Dissecting and Reassembling Color Correction Algorithms for Image Stitching

Fabio Bellavia and Carlo Colombo

***Abstract*— This paper introduces a new compositional framework for classifying color correction methods according to their two main computational units. The framework was used to dissect fifteen among the best color correction algorithms and the computational units so derived, with the addition of four new units specifically designed for this work, were then reassembled in a combinatorial way to originate about one hundred distinct color correction methods, most of which never considered before. The above color correction methods were tested on three different existing datasets, including both real and artificial color transformations, plus a novel dataset of real image pairs categorized according to the kind of color alterations induced by specific acquisition setups. Differently from previous evaluations, special emphasis was given to effectiveness in real world applications, such as image mosaicing and stitching, where robustness with respect to strong image misalignments and light scattering effects is required. Experimental evidence is provided for the first time in terms of the most recent perceptual image quality metrics, which are known to be the closest to human judgment. Comparative results show that combinations of the new computational units are the most effective for real stitching scenarios, regardless of the specific source of color alteration. On the other hand, in the case of accurate image alignment and artificial color alterations, the best performing methods either use one of the new computational units, or are made up of fresh combinations of existing units.**

***Index Terms*— Color correction, compositional framework, image stitching, image mosaicing.**

I. Introduction

COLOR correction is an image processing technique with several applications, from photometric registration in image mosaicing and stitching [1] to image enhancement and recoloring for visual effects generation [2]. Its aim is to transfer color properties from a source image to a target image. In applications such as image mosaicing, it is also required that color attributes remain consistent with the image geometric structures, so as to prevent alterations of the original image content at the semantic level. For this purpose, a preliminary geometric alignment of the input images is required to obtain color correspondences. Despite the recent progress in image stitching [3], [4], only a coarse image alignment is often obtained in practical situations, giving rise to wrong color correspondences that can remarkably affect the final results.

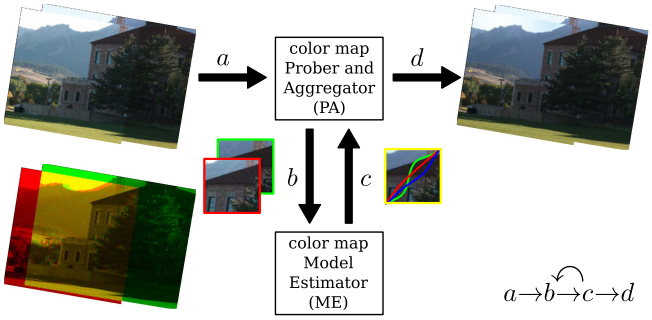


Fig. 1. Color mapping CUs. Source and target images are shown superimposed before and after color correction and as anaglyphs to emphasize image misalignments. PA determines the image subregions that serve as input to ME, and combines the ME output color maps to obtain the final result (best viewed in color and zoomed in).

Several color correction algorithms have been proposed and analyzed in the last few years. Their strengths and weaknesses have been outlined in recent surveys [5] and evaluations [6], that classify color correction methods into *model-based parametric* vs *model-less non-parametric*, or *local* vs *global* approaches (see Sec. II).

A. Paper Contributions

In this paper, we introduce a *compositional framework* for classifying color correction methods in a new way. The idea stems from the observation that any color correction method can be decomposed into two main *Computational Units (CUs)*. These are (1) the low-level color map *Model Estimator (ME)*, that actually computes the color maps, and (2) the high-level color map *Prober and Aggregator (PA)*, that organizes, combines and applies the color maps. As shown in the block diagram of Fig. 1, the PA unit (a) receives as input an image pair, computes sets of pixel correspondences and (b) outputs them to the ME unit, then (c) inputs from ME one or more color maps, and finally (d) provides as output the corrected image. The two CUs are distinct yet mutually interdependent, providing input data to each other in steps (b)-(c), that can be iterated according to the PA used.

The main contributions of this work arise from the above framework, which allowed us both to investigate existing methods from a new perspective, and to develop more effective solutions to the color correction problem. Specifically:

– 15 among the most successful literature approaches were revisited and categorized according to the specific

Manuscript received June 27, 2016; revised February 22, 2017, June 26, 2017, and September 6, 2017; accepted September 24, 2017. Date of publication September 27, 2017; date of current version November 14, 2017. This work was supported in part by the SUONO project (SCN_00306) through the Italian Ministry of Education and Research. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Keigo Hirakawa. *(Corresponding author: Fabio Bellavia.)*

The authors are with the Department of Information Engineering, University of Florence, 50139 Firenze, Italy (e-mail: fabio.bellavia@unifi.it; carlo.colombo@unifi.it).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The pdf provides additional material about dissecting and reassembling color correction algorithms for image stitching. The total size of the file is 29.2 MB. Contact bellavia.fabio@gmail.com for further questions about this work.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIP.2017.2757262

1057-7149 © 2017 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission.
See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

ME/PA pair they employ. The analysis revealed that many methods share either one of their CUs. As a result, only 12 distinct MEs and 5 distinct PAs were found, to which four novel CUs (2 MEs and 2 PAs), expressly designed for this work, were added. Sec. III addresses the above computational unit categorization and design.

– The combinatorial nature of our compositional framework led us to perform an exhaustive comparative evaluation of all the color correction methods that can be assembled with the available MEs and PAs, for a total of $14 \times 7 = 98$ different methods, 83 of which never considered before (see Sec. IV).

As anticipated, another contribution of this paper is the design of four novel CUs (2 MEs and 2 PAs), which according to the experiments can be used to build the best performing algorithms. The first ME, named *GPS (Gradient Preserving Spline)*, employs a monotone cubic spline to locally model the correction function. It takes into account not only the color values of corresponding image pixels, but also the gradient of both the source and target images so as to preserve the image structure. The second ME, referred to as *FGPS (Fast GPS)*, introduces an approximated yet faster coarse-to-fine spline search space reduction with respect to GPS, thus extending the idea originally presented in [7]. Of the two PAs, the *Linear Color Propagation (LCP)* CU is based on the approach first introduced in [7], now also integrating a global color map estimation step and further refinements. The other PA, called *Best Local to Global (BLG)*, extracts local color maps from the input data and globally selects and combines the best ones to form a final color palette. Both PAs can infer local color properties unattainable by global methods, and then propagate global color models to the non-overlapping area of the target image.

As a final contribution of the paper, a very thorough experimental evaluation of the 98 color correction methods was carried out. In particular, besides testing with 123 image pairs from three different existing datasets [2], [6], [7], we further experimented with a novel dataset of 127 real image pairs (thus increasing the number of tested image pairs to 250), created ad hoc for this work. Image pairs from this dataset are classified according to the four main acquisition setups giving rise to color alterations. Unlike previous evaluations, specific tests were done in order to assess the robustness of the evaluated methods in the presence of hard color inconsistencies, such as those due to image misalignments and light scattering effects, which is a critical aspect in real world applications such as image stitching. Moreover, unlike previous evaluations, the image quality metrics used for all the experiments were the recent state-of-the-art *improved Color Image Difference (iCID)* measure [8] and the *Feature Similarity (FSIM)* index [9], which are known to be the closest to human judgment.

The remainder of the paper is organized as follows. Sec. II addresses related work on color correction. Computational units are introduced in Sec. III, and results are discussed in Sec. IV. Finally, conclusions are drawn and future work is outlined in Sec. V.

II. Related Work

Recent surveys on color correction algorithms can be found in [5] and [6]. Color correction techniques can be classified into *model-based parametric* and *model-less non-parametric* approaches. The former assume a known color distribution model for both the source and target images, to be inferred from the input data. Conversely, no explicit assumptions are made for model-less non-parametric methods, usually inferred directly from color histograms and employing a look-up table to record the color map. Color correction methods can also be divided into *global* and *local* approaches. While in global approaches a single color transfer function is estimated and applied to the whole image, in local approaches multiple color maps are computed for different areas of the image, previously segmented according to their spatial and chromatic characteristics.

The earliest work on color correction can be traced back to Reinhard [10]. This global approach consists in rearranging the color distribution of the target image to have the same mean and variance of the source image. Color space is first converted into the $l\alpha\beta$ space representation in order to decorrelate color channels, on which to independently apply the transformation. An alternative solution is proposed in [11], where a linear transformation is applied to decorrelate color channels.

Local approaches give generally better and more accurate results on complex scenes [12], since the assumption that a single global color palette is sufficient to cover all the color maps is often unrealistic. To overcome this issue, Reinhard's method can be extended in order to obtain a more accurate color map as the combination of several weighted color maps after segmenting the image into several regions [11], [13], [14]. This is achieved, for instance, with mean shift or soft color segmentations.

A very popular model-based color correction approach is gain compensation [1], originally introduced to address symmetric color balancing in panoramic mosaicing by a least-square minimization. This method was further improved by introducing block-wise smooth multiple models [15]. Other model-based approaches fit linear or polynomial least-squares transformations, working simultaneously on all color channels [16]. Models based on weighted affine transformations [2], splines [17], Gaussian mixture models [18] and nonlinear manifold learning approaches [19] have also been proposed.

Straight model-less color histogram transformations can be derived by histogram matching [20], and further refined using segmentation and Bayesian inference [21]. Since histogram matching is a channel-wise operation, the Radon transform is used in [22] to define one-dimensional subspaces on which to apply the histogram matching; then, back-projection is applied to return back to the original color space. Color histogram peaks can also be used to define image 'principal regions' [23], modeling a polynomial mapping function between corresponding color histogram peaks of the input images.

The monotone constraint is often imposed on the color map, since it holds for the camera radiometric response function that models the image formation. This constraint is used in the case

of spline-based color map models [17] but also for model-less approaches. In particular, tensor voting [24] and maximum likelihood estimation of the brightness transfer function [25] model the color correction map according to this constraint.

Color mapping requires to find color correspondences between the source and target images. In the case of local methods, each segmented region should be sufficiently wide to tolerate color correspondence mismatches that can occur due to coarse image alignment [21]. Keypoint-based matching can be used to define and grow corresponding image regions instead of segmenting the images [26], thus bypassing the rigid transformation constraint imposed by mosaicing, but also increasing the risk of color mismatches.

III. Compositional Framework

According to our compositional framework, color correction algorithms can be organized into two main CUs (see again Fig. 1). The low-level ME is the inner core of the algorithm. Given as input a set $\mathcal{P} = \{(I_s(\mathbf{x}), I_t(\mathbf{x})\}$ of spatially referenced corresponding color pairs, where $I_s(\mathbf{x})$ and $I_t(\mathbf{x})$ are the color values at $\mathbf{x}$ in the source and target images, respectively, ME generates as output a color map $C_{\mathcal{P}}$. Assuming that input and output are coarsely registered 24 bit RGB color images, $C_{\mathcal{P}}$ is defined as

$$C_{\mathcal{P}} : \mathbb{R}^3 \rightarrow \mathbb{R}^3 = U \circ \mathrm{M} \circ T \tag{1}$$

where $T : \mathbb{R}^3 \rightarrow \mathbb{R}^n$, $U : \mathbb{R}^m \rightarrow \mathbb{R}^3$ are invertible, fixed, space projection functions and $\mathrm{M} \in \mathbb{R}^{m \times n}$ is a matrix inducing a linear map. On the other hand, PA provides ME with multiple color correspondence sets $\mathcal{P}$, combines and applies the output color maps $C_{\mathcal{P}}$, thus synthesizing from the spatially registered input images I_s and I_t the final corrected source image I_c. For example, a global PA computes only a single color correspondence set $\mathcal{P}$ of overlapping pixels between the images, and applies the resulting color map $C_{\mathcal{P}}$ to the whole source image I_s.

In the next subsection, the most relevant color correction methods are revised according to our compositional framework, and the T, U and M functions defining the corresponding MEs are explicitly outlined and reported. Analogously, all PAs are discussed in detail in Sec. III-B. A summary of all the MEs and PAs analyzed in this paper is reported in Table I for the readers' convenience.

A. Color Map Model Estimators (MEs)

1) Reinhard's (R) [10]: The function

$$T : [R\, G\, B]^{\mathrm{T}} \mapsto [l\, \alpha\, \beta\, 1]^{\mathrm{T}} \tag{2}$$

maps RGB to $l\alpha\beta$ space in homogeneous coordinates, $U = T^{-1}$, and $\mathrm{M} \in \mathbb{R}^{4\times4}$ is computed as

$$\mathrm{M} = \begin{bmatrix} \frac{\sigma_t^l}{\sigma_s^l} & 0 & 0 & \mu_t^l - \mu_s^l \frac{\sigma_t^l}{\sigma_s^l} \\ 0 & \frac{\sigma_t^\alpha}{\sigma_s^\alpha} & 0 & \mu_t^\alpha - \mu_s^\alpha \frac{\sigma_t^\alpha}{\sigma_s^\alpha} \\ 0 & 0 & \frac{\sigma_t^\beta}{\sigma_s^\beta} & \mu_t^\beta - \mu_s^\beta \frac{\sigma_t^\beta}{\sigma_s^\beta} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

TABLE I
Computational Units Summary

ME	color map Model Estimator	Section III-A
R	Reinhard's	1
CS	Correlated Space	2
GC	Gain Compensation	3
3M	3×3 Map	4
PM	2^{nd} Order Polynomial Map	5
AM	Affine Map	6
PR	Principal Regions	7
HM	Histogram Matching	8
TG	Truncated Gaussian	9
G	Gaussian	10
MS	Monotone Spline	11
3MS	Correlated Monotone Spline	12
GPS	Gradient Preserving Spline	13
FGPS	Fast Gradient Preserving Spline	14

(a)

PA	color map Prober and Aggregator	Section III-B
GL	Global	1
P	Pitié's	2
CIM	Color Influence Map	3
TS	Soft Tai's Segmentation	4
MSS	Mean Shift Segmentation	5
LCP	Linear Color Propagation	6
BLG	Best Local to Global	7

(b)

where μ_i^k and σ_i^k are respectively the mean and standard deviation for channel k and image $i \in \{s, t\}$ in the input set $\mathcal{P}$.

2) Correlated Space (CS) [11]: The function

$$T : [R\, G\, B]^{\mathrm{T}} \mapsto [R\, G\, B\, 1]^{\mathrm{T}} \tag{4}$$

converts its input into homogeneous coordinates, $U = T^{-1}$, and

$$\mathrm{M} = \mathrm{A}_t^{-1} \mathrm{R}_t^{\mathrm{T}} \mathrm{D}_t^{-1/2} \mathrm{D}_s^{1/2} \mathrm{R}_s \mathrm{A}_s \tag{5}$$

where $C_i = \mathrm{R}_i^{\mathrm{T}} \mathrm{D}_i \mathrm{R}_i$ is the eigendecomposition of the autocovariance matrix $C_i \in \mathbb{R}^{3\times3}$ provided by $\mathcal{P}$ for image I_i. The matrix

$$\mathrm{A}_i = \begin{bmatrix} 1 & 0 & 0 & -\mu_i^R \\ 0 & 1 & 0 & -\mu_i^G \\ 0 & 0 & 1 & -\mu_i^B \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

translates its input into a zero mean distribution.

3) Gain Compensation (GC) [1]: T, U are the identity function, and

$$\mathrm{M} = \begin{bmatrix} g^R & 0 & 0 \\ 0 & g^G & 0 \\ 0 & 0 & g^B \end{bmatrix} \tag{7}$$

where the gain g^k for channel k is computed as

$$g^k = \frac{\mu_s^k \mu_t^k \sigma_g^2 + N^2 \sigma_N^2}{{\mu_t^k}^2 \sigma_g^2 + N^2 \sigma_N^2} \tag{8}$$

with $N = |\mathcal{P}|$, and σ_g, σ_N two fixed constants that avoid the zero solution. This formulation is derived from the original

symmetric minimization error [1] by setting to 1 the gain for the target image.

4) 3 × 3 Map (3M) [16]: T, U are the identity function, while $\mathrm{M} \in \mathbb{R}^{3\times 3}$ is obtained by least squares minimization so that

$$\begin{bmatrix} R_t \\ G_t \\ B_t \end{bmatrix} = \mathrm{M} \begin{bmatrix} R_s \\ G_s \\ B_s \end{bmatrix} \tag{9}$$

where $I_i(\mathbf{x}) = [R_i\, G_i\, B_i]^{\mathrm{T}}$ and $(I_s(\mathbf{x}), I_t(\mathbf{x})) \in \mathcal{P}$ is a color correspondence.

5) 2^{nd} Order Polynomial Map (PM) [16]: $\mathrm{M} \in \mathbb{R}^{3\times 7}$ is computed analogously to 3M, but on a different space, since

$$T : [R\, G\, B]^{\mathrm{T}} \mapsto [R^2\, G^2\, B^2\, R\, G\, B\, 1]^{\mathrm{T}} \tag{10}$$

and U is the identity function.

6) Affine Map (AM) [2]: T maps to homogeneous coordinates as in Eq. (4), $U = T^{-1}$, and the matrix M, obtained by least square minimization, is constrained to be an affine map in $\mathbb{R}^3$.

7) Principal Regions (PR) [23]: T projects into a higher dimension $l\alpha\beta$ space

$$T : [R\, G\, B]^{\mathrm{T}} \mapsto [l^2\, \alpha^2\, \beta^2\, l\, \alpha\, \beta\, 1]^{\mathrm{T}} \tag{11}$$

while $U : \mathbb{R}^3 \to \mathbb{R}^3$ maps $l\alpha\beta$ to RGB. M is computed according to the 'principal regions' of images [23]. Principal regions are defined according to the 3 highest peaks in the hue histogram of the image. For each channel, the average color value for corresponding principal regions between the two images is used to compute a polynomial mapping function, yielding a matrix M of the form

$$\mathrm{M} = \begin{bmatrix} a & 0 & 0 & b & 0 & 0 & c \\ 0 & d & 0 & 0 & e & 0 & f \\ 0 & 0 & g & 0 & 0 & h & i \end{bmatrix} \tag{12}$$

8) Histogram Matching (HM) [20]: For each 8-bit RGB channel k a lookup table $H^k(x) = y$ is defined, which maps all the possible 256 color values according to the cumulative distribution of the corresponding color value channels as described in [20]. In this case, the function

$$\begin{aligned} &T : [R\, G\, B]^{\mathrm{T}} \\ &\quad \mapsto [0 \cdots 0\, 1_{R+1}\, 0 \cdots 0\, 1_{G+256}\, 0 \cdots 0\, 1_{B+512}\, 0 \cdots 0]^{\mathrm{T}} \end{aligned} \tag{13}$$

maps RGB to a $256 \times 3 = 768$ binary space, i.e. to a vector which is zero everywhere except at the positions defined by the RGB value. U can be written as the block matrix $\mathrm{U} \in \mathbb{R}^{3\times 768}$

$$\mathrm{U} = \begin{bmatrix} \mathbf{v} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{v} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{v} \end{bmatrix} \tag{14}$$

where $\mathbf{0} \in \mathbb{R}^{1\times 256}$ is a zero vector and $\mathbf{v} = [0\, 1\, \ldots\, 255]$. The binary block matrix M is defined as

$$\mathrm{M} = \begin{bmatrix} \mathrm{L}^R & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathrm{L}^G & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathrm{L}^B \end{bmatrix} \tag{15}$$

where each $\mathrm{L}^k \in \mathbb{R}^{256\times 256}$ is such that

$$\mathrm{L}^k_{i,j} = \begin{cases} 1 \text{ if } H^k(j) = i \\ 0 \text{ otherwise} \end{cases} \tag{16}$$

9) Truncated Gaussian (TG) [21]: Lookup tables H^k are derived from local joint image histograms modeled as collections of truncated Gaussians using a maximum likelihood estimation procedure. These lookup tables are used to define the functions T, U and M as for HM (see Eqs. (13)-(16)).

10) Gaussian (G) [21]: This ME differs from TG only for the use of the classical Gaussian distribution instead of the truncated Gaussian distribution.

11) Monotone Spline (MS) [17]: Lookup tables H^k are derived according to a channel-wise mapping into splines. In particular, piece-wise cubic splines S^k are estimated from data independently for each channel, so that $H^k(x) = \mathrm{round}(S^k(x))$. Splines are constrained to be monotone and to have 6 knots, two of them fixed so that the color values 0 and 255 map to themselves. Outliers are discarded and the spline is re-estimated to improve the model. Functions T, U and M are defined as for HM.

12) Correlated MS (3MS) [17]: This is the proper color mapping described in the original paper. The spline model obtained by MS is concatenated with 3M to take into account channel correlation.

13) Gradient Preserving Spline (GPS): This is the first of the two MEs specially designed for this work. Lookup tables H^k that map channel-wise to splines are used, and T, U and M are derived as for HM. Although MS also is based on splines, our approach is altogether different as we show in the following. We employ the monotone piecewise Hermite cubic spline interpolation procedure described in [27], with only 4 knots, two of them fixed so as to map the 0 and 255 color values to themselves. For each channel k, we look for the spline minimizing the weighted error sum

$$E^k = w_\mu\, \varepsilon^k_\mu + w_\sigma\, \varepsilon^k_\sigma + \sum_{d\in\{x,y\}} \sum_{m\in\{s,t\}} w_{d,m}\, \varepsilon^k_{d,m} \tag{17}$$

among all the possible splines meeting the previous constraints. The errors

$$\varepsilon^k_\mu = |\mu^k_c - \mu^k_t| \tag{18}$$

$$\varepsilon^k_\sigma = |\sigma^k_c - \sigma^k_t| \tag{19}$$

are respectively the absolute differences between the mean and the standard deviations of the color values of the corrected source image I_c and the target image I_t computed in $\mathcal{P}$ for channel k, and minimize the color distribution in the spirit of Reinhard's method. The term

$$\varepsilon^k_{d,m} = \frac{1}{|\mathcal{P}|} \sum_{\mathcal{P}} \left| \frac{\partial I^k_c}{\partial d}(\mathbf{x}_s) - \frac{\partial I^k_m}{\partial d}(\mathbf{x}_m) \right| \tag{20}$$

in Eq. (17) improves structure similarity with the target image while also preserving the image structure of the source image I_s, by explicitly taking into account the derivatives for each channel k along the direction d. The error weight w_μ in Eq. (17) was set experimentally to 0.5 while the other five were set to 0.1 so that all weights add to unity. According

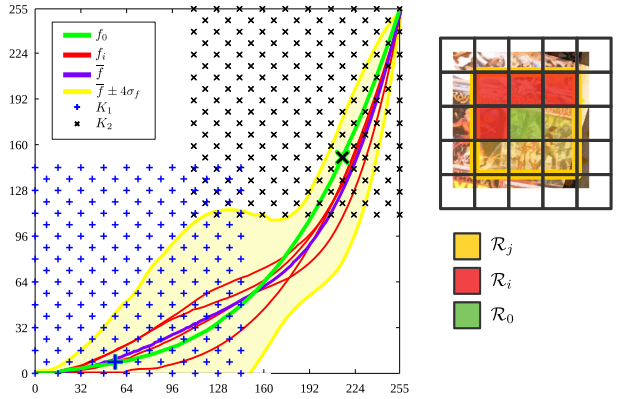


Fig. 2. The checkerboard-like sampling grids (left) for the two knots K_1 and K_2 leading to the spline with minimum error for the correspondence set $\mathcal{P}$. Inside a neighborhood $\mathcal{R}_j$ (right, yellow squares) of the current region $\mathcal{R}_0$ (green square), already computed spline color maps f_i (red splines) for regions $\mathcal{R}_i$ (red squares) can be used to restrain the sampled search space (yellow band) according to their mean $\overline{f}$ (purple spline) and standard deviation σ_f in order to obtain the spline color map f_0 for $\mathcal{R}_0$ (green spline) minimizing the error (best viewed in color and zoomed in).

to our preliminary tests, variations up to 20% around these values do not affect significantly the final result.

Since finding an analytical solution for minimizing the error E^k is not trivial, an exhaustive search for the two free knots defining the spline is carried out. In the case of 8-bit color channels, imposing only the monotone condition and neglecting that different knot pairs can give rise to the same spline color map, this would yield to an upper bound of $q = (n\,(n-1)/2)^2 = n^2(n-1)^2/4$ different error values to test, which for $n = 256$ amounts to about 10^9, an unfeasible number in practice. Nevertheless, both the time spent to evaluate the error E^k on a given spline and the solution search space can dramatically be reduced by employing three suitable heuristics, thus obtaining a near-optimal solution.

The first heuristic is motivated by the observation that, for common images, the error on the mean color value ε_μ^k is dominant in Eq. (17), while the other errors just refine the solution. Therefore, if the error ε_μ^k for the current solution is greater than $\varepsilon_\mu^k + 15$ for the best solution so far, the current solution is discarded, thus avoiding the computation of the full error E^k and saving time.

The second heuristic can be derived by observing that (1) some knot arrangements are unlikely to occur and (2) perturbing the position of a knot changes only slightly the error E^k. In particular, for each knot, instead of a full range value search, we can define two uniform square grids in the range of $[0, 144]$ and $[111, 255]$ respectively, with a step of 8 and a checkerboard-like alternate grid sampling (see Fig. 2). Such sampling scheme reduces the search space to approximately 3×10^4 possible splines, i.e. by five orders of magnitude, while still maintaining a near-optimal solution.

The last heuristic arises from the following observation: In the case of PAs working locally (see Sec. III-B), for each channel, spline color maps $f_j = \mathcal{C}_{\mathcal{P}_j}$ of neighboring regions $\mathcal{R}_j$ must change smoothly, as they are related to region areas with similar color properties. Hence, when computing the color map f_0 of the region $\mathcal{R}_0$, one can take advantage of the already computed neighborhood color maps f_i, $1 \leq i \leq b$ (red splines in Fig. 2). If at least $b \geq 3$ such neighborhood regions exist, then the search space of the two free knots of f_0 (green line) can be statistically constrained within a narrow region (yellow band). Indeed, given a knot in the form $(x, f_0(x))$, the error E^k is evaluated only if it holds

$$|f_0(x) - \overline{f}(x)| \leq h \tag{21}$$

for both knots defining the spline, with

$$h = \min(a, 4\,\sigma_{f(x)}) \tag{22}$$

where $\overline{f}(x)$ and $\sigma_{f(x)}$ are respectively the mean and standard deviation of the already computed $f_i(x)$ mapped values (purple and yellow splines respectively), and $a = 48$ is used to limit the standard deviation value. This heuristic, which typically halves the search space, is used only in combination with LCP and BLG (see Sec. III-B and Fig. 4) since other PAs are either global or do not provide uniform neighborhood regions.

Further speed-up improvements can be obtained with trivial optimizations, including a pre-computation of the spline maps for each knot pair and a parallel implementation of the code.

14) Fast Gradient Preserving Spline (FGPS): In order to reduce even further the spline search space of GPS, a coarse-to-fine approach was designed. The spline search space $\Gamma = \{f_1, f_2, \ldots\}$ is split into several subsets V_r, $r > 0$, each induced by the *representative splines* $f^r \in \Gamma$ through

$$V_r = \{f \in \Gamma : \forall x \;\; |f^r(x) - f(x)| \leq t_0\} \tag{23}$$

for a given threshold t_0. In other words, the subset V_r contains all the splines in the search space Γ that are inside a band of width t_0 around f^r (see Fig. 3). A greedy strategy is adopted to construct the set $F = \{f^r\}$ of representative splines. F, initially containing only the identity color map, is grown by iterating on the whole search space Γ and including at each iteration i the spline f_i under evaluation only if

$$\forall f^r \in F \;\; \exists x : \;\; |f^r(x) - f_i(x)| > t_1 \tag{24}$$

where $t_1 \leq t_0$, i.e. the current spline f_i is included in F only if f_i does not fall completely inside a band of width t_1 for each representative spline f^r already included in F (see Fig. 3). By construction, the union of the subsets V_r covers the whole search space, but it does not form a partition, since non-empty intersections exist between V_r subsets. For each channel k, the error E^k is first evaluated on the set F of representative splines, by choosing the best representative spline $f^{r^\star}$ minimizing the error E^k. Then, only splines in the subset $V_{r^\star}$ are evaluated. According to our experiments, setting $t_0 = 16$ and $t_1 = 12$ gives a good compromise between speed and correctness of the solution, leading to a further reduction of the spline search space ranging from two to eight times.

B. Color Map Probers And Aggregators (PAs)

1) Global (GL): This is the simplest PA. The color map $C_{\mathcal{P}}$ is applied to the whole source image I_s to obtain the corrected output image I_c, where $\mathcal{P} = \{(I_s(\mathbf{x}), I_t(\mathbf{x})) : \mathbf{x} \in I_s \cap I_t\}$ contains the color correspondences in the overlapping area between the source and target images I_s and I_t.

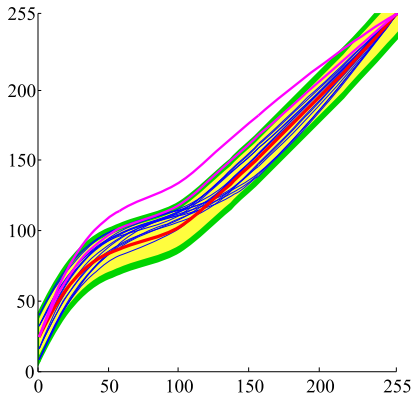


Fig. 3. A representative spline f^r (red) and some splines of its subset V_r (blue), falling inside the band limited by t_0 (green). Other representative splines (purple) cannot fall completely inside the t_1 band (yellow) (best viewed in color and zoomed in).

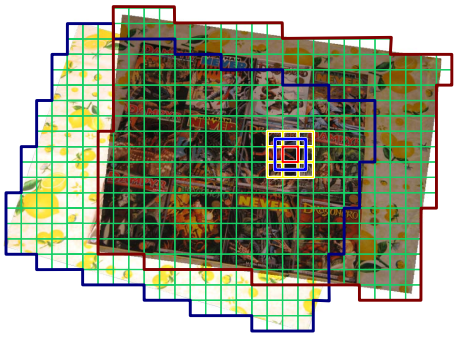


Fig. 4. The whole area $I_s \cup I_t$ is divided by a grid (green) into overlapping image square regions $\mathcal{R}_b$ (blue square), with central cell $\overline{\mathcal{R}}_b$ (red square). Color maps of neighboring cells (yellow squares) can be used to reduce the search space and to speed up the minimization when GPS or FGPS are used, see Sec. III-A.13 (best viewed in color and zoomed in).

2) Pitié's (P) [22]: Originally combined with HM, this PA progressively transforms the source image I_s into the target image I_t by projections onto random orthonormal basis sets, i.e. by multiplying the color values by a rotation matrix $\mathrm{R} \in \mathbb{R}^{3\times3}$. This is aimed at decorrelating the signal in the RGB space in order to support those MEs that operate channel-wise. The input images in the new space $I_{\tilde{s}}$ and $I_{\tilde{t}}$ are then passed to ME, where $\mathcal{P}$ is computed as for GL but in the space induced by the rotation matrix R. The corrected image $I_{\tilde{c}}$ is back-projected into the original RGB space, thus obtaining I_c, that is used in the next iteration as the new estimate of I_s. More specifically, the RGB vector $I_i(\mathbf{x})$ for the pixel $\mathbf{x}$ in the generic image I_i is projected as

$$I_{\tilde{i}}(\mathbf{x}) = \mathrm{R} I_i(\mathbf{x}) \tag{25}$$

With the assumption that

$$I_c(\mathbf{x}) = I_s(\mathbf{x}) + \delta \tag{26}$$

one obtains

$$I_{\tilde{c}}(\mathbf{x}) = \mathrm{R} I_c(\mathbf{x}) = \mathrm{R}(I_s(\mathbf{x}) + \delta) = I_{\tilde{s}}(\mathbf{x}) + \mathrm{R}\delta \tag{27}$$

so that δ can be derived in the least square sense by solving for all the considered $\mathbf{x}$

$$\mathrm{R}\delta = I_{\tilde{c}}(\mathbf{x}) - I_{\tilde{s}}(\mathbf{x}) \tag{28}$$

3) Color Influence Map (CIM) [13]: This PA was originally combined with R. The source image I_s is partitioned into n regions $\mathcal{R}_q$ by mean shift segmentation [28]. The set $\mathcal{P}_q = \{(I_s(\mathbf{x}), I_t(\mathbf{x})) : \mathbf{x} \in \mathcal{R}_q\}$ of corresponding color values for each region $\mathcal{R}_q$ is passed to ME, thus producing n color maps $C_{\mathcal{P}_q}$. The final output image is obtained as the weighted sum of each color map $C_{\mathcal{P}_q}$. In particular, defining the CIM weight w for the region q on the $l\alpha\beta$ space as

$$w(\mathbf{x}, q) = e^{-3\|I_s(\mathbf{x}) - \mu_{\mathcal{R}_q}\|^2} \tag{29}$$

where $\mu_{\mathcal{R}_q}$ is the mean color value of I_s over $\mathcal{R}_q$, the final image I_c is obtained as

$$I_c(\mathbf{x}) = \sum_{q=1}^{n} \frac{w(\mathbf{x}, q) C_{\mathcal{P}_q}(I_s(\mathbf{x}))}{\sum_{\bar{q}=1}^{n} w(\mathbf{x}, \bar{q})} \tag{30}$$

Note that a similar approach using manually segmented regions had already been proposed in [11].

4) Soft Tai's Segmentation (TS) [29]: Designed to work in conjunction with R, this PA operates analogously to CIM. However, weights w are obtained directly with the soft color segmentation algorithm exploiting Gaussian mixture models described in [29].

5) Mean Shift Segmentation (MSS) [21]: Paired in the original paper with TG, this unit considers two distinct mean shift segmentations of I_s. The first segmentation works on the whole image, and produces n_a regions $\mathcal{R}_a$; the second one works only on the overlapping area $I_s \cap I_t$ and produces n_o regions $\mathcal{R}_o$. As with CIM, the n_o sets $\mathcal{P}_o$ corresponding to the regions $\mathcal{R}_o$ are given to ME, that outputs the color maps $C_{\mathcal{P}_o}$. Colors inside each $\mathcal{R}_o$ region are corrected accordingly. Colors inside regions $\mathcal{R}_a$ that are outside the overlap area are corrected according to the closest color map $C_{\mathcal{P}_k}$, where

$$k = \underset{o}{\operatorname{argmin}} \parallel \mu_{\mathcal{R}_a} - \mu_{\mathcal{R}_o} \parallel \tag{31}$$

6) Linear Color Propagation (LCP): This is the first of the two PAs specially designed for this work. This PA completes and refines the blending scheme of [1], applying color correction into non overlapping image areas as the weighted combination of propagated local color maps with a global color map.

The bounding box of $I_s \cup I_t$ is divided by a $g_1 \times g_2$ grid into overlapping 64×64 pixel regions; the grid step is 32 pixels (see Fig. 4). We denote the b-th square region by $\mathcal{R}_b$ and its central 32×32 grid cell by $\overline{\mathcal{R}}_b$. The local color maps $C_{\mathcal{P}_b}$ are derived for each $\overline{\mathcal{R}}_b$ in the overlapping area, where $\mathcal{P}_b = \{(I_s(\mathbf{x}), I_t(\mathbf{x})) : \mathbf{x} \in \mathcal{R}_b, \mathcal{R}_b \subset I_s \cap I_t\}$. In order to improve method robustness especially in the case of bad image registration, $C_{\mathcal{P}_b}$ is discarded if the average rms error between I_c and I_t in $\overline{\mathcal{R}}_b$ exceeds a threshold of 32. Surviving local color maps are then propagated into the non-overlapping area by smoothly mixing them with a global color map.

This global color map, referred to as C_g, is obtained by a weighted average of the local color maps with a channel-wise monotone piecewise Hermite cubic spline. In detail, indicating with superscript k the k-th color channel, internal

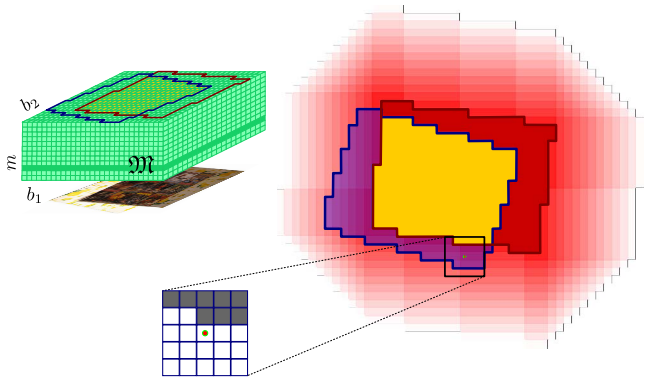


Fig. 5. The binary mask on the target image I_t (red boundaries) is expanded sequentially through morphological dilation, and its intersection with the source image I_s (blue boundaries) is used for propagating color maps and computing the weight mask z (lighter red corresponds to later expansion steps and lower weight values for z). Given the initial $\mathfrak{M}$ (see text) including only color maps for the intersection area $I_s \cap I_t$ (yellow), at each dilation step convolution is applied to each $\mathfrak{M}$ slice (dark green) to propagate color maps. For each slice, only values present at the previous iteration (colors darker than the kernel center on the right) are used. Kernel weights for slice values not included at the previous dilation step are set to zero (white entries) and the kernel is re-normalized (best viewed in color and zoomed in).

knots $(x, f^k(x))$ for the spline f^k are such that

$$f^k(x) = \frac{1}{|\{I_s^k(\mathbf{x}) = x\}|} \sum_b \sum_{\substack{\mathbf{x} \in \overline{\mathcal{R}}_b, \\ I_s^k(\mathbf{x}) = x}} C^k_{\mathcal{P}_b}(I_s(\mathbf{x})) \tag{32}$$

where $\mathbf{x} \in I_s \cap I_t$ and $|\cdot|$ denotes the set cardinality. Note that in the case of a zero denominator in Eq. (32), no knot is defined for x.

Color maps $C_{\mathcal{P}_b}$ are propagated as illustrated in Fig. 5 and discussed hereafter. The grid of Fig. 4 is employed to create a $g_1 \times g_2$ binary mask, with values 1 if the corresponding grid cells include pixels from I_t, and 0 otherwise. Successive expansions of the binary mask are then carried out using the dilation morphology operator [30] with a square 3×3 kernel, until the resulting area covered by 1's is three times the initial area. At each dilation step, color maps $C_{\mathcal{P}_b}$ for the grid cells $\overline{\mathcal{R}}_b$ corresponding to new 1's into the mask are computed by interpolation. For this purpose, assuming from Eq. 1 that $C_{\mathcal{P}_b} = U \circ \mathrm{M}_b \circ T$ with $\mathrm{M}_b \in \mathbb{R}^{m_1 \times m_2}$, a three-dimensional matrix $\mathfrak{M} \in \mathbb{R}^{g_1 \times g_2 \times (m_1 m_2)}$ is created, by stacking the matrices M_b, according to the grid. That is, if $C_{\mathcal{P}_b}$ is defined, $\mathfrak{M}_{b_1,b_2,m}$ is the m-th element of M_b, considered as a column vector, with b_1, b_2 the grid indexes of $\overline{\mathcal{R}}_b$, otherwise $\mathfrak{M}_{b_1,b_2,m} = 0$. Color map values corresponding to the grid cells added at the current mask dilation step are obtained by convolution of each two-dimensional slice of size $g_1 \times g_2$ obtained from $\mathfrak{M}$ by iterating over index m, with a 5×5 re-weighted binomial kernel. The kernel is re-normalized after setting to zero the weight values corresponding to zeros in the binary mask.

Once the global and local color maps have been obtained, they are combined to produce the color corrected image. A weight $z = (\bar{l} - l)/\bar{l}$ is associated to each grid cell, where l is the dilation step at the time the cell was included into the binary mask, and $\bar{l}$ is the final number of dilations made (l is set to zero for all the cells in the overlapping area).

(a) (b)

Fig. 6. (a) The source input image I_s and (b) the final corrected image I_c obtained by using the LCP unit in combination with the AM unit. Both images are superimposed on the input target image I_t for a better understanding (best viewed in color and zoomed in).

The weight map z as well as all slices of $\mathfrak{M}$ are resized $32\times$ to the original image size through bicubic interpolation, thus obtaining for each pixel location $\mathbf{x}$ a weight $z(\mathbf{x})$ and a color map $C_{\mathcal{P}_\mathbf{x}}$. Finally, the corrected source image is computed as

$$I_c(\mathbf{x}) = (1 - z(\mathbf{x}))\, C_g(I_s(\mathbf{x})) + z(\mathbf{x})\, C_{\mathcal{P}_\mathbf{x}}(I_s(\mathbf{x})) \tag{33}$$

i.e. as the combination of the local propagated color maps $C_{\mathcal{P}_b}$ and the global color map C_g, weighted according to the distance from I_t. An example of the final output I_c can be found in Fig. 6.

7) Best Local to Global (BLG): The second original PA computes an interpolated color palette as in [2]. Grid-wise color maps $C_{\mathcal{P}_b}$ are obtained as for LCP and the best cell index

$$d(\mathbf{x}) = \operatorname*{argmin}_b \parallel C_{\mathcal{P}_b}(I_s(\mathbf{x})) - I_t(\mathbf{x}) \parallel \tag{34}$$

is associated to each pair of corresponding color values $(I_s(\mathbf{x}), I_t(\mathbf{x}))$ in the overlapping area. Note that d can differ from the cell index the pixel belongs to.

The color palette is computed explicitly by quantizing the RGB color space. In particular, each color channel is quantized into $q = 21$ levels so that $q^3 = 9261$ quantized colors are obtained. Color similarity $p(\mathbf{c}_i, \mathbf{x})$ between the i-th quantized color $\mathbf{c}_i$ and the RGB value of the pixel in I_s is defined as

$$p(\mathbf{c}_i, \mathbf{x}) = \begin{cases} 1 - \frac{\|\mathbf{c}_i - I_s(\mathbf{x})\|}{Q} & \text{if } \parallel \mathbf{c}_i - I_s(\mathbf{x}) \parallel \leq Q \\ 0, & \text{otherwise} \end{cases} \tag{35}$$

where $Q = 255\sqrt{3}/q$ is the maximal Euclidean distance between two quantized color neighbors in the three-dimensional RGB color space. The color map C_i associated to a quantized color $\mathbf{c}_i$ is the weighted sum of the best color maps $C_{\mathcal{P}_{d(\mathbf{x})}}$ for pixels inside the overlapping area

$$C_i = \sum_{\mathbf{x} \in I_s \cap I_t} \frac{p(\mathbf{c}_i, \mathbf{x})\, C_{\mathcal{P}_{d(\mathbf{x})}}}{\sum_{\bar{\mathbf{x}} \in I_s \cap I_t} p(\mathbf{c}_i, \bar{\mathbf{x}})} \tag{36}$$

where linear combinations of color maps are intended in the space induced by the transformation T of ME (see Eq. (1)), i.e.

$$a C_{\mathcal{P}_x} + b\, C_{\mathcal{P}_y} = U \circ (a \mathrm{M}_x + b \mathrm{M}_y) \circ T \tag{37}$$

Finally, the color value of a pixel $\mathbf{x}$, even not in the intersection area of the input images is computed as

$$I_c(\mathbf{x}) = C_v(I_s(\mathbf{x})) \tag{38}$$

where

$$v = \underset{1 \le i \le q^3}{\operatorname{argmin}} \parallel \mathbf{c}_i - I_s(\mathbf{x}) \parallel \tag{39}$$

is the index of the quantized color closest to $I_s(\mathbf{x})$.

IV. Evaluation

A. Experimental Setup

Following the recent literature on the subject [3], [4], [31], [32], stitching can be regarded as an extension of image mosaicing to the case where homography constraints are relaxed and hold only locally, so that registered images are just coarsely aligned.

For our evaluation, we relied mainly on the protocol described in [6] (the most complete comparison of color correction methods for stitching to date, here significantly extended), which assumes no known color transform model, this being the most general and sensible way to address the problem. Indeed, most recent datasets avoid to refer to any particular image acquisition conditions, i.e. operating setups (e.g. single vs multiple cameras, fixed vs changing camera parameters), due to the impossibility for the common user to have a fully controlled environment and the right level of knowledge and expertise. Nevertheless, we verified experimentally that the results of the best color correction methods are virtually uncorrelated with the acquisition setup and therefore with the associated color alteration (see Sec. IV-B).

The protocol of [6] was suitably modified so as (a) to take advantage from the improved image quality metrics published later, and (b) to better handle coarsely registered images. We used image pairs from three different existing datasets, plus a novel dataset explicitly designed to investigate the behavior of the color correction methods under different acquisition conditions. The first dataset [6] is subdivided into two classes of 30 real and 40 synthetic scenes. The real scene images of this dataset have been obtained from various sources, including image frames from multi-view video applications, photos with or without flash lighting or under different capture modes, and aerial image clips taken in the same place at different times. In the case of synthetic scenes, a perfect image alignment exists and color alterations have been obtained by software editing. The second dataset [2] contains 15 already registered image pairs mainly taken with different camera types and setups, or undergoing palette recoloring. The third dataset [7] is made up of 38 color image pairs, belonging to 6 real and almost planar scenes, obtained by varying image exposure only. The reader may refer to [2], [6], [7] for further details about the transformations involved in the related datasets. The novel dataset contains 127 image pairs taken from real image stitching application scenarios. These image pairs are divided into four groups of almost equal size, containing respectively image pairs obtained with (1) distinct cameras (NVIDIA Shield Tablet, Huawei P8, Huawei P9 light, SONY SLT-A37), (2) same camera (NVIDIA Shield Tablet) with locked setup parameters or (3) varying ISO, exposure and white balance, and (4) fixed parameters but different environmental illumination conditions. For all datasets, image pairs were registered through planar homographies according to the method described in [33] if the registration was not already available. The overall set of 250 image pairs was manually split into two working datasets, referred to as D_1 and D_2. Specifically, D_1 contains 158 realistic stitching image pairs with relevant image misalignments or other color inconsistencies (e.g. a shadow is present in the source image image but not in the target image), while D_2 contains the remaining images, mostly exhibiting a very accurate geometric registration and artificial color alterations (see the additional material). Notice that D_1 incorporates all the 127 new image pairs from the four groups corresponding to different acquisition conditions, respectively denoted as $D_{1.1\text{-}4}$, while D_2 includes all the synthetic images from [6] and all the repainting images from [2].

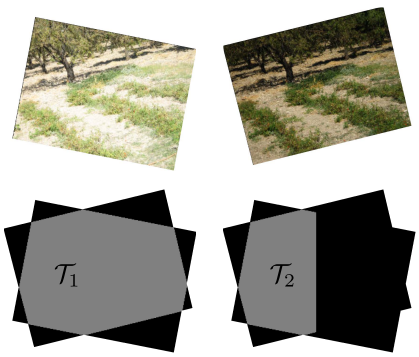


Fig. 7. Input images (top row) and the actual overlapping area (bottom row) used in tests $\mathcal{T}_1$ and $\mathcal{T}_2$. (best viewed in color).

On each image pair, we performed two distinct tests in which, given the input source and target images I_s and I_t, the corrected image I_c is output and evaluated accordingly. In the first test, named $\mathcal{T}_1$, we evaluated both the combinations (I_a, I_b) and (I_b, I_a) as input image pairs (I_s, I_t) that can be obtained by interchanging the role of the source and target images, according to a chosen image quality metric. In particular, the recent iCID measure [8] and FSIM$_C$ index [9] (i.e. the FSIM index extended to color images) were selected as the closest to human judgment. These quality metrics work better than simpler measures as PSNR [34] and SCIELAB [35] and, in contrast to the SSIM index [36], can deal with color images, which is an essential requirement in our case. Additionally, in the second test $\mathcal{T}_2$, only the image pairs with an overlap greater than 25% were selected, and I_t was set to 50% of its overlapping area when computing I_c (of course, the whole I_t is used for the comparison against I_c), thus gaining a further insight into the sensitivity of the algorithms with respect to input data size and outliers (see Fig. 7).

We ran all the possible ME/PA combinations, listed in Table I, for a total of $\mathcal{M} = 14 \times 7 = 98$ different color correction algorithms. Matlab code, datasets and low-resolution output images are freely available.[1] Implementations of P and TS were derived respectively from [22] and [6]. Parameters for the Edison mean shift implementation [28] used by CIM and MSS were tuned on a dataset sample to get the best results.

[1] https://drive.google.com/open?id=0B_3Nh0OK9BclQkt4empQVU5yVE0

Since, whatever the quality metric used, absolute errors may vary dramatically from an image pair (I_1, I_2) to another, in order to better appreciate the relative differences between the different methods $\mathcal{A}_i$ under examination, $i = 1, \ldots, \mathcal{M}$, we define and use the *soft rank*

$$r(\mathcal{A}_i, I_1, I_2) = \frac{|\varepsilon(\mathcal{A}_i, I_1, I_2) - b(I_1, I_2) + \epsilon|}{\sum_{j=1}^{\mathcal{M}} |\varepsilon(\mathcal{A}_j, I_1, I_2) - b(I_1, I_2) + \epsilon|} \quad (40)$$

where $\varepsilon(\mathcal{A}_i, I_1, I_2)$ is the error, in terms of either iCID or FSIM_C in the overlapping area $I_1 \cap I_2$, obtained when the color correction method $\mathcal{A}_i$ is used with the image pair (I_1, I_2), ϵ is a small constant value avoiding a zero denominator, and

$$b(I_1, I_2) = \min_{j=1,\ldots,\mathcal{M}} \varepsilon(\mathcal{A}_j, I_1, I_2) \quad (41)$$

is the best value among all methods for the image pair (I_1, I_2) in the case of the iCID metric. When FSIM_C is employed, the min function in Eq. (41) must be replaced by the max function, since FSIM_C is a similarity measure. The soft rank $r(\mathcal{A}_i, I_1, I_2)$ ranges in $[0, 1]$, since

$$\sum_{i=1}^{\mathcal{M}} r(\mathcal{A}_i, I_1, I_2) = 1 \quad (42)$$

and achieves lower values for better methods. Notice that the soft rank of Eq. (40) is proportional (by the constant $\mathcal{M}$) to the error $\Delta_i = |\varepsilon(\mathcal{A}_i, I_1, I_2) - b(I_1, I_2) + \epsilon|$ normalized with respect to its average μ_Δ

$$r(\mathcal{A}_i, I_1, I_2) \propto \frac{\Delta_i}{\mu_\Delta} \quad (43)$$

This is more robust and stable than normalizing by the maximum error (e.g. removing the worst method does not change significantly the rank). Notice also from Eq. (42) that the average soft rank value is $\frac{1}{\mathcal{M}} = \frac{1}{98} \approx 1\%$.

A quantitative and exhaustive evaluation of all the possible ME/PA combinations requires a careful selection of the evaluation protocol to avoid results in contrast with the common human perception. In order to evaluate a given method $\mathcal{A}_i$, we compare, as it is usual in the literature [6], the corrected and target images, i.e.

$$r_t = r(\mathcal{A}_i, I_c, I_t) \quad (44)$$

Nevertheless, since in the real case of mosaicing and more general stitching applications we are dealing with coarsely registered input images, such straightforward comparison, requiring an almost perfect registration, may lead to erroneous evaluations whatever the image quality measure used [37]. For instance, if an object moves between the two input images I_s and I_t, a color correction method that simply copies the I_t pixel data into I_s to get the corrected image I_c would achieve the best score with r_t, notwithstanding the fact that the object in the corrected image is in the wrong location. This is due to the fact that Eq. (44) does not take into account altogether the structural content of the source image I_s. Therefore, to avoid erroneous comparative results especially in the case of images from dataset D_1, composed of real scene data, the methods were tested not only with r_t, but also with the average between the results of the comparison of I_c with both the input images I_s and I_t, i.e.

$$r_g = \frac{r_t + r_s}{2} \quad (45)$$

where

$$r_s = r(\mathcal{A}_i, I_c, I_s) \quad (46)$$

B. Results

Tables II-III show the soft ranks r_g and r_t (expressed as percentages, the average error being about 1%, see Sec. IV-A) for test $\mathcal{T}_1$ with both FSIM_C and iCID, on datasets D_1 and D_2, respectively. Each table entry corresponds to a different ME/PA pair under test. The five best results obtained for each soft rank are shown in bold. Subscripts in Table II(III) refer to the associated result examples of Fig. 8(9).

According to Table II, GPS/LCP and its faster counterpart FGPS/LCP (both assembled with new CUs proposed in this work), followed by HM/LCP, are the only methods that rank among the best top five in r_g with both iCID and FSIM_C. Figure 8 shows color correction results obtained by the four best ME/PA pairs of Table II on an example image pair of dataset D_1. Visual inspection of Fig. 8(c-1) confirms that these three methods provide the least number of artifacts and the most natural color of all. Therefore, numerical results are in good accordance with human judgment. However, the top five ranked ME/PA pairs with r_t/FSIM_C are completely different from those obtained with r_t/iCID. There is no method that performs clearly better than the others on D_1 images with both error metrics. This corroborates our observation of Sec. IV-A that, in the presence of image misalignments, the soft rank r_t may not be appropriate for comparing color correction results, nor be in agreement with perceptual evidence. Nevertheless, notice that both GPS/LCP and FGPS/LCP are also top-ranked for r_t/FSIM_C.

On the other hand, results with both soft ranks r_t and r_g on dataset D_2, containing image pairs with accurate geometric registration and artificial color alterations, are more similar and consistent among them (see Table III). In particular, the AM/BLG pair (BLG being one of the new PAs proposed in this paper) achieves the top ranked results with all four combinations of soft ranks and quality metrics. Additionally, GPS/BLG and PM/TS are among the top ranked pairs with r_g and both quality metrics, the latter being also top-ranked for r_t/iCID. Finally, PM/MSS, TG/MSS and G/MSS are among the five top ranked pairs with r_t and both quality metrics (note that PM/MSS is composed of CUs never considered together before). From the visual example of Fig. 9, it is worth noting that, unlike the D_1 dataset case, there are no strongly noticeable perceptual differences among the best performing methods for the D_2 dataset, so that any of the above methods would give visually similar results.

The contrasting conclusions drawn out from D_1 and D_2 are compatible with the different input data employed in the datasets. In particular, images from D_1 are from realistic scenarios, contain more wrong color correspondences and have less content overlap to reconstruct the whole color mapping function. For this kind of images, working (as done e.g. in

TABLE II

SOFT RANK ON DATASET D_1, TEST $\mathcal{T}_1$, BEST FIVE ME/PA PAIRS ARE IN BOLD, DARKER VALUES ARE BETTER (BEST VIEWED IN COLOR)

	r_g/FSIM$_C$ (%)							r_g/iCID (%)							r_t/FSIM$_C$ (%)							r_t/iCID (%)						
	GL	LCP	BLG	CIM	MSS	TS	P	GL	LCP	BLG	CIM	MSS	TS	P	GL	LCP	BLG	CIM	MSS	TS	P	GL	LCP	BLG	CIM	MSS	TS	P
R	0.457	**0.429**	0.538	0.586	0.672	0.475	**0.428**$_g$	0.745	0.620	0.583	0.976	0.835	0.693	0.660	0.546	0.384	0.300	0.671	0.653	0.522	0.500	0.875	0.514	0.359	1.174	0.868	0.766	0.748
CS	0.501	0.573	0.513	0.706	2.761	1.392	0.970	0.769	0.699	0.566	1.208	2.683	1.739	1.424	0.755	0.601	**0.272**	0.822	3.104	1.687	1.217	0.994	0.647	0.315	1.418	3.158	2.095	1.736
GC	0.719	0.697	0.506	0.648	0.722	1.117	1.182	0.830	0.714	0.580	0.901	0.783	1.239	1.335	0.846	0.709	0.317	0.710	0.758	1.281	1.370	1.181	0.754	0.427	1.075	0.920	1.618	1.782
3M	0.728	0.715	0.790	1.332	0.781	0.753	0.760	0.735	0.652	0.728	1.494	0.743	0.747	0.750	0.836	0.557	0.489	1.473	0.702	0.793	0.831	1.046	0.494	0.466	1.842	0.737	0.939	0.997
PM	0.612	1.333	0.789	1.819	0.777	0.693	0.694	0.647	0.920	0.691	1.961	0.687	0.683	0.683	0.485	1.072	0.441	1.846	0.495	0.496	0.509	0.482	0.519	0.326	2.084	0.355	0.455	0.466
AM	0.610	0.987	0.744	1.377	0.729	0.615	0.608	0.692	0.721	0.674	1.515	0.692	0.690	0.687	0.730	0.793	0.430	1.447	0.585	0.648	0.689	0.726	0.351	0.327	1.602	0.470	0.659	0.700
PR	2.101	1.437	0.693	2.529	2.300	4.015	4.064	2.056	1.529	0.658	2.609	2.236	3.734	3.899	2.520	1.410	0.405	3.000	2.833	4.720	4.767	2.485	1.571	0.378	3.182	2.772	4.379	4.580
HM	0.468	**0.423**$_h$	0.618	0.598	0.659	0.566	0.554	0.663	**0.538**$_h$	0.620	0.835	0.801	0.690	0.669	0.425	0.295	0.384	0.724	0.526	0.487	0.471	0.657	0.453	0.447	1.168	0.727	0.633	0.605
TG	0.575	0.734	0.748	1.362	0.739	3.950	4.871	0.656	0.571	0.657	1.427	0.686	2.513	2.978	0.560	0.465	0.453	1.354	0.657	4.394	5.520	0.602	**0.175**$_k$	0.322	1.467	0.539	2.618	3.154
G	0.677	0.879	0.769	1.586	0.899	4.433	5.377	0.698	0.629	0.666	1.585	0.748	2.763	3.248	0.681	0.621	0.472	1.617	0.830	4.948	6.116	0.635	**0.219**$_l$	0.329	1.664	0.587	2.883	3.452
MS	0.618	0.557	0.630	0.701	0.686	0.686	0.695	0.711	**0.495**$_j$	0.594	0.877	0.678	0.708	0.712	0.567	0.336	0.361	0.695	0.525	0.572	0.580	0.670	**0.163**$_j$	**0.307**	0.944	0.487	0.535	0.532
3MS	0.665	0.592	0.726	0.952	0.743	0.703	0.710	0.742	**0.514**$_i$	0.663	1.172	0.691	0.704	0.715	0.559	0.304	0.420	0.935	0.518	0.553	0.565	0.609	**0.103**$_i$	0.335	1.249	0.414	0.492	0.506
GPS	0.466	**0.303**$_f$	0.483	0.472	0.559	0.510	0.497	0.767	**0.487**$_f$	0.546	0.796	0.788	0.776	0.774	0.488	**0.222**$_f$	**0.254**$_d$	0.562	0.512	0.524	0.519	0.840	0.358	0.324	1.014	0.776	0.844	0.856
FGPS	0.528	**0.315**$_e$	0.506	0.515	0.583	0.578	0.598	0.865	**0.498**$_e$	0.551	0.874	0.849	0.877	0.889	0.554	**0.232**$_e$	**0.268**$_c$	0.625	0.534	0.605	0.637	0.948	0.374	0.320	1.142	0.835	0.947	0.979

TABLE III

SOFT RANK ON DATASET D_2, TEST $\mathcal{T}_1$, BEST FIVE ME/PA PAIRS ARE IN BOLD, DARKER VALUES ARE BETTER (BEST VIEWED IN COLOR)

	r_g/FSIM$_C$ (%)							r_g/iCID (%)							r_t/FSIM$_C$ (%)							r_t/iCID (%)						
	GL	LCP	BLG	CIM	MSS	TS	P	GL	LCP	BLG	CIM	MSS	TS	P	GL	LCP	BLG	CIM	MSS	TS	P	GL	LCP	BLG	CIM	MSS	TS	P
R	0.989	0.629	**0.397**$_e$	0.706	0.872	0.736	1.011	1.108	0.781	0.482	0.967	0.906	0.799	0.961	0.967	0.507	0.151	0.635	0.783	0.652	0.932	1.178	0.619	0.187	0.906	0.768	0.674	0.944
CS	0.638	1.248	0.522	0.617	0.829	7.646	1.665	0.784	1.090	0.578	1.014	0.908	6.079	2.616	0.838	1.421	0.310	0.654	0.857	9.294	2.096	0.822	1.112	0.288	0.960	0.773	7.979	3.329
GC	1.307	1.310	0.560	1.040	1.486	1.176	2.265	1.204	1.104	0.593	1.118	1.161	1.043	2.455	1.518	1.445	0.398	1.136	1.675	1.328	2.677	1.776	1.341	0.425	1.386	1.383	1.266	3.408
3M	1.270	0.933	0.630	1.027	1.144	1.023	1.324	1.004	0.761	0.606	0.982	0.871	0.827	1.064	1.520	0.963	0.471	1.135	1.199	1.098	1.536	1.536	0.857	0.404	1.270	0.955	0.945	1.500
PM	**0.398**$_f$	0.832	0.423	0.519	0.413	**0.399**$_j$	0.434	0.494	0.735	0.502	0.688	**0.465**$_h$	**0.472**$_j$	0.502	0.237	0.702	**0.135**	0.349	**0.121**$_h$	0.160	0.229	0.222	0.531	0.160	0.510	**0.064**$_h$	**0.147**$_j$	0.201
AM	0.497	0.645	**0.379**$_c$	0.435	0.453	0.443	0.500	0.560	0.580	**0.472**$_c$	0.662	0.498	0.520	0.553	0.628	0.647	**0.126**$_c$	0.387	0.356	0.341	0.564	0.526	0.404	**0.144**$_c$	0.484	0.235	0.296	0.475
PR	2.062	1.642	0.425	1.996	3.546	5.387	9.791	2.150	1.686	0.502	2.447	2.674	4.317	8.922	2.342	2.030	0.185	2.395	4.350	7.168	12.26	2.619	2.066	0.218	3.013	3.249	5.935	11.95
HM	0.464	0.547	0.636	0.595	0.494	0.472	0.634	0.591	0.561	0.640	0.652	0.578	0.584	0.641	0.237	0.377	0.411	0.521	0.193	0.236	0.405	0.368	0.382	0.445	0.720	0.249	0.334	0.387
TG	0.408	0.475	0.471	0.737	0.448	0.437	2.219	0.505	0.496	0.515	0.854	0.485	0.513	1.556	0.170	0.259	0.201	0.602	**0.119**$_i$	0.193	2.242	0.266	0.210	0.205	0.738	**0.119**$_i$	0.264	1.525
G	0.412	0.490	0.478	0.768	0.460	0.446	2.382	0.504	0.501	0.520	0.887	0.485	0.514	1.639	0.179	0.280	0.212	0.641	**0.134**$_g$	0.207	2.420	0.266	0.220	0.215	0.786	**0.119**$_g$	0.267	1.620
MS	0.709	0.572	0.513	0.596	0.650	0.637	0.607	0.747	0.595	0.545	0.695	0.657	0.673	0.618	0.633	0.447	0.283	0.508	0.457	0.511	0.409	0.656	0.411	0.271	0.648	0.404	0.513	0.357
3MS	0.690	0.529	0.478	0.581	0.612	0.592	0.573	0.725	0.540	0.519	0.681	0.621	0.602	0.596	0.568	0.363	0.225	0.442	0.347	0.420	0.350	0.514	0.269	0.198	0.526	0.254	0.335	0.310
GPS	0.824	0.513	**0.399**$_d$	0.614	0.742	0.623	0.929	1.088	0.594	**0.478**$_d$	0.843	0.846	0.748	1.037	0.853	0.444	0.153	0.598	0.650	0.548	0.937	1.133	0.451	0.185	0.927	0.697	0.630	1.020
FGPS	1.010	0.632	0.403	0.888	1.039	0.817	1.137	1.380	0.685	**0.479**	1.099	1.181	0.974	1.267	1.045	0.593	0.153	0.934	1.026	0.794	1.162	1.524	0.608	0.185	1.384	1.177	0.951	1.308

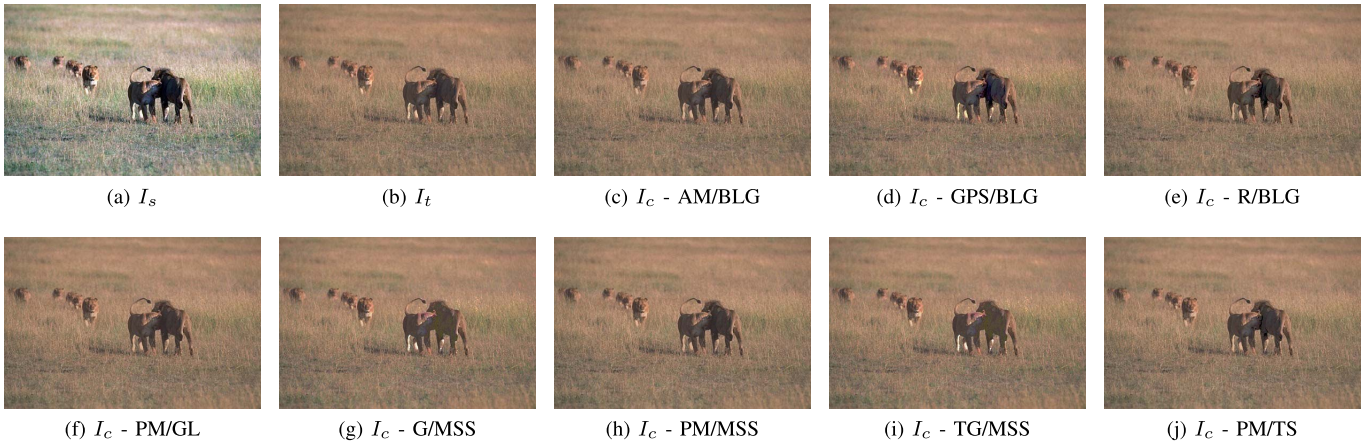
(a) I_s (b) I_t (c) I_c - FGPS/BLG (d) I_c - GPS/BLG

(e) I_c - FGPS/LCP (f) I_c - GPS/LCP (g) I_c - R/P (h) I_c - HM/LCP

(i) I_c - 3MS/LCP (j) I_c - MS/LCP (k) I_c - TG/LCP (l) I_c - G/LCP

Fig. 8. Results on an example image pair for the first four top ranked ME/PA pairs (c-l) on dataset D_1, test $\mathcal{T}_1$, according to r_t and r_g for FSIM$_C$ and iCID. I_s and I_c are superimposed on I_t for a better understanding (best viewed in color and zoomed in).

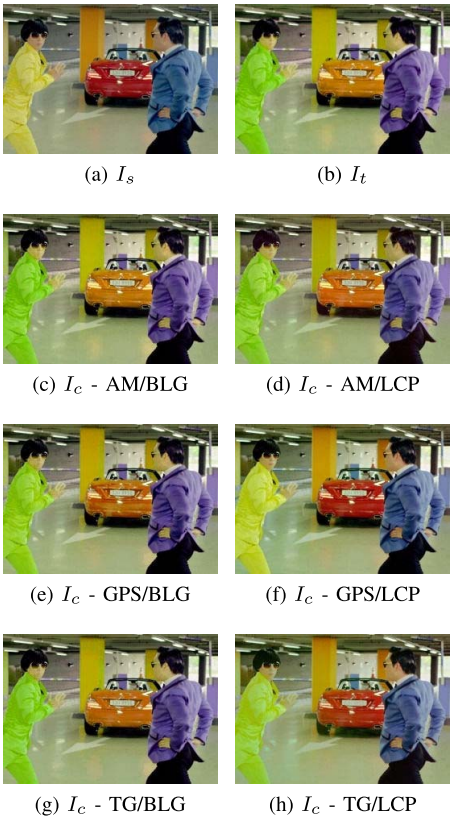
the GPS/LCP method) on robust characteristics such as the average color values and image gradient, yields the most satisfactory results. This said, it is not true, as one may hypothesize, that approaches with fewer parameters are most robust and therefore should do better than others. Actually, despite their apparently simple model formulation, GPS and FGPS (the best MEs for D_1) have more implicit hidden parameters than others. As a matter of fact, error minimization



(a) I_s (b) I_t (c) I_c - AM/BLG (d) I_c - GPS/BLG (e) I_c - R/BLG

(f) I_c - PM/GL (g) I_c - G/MSS (h) I_c - PM/MSS (i) I_c - TG/MSS (j) I_c - PM/TS

Fig. 9. Results on an example image pair for the first four top ranked ME/PA pairs on dataset D_2, test $\mathcal{T}_2$, according to r_t and r_g for FSIM_C and iCID. I_s and I_c are superimposed on I_t for a better understanding (best viewed in color and zoomed in).



(a) I_s (b) I_t

(c) I_c - AM/BLG (d) I_c - AM/LCP

(e) I_c - GPS/BLG (f) I_c - GPS/LCP

(g) I_c - TG/BLG (h) I_c - TG/LCP

Fig. 10. Example of different outputs for intra-channel color transformations (best viewed in color and zoomed in).

with these methods is so complex that we gave up finding an analytical, yet approximate, solution, and decided to perform a brute force space search. On the other hand, when color correspondences are more complete and correct, and color alteration less realistic, such as with the D_2 input images, CUs that "overfit" the data (e.g. the AM and PM MEs and the BLG and MSS PAs) achieve the best accuracy. Notice also that segmentation-based PAs such as MSS and TS are unlikely to give good results in D_1, especially in the non-overlapping area, yet are expected to do quite well in D_2. Similar ranking considerations hold in the case of test $\mathcal{T}_2$ (more detailed results are reported as additional material), but clearly the quality of the final corrected image is lower, as less data correspondences are available as input for the algorithms.

An additional observation must be made about MEs working channel-wise (i.e. GC, HM, TG, G, GPS, FGPS and MS) in the presence of some limited, application-specific color transformations, such as those employed for palette recoloring (see Fig. 10). These MEs, when combined with PAs based on linear combinations (e.g. GL and LCP), are mostly unable to handle intra-channel color transformations such as those from red to green. This does not hold for non-linear PAs (e.g. BLG and MSS).

Concerning image pairs from realistic image stitching scenarios, we further evaluated the color correction methods according to datasets $D_{1.1\text{-}4}$ given by splitting the novel 127 image pairs created for this work, representing four main sources of color alterations (see Sec. IV-A). Tables IV-V report the top five soft ranks ME/PA pairs for each evaluated test and dataset (detailed results can be found in the additional material). According to the previous observations, we focus our attention only on soft rank r_g due to the nature of considered images. Results show that for $D_{1.1}$, containing input images from distinct cameras, GPS/LCP and FGPS/LCP are still the best methods. In the case $D_{1.2}$ of stitching images from the same camera with a locked setup, the same results as for the whole dataset D_1 hold for test $\mathcal{T}_1$, while GC/GL is the best method for test $\mathcal{T}_2$. This is not surprising, as in test $\mathcal{T}_2$ the number of color correspondence outliers is really high due to the low image overlap, and in the case of a single camera and a locked setup the true color transformation is close to the identity map. Therefore, GC/GL, which we found generally to alter the original colors of I_s less than other methods, performs best. On the other hand, for $D_{1.3}$, where images come from the same camera but with varying ISO, exposure and white balance, GPS/BLG and FGPS/BLG do better than GPS/LCP and FGPS/LCP, which nevertheless are still among the best top-ranked methods. This is probably due to some recoloring effects caused by the white balance adjustment, which alters intra-channel color relations as discussed above.

TABLE IV

TOP FIVE ME/PA PAIRS FOR SOFT RANK r_g ON EACH TEST AND DATASET, BOLD ENTRIES INDICATE COMBINATIONS OF NEW CUS ONLY

		r_g/FSIM$_C$										r_g/iCID									
		1^{st}	%	2^{nd}	%	3^{rd}	%	4^{th}	%	5^{th}	%	1^{st}	%	2^{nd}	%	3^{rd}	%	4^{th}	%	5^{th}	%
$\mathcal{T}_1$	D_1	**GPS/LCP**	0.303	**FGPS/LCP**	0.315	HM/LCP	0.423	R/P	0.428	R/LCP	0.429	**GPS/LCP**	0.487	**FGPS/LCP**	0.498	MS/LCP	0.495	3MS/LCP	0.514	HM/LCP	0.538
	D_2	AM/BLG	0.380	PM/GL	0.398	R/BLG	0.398	**GPS/BLG**	0.399	PM/TS	0.400	PM/MSS	0.463	PM/TS	0.470	AM/BLG	0.471	**GPS/BLG**	0.476	**FGPS/BLG**	0.477
	$D_{1.1}$	**GPS/LCP**	0.255	**FGPS/LCP**	0.271	CS/GL	0.315	HM/GL	0.334	HM/LCP	0.335	**GPS/LCP**	0.533	**GPS/BLG**	0.540	**FGPS/LCP**	0.540	**FGPS/BLG**	0.552	MS/LCP	0.569
	$D_{1.2}$	HM/LCP	0.122	**GPS/LCP**	0.131	**FGPS/LCP**	0.155	R/TS	0.158	R/P	0.158	HM/LCP	0.216	**GPS/LCP**	0.222	GC/LCP	0.234	**FGPS/LCP**	0.254	R/LCP	0.255
	$D_{1.3}$	**FGPS/LCP**	0.411	**GPS/LCP**	0.413	**FGPS/BLG**	0.446	**GPS/BLG**	0.451	CS/BLG	0.474	TG/LCP	0.558	**FGPS/BLG**	0.562	**GPS/BLG**	0.567	3MS/LCP	0.578	CS/BLG	0.588
	$D_{1.4}$	**GPS/LCP**	0.342	**FGPS/LCP**	0.342	3MS/LCP	0.449	MS/LCP	0.451	R/LCP	0.458	MS/LCP	0.449	3MS/LCP	0.456	**GPS/LCP**	0.497	**FGPS/LCP**	0.502	TG/LCP	0.532
$\mathcal{T}_2$	D_1	**GPS/LCP**	0.319	**FGPS/LCP**	0.333	R/P	0.385	R/GL	0.412	GPS/GL	0.417	**GPS/LCP**	0.498	**FGPS/LCP**	0.516	**GPS/BLG**	0.523	MS/LCP	0.532	GC/BLG	0.538
	D_2	AM/BLG	0.385	PM/TS	0.390	PM/GL	0.394	R/BLG	0.407	**GPS/BLG**	0.408	PM/TS	0.457	AM/BLG	0.469	**GPS/BLG**	0.473	**FGPS/BLG**	0.476	PM/BLG	0.476
	$D_{1.1}$	**GPS/LCP**	0.256	**FGPS/LCP**	0.274	HM/GL	0.322	GPS/TS	0.323	GPS/P	0.323	**GPS/BLG**	0.515	**GPS/LCP**	0.519	**FGPS/LCP**	0.536	GC/BLG	0.537	**FGPS/BLG**	0.543
	$D_{1.2}$	GC/GL	0.077	R/TS	0.093	R/P	0.093	GC/CIM	0.099	R/GC	0.106	3M/GL	0.125	GC/GL	0.127	3M/TS	0.139	3M/P	0.139	GC/MSS	0.154
	$D_{1.3}$	**GPS/LCP**	0.391	**FGPS/LCP**	0.395	**GPS/BLG**	0.456	HM/GL	0.461	**FGPS/BLG**	0.472	**GPS/BLG**	0.558	TG/LCP	0.565	**FGPS/BLG**	0.569	**GPS/LCP**	0.581	**FGPS/LCP**	0.591
	$D_{1.4}$	**GPS/LCP**	0.411	**FGPS/LCP**	0.412	GPS/GL	0.458	R/GL	0.471	CS/GL	0.491	MS/LCP	0.564	**GPS/LCP**	0.572	3MS/LCP	0.573	**GPS/BLG**	0.578	**FGPS/LCP**	0.584

TABLE V

TOP FIVE ME/PA PAIRS FOR SOFT RANK r_t ON EACH TEST AND DATASET, BOLD ENTRIES INDICATE COMBINATIONS OF NEW CUS ONLY

		r_t/FSIM$_C$										r_t/iCID									
		1^{st}	%	2^{nd}	%	3^{rd}	%	4^{th}	%	5^{th}	%	1^{st}	%	2^{nd}	%	3^{rd}	%	4^{th}	%	5^{th}	%
$\mathcal{T}_1$	D_1	**GPS/LCP**	0.222	**FGPS/LCP**	0.232	**GPS/BLG**	0.254	**FGPS/BLG**	0.268	CS/BLG	0.272	3MS/LCP	0.103	MS/LCP	0.163	TG/LCP	0.175	G/LCP	0.219	MS/BLG	0.307
	D_2	TG/MSS	0.119	PM/MSS	0.121	AM/BLG	0.126	G/MSS	0.134	PM/BLG	0.135	PM/MSS	0.064	TG/MSS	0.119	G/MSS	0.119	AM/BLG	0.144	PM/TS	0.147
	$D_{1.1}$	**GPS/LCP**	0.208	**FGPS/LCP**	0.214	HM/LCP	0.238	**GPS/BLG**	0.264	CS/BLG	0.282	3MS/LCP	0.118	TG/LCP	0.138	MS/LCP	0.153	G/LCP	0.172	AM/LCP	0.329
	$D_{1.2}$	HM/LCP	0.056	**GPS/LCP**	0.126	**FGPS/LCP**	0.139	R/LCP	0.161	MS/LCP	0.198	3MS/LCP	0.049	MS/LCP	0.050	AM/LCP	0.161	3M/LCP	0.164	HM/LCP	0.179
	$D_{1.3}$	**GPS/BLG**	0.144	**FGPS/BLG**	0.149	CS/BLG	0.180	3MS/LCP	0.189	**GPS/LCP**	0.194	3MS/LCP	0.117	TG/LCP	0.124	G/LCP	0.178	CS/BLG	0.198	**FGPS/BLG**	0.200
	$D_{1.4}$	3MS/LCP	0.118	MS/LCP	0.187	**GPS/LCP**	0.260	**FGPS/LCP**	0.264	TG/LCP	0.283	3MS/LCP	0.043	MS/LCP	0.081	TG/LCP	0.111	G/LCP	0.135	**GPS/LCP**	0.289
$\mathcal{T}_2$	D_1	**GPS/LCP**	0.210	**FGPS/LCP**	0.223	**GPS/BLG**	0.273	HM/LCP	0.287	**FGPS/BLG**	0.305	3MS/LCP	0.225	MS/LCP	0.252	**GPS/BLG**	0.274	TG/LCP	0.283	MS/BLG	0.292
	D_2	PM/BLG	0.127	PM/TS	0.136	AM/BLG	0.139	R/BLG	0.161	**GPS/BLG**	0.167	PM/TS	0.104	PM/BLG	0.113	AM/BLG	0.129	**GPS/BLG**	0.168	**FGPS/BLG**	0.173
	$D_{1.1}$	**GPS/LCP**	0.191	**FGPS/LCP**	0.201	HM/LCP	0.227	HM/GL	0.262	GC/BLG	0.266	3MS/LCP	0.222	MS/LCP	0.233	TG/LCP	0.282	MS/BLG	0.305	G/LCP	0.315
	$D_{1.2}$	HM/LCP	0.070	**GPS/LCP**	0.123	GC/GL	0.133	GC/CIM	0.136	HM/GL	0.136	MS/LCP	0.127	HM/LCP	0.135	GC/BLG	0.146	3M/GL	0.156	3MS/GL	0.157
	$D_{1.3}$	**GPS/LCP**	0.145	HM/GL	0.152	**FGPS/LCP**	0.158	**GPS/BLG**	0.159	3MS/LCP	0.183	TG/LCP	0.176	**GPS/BLG**	0.182	3MS/LCP	0.185	**FGPS/BLG**	0.207	G/LCP	0.236
	$D_{1.4}$	**GPS/LCP**	0.290	3MS/LCP	0.295	**FGPS/LCP**	0.305	MS/LCP	0.306	TG/LCP	0.359	3MS/LCP	0.244	MS/LCP	0.255	TG/LCP	0.280	**GPS/BLG**	0.289	**FGPS/BLG**	0.303

Finally, for $D_{1.4}$, having images from a fixed camera and varying environmental illumination, GPS/LCP and FGPS/LCP are still the best methods, followed by 3MS/LCP and MS/LCP.

To summarize: Referring again to Tables IV-V, GPS and FGPS are the best MEs in the case of real stitching input images, either paired with LCP (bold black entries) or, in the case of intra-channel transformation, with BLG (bold red entries). Moreover, LCP and BLG are generally the best PAs for other MEs. For what concerns real stitching applications, we can conclude that there is no evidence that a given source of color distortion noticeably affects the output of the best color correction methods. Conversely, in the case of artificial color alterations and finer image registration, BLG and MSS are the best PAs, combined with PM followed by AM, TG, G and GPS.

C. More about soft ranks and quality metrics

Table VI reports the Spearman rank-order correlation coefficient (SROCC) [38] between soft ranks r_t, r_s and r_g, obtained by comparing the corrected image I_c respectively against the target I_t, the source I_s or both input images. Both FSIM$_C$ and iCID on the D_1 and D_2 datasets (see Sec. IV-A) are considered. SROCC is often used for the analysis of image quality metrics [8], [9], [36], [39].

In the case of iCID vs iCID (bottom right block matrices), r_g is strongly correlated with r_t, less with r_s, on both D_1 and D_2. Similarly, r_t and r_s are strongly uncorrelated. This implies that r_g is unlikely to be minimum when I_c is half way between I_t and I_s for iCID. Note also that ranks on D_2 are more correlated to each other than ranks on D_1, as accurate image alignment implies anyway a higher structural similarity inside image pairs.

In the case of FSIM$_C$ vs FSIM$_C$ (top left block matrices), r_t and r_s have a good correlation on D_2 but not on D_1, while

TABLE VI

SPEARMAN RANK-ORDER CORRELATION COEFFICIENT (%) FOR DIFFERENT SOFT RANKS AND IMAGE QUALITY METRICS

		FSIM$_C$			iCID		
		r_g	r_t	r_s	r_g	r_t	r_s
FSIM$_C$	r_g	100	81	90	65	43	82
	r_t	81	100	53	84	78	50
	r_s	90	53	100	42	14	90
iCID	r_g	65	84	42	100	88	51
	r_t	43	78	14	88	100	21
	r_s	82	50	90	51	21	100

(a) D_1

		FSIM$_C$			iCID		
		r_g	r_t	r_s	r_g	r_t	r_s
FSIM$_C$	r_g	100	97	89	96	94	57
	r_t	97	100	77	95	97	47
	r_s	89	77	100	80	74	65
iCID	r_g	96	95	80	100	98	61
	r_t	94	97	74	98	100	48
	r_s	57	47	65	61	48	100

(b) D_2

r_g has a strong correlation with both r_t and r_s. This is likely due to the fact that, differently to iCID, FSIM$_C$ relies more on image structural content than on color intensities, as geometric transformations become stronger.

Further evidence of this behavior can be observed by inspecting SROCC for FSIM$_C$ vs iCID (top right and bottom left block matrices). On D_1 (see Table VI(a)), r_g/iCID is not strongly correlated with r_g/FSIM$_C$, and is more correlated with r_t/FSIM$_C$ than with r_s/FSIM$_C$. The opposite happens for r_g/FSIM$_C$, which is less correlated with r_t/iCID than with r_s/iCID. This does not happen on D_2 (see Table VI(b)), where in any case r_g is more correlated with r_t than with r_s, thanks to the accurate geometric registration of the images.

To summarize: For accurately registered images, r_g gives similar results whatever the error quality metric used (FSIM$_C$ or iCID), and is more in accordance with r_t than with r_s, as quality metrics are designed to privilege more color content than structural context. On the other hand, for coarsely registered images, r_g/FSIM$_C$ tends to emphasize more than r_g/iCID the structural context of I_s (taken into account by r_s) over the color content of I_t (taken into account by r_t). As a result, special care should be taken when using Table II to select a good color correction method for

TABLE VII
AVERAGE RUNNING TIME (S), DARKER VALUES ARE BETTER (BEST VIEWED IN COLOR)

	GL	LCP	BLG	CIM	MSS	TS	P
R	0.38	2.41	18.44	9.56	12.84	334.08	9.04
CS	1.02	2.11	5.51	7.94	11.63	336.18	4.04
GC	1.13	1.87	5.13	7.58	11.27	335.18	3.82
3M	1.05	2.11	5.40	7.67	11.65	337.18	4.37
PM	1.14	2.40	5.97	7.88	11.64	336.70	5.63
AM	1.21	2.09	5.51	7.83	11.63	337.03	4.66
PR	1.26	2.92	19.40	9.48	13.17	334.54	11.60
HM	1.47	11.77	9.20	8.10	11.87	334.41	19.83
TG	9.16	107.99	101.92	59.28	26.42	343.08	297.52
G	1.46	15.08	12.28	9.89	12.13	320.46	24.80
MS	1.62	16.43	13.83	10.29	13.36	336.06	29.95
3MS	1.73	16.66	14.81	10.51	13.48	322.88	31.04
GPS	42.76	50.62	47.75	685.12	501.15	446.37	2318.54
FGPS	13.55	31.43	28.70	192.05	134.81	368.25	763.54

coarsely registered images. To be sure that the chosen method works reasonably well, it should have a high rank with both r_g/FSIM$_C$ and r_g/iCID, and not with only one of them. Following this observation, we can conclude that, as already emerged from the analysis of Table II, both GPS/LCP and FGPS/LCP are ideal candidates for color correction with coarsely registered images, as they rank among the first with r_g whatever the quality metric, and are also visually in accordance with human judgment.

D. Running Time

Table VII shows the average running time for input images of about 800×600 pixels. Times refer to our 6 core multi-threaded Matlab implementation on a Intel i7 at 4GHz.

Concerning the running time for PAs, GL is obviously the fastest with any ME, followed by LCP and BLG, both working block-wise on the images. CIM and MSS, which are based on mean shift segmentation, come next, followed by P, when combined with all but GPS, FGPS and TG. The worst running times are obtained with TS whichever ME, and by P when combined with either GPS, FGPS and TG.

An insight into the time performance of MEs is given by the GL column of Table VII, dealing with the simplest PA. Data show that almost all MEs run in less than 2 seconds, except for GPS, FGPS and TG, whose running times are still reasonable. In addition, both GPS and FGPS, when coupled with LCP and BLG, that provide the most promising visual results (see Sec. IV-B), are still conveniently usable with respect to GL. Notice also that FGPS provides a remarkable running time improvement over GPS, with a speedup of about $4\times$.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we introduced a new compositional framework for classifying color correction methods in terms of ME/PA pairs. This framework is completely general and comprehensive, and allows for a clearer analysis of color correction methods, providing a deeper insight into their properties.

We revisited and categorized 15 of the existing color correction methods according to this framework, identifying and combining pairwise their MEs and PAs to design new methods never considered before. We also designed four novel CUs (2 MEs and 2 PAs), that were found to be the building material for the best performing algorithms.

In order to evaluate all possible 98 color correction methods that arise from the proposed framework, image pairs from several distinct datasets were used. Differently from previous evaluations, our tests focused on performance evaluation in the presence of image misalignments, bearing in mind real world applications such as image mosaicing and stitching. To the best of our knowledge, this is the first evaluation on a so large dataset, explicitly takes into account coarse image registration issues, and is also the first to employ the latest state-of-the-art image quality metrics, known to be the closest to human judgment. According to our comparative evaluations, the GPS/LCP and FGPS/LCP methods, built upon our proposed CUs, achieved the top ranking and obtained the most robust results in the case of real stitching scenarios. In the presence of accurate image registration and artificial color alterations, AM/BLG, whose PA was also proposed in this paper, obtained the most accurate results, followed by PM/MSS, which is an original combination of existing CUs.

Our compositional framework is quite general. Therefore, we plan as future work to add further CUs from existing color correction methods which were not considered in this paper. We also plan to extend the datasets and the quality assessment metrics employed, for instance by applying on accurate registered images known geometric transformations to mimic a coarse image alignment, thus making the choice of the ground-truth and of the error metrics more reliable. Even more importantly, we think that an interesting direction for future research is the investigation of computational chains composed of more than one pair of ME and PA, which could prove effective at overcoming the weaknesses of a given single ME/PA pair.

ACKNOWLEDGMENTS

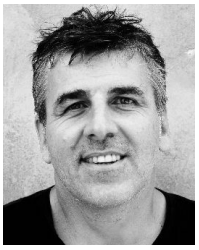
We would like to thank the authors of [2] and [6] for granting access to their datasets and Giosuè Lo Bosco, Domenico Tegolo and Cesare Valenti for granting access to the computational resources of the University of Palermo.

REFERENCES

[1] M. Brown and D. G. Lowe, "Automatic panoramic image stitching using invariant features," *Int. J. Comput. Vis.*, vol. 74, no. 1, pp. 59–73, Aug. 2007.
[2] Y. Hwang, J.-Y. Lee, I. S. Kweon, and S. J. Kim, "Color transfer using probabilistic moving least squares," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 3342–3349.
[3] J. Zaragoza, T.-J. Chin, Q.-H. Tran, M. S. Brown, and D. Suter, "As-projective-as-possible image stitching with moving DLT," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 7, pp. 1285–1298, Jul. 2014.
[4] H.-R. Su and S.-H. Lai, "Non-rigid registration of images with geometric and photometric deformation by using local affine Fourier-moment matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 2874–2882.
[5] H. S. Faridul *et al.*, "Colour mapping: A review of recent methods, extensions and applications," *Comput. Graph. Forum*, vol. 35, no. 1, pp. 59–88, 2016.
[6] W. Xu and J. Mulligan, "Performance evaluation of color correction approaches for automatic multi-view image and video stitching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2010, pp. 263–270.
[7] F. Bellavia and C. Colombo, "Color correction for image stitching by monotone cubic spline interpolation," in *Proc. Iberian Conf. Pattern Recognit. Image Anal. (IbPRIA)*, 2015, pp. 165–172.

[8] J. Preiss, F. Fernandes, and P. Urban, "Color-image quality assessment: From prediction to optimization," *IEEE Trans. Image Process.*, vol. 23, no. 3, pp. 1366–1378, Mar. 2014.
[9] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.
[10] E. Reinhard, M. Adhikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Comput. Graph. Appl.*, vol. 21, no. 5, pp. 34–41, Sep./Oct. 2001.
[11] X. Xiao and L. Ma, "Color transfer in correlated color space," in *Proc. ACM Int. Conf. Virtual Reality Continuum Appl.*, 2006, pp. 305–309.
[12] J. Yin and J. R. Cooperstock, "Color correction methods with applications to digital projection environments," *J. Winter School Comput. Graph.*, vol. 12, no. 3, pp. 499–506, 2004.
[13] M. Oliveira, A. Sappa, and V. Santos, "Unsupervised local color correction for coarsely registered images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 201–208.
[14] Y.-W. Taiand, J. Jia, and C.-K. Tang, "Local color transfer via probabilistic segmentation by expectation-maximization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2005, pp. 747–754.
[15] M. Uyttendaele, A. Eden, and R. Szeliski, "Eliminating ghosting and exposure artifacts in image mosaics," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2. Dec. 2001, pp. 509–516.
[16] A. Ilie and G. Welch, "Ensuring color consistency across multiple cameras," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2005, pp. 1268–1275.
[17] H. S. Faridul, J. Stauder, J. Kervec, and A. Tremeau, "Approximate cross channel color mapping from sparse color correspondences," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, Dec. 2013, pp. 860–867.
[18] M. Grogan and R. Dahyot. (May 2017). "Robust registration of Gaussian mixtures for colour transfer." [Online]. Available: https://arxiv.org/abs/1705.06091
[19] D. Liao, Y. Qian, and Z.-N. Li, "Semisupervised manifold learning for color transfer between multiview images," in *Proc. Int. Conf. Pattern Recognit. (ICPR)*, Dec. 2016, pp. 259–264.
[20] 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, Sep. 2008.
[21] M. Oliveira, A. D. Sappa, and V. Santos, "A probabilistic approach for color correction in image mosaicking applications," *IEEE Trans. Image Process.*, vol. 24, no. 2, pp. 508–523, Feb. 2015.
[22] F. Pitié, A. C. Kokaram, and R. Dahyot, "Automated colour grading using colour distribution transfer," *Comput. Vis. Image Understand.*, vol. 107, nos. 1–2, pp. 123–137, Jul. 2007. [Online]. Available: http://www.mee.tcd.ie/~sigmedia/Research/ColourGrading
[23] M. Zhang and N. Georganas, "Fast color correction using principal regions mapping in different color spaces," *Real-Time Imag.*, vol. 10, no. 1, pp. 23–30, 2004.
[24] J. Jia and C.-K. Tang, "Tensor voting for image correction by global and local intensity alignment," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 1, pp. 36–50, Jan. 2005.
[25] S. J. Kim and M. Pollefeys, "Robust radiometric calibration and vignetting correction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 4, pp. 562–576, Apr. 2008.
[26] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski, "Non-rigid dense correspondence with applications for image enhancement," *ACM Trans. Graph.*, vol. 30, no. 4, pp. 70:1–70:10, 2011.
[27] F. N. Fritsch and R. E. Carlson, "Monotone piecewise cubic interpolation," *SIAM J. Numer. Anal.*, vol. 17, no. 2, pp. 238–246, 1980.
[28] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 5, pp. 603–619, May 2002. [Online]. Available: http://coewww.rutgers.edu/riul/research/code/EDISON/
[29] Y.-W. Tai, J. Jia, and C.-K. Tang, "Soft color segmentation and its applications," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 9, pp. 1520–1537, Sep. 2007.
[30] P. Soille, *Morphological Image Analysis: Principles and Applications*, 2nd ed. Berlin, Germany: Springer-Verlag, 2003.
[31] C.-C. Lin, S. U. Pankanti, K. N. Ramamurthy, and A. Y. Aravkin, "Adaptive as-natural-as-possible image stitching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1155–1163.
[32] G. Zhang, Y. He, W. Chen, J. Jia, and H. Bao, "Multi-viewpoint panorama construction with wide-baseline images," *IEEE Trans. Image Process.*, vol. 25, no. 7, pp. 3099–3111, Jul. 2016.
[33] F. Bellavia and C. Colombo, "Estimating the best reference homography for planar mosaics from videos," in *Proc. Int. Conf. Comput. Vis. Theory Appl. (VISAPP)*, 2015, pp. 512–519.
[34] D. Salomon, *Data Compression: The Complete Reference*. Berlin, Germany: Springer-Verlag, 2006.
[35] X. Zhang and B. A. Wandell, "A probabilistic approach for color correction in image mosaicking applications," *J. Soc. Inf. Display*, vol. 5, no. 1, pp. 61–64, 1997.
[36] 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, Apr. 2004.
[37] Y. Niu, H. Zhang, W. Guo, and R. Ji, "Image quality assessment for color correction based on color contrast similarity and color value difference," *IEEE Trans. Circuits Syst. Video Technol.*, to be published. [Online]. Available: http://ieeexplore.ieee.org/document/7763834/
[38] C. Spearman, "The proof and measurement of association between two things," *Amer. J. Psychol.*, vol. 15, no. 1, pp. 72–101, 1904.
[39] S. Le Moan and P. Urban, "Image-difference prediction: From color to spectral," *IEEE Trans. Image Process.*, vol. 23, no. 5, pp. 2058–2068, May 2014. [Online]. Available: http://ieeexplore.ieee.org/document/6763103/

Fabio Bellavia received his PhD degree in Computer Science from the University of Palermo, Italy, in 2011. His research interests include computer vision and image processing, in particular feature detectors and descriptors, stereo matching, 3D reconstruction, mosaicing, color correction, and related evaluation methods. He is currently a postdoc at the Computational Vision Group, Department of Information Engineering, University of Florence, Italy, where he is developing intelligent vision systems for autonomous underwater vehicles and image forensics, in the context of national and European projects.



Carlo Colombo received his Ph.D. in Robotics from the Sant'Anna School of University Studies and Doctoral Research, Pisa, Italy, in 1996. He is currently associate professor at the Department of Information Engineering, University of Florence, where he leads the Computational Vision Group. His research focuses on computer vision for autonomous robotics, biomedicine and aids to disabled people, advanced human-machine interaction, multimedia systems and image forensics, on which topics he has published over 100 papers. He has been associate/area editor for the journals *Robotics and Autonomous Systems*, *Journal of Multimedia*, and *Computer Vision and Image Understanding*. He has also been General co-Chair of ECCV 2012, and served in 2014 as guest co-Editor of a special issue of the *International Journal of Computer Vision*.