac.be; pschelke@etro.vub.ac.be). This paper has supplementary downloadable material available at <http://ieeexplore.ieee.org>, provided by the author. The material includes a video demo. Contact splu@etro.vub.ac.be for further questions about this work.

input images and by finding an optimal reconstruction that effectively alters the color histogram, while preserving the structural information as well as colors at important feature points.

The main contributions of the proposed work are as follows.

- We propose a novel approach for color correction which preserves structural information in the corrected output and provides color consistencies of corresponding spatio-temporal feature points.
- The color correction problem is formulated as an optimization problem, which is solved using an efficient Laplacian matrix optimization framework.
- To overcome the problem that no ground truth in this research domain exists, we introduce a novel forward-reverse evaluation methodology to evaluate the color corrected results.

The remainder of this paper is organized as follows. In the next section we discuss the related work. The proposed multiview video color correction method is presented in detail in Section III. Section IV reports and discusses the experimental results and compares our method with existing reference techniques from the literature. Finally, Section V draws the conclusions of our work.

II. RELATED WORK

A wide variety of color correction methods have been proposed in the literature. Multiview video color correction techniques can be roughly divided into three categories, namely, (i) hardware-based approaches performing color calibration in the acquisition step [12], [13], (ii) techniques performing prefiltering after acquisition, and (iii) those performing synchronous correction during encoding. Here we focus on the last two categories, as hardware-based approaches are beyond the scope of this paper.

As a preprocessing step in video compression, a block-based matching and average color based least-squares regression method is proposed in [9] to compensate for color differences in the different input views. Kim *et al.* [14] also introduce a combination of algorithms for color and focus correction as a prefiltering step before encoding. In [15], the authors propose a block-based histogram matching method, which needs to perform spatial prediction for matching image parts which is time consuming. Yamamoto *et al.* [16] introduce a Gaussian filter-based color detection method with lookup tables to correct the multiview color differences. To build nonlinear lookup tables, researchers also introduce dynamic programming [17] or polynomial basis [18] optimization by local Gaussian-like analysis in the image space. In [19], the input image is segmented and corrected by locally corresponding keypoints. Shao *et al.* introduce both preferred region selection [20] and linear regression [21] to find the correction coefficients for local regions. By constructing a spatial variant color discrepancy model, they extend their multiview color correction work in the temporal direction. Their work requires constructing a precomputed disparity map using a linear operator and performs a PCA-based relevant color selection being however limited to linear corrections.

Most existing preprocessing methods do not pay much attention to the original local texture information when searching for a globally optimal color correction solution. To increase encoding gains in multiview video, a histogram matching-based luminance and chrominance correction method is proposed in [7], [22] to align all camera views to a reference view in the center of the camera array. Fezza *et al.* also introduce a histogram matching-based reference view decision and color correction model [11]. Cumulative histogram-based matching can be seen as a special kind of nonlinear matching when applied to the entire image, being a useful technique to correct global discrepancies in the color space. However, the major drawback of the histogram matching method is that the color matching is based on the global color distributions of the input and reference images, and therefore the original texture or gradient information can be easily destroyed. As it will be shown in this paper, global histogram-based matching does indeed change the local gradient structures when performing color correction. In image-based rendering, the main goal is to make use of such complex surface characteristics (gradients) and effectively preserving them is of paramount importance for a realistic rendering result. To overcome this drawback, linear scaling [9], [23], pairwise basis function [8] or high-order polynomials [9] are often used to fit the color distribution. Ilie *et al.* [12] construct a linear transformation to modify the camera's color by identifying an optimal 3×3 transform matrix. Li *et al.* [24] introduce an over determined multiview color calibration method. Different from our approach, their method is based on global linear-wise correspondence correction and a homogeneous linear system by dynamic range shaping.

Color compensation performed in coding systems has also been well researched recently. The basic philosophy in this kind of methods is to subtract the average difference between the reference view and the current one when performing motion compensation in the encoder. In [8], the DC coefficients at macroblock (MB) level are recomputed according to the current image and the corresponding MB in the reference camera. Similarly, Lee *et al.* [25] propose a DC coefficient modification scheme and integrate it into the MPEG Joint Multiview Video Model (JMVM) reference software. Yamamoto *et al.* [26] also use correction lookup tables during the compression process for interview prediction. In general, this family of methods usually needs to predict the illumination differences between MBs and compress the predicted color differences into the bit stream. Block-matching correction methods would also result in outliers because of some poor matches. Following the assumption that space-time neighboring frames depict close similarity, a MB-based color annotation and colorization postprocessing method is proposed in [27]. Their work is based on directional spatial prediction used for intra-frame coding, and performs a prediction-based reference generation according to the block's weighted average instead of the original gradient information. As discussed in [27], MB-based color compensation easily suffers from blocking artifacts. Furthermore, the modification strategies of the motion estimation (ME), motion compensation (MC) and other related reconstruction processes do increase complexity and cost at both encoding and decoding sides.

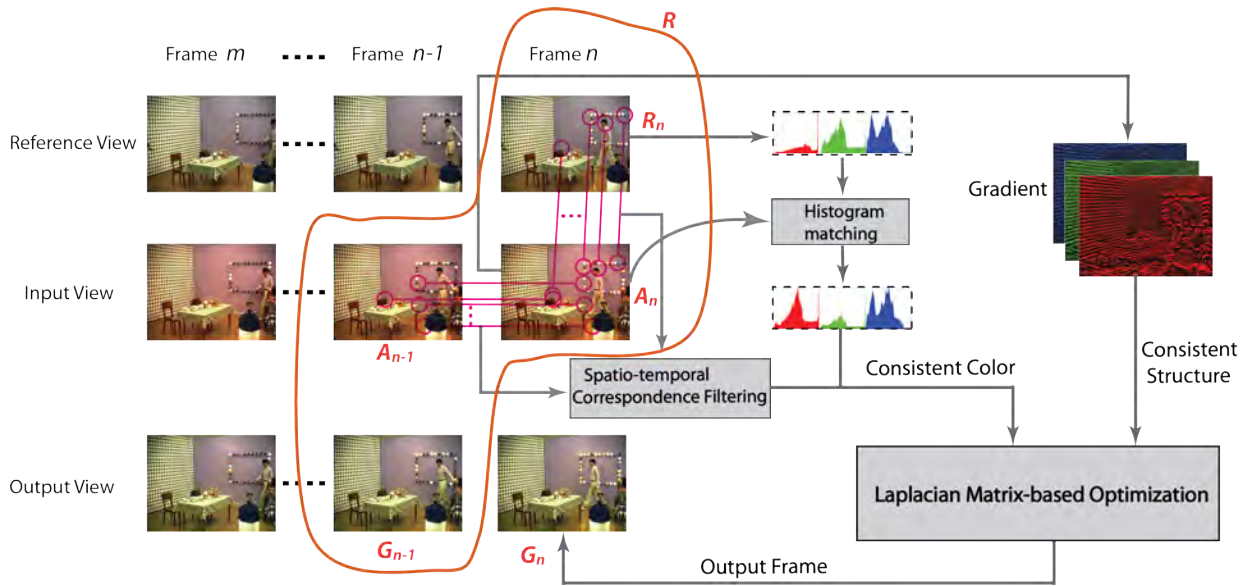


Fig. 1. The generic framework describing the proposed multiview video color correction algorithm.

Color correction related image/video processing is a hot issue in many other research fields, e.g., interactive recolorization [28], [29], image texture decomposition [30], video sequence matching [31] and color transfer between images [32], [33]. At the same time, color correction is a fundamental topic in stitching and composition using images captured from different views; we refer the reader to recent surveys and evaluations of color image/video editing techniques [34], [35]. Despite of this broad spectrum of methods and research interests, to the best of our knowledge, none of the existing color correction works does take advantage of the inherent coherency of multiple viewpoints in the same scene or of temporal consistencies in the same viewpoint. These are aspects that are accounted for in the proposed approach, as detailed next.

III. PROPOSED ALGORITHM

A. Overview

The proposed multiview video color correction framework is depicted in Fig. 1. To compute the optimal color for the input view according to the reference view, we construct a Laplacian matrix optimization problem by taking into account the original texture's structural information, the color of spatio-temporally consistent correspondence pairs as well as histogram-based matching. The corrected image should satisfy that its structural information (we focus on the gradient) is as similar as possible to the input, and the spatio-temporal correspondence pairs are consistently preserved with those not only in temporally adjacent frames from the same view but also in the synchronous frame of the reference view. The proposed color correction approach is detailed next.

B. Model Formulation

Suppose that A_n is the n th input frame from a camera, R_n is the synchronous reference image from a reference camera

and the target image G_n is the expected output color image. From the 3D geometry perspective, the 3D Euclidean scene (*visual space*) \mathbb{V}^3 should follow the mapping relation

$$f: \mathbb{V}^3 \mapsto \mathbb{E}^2 \times \mathbb{D}^1, \quad (1)$$

where \mathbb{E}^2 is the 2D Euclidean plane (*pictorial space*) and \mathbb{D}^1 is a 1D affine structure (e.g., *depth space*) [36]. If such scene mapping is lossless and the viewpoint warping between different cameras was bijective according to the scene elements (or pixels with discrete format), the color transition between A_n and R_n would be perfectly described. However, in current multiview camera models, the scene mapping relation is far from lossless and the reconstructed content contains missing (or invalid mapping) points. Furthermore, temporal fluctuations of color (color flickering) are produced by different camera exposure times, color temperature or noise. The main principle that drives the proposed color correction model is that the color of corresponding pairs of points should be consistent when viewing them from different neighboring cameras. Furthermore, the structure information of the output scene content G_n should be consistent with that of the original A_n . In the following, we construct an energy formulation and optimization problem to mathematically describe the aforementioned *color consistency* and *structure consistency* concepts.

Color consistency. The color consistency (*data term*) means that the color of the content in target G_n should be as *consistent* as possible to that of both the reference image R_n and the previously corrected image G_{n-1} . Firstly, we define a reference set of known images $R = \{R_n, A_n, A_{n-1}, G_{n-1}\}$, and an intermediate pixel set \tilde{G} , having the same size of G_n , in which each pixel is calculated from one of the images in R . Hence, the color consistency energy is defined as

$$E_d = \sum_{\mathbf{u}} (G_n(\mathbf{u}) - \tilde{G}(\mathbf{u}))^2, \quad (2)$$

where \mathbf{u} is the pixel position of the image. To compute \tilde{G} from R , feature matching and 3D mapping relation between cameras can be taken into account. In III-C the proposed construction of \tilde{G} is detailed.

Structure consistency. To preserve the scene content in the output image, we introduce the preservation of the image gradient as a means to enforce structure consistency. In other words, the gradient in G_n should be preserved such that it is as *similar* as possible to the gradient of the input image A_n . Formally, the structure consistency energy (*regularization term*) is defined as:

$$E_r = \sum_{\mathbf{u}} \left(\left(\frac{\partial G_n(\mathbf{u})}{\partial x} - \frac{\partial A_n(\mathbf{u})}{\partial x} \right)^2 + \left(\frac{\partial G_n(\mathbf{u})}{\partial y} - \frac{\partial A_n(\mathbf{u})}{\partial y} \right)^2 \right), \quad (3)$$

where $\frac{\partial}{\partial x}$ and $\frac{\partial}{\partial y}$ are the discrete differentiation operator in the horizontal and vertical directions, respectively. It should be noted that this equation can be rewritten in the following matrix-vector form:

$$E_r = (D_x G_n - D_x A_n)^T (D_x G_n - D_x A_n) + (D_y G_n - D_y A_n)^T (D_y G_n - D_y A_n), \quad (4)$$

where matrices D_x and D_y are forward discrete differentiation operators in horizontal and vertical directions, respectively. $(\cdot)^T$ denotes the transpose operator of a matrix. Because of the structure consistency between A_n and G_n , we assume that feature matching (mainly based on gradient and luminance information) between any image, name it B_n , and A_n is the same as the matching between B_n and G_n .

Total energy. The proposed total color correction energy consists of the color consistency energy and structure consistency energy, that is:

$$E = E_d + \beta_r E_r, \quad (5)$$

where β_r controls the balance between the color consistency energy E_d and structure consistency energy E_r . Modelling of E_d is explained in the next section. Multiview color correction is posed thus as a global optimization problem wherein the total energy E in the output image is minimized.

C. Solution

To solve Eq. (5) and to perform multiview video color correction, we introduce correspondence modeling and filtering schemes as well as a Laplacian matrix-based optimization solution.

Correspondence modelling. To construct the pixel set \tilde{G} from the reference image set R , we consider three classes of image-wise correspondences:

- 1) temporal correspondence mapping between temporally adjacent frames in the output camera;
- 2) spatial correspondence mapping between input (or output) and reference frames;
- 3) additional correspondence mapping, where pixels that cannot correspond to a temporal or spatial neighbor are matched to a histogram corrected version of the input.

The first class of mappings describes the color consistency in the temporal direction of the input view, and we define this class of relevant pixels in \tilde{G} as the set of \tilde{G}_t . The second class describes the color consistency between different camera views, and the mapping pixels are denoted as \tilde{G}_c . Ideally, the mapping between \tilde{G} and G_n is bijective. However, not only are there overlapping points, but there are also missing points when performing the 3D warping from R_n to G_n or matching between A_n and G_{n-1} and these are caused by discretization, occlusion, scene motion, etc. For this reason, the first two mapping classes cannot constitute a complete bijective mapping. Therefore, we introduce the supplementary pixel set \tilde{G}_h to complete the full bijective mapping relation between \tilde{G} and G_n and this pixel set should satisfy the following property:

$$\tilde{G}_h = \tilde{G} \setminus (\tilde{G}_t \cup \tilde{G}_c), \quad (6)$$

where (\setminus) indicates the relative complement of two sets and (\cup) denotes the union operator.

Suppose H_n is a histogram matching image from A_n to R_n , and $p(\mathbf{u}, \mathbf{v})$ denotes an aforementioned correspondence pair between a point located at \mathbf{u} (in \tilde{G}) and another pixel at \mathbf{v} (in R_n , G_{n-1} or H_n). According to the mapping classes, we further denote the sets of corresponding point pairs as Ψ_t , Ψ_c and Ψ_h , with different weighting factors β_t , β_c and β_h , respectively. Note that in $\Psi_h(\mathbf{u}, \mathbf{v})$ each correspondence pair should follow $\mathbf{u} = \mathbf{v}$, which means that the points are co-located in the same position of the image space. The pixel values in \tilde{G} are computed as

$$\tilde{G}(\mathbf{u}) = \begin{cases} G_{n-1}(\mathbf{v}) + \alpha_n(\mathbf{u}, \mathbf{v}) & \text{if } \mathbf{u} \in \tilde{G}_t, p(\mathbf{u}, \mathbf{v}) \in \Psi_t \\ R_n(\mathbf{v}) + \tau_n(\mathbf{u}, \mathbf{v}) & \text{if } \mathbf{u} \in \tilde{G}_c, p(\mathbf{u}, \mathbf{v}) \in \Psi_c \\ H_n(\mathbf{v}) & \text{otherwise} \end{cases}, \quad (7)$$

where $\alpha_n(\mathbf{u}, \mathbf{v})$ indicates the original temporal color differences between \mathbf{u} (in A_n) and \mathbf{v} (in A_{n-1}), and $\tau_n(\mathbf{u}, \mathbf{v})$ is the color transition term between \mathbf{u} (in G_n) and \mathbf{v} (in R_n). According to this equation, for each point belonging to the first class, its target color value is computed using its corresponding point in G_{n-1} and the original temporal color difference in the input camera. Similarly, in the second mapping class each pixel's target color is computed based on the corresponding

$$E_d = \sum_{\substack{\mathbf{u} \in \tilde{G}_t, \\ p(\mathbf{u}, \mathbf{v}) \in \Psi_t}} \beta_t (G_n(\mathbf{u}) - \tilde{G}(\mathbf{u}))^2 + \sum_{\substack{\mathbf{u} \in \tilde{G}_c, \\ p(\mathbf{u}, \mathbf{v}) \in \Psi_c}} \beta_c (G_n(\mathbf{u}) - \tilde{G}(\mathbf{u}))^2 + \sum_{\substack{\mathbf{u} \in \tilde{G}_h, \\ p(\mathbf{u}, \mathbf{v}) \in \Psi_h}} \beta_h (G_n(\mathbf{u}) - \tilde{G}(\mathbf{u}))^2 = \\ \sum_{\substack{\mathbf{u} \in \tilde{G}_t, \\ p(\mathbf{u}, \mathbf{v}) \in \Psi_t}} \beta_t (G_n(\mathbf{u}) - G_{n-1}(\mathbf{v}) - \alpha_n(\mathbf{u}, \mathbf{v}))^2 + \sum_{\substack{\mathbf{u} \in \tilde{G}_c, \\ p(\mathbf{u}, \mathbf{v}) \in \Psi_c}} \beta_c (G_n(\mathbf{u}) - R_n(\mathbf{v}) - \tau_n(\mathbf{u}, \mathbf{v}))^2 + \sum_{\substack{\mathbf{u} \in \tilde{G}_h, \\ p(\mathbf{u}, \mathbf{v}) \in \Psi_h}} \beta_h (G_n(\mathbf{u}) - H_n(\mathbf{v}))^2 \quad (8)$$

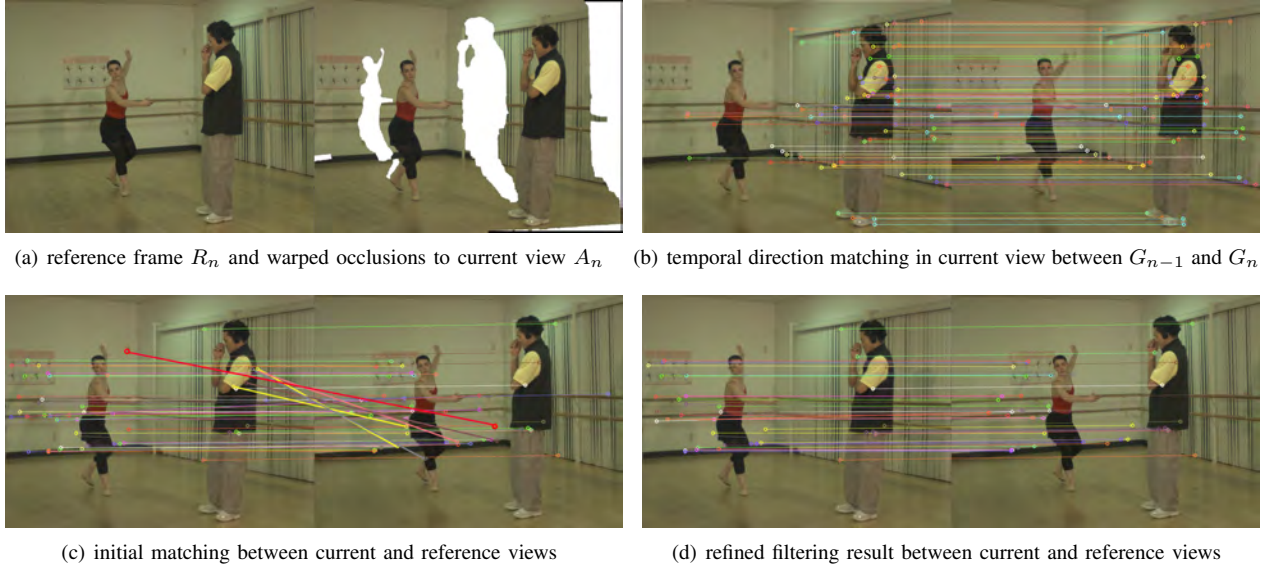


Fig. 2. Corresponding feature pairs filtering effect for multiview video. (a) reference frame and occlusion regions when warping to current view; (b) temporal correspondence pairs according to SURF feature matching [37] between adjacent frames in current view; (c) initial matching result between current and reference view by feature matching; (d) refined filtering result between current and reference view.

point in the reference camera and the transition color between the two cameras. Each pixel’s target color in the last mapping set is equal to the corresponding pixel’s color in H_n .

Finally, by recalculating the valid mapping points with the class formulation as described in Eq. (7), the color consistency energy in Eq. (2) can be rewritten as Eq. (8). Note again that in order to preserve the temporal correspondence matching, we use the uncorrected image to perform matching, while we use the corrected color in the previous frame to finalize the color correction in current frame. For instance, we search for corresponding points by performing the matching between A_{n-1} and A_n , but we choose the color of corresponding target pixels from G_{n-1} (instead of those from A_{n-1}) for temporally consistent color preservation in the optimization.

Spatio-temporal correspondence pair filtering. Correspondence matching for multiview color correction can be performed using block-based matching methods [9], [27] or feature point matching [11], [16], [20]. One of the advantages of block-based matching is that it can produce far more matching points between images [9], but these matches are often less reliable compared to feature point matching. Hence, we employ Speeded-Up Robust Feature (SURF) [37] matching in order to obtain pairwise corresponding points.

Although the popular SURF approach produces good matching results, we can improve its performance in multiview video correspondence matching by applying several filtering techniques, as detailed next. Ideally, the matching vectors of the scene content between different cameras should be decided by the camera view warping and by the object’s velocity in the scene for temporally adjacent frames. Moreover, local matching vectors should tend to be locally consistent (cfr. the rigid body motion assumption in optical flow). Thus, we filter out wrong correspondences by applying this principle. Note that we followed a similar methodology in multiview video inpainting [38], but there the correspondences are constructed

and filtered based on the immediate spatial neighbors.

Firstly, the camera array in multiview video capture systems is relatively dense (with many of the cameras being distributed in the same horizontal line). Thus, for spatially corresponding matching points we introduce a threshold to eliminate outliers for which the matching vectors would be too large:

$$\Gamma_c(p(\mathbf{u}, \mathbf{v})) = \begin{cases} 1 & \text{if } d_c(\mathbf{u}, \mathbf{v}) < \delta_c(\mathbf{u}, \mathbf{v}) \\ 0 & \text{otherwise} \end{cases}, \quad (9)$$

where $d_c(\mathbf{u}, \mathbf{v})$ is the ideal matching distance between a pixel at \mathbf{u} (in A_n) and its corresponding point at \mathbf{v} (in one image of R) by camera warping, and $\delta_c(\mathbf{u}, \mathbf{v})$ is a threshold on this distance. Because most object motion is in horizontal direction, we empirically determined $\delta_c(\mathbf{u}, \mathbf{v})$ as 300 and 50 pixels in the horizontal and vertical directions, respectively.

Furthermore, since the content captured by different cameras are in the same scene, the matched content should not appear in the camera warping hole. In other words, the matched corresponding points in one camera should not have disappeared (be occluded, strictly speaking) in another camera. Hence, we define a matching depth filter to avoid this effect:

$$\Gamma_w(p(\mathbf{u}, \mathbf{v})) = \begin{cases} 1 & \text{if } d_w(\mathbf{u}, \mathbf{v}) = 1 \\ 0 & \text{otherwise} \end{cases}, \quad (10)$$

where $d_w(\mathbf{u}, \mathbf{v})$ is a binary mask image which indicates which pixels of the reference are visible to the input camera. Take Fig. 2(a) as an example, when R_n (the left part) is warped to A_n (the right part), there are some occlusion regions and we define each pixel’s d_w value as zero in such regions. For a pixel located at \mathbf{u} (in A_n) and its corresponding pixel at \mathbf{v} (in R_n), if $d_w(\mathbf{u}, \mathbf{v}) = 0$, it means the matching is outside of the input image or occluded (by a foreground object), $\Gamma_w(p(\mathbf{u}, \mathbf{v}))$ ensures that there will be no matching between these points.

As an example, Fig. 2(c) and Fig. 2(d) show the effect with and without the matching vector filters, respectively, and

TABLE I
COMPLEXITY OF THE PROPOSED MULTIVIEW VIDEO CORRECTION ALGORITHM

Test Sequence	<i>Objects2</i>	<i>Race1</i>		<i>Rena</i>	<i>Flamenco2</i>	<i>Ballet</i>
Resolution	640 × 480	640 × 480	320 × 240	640 × 480	640 × 480	1024 × 768
Total frames	600	600	600	300	300	100
Total views	8	8	8	16	5	8
Input view number	2	4	2	9	1	6
Reference view number	1	1	1	1	3	3
Execution time (sec/frame)	matching: 0.125*2 solver: 0.895*3	matching: 0.11*2 solver: 0.878*3	matching: 0.03*2 solver: 0.279*3	matching: 0.113*2 solver: 0.883*3	matching: 0.138*2 solver: 0.872*3	matching: 0.825*2 solver: 2.152*3

Fig. 2(b) also shows the corresponding matching point pairs in the temporal direction. Our method allows for using any matching method (dense/sparse, accurate/fast) such as the ones in [39].

Global optimization. According to Eq. (4) and Eq. (8), the total energy function of Eq. (5) can be further reformulated as Eq. (11) with matrix notations. Note that in Eq. (11) all matrices are $N \times N$ in size (N is the size of the image). The diagonal matrix Q is determined by aforementioned three classes of matching points and each diagonal element in it is equal to the corresponding point's weighting factor. For instance, if a pixel from $G_{n-1}(\mathbf{v})$ is chosen as a target temporal correspondence point, we set $Q(\mathbf{u}, \mathbf{u}) = \beta_t$ and the color $\tilde{G}(\mathbf{u}, \mathbf{u}) = G_{n-1}(\mathbf{v}) + \alpha_n(\mathbf{u}, \mathbf{v})$. Similarly, when calculating a pixel from the second matching class, it follows $Q(\mathbf{u}, \mathbf{u}) = \beta_c$ and $\tilde{G}(\mathbf{u}, \mathbf{u}) = R_n(\mathbf{v}) + \tau_n(\mathbf{u}, \mathbf{v})$. To simplify the problem, in our implementation we set the transition factor $\tau_n(\mathbf{u}, \mathbf{v}) = 0$, which means that our optimization attempts to produce the same color for correspondence pairs between input and reference views. The matrix W is also diagonal in which all diagonal elements are β_r .

Because the proposed multiview video color correction approach is formulated as an optimization problem wherein the total cost function E given by Eq. (11) is minimized, the minimum of the discrete quadratic form of E is obtained by setting $dE/dG_n = 0$. Therefore, the optimal color correction solution can be found by solving the following linear system:

$$\begin{aligned} [Q + D_x^T W D_x + D_y^T W D_y] G_n = \\ Q \tilde{G} + (D_x^T W D_x + D_y^T W D_y) A_n, \end{aligned} \quad (12)$$

where D_x^T and D_y^T are backward discrete differentiation operators in horizontal and vertical directions, respectively. Eq. (12) is typically a very large linear system of equations with discrete Laplace operators. For instance, the inhomogeneous Laplacian matrix in the left-hand side of Eq. (12), $Q + D_x^T W D_x + D_y^T W D_y$, is 576 GB in size when the input multiview video resolution is 1024×768 . Nevertheless, this Laplacian matrix (as well as the entire right-hand side) is sparse diagonal-like and is symmetric positive definite; hence, a sparse Laplacian matrix solution can be directly obtained by Gaussian Elimination and Lower-Upper (LU) based Cholesky

factorization. Faster iterative algorithms, e.g., Jacobi, Gauss-Seidel and further Preconditioned Conjugate Gradients (PCG) are also applicable. In our implementation, we use Hierarchical Sparsification and Compensation (HSC) [40], which has a small operation count and wall-clock time due to an overall low-complexity construction, dramatically reducing the matrix condition number to efficiently solve the problem (the detailed computational performance of our solution will be shown in IV-D).

IV. EXPERIMENTAL RESULTS AND DISCUSSION

We performed experiments using the proposed method on a number of multiview video sequences including "Ballet", "Objects2", "Flamenco2", "Race1" and "Rena". The sequences characteristics are detailed in Table I. We remark that the experiments are carried out on various inter-camera distances. We note also that, for the purpose of comparison, we have selected the same reference views as those used in literature [7], [11], [21].

In general, we employ the Structural Similarity (SSIM) metric [41] to evaluate the structure information for the corrected color video. The Peak Signal to Noise Ratio (PSNR) is also used to calculate the produced color error, while in the output result we only evaluate the corresponding points. Additionally, we construct an evaluation model in which the color of the entire output image is assessed using the PSNR metric. In our implementation, all the experiments are performed in YUV 4:4:4 color space.

A. Implementation Details

For simplicity, we set the temporal weighting factor $\beta_t = \beta_c/5$, which means that the spatial energy term has a stronger relative importance than the temporal energy between adjacent frames (as it will be shown, this yields sufficiently low color fluctuations between temporally adjacent frames). By fixing the factor $\beta_h = 1$, the detailed relation between the performance of the color correction method and the critical parameters, β_c and β_r , is also investigated. As can be seen in Fig. 3, both aforementioned parameters are independently assigned between 0.01 and 100 respectively. The vertical axis

$$E = (G_n - \tilde{G})^T Q (G_n - \tilde{G}) + (D_x G_n - D_x A_n)^T W (D_x G_n - D_x A_n) + (D_y G_n - D_y A_n)^T W (D_y G_n - D_y A_n) \quad (11)$$

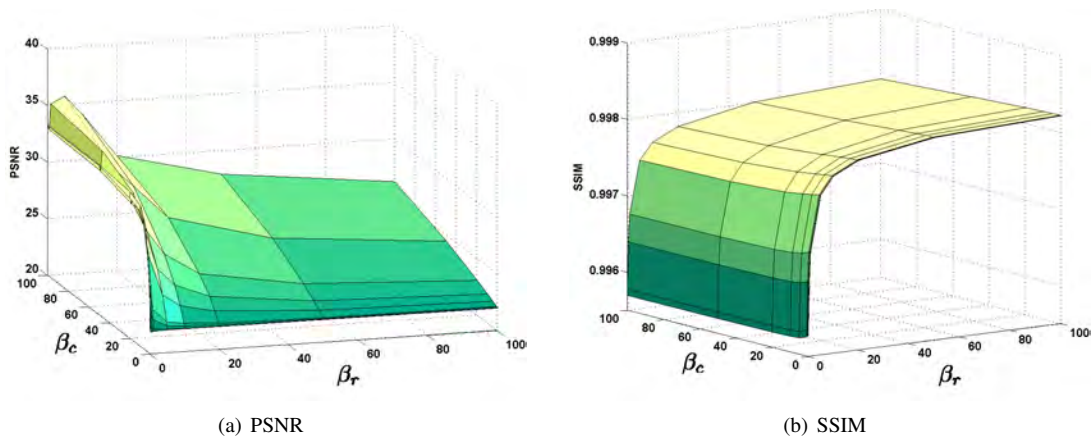


Fig. 3. Critical parameters selection and color correction performance distribution (based on the *Flamenco2* sequence).

indicates the PSNR and SSIM measured relative to the quality of the reconstructed image’s color and structure information, respectively. With the increase of β_r , the gradient energy in Eq. (5) gains more importance, so the SSIM value is higher (see Fig. 3(b)) and the structure is more similar to that of the original image. However, such processing affects the color values between corresponding matching points, thus the PSNR is reduced. Similarly, when β_c increases, the SSIM slowly decreases. For all tested sequences, we found an acceptable trade-off by setting $5 \leq \beta_r \leq 10$ and $50 \leq \beta_c \leq 100$. Thus, to simplify we fix $\beta_r = 6$ and $\beta_c = 100$ in our system.

B. Results

Visual examples produced by the proposed approach can be seen in Fig. 4. The first two rows show color correction result based on the *Objects2* sequence. Given the input video (Fig. 4(a)) and the reference video (Fig. 4(b)), the proposed color correction result is given in Fig. 4(d). We can see that the proposed method can produce the desired reference color and is still similar to the original input. Because histogram matching is generally used in the literature (e.g., in [7], [10], [22]), we compare against the method introduced in [7] (see Fig. 4(c)). In addition, Fig. 4(e) – 4(h) show the zoom-in of the rectangle content in the first row, respectively. The histogram-based matching clearly produces visual artifacts even in flat color regions. In contrast, our method effectively prevents such visual artifacts when correcting the image color.

To further illustrate the difference between above-mentioned approaches, difference images between the original image and the corrected image are shown for both methods in Fig. 4(i) and Fig. 4(j) respectively. It should be pointed out that all the difference images are normalized to the same scale and all the pixels of the difference images are lying in the range from 0 to V_{max} . V_{max} is the maximal absolute value for both the histogram matching error and the error by the proposed method. According to these difference images, the histogram-based method generates sharp errors in the corrected image while the proposed approach yields smoother and more subtle changes.

The last two images in Fig. 4 shows another example of the color difference for the *Ballet* sequence. One notices that

around the handle bar, the bright region on the curtain and on the body of the persons, smoother and smaller mismatchings are generated by the proposed approach compared to the histogram-based method. This example shows that even if the colors in this sequence are reasonably close across views, because of the coarse matching by global cumulative matching, the histogram-based color correction fails to preserve local structure in multiview video color correction. In contrast, benefiting from the structure preservation energy, the proposed approach ensures that the output’s structural information remains similar to that of the original image.

Fig. 5 shows another experiment, comparing our approach with other methods on the *Race1* sequence. The image of Fig. 5(g) was provided to us by the author of [21]. Similar to the previous results, to visualize the color differences, V_{max} is set as the maximal absolute value of all described methods (i.e., the histogram matching, the proposed and the discrepancy model [21]). Again, the proposed approach produces relatively smoother differences than the histogram-based method. Comparing with the discrepancy model introduced in [21], the contour of the gradually altered sky, wheels and road in our result is closer to the original image. One notes also that the SSIM differences are not proportional to the visual differences between pictures, and that even very small SSIM differences can correspond to substantial visual differences between pictures.

Fig. 6 shows the visual comparison with other methods when performing color correction for all input views at the same time. In this figure, the first row depicts five different input views including a reference camera (highlighted with a yellow rectangle). The images in second row are the color corrected results obtained with histogram matching [7]. The third row shows the corrected images obtained with the recently published CH-GOP (Calculated Histogram using only a Group Of Pictures) approach of Fezza *et al.* [11]. The fourth row of images are obtained with our method. The last three rows show the color difference images relative to the original pictures in each view obtained by above-mentioned approaches. One notices that the contours in the color difference images are much smoother for the proposed approach compared to histogram matching [7] and CH-GOP

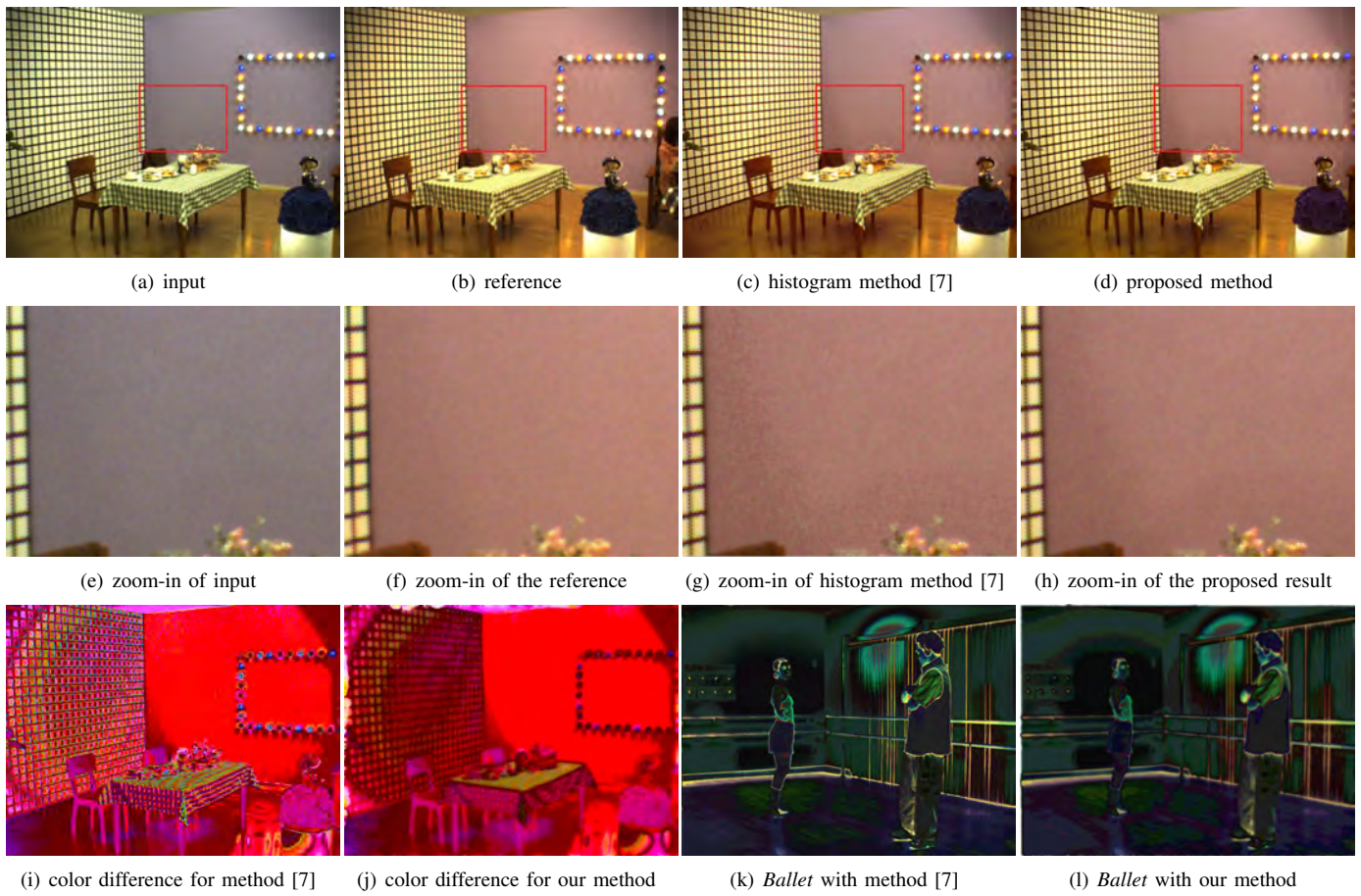


Fig. 4. Color correction results. Top row: original image, reference, histogram matching result [7] and the proposed result. Second row: zoom-in results from the top row respectively. Third row: the visualized color difference between the original image and the corrected image (the last two images are another example for the *Ballet* sequence). Note that our method can effectively preserve original texture structure information without smoothing in the gradient domain.

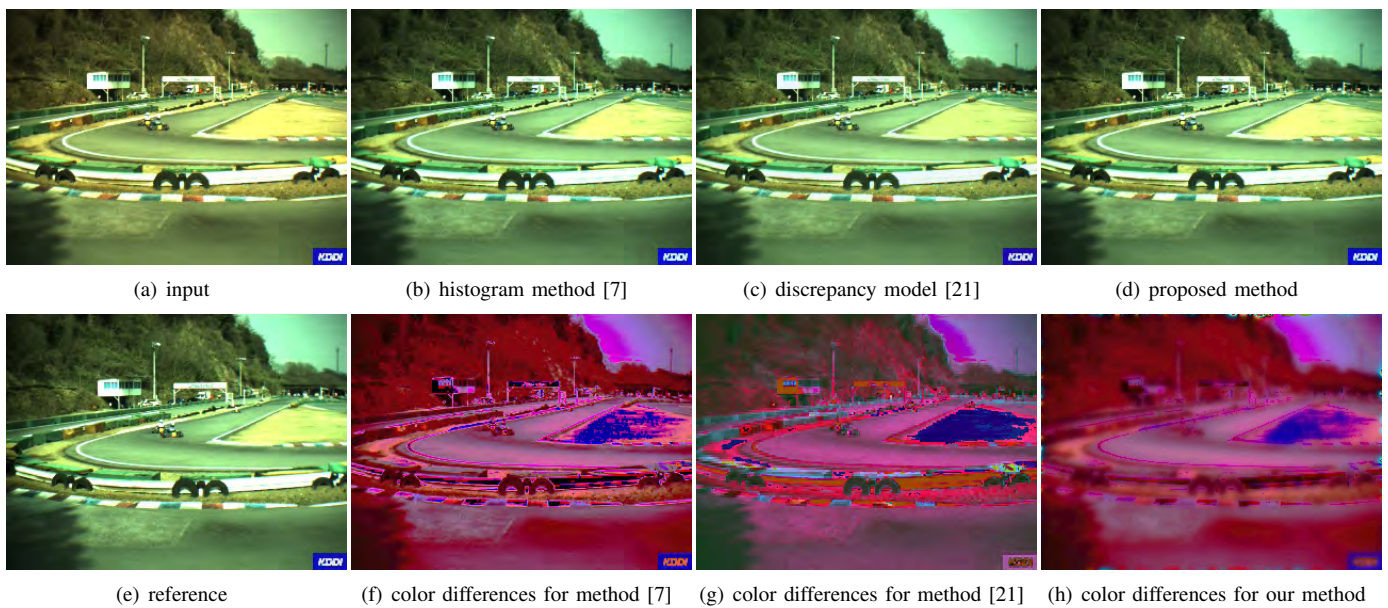


Fig. 5. Comparison with histogram-based [7] and discrepancy model [21]. Note that the proposed approach produces a smooth result while preserving the original structural information. The difference images of 5(f) – 5(h) are normalized in the same scale.

method [11]. This indicates a better preservation of the structural information offered by the proposed approach compared

to the reference techniques. Moreover, compared to the CH-GOP method [11], our approach can effectively avoid both

color variations on the ground and the strongly lighted areas (see the top of the images in Fig. 6(f)).

For the objective evaluation, we employ the PSNR and SSIM metrics to demonstrate the advantage of the proposed method (see Fig. 7). Note again that the PSNR is performed for the color of the consistent corresponding points both between adjacent frames and between cameras. Because there is no ground truth for multiview video color correction, it does not make sense to simply calculate the PSNR for the whole image. As it is clear from Fig. 7(b), the PSNR value for every frame is higher than the PSNR obtained with the histogram-based result. Similar PSNR results are obtained for other sequences - see Fig. 7(a) to Fig. 7(f), with average PSNR gains ranging from 0.52 dB (*Flamenco2* sequence) to 0.86 dB (*Rena* sequence). Although the point correspondences are relatively sparse (the number in every frame depending on the image content is between 30 and 200 points), these results indicate that our approach provides better color consistencies for feature points than the histogram-based method.

The SSIM values on the whole frame are calculated, as shown in Fig. 7(g) – Fig. 7(l). It is clear from these results that our approach effectively improves the SSIM values, systematically outperforming the histogram-based technique.

The objective correspondence color distortion and frame-based structure distortion metrics both clearly indicate the effectiveness of the proposed algorithm. Additionally, the SSIM results show that the proposed method can reasonably avoid fluctuating effects between successive frames. This is because the proposed method does not need to specify key-frame intervals by foreground/background detection or by making Gaussian color distribution assumptions in temporally adjacent frames as in [21] or perform spatio-temporal histogram calculations [7]; moreover, we do not need to estimate complicated correction parameters, thus our method can significantly avoid the negative cumulative effect for sudden scene changes.

C. Forward-reverse Evaluation

In multiview video currently there is no ground truth video for color correction. In other words, since the corrected video and the reference video originate from cameras at different viewpoints, a certain displacement between the views, different lighting as well as complex reflection, cause that the colors in the cameras are not well calibrated. Conversely, the color from one camera is difficult to be perfectly mapped to another camera for the same reasons.

The experiments so far have evaluated the PSNR for the color of corresponding points and the SSIM for the structure information of the whole frame, indicating that the proposed method outperforms its competitors. However, such evaluations are probably insufficient for the color distortion evaluation of the whole image. To address this, dedicated evaluation methods have been introduced in the literature. For instance, a distortion function using the gamma curve and linear transfer is presented in [16]. However, the real uncorrected colors between different views are difficult to be fitted by such linear transfer based model. In contrast, here we propose a forward-reverse evaluation approach for multiview video color correction.

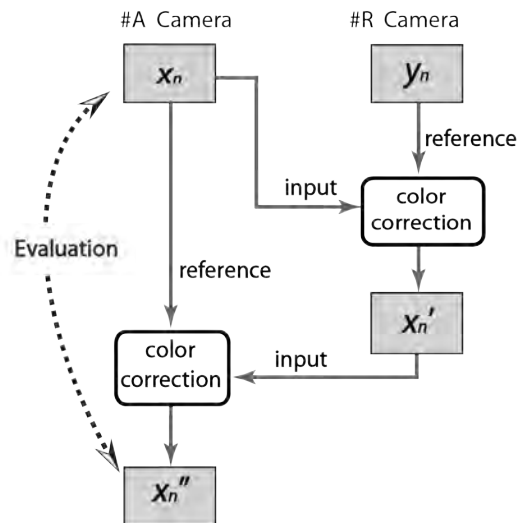


Fig. 8. Proposed forward-reverse color correction evaluation.

The block scheme of the proposed forward-reverse evaluation model is shown in Fig. 8. Firstly, the input video is corrected according to the video captured by the reference camera. Ideally, in this step the target color should be determined from the corresponding points in the reference image, while the target structure information should remain close to the one of the input image. Next, we reverse the correction operation and align the color of the corrected image back according to the original input camera view. In this loop-locked processing, the final result can be objectively compared to the original image using classical objective evaluation metrics such as the PSNR and SSIM. Therefore, we perform the proposed color correction approach to correct the video color and re-correct it back following the forward-reverse evaluation model. We also perform this analysis for the histogram-based method for comparison purposes.

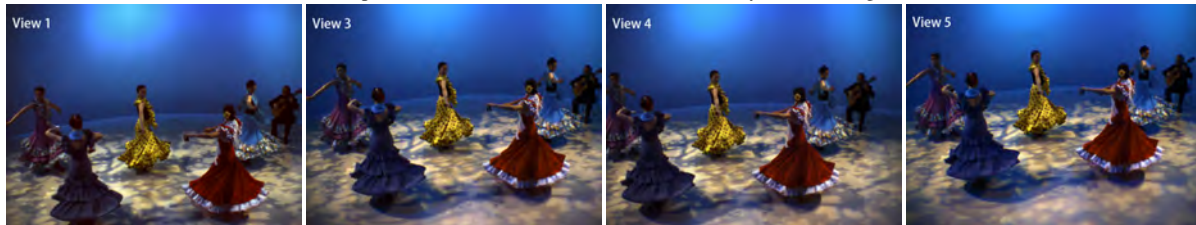
Fig. 9 shows the results of the forward-reverse evaluation. Among the experiments, the average PSNRs are between 33.80 (*Objects2* sequence) and 36.89 (*Race1* sequence), and the SSIMs are between 0.96 (*Objects2* sequence) and 0.97 (*Flamenco2* sequence). Again, the proposed approach outperforms the histogram-based matching method in all our experiments. Comparing with the histogram-based method, the average PSNR gains of our approach range from 0.66 dB (*Flamenco2* sequence) to 1.34 dB (*Objects2* sequence), and the average SSIM gains range from 1.0% (*Flamenco2* sequence) to 3.2% (*Objects2* sequence). These metrics also demonstrate that the proposed approach can effectively preserve the whole image color and texture structure, and that the color correction result is better than that of the cumulative histogram-based matching method.

D. Computational Complexity

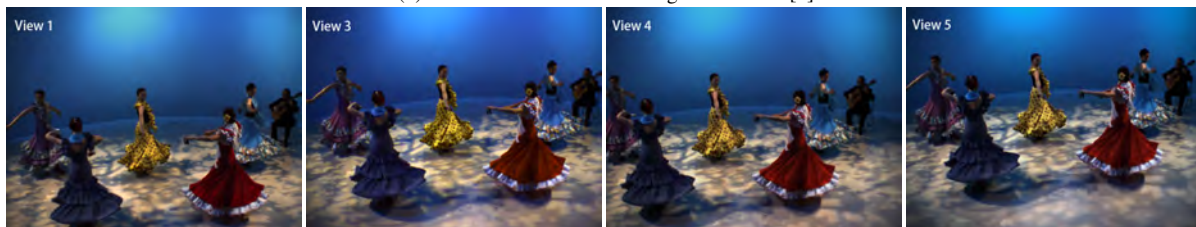
We implemented our system in Microsoft Visual Studio C++ 2010 platform combining this implementation with the Laplacian matrix optimization tools running in MATLAB on a high-end portable computer with 2.3GHz Quad-Core Intel-i7 CPU and 8GB memory.



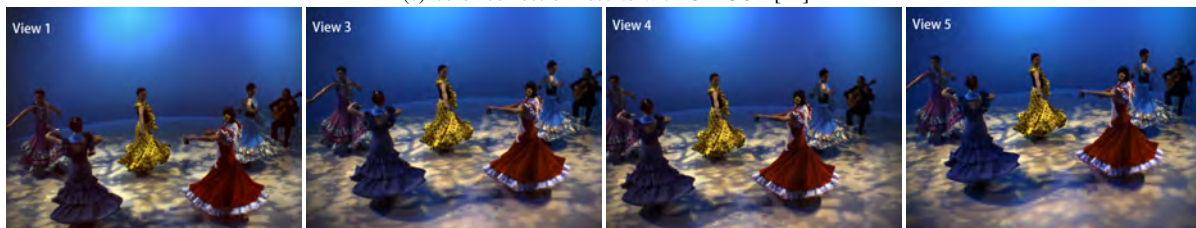
(a) input views and the reference camera (with a yellow rectangle)



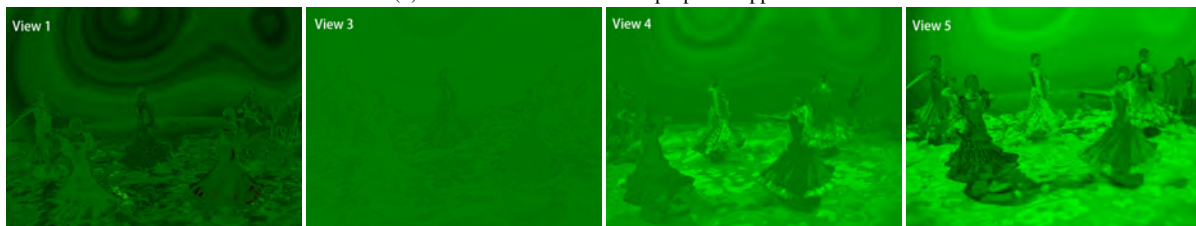
(b) corrected results with histogram method [7]



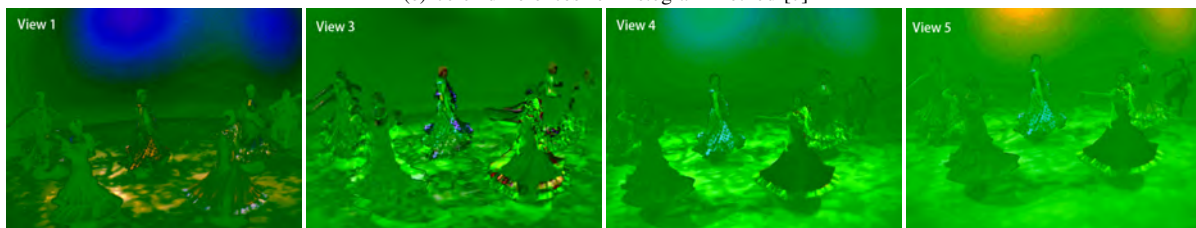
(c) color correction results with CH-GOP [11]



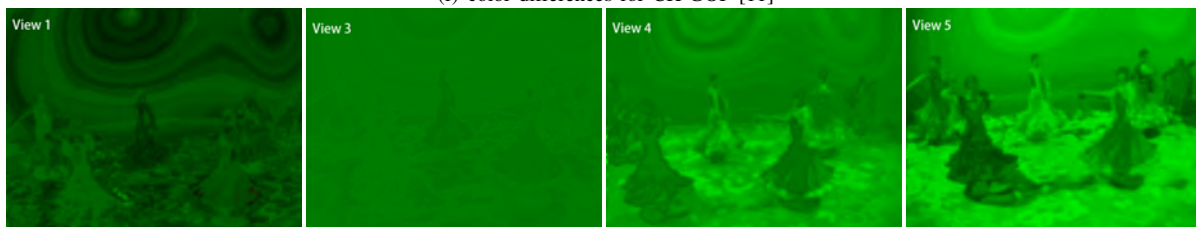
(d) Color correction with the proposed approach



(e) color differences for histogram method [7]



(f) color differences for CH-GOP [11]



(g) color differences for the proposed approach

Fig. 6. Comparison with histogram-based [7] and CH-GOP [11] for all views. The difference images in the last three rows are normalized in the same scale.

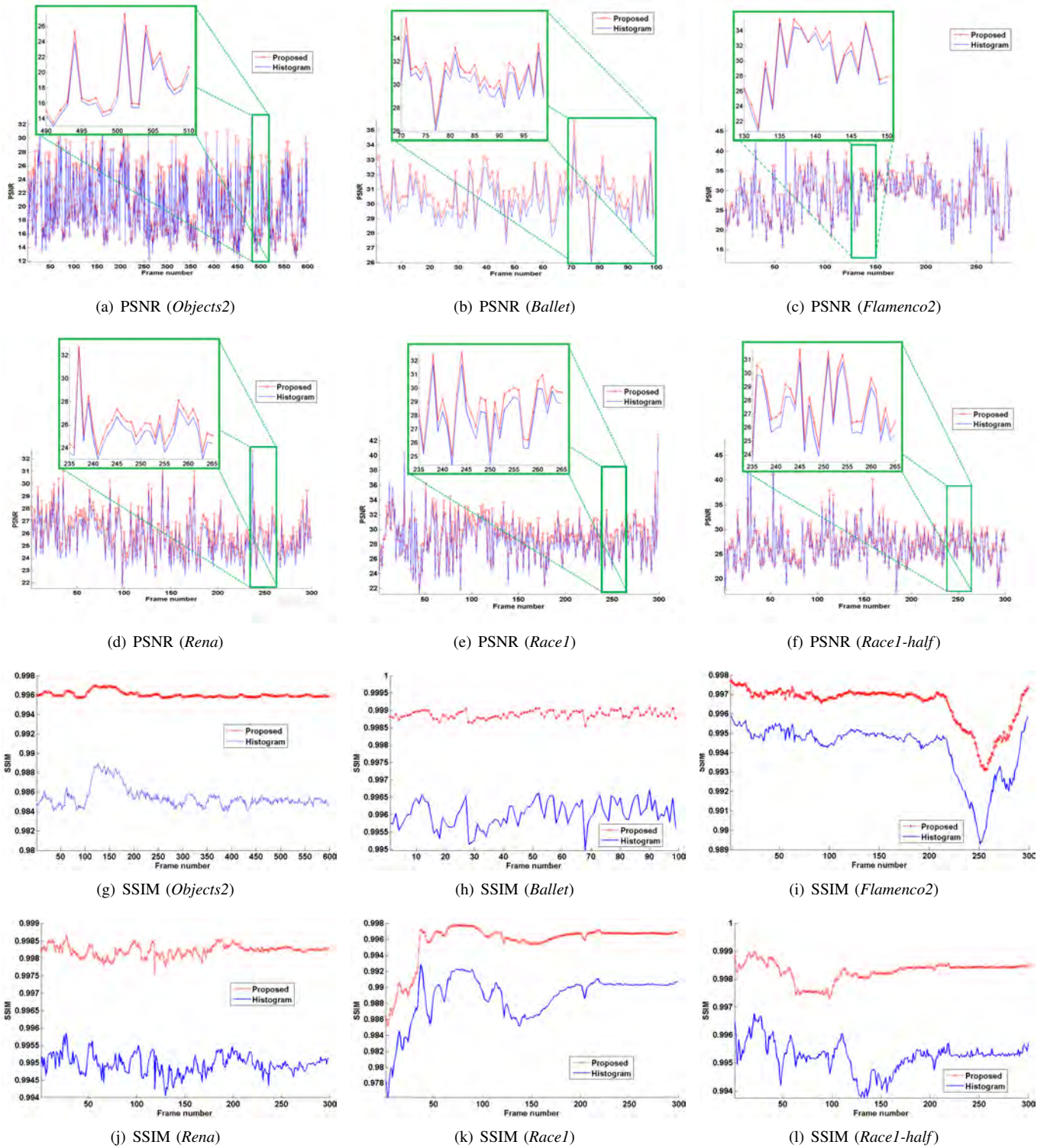


Fig. 7. Color correction result comparing with histogram-based [7]. Note that PSNR is calculated for the sparse feature matches and SSIM is for the whole image.

The run-times of the proposed algorithm are depicted in Table I. In our system, the computational complexity mainly lies in our correspondence matching and in finding the Laplacian matrix optimization solution. The correspondence matching processing is performed not only for temporally adjacent frames in the same view but also for synchronous frames

between input and reference views. Additionally, for each frame we perform the Laplacian matrix optimization in every color channel independently. For the low resolution *Race1* sequence (320×240), the time of single matching and matrix optimization are 0.03 and 0.279 seconds, respectively, and the total time consumption of every frame is less than 1.0s. For

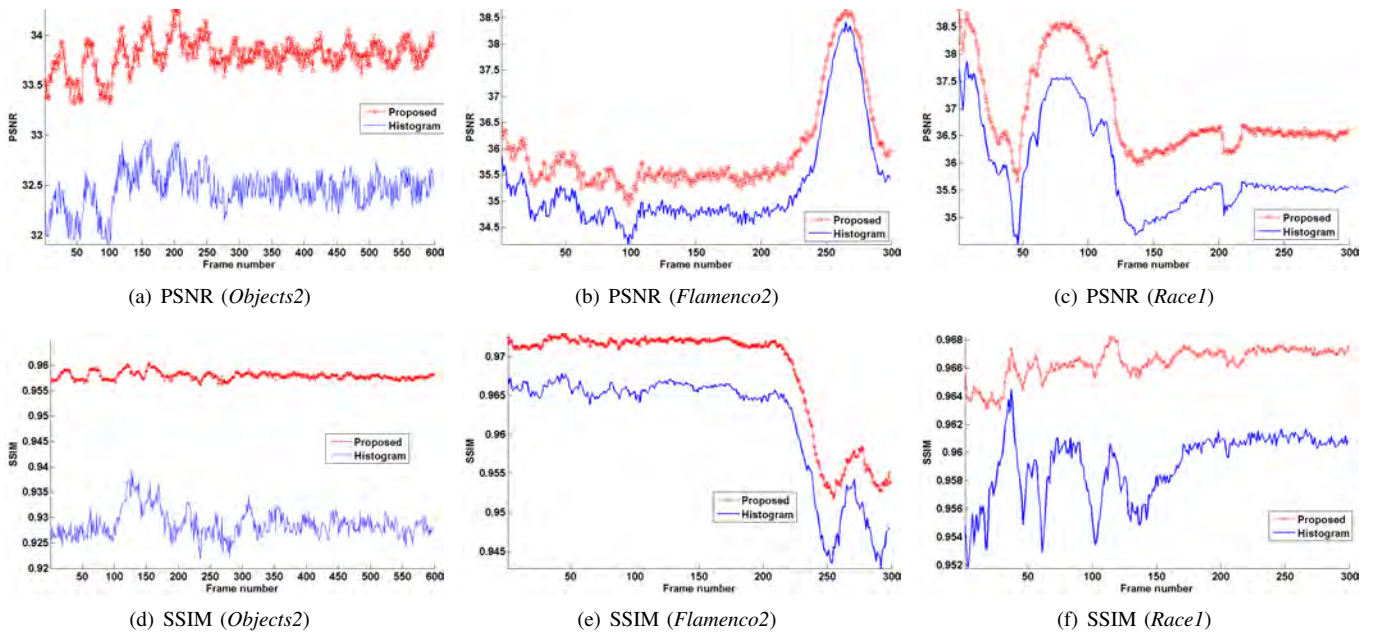


Fig. 9. Evaluation of the proposed approach and comparison against histogram-based matching [7]. The average PSNR gains range from 0.66 dB (*Flamenco2* sequence) to 1.34 dB (*Objects2* sequence), and the average SSIM gains range from 1.0% (*Flamenco2* sequence) to 3.2% (*Objects2* sequence)

all sequences with 640×480 resolution, the execution times for matching, matrix optimization and the total time for one frame are about 0.12s, 0.88s and 3.0s, respectively. For the higher resolution *Ballet* sequence, the total computation time per frame is about 8.3s. One concludes that the complexity is highly dependent on the video resolution, but also the scene content will have a direct impact since the content determines the number of SURF matches that can be detected.

E. Discussion

It is interesting to investigate what happens if one encounters inaccuracies in the pairwise matching method. First of all, we have to observe that the mismatched points can be effectively filtered out by our approach (see Fig. 2(c)). Thus, the influence of such outliers is significantly reduced. Secondly, after filtering, the set of feature points is relatively sparse, hence, eventual pairwise matching inaccuracies have a small relative importance in the overall cost function (Eq. (5)). To assess this effect, we have performed an experiment by introducing random errors in inter-view scene mapping. We randomly select a given subset of inter-view matching pairs and we modify the locations of the selected points in the target view by adding small random offsets; the distribution of this noise component is a truncated Gaussian with a variance of up to 1 pixel, and we set the maximum offset to be of 2 pixels. This noise component substantially influences the PSNR on the feature points, with an average decrease of approximately 4 dBs, but the SSIM on the global frame is only marginally affected, with an average decrease of only 0.01%. It is important to observe, however, that this is due to the use of a sparse feature point set. Increasing the number of feature points (up to a dense set) using a different matching technique - e.g. optical flow - increases the relative importance of feature-point matching in the overall cost function. The

lack of accurate matching, simulated here by the use of noise perturbation, is also expected to significantly affect the results. Adopting dense and accurate matching techniques in our approach (for instance matching based on multiple previous frames) is left as topic of future investigation.

The proposed color correction approach can be extended to incorporate more complicated reference view selection models. Some more complicated selection of the reference view could be introduced by considering all captured views. To handle them using our optimization solution, the target texture structure consistency energy and the target color consistency energy for the temporal correspondence pairs remain the same as in Eq. 5, while the spatial color consistency energy should be adjusted to match to the desired camera view and baseline distance between reference and target cameras.

Limitations. Although the proposed method enables high quality color correction, there are a few issues that should be further investigated. First, as can be seen in the visualized difference figure, the gradient errors generated by our approach frequently happen around the border of the output image. This is because we need to calculate in every pixel first derivatives of the gradient and around the border some pixels' inaccurate gradient will result in the propagation of errors to their neighbors by the optimization solution. This drawback would be fixed by extrapolating (e.g., mirror padding) and cropping the image before and after the color correction respectively. In addition, we take into account the temporal color transition according to the original color difference between successive frames from the input camera. Furthermore, we do not consider in our implementation the transition term between different cameras. By introducing more complicated models (e.g., light field or reflection analysis) and setting appropriate spatio-temporal color transition terms in our data term (Eq. (8)), the correction result can be expected to

improve even more, in particular for large baselines between reference and target cameras. Robustly adapting our approach in some extreme situations (such as shot transitions) also needs more complicated content-aware modeling. Finally, our implementation is currently not optimized for speed as it runs on a single thread. Performing the computations on chroma subsampled data and following a multi-scale approach could greatly accelerate our system.

V. CONCLUSIONS

This paper presents a novel multiview video color correction approach based on spatio-temporal global energy optimization. The proposed method aims at maintaining the original image gradient consistency and providing spatio-temporal color correspondence preservation, and achieves these goals by solving a Laplacian matrix optimization problem. Experiments show that the proposed multiview color correction algorithm can effectively correct scene color while preserving the structure information for multiple cameras.

The presented work suggests some avenues for future research. For example, we would like to investigate some improved prediction schemes for multiview video coding systems in order to further improve the compression performance in multiview video via color correction. Several techniques making use of histogram matching have investigated this idea in the past [7]; it would also be desirable to improve the compression performance in multiview video coding based on the proposed color correction approach. This is left as topic of further investigation.

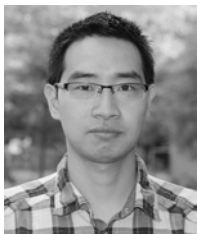
ACKNOWLEDGMENTS

The authors would like to thank the editor and anonymous reviewers for their insightful comments in improving the paper. Many thanks to S.A. Fezza, M.-C. Larabi and F. Shao for sharing their images of corresponding papers. This work was supported by the iMinds visualization research program (HIVIZ), iMinds vzw and IWT in the context of the ASPRO+ project.

REFERENCES

- [1] A. Kubota, A. Smolic, M. Magnor, M. Tanimoto, T. Chen, and C. Zhang, "Multiview imaging and 3DTV," *IEEE Signal Process. Mag.*, vol. 24, no. 6, pp. 10–21, 2007.
- [2] A. Vetro, T. Wiegand, and G. J. Sullivan, "Overview of the stereo and multiview video coding extensions of the H. 264/MPEG-4 AVC standard," *Proc. IEEE*, vol. 99, no. 4, pp. 626–642, 2011.
- [3] H.-Y. Shum, S.-C. Chan, and S. B. Kang, *Image-Based Rendering*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.
- [4] M. Pölonen, J. Hakala, R. Bilcu, T. Jävenpää, J. Häkkinen, and M. Salmimaa, "Color asymmetry in 3d imaging: Influence on the viewing experience," *3D Research*, vol. 3, no. 3, pp. 1–10, 2012.
- [5] J. Chen, J. Zhou, J. Sun, and A. C. Bovik, "Binocular mismatch induced by luminance discrepancies on stereoscopic images," in *Int. Conf. Multimedia Expo*, 2014, pp. 1–6.
- [6] M. Salmimaa, J. Hakala, M. Pölonen, 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.
- [7] U. Fecker, M. Barkowsky, and A. Kaup, "Histogram-based prefiltering for luminance and chrominance compensation of multiview video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 9, pp. 1258–1267, 2008.
- [8] H. Jae-Ho, C. Sukhee, and L. Yung-Lyul, "Adaptive local illumination change compensation method for h.264/avc-based multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1496–1505, 2007.
- [9] C. Doutre and P. Nasiopoulos, "Color correction preprocessing for multiview video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 9, pp. 1400–1406, 2009.
- [10] A. P. Hekstra, J. Beerends, D. Ledermann, F. De Caluwe, S. Kohler, R. Koenen, S. Rihs, M. Ehrsam, and D. Schlauss, "PVQM - a perceptual video quality measure," *Signal Process. Image Commun.*, vol. 17, no. 10, pp. 781–798, 2002.
- [11] S. A. Fezza, M.-C. Larabi, and K. M. Faraoun, "Feature-based color correction of multiview video for coding and rendering enhancement," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 24, no. 9, pp. 1486–1498, 2014.
- [12] A. Ilie and G. Welch, "Ensuring color consistency across multiple cameras," in *Proc. Int. Conf. Comput. Vision*, vol. 2, Beijing, China, 2005, pp. 1268–1275.
- [13] J. Zhong, B. Kleijn, and X. Hu, "Camera control in multi-camera systems for video quality enhancement," *IEEE Sensors Journal*, vol. 14, no. 9, pp. 2955–2966, 2013.
- [14] W. Kim, J. Kim, M. Choi, I.-J. Chang, and J. Kim, "Low complexity image correction using color and focus matching for stereo video coding," in *Proc. Int. Symp. Circuits Syst.*, Beijing, China, 2013, pp. 2912–2915.
- [15] B. Shi, Y. Li, L. Liu, and C. Xu, "Block-based color correction algorithm for multi-view video coding," in *Proc. Int. Conf. Multimedia and Expo*, New York City, United States, 2009, pp. 65–68.
- [16] K. Yamamoto and R. Oi, "Color correction for multi-view video using energy minimization of view networks," *Int. J. Autom. Comput.*, vol. 5, no. 3, pp. 234–245, 2008.
- [17] M. Panahpour Tehrani, A. Ishikawa, S. Sakazawa, and A. Koike, "Iterative colour correction of multicamera systems using corresponding feature points," *J. Visual Commun. Image Represent.*, vol. 21, no. 5, pp. 377–391, 2010.
- [18] C. Mouffranc and V. Nozick, "Colorimetric correction for stereoscopic camera arrays," in *Proc. Asian Conf. Comput. Vision*. Daejeon, Korea: Springer, 2012, pp. 206–217.
- [19] Q. Wang, P. Yan, Y. Yuan, and X. Li, "Robust color correction in stereo vision," in *Proc. Int. Conf. Image Process.*, Brussels, Belgium, 2011, pp. 965–968.
- [20] F. Shao, G. Jiang, M. Yu, and K. Chen, "A content-adaptive multi-view video color correction algorithm," in *Proc. Int. Conf. Acoust. Speech Signal Process.*, vol. 1, Honolulu, Hawaii, 2007, pp. 969–972.
- [21] F. Shao, G.-Y. Jiang, M. Yu, and Y.-S. Ho, "Fast color correction for multi-view video by modeling spatio-temporal variation," *J. Visual Commun. Image Represent.*, vol. 21, no. 5, pp. 392–403, 2010.
- [22] U. Fecker, M. Barkowsky, and A. Kaup, "Improving the prediction efficiency for multi-view video coding using histogram matching," in *Picture Coding Symposium (PCS)*, Beijing, China, April 2006.
- [23] Y. Chen, C. Cai, and J. Liu, "Yuv correction for multi-view video compression," in *Proc. Int. Conf. Pattern Recognit.*, vol. 3, Hong Kong, China, 2006, pp. 734–737.
- [24] K. Li, Q. Dai, and W. Xu, "Collaborative color calibration for multi-camera systems," *Signal Process. Image Commun.*, vol. 26, no. 1, pp. 48–60, 2011.
- [25] Y.-L. Lee, J. Hur, Y. Lee, K. Han, S. Cho, N. Hur, J. Kim, J. Kim, P. Lai, and A. Ortega, "Ce11: Illumination compensation," in *Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, Document JVT-U052r2*, Hangzhou, China, Oct. 2006, pp. 20–27.
- [26] K. Yamamoto, M. Kitahara, H. Kimata, T. Yendo, T. Fujii, M. Tanimoto, S. Shimizu, K. Kamikura, and Y. Yashima, "Multiview video coding using view interpolation and color correction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 11, pp. 1436–1449, 2007.
- [27] B. Shi, Y. Li, L. Liu, and C. Xu, "Color correction and compression for multi-view video using H.264 features," in *Proc. Asian Conf. on Comput. Vision*. Queenstown, New Zealand: Springer, 2009, pp. 43–52.
- [28] D. Lischinski, Z. Farbman, M. Uyttendaele, and R. Szeliski, "Interactive local adjustment of tonal values," *ACM Trans. Graph.*, vol. 25, pp. 646–653, 2006.
- [29] P. Bhat, C. L. Zitnick, M. Cohen, and B. Curless, "Gradientshop: A gradient-domain optimization framework for image and video filtering," *ACM Trans. Graph.*, vol. 29, no. 2, p. 10, 2010.
- [30] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski, "Edge-preserving decompositions for multi-scale tone and detail manipulation," *ACM Trans. Graph.*, vol. 27, pp. 67:1–10, 2008.

- [31] Y. Lei, W. Luo, Y. Wang, and J. Huang, "Video sequence matching based on the invariance of color correlation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 9, pp. 1332–1343, 2012.
- [32] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Comput. Graph. Appl.*, vol. 21, no. 5, pp. 34–41, 2001.
- [33] X. Xiao and L. Ma, "Gradient-preserving color transfer," *Comput. Graph. Forum*, vol. 28, no. 7, pp. 1879–1886, 2009.
- [34] W. Xu and J. Mulligan, "Performance evaluation of color correction approaches for automatic multi-view image and video stitching," in *Proc. Conf. Comput. Vision Pattern Recognit.*, San Francisco, United States, 2010, pp. 263–270.
- [35] A. Gijsenij, T. Gevers, and J. van de Weijer, "Computational color constancy: Survey and experiments," *IEEE Trans. Image Process.*, vol. 20, no. 9, pp. 2475–2489, 2011.
- [36] J. Koenderink and A. van Doorn, "Gauge fields in pictorial space," *SIAM J. Imag. Sci.*, vol. 5, no. 4, pp. 1213–1233, 2012.
- [37] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool, "Speeded-up robust features (SURF)," *Comput. Vision Image Understanding*, vol. 110, no. 3, pp. 346–359, 2008.
- [38] S.-P. Lu, B. Ceulemans, A. Munteanu, and P. Schelkens, "Performance optimizations for patchmatch-based pixel-level multiview inpainting," in *Proc. Int. Conf. 3D Imaging*. Liege, Belgium: IEEE, 2013, pp. 1–7.
- [39] D. Scharstein and R. Szeliski, "Middlebury stereo evaluation - version 2." [Online]. Available: <http://vision.middlebury.edu/stereo/>
- [40] D. Krishnan, R. Fattal, and R. Szeliski, "Efficient preconditioning of laplacian matrices for computer graphics," *ACM Trans. Graph.*, vol. 32, no. 4, pp. 142:1–15, 2013.
- [41] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.



Shao-Ping Lu received the Ph.D. degree in Computer Science from Tsinghua University, China, in July 2012. From November 2012, he has been a Postdoctoral Researcher at Vrije Universiteit Brussels (VUB), Brussels, Belgium. His primary research areas are image & video processing and editing, with particular interests in multi-view video construction, representation and compression.



Beerend Ceulemans received the BSc degree in electrical and information technology engineering in 2011 and the MSc in computer science in 2013, both from the Vrije Universiteit Brussel (VUB). In September 2013, he started as a Ph.D candidate in the Department of Electronics and Informatics (ETRO), VUB. His research interest are centered on virtual viewpoint synthesis for autostereoscopic 3D screens and free-viewpoint video and involve image and video processing, computer vision, statistical signal processing and machine learning.



Adrian Munteanu (M'07) received the M.Sc. degree in electronics from Politehnica University of Bucharest, Romania, in 1994, the M.Sc. degree in biomedical engineering from University of Patras, Greece, in 1996, and the Doctorate degree in applied sciences (awarded with the highest distinction and congratulations of the jury members) from Vrije Universiteit Brussel (VUB), Belgium, in 2003.

In the period 2004–2010 he was post-doctoral fellow with the Fund for Scientific Research - Flanders (FWO), Belgium, and since 2007, he is professor at VUB. Adrian Munteanu is also research leader of the 4Media group at iMinds institute in Belgium. His research interests include scalable image and video coding, distributed video coding, scalable coding of 3D graphics, 3D video coding, error-resilient coding, multiresolution image and video analysis, video segmentation and indexing, multimedia transmission over networks and statistical modeling.

Dr. Munteanu is the author or co-author of more than 200 journal and conference publications, book chapters, patent applications and contributions to standards. He is the recipient of the 2004 BARCO-FWO prize for his PhD work. Dr. Munteanu currently serves as Associate Editor for IEEE Transactions on Multimedia.



Peter Schelkens (M'97) obtained an electrical engineering degree in applied physics, a biomedical engineering degree (medical physics), and a PhD degree in applied sciences from the Vrije Universiteit Brussel (VUB). He currently holds a professorship at the Department of Electronics and Informatics (ETRO) at the Vrije Universiteit Brussel (VUB) and is research director at the iMinds research institute. He is also member of the board of councillors of the microelectronics research center Imec in Leuven.

The research interests of Peter Schelkens are situated in the field of multidimensional signal processing while especially focussing on cross-disciplinary research. In 2013, he obtained an EU ERC Consolidator Grant focusing on digital holography.

He (co-)authored of more than 200 journal and conference publications and is co-editor of the books, "The JPEG 2000 Suite" and "Optical and Digital Image Processing", published respectively in 2009 and 2011 by Wiley. Since 2014, he is associate editor of the IEEE Transactions on Circuits and Systems for Video Technology. He is elected committee member of the IEEE Image, Video, and Multidimensional Signal Processing (IVMSP) Technical Committee and the IEEE Multimedia Signal Processing (MMSP) Technical Committee. Peter Schelkens is participating in the ISO/IEC JTC1/SC29/WG1 (JPEG) and WG11 (MPEG) standardization activities. From 2012 onwards he is acting as rapporteur/chair of JPEG Coding and Analysis Technologies subgroup. Peter Schelkens is also co-founder of the spin-off company, Universum Digitalis (www.universumdigitalis.com).