DOI: 10.1111/cgf.12008

COMPUTER GRAPHICS *forum*
Volume 32 (2013), number 1 pp. 190–203

Content-Based Colour Transfer

Fuzhang Wu[1], Weiming Dong[1], Yan Kong[1], Xing Mei[1], Jean-Claude Paul[2] and Xiaopeng Zhang[1]

[1]LIAMA-NLPR, Institute of Automation, Chinese Academy of Sciences, China
{fuzhangwu, Weiming.Dong, Xiaopeng.Zhang}@ia.ac.cn
[2]INRIA, France

Abstract

This paper presents a novel content-based method for transferring the colour patterns between images. Unlike previous methods that rely on image colour statistics, our method puts an emphasis on high-level scene content analysis. We first automatically extract the foreground subject areas and background scene layout from the scene. The semantic correspondences of the regions between source and target images are established. In the second step, the source image is re-coloured in a novel optimization framework, which incorporates the extracted content information and the spatial distributions of the target colour styles. A new progressive transfer scheme is proposed to integrate the advantages of both global and local transfer algorithms, as well as avoid the over-segmentation artefact in the result. Experiments show that with a better understanding of the scene contents, our method well preserves the spatial layout, the colour distribution and the visual coherence in the transfer process. As an interesting extension, our method can also be used to re-colour video clips with spatially-varied colour effects.

Keywords: color transfer, scene content analysis, progressive transfer

ACM CCS: Computing methodologies—Image processing

1. Introduction

Colour transfer is a practical method to change the appearance of a source image/video according to the colour pattern of a target image. It has various applications in movie post-production, artistic design and photo enhancement. An ideal image colour transfer algorithm should keep the scene of the source image and accurately apply all the dominant colour styles of the target. In many cases, the spatial distribution of the target image should also be presented in the result.

In automatic colour transfer techniques such as [RAGS01, TJT05, PKD07, PR11], they extract various colour statistics from both images and establish specific mapping functions, such that the re-coloured source scene shares similar colour theme with the target. In many cases, the low-level colour statistic information cannot fully describe the real content of the image and some regions may be incorrectly coloured during the mapping process. To avoid such visual artefacts, some interactive methods utilize user-specified colour strokes to ensure correct colour transfer between different image regions, therefore achieve desirable results [LWX07, AP10, LSZ12]. However, it can be difficult for an amateur to specify those strokes in a compatible and perceptually coherent manner. Another problem with previous methods is that they do not consider the spatial distribution characteristics when transferring colours, which might damage the specific visual coherence (Figures 1d and e) or even lose important colour styles of the target image (Figures 2d and e) in the result.

To accurately transfer colours between images with minimal manual labour, we present a system which benefits from the image content analysis. The basic inspirations of our method is that the transfer process should be performed between image regions with similar semantic contents, especially when the obviously different colour patterns appear in different regions of the target image (Figure 1). Moreover, previous methods will easily fail if the number of dominant colour styles of the source image is much less than the one of the target image (Figure 2). To address those problems, in this paper, we present a content-based colour transfer technology. We first perform scene analysis to detect the subject areas (salient objects or clarity areas) and recover the background surface layout (sky, ground, vertical). With this content information, the system transfers colours between the corresponding regions on a super-pixel level. For example, our system will separately transfer the sky colours of the target to the sky in the source (if exists) without affecting the colour patterns of other regions. For such corresponding regions, we present a novel local colour transfer algorithm that integrates spatial distribution of target dominant colours in an optimization framework. We also develop a new progressive transfer scheme to integrate the

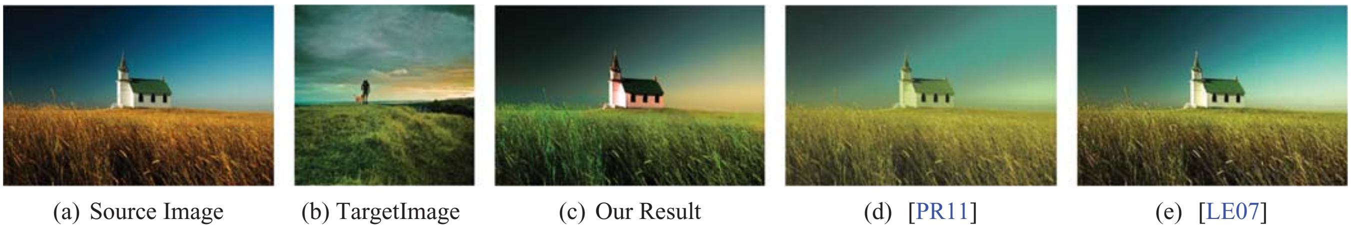

(a) Source Image (b) TargetImage (c) Our Result (d) [PR11] (e) [LE07]

Figure 1: *Our method firstly automatically analyses the scene content structure and divide it into different regions such as salient objects, sky and ground. After that, a novel distribution-aware colour transfer process is performed on the corresponding pairs (e.g. transfer the colour style from target sky to source sky). Our method well preserves the colour spatial distribution and the visual coherence in the result (c). We can see that the colour layer effect of the target sky is lost in (d) and (e).*

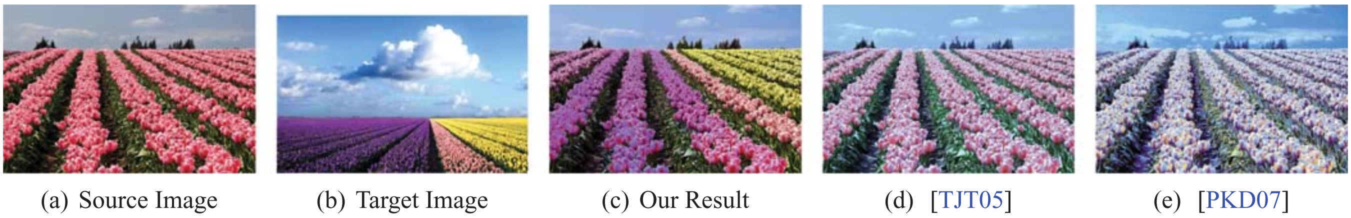

(a) Source Image (b) Target Image (c) Our Result (d) [TJT05] (e) [PKD07]

Figure 2: *Our result accurately presents the spatial distribution of the colourful tulips in the target image.*

advantages of both global and local transfer algorithms. That can also avoid the over-segmentation artefact which often occurs in the results of local colour transfer methods. As a result, all the obvious colour styles, as well as their corresponding spatial distribution in the target image, can be reproduced in the output image without damaging scene contents. To the best of our knowledge, our method is the first trial to challenge this kind of problems in colour transfer research, especially for the example like Figure 2. Moreover, our image colour transfer approach can be efficiently extended to re-colour videos to generate spatially varied colour patterns.

2. Related Work

Automatic colour transfer algorithm was firstly presented by Reinhard *et al.* [RAGS01]. The pixel colour values of the source image are transformed by matching the global colour mean and standard deviation of the target in an uncorrelated colour space $l\alpha\beta$. Chang *et al.* [CSUN05] proposed a perception-based scheme for transferring colours based on the basic colour categories, which are derived through a psychophysical experiments. This framework was extended to stylize the video frame sequences in [CSN07]. Neumann and Neumann [NN05] used 3D histogram matching to transfer the colour style of a source image into an arbitrary given target image having a different distribution. Tai *et al.* [TJT05] modelled the image colour distribution as Gaussian mixture models and solved the regional colour transfer problem by expectation maximization. Pitié *et al.* [PKD07] proposed automated colour grading method by transferring an N-dimensional probability distribution function to another. Dong *et al.* [DBZP10] extracted the same number of dominant colours (DCD) from the two input images and find an optimal one-to-one mapping between the dominant colour sets. Pouli and Reinhard [PR11] presented a histogram reshaping technique which allowed users to select how well the colour palette of the source image should be matched to that of the target. To further optimized the scene details in the result, Xiao and Ma [XM09] employed Poisson compositing [PGB03] to preserve the gradients of the source image. However, as shown in Figure 2, the commonly used colour transfer algorithms fail to generate a satisfied result if the number of source dominant colours is much less than the target one, especially when the colour styles are transferred between images having similar contents.

Interactive methods made colour transfer more controllable in the photo editing process. Luan *et al.* [LWX07] introduced a brush by which the user specifies the source and destination image regions for colour transfer. Wen *et al.* [WHCO08] used strokes in both the source and target image for multiple local transfers. An and Pellacini [AP10] used a non-linear constrained parametric model to transfer colours between image regions defined by pairs of strokes. Liu *et al.* [LSZ12] presented ellipsoid colour mixture map to realize selective colour transfers. Compared with those interactive frameworks, our image content structure extraction technology can help to maximize the automation.

The concept of colour transfer was extended to other applications. Yang *et al.* [YP08] presented a mood-transferring method between colour images by using histogram-matching to preserve spatial coherence. Xue *et al.* [XWT*08] tried to transfer the weathering effects between the input images instead of transferring colour. Murray *et al.* [MSMP11] modified the colour content of an input image given only a concept specified by a user in natural language. Data-driven methods were presented for image colour enhancement by quantifying image-colour theme difference [WYW*10] and non-rigid dense correspondence [HSGL11]. Recently, Chia *et al.* [CZG*11] proposed a semantic colourization system that leverages the image content on the Internet.

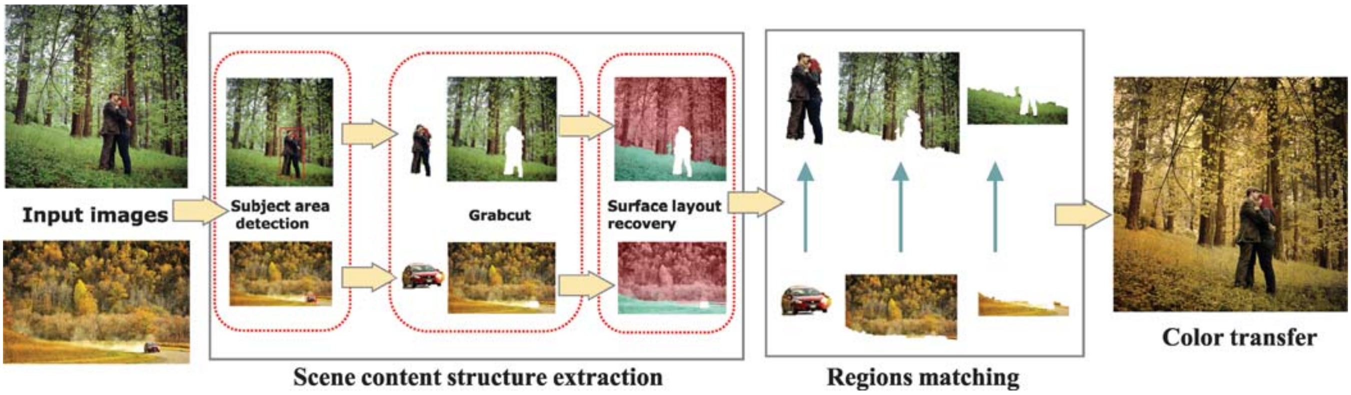


Figure 3: *System pipeline. We separately transfer the colour patterns between the corresponding semantic regions.*

3. Overview

A block diagram of our system framework is shown in Figure 3. To transfer the colour patterns between the source and target image, we first detect the subject area(s) in both images and recover the surface layout of the backgrounds (Section 4). The content information is then used to guide the colour transfer process. A novel distribution-aware colour transfer algorithm is presented to transfer the colour styles between the corresponding regions of the source and target image (Section 5). The spatial distribution and visual details of the target colours can also be preserved in the result. Furthermore, an extension of the method is proposed to transfer spatial colour distribution features to the video clips (Section 6).

4. Scene Content Analysis

Understanding the content structures of the input images will facilitate the colour transfer process to generate a natural looking result. For each input image, we detect the subject area(s) and recover the surface layout of the background.

4.1. Subject area detection

The detection of subject area depends on the image content. In our system, we set a composition-based detection method as the default operator to automatically extract the subject areas from the input images. The user can also choose another two operators to get a more accurate result when the input images contain defocus effects or faces.

4.1.1. *Salient object detection*

For normal images with clear backgrounds, the salient objects are the subject areas. We first use a composition-based method [FWT*11] to detect the windows which contain the salient objects. Based on these windows, we then apply Grabcut method [RKB04] to extract the contour of the objects. Figure 4(a) shows a salient object detection result by our method.

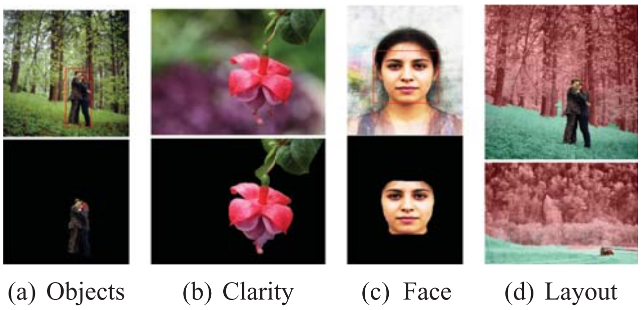

(a) Objects (b) Clarity (c) Face (d) Layout

Figure 4: *Image scene content analysis. In (d), we use different colours to indicate different semantic regions, here green for support (ground) and red for vertical surfaces.*

4.1.2. *Clarity based detection*

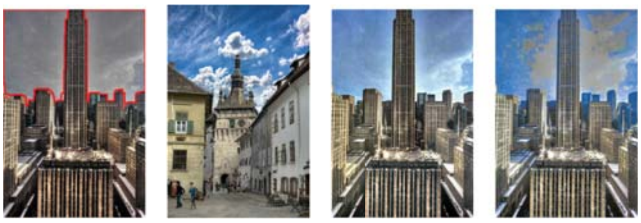
For a low depth of field image or close-up photo with blurred background, we always treat the clear area(s) as the subject content. We provide a clarity-based method to help obtain the clear area(s) in an image. We first use the method in [LT08] to obtain a clarity mask which labels each pixel as clear or blur. Then, the image is over-segmented by applying mean shift segmentation [CM02]. We label a super-pixel as clear if more than half of its pixels are labelled as clear in the clarity mask. An example of our method is shown in Figure 4(b). Disjoint clear areas are treated as different subject areas in our system.

4.1.3. *Face detection*

Faces are special salient objects in an image, especially for portrait photos. We provide a face detector [XZST07] to help locate the windows which contain the faces. Graph-cut [RKB04] is also employed to extract the contours. Figure 4(c) shows a face detection result.

4.2. Background layout recovery

After detecting the subject area(s) in an image, we further segment the background into different semantic regions. We use the method in [HEH07] to recover the surface layout of the background. As



(a) Source (b) Target (c) Our result (d) [PKD07]

Figure 5: *The disjoint sky (the two sub-regions outlined by red boundaries) in (a) is treated as a whole semantic region in the matching and transfer process.*

a result, every pixel in the image is categorized into one of three classes: *support (ground)*, *vertical* and *sky*. Based on this categorization, we get a semantic label mask that indicates the pixels' types (e.g. $label = 1$ means *ground*). Specifically, we treat the pixels with the same label as a whole semantic region in our following transfer method, no matter if these pixels are locate in different disjoint areas or not (Figure 5). This treatment ensures that the disjoint sub-regions with identical semantic label are formed together in the transfer process so as to generate an integrated view in the result, which can avoid the possible discontinuity of the colour patterns. Therefore, the background image is segmented into three parts: *sky region*, *ground region* and *vertical region*, as shown in Figure 4(d) as examples.

In most cases, the segmentation results obtained from both the subject detection and the surface layout recovery are sufficient for our colour transfer framework. If the automatic segmentation is not satisfied, paint selection technique [LSS09] is used to refine the segmentation results.

4.3. Semantic regions matching

After extracting the content structures of the input images, we match the semantic parts together and transfer the colour styles between each pair. For the image backgrounds, we separately match the source sky (vertical, ground) to the corresponding target sky (vertical, ground) if they both exist according to the label masks (e.g. Figure 4d, 'red' vertical is matched to 'red' vertical). The matching is straightforward since the semantic regions in the background are unique.

For the salient objects, we can directly construct the matching if there is only one object in both of the input images. However, it will be more complex if there are multiple salient objects in the images. Denote N_s and N_t as the number of objects extracted from the source and target image separately. If $N_s > N_t$, we first randomly sample N_t objects from the source image and then find a matching target object t_i for each sampled object s_i by optimizing the metric:

$$\min_{\{t_i\}} \sum_{i}^{N_t} \|c_{s_i} - c_{t_i}\|, \quad t_i \neq t_j \quad if \quad s_i \neq s_j, \tag{1}$$

where $c_{s(t)_i}$ is the centroid of the objects in relative coordinates (normalized to [0, 1] according to its position in the image). We use the one-to-one EMD mapping framework in [DBZP10] to optimize the metric. For those unsampled source objects, we set the matching target objects to be the same as their nearest sampled objects in the source image.

For the case $N_s \leq N_t$, we instead sample N_s objects from the target image and reformat Equation (1) as $\min_{\{t_i\}} \sum_i^{N_s} \|c_{s_i} - c_{t_i}\|$ to find the optimal matching target objects for all of the source objects. Since we use a location constraint, the object matching can guarantee our final colour transfer result to preserve the spatial distribution of the target colours in object level. A multiple objects transfer result produced by our method is shown in Figure 10.

Based on these matched semantic regions, we perform our colour transfer operations (Section 5) for each pair separately. Moreover, we consider the content structures of input images are different if there is any semantic region in either of the input images that cannot find a matching region in the other image. To solve this problem, we first globally transfer the luminance of the target image to the source by using the method in [RAGS01]. Based on this result, we then perform our colour transfer operations between the existing pairs and leaving the other regions unchanged, as shown in Figure 16.

5. Distribution-Aware Colour Transfer

Utilizing the content structure information of input images, we separately transfer the colour style of each semantic region in the target image to the corresponding one in the source. We present a novel distribution-aware colour transfer algorithm to preserve the spatial distribution of target colours in the result. For each matched pair, we first segment the regions into super-pixels, where the colour and spatial information are used together to measure the pixels' similarity during the segmentation process. Since the colour spatial distribution is represented in the segmentation results, we consider it as an additional constraint in the colour transfer process to preserve the colour spatial consistency. CIE $L^*a^*b^*$ colour model is used in all our experiments.

5.1. Segmentation

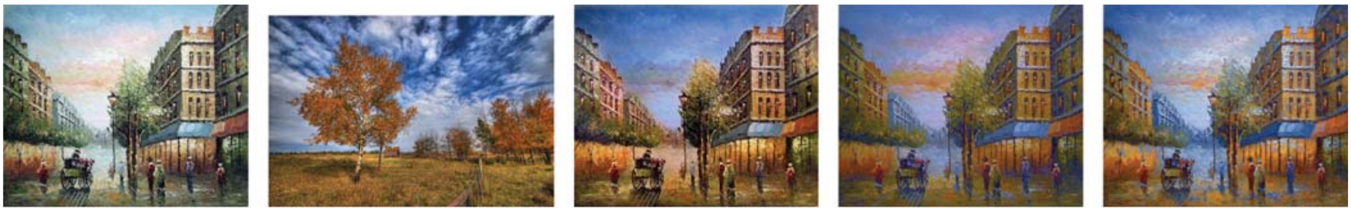
For each region (composed by the pixels with the same semantic label got in Section 4), we use Mean Shift [CM02] to segment it into super-pixels, which will keep the spatial consistency in the segmentation process. The following features are calculated for each super-pixel P_i:

$$P = \{p, \vec{c}, \mu, \sigma, \mathbb{S}\}, \tag{2}$$

where p is the percentage of pixels in the image, $\vec{c}$ is the centroid of the super-pixel, $\boldsymbol{\mu}$ and $\boldsymbol{\sigma}$ are the colour mean and standard deviation value of the pixel colours and $\mathbb{S}$ is a set containing all the pixel colours in a super-pixel. Since the corresponding semantic regions in source and target images may locate in different positions and appear in different geometrical shapes, we calculate their bounding boxes (denote P_{ul} as the upper left position) and align the regions by setting P_{ul} to be the origins of their local coordinate systems. We then set the centroid of each super-pixel as a relative position:

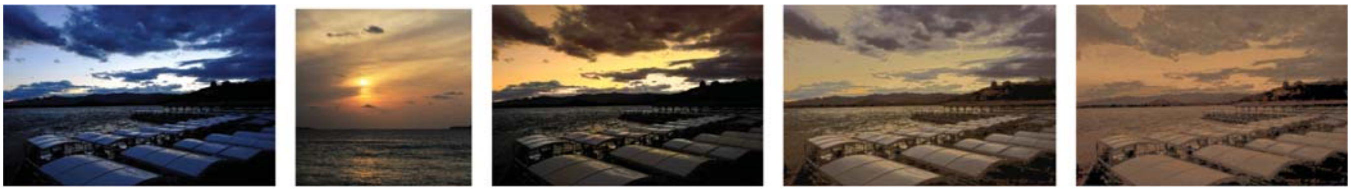
$$\vec{c} = \vec{c}_e - P_{ul}, \tag{3}$$

where $\vec{c}_e$ is the exact centroid of the super-pixel. We normalize $\vec{c}$, μ and σ all to the range [0, 1].



(a) Source Image (b) Target Image (c) Our Result (d) [PR11] (e) [PKD07]

Figure 6: *Our method better preserves the global brightness and local contrast of the original scene. The target colour patterns are also better depicted in the result (no blue colour bleeding on the buildings and street). The reason is that the colour styles in the target sky are only transferred to the source sky and do not affect the other regions (e.g. the roads in (d) and (e)).*



(a) Source Image (b) Target Image (c) Our Result (d) [PR11] (e) [TJT05]

Figure 7: *Our method avoids the foggy artefacts in (d) and (e), while preserving both the global theme (nightfall in a sunny day) and the local colour distribution (the spatial layer of the colours in the sky and the shine on the boat).*

5.2. Soft mapping construction

Based on the segmented super-pixels, we build a connection between the colour patterns of the matched source and target semantic regions. Unlike the one-to-one mapping scheme in [DBZP10], we instead build a soft many-to-many mapping between the two super-pixel sets, which can generate more natural results in most cases. Denote f_{ij} as the probability of the ith source super-pixel maps to the jth target super-pixel, and c_{ij} as the mapping cost between them, the total cost of the mapping is optimized by the metric:

$$\min_{\{f_{ij}\}} \sum_{i=1}^{N_s} \sum_{j=1}^{N_t} f_{ij} c_{ij}, \tag{4}$$

where N_s denotes super-pixel number of the source image and N_t denotes target one. The probability f_{ij} is subject to the following constraints:

$$\sum_{j=1}^{N_t} f_{ij} = p_i^s, \quad i = 1, 2, \ldots, N_s,$$

$$\sum_{i=1}^{N_s} f_{ij} = p_j^t, \quad j = 1, 2, \ldots, N_t,$$

where p_i^s and p_j^t denote the percentage of pixels in the image corresponding to the super-pixel (labels s and t denote the source and target image, respectively). We set a same percentage $p_i = \frac{1}{N}$ to all the super-pixels, which means that they have equal importance in the transfer process. In previous works [FK10, LE07, DBZP10], only colour difference is used to measure the mapping cost. In order to preserve the spatial distribution of the target colour styles, we modify the mapping cost function c_{ij} by adding the spatial location information as a constraint, which is formulated as

$$c_{ij} = \exp\left(\frac{\left\|\mu_i^s - \mu_j^t\right\|^2}{\delta_c}\right) \cdot \exp\left(\frac{\left\|\vec{c}_i^{\,s} - \vec{c}_j^{\,t}\right\|^2}{\delta_s}\right), \tag{5}$$

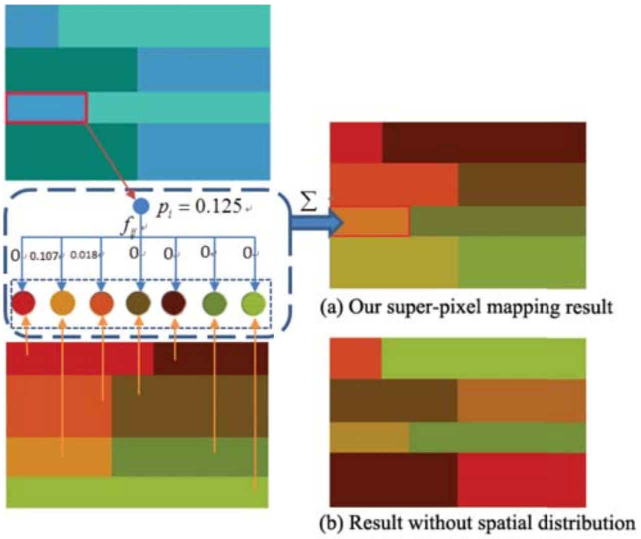
where $\vec{c} = (x, y)$ is the super-pixel centroid (Equation 3), $\mu = (\mu^L, \mu^a, \mu^b)$ is the colour mean value, and δ_c and δ_s are the parameters to control the contributions of colour and spatial features, respectively. We set $\delta_c = 0.95$ and $\delta_s = 0.6$ as default.

We adopt the EMD framework [RTG00] to solve the linear program Equation (4). After that, we formulate the following transform functions to construct the mappings between the features of the source and target super-pixels:

$$\begin{aligned}\Phi(\mu_i^s) &= \frac{\sum_{j=1}^{N_t} f_{ij}\mu_j^t}{\sum_{j=1}^{N_t} f_{ij}},\\ \Phi(\sigma_i^s) &= \frac{\sum_{j=1}^{N_t} f_{ij}\sigma_j^t}{\sum_{j=1}^{N_t} f_{ij}}.\end{aligned} \tag{6}$$

Apparently, the transform functions softly map each source super-pixel to multiple target super-pixels by a weighted average scheme. The weight f_{ij} is solved from Equation (4).

We illustrate our soft mapping scheme between the source and target super-pixels of the corresponding regions in Figure 8. There are eight super-pixels in the source and seven in the target, while the number of colours in the source is less than the one of the target. We can see that after optimizing Equation (4), each source super-pixel is mapped to the target ones by a weighted average formulation. The spatial constraint in Equation (5) helps to preserve the spatial distribution of the target colour styles in the result.



(a) Our super-pixel mapping result

(b) Result without spatial distribution

Figure 8: *Illustration of soft mapping between super-pixels. (b) is generated by ignoring the spatial constraint in Equation (5), so the spatial feature of the target colours is lost.*

We use soft boundaries for the source image to avoid the artefacts caused by segmentation. When transforming the pixel colour $I(x, y)$, we also consider the influence from its neighbouring super-pixels. In the source image, denote P_i as the super-pixel which pixel $I(x, y)$ belongs to the set of neighbouring super-pixels of a pixel $I(x, y)$ is defined as

$$\mathbb{N}(x, y) = \{P_i\} \cup \{P_{i'} | \exists I(x, y) \in \mathbb{S}_i, I(x', y') \in \mathbb{S}_{i'} : |x - x'| + |y - y'| = 1\}. \tag{7}$$

For each neighbouring super-pixel $P_k \in \mathbb{N}(x, y)$, we calculate the probability that the pixel colour $I(x, y)$ belongs to it as:

$$p'_k(x, y) = \frac{1}{Z} D(I(x, y), P_k), \tag{8}$$

where $Z = \sum_{P_k \in \mathbb{N}(x,y)} D(I(x, y), P_k)$ is the normalization factor. $D(I(x, y), P_k)$ is the similarity between the pixel colour $I(x, y)$ and its neighbouring super-pixel P_k. We adopt the bilateral filter to simultaneously smooth the colour and spatial similarity:

$$D(I(x, y), P_k) = \exp\left(-\frac{\left\| I(x, y) - \mu_k^s \right\|^2}{\delta'_c}\right) \cdot \exp\left(-\frac{\left\| (x, y) - \vec{c}_k^{\,s} \right\|^2}{\delta'_s}\right).$$

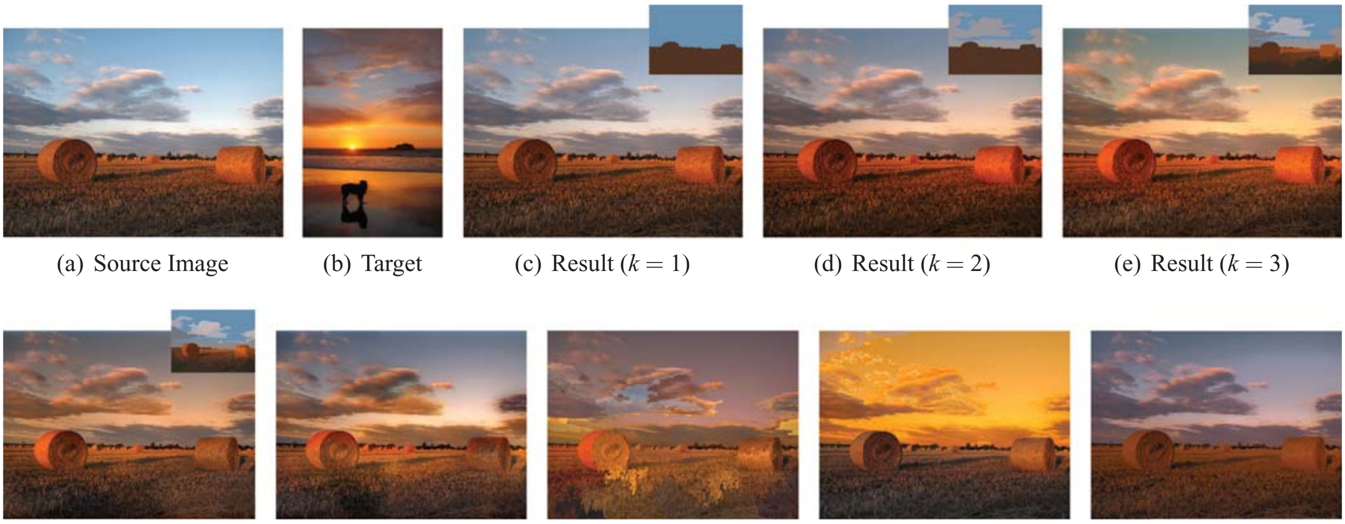
We set $\delta'_s = 0.4$ and $\delta'_c = 0.95$ in all our experiments. Users can properly increase the range of neighbourhood in Equation (7) if the input images are large.

5.3. Pixel colour transform

For an output pixel $I^o(x, y)$, we utilize the probability $p'_k(x, y)$ to smooth the contributions of the neighbouring super-pixels $\mathbb{N}(x, y)$, which is computed as

$$I^o(x, y) = \sum_k p'_k(x, y) \left(\frac{\Phi\left(\sigma_k^s\right)}{\sigma_k^s} (I(x, y) - \mu_k^s) + \Phi\left(\mu_k^s\right) \right), \tag{9}$$

where $p'_k(x, y)$ is the probability that the pixel $I(x, y)$ belongs to the neighbouring super-pixel P_k (Equation 8), $\Phi(\mu_k^s)$ and $\Phi(\sigma_k^s)$ are the new mapped mean and standard deviation of region P_k, respectively (Equation 6). Since the smoothing operation only involves the neighbouring super-pixels, our algorithm is very efficient. In some cases, the local gradient of the source image may be not



(a) Source Image (b) Target (c) Result ($k = 1$) (d) Result ($k = 2$) (e) Result ($k = 3$)

(f) Final Result ($k = 4$) (g) Without progressive (h) Without soft mapping (i) [TJT05] (j) [PR11]

Figure 9: *The iterative results of our progressive colour transfer process. The upper right images in (c)–(f) are the corresponding super-pixels. In (h), we show the result without using soft mapping and soft boundaries, segmentation-like artefacts occur.*

well preserved in the result and causes scene fidelity problem. To address this problem, we use our output as the intermediate image and apply the gradient-preserving method [XM09] to generate the final result, so as to preserve the scene details of the source image.

5.4. Progressive transfer

To be more robust, we build an iterative framework. The target colour pattern is progressively transferred to the source image by modifying the minimum super-pixel size M_k in each step. After the scene content analysis, we first transfer the colour styles between the source and target images at the largest scale by segmenting each content region into one super-pixel. The result is used as the source image in the next step. Then, we progressively reduce the minimum super-pixel size at each iteration by setting $M_k = 0.2 \cdot M_{k-1}$ and update the pixel colour with the following formulations:

$$I_k^{o'}(x, y) = (1 - \alpha_k) \cdot I_{k-1}^{o'}(x, y) + \alpha_k \cdot I_k^o(x, y), \qquad (10)$$

$$\alpha_k = 1.0 - \frac{1.0}{1.0 + \exp(0.5 \cdot (k - 1))},$$

where $I_k^o(x, y)$ is the result produced by Equation (9) with current super-pixel size and α_k is a weight parameter to control the contribution of each scale. In the first iteration ($k = 1$), the source image is used as the initial value of $I_{k-1}^{o'}(x, y)$. We update the resultant image based on this multi-scale framework until the super-pixel is small enough. In our experiment, we stop this iteration when $M_k < 300$. Users can properly increase the size of M_k if the input images are large.

Our progressive approach performs both global and local colour transfer during the iteration, which can preserve the global colour styles as well as the local details in the result. Figures 9(c)– (e) show the intermediate results of our progress transfer process. We can see that the initial results in Figures 9(c) and (d) tend to access the target colour styles globally, while the final fine scale result in Figure 9(f) preserves the local details. Since this progressive framework enables the finer scale transfer to begin with a source image that has the similar global colour styles with the target, it can also avoid undesirable over-segmentation artefacts that often occur when colour transfer is performed only in a fine scale. For example, in Figure 9(g), over-segmentation artefacts occur in the left haystack and lower part of the sky, which is inconsistent with smooth appearance in the source image. Compared with the results in Figures 9(i) and (j), our method better preserves the colour patterns of the target image by presenting the sunset glow effect mainly in the middle part of the result. The sunset glow excessively appears in [TJT05] and is almost lost in [PR11].

6. Extension to Videos

In this section, we extend our framework to videos. An intuitive way is applying the colour styles in the target image to the source video frame by frame. However, this direct operation will cause a perceptible flicker due to the motions in the frame sequences. Moreover, it is more reasonable to preserve the spatial distribution of the target colour styles in the whole video scene rather than in each of the single frame.

In order to keep the colour coherence of the video frame sequences during the transfer process, we first stitch the video frames and restore the scene structure of the video. The SIFT-based stitching algorithm [BL07] is used to create a panoramic image by combining the source video frames. Then, we use our colour transfer framework to apply the target colour styles to the panoramic image. Since there is a projective transformation between each of the video frame and the panoramic image, we can update the frame's colour depending on the transferred panorama as follows:

$$I^o(x, y, t) = \sum_k p'_k(x', y') \left(\frac{\Phi\left(\sigma_k^s\right)}{\sigma_k^s} (I(x, y) - \mu_k^s) + \Phi\left(\mu_k^s\right) \right),$$

$$(x', y', 1)^T = H_t \cdot (x, y, 1)^T, \qquad (11)$$

where $I(, t)$ is the tth image from the source video frame sequence, H_t is the homography matrix representing the projective transformation between the tth image and the panorama, and $(x', y', 1)^T$ is the corresponding homogeneous coordinates in the panoramic image. This function guarantees that the overlapping pixel between two successive frames will present in coherent colour styles since they are projected to the same panoramic super-pixel. The examples of video colour transfer are shown in the supplemental video, we can see that the colour spatial distribution of the target image is separately transferred to different frame sequences to make a coherent visual effect.

7. Results and Discussions

We implement our method on a machine with an Intel Core 7 Quad 2.66 GHz processor and 4 GB RAM. The amount of time of the subject area detection and background surface layout is typically about 20–30 s in total for the images (from 500 × 333 to 500 × 500) used in this work. The layout recovery results are generated by the code subscribed by the authors of [HEH07]. We achieve real-time performance at our distribution-aware colour transfer process after the scene content analysis.

Figure 1 shows a typical example that benefits from our method. The colour features of the salient objects (the man and the dog), the sky and the ground of the target image are separately transferred to the corresponding regions of the source image. This scheme is very helpful in accurately presenting the desired visual appearance in the resultant image.

We compare our method with some commonly used colour transfer algorithms in Figures 2, 6, 7(a), and 12–14. The results of [TJT05] are generated by the code subscribed by the authors of [XM10] (http://ia.cs.colorado.edu/wxu/color_correction.htm). The results of [PKD07], [LE07] , [XM09] and [PR11] are generated from codes which are subscribed by the authors. Results show that our method can both re-produce the local details and global spatial distribution of the target colour pattern, while the other methods fail in most cases. The main reason is that the methods based on GMM [TJT05, LE07] or histogram re-shaping [PKD07, PR11] usually tend to categorize the similar colours in the source image as

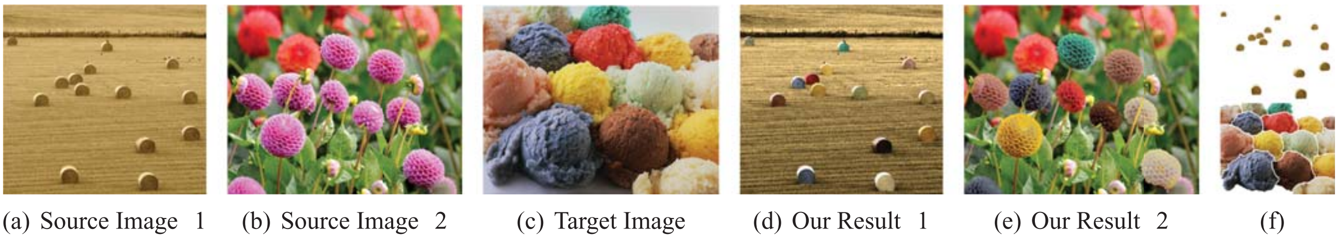

(a) Source Image 1 (b) Source Image 2 (c) Target Image (d) Our Result 1 (e) Our Result 2 (f)

Figure 10: *The colour spatial distribution of the ice cream in (c) is transferred to both the haystacks in (a) and the flowers in (b). We use Paint Selection to extract the haystacks in (a) and separate the ice creams in (c), as shown in (f).*

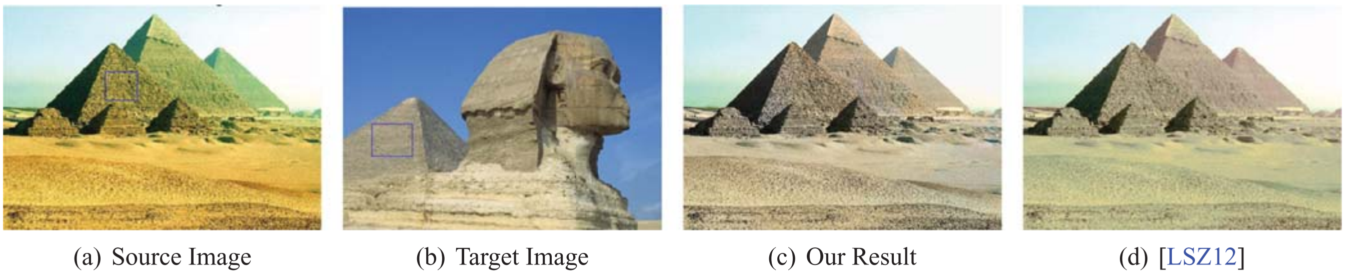

(a) Source Image (b) Target Image (c) Our Result (d) [LSZ12]

Figure 11: *Our method automatically generate more natural result without any user interaction. The sand in our result depicts the colour of the bottom part of the Sphinx in (b), without causing the colour distortion artefact in (d).*

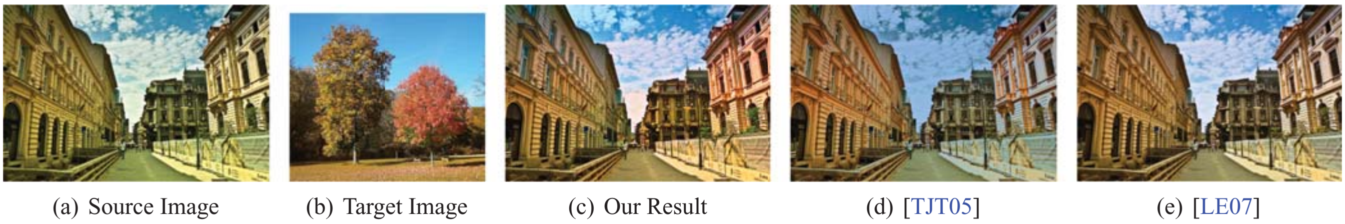

(a) Source Image (b) Target Image (c) Our Result (d) [TJT05] (e) [LE07]

Figure 12: *The two buildings separately present the colour styles of the two trees in (b). The colour spatial information is lost in (d), while the red style is disappeared in (e). Our result also better preserves the brightness and contrast of the scene than (d).*

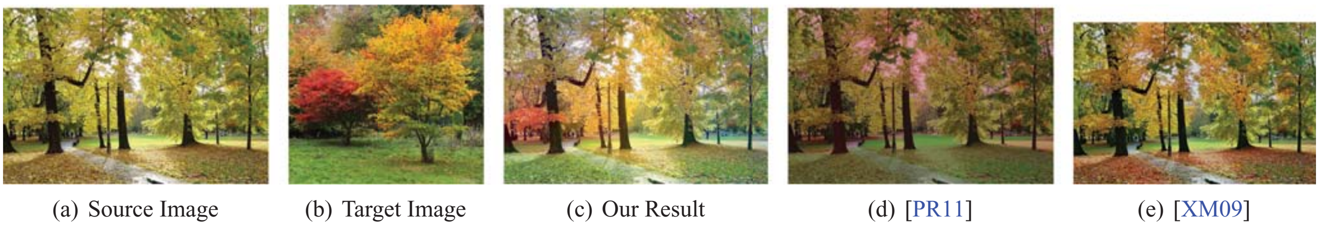

(a) Source Image (b) Target Image (c) Our Result (d) [PR11] (e) [XM09]

Figure 13: *Our result better preserves both the target colour pattern structure and the source scene details.*

a whole and transform them together during the transfer process, without considering their spatial information. Therefore, the similar colours of the source image will also appear to be a similar colour style in the result. However, when the number of colour styles in the source image is much less than the one of the target image, that scheme will cause the result to be assigned the average of the dominant colour styles of the target image, as shown in Figures 1(d), 1(e), 2(d), 2(e), 21(d), 15(d) and 15(e). The integration of spatial constraint in our algorithm can solve the above problem. The similar colours are classified into different categories according to their geometric location, so they can depict different colour styles of the target image in the result. On the other hand, our progressive framework combines the advantages of both the global and local colour transfer schemes and avoids the over-segmentation artefact in the result which often occurs in most local transfer algorithms.

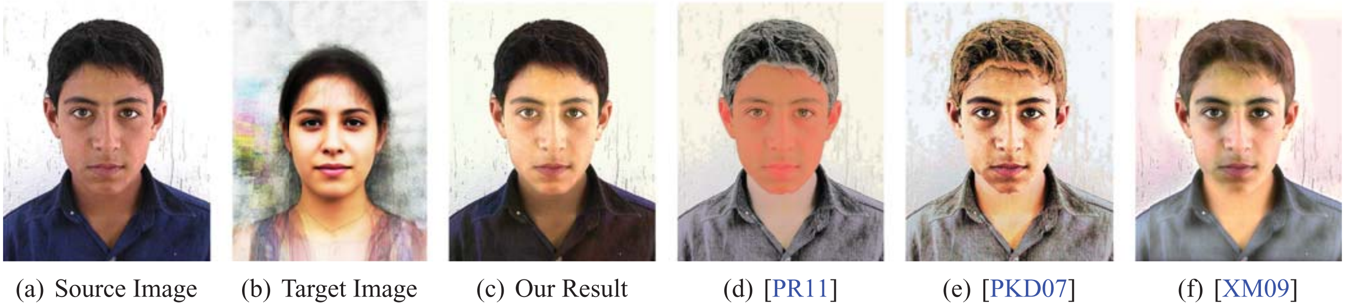

(a) Source Image (b) Target Image (c) Our Result (d) [PR11] (e) [PKD07] (f) [XM09]

Figure 14: *The colour styles of the faces are separately transferred with the help of our face detection operator.*

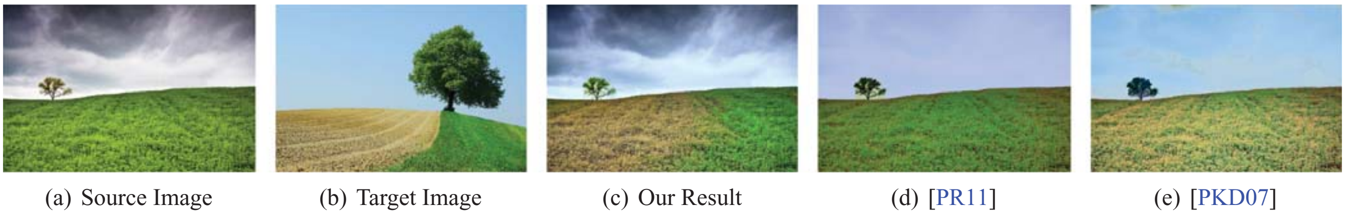

(a) Source Image (b) Target Image (c) Our Result (d) [PR11] (e) [PKD07]

Figure 15: *Comparison of the colour transfer results. We integrate the content information also to the other methods.*

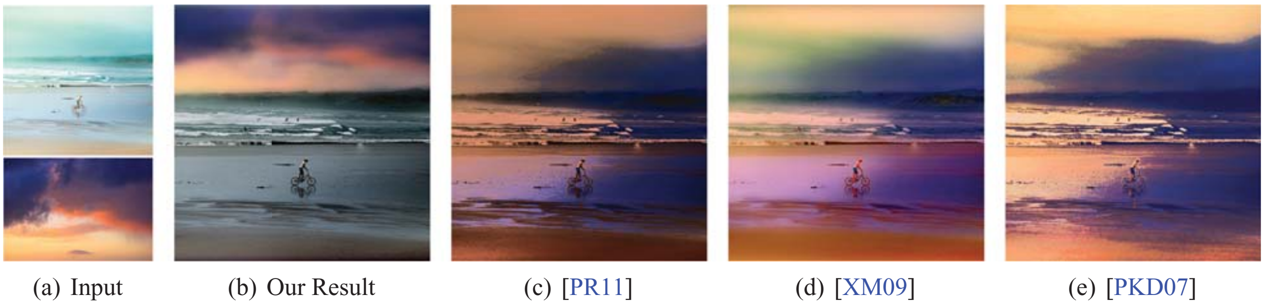

(a) Input (b) Our Result (c) [PR11] (d) [XM09] (e) [PKD07]

Figure 16: *When the input content structures are different, we first globally transfer the luminance and then perform local transfer operation between the skies. The sky in our result is more consistent with the target and the real natural phenomena.*

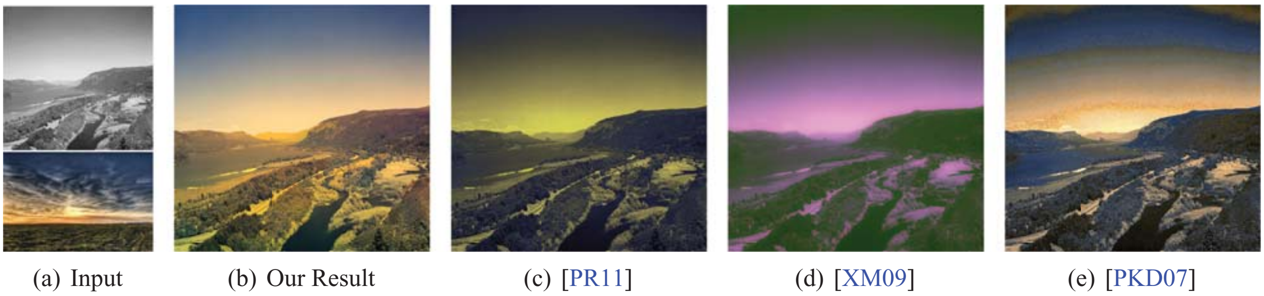

(a) Input (b) Our Result (c) [PR11] (d) [XM09] (e) [PKD07]

Figure 17: *Our method better depicts both the sky (especially the spatial layer of the colours) and the grass (the yellow–green effect caused by the sunlight) colour patterns of (b) than other ones. Our result also avoids the segmentation-like artefact in (e).*

Figure 10 shows two examples containing multiple salient objects, we can see that the spatial distribution of the target object colour styles are well represented in our results. In Figure 11, we compare our method with the manually selective colour transfer method [LSZ12]. Figures 6 and 12 exhibit two more examples which are transferred between two images with obvious different contents (city street and wild field). Results show that our method can nicely preserve integrity as well as spatial distribution of the

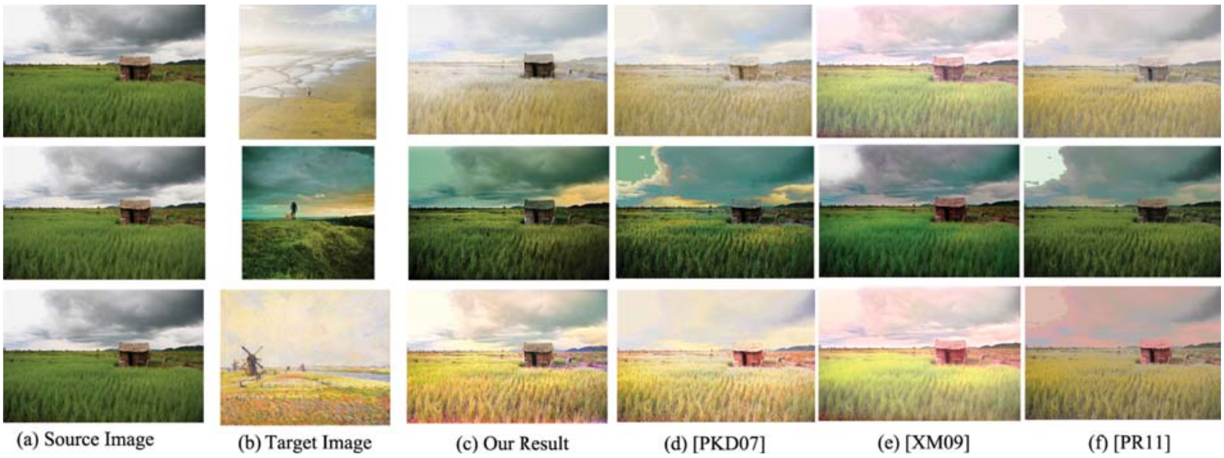


Figure 18: *Results produced by one source image and different target images. Our algorithm is robust in all examples.*

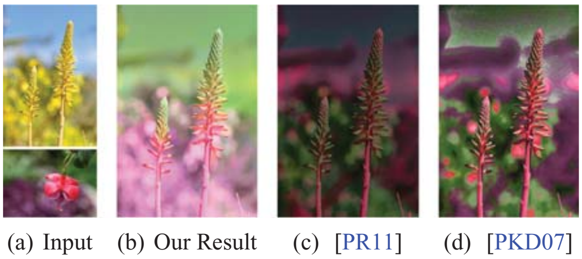


Figure 19: *In our result, colour styles are separately transferred between clear and smooth regions.*

target colour styles, while the compared methods cannot. In Figure 14, we detect the faces from the portraits and separately transfer the colour styles of the face and the background from the target to the source image. Both the colour patterns are well preserved in our result. In Figure 15, we also integrate the content structure information into the other methods. However, with our distribution-aware colour transfer algorithm, our result in Figure 15(c) is still better than the other two images in preserving the details of the source sky and reproducing the colour spatial distribution of the target grass. In Figure 17, we show an example of colourization.

In Figure 18, we use the results generated by using one source image and multiple target images to show the robustness of our method. We can see that our results well preserve the spatial distribution of the target colour styles in all results, while also avoiding the sky colour bleeding in [PKD07]'s results and colour distortion in [XM09] and [PR11]'s results.

In Figure 19, we extract the clear regions from the close-up input images and separately transfer the colour styles between the source and target foreground objects, as well as the blur backgrounds. The top parts of the flowers in the result image appear green styles because they inherit the colour feature of the clear leaves in the target image, which is benefited from our distribution-aware colour transfer scheme.

To further evaluate our method, we perform a user study to compare the results from different methods. One hundred four users from different ages and backgrounds attended the comparison of 17 sets of the colour transfer images. All the stimuli are shown in the supporting information. In the experiment, we show the source image, the target image, our result and the images of the competitors. For each group, the results are randomly displayed in two rows within one page. In order to avoid the interference of the similar quality results to the statistic data, we ask the participants to choose maximum two images they like from each group. Table 1 shows the statistics. Each row shows the percentages of our method and the competitors been chosen by the participants. From the statistics, our method generally outperforms all competitors.

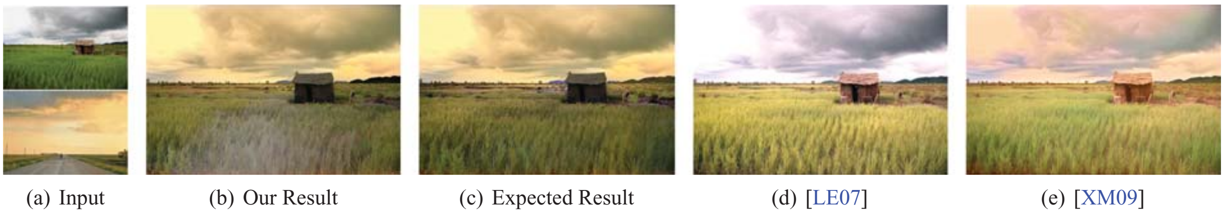
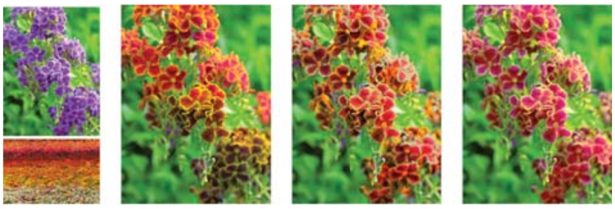


Figure 20: *Our algorithm may generate an unnatural result if the scene content is not precisely extracted. (c) is generated by manually removing the road from the 'ground' region of the target and only transfer the colour styles of the grass to the corresponding 'ground' region (the grass) of the source image.*



(a) Input (b) Our result (c) [PKD07] (d) [XM09]

Figure 21: *The spatial feature of target flowers is preserved. However, different colour styles may appear in a single flower.*

8. Conclusion and Future Work

We propose a novel colour transfer scheme that utilizes subject area detection and surface layout recovery to minimize user effort and generate accurate results. A new local colour transfer algorithm is presented, which integrates spatial distribution of target colour styles in the optimization process. An iterative framework is proposed to increase the robustness of the method, which can ensure the transferring of both the global and local colour styles, while also eliminating the over-segmentation artefacts. Our method can achieve a good and reasonable result even though the number of the source dominant colour styles is much less than the target one. The spatial distribution of the target dominant colour styles are also nicely preserved in the output image. We also extend our algorithm to handle video-image colour transfer. The special colour distribution of the target image can be reproduced in an panoramic-like video.

There are limitations of our content-based colour transfer framework that we plan to investigate in future work. One is that the subject area detection methods may be inaccurate with fine-scale structure and unapparent boundary. For example, in Figures 10(a) and (c), we need to use Paint Selection to help to extract some small haystacks and ice creams. The surface layout recovery method is not accurate enough for some examples which contain complex scene structures. As shown in Figure 20, the road and the grass in the target image are both categorized as ‘ground’, so the colour styles of the road are also transferred to the grass of the source image. However, the image semantic segmentation is still an open problem and we believe that our system will be more robust by using some more precise image understanding algorithms. Our distribution-aware colour transfer scheme may break the wholeness of the clustered objects. As shown in Figure 21(b), sometimes, different colour styles appear in one single flower. Therefore, from the user study, we can see that many participants also choose Figure 21(d) as a favourite result. Some of the target colours may be lost in the result if the content of the source image is ‘less’ than the target, especially when the salient objects in the source image are less than the ones of the target (e.g. Figure 10e). Another is that colour transfer for background regions will not always exactly reproduce the spatial distribution of the target colour styles in the result, especially when the corresponding region of the source image contains an apparently different spatial colour distribution (an example is shown in the supporting information). However, in most cases, those results are actually more semantic reasonable than simply duplicating the colour distribution from the target image. Moreover, videos with large dynamic objects are generally less accurate than for static scenes, since the motion of dynamic objects will affect the accuracy of panoramic stitching. Additionally, we also plan to extend our algorithm to incorporate with internet photos which can subscribe more information to the input images.

Table 1: *User study.*

	Ours	[PR11]	[TJT05]	[LE07]	[PKD07]	[XM09]
Figure 1	65.3%	15.3%	5.77%	17.3%	23%	25%
Figure 2	57.6%	1.92%	17.3%	11.5%	15.3%	13.4%
Figure 6	61.5%	17.3%	5.77%	0%	11.5%	21.2%
Figure 7	59.6%	1.92%	3.8%	23%	34.6%	5.77%
Figure 9	50%	1.92%	9.6%	19.2%	28.5%	11.5%
Figure 12	55.7%	0%	3.85%	25%	15.3%	21.1%
Figure 13	75%	1.92%	1.92%	9.62%	17.3%	19.2%
Figure 14	67.3%	0%	1.92%	32.6%	5.77%	23%
Figure 15	50%	21.1%	5.77%	13.4%	15.3%	15.3%
Figure 16	40.3%	11.5%	3.85%	28.8%	9.6%	26.9%
Figure 17	80.7%	13.4%	0%	3.85%	17.3%	1.92%
Figure 18-1	53.8%	3.8%	1.92%	40.3%	15.3%	1.92%
Figure 18-2	61.5%	5.77%	13.4%	3.85%	25%	19.2%
Figure 18-3	57.6%	3.85%	0%	23%	25%	9.6%
Figure 19	71.6%	1.92%	0%	11.5%	15.3%	19.6%
Figure 20	46.1%	3.85%	7.69%	25%	15.3%	36.5%
Figure 21	48%	0%	11.5%	13.4%	48%	23%

Acknowledgements

We thank anonymous reviewers for their valuable input. We thank Tania Pouli for helping to make some results. We thank some flickr members (listed in the supporting information) who kindly allow us to use their images under Creative Commons License. This work is supported by National Natural Science Foundation of China under No. 61172104, 61271430, 61201402 and 61202324, by Beijing Natural Science Foundation (Content-Aware Image Synthesis and Its Applications, No. 4112061), and by SRF for ROCS, SEM.

References

[AP10] An X., Pellacini F.: User-controllable color transfer. *Computer Graphics Forum 29*, 2 (2010), 263–271.

[BL07] Brown M., Lowe D. G.: Automatic panoramic image stitching using invariant features. *International Journal of Computer Vision 74*, 1 (August 2007), 59–73.

[CM02] Comaniciu D., Meer P.: Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence 24*, 5 (May 2002), 603–619.

[CSN07] Chang Y., Saito S., Nakajima M.: Example-based color transformation of image and video using basic color categories. *IEEE Transactions on Image Processing 16*, 2 (2007), 329–336.

[CSUN05] CHANG Y., SAITO S., UCHIKAWA K., NAKAJIMA M.: Example-based color stylization of images. *ACM Transactions on Applied Perception 2*, 3 (2005), 322–345.

[CZG*11] CHIA A. Y.-S., ZHUO S., GUPTA R. K., TAI Y.-W., CHO S.-Y., TAN P., LIN S.: Semantic colorization with internet images. *ACM Transactions on Graphics 30*, 6 (2011), 156:1–156:8.

[DBZP10] DONG W., BAO G., ZHANG X., PAUL J.-C.: Fast local color transfer via dominant colors mapping. In *Proceedings of the ACM SIGGRAPH ASIA 2010 Sketches* (New York, NY, USA, 2010), ACM, pp. 46:1–46:2.

[FK10] FREEDMAN D., KISILEV P.: Computing color transforms with applications to image editing. *Journal of Mathematical Imaging and Vision 37*, 3 (July 2010), 220–231.

[FWT*11] FENG J., WEI Y., TAO L., ZHANG C., SUN J.: Salient object detection by composition. In *ICCV 2011: Proceedings of the IEEE International Conference on Computer Vision* (November 2011), pp. 1028–1035.

[HEH07] HOIEM D., EFROS A. A., HEBERT M.: Recovering surface layout from an image. *International Journal of Computer Vision 75*, 1 (October 2007), 151–172.

[HSGL11] HACOHEN Y., SHECHTMAN E., GOLDMAN D. B., LISCHINSKI D.: Non-rigid dense correspondence with applications for image enhancement. *ACM Transactions on Graphics 30*, 4 (August 2011), 70:1–70:10.

[LE07] LALONDE J.-F., EFROS A.: Using color compatibility for assessing image realism. In *Proceedings of the Computer Vision 2007 (ICCV 2007)* (October 2007), pp. 1–8.

[LSS09] LIU J., SUN J., SHUM H.-Y.: Paint selection. *ACM Transactions on Graphics 28*, 3 (2009), 69:1–69:7.

[LSZ12] LIU S., SUN H., ZHANG X.: Selective color transferring via ellipsoid color mixture map. *Journal of Visual Communication and Image Representation 23* (January 2012), 173–181.

[LT08] LUO Y., TANG X.: Photo and video quality evaluation: Focusing on the subject. In *ECCV '08: Proceedings of the 10th European Conference on Computer Vision: Part III* (Berlin, Heidelberg, 2008), Springer-Verlag, pp. 386–399.

[LWX07] LUAN Q., WEN F., XU Y.-Q.: Color transfer brush. In *Proceedings of the 15th Pacific Conference on Computer Graphics and Applications* (Washington, DC, USA, 2007), IEEE Computer Society, pp. 465–468.

[MSMP11] MURRAY N., SKAFF S., MARCHESOTTI L., PERRONNIN F.: Towards automatic concept transfer. In *NPAR'11: Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Non-Photorealistic Animation and Rendering* (New York, NY, USA, 2011), ACM, pp. 167–176.

[NN05] NEUMANN L., NEUMANN A.: Color style transfer techniques using hue, lightness and saturation histogram matching. In *Proceedings of Computational Aesthetics in Graphics, Visualization and Imaging* (2005), pp. 111–122.

[PGB03] PÉREZ P., GANGNET M., BLAKE A.: Poisson image editing. *ACM Transactions on Graphics 22* (July 2003), 313–318.

[PKD07] PITIÉ F., KOKARAM A. C., DAHYOT R.: Automated colour grading using colour distribution transfer. *Computer Vision and Image Understanding 107*, 1–2 (2007), 123–137.

[PR11] POULI T., REINHARD E.: Progressive color transfer for images of arbitrary dynamic range. *Computers & Graphics 35*, 1 (2011), 67–80.

[RAGS01] REINHARD E., ASHIKHMIN M., GOOCH B., SHIRLEY P.: Color transfer between images. *IEEE Computer Graphics and Applications 21*, 5 (2001), 34–41.

[RKB04] ROTHER C., KOLMOGOROV V., BLAKE A.: "grabcut": Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics 23* (August 2004), 309–314.

[RTG00] RUBNER Y., TOMASI C., GUIBAS L. J.: The earth mover's distance as a metric for image retrieval. *International Journal of Computer Vision 40*, 2 (2000), 99–121.

[TJT05] TAI Y.-W., JIA J., TANG C.-K.: Local color transfer via probabilistic segmentation by expectation-maximization. In *Proceedings of CVPR'05 - Volume 1* (Washington, DC, USA, 2005), IEEE Computer Society, pp. 747–754.

[WHCO08] WEN C.-L., HSIEH C.-H., CHEN B.-Y., OUHYOUNG M.: Example-based multiple local color transfer by strokes. *Computer Graphics Forum 27*, 7 (2008), 1765–1772.

[WYW*10] WANG B., YU Y., WONG T.-T., CHEN C., XU Y.-Q.: Data-driven image color theme enhancement. *ACM Transactions on Graphics 29*, 6 (2010), 146:1–146:10.

[XM09] XIAO X., MA L.: Gradient-preserving color transfer. *Computer Graphics Forum 28*, 7 (2009), 34–41.

[XM10] XU W., MULLIGAN J.: Performance evaluation of color correction approaches for automatic multi-view image and video stitching. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2010), pp. 263–270.

[XWT*08] XUE S., WANG J., TONG X., DAI Q., GUO B.: Image-based material weathering. *Computer Graphics Forum 27*, 2 (2008), 617–626.

[XZST07] XIAO R., ZHU H., SUN H., TANG X.: Dynamic cascades for face detection. In *ICCV 2007: Proceedings of the IEEE 11th International Conference on Computer Vision* (October 2007), pp. 1–8.

[YP08] YANG C.-K., PENG L.-K.: Automatic mood-transferring between color images. *IEEE Computer Graphics and Applications 28* (March 2008), 52–61.

Supporting Information

Additional Supporting Information may be found in the online version of this article at the publisher's web site:

Video S1: Transferring the colour styles of the target image to the source video.

Figure S1: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 1 in our paper.

Figure S2: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 2 in our paper.

Figure S3: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 6 in our paper.

Figure S4: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 7 in our paper.

Figure S5: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 9 in our paper

Figure S6: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 12 in our paper

Figure S7: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 13 in our paper.

Figure S8: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 14 in our paper.

Figure S9: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 15 in our paper.

Figure S10: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 16 in our paper.

Figure S11: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 17 in our paper.

Figure S12: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 18(top) in our paper.

Figure S13: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 18 (middle) in our paper.

Figure S14: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 18 (bottom) in our paper.

Figure S15: (a) Input images. (b) Our result. (c) Result produced by [LE07]. (d) Result produced by [PKD07]. (e) Result produced by [XM09]. (f) Result produced by [TJT05]. (g) Result produced by [PR11]. Corresponding to Figure 19 in our paper.

Figure S16: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11]. Corresponding to Figure 20 in our paper.

Figure S17: (a) Source image. (b) Our result. (c) Result produced by [LE07]. (d) Result produced by [PKD07]. (e) Result produced by [XM09]. (f) Result produced by [PR11]. Corresponding to Figure 21 in our paper.

Figure S18: We flip the original target image horizontally to make a spatial distribution conflict between the source and target image. Our result still preserves the spatial distribution of the target colours. (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11].

Figure S19: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11].

Figure S20: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11].

Figure S21: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11].

Figure S22: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (e) Result produced by [PKD07]. (f) Result produced by [XM09]. (g) Result produced by [TJT05]. (h) Result produced by [PR11].

Figure S23: (a) Source image. (b) Target image. (c) Our result. (d) Result produced by [LE07]. (d) Result produced by [TJT05]. (f) Result produced by [PR11].