A closed-form solution for multi-view color correction with gradient preservation

Menghan Xia[a], Jian Yao[a,*], Zhi Gao[a,b,*]

[a] *Computer Vision and Remote Sensing (CVRS) Lab, School of Remote Sensing and Information Engineering, Wuhan University, Wuhan, Hubei, PR China*
[b] *Temasek Laboratories, National University of Singapore, Singapore*

ABSTRACT

Color correction across multiple images for consistency is a challenging problem in image mosaicking. To facilitate the global color optimization, existing methods mainly use less flexible correction models, e.g., linear or gamma function, which often struggle to cover the practically existing color discrepancy. In this paper, we present a novel color consistency correction method that models the color remapping function with a parameterized spline curve for each image. Thanks to the flexible model representation, on the one hand, our method has the capability to handle very challenging cases, i.e. hundreds of images with drastic color discrepancy; On the other, some important image properties, e.g., image gradient, contrast, and dynamic range, can be effective formulated with model parameters. Thus, the visual quality of individual images, along with the global color consistency, are comprehensively considered in our specifically designed cost function, which is solved in a closed form via convex quadratic programming. In addition, considering the possibly existing alteration objects within inter-image overlaps, we also propose an functional change detection algorithm with gradient and color features utilized, which guarantees the accuracy of the extracted color correspondences. We have tested the proposed approach on several challenging dataset of diverse genres, which shows that our method substantially outperforms state-of-the-art methods in both visual quality and quantitative metrics.

1. Introduction

We now are in an era of images due to the increasing proliferation of image capturing equipment, such as SLR cameras and smartphones. Moreover, such image flood is fueled by the internet, social media, and other types of data sharing platforms. The easy accessibility of images facilitates the research of a variety of vision-based problems, including image recognition (Alex Krizhevsky and Hinton, 2012), panoramic imaging (Brown and Lowe, 2007), image rendering (Laparra et al., 2017), and virtual navigation (Snavely et al., 2006). However, in most practical applications, images of the same scene might show inconsistent tones due to the different atmosphere illuminations, exposure times and camera response functions. Such photometric discrepancy must be corrected or calibrated in advance of being further applied to those 2D/3D rendering tasks (Xiong and Pulli, 2010; Shen et al., 2016). Otherwise, visually unpleasant color artifacts would be induced, such as the image composited from multi-temporal satellite images shown in Fig. 1. In this paper, we focus on addressing the color correction problem of multi-view image mosaicking.

Color correction plays a critical role in producing a mosaicked image with visually natural appearance. However, it is fairly challenging to remove the color discrepancy across multiple images in a quality-aware manner. On the one hand, it remains a open issue to find an ideal color model that can physically correctly transform all the component images into a consistent status; On the other, the original distribution of image pixels represents some special semantic information, e.g., object contour, material texture, might be disturbed during the color transformation. In some advanced color transfer techniques (Su et al., 2012; Su et al., 2014), the original image information except for colors, like gradient details, are well preserved in the resultant image. However, such technique that only involves two images (one reference and one target image) can hardly be applied to correct inconsistent colors for multiple images. Repeatedly transferring colors through all the image pairs (Xu and Mulligan, 2010; Pan et al., 2010; Xia et al., 2016) only works for the case of a few images, for which the accumulation error would not get that severe. In practice, the mainstream methods formulate it as a global optimization problem to minimize the color discrepancy between all the image pairs. Anyhow, these methods mainly use less flexible models, e.g., linear model (Brown and
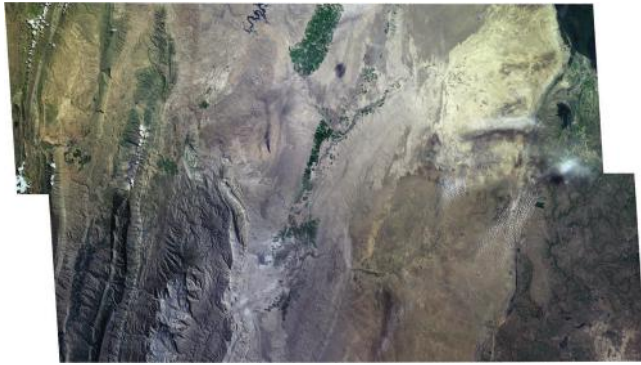

Fig. 1. A mosaicked image that is composited from 8 multi-temporal satellite images. Even processed with seamline optimization and boundary fusion in *Enblend* Available at: http://enblend.sourceforge.net/, the color discrepancy among different source images is still very noticeable.

Lowe, 2007; Pierre Moulon and Monasse, 2013; Shen et al., 2016), gamma model (Xiong and Pulli, 2010; Park et al., 2016), and consider no other visual properties in their cost function, which tend to make the result visually unpleasant unless resorting to tedious post-processing.

To address the problems discussed above, we present an effective color consistency optimization approach based on our previous work (Xia et al., 2017). First of all, we clarify that we still exploit the quadratic spine curve as our color correction model, which is flexible enough to express complex color transformation. In addition, we extend the previous work in the following two aspects: (i) to improve the efficiency and robustness, we exploit the gradient and color features comprehensively, and develop a more efficient change detection algorithm that works effectively to eliminate the alternations for color correspondence estimation. (ii) to improve the performance of gradient details preservation, we incorporate the gradient loss of each image into the cost function. Finally, the global optimal solution of the objective function is obtained with a closed form via convex quadratic programming. Compared with several state-of-the-art methods, the results of extensive experiments demonstrate that our approach illustrates significant superiority in gradient-preserved color consistency and running efficiency.

The paper is organized as follows. Section 2 provides an overview of related works. Section 3 introduces the problem we focus on and defines it mathematically. Section 4 describes the details of our proposed color correction algorithm. Experiments on challenging datasets and comparative evaluation are reported in Section 5. Finally, Section 6 presents conclusions and discussions on future work.

2. Related works

There are two techniques that achieve color consistency between/across images by adjusting color palettes, namely, color transfer and color consistency optimization, as discussed below respectively.

2.1. Color transfer

In Gooch et al. (2001), first proposed the concept of color transfer, which aims to propagate the color characteristics of one image to another. Based on this baseline approach, many works have been proposed. As reviewed in Xu and Mulligan (2010), at the preliminary stage, the method focused on decoupling the correlation of different color channels (Pitié et al., 2007) and conducting elaborative content-based color transfer (Tai et al., 2005). In the past decade, the community focused more on the research to improve the performance in terms of artifacts suppression and detail preservation. In Xiao and Ma (2009), a gradient-preserving model was proposed, which formulated the task of color transfer as an optimization problem. Su et al. (2012) obtained fairly competitive results of grain-free and detail preservation via conducting color mapping and detail boosting separately on the base of

gradient-aware decomposition. Following the same strategy, they developed a more complete and sophisticated algorithm to suppress the corruptive artifacts (Su et al., 2014). However, its detail boosting idea is based on the gradient enhancement technique, thus the retained structures might be incoherent to the original ones in magnitude.

In order to eliminate the distortion of color, Nguyen et al. (2014) exploited the algorithm of Xiao and Ma (2009) in the channel of luminance, followed by the color gamut alignment. Such a technique was good at eliminating color distortion but resulted in poor color fidelity due to the simple model applied for color transfer. In order to realize color mapping more accurately, Hwang et al. (2014) designed an algorithm to independently correct each pixel's color. Therein, an independent affine model for each pixel was obtained via solving a probabilistic moving least square problem based on the color feature correspondences. Nevertheless, the amount and distribution rationality of the matched features cannot be guaranteed. In addition, some color transfer methods based on high-level processing such as semantic segmentation, content recognition have been proposed (Wu et al., 2013; Frigo et al., 2014). More recently, learning-based color transfer methods have been proposed (Wang et al., 2011; Bychkovshy et al., 2011; He et al., 2017; Luan et al., 2017) and marvelous results can be obtained by exploiting the large-scale learnt priors. Nevertheless, as discussed in Section 1, color transfer techniques are intrinsically infeasible to the problem of multi-view color consistency that is the focus of this paper.

Besides, color stabilization (Ilie and Welch, 2005), as a special branch of color transfer technique, particularly targets for addressing the incoherent colors among images of the same scene, captured by different camera setup under the same illumination. Deducing from the general color processing pipeline of cameras, Vazquez-Corral and Bertalmío (2014) propose to estimate a 3×3 matrix and the gamma coefficient, so as to map the color of the source image to the domain of reference image. To address color stabilization for cinema footage, they locally approximate the log-encoding curves by gamma curves, whose gamma-values are accurately estimated by leveraging the corresponding achromatic matches (Vazquez-Corral and Bertalmío, 2016). To stabilize the colors among temporal frames, Frigo et al. (2016) utilize a minimally viable color correction model, in conjunction with an effective estimation of dominant motion. The final solution is a temporally weighted correction, which is feasible to real-time application. As they are, color stabilization methods only concern the color discrepancy induced by different camera setting, and moreover requires all the source images to directly overlap with the reference image. However, it is usually not the case in most multi-view color correction situations, which makes such technique less applicable.

2.2. Color consistency optimization

Color consistency optimization typically targets to improve the color consistency of three or more images in a global optimization manner. Obviously, it excludes those stepwise methods, e.g., sequentially applying color transfer to propagate the color style of one image to all the other images. In Brown and Lowe (2007), Brown et al. performed gain compensation for multiple images via global optimization, and such technique has been incorporated in the panorama software *Autostitch*. To avoid the over-saturation of luminance, Xiong and Pulli (2010) proposed to replace the linear model with a gamma model for the luminance channel. However, this also resulted in a side effect that its performance on luminance consistency decreased. In Qian et al. (2013), a manifold method leveraging on the intrinsic color structure which was obtained in overlap regions was proposed to remove the color inconsistency. However, its high computational cost and strict requirement on pixel-wise alignment severely limited its applications.

In 3D modeling research, Shen et al. (2016) proposed a linear model based method to adjust the color histograms of sequential images that are used for textures mapping of 3D models. This linear model is efficient and easy to implement but its limited flexibility fails to handle significant color discrepancy and provide possibility to consider other
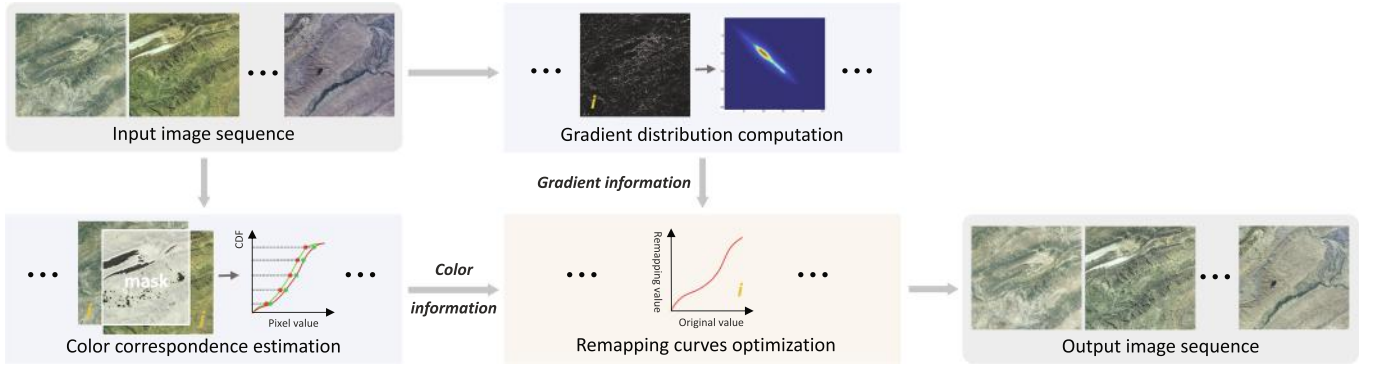
Fig. 2. The overview of our gradient-preserving color correction framework. At first, the color correspondences are estimated from the associated image overlaps with possibly existing alterations masked out. Also, the gradient distribution of input images are collected individually. Then, both the color consistency and gradient preservation are considered during the global optimization on remapping curves.

visual properties, such as gradient or contrast. In photo editing research, HaCohen et al. (2013) modeled the remapping curves by quadratic spline directly and optimized them over a photo collection for a consistent color style. Therein, the color correspondences were estimated via non-rigid dense matching (HaCohen et al., 2011). Such a method is sophisticated enough to correct large color disparity, however, its premise of dense matching renders it really computationally expensive. To deal with such dilemma, Park et al. (2016) proposed to exploit sparse feature matching and estimate the model parameters via robust low-rank matrix factorization. However, comparing to HaCohen et al. (2013), its color correction ability drops due to the inferior model flexibility. Nevertheless, none of these methods has explicitly considered gradient preservation or contrast enhancement during color correction as our work. In addition, those radiometric calibration algorithms (Kim and Pollefeys, 2008; Díaz and Sturm, 2011) that estimated the response functions of sequential images can also be used to eliminate color discrepancy among images in some situations. However, its dependence on specific prior assumptions makes it practically infeasible to color correction in most image mosaicking applications.

3. Problem formulation

The input of our color correction method is a set of image that shares overlap with at least one of other images in the set. Specifically, all the images have to be geometrically aligned so that the image adjacent relationships and the associated overlaps could be used as known information. Our goal is to eliminate the color discrepancy across images while preserve the original details of each individual image.

To adjust the color distribution, we propose to apply an independent color transformation on each input image. In particular, the color transformation is formulated by a monotonically increasing quadratic spline curve (remapping curve) that is parameterized with M anchor points ($M = 6$ as default). In Fig. 3, the red anchors $\{(c_k, \tilde{c}_k)\}_{k=1}^{M}$ are semi-free 2D points, where the horizontal coordinates $\{c_k\}_{k=1}^{M}$ are evenly fixed while the vertical coordinates $\{\tilde{c}_k\}_{k=1}^{M}$ are fully free. Particularly, the horizontal coordinates is set to determine the range of the remapping curve, which just covers the range of original intensity values $[v_s, v_e]$ of each image; And the vertical coordinates serves as the real model parameters that finally determines the shape of the remapping curve. Then, for any image of the sequence, the color transformation is expressed as:

$$f_i = \arg(\tilde{c}_1^i, \tilde{c}_2^i, ..., \tilde{c}_M^i), \tag{1}$$

Note that we only consider single-channel case here for simplicity. For color images, the color correction is performed in *YCbCr* space, where the luminance channel and chromatic channels are separated and each channel has specific upper and lower bounds, which facilitate the quality-aware color consistency optimization as detailed in Section

4.2. With the model defined, we aim to find a group of optimal remapping curves that make all the images present consistent color while their original gradients are preserved as much as possible. To this end, all the desired properties are formulated as a differentiable cost function, then it is solved by convex quadratic programming.

4. Proposed color correction method

As illustrated in Fig. 2, there are two stages in our color correction framework: data preparing stage that includes color correspondence extraction and spatial gradient statistics, and model optimization stage.

4.1. Color correspondences extraction

Based on the image adjacency topology, we estimate the color correspondences in the overlaps of image pairs. Similar to Pierre Moulon and Monasse (2013), we exploit the color histogram of overlap regions to extract correspondences. Specifically, for each image pair, we first compute the cumulative distribution functions (CDFs, namely the cumulative histogram of image values) within the overlap region, then take the corresponding equal-probability quantiles of the two CDFs as the color correspondences. Particularly, in case of some alteration objects existing, it is important to detect and remove them from the
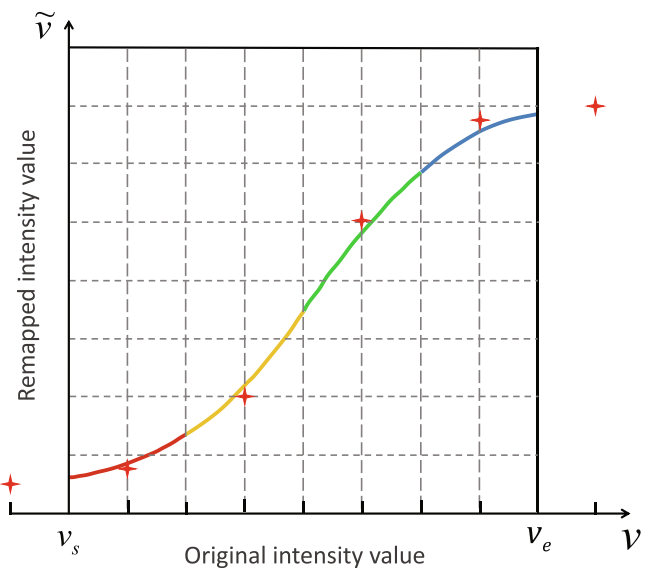

Fig. 3. Color remapping curve modeled with a piecewise quadratic spline. By defining the horizontal distribution of anchors (red cross points), the curve just covers the intensity range of the original image $[v_s, v_e]$.
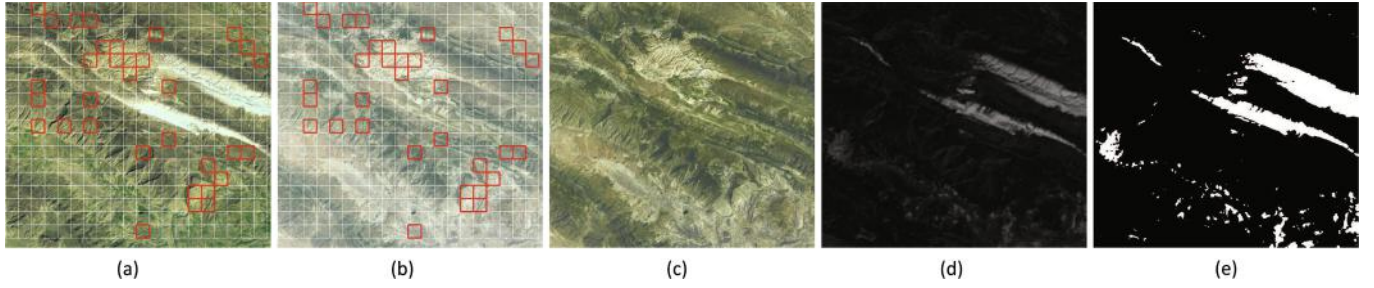
Fig. 4. (a) & (b): the overlap regions of an image pair where the selected seed grids marked with red boxes. (c): color corrected result of (b) by referring (a) based on the color correspondences extracted from seed grids. (d): the differential image of (a) and (c) with channels averaged. (e): the final binary mask with white pixels indicating alteration regions.

overlaps before computing the CDFs. Otherwise, the accuracy of the obtained correspondences would get affected.

Alteration detection algorithms (Shin et al., 2010) are widely used in the field of surface resource survillance. These algorithms usually have a good accuracy in detection result, though involve high computational complexity in the meanwhile. Anyhow, in the context of histogram matching, the detection accuracy is not that desired only if the real alterations are fully contained in our detected results. Considering this, we develop a fast alteration detection method that is functionally fit to our color correction framework. That is, we firstly try to find a small set but reliable color correspondences, then they are utilized as the seeds to remove the major color discrepancy between the two target images, which is followed by computing the differential image. Here, we adopt an effective strategy to acquire such corresponding seeds. According to our experiments, the histogram of gradient orientation (HOG) of local patches is a robust similarity metric that is relatively insensitive to color change. Given an image pair $\{\mathbf{I}_i, \mathbf{I}_j\}$, we divide their overlap regions ($\mathbf{I}_{ij}$ and $\mathbf{I}_{ji}$) into grids (grid size: 32×32 pixels) evenly and compute the bin vector of HOG $\mathbf{h}$, followed by estimating the average gradient value g of each grid, as the example shown in Fig. 4 (a) and (b). Let the grid gradient statistic of $\mathbf{I}_{ij}$ and $\mathbf{I}_{ji}$ be $\{(\mathbf{h}_k^i, g_k^j)\}_{k=1}^{S}$ and $\{(\mathbf{h}_k^k, g_j^k)\}_{k=1}^{S}$, where S is the number of grids. In order to select the most reliable grids with consistent contents, we sort them by the gradient distance in ascending order, where the gradient distance of k-th grid is calculated as:

$$D_{grad}^k = \frac{1}{L} \sum_{l=1}^{L} \frac{|\mathbf{h}_i^k(l) - \mathbf{h}_j^k(l)|}{\max(\mathbf{h}_i^k(l), \mathbf{h}_j^k(l))}, \tag{2}$$

where L is the bin number of HOG ($L = 36$ as default). In addition to the
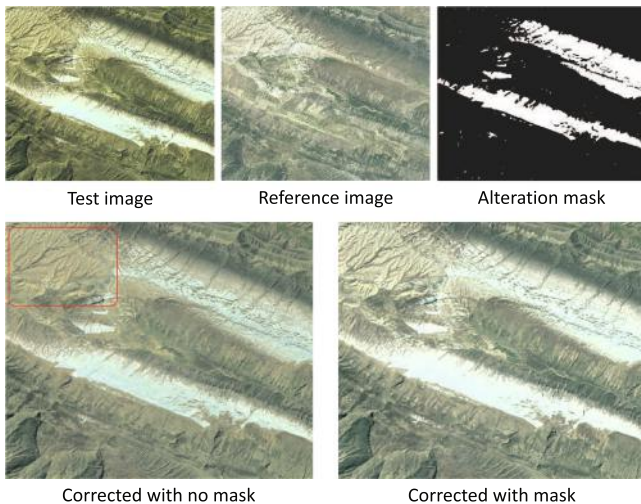

Fig. 5. Comparison between the resultant images corrected with and without using masks. The red box marks the problematic region.

gradient distance, we also consider the gradient magnitude of each grid, because the weak-texture areas with low gradient magnitude (e.g. lake or cloud region) make the gradient similarity noisy and unreliable. In practice, all the low-confident grid subset with $\{\min(g_i^k, g_j^k) < t\}_{k=1}^{S}$ ($t = 1.0$ as default) are eliminated in advance. Then, we take the least 10% in gradient distances from the rest grids as the seed regions, e.g., the red girds in Fig. 4 (a) and (b).

Based on the corresponding pixel colors in the seed grids, we employ histogram mapping to correct the color of one image with the other as reference. When the major color difference disappears, we can compute the differential image between $\mathbf{I}_{ij}$ and $\mathbf{I}_{ji}$, which are preprocessed by mean filtering to handle the possible misalignment in advance. Then, the differential image are classified into two clusters by K-Means algorithm (Krishna and Murty, 1999), where the cluster with the larger mean value indicates those alteration pixels if the inter-cluster distance exceeds a certain threshold, otherwise no alteration exists. Finally, a binary mask is obtained to filter out the alteration pixels, as shown in Fig. 4(e). We declare that the accuracy of this auxiliary algorithm can not compare with those formal alteration detection approaches, but the efficiency and acceptable accuracy that make it tightly matches the whole framework for color consistency correction. In Fig. 5, the comparative results illustrate the effectiveness of our generated alteration mask in improving the color correction result.

4.2. Model parameters optimization

With the color correspondences acquired, we discuss how to design the global optimization function and its solver. Given an image sequence $\{\mathbf{I}_i\}_{i=1}^{N}$, we attempt to estimate $\{f_i\}_{i=1}^{N}$, a set of color transforms which could achieve color consistency across images and gradient preservation for each individual image. In our approach, the remapping curves are optimized in each channel of YCbCr space independently. For any of the three channels, we formulate the cost function as:

$$E = \sum_{\mathbf{I}_i \cap \mathbf{I}_j \neq \varnothing} w_{ij} E_{color}(f_i, f_j) + \lambda \sum_{i=1}^{n} E_{quality}(f_i)$$
$$\text{subject to:} C_{rigid}(f_i), \forall i \in [1, N], \tag{3}$$

where w_{ij} is a weight and the value is set proportional to the overlap region between $\mathbf{I}_i$ and $\mathbf{I}_j$. We set $\lambda = \xi \frac{K}{M}$ to balance the two terms, where $K = 16$ in our experiments, and $\xi \in [0.5, 5]$ is recommended. In fact, our method is insensitive to λ.

To evaluate the color consistency between images, we define the color term to penalize the deviation, as below:

$$E_{color}(f_i, f_j) = \sum_{k=1}^{K} ||f_i(v_k^i) - f_j(v_k^j)||_2, \tag{4}$$

where $||\bullet||_2$ denotes L_2 norm. $\{v_k^i, v_k^j\}_{k=1}^{K}$ denotes the color correspondences between $\mathbf{I}_i$ and $\mathbf{I}_j$.
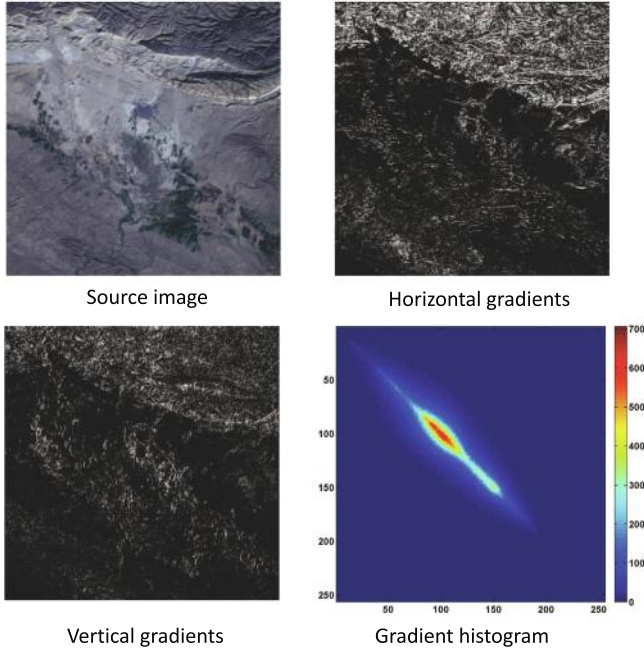
Fig. 6. 2D gradient histogram counted from both horizontal gradient map and vertical gradient map. It is easy to find that the major gradient components of an image only take up a small part of the statistical space.

As a unary term, $E_{quality}(f_i)$ imposes other constraints, i.e. gradient preservation, stretching of the dynamic range and parameter regularization, on the color transformation of each image.

$$E_{quality}(f_i) = \frac{\alpha}{S_i} \sum_{s=1}^{S_i} ||(f_i(p_s^i) - f_i(q_s^i)) - (p_s^i - q_s^i)||_2$$
$$- \beta||f_i(u_{0.05}^i) - f_i(u_{0.95}^i)||_2 + \sum_{t=1}^{M-1} ||f_i(\dot{v}_t^i) - \dot{v}_t^i||_2, \tag{5}$$

where (p_s^i, q_s^i) denote the s-th key gradient element of $\mathbf{I}_i$. Naturally, gradient element (p, q) is defined by two adjacent values p and q, which define the 2D position of a non-zero bin in the gradient histogram as shown in Fig. 6. p_i denotes the amount of non-zero bins in the gradient histogram of $\mathbf{I}_i$ with symmetric bins merged, since gradient components (p, q) and (q, p) have the same effect in the gradient preservation term. Note that we only use the top 90% gradient elements in our optimization. For contrast issue, the second term penalizes the dynamic range (u_θ^i denotes the θ-percentile of the CDF of $\mathbf{I}_i$). Such a term contributes to preventing trivial solutions that all the images present a consistent but dim tone. It should be noted that the constraints of gradient preservation and dynamic range are enforced in the luminance channel only because the main gradient (or structure) information is in this channel, as illustrated in Fig. 7. Specifically, we set $\alpha = 20M$ and $\beta = 2M$ for the luminance channel (Y), set $\alpha = 0$ and $\beta = 0$ for the two chromatic channels (Cb,Cr). In the last term of $E_{quality}$, $\{\dot{v}_k^i\}_{k=1}^{M-1}$ represent the horizontal coordinates of joint points (which are marked in different colors in Fig. 3). According to the definition of quadratic spline curve, we have
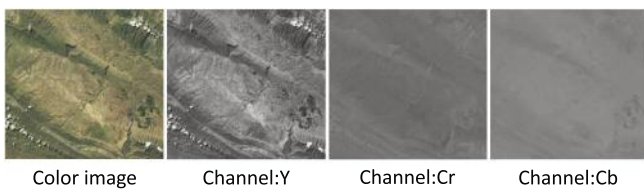

Fig. 7. A color image represented by channels in YCbCr space. As observed, most of gradient information is contained in the luminance channel Y.

$\dot{v}_k^i = \frac{c_k^i + c_{k+1}^i}{2}$, $k = \{1, 2, ..., M-1\}$. The contribution of this term is twofold: i) encouraging the transformed images similar to their original appearance; ii) bundling each image's model parameters in a unified manner.

As a qualified remapping curve, two fundamental requirements, i.e. monotonic increasing and mapping range within the color gamut, should be guaranteed. And it is formulated as the hard constraint $C_{rigid}(f_i)$:

$$C_{rigid}(f_i): \begin{cases} \tau_l \leqslant f'_i(v_k^i) \leqslant \tau_u, \forall v_k^i \in [v_s, v_e] \\ v_{min} \leqslant f_i(u_{0.01}^i), f_i(u_{0.99}^i) \leqslant v_{max}, \end{cases} \tag{6}$$

where τ_l and τ_u define the lower and upper bounds of the slope of the remapping curve. $[v_{min}, v_{max}]$ defines the range of gamut in the corresponding channels, such as $[0, 255]$ of Y. In particular, the slop range (especially its lower boundary) is a trade-off between color consistency and detail preservation. Fortunately, a more strict gradient preservation strategy has been imposed in the term $E_{quality}$, thus we simply set a relatively loose slop range $[0.3, 5]$ for the remapping curves of every channel. Finally, the model parameters that minimize the cost function are solved by convex quadratic programming (Goldfarb and Idnani, 1983), which is detailed in Section 4.3.

4.3. Implementation details

Our color correction model is a kind of piecewise interpolation function, where each local curve segment is determined by only three continuous anchors. Given a pixel value v_k^i, which is in the scope of anchors $\{(c_r^i, \tilde{c}_r^i)\}_r^{r+2}$, we can estimate its remapped value $\tilde{v}_k^i$ according to the quadratic spline interpolation equation:

$$\begin{cases} v_k^i = \frac{1}{2}[(1 - 2t + t^2)c_r^i + (1 + 2t - 2t^2)c_{r+1}^i + t^2 c_{r+2}^i] \\ \tilde{v}_k^i = \frac{1}{2}[(1 - 2t + t^2)\tilde{c}_r^i + (1 + 2t - 2t^2)\tilde{c}_{r+1}^i + t^2\tilde{c}_{r+2}^i], \end{cases} \tag{7}$$

where the unknowns include the interpolation coefficient $t \in [0, 1]$ and $\tilde{v}_k^i$. Here, $\tilde{v}_k^i$ can be expressed as the function of model parameters $\{\tilde{c}_r^i\}_r^{r+2}$ by eliminating t, namely f_i. Actually, the interpolation coefficients of all color correspondences could be computed in advance of the global optimization.

Based on the remapping function f_i, we further deduce the formula of its derivative f'_i. According to Eq. 7, the slope value at v_k^i is expressed as:

$$f'_i(v_k^i) = \frac{\partial \tilde{v}^i / \partial t}{\partial v^i / \partial t} = \frac{(\tilde{c}_r^i - 2\tilde{c}_{r+1}^i + \tilde{c}_{r+2}^i)t + \tilde{c}_{r+1}^i - \tilde{c}_r^i}{(c_r^i - 2c_{r+1}^i + c_{r+2}^i)t + c_{r+1}^i - c_r^i} \tag{8}$$

where $c_r^i - 2c_{r+1}^i + c_{r+2}^i = 0$ since $\{c_r^i\}_{i=1}^M$ are constants (as shown in Fig. 3). So, $f'_i(v_k^i)$ equals to a linear function with the free variable $t \in [0, 1]$. Thus, the constraint $f'_i \in [\tau_b, \tau_u]$ can be expressed as the linear inequalities:

$$\begin{cases} \tau_b \leqslant \frac{\tilde{c}_{r+1}^i - \tilde{c}_r^i}{c_{r+1}^i - c_r^i} \leqslant \tau_u, t = 0 \\ \tau_b \leqslant \frac{\tilde{c}_{r+2}^i - \tilde{c}_{r+1}^i}{c_{r+1}^i - c_r^i} \leqslant \tau_u, t = 1, \end{cases} \tag{9}$$

Now, we can express our global optimization function with the unknown parameters $\{(\tilde{c}_1^i, \tilde{c}_2^i, ..., \tilde{c}_M^i)_{i=1}^N\}$. Substituting Eq. 7 into Eq. 4 and Eq. 5, the formula of optimization function in Eq. 3 turns into a quadratic polynomials. Substituting Eq. 7 and Eq. 8 into Eq. 6, the feasible region of the model parameters forms a convex set. Then, Eq. 3 is transformed to the standard form of constrained quadratic programming and its minimization problem can be solved with a *convex quadratic programming algorithm*[1] efficiently.

Parameter settings: There are five hyperparameters in our

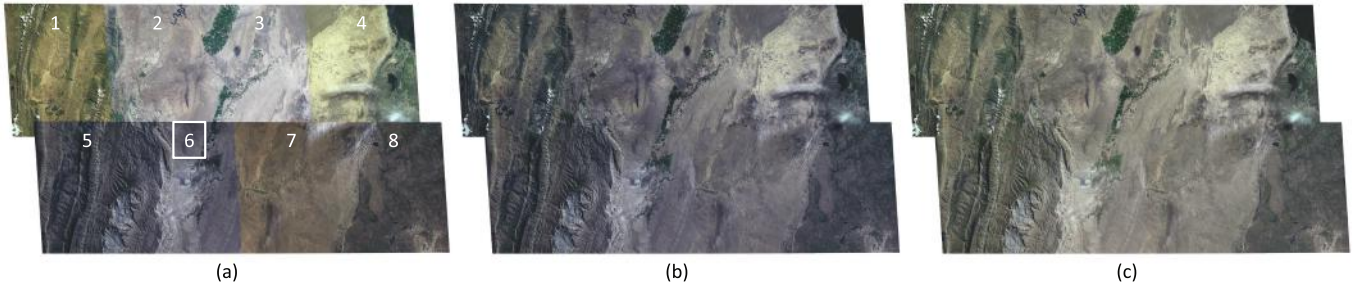[1] QuadProg + +: https://github.com/liuq/QuadProgpp

Fig. 8. (a) Original images with geometrically aligned. (b) Our corrected result with image 6 specified as a reference, where the purple tone has been transferred to its neighborhood successfully. (c) Our corrected result with no reference assigned (bottom). Particularly, (b) and (c) are post-processed by *Enblend* for boundary fusion.

Table 1
Dataset list.

Dataset	Image Size	Image Num.	Mean Overlap	Platform
LAND	2758 × 2290	8	20 %	ZY-3
COAST	2270 × 1939	16	20 %	ZY-3
MOUNTAIN	2415 × 1676	20	15 %	GF-3
VILLAGE	800 × 600	130	60 %	UAV
KITCHEN	800 × 1200	15	50 %	SLR Camera
CAMPUS	480 × 640	18	40 %	SLR Camera

optimization function: anchor number of each curve M, color correspondence number of each image pair K and three balance coefficients λ, α and β. The recommended values ($M = 6$, $K = 16$) of the first two parameters are applicable to almost all kinds of data. The role of the latter three coefficients is to balance the numerical magnitudes of different energy terms, and their default values are given according to our testing experience. During the testing, we find that these three parameters are insensitive to the scenes. In our evaluation experiments, we simply use all the default parameter values in the experiments of this paper and the results are quite competitive and stable.

User Interaction: From Section 2.2, we observe that all the images are treated equally in the global optimization which targets for a globally consistent appearance. Nevertheless, sometimes, we might want to see the optimization result showing our preferred color style. This can be realized by setting a few reference images whose colors are fixed and would be propagated to others during the global optimization. That means, users can interact with the optimization procedure by selecting a few reference images, whose colors might either be the original one or manually edited to the users preferred style. In Fig. 8, we illustrate the comparative results that are generated by our method with and without reference image appointed. As observed, the color style of the assigned reference image is retained on itself and transferred to other neighborhood images effectively.

5. Experimental results

Dataset: We evaluate our approach on six datasets that cover various data genres, including four remote sensing satellite/UAV datasets and two normal panorama image datasets. The key information of our datasets are described in Table 1. LAND and COAST are multi-temporal images acquired by Chinese ZY-3 satellite, which present obvious color discrepancy; MOUNTAIN also consists of multi-temporal images acquired by Chinese GF-2 satellite, where the tonal inconsistence is mainly resulted by the variations of illumination and vegetation colors; VILLAGE consists of hundreds of UAV images that are captured at a flight height of 786 meters from the ground. KITCHEN and CAMPUS are normal image sequences acquired for 360° panorama composition. Particularly, the original colors of VILLAGE and KITCHEN are disturbed to some extent purposely, so as to raise the challenge for color correction.

Evaluation Metrics: Since PSNR (Huynh-Thu and Ghanbari, 2008) and SSIM (Zhou Wang et al., 2004) are infeasible to evaluate similarity between non-pixel-wise corresponding images, we define two more practically effective metrics to evaluate the results quantitatively: Color Distance (CD) is used for computing color inconsistency between images, and Gradient Loss (GL) is used for computing gradient variation for an image undergone processing. Specifically, they are formulated as:

$$\begin{cases} \text{CD} = \displaystyle\sum_{\mathbf{I}_i \cap \mathbf{I}_j \neq \varnothing} \bar{w}_{ij} \frac{\Delta \text{H}(\hat{\mathbf{I}}_{ij}, \hat{\mathbf{I}}_{ji})}{N_b} \\ \text{GL} = \dfrac{1}{n} \displaystyle\sum_{i=1}^{n} \frac{\Delta \text{G}(\mathbf{I}_i, \hat{\mathbf{I}}_i)}{N_p} \end{cases}, \tag{10}$$

where $\mathbf{I}_i$ is the source image, $\hat{\mathbf{I}}_i$ denotes the corrected result, $\hat{\mathbf{I}}_{ij}$ represents the region of $\hat{\mathbf{I}}_i$ overlapped with $\hat{\mathbf{I}}_j$. The normalized weight $\bar{w}_{ij}$ is set proportional to the area of $\hat{\mathbf{I}}_{ij}$ ($\sum \bar{w}_{ij} = 1$). Moreover, $\Delta \text{H}(\cdot)$ computes the difference between histograms by bins, $\Delta \text{G}(\cdot)$ computes the difference between gradient orientation maps by pixels. N_b is the bin number of a histogram, and N_p is the pixel numbers of $\mathbf{I}_i$.

5.1. Effects of gradient preservation

The image gradient is one of the most important low-level features that represent semantic information, such as object structures, surface textures, etc. However, correcting the color of an image, alike to any image processing that involves pixel value modification, would inevitably change the gradient distribution of this image as side effects. Considering this problem, we add a special energy term in $E_{quality}$ to penalize the gradient variation during color consistency optimization, compared to our previous work (Xia et al., 2017). To see the effect brought by this modification, we conduct a comparative experiment on a challenging dataset LAND that covers lots of weak-texture regions. Here, for meaningful analysis in limited space, we focus on the comparison among the results of Shen et al. (2016),Park et al. (2016), our approach without gradient preservation term ($\alpha = 0$), and our full approach in Fig. 9, and the fully quantitative comparison is depicted in Fig. 10. The detailed description of these competing methods are presented in Section 5.2. Particularly, to present the color contrast between adjacent images, both the input and corrected images are displayed in the form of a mosaicked image with the component images overlaid sequentially.

Based on both qualitative and quantitative results, our approach has been verified to perform better than other algorithms in terms of color consistency. This benefit is greatly from the flexibility of our color correction model, though its flexibility is prone to causing gradient loss. As the enlarged regions show, the gradient loss of (d) is more obvious than that of (b) and (c) in Fig. 9. As expected, the gradient preservation term helps to retain most of the gradient information in the corrected result, as shown in Fig. 9 (e). Note that the gain in gradient preservation is also associated with the slightly decreased performance in color consistency, since this additionally introduced constraint may limit the flexibility of the model. In summary, such a compromise is worthwhile because the increased gradient information is visually observable while the decreased color consistency
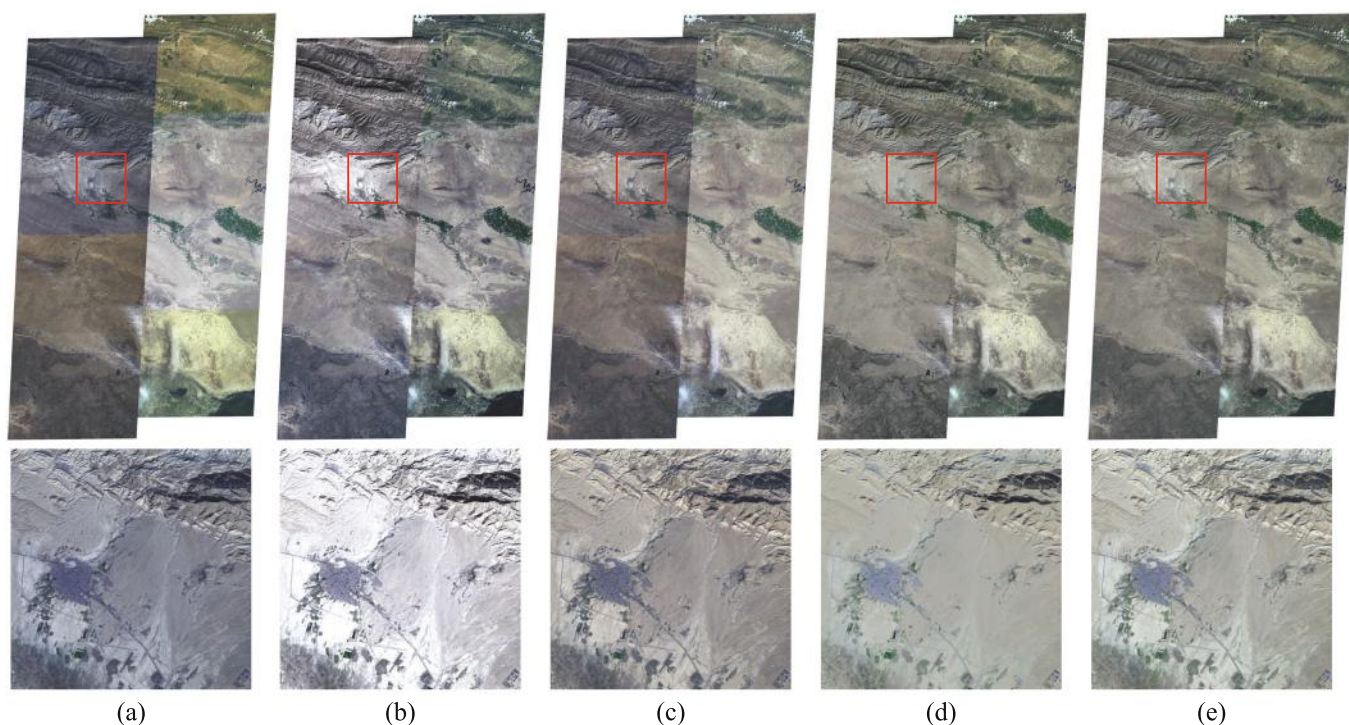
(a)	(b)	(c)	(d)	(e)

Fig. 9. Visual comparisons on gradient preservation of different methods on LAND: the input images (a), the corrected results of Shen et al. (2016) (b), Park et al. (2016) (c), our approach without gradient preservation term (d) and our full approach (e). Specifically, a challenging region marked with red box is enlarged for detailed observation.
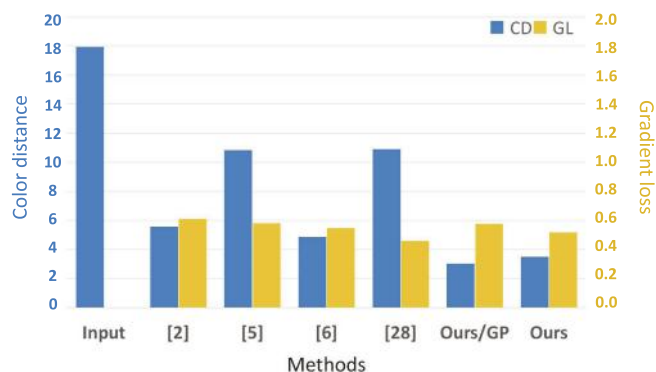

Fig. 10. Numerical comparisons on the gradient preservation of different methods on LAND: the source images, the corrected results of Brown and Lowe (2007),Xiong and Pulli (2010), Shen et al. (2016),Park et al. (2016), our method using no gradient preservation (Ours/GP) and our method.

nearly causes no noticeable defects. The comprehensive quality of (e) is improved over (d) using our gradient preservation strategy.

Using linear function model Brown and Lowe (2007) and Shen et al. (2016) have roughly similar performance in gradient preservation in theory. Comparing to Brown and Lowe (2007),Xiong and Pulli (2010) that employ gamma model to avoid overly saturated luminance, their performance is inferior in color correction. Fortunately, because of the small pixel variation, they are better in preventing gradient loss. Noticeably, Park et al. (2016) has the best performance in gradient preservation, which can be explained by the albedo-based correction model which is supposed to be more similar to the practical imaging principle. However, compared to our method, such a model has an inferior capability in multi-view color correction, which is reflected both quantitatively and qualitatively.

5.2. Comparative evaluation

Based on our survey on color consistency correction, all the mainstream methods work on the global optimization on a individually defined correction model, including: gain model based Brown and Lowe (2007), gamma-gain model based Xiong and Pulli (2010), linear model based Pierre Moulon and Monasse (2013), Shen et al. (2016), and albedo calibration model based Park et al. (2016). The detailed comparisons in model formulation, color correspondence estimation, optimization solver, and working color space, are made in Table 2. In fact, the gain model is a special linear model with zero-valued additive coefficient, and gamma-gain model combines gamma model for luminance channel and gain model for chromatic channels. Besides, Shen et al. (2016) uses the same correction model as Pierre Moulon and Monasse (2013) but solves parameters with an more advanced optimization function, so only Shen et al. (2016) is compared here. Additionally, our C++ implemented source code is publicly available at Github[2].

5.2.1. Performance on remote sensing images

Remote sensing image mosaicking usually involves stitching sequential images distributed in multiple strips, which makes each image associated with omni-directional neighbors. Moreover, we sometimes need to use remote sensing images that are captured at different times or even by different observing platform. These application requirements actually pose challenge to color correction methods. In this section, we compare our proposed method with those competing methods introduced above, on three challenging dataset: COAST, MOUNTAIN, and VILLAGE. Firstly, we illustrate the comparative results of COAST in Fig. 11, from which we find that (f) has a noticeable superiority over any other results in color consistency. This is the benefits of our flexible correction model, which has a superior capability in color transformation. In comparison, (b) shows an hazy appearance as a whole,

---

[2] Download: https://github.com/MenghanXia/CoColour

Table 2

Model formulation of competing methods: Brown and Lowe's (2007), Xiong and Pulli's (2010), Shen et al.'s (2016), Park et al.'s (2016) and ours.

Method	Model	Correspondence	Solver	Color Space
Brown and Lowe (2007)	$\hat{v} = av$	Mean of overlap	Least square algorithm	RGB
Xiong and Pulli (2010)	$\hat{v} = v^\gamma$ & $\hat{v} = av$	CDF of overlap	Least square algorithm	YCbCr
Shen et al. (2016)	$\hat{v} = av + b$	CDF of overlap	Least square algorithm	YCbCr
Park et al. (2016)	$\hat{v} = (av)^\gamma$	SIFT points	Matrix factorization Cabral et al. (2013)	RGB
Ours	$\hat{v} = f(v)$ in Eq. 7	CDF of overlap	Quadratic programming	YCbCr

especially in the right strip as marked with blue box; (c) suffers the low contrast globally with obvious color discrepancy residuals; (d) contains some overly dark regions and unnaturally bleeding colors in the bottom-left image; (e) has some residual yellow tone that is not completely removed (as marked in blue boxes), along with obvious tonal difference remained across image strips. As the contrast of all the corrected images will be stretched linearly as a post-processing in Shen et al. (2016), (d) illustrates the highest contrast but some regions are darkened overly. On the contrary, without any contrast concern, (b) suffers from dimming tone, which are not as sharp as (d), (e) and (f), which makes it visually unpleasant. Somehow, due to the effect of gamma model, (c) is bright overall and has no problem of dimming tone, but the limited flexibility causes the worst consistency for (c). The quantitative evaluation on color consistency and gradient preservation is illustrated in Table 3, which reflects the coherent conclusion we observed on the visual results. Besides, thanks to the gradient preservation design, our approach achieves the second-best performance in gradient preservation in COAST, only inferior to Park et al. (2016) slightly.

In Fig. 12, the color corrected results of MOUNTAIN are illustrated. As we can see, except for the two images in the top left, the color discrepancy of MOUNTAIN is relatively moderate, and the major problem lies in the illumination variations. Overall, all these methods works well on this dataset, but, there are still two challenging points that identify the performances of different methods. The first point is the input color of the top-left regions, which are particularly different with the other part. From Fig. 12, we find that all the results, except (b) and (f), suffer from the inconsistent color between the top-left part and other part. As marked with blue box, the image in (c) even shows the artifact of color shift, i.e. an kind of abnormal yellow tone. The second point is the illumination discrepancy between different strips. The residual illumination inconsistency is marked with red boxes on the results, e.g., (b), (c), (d) in Fig. 12. As expected, (f) generated by our method presents the globally consistent color and illumination exclusively. Again, the quantitative results in Table 3 reflects the same conclusion.

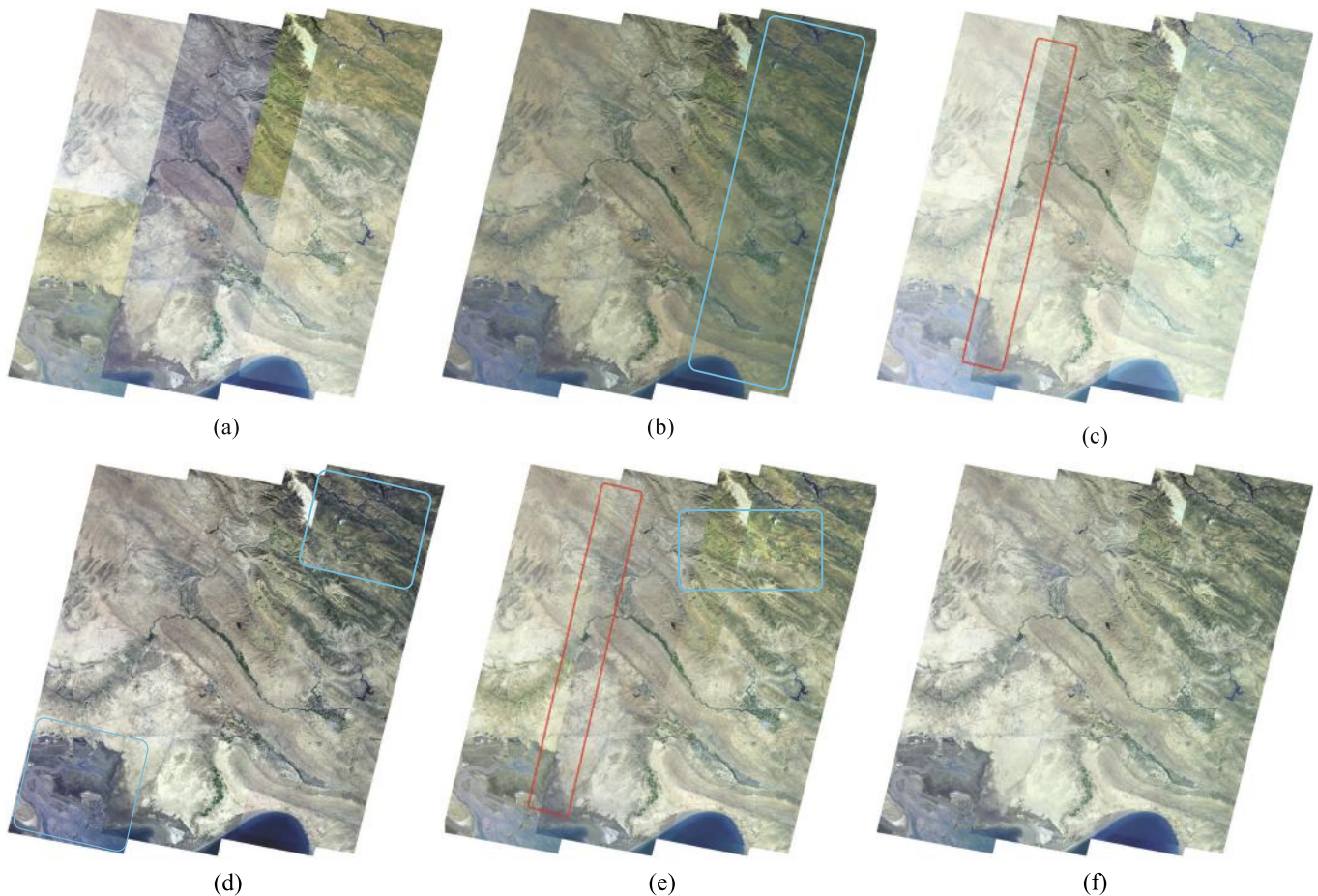In Fig. 13, we further evaluate the performance of our method on a

Fig. 11. The color correction results on COAST dataset: the input images (a), the corrected results of Brown and Lowe (2007) (b), Xiong and Pulli (2010) (c), Shen et al. (2016) (d), Park et al. (2016) (e) and ours (f). To indicate the existing differences, the red and blue boxes highlight the regions with color inconsistency and abnormal tone respectively.

Table 3
Quantitative evaluation on different methods. Smaller color disparity (CD) and gradient loss (GL) indicates better quality.

Method	COAST		MOUNTAIN		VILLAGE		KITCHEN		CAMPUS	
	#CD	#GL	#CD	#GL	#CD	#GL	#CD	#GL	#CD	#GL
Input	13.01	0.000	13.56	0.000	18.07	0.000	23.27	0.000	13.03	0.000
Brown and Lowe's (2007)	4.93	0.642	4.99	0.598	4.42	0.702	7.75	1.102	8.25	0.543
Xiong and Pulli's (2010)	7.26	0.645	9.04	0.532	10.80	0.636	13.93	1.079	10.82	0.503
Shen et al.'s (2016)	4.78	0.545	5.05	0.632	3.79	**0.635**	6.29	1.131	5.01	0.536
Park et al.'s (2016)	6.56	**0.472**	10.45	**0.413**	5.21	1.126	8.67	**0.937**	7.37	**0.390**
Proposed method	**3.53**	0.519	**2.89**	0.5.89	**1.77**	0.636	**4.18**	1.078	**4.23**	0.514

large-scale dataset VILLAGE consisting of 130 images. Although the input images display an intensive color and illumination variations, all the methods work properly to achieve greatly improved results. Anyway, some visually unpleasant effects still exist in (b), (c), (d), (e), which just reflect those intrinsic drawbacks of these algorithms. As discussed before, the inconsistent illumination in (c) is resulted from the limited flexibility of gamma model used in Xiong and Pulli (2010). As (b) shows, the corrected result of Brown and Lowe (2007) suffers the same problem as Shen et al. (2016), i.e. unpleasant dynamic range. It is because that their optimization function only considers the color disparity between adjacent images and dynamic range of the corrected

images are never concerned. On the contrary, there is a special term in our optimization function that penalizes each image's dynamic range to narrow down so our results always show satisfying dynamic range. As (e) shows, Park et al. (2016) works well in optimizing color saturation but results in inferior tonal consistency. Abnormally, the always best method (Park et al., 2016) in gradient preservation turns the worst in VILLAGE as shown in Table 3. It might be explained that the gradient distribution of the manually edited source images of VILLAGE makes the gradient loss metric (GL) less reliable. Anyway, alike to the result shown before, our method achieves the most visually pleasant result illustrated in (f).
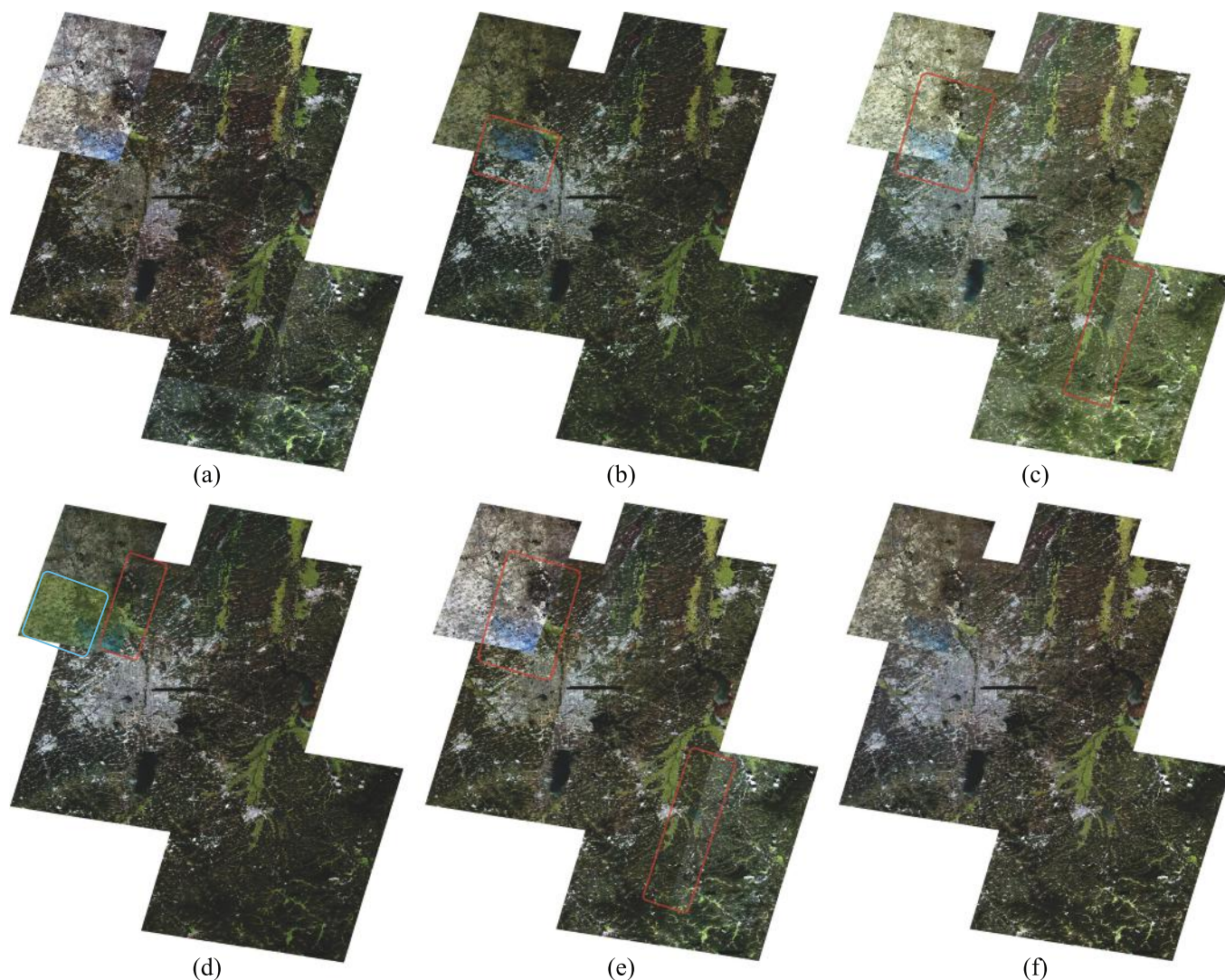

Fig. 12. The color correction results on MOUNTAIN dataset: the input images (a), the corrected results of Brown and Lowe (2007) (b), Xiong and Pulli (2010) (c), Shen et al. (2016) (d), Park et al. (2016) (e) and ours (f). To indicate the existing differences, the red and blue boxes highlight the regions with residual tonal disparity and abnormal tone respectively.
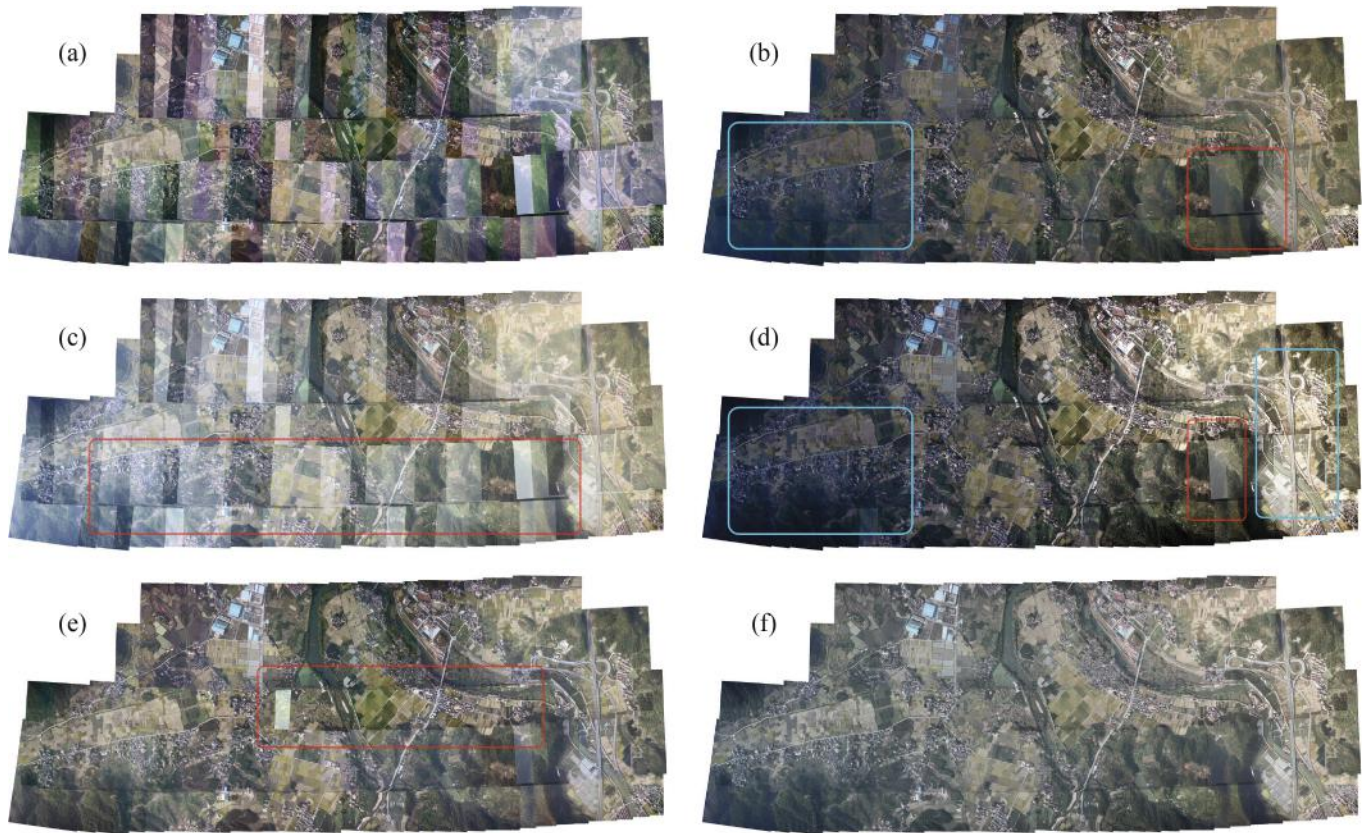
Fig. 13. The color correction results on VILLAGE dataset: the input images (a), the corrected results of Brown and Lowe (2007) (b), Xiong and Pulli (2010) (c), Shen et al. (2016) (d), Park et al. (2016) (e) and ours (f). To indicate the existing differences, the red and blue boxes highlight the regions with residual tonal disparity and abnormal to.ne respectively.
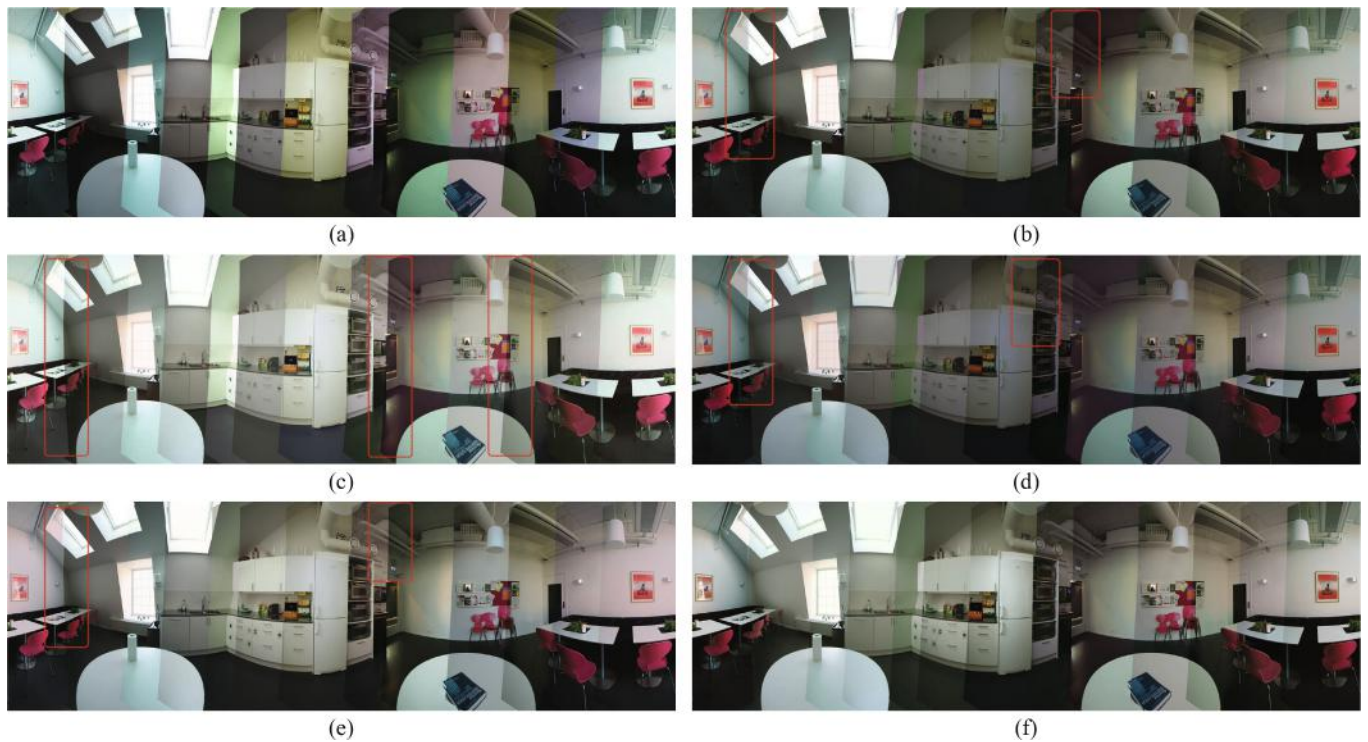

Fig. 14. The color correction results on KITCHEN dataset: the input images (a), the corrected results of Brown and Lowe (2007) (b), Xiong and Pulli (2010) (c), Shen et al. (2016) (d), Park et al. (2016) (e) and ours (f). To indicate the existing differences, the red boxes highlight the regions with residual.tonal disparity.

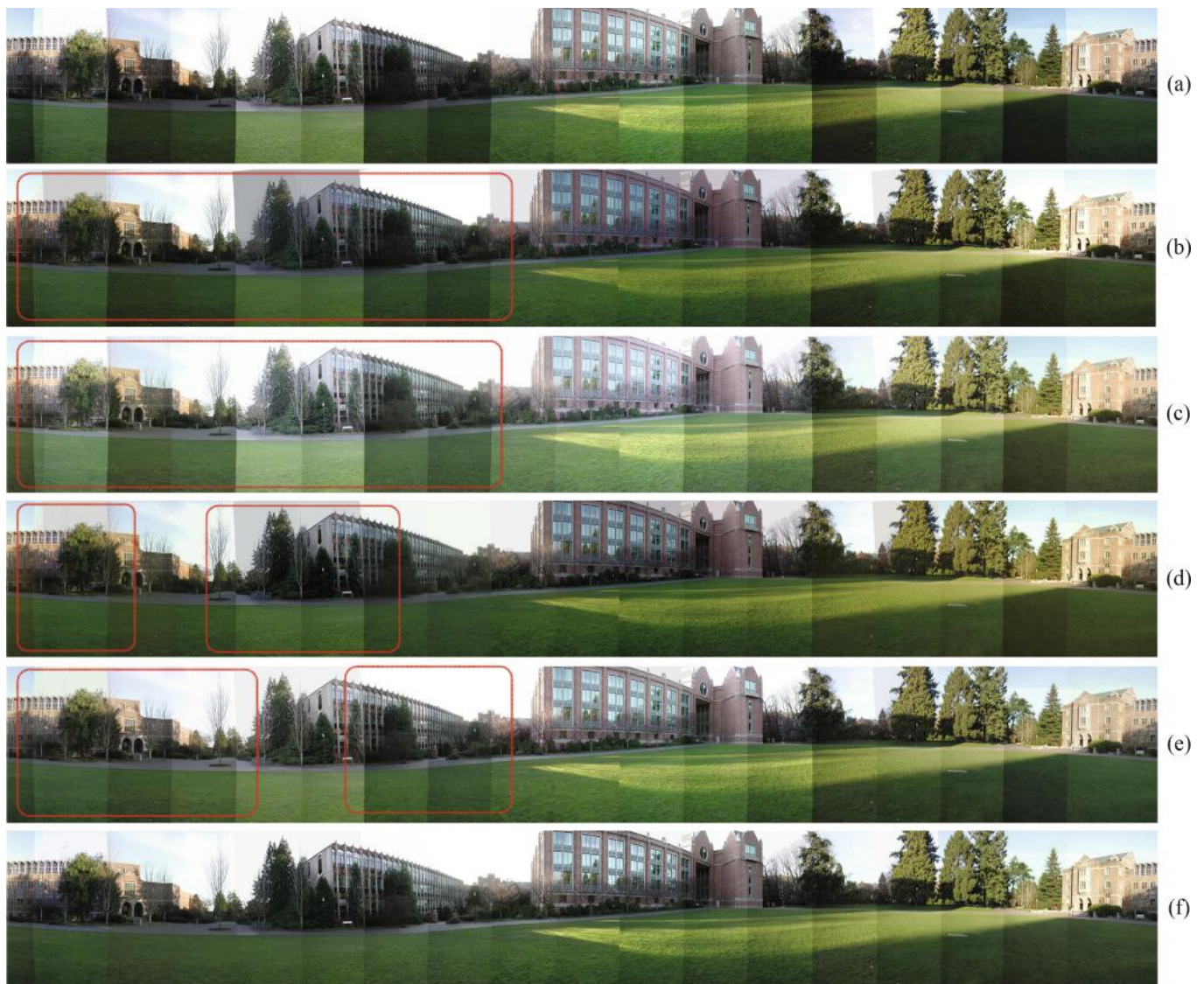**Fig. 15.** The color correction results on CAMPUS dataset: the input images (a), the corrected results of Brown and Lowe (2007) (b), Xiong and Pulli (2010) (c), Shen et al. (2016) (d), Park et al. (2016) (e) and ours (f). To indicate the existing differences, the red boxes highlight the regions with residual tonal disparity.

5.2.2. Performance on panorama images

To test our method on normal images, we perform comparative experiments on two publicly available panorama datasets. Fig. 14 illustrates the input images and corrected results of KITCHEN. It is easy to observe that (f) shows the best color consistency, which is also supported by the quantitative evaluation in Table 3. On overall visual effect, (f) is slightly superior ro (e) and significantly better than (b), (c) and (d), which tells the advantage of our flexible model and quality-

aware cost function. Different from the color casting problem of KITCHEN, the images of CMAPUS present significant luminance variations. The composited panoramas of input images and corrected images are shown in Fig. 15. Comparing the regions of red boxes, we can find the outstanding capability of our method in eliminating the tonal inconsistency. As a constrained version of Shen et al. (2016), Brown and Lowe (2007) can only obtain an inferior result, and Xiong and Pulli (2010) also achieves less satisfying luminance consistency because of the conservative gamma model. As for gradient preservation, Park et al. (2016) gets the best score again in terms of the metric shown in Table 3. It should be noted that although none method removes the tonal difference completely, the residual inconsistency could be easily concealed by blending algorithms in post processing.

In summary, Shen et al. (2016) has a better performance than Park et al. (2016) in terms of color consistency, while Park et al. (2016) is superior to Shen et al. (2016) in other aspects, such as gradient preservation, natural contrast and color fidelity. Because of the essential limitation of their employed simple model, Brown and Lowe (2007) and Xiong and Pulli (2010) are only applicable to situations of moderate color discrepancy. Benefiting from the quality-aware cost function and the flexible model, our approach demonstrates the best comprehensive capability in generating a visually pleasant correction result.

Table 4
Timing of different methods on the dataset list. Unit: second.

Dataset	Brown and Lowe (2007)	Xiong and Pulli (2010)	Shen et al. (2016)	Park et al. (2016)	Proposed method
LAND	21.8	22.3	24.2	1705.1	26.2
COAST	44.5	46.0	48.1	3295.1	46.5
MOUNTAIN	91.1	87.5	99.5	5936.2	96.0
VILLAGE	171.0	213.1	239.6	9531.4	231.2
KITCHEN	3.6	4.3	4.7	102.4	4.4
CAMPUS	2.2	2.4	2.8	90.5	2.5
Total	334.2	375.6	418.9	20660.7	406.8

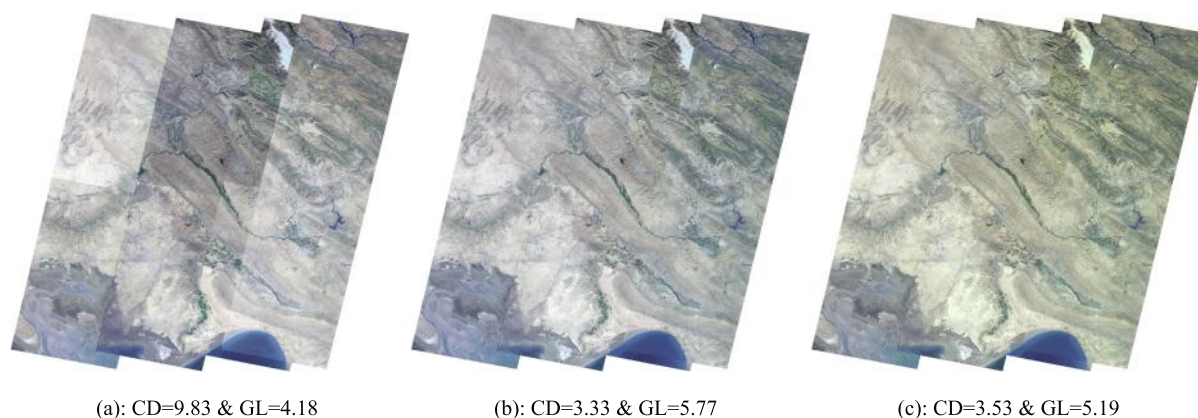(a): CD=9.83 & GL=4.18   (b): CD=3.33 & GL=5.77   (c): CD=3.53 & GL=5.19

Fig. 16. The comparative effect of color constancy used for color consistency. The result processed by color constancy algorithm only (a); The results preprocessed by color constancy algorithm and followed by our proposed method (b); The result processed by our proposed method only (c). For reference, the color distance (CD) and gradient loss (GL) of each result are annotated.

5.3. Efficiency analysis

Methods (Brown and Lowe, 2007; Xiong and Pulli, 2010; Shen et al., 2016) and our proposed method are all implemented in $C++$, while the *MATLAB* code of Park et al. (2016) is provided by the authors. In this paper, all the experiments are conducted on a desktop PC with 8 GB RAM and Intel i7-2600 CPU @ 3.40 GHz. The running time is reported in Table 3. Note that the change detection algorithm is only performed for those image pairs with obvious alterations, such as the top right part of COAST, and the time consumption of this part is not counted here. Despite of using the quadratic spline model, our approach still has an efficiency of the same level with those linear models based methods (Brown and Lowe, 2007; Xiong and Pulli, 2010; Shen et al., 2016). This is because the convex quadratic form of our optimization function and the convex solution space provide a closed-form solution through convex quadratic programming. In our experiments, Brown and Lowe (2007), Xiong and Pulli (2010), Shen et al. (2016) and our method employ the same algorithm to extract color correspondences, which occupy the major computational complexity in practice. Anyhow, the linear model based non-linear optimization function makes Shen et al. (2016) a bit slower than Brown and Lowe (2007) and Xiong and Pulli (2010). As Park et al. (2016) was implemented in *MATLAB*, the running time provided in Table 4 can not be directly used for comparison. Thus, we briefly give an complexity analysis on the algorithm itself. First, the color correspondences are obtained by multi-view feature matching, which is pretty time-consuming. Moreover, the iterative scheme of matrix decomposition are also not as efficient as our closed-form solution. The twofold factors definitely make the computation of Park et al. (2016) much larger than other methods.

5.4. Extended discussion

In the experiments, our proposed method demonstrates a outstanding performance in achieving global color consistency. However, our color correction framework works in the manner of channel by channel, which makes no guarantee to the white balance of the final result. In this section, we attempt to analyze the effect of color constancy algorithms that aims to remove color shift and recover the while balance, which might be used as a baseline or preprocessing module for color consistency problem. Here, the weighted Gray-Edge (Gijsenij et al., 2012), as one of the state-of-the-art algorithms in color constancy, is employed in our experiment. In Fig. 16, we first perform color constancy on the input images individually and get the result shown in (a); Then, we further apply our proposed method on (a) and get the result shown in (b). From the results, we can draw two important conclusions. Firstly, performing color constancy can remove some color

discrepancy that was caused by poor white balance, but fail to handle those inconsistent effects introduced by the variations of observing objects, e.g., the changed radiation characteristics of soil and rivers, and also from the different illuminations; Secondly, using color constancy as the preprocessing module makes little contribution to the global consistency, as compared between (b) and (c) in Fig. 16, though it makes sense in while balance. In summary, color constancy alone is inapplicable to addressing the multi-view color consistency problem, but it helps in improving the white balance of the final result when used as preprocessing.

6. Conclusions

In this paper, we propose a method for optimizing the color consistency across multiple images, and its effectiveness is verified via extensive experiments. An alteration detection algorithm is proposed to guarantee the accuracy of color correspondence. Our key contribution is the novel cost function which is based on the parameterized quadratic spline model, turning visually semantic requirements into parametric expression. Firstly, the flexibility of this spline model shows great ability in color consistency optimization. Secondly, the gradient preservation and dynamic range constraints make great effects in improving image quality. Moreover, the global optimal solution in a closed form is obtained with convex quadratic programming. Extensive experiments on representative datasets verify the validity and generality of our proposed method. Based on the comparison against existing representative algorithms, our method achieves the best performance in terms of both effectiveness and efficiency. However, we conclude that the single-channel optimization strategy still can not solve the color cast (white balance) problem. Therefore, multi-channel joint adjustment and the idea of response function calibration might inspire constructive solutions, which will be investigated in the future works.

Appendix A. Supplementary material

Supplementary data associated with this article can be found, in the online version, at https://doi.org/10.1016/j.isprsjprs.2019.09.004.

References

Alex Krizhevsky, I.S., Hinton, G.E., 2012. Imagenet classification with seep convolutional neural networks. In: Advances in Neural Information Processing Systems (NIPS).

Brown, M., Lowe, D.G., 2007. Automatic panoramic image stitching using invariant features. Int. J. Comput. Vis. (IJCV) 74 (1), 59–73.

Bychkovsky, V., Paris, S., Chan, E., Durand, F., 2011. Learning photographic global tonal adjustment with a database of input/output image pairs. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Cabral, R., la Torre, F.D., Costeira, J.P., Bernardino, A., 2013. Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition. In: IEEE International Conference on Computer Vision (ICCV).

Díaz, M., Sturm, P., 2011. Radiometric calibration using photo collections. In: International Conference on Computational Photography (ICCP).

Frigo, O., Sabater, N., Demoulin, V., Hellier, P., 2014. Optimal transportation for example-guided color transfer. In: Asian Conference on Computer Vision (ACCV).

Frigo, O., Sabater, N., Delon, J., Hellier, P., 2016. Motion driven tonal stabilization. IEEE Trans. Image Process.(TIP) 25 (11), 5455–5468.

Gijsenij, A., Gevers, T., van de Weijer, J., 2012. Improving color constancy by photometric edge weighting. IEEE Trans. Patt. Anal. Mach. Intell. (TPAMI) 34 (5), 918–929.

Goldfarb, D., Idnani, A., 1983. A numerically stable dual method for solving strictly convex quadratic programs. Math. Program. 27 (1), 1–33.

Gooch, B., Ashikhmin, M., Reinhard, E., Shirley, P., 2001. Color transfer between images. IEEE Comput Graph Appl (CGA) 21 (5), 34–41.

HaCohen, Y., Shechtman, E., Goldman, D.B., Lischinski, D., 2011. Non-rigid dense correspondence with applications for image enhancement. ACM Trans. Graph. (TOG) 30 (4), 70.

HaCohen, Y., Shechtman, E., Goldman, D.B., Lischinski, D., 2013. Optimizing color consistency in photo collections. ACM Trans. Graph. (TOG) 32 (4), 38.

He, M., Liao, J., Yuan, L., Sander, P.V., 2017. Neural color transfer between images, arXiv preprint. arXiv: 1710.00756.

Huynh-Thu, Q., Ghanbari, M., 2008. Scope of validity of psnr in image/video quality assessment. Electron. Lett. 44 (13), 800–801.

Hwang, Y., Lee, J.-Y., Kweon, I.S., Kim, S.J., 2014. Color transfer using probabilistic moving least squares. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Ilie, A., Welch, G., 2005. Ensuring color consistency across multiple cameras. In: IEEE International Conference on Computer Vision (ICCV).

Kim, S.J., Pollefeys, M., 2008. Robust radiometric calibration and vignetting correction. IEEE Trans. Pattern Anal. Mach. (TPAMI) 30 (4), 562–576.

Krishna, K., Murty, M.N., 1999. Genetic k-means algorithm. IEEE Trans. Syst. Man Cybernet. 29 (3), 433–439.

Laparra V, Berardino A, Ballé J, Simoncelli EP. Perceptually optimized image rendering, arXiv preprint (2017) arXiv:1701.06641.

Luan, F., Paris, S., Shechtman, E., Bala, K., 2017. Deep photo style transfer. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Nguyen, R.M.H., Kim, S.J., Brown, M.S., 2014. Illuminant aware gamut-based color transfer. Comput Graph Forum (CGF) 33 (7), 319–328.

Pan, J., Wang, M., Li, D., Li, J., 2010. A network-based radiometric equalization approach for digital aerial orthoimages. IEEE Geosci Remote Sens Lett (GRSL) 7 (2), 401–405.

Park, J., Tai, Y.-W., Sinha, S.N., Kweon, I.S., 2016. Efficient and robust color consistency for community photo collections. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Pierre Moulon, B.D., Monasse, P., 2013. Global multiple-view color consistency. In: Conference on Visual Media Production (CVMP).

Pitié, F., Kokaram, A.C., Dahyot, R., 2007. Automated colour grading using colour distribution transfer. Comput Vis Image Understand (CVIU) 107 (1–2), 123–137.

Qian, Y., Liao, D., Zhou, J., 2013. Manifold alignment based color transfer for multiview image stitching. In: IEEE International Conference on Image Processing (ICIP).

Shen, T., Wang, J., Fang, T., Quan, L., 2016. Color correction for image-based modeling in the large. In: Asian Conference on Computer Vision (ACCV).

Shin, Y.-H., Park, M.-G., Jeon, Y.-S., Moon, Y.-S., Lee, S.-H., Yoon, K.-J., 2010. Tone correction with dynamic objects for seamless image mosaic. In: European Conference on Computer Vision (ECCV).

Snavely, N., Seitz, S.M., Szeliski, R., 2006. Photo tourism: exploring photo collections in 3D. ACM Trans Graph (TOG) 25 (3), 835–846.

Su, Z., Deng, D., Yang, X., Luo, X., 2012. Color transfer based on multiscale gradient-aware decomposition and color distribution mapping. In: ACM International Conference on Multimedia (MM).

Su, Z., Zeng, K., Liu, L., Li, B., Luo, X., 2014. Corruptive artifacts suppression for example-based color transfer. IEEE Trans Multimedia (TMM) 16 (4), 988–999.

Tai, Y.-W., Jia, J., Tang, C.-K., 2005. Local color transfer via probabilistic segmentation by Expectation-Maximization. In: IEEE Conf Comput Vis Patt Recognit (CVPR).

Vazquez-Corral, J., Bertalmío, M., 2014. Color stabilization along time and across shots of the same scene, for one or several cameras of unknown specifications. IEEE Trans Image Process (TIP) 23 (10), 4564–4575.

Vazquez-Corral, J., Bertalmío, M., 2016. Log-encoding estimation for color stabilization of cinematic footage. In: IEEE Int Conf Image Process (ICIP).

Wang, B., Yu, Y., Xu, Y.-Q., 2011. Example-based image color and tone style enhancement. ACM Trans Graph (TOG) 30 (4), 64.

Wu, F., Dong, W., Kong, Y., Zhang, X., 2013. Content-based colour transfer. Comput Graph Forum (CGF) 32 (1), 190–203.

Xia, M., Yao, J., Li, L., Xie, R., Liu, Y., 2016. Consistent tonal correction for multi-view remote sensing image mosaicking. In: ISPRS Annals of Photogrammetry, Remote Sensing and Spatial Information Sciences.

Xia, M., Yao, J., Xie, R., Zhang, M., Xiao, J., 2017. Color consistency correction based on remapping optimization for image stitching. In: IEEE International Conference on Computer Vision Workshop (ICCVW).

Xiao, X., Ma, L., 2009. Gradient-preserving color transfer. Comput Graph Forum (CGF) 28 (7), 1879–1886.

Xiong, Y., Pulli, K., 2010. Color matching of image sequences with combined gamma and linear corrections. In: International Conference on ACM Multimedia (MM).

Xu, W., Mulligan, J., 2010. Performance evaluation of color correction approaches for automatic multi-view image and video stitching. IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Zhou Wang, H.R.S., Bovik, Alan Conrad, Simoncelli, E.P., 2004. Image quality assessment: from error visibility to structural similarity. IEEE Trans. Image Process. 13 (4), 600–612.