Woodblock image decomposition of Chinese new year paintings

Article *in* Multimedia Tools and Applications · March 2019

DOI: 10.1007/s11042-018-6447-x

5 authors, including:

Wei Feng
Tianjin University
115 PUBLICATIONS **1,427** CITATIONS
SEE PROFILE

Jiawan Zhang
Tianjin University
141 PUBLICATIONS **884** CITATIONS
SEE PROFILE

Multimedia Tools and Applications
https://doi.org/10.1007/s11042-018-6447-x

Woodblock image decomposition of Chinese new year paintings

Liang Wan[1] · Ye Liu[1] · Haipeng Dai[2] · Wei Feng[2] · Jiawan Zhang[1]

Received: 29 September 2017 / Revised: 15 May 2018 / Accepted: 20 July 2018 /
Published online: 14 August 2018

Abstract

Woodblock printed Chinese new year (WNY) painting has been a popular art form in Chinese folk culture. To make a WNY painting involves carving images on woodblocks and printing colors using woodblocks. Although thousands of WNY paintings were preserved, the ten-year national survey reveals that a great number of woodblocks were damaged or lost. In this paper, we study a novel problem of decomposing woodblock images from WNY paintings, which currently requires a tremendous amount of manual labor. We also find that the state-of-the-art methods of natural image segmentation generate poor results in our application. Instead of using sophisticated schemes, we develop a simple yet robust decomposition approach, which contains the extraction of line block image and the separation of color block images. The effectiveness of the proposed approach is validated through both quantitative evaluation and visual quality comparison with six state-of-the-art methods on multiple WNY paintings.

Keywords Woodblock printed Chinese new year paintings · Woodblock image decomposition · Line block · Color block

1 Introduction

The Chinese new year painting is a unique art form in Chinese folk culture, having a long history since Tang dynasty (618 A.D.) [8, 26]. As a prevalent house decoration for celebrating the Chinese New Year, new year paintings are characterized by auspicious and

✉ Wei Feng
wfeng@tju.edu.cn

Liang Wan
lwan@tju.edu.cn

[1] School of Computer Software, Tianjin University, Tianjin, China

[2] School of Computer Science and Technology, Tianjin University, Tianjin, China

joyous subjects covering door gods, folk tales, lucky mascots of birds and flowers, etc. New year paintings have different types according to their generation techniques. The most famous one is woodblock printed new year painting (WNY painting for short). Its generation involves carving an image on woodblocks and printing colors on papers using woodblocks. Usually a WNY painting uses one line block depicting the *sketch* of the painting, and several color blocks each depicting different *non-overlapped regions* in one color. As engraving woodblocks is a sophisticated skill, the paintings' woodblocks have been regarded as artistic works owning precious values in culture research and heritage preservation. Researchers could analyze the properites of workblocks, for example the amount and apperance, to help identify the origins of WNY paintings.

The ten-year national survey of WNY paintings [8], starting from 2001, reveals the endangered situations faced by WNY paintings. Although thousands of WNY paintings were found and archived in form of digital pictures, most of them had their woodblocks (partially) lost or seriously damaged. Reconstructing woodblocks requires a tremendous amount of manual labors, in which one big challenge is fast and accurate differentiation of color regions and line sketches from WNY paintings that suffer from color fading, bleeding, and other artifacts.

In this paper, we study the problem of extracting woodblock images from digital pictures of WNY paintings (Fig. 1). As far as we know, our research is the first attempt for addressing this problem. In the literature, there exists color decomposition techniques for Japanese woodblock printing art, Ukiyo-e [16, 18, 22]. However, Japanese Ukiyo-e paintings have different apperance characteristics from WNY paintings, and most techniques were focused on developing color-mixing models. Our goal is to decompose a WNY painting into one line block image and multiple color block images (line/color block for short). Although our task seems to be simple, directly applying clustering or segmentation methods [4, 7, 9, 10, 15], which are successful for natural images, can yield rather poor results. Instead of using sophisticated schemes as in many general-purpose segmentation methods, we design a simple yet effective decomposition approach for WNY paintings, by employing a considerable amount of foreknowledge about WNY paintings.

We start our work by summarizing properties of WNY paintings and their generation procedure. The decomposition framework separates the decomposition of line blocks from the decomposition of color blocks, and iterates the two tasks in an alternative manner. For line block decomposition, we train a binary SVM classifier to extract primary portions of line block, which we called localized SVM classifier. The WNY painting image with the primary line information removed is fed into color block decomposition, for which we apply an efficient color clustering-based method that is able to determine the number of color blocks and get clean color blocks. In addition, we extend the FDoG algorithm [12] to upgrade the primary line block, yielding a more visually complete line block. Experiments on multiple WNY paintings show plausible decomposition results. The effectiveness of our approach is validated in both visual and quantitative comparison with six state-of-the-art methods for color clustering and image segmentation.

2 Related work

Our work is focused on analyzing woodblock new year paintings, which have not been studied before. In this section, we briefly review the existing research work from the aspects of related applications and techniques.

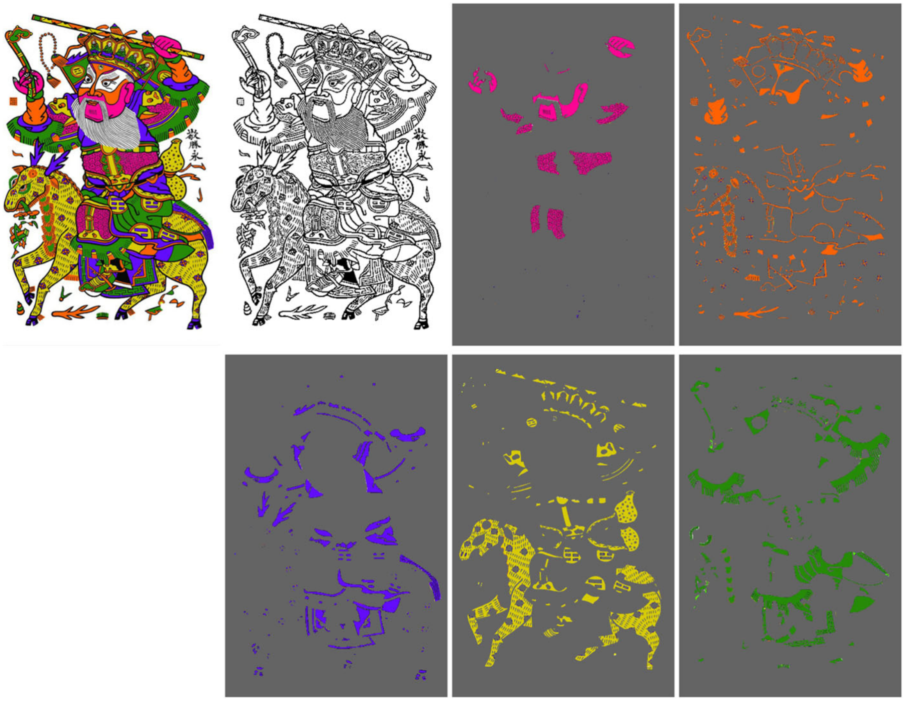


Fig. 1 Woodblock image decomposition of one Chinese new year painting. From left to right are the input painting, the generated line block and five color blocks

2.1 Color decomposition of Japanese Ukiyo-e

In the viewpoint of application, our work is mostly related to color decomposition for Japanese Ukiyo-e [16–18, 22], which is also a woodblock printing art. Unlike WNY paintings, Japanese Ukiyo-e paintings contain severe color blending, while the colors are usually not staturated, and they seldom have dark outlines. Previous work on Ukiyo-e decomposition [22, 23] was focused on color-mixing analysis according to some physical paint blending models, while the analysis has been conducted on small painting portions with 2 color paints only. Hence, these techniques are not applicable to our application.

2.2 Digital reconstruction of cartoons

Our application is also related to the segmentation of cartoons. Cartoons are generally composed of regions with simple coloring and wide decorative lines [20, 21, 28]. For example, Sỳkora et al. [20, 21] segmented black and white cartoons. Their methods require cartoon regions to be enclosed by clearly strong outlines, which is not satisified for WNY paintings. Zhang et al. [28] segmented color cartoons. They first extracted decorative lines with small finite width, then detected edges using Canny detector, and finally employed a trapped-ball segmentation. Their approach may fail to detect weak edges due to Canny detector, and generates oversegmentation results. On the contrast, we need to handle weak edges (due to the degradation of outlines) and avoid oversegmentation (to estimate the correct number of color blocks) in the case of WNY paintings.

2.3 Color clustering and natural image segmentation

In the viewpoint of technique, our decomposition is highly related to color clustering and image segmentation methods.

Color clustering techniques group pixels with similar colors in color spaces such as RGB, Lab or HSV. Typical clustering algorithms include *k*-means clustering [10, 15], spectral clustering [25], tensor voting [6], and etc. As we know, *k*-means clustering [10] is able to yield good results, however it is sensitive to the selection of initial centers. Mignotte [15] performed k-means clustering in six color spaces, and integrated the results via one more k-means clustering to get more plausible partitions. However, these methods assume that the cluster number is already known, while for WNY painting decomposition, we have to estimate the number of clusters.

Image segmentation clusters pixels by considering color similarity as well as spatial coherence. As a popular method, meanshift clustering [3, 4, 27] can determine the number of clusters automatically, however it is sensitive to local peaks, leading to an over-segmentation result. Hierarchical image segmentation shows good performance for natural images, which can obtain segmentation results at different complexity levels. For example, Feng et al. [9] modelled the image segmentation problem as optimizing an Markov random fields with an unknown number of labels. Arbelaez et al. [1] developed a global optimization framework based on spectral clustering to detect contours, and transformed contours into a hierarchical segmentation tree. Donoser and Schmalstieg [7] utilized a discrete-continuous optimization to get results competitive to [1] while reducing computation time and memory demands.

These general purpose segmentation methods work well for natural images, yet produce poor results for our application. One possible reason is that they are more sensitive to color variations in WNY paintings, which occur due to the unflattened surface of woodblocks.

3 Materials & methods

3.1 Characteristics of WNY paintings

WNY paintings from different origins share a similar production process. After designing a painting, artists engrave the painting's outlines on a woodblock as a relief pattern, forming the line block (Fig. 2a). The portions to be dyed in a specific color are carved on individual woodblocks, forming color blocks. The number of color blocks is usually less than a dozen. Once the woodblocks are engraved, a WNY painting is produced by first printing the line block (dyed with black paint) on a paper, and then overprinting the color blocks (dyed with different color paints) usually in a light-to-dark order. By observing various WNY paintings, we summarize four distinctive characteristics as follows:

1. Almost all outlines are printed in black, and have a certain width not being just one-pixel wide. Besides outlines, there may exist big-sized black-colored objects, such as eyes or hairs engraved in the line block (Figs. 2b and 3), and these objects can have different sizes.
2. A WNY painting uses a limited number of strong and contrasting color paints.
3. There exist color bleedings and less blackish lines (Fig. 2b). These are mainly induced by the misalignment of woodblocks in the carving and overprinting process.

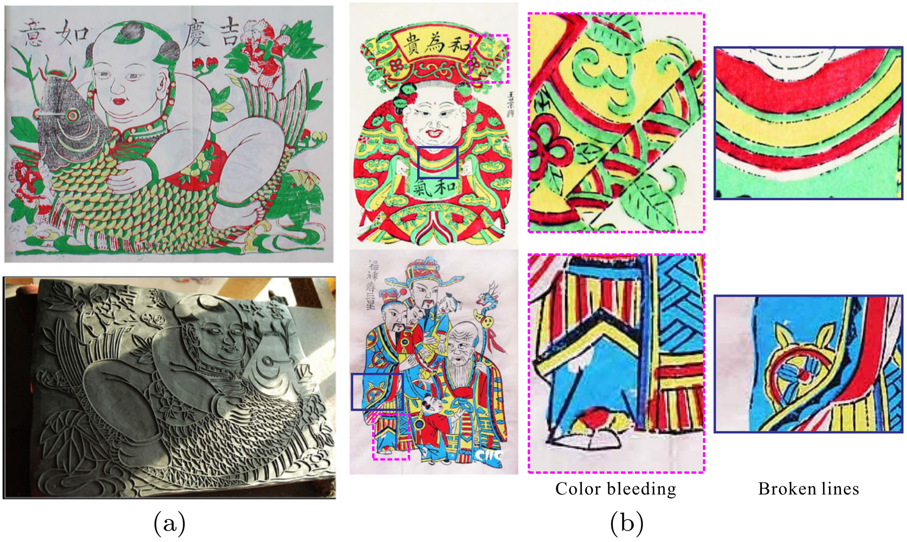


Fig. 2 WNY painting and color artifacts: **a** one new year painting and its line block; **b** color bleeding around lines or region boundaries, and broken lines

4. The paintings may contain various artifacts (Fig. 2b), including broken black lines, and non-uniform colors which are made by the unflattened surface of woodblocks.

3.2 Problem formulation

Given a WNY painting I, our goal is decomposing it into a line block L and a set of color blocks $\{C_i | i = 1, \ldots, M\}$, where M is the number of color paints. The decomposition can be formulated as:

$$I = L \cup C_1 \cdots \cup C_M, \tag{1}$$

where the operator $\cup$ computes the set union. In this formulation, the paper's color that is usually white or yellowish is taken into account. Note that line block has characteristics different from color blocks, as the line regions usually occupy much smaller portions and suffer from color bleeding and broken pieces. Accordingly, we develop a framework as illustrated in Fig. 3, in which the decomposition of line block and the decomposition of color blocks are treated in different ways.

Our framework starts by first extracting an initial line block L_p using a SVM classifier, which is trained using a WNY painting dataset (referred as the global SVM classifier). We then apply a color clustering-based method on the residue image that has L_p removed from I. Our method is able to determine the number of color paints M automatically. Next, we extract a refined line block using another SVM classifier, which is trained by adding samples from the extracted color blocks to the original training data (referred as the localized SVM classifier). The above process iterates 2–3 times to get the refined primary line block, denoted as L'_p, and refined color blocks. Afterwards, we apply a guided FDoG filter

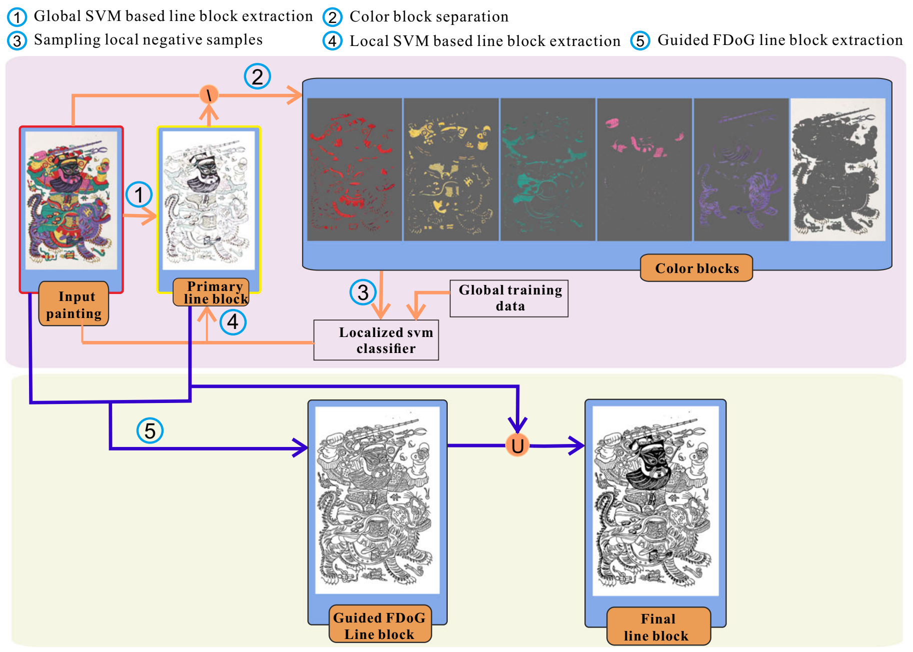


Fig. 3 Our framework. See text for a detailed explanation

starting from L'_p, to produce more visually complete line block L_g. The final line block is the combination of L'_p and L_g.

In the following, we present three major components in the framework, i.e. primary line block extraction, color block separation, and line block enhancement.

3.3 Primary line block extraction

The line block extraction can be regarded as a binary classification problem. The naive way to apply a threshold on the whole painting usually fails to detect weak edges, since lines and colors may not perfectly match after image carving, which may result in faded line colors. In this paper, we utilize binary SVM classifier [5, 19], which is proven to be simple yet effective for binary classification, to detect blackish pixels.

We prepare a WNY painting dataset containing 25 pictures, for each of which the color blocks are manually extracted using Photoshop. We train a global binary SVM classifier by automatically selecting negative samples (20,623 in total), i.e. color block pixels, and manually choosing positive samples (7,400 in total), i.e. line block pixels, from 10 WNY paintings. The SVM kernel function is set as a Gaussian kernel. Figure 4b shows an example of L_p extracted by the global SVM calssifier. We can see the global SVM classifier is helpful to extract weak lines which are polluted by color paints. However, the global SVM may wrongly classify dark colored pixels to the line block, like purple pixels in Fig. 4d. To alleviate this problem, we train a local SVM classifier by adding negative samples, i.e. pixels from extracted color blocks, to the global training data. By repeating the whole process 2–3 times, we obtain a reliable line block L'_p. As demonstrated in Fig. 4e, many wrongly classified purple pixels are removed.

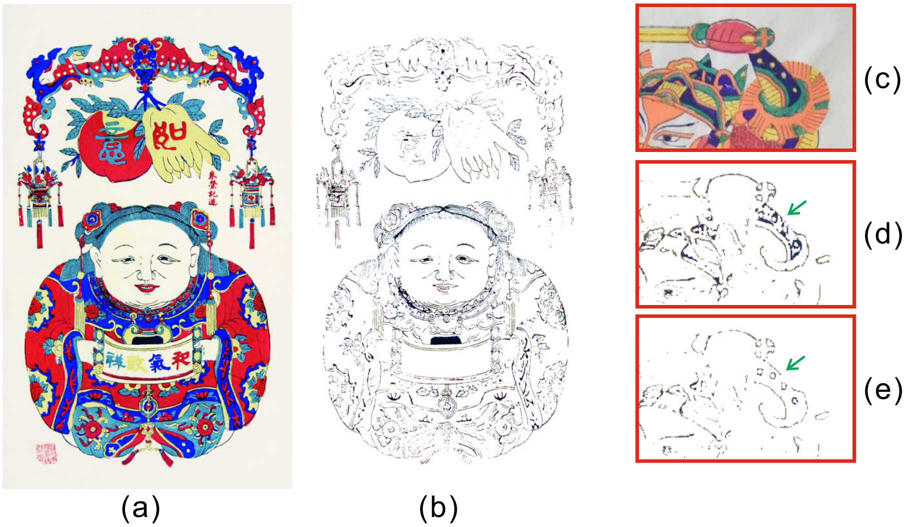


Fig. 4 Primary line block extraction: **a** input WNY painting; **b** primary line block; **c** a part of another painting; **d** primary line block of (**c**); and **e** refined primary line block of (**c**)

3.4 Color block separation

After the primary line block is determined, we construct a residue image by subtracting the primary line block from the input painting image, denoted as

$$I' = I \setminus L'_p, \tag{2}$$

where the operator \ represents set difference. We develop a three-step algorithm for color block separation: 1) estimate dominant colors from the residue image I'; 2) cluster the pixels in I' to color blocks; 3) refine clustering results based on local color distributions. Since color separation is performed in an iterative process, without introducing ambiguity, we report the results in the last iteration.

Dominant color estimation Since colors in WNY paintings often have large contrast, we estimate dominant colors using a scheme similar to local extrema detection in SIFT [14]. Specifically, we use color histogram to count the number of pixels with similar colors, and the histogram peaks are regarded as dominant colors. In practice, we test HSV, RGB, and Lab color spaces, and find that Lab performs best. We also find that the L channel has little influence, since most colors are saturated.

Let a color pixel be $I'_i = [l_i, a_i, b_i]^T$. In the normalized 2D histogram for a, b channels, bin B_{st} is taken as a dominant peak if it is a local maximum and larger than a small threshold ξ. ξ is empirically chosen to avoid detecting trivial peaks. The corresponding dominant color is given by

$$p^0_j = [\bar{l}_j, a_s, b_t]^T, \tag{3}$$

where $\bar{l}_j$ is the average L-channel value of pixels belonging to B_{st}, and a_s, b_t are the mean color values of the corresponding quantized ranges. Figure 5 shows two examples and their dominant colors.

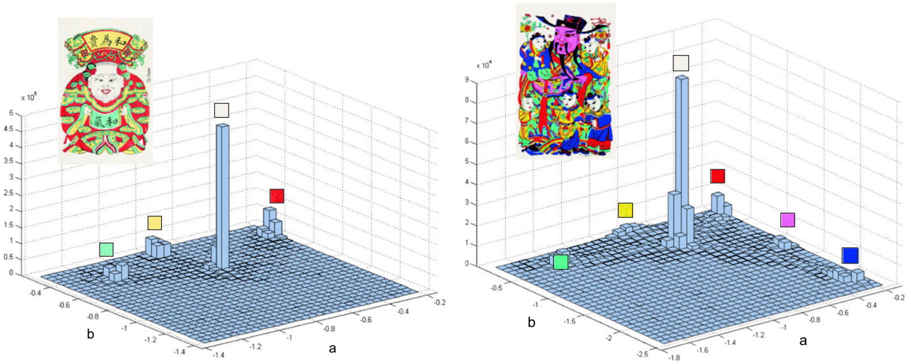


Fig. 5 Dominant color estimation

Mixed-space color clustering The dominant color estimation generates the number of clusters and reliable initial cluster centers $\{p_j^0\}$. Next we use KMeans clustering to yield a stable partition of the pixels in I'. For each pixel I'_i, let its color label be

$$T_c(i) = \arg\min_j D(p_j, I'_i), \tag{4}$$

where p_j is the j-th cluster center, and $D(\cdot)$ computes the color distance between p_j and I'_i.

We find that using $[l_i, a_i, b_i]^T$ in color distance yields plausible clustering results, however some pixels that differ greatly in RGB color space may be wrongly clustered (Fig. 6b). Like [15] that fuses distances in multiple color spaces, we represent a pixel using the concatenation of its Lab values and RGB values, i.e. $I'_i = [l_i, a_i, b_i, r_i, g_i, b_i]^T$. The distance metric is defined as a linear combination of the norm-2 distances measured in Lab and RGB color spaces respectively, given by

$$D(p_j, I'_i) = \lambda \| p_j - I'_i \|^{lab} + (1 - \lambda) \| p_j - I'_i \|^{rgb}. \tag{5}$$

We empirically set $\lambda = 0.8$, with more contributions from Lab color space. As demonstrated in Fig. 6, the combination of RGB and Lab color distance is better than the pure Lab distance.

Color block amendment After a closer inspection of the clustered results (Fig. 7), we notice there may exist scattered pixels around color region boundaries. This phenomenon largely results from color bleeding between two adjacent color regions. To alleviate this problem, we apply an amendment to locally adjust the cluster labels of pixels. Specifically, we assign one pixel the label of the color cluster which has the largest number of neighoring pixels in a local window. Figure 7 shows the amendment results for Fig. 4a. If two colors in the local color histogram have almost equal distributions, we choose one color randomly. Since we do not remove any pixel but change the pixel's cluster label, the amendment avoids extra holes which may result from using morphological operators.

3.5 Line block enhancement

After the iteration stops, we get the refined primary line block L'_p, like Fig. 8b. We observe that L'_p contains big-sized black regions, but may have broken lines and may not catch

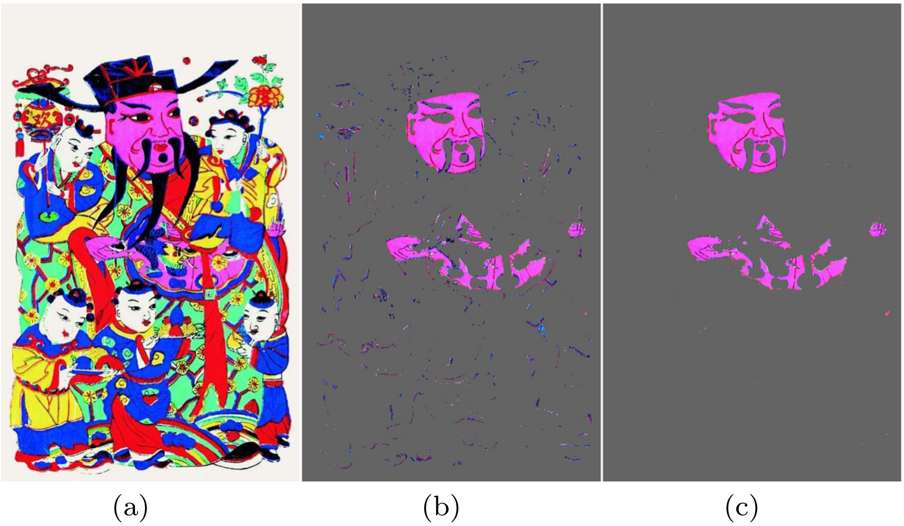


Fig. 6 **a** Input WNY painting; **b** One color block using pure Lab distance ($\lambda = 1.0$); and **c** The color block using the combination of Lab and RGB distance ($\lambda = 0.8$)

important polluted lines. In the literature, there are some algorithms effective to extract coherent and smooth lines. A successful one is FDoG filter [12], which is an edge flow-based DoG filter. Our experiments show it is helpful to detect polluted lines in WNY paintings (Fig. 8f). However, directly applying the FDoG filter on the input image may suffer from obvious artifacts. As clearly shown in the rightmost figures in Fig. 8h and i, it is sensitive to uneven paints and produces lines at boundaries of two color segments. In our work, we extend the FDoG filter and use the primary line block L'_p as a guidance. The middle figures in Fig. 8h and i demonstrate the benefit of our approach.

As illustrated in Fig. 9, the original FDoG filter response at pixel x, $H(x)$, is a weighted 1-D DoG response $F(s)$ along the edge flow direction from $-S$ to S, given by

$$H(x) = \int_{-S}^{S} G_{\sigma_m}(s)F(s)ds. \tag{6}$$

$F(s)$ is computed in the direction along $-T$ to T which is perpendicular to the edge flow direction, given by

$$F(s) = \int_{-T}^{T} I(l_s(t))f(t)dt, \quad f(t) = G_{\sigma_c(t)} - \rho * G_{\sigma_s(t)}. \tag{7}$$

G_σ is a 1D Gaussian function with variance σ. Thresholding $H(x)$ can yield lines which are coherent and have small gaps filled. Note that (6) and (7) are continuous functions. In the implementation, we resample the image grid to get samples for the computation. For more details about FDoG, the interested readers are referred to [12].

Our guided FDoG starts from the refined primary line block L'_p, and detects new line pixels in neighboring regions of existing lines. In the first filtering pass, we compute FDoG

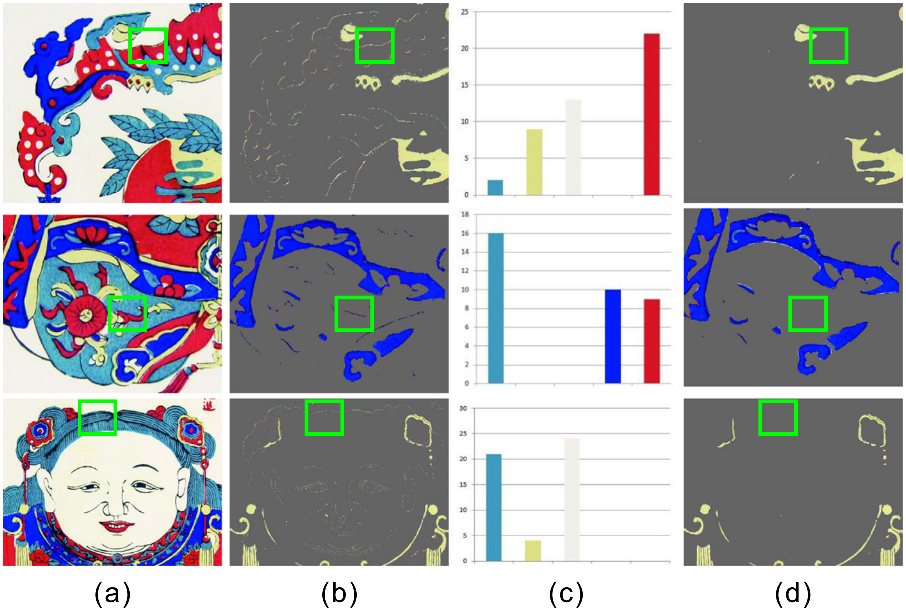


Fig. 7 Color block amendment: **a** a portion of the WNY painting in Fig. 4a; **b** the clustering results; **c** the local color histogram of the square-marked neighborhood; and **d** the amended results for (**b**)

responses only at pixels which have at least one neighboring pixel in L'_p, and for other pixels, we set their FDoG responses to zero. The guided FDoG is defined as

$$H(x) = \begin{cases} \int_{-S}^{S} G_{\sigma_m}(s)F(s)ds, & \text{if } \mathcal{N}_S(x) \cap L'_p \neq \emptyset, \\ 0, & \text{otherwise.} \end{cases} \qquad (8)$$

The symbol $\mathcal{N}_S(x)$ denotes the neighbors of pixel x along the edge flow direction, which is computed from the input WNY painting I. Thresholding $H(x)$ can grow coherent lines only around L'_p, and get the guided line block L_g. We then perform the guided FDoG on L_g iteratively until L_g does not change.

Figure 8 shows examples for L'_p and guided line blocks at iteration 1, 10 and 16. The regions in Fig. 8g demonstrate the gradual completion of some important lines. From blow ups in Fig. 8g and h, we can see that most noisy line segments from the original FDoG filering are suppressed and the boundaries around the characters are removed in the final guided line block L_g. On the other hand, L'_p contains large black regions which are missing in L_g. Therefore, our final line block L is defined as the union of L'_p and L_g, given by

$$L = L'_P \cup L_g. \qquad (9)$$

4 Experiments and discussion

4.1 Dataset and results

In our experiments, we build a dataset including 25 WNY paintings, covering different themes, color paints, and production locations. The painting images have middle sizes, with

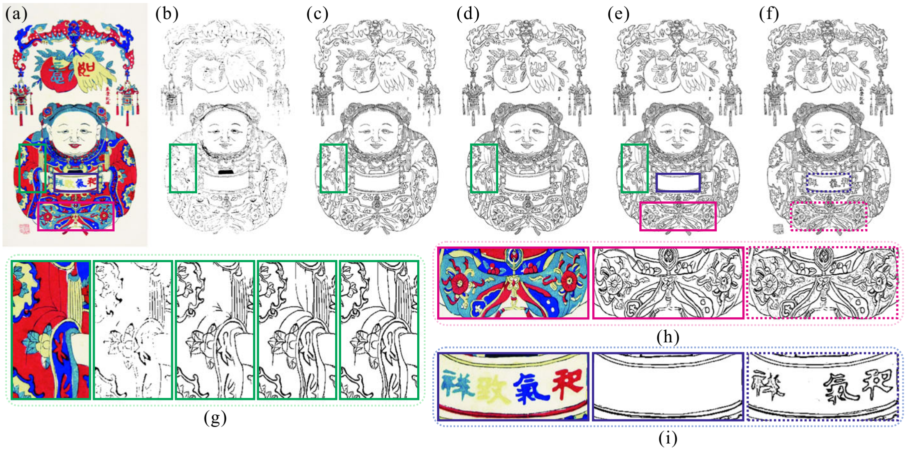


Fig. 8 Guided FDoG: **a** input WNY painting I; **b** the refined primary line block L'_p; **c**, **d**, **e** are the intermediate results at iteration 1, 10, 16 of iterative guided FDoG; **f** FDoG result of (**a**); **g** blow-up of the green box regions in (**a**, **b**, **c**, **d**, **e**); **h** blow up of the red box regions in (**a**, **e**, **f**); and **i** blow up of the blue box regions in (**a**, **e**, **f**)

a pixel number ranging from 0.4 M to 1.2 M. Figure 10 shows some decomposition results. WNY paintings are typically composed of 4 to 6 woodblocks, with varying color paints. Here the blocks corresponding to the paper are omitted. We can obviously see that our method produces high quality line blocks and color blocks.

We now report the timing performance of our method and manual decomposition. We use the basic SVM training/deduction library [2], and implement the proposed method on a computer installed with Intel(R) Core (TM) i5-3330 CPU @ 3.00 GHz and 8.0G memory. The decomposition timing depends on the image size, the number of blocks, and the picture's complexity. For example, using our non-optimized implementation to decompose Fig. 13 (image size 600 × 800, 4 color blocks) costs about 13.5 min. The framework (Fig. 3) iterates for three times. The most costly operations are training localized SVM classifier (6 min in total) and prediction using SVM (6.4 min in total). In comparison, it takes more than 40 min to manually decompose this WNY picture.

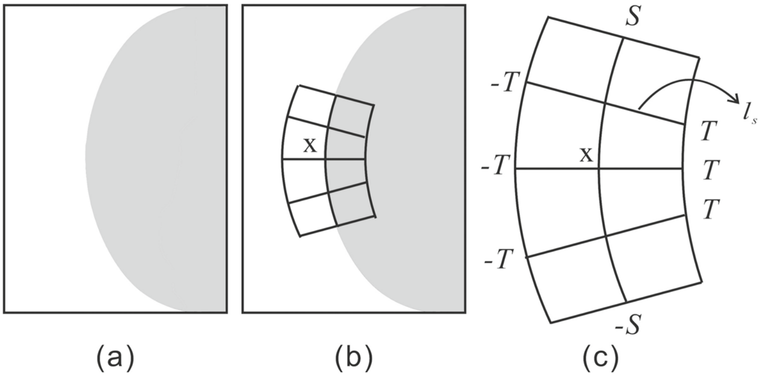


Fig. 9 FDoG filtering: **a** input; **b** kernel at pixel x; **c** kernel enlarged

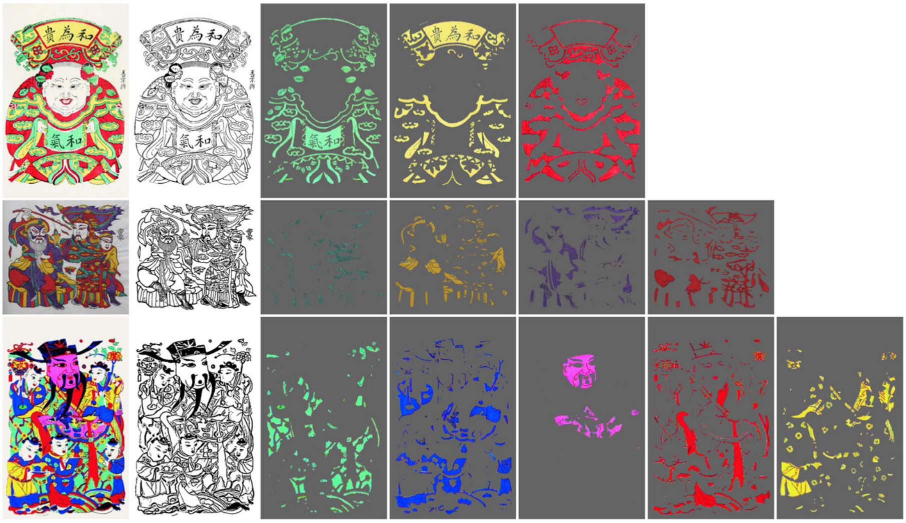


Fig. 10 Results for woodblock image decomposition of Chinese new year paintings. The first column is the input paintings, the second column is the line block and the rest columns are the color blocks

4.2 Comparison with state-of-the-art methods

We make comparison with six state-of-the-art methods for color clustering and image segmentation. They are

1. standard KMeans clustering (KMeans);
2. fused KMeans clustering (FRC) [15];
3. meanshift clustering (MS) [4];
4. net-structured graph cuts (NSGC) [9];
5. hierarchical segmentation using ultra contour map (UCM) [1];
6. hierarchical segmentation using a discrete-continuous optimization of oriented gradient signals (DC-Seg) [7].

We divide the above six methods into two categories. KMeans and FRC need users to specify the cluster number, denoted as *K-clustering methods*, while the rest four methods, i.e. MS, NSGC, UCM and DC-Seg, can automatically determine the cluster number, denoted as *self-validated methods*.

As described in Section 3.3, we manually decompose the reference color blocks for each WNY painting. Note that it is pretty difficult to make reference line block, because the thin lines are often polluted and usually occupy small portions in WNY paintings, however the real lines should have at least a limited width in woodblocks. In our experiments, we take the final line block L as the reference line block, but ignore it in quantitative evaluations.

4.2.1 Quantitative evaluation criteria

Since the decomposition of color blocks is indeed a segmentation problem, we evaluate the segmentation accuracy of decomposition results.

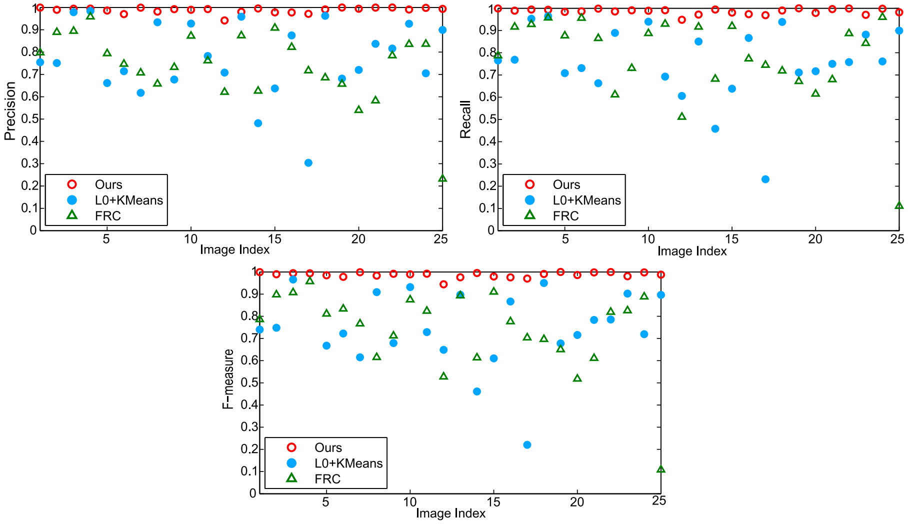


Fig. 11 The average precision, recall and F1 scores of our method, L0+KMeans, and FRC

The commonly-used metrics for segmentation accuracy are *precision*, *recall* and *F-1 measure* scores. Note that they require the decomposition results have the same number as the reference. Hence, these metrics are only applicable to *K-clustering* methods (i.e. KMeans and FRC) and our method, since they can generate the same number of woodblock images as the reference.

To also evaluate the performance of *self-validated methods* (i.e. MS, NSGC, UCM and DC-Seg), which have a varying cluster number, we define two new metrics. The first metric is *recomposition error*. We define it as the MSE difference between the mean image recomposed from the decomposition results (denoted as R) and that from the reference color blocks (denoted as M), which is given by

$$RM\_MSE = \frac{1}{N}\sum_i (R_i - M_i)^2, \tag{10}$$

where N is the number of valid pixels, and R_i (M_i) is the mean color of the decomposed (reference) color block which the i-th pixel belongs to. The larger the recomposition error is, the more amount of pixels are wrongly clustered.

Secondly, we define the *labelling consistency ratio* metric, following the idea that two pixels from one reference color block are expected to fall in the same cluster for a given segmentation method. Supposing X pixels ($X = 10,000$) are sampled from one reference color block, we get $\frac{1}{2}X(X-1)$ pixel pairs. If there are Y pixel pairs, in each of which two pixels have the same cluster label, the labeling consistency ratio (LCR) for this color block is defined as:

$$LCR = \frac{2Y}{X(X-1)}. \tag{11}$$

According to the definition, a good segmentation method has the LCR value approach to 1.

4.3 Quantitative comparison

To ensure *K-clustering* methods (i.e. KMeans and FRC [15]) generate the same number of cluster as the reference blocks, we set the cluster number as $M + 1$, where M is the true number of color blocks, with an extra cluster representing the line block. In the experiments, KMeans clustering is performed on RGB space and cluster centres are randomly initialized from image pixels. For *self-validated* methods (i.e. (MS) [4], NSGC [9], UCM [1], DC-Seg[7]), we use the authors' codes and adopt the default parameters.

Recall that the precision, recall and F1 measure metrics compare the decomposition results and their corresponding reference color block. It is a must to need to build the correspondence. Here, we compare the mean color value of color blocks, and for each decomposed color block we choose the reference one with the minimum difference as its correspondence. Then for each WNY painting with M clusters, we can get M scores. The average scores of each WNY painting in the dataset are plotted in Fig. 11. As it is shown, KMeans and FRC have rather low scores for most of WNY paintings. It is because the two methods are sensitive to the initialization and may cluster different color pixels into one block due to their distance measures.

Figure 12 plots the results of the recomposition error for all the comparing methods. It is noted that since L contains some colored pixels due to the FDog filtering, we take L as an mask and ignore all its pixels in the recomposition error calculation. In this figure, the line segments connecting the measures are drawn to help differentiate the results from various methods, because we find that it is hard to read the figure if only measures are plotted. In comparison, UCM and NSGC have rather high MSE values since they wrongly cluster many pixels. Our method has the lowest MSE values for all the WNY paintings.

Figure 13 plots the average LCR scores for the WNY painting dataset. It is obvious that the comparing methods gather into three groups. One group is made of UCM, MS and DC-Seg, all of which have rather low LCR scores. It is because these three methods generate multiple segments more than the man-made reference. Our method achieves the highest LCR scores for all the WNY paintings, and approaches 1 for many paintings. The rest of other methods report LCR scores ranging between 0.6 and 0.9. In summary, our decomposition results are highly consistent with the man-made reference.

In Figs. 12 and 13, we also consider the possible impact of image smoothing. Since WNY painting pictures suffer from color variances, it is natural to consider applying

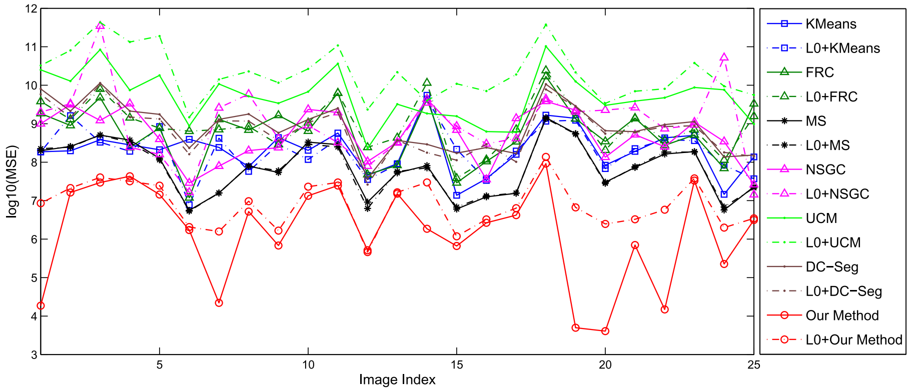


Fig. 12 The recomposition MSE errors of the WNY painting dataset for all the methods with and without L_0 image smoothing

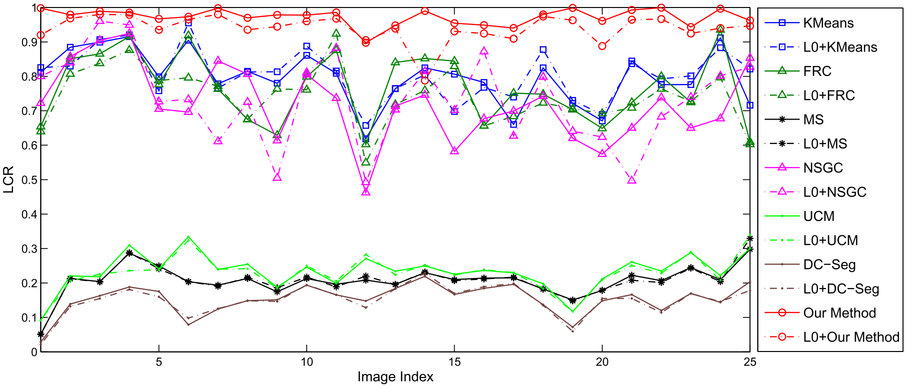


Fig. 13 The LCR scores of the WNY painting dataset for all the methods with and without L_0 image smoothing

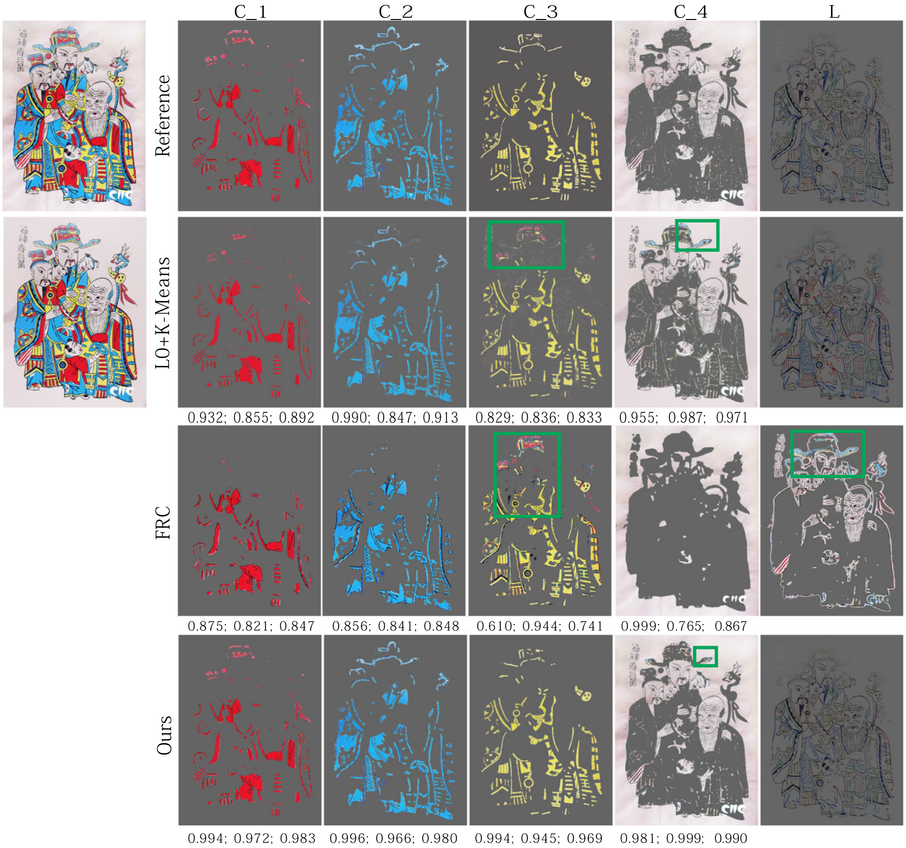


Fig. 14 Comparison with L0+KMeans and FRC on one painting. The three values below each color block represent precision, recall and F1 measure scores of this color block

edge-preserving smoothing filters to WNY paintings in preprocessing. Here we adopt edge-preserving L_0 smoothing filter [13], with prefix “L0+” for each method. We can see that image smoothing has smaller impact on the LCR score than the recomposition MSE. In more detail, L_0 smoothing has positive yet not significant effects on KMeans clustering, no obvious effects on MS and DC-Seg, negative effects on UCM, FRC and our method, and unstable effects on NSGC. We have also tried bilateral filtering [24] and guided filtering [11] as smoothing operators, and obtained similar results. Accordingly, in the following visual comparison, we will compare L0+KMeans, FRC, MS, NSGC, UCM, DC-Seg with our method.

4.4 Visual comparison

Figures 14 and 15 show decomposition results for one WNY painting (with image index as 23). We can see that FRC clusters apparently different colored pixels in one color block. It may be because FRC fuses KMeans results in six different color spaces. The KMeans clustering produces scattered pixels that may be falsely clustered. In this example, KMeans gets one block which looks like line block, but this is an occasional case. Also note its performance is unstable. We have run it for several times and choose the best result. In contrast, our method successfully separates the WNY painting into rather clean color blocks, which are visually close to the man-made reference.

Since MS, UCM, DC-Seg and NSGC can hardly generate the right cluster number, we display decomposition results with pseudo colors in Fig. 15. As these methods have no line blocks, we take their boundary maps as line blocks. From the result, we can see MS, UCM, and DC-Seg tend to over-segment the painting, obtaining much more clusters than the man-made reference, while NSGC generates a cluster number close to the man-made reference. What’s more, the four methods fail to segment thin objects, e.g. the top-left characters.

Figure 16 shows the mean images of the man-made reference and comparative methods for one WNY painting (with image index as 25). We can clearly see that FRC produces quite

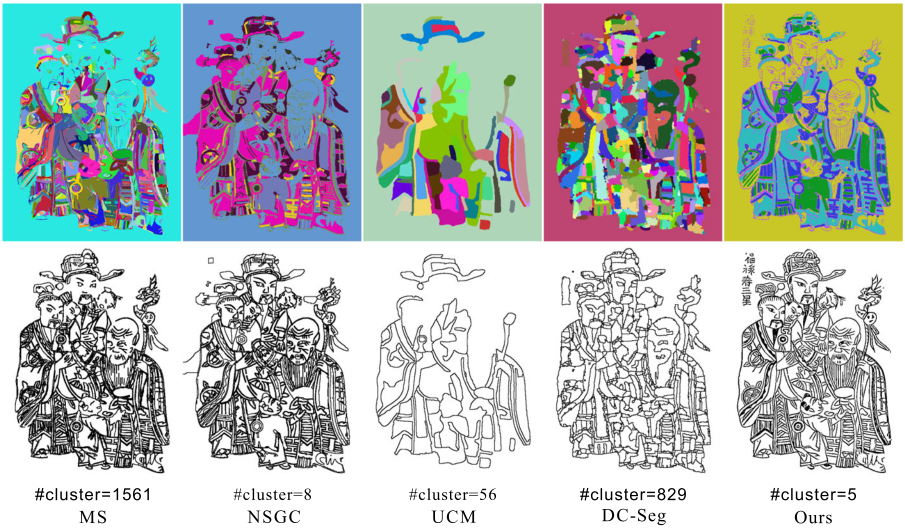
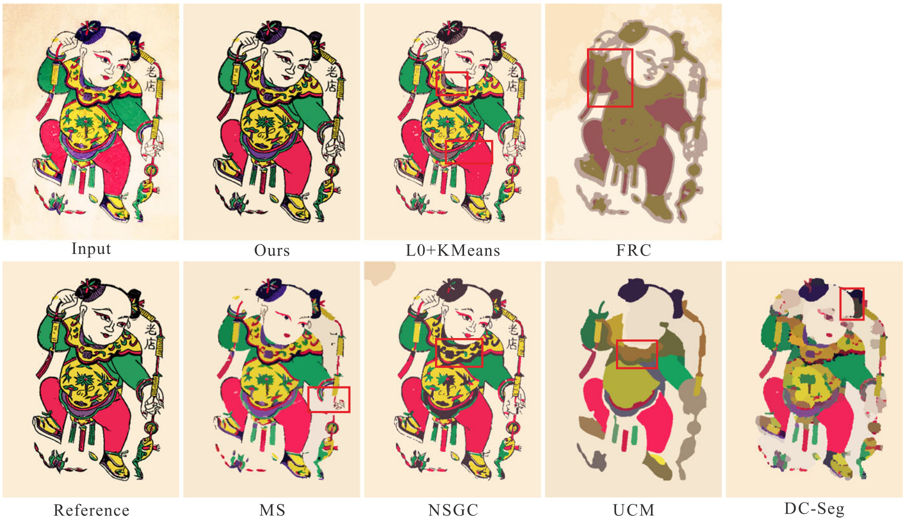


Fig. 15 Comparison with MS, NSGC, UCM, DC-Seg for one new year painting. The first row are the decomposition results and each pseudo color represents a cluster. The second row show the segment boundaries and our line block



Input Ours L0+KMeans FRC

Reference MS NSGC UCM DC-Seg

Fig. 16 Visual comparison of recomposition errors. The mean image of the man-made reference is shown, followed by those of results from L0+KMeans, FRC, MS, NSGC, UCM, DC-Seg and our method

different color appearance. It is due to the mixing of clustered results in six color spaces. Both UCM and DC-Seg maintain major object structures, but lose much details. KMeans and NSGC generate recomposed mean images more similar to the reference mean image, while there are clustering errors in local regions (see the red boxes).

5 Conclusion

In this paper, we present a novel problem of decomposing woodblock images from Chinese new year paintings. After summarizing charateterisitics of WNY paintings, we propose a feasible framework which differentiates the decomposition of line block and that of color blocks. Our line block is firstly obtained by applying a localized binary SVM-classifier and later refined via a guided FDoG filtering. The color blocks are separated by using an efficient color clustering-based method. Experimental results show that the obtained line block is visually complete and the color blocks are clean. The comparison with the state-of-the-art color clustering and image segmentation techniques further validates the advantages of the proposed method.

Our method has spaces to improve. Since the final line block is built on the refined primary line block L'_p, if some isolated yet important lines are not detected in L'_p, our guided FDoG may miss those lines. In addition, the guided FDoG may still introduce some lines between two color regions (see the hairs in Fig. 8). We would also like to explore additional user-friendly ways that facilitate users to further refine line block and color blocks.

Acknowledgements This work is supported by the National Natural Science Foundation of China (61572354, 61671325).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

References

1. Arbelaez P, Maire M, Fowlkes C, Malik J (2011) Contour detection and hierarchical image segmentation. IEEE Trans Pattern Anal Mach Intell 33(5):898–916
2. Chang C-C, Lin C-J (2011) Libsvm: a library for support vector machines. ACM Trans Intell Syst Technol (TIST) 2(3):27
3. Cheng Y (1995) Mean shift, mode seeking, and clustering. IEEE Trans Pattern Anal Mach Intell 17(8):790–799
4. Comaniciu D, Meer P (2002) Mean shift: a robust approach toward feature space analysis. IEEE Trans Pattern Anal Mach Intell 24(5):603–619
5. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):273–297
6. Dinh TN, Park J, Lee C, Lee G (2010) Tensor voting based color clustering. In: IEEE international conference on pattern recognition, pp 597–600
7. Donoser M, Schmalstieg D (2014) Discrete-continuous gradient orientation estimation for faster image segmentation. In: IEEE conference on computer vision and pattern recognition, pp 3158–3165
8. Feng J (2006) Chinese New Year painting corpus. Zhonghua Book Company
9. Feng W, Jia J, Liu Z-Q (2010) Self-validated labeling of markov random fields for image segmentation. IEEE Trans Pattern Anal Mach Intell 32(10):1871–1887
10. Hartigan JA, Wong MA (1979) Algorithm as 136: a k-means clustering algorithm. J R Stat Soc Ser C (Appl Stat) 28(1):100–108
11. He K, Sun J, Tang X (2013) Guided image filtering. IEEE Trans Pattern Anal Mach Intell 35(6):1397–1409
12. Kang H, Lee S, Chui CK (2007) Coherent line drawing. In: ACM international symposium on non-photorealistic animation and rendering, pp 43–50
13. Li Xu, Cewu Lu, Yi Xu, Jia J (2011) Image smoothing via L_0 gradient minimization. ACM Trans Graph 30(6):174–174
14. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
15. Mignotte M (2008) Segmentation by fusion of histogram-based k-means clusters in different color spaces. IEEE Trans Image Process 17(5):780–787
16. Mizuno S, Okada M, Toriwaki J-I (1998) Virtual sculpting and virtual woodcut printing. Vis Comput 14(2):39–51
17. Mizuno S, Kasaura T, Okouchi T, Yamamoto S, Okada M, Toriwaki J (2000) Automatic generation of virtual woodblocks and multicolor woodblock printing. Comput Graphics Forum 19(3):51–58
18. Okada M, Mizuno S, Toriwaki JI (2001) Digital Ukiyo-e preserving project: intelligent coding and constructing archives of printing blocks. In: IEEE international conference on virtual systems and multimedia, pp 209–217
19. Suykens JAK, Vandewalle J (1999) Least squares support vector machine classifiers. Neural Process Lett 9(3):293–300
20. Sỳkora D, Buriánek J, Žára J (2005) Colorization of black-and-white cartoons. Image Vis Comput 23(9):767–782
21. Sỳkora D, Buriánek J, Žára J (2005) Sketching cartoons by example. In: Second eurographics workshop sketch-based interfaces and modeling, pp 27–34
22. Terai T, Mizuno S, Okada M (2004) A color decomposition method for preserving Ukiyo-e woodblocks. In: Eurographics
23. Terai T, Mizuno S, Okada M (2005) Color decomposition for reproducing multi-color woodblock prints. In: ACM Siggraph Posters, p 30
24. Tomasi C, Manduchi R (1998) Bilateral filtering for gray and color images. In: IEEE international conference on computer vision, pp 839–846
25. Von Luxburg U (2007) A tutorial on spectral clustering. Stat Comput 17(4):395–416
26. Wang S (2002) The history of Chinese new year paintings. Beijing Industrial Art Press
27. Yang L, Meer P, Foran DJ (2007) Multiple class segmentation using a unified framework over mean-shift patches. In: IEEE conference on computer vision and pattern recognition, pp 1–8
28. Zhang S-H, Chen T, Zhang Y-F, Hu S-M, Martin RR (2009) Vectorizing cartoon animations. IEEE Trans Vis Comput Graph 15(4):618–629

Liang Wan received the B.Eng and M.Eng degrees in computer science and engineering from Northwestern Polytechnical University, P.R. China, in 2000 and 2003, respectively. She obtained a Ph.D. degree in computer science and engineering from The Chinese University of Hong Kong in 2007. She is currently a full Professor in the School of Computer Software, Tianjin University, P. R. China. Her research interest is mainly on intelligient image synthesis, including image-based rendering, image navigation, pre-computed lighting, and panoramic image processing.

Ye Liu received the M.E. degree in software engineering from the School of Computer Software, Tianjin University, P.R. China, in 2017. Currently she is pursuing the Ph.D. degree in the School of Computer Software, Tianjin University. Her main research interests include structure from motion and superresolution.

Haipeng Dai received the M.E. degree in computer science and engineering from the School of Computer Software, Tianjin University, P.R. China, in 2015. This work was initially conducted by Haipeng Dai before her graduation.

Wei Feng received the B.S. and M.Phil. degrees in Computer Science from Northwestern Polytechnical University, China, in 2000 and 2003 respectively, and the Ph.D. degree in Computer Science from City University of Hong Kong in 2008. From 2008 to 2010, he worked as research fellow at the Chinese University of Hong Kong and City University of Hong Kong, respectively. He is currently a professor in school of computer science and technology, Tianjin University. His major research interest is media computing, specifically including general Markov Random Fields modeling, discrete/continuous energy minimization, image segmentation, semi-supervised clustering, structural authentication, and generic pattern recognition. He got the support of the Program for New Century Excellent Talents in University, China, in 2011.

Jiawan Zhang received the BSci, MPhil, and PhD degrees in computer science from Tianjin University in 1997, 2001, and 2004, respectively. Currently, he is working as a professor in the School of Computer Software, Tianjin University. His main research interests include computer graphics, visual analytics and digital culture heritage. He is a member of the IEEE and ACM, and a senior member of China Society of Image and Graphics, a senior member of China Computer Federation, and a co-chair of Tianjin Society of Image and Graphics.