Colorization Using Quaternion Algebra with Automatic Scribble Generation

Xiaowei Ding*, Yi Xu, Lei Deng, and Xiaokang Yang

Institute of Image Communication and Information Processing
ShangHai Jiao Tong University
ShangHai, China
{dingxiaowei,xuyi, dl0729,xkyang}@sjtu.edu.cn

Abstract. In current colorization techniques, major user intervention is required in the form of tedious, time-consuming scribble drawing. Moreover, color leakage usually occurs across contours and object boundaries. In this paper, we focus on automatic scribble generation and structure-preservation mechanism, which are still open issues of colorization. Firstly, we generate scribbles automatically along points where the spatial distribution entropy achieves locally extreme value. Given the color scribbles, we compute quaternion wavelet phases to conduct colorization along equal-phase lines. These lines across scribbles and monochrome patches locate textures with similar pattern distribution. Contour 'strength' model is also established in scale space to direct color propagation among similar edge structures. Finally, we reconstruct color image patches as vector elements using polar representation in quaternion algebra, well-preserving interrelationship between color channels. The experimental results demonstrate that the proposed colorization method can achieve natural color transitions between different objects with automatically generated scribbles.

Keywords: colorization, automatic scribbles generation, structure-preservation mechanism, quaternion algebra.

1 Introduction

Colorization is an image processing technique which allows us to add colors to monochrome images with the help of manual work. The basic idea of colorization is to set up a mapping from scalar-valued intensity image to vector-valued color image using a few color scribbles provided by users. Therefore, color scribbles should be selected carefully to provide sufficient color reference for monochrome regions. Correct and rapid color propagation should also be triggered from these color scribbles using a color optimization process. In addition, scribbling can be tedious for images with complex details, and requires some skill to obtain natural-looking results.

In scribble-based colorization techniques, most researches are motivated by Levin *et al.* 's work [1], taking the simple premise that nearby pixels in space-time

* Corresponding author.

K. Schoeffmann et al. (Eds.): MMM 2012, LNCS 7131, pp. 103–114, 2012.

that have similar gray levels should also have similar colors. Color leakage tends to occur across contours and object boundaries, lacking structure preservation mechanism. To achieve more natural appearance in colorization results, Drew makes the assumption that the contrast in the colorized image should match the gradient perceived in the original grayscale image [9]. To produce more reliable edge-preserving colorization results, Kim *et al.* enforce color consistency in the areas bounded by the edges [11]. Lezoray's work requires the user to provide more complex scribble guidance to achieve encouraging results, where colorization is considered as a graph regularization problem for a function mapping vertices to chrominance [12]. Heu *et al.* establishes priority propagation mechanism for colorization, ensuring the structure related area with the highest priority can be most reliably colorized [13]. Two basis issues are important in these works, that is, reliable scribble guidance and structure preservation.

Recently, some authors present example-based colorization techniques using one grayscale target image and multiple color reference images. Welsh *et al.* [2] transfer color from reference color image to target monochrome image by matching luminance and texture information between the images. Rather than relying on a series of independent pixel-level decisions, Irony *et al.* [3] develop a new strategy that attempts to account for the higher-level context of each pixel using spatial consistency constraint. X.P. Liu *et al.* focus on tackling illumination differences between grayscale target and color reference images by separating an image into a reflectance (albedo) component and an illumination (shading) component [14]. Sykora *et al.* [10] design a colorization scheme especially suitable for cartoon, which can exploit good image segmentation results prior to colorization. In these methods, the quality of colorized image greatly relies on the similarity between reference image and target image. Moreover, the reference image dataset might be difficult to achieve. In this paper, we are still interested in scribble-based colorization work with attempts to provide automatic scribble generation technique and well preserve image structures during color optimization process. Figure 1 shows the main steps of our colorization algorithm with comparison between our colorization result and the original color image as a rough illustration of our algorithm. The proposed algorithm automatically generates reliable and precise scribbles on input grayscale images based on spatial distribution entropy (SDE). As a result, users only need to decide the color schemes of the scribbles. Given the color scribbles, we compute quaternion wavelet phases to conduct colorization along equal-phase lines. These lines across scribbles and monochrome patches locate textures with similar pattern distribution. A structural priority mechanism based on contour 'strength' model is also established in scale space to direct color propagation among edge structures with different priority. Finally, we reconstruct color image patches as vector elements using polar representation in quaternion algebra, thus, avoid color distortion resulted from isolation of each color channel during colorization process.

The rest of this paper is organized as follows. In Section 2, automatic scribble generation technique is presented in detail. Section 3 proposes a new color optimization process, in which magnitude and phases are separately recovered

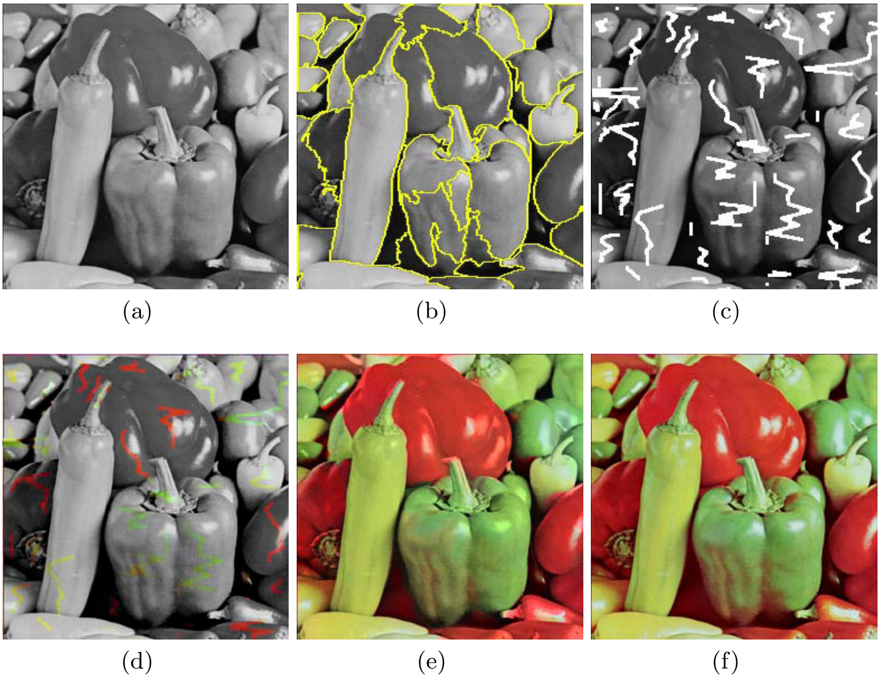


Fig. 1. Example of colorization algorithm in this paper: (a) the input grayscale image: peppers; (b) rough image segmentation for scribble generation; (c) generate scribbles automatically; (d) choose color for scribbles; (e) colorization result; (f) the original color image

under a quaternion color representation framework. Equal-phase lines and contour 'strength' model are computed to direct color propagation among similar image structures. This is followed by the experimental results in Section 4, where a comparison with the state-of-art methods is provided. Finally, conclusion remarks are drawn in Section 5.

2 Automatic Scribble Generation

Scribble plays a significant role in colorization. It contains all the color information that can be used in colorization process. However, images with complex structure need a large amount of careful scribbles by an experienced user, and also, hand-made scribbles will not necessarily trigger the most effective color propagation in consideration of extensive image contents. In this section, we propose an automatic scribble generation algorithm based on spatial distribution entropy, placing scribbles within the regions of high information density. As a result, the requested color information of each homogeneous segment is dominantly contained in the neighborhood of these scribbles.

2.1 The Spatial Distribution Entropy Concept

To measure the spatial distribution of information density in one area, we define the spatial distribution entropy(SDE) at one point in this section. Annular distribution model [5] which performs in a similar way to human visual system is adopted to define the scope of statistics. (see Figure 2 (b)).

Denote:

1. $[p_{xy}]_{M\times N}$ be the image matrix, where p_{xy} is the grayscale value at point (x, y).
2. $U_A = \{(x, y)|(x, y) \in A\}$ be the set of all the pixels in area A.
3. Q be the number of possible values for a single pixel ($Q = 256$ for a typical grayscale image). And the values are represented as $V_1, V_2, ..., V_Q$
4. $S_l = \{(x, y)|(x, y) \in A, p_{x,y} = V_l\}$ be the set of all the pixels in U_A that have value V_l.
5. M be the number of annulars in one scope.
6. $C_l = (x_l, y_l)$ be the center of area S_l where $x_l = \frac{1}{|S_l|} \times \sum_{(x,y)\in S_l} x$, $y_l = \frac{1}{|S_l|} \times \sum_{(x,y)\in S_l} y$.
7. Operator $|\cdot|$ counts the number of elements in a set.

We define the spatial distribution entropy at point p as:

$$E_p = -\sum_{m=1}^{M}\sum_{v=V_1}^{V_Q} \frac{|s_{vm}|}{|U_m|} \times log_2 \frac{|s_{vm}|}{|U_m|} \tag{1}$$

According to the entropy concept in information theory, signals with higher entropy is comparatively harder to predict. SDE measures efficiently this attribute which varies with position parameter in a 2D image signal.

2.2 Scribble Searching Strategy

To avoid one single scribble striding across different homogeneous segments, the image is first oversegmented into closed areas using Graph-Based Image Segmentation algorithm [4]. Then, we perform a search in each segment for the points, around which high information density is accessible. Spatial distribution entropy is adopted to compute information density. In the end, these points are interpolated to form smooth scribbles.

To ensure that there is enough color information for colorization process in a limited number of scribbles, we locate the scribbles at the places possessing high spatial distribution entropy(SDE). Figure 2 intuitively illustrates the details of this procedure. The search starts with the point C_l (the starting point) defined in Section 2.1. Implementation details of adaptive scope radius and selection of scopes in 8 directions are given in caption of Figure 2. Our search strategy allows backtracking of arbitrary number of steps to avoid a bad path brought by any unwise step in this procedure. In addition, it is possible that points are densely selected in a small area due to high entropy. We treat this situation as re-entry and avoid it by measuring the concentration of selected points in

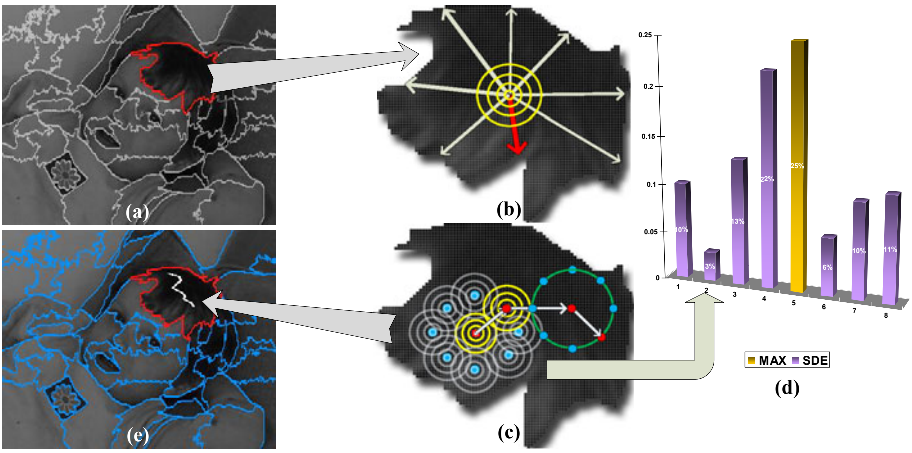


Fig. 2. Implementation details of our scribble generation algorithm (a) Searching procedure are conducted inside one segmented region (red) for each scribble line. (b) Adaptive scope radius is linearly related to the minimum distance between center of scope and region edge (red arrow). (c) Candidate scopes in 8 directions partially overlap with the current(central) scope and the searching path possesses 8 degrees of freedom. (d) Histogram of SDE in 8 directions of the current scope. (e) Fitting all the points on searching path with smooth curve, then one scribble line is generated (the white line in red region).

current scope. To direct the scribble drawing in a homogenous area, we concern about those candidate scopes whose correlation coefficients to current scope are higher than a threshold. Correlations between two scopes are measured by their correlation coefficient of grayscale histograms. Once all the possible scopes in 8 directions fail to satisfy this constraint, we need to backtrack to the previous step to redirect the searching path. Iterate this process until backtracking path is exhausted. We show the flow diagram of our searching strategy in one segment in Figure 3.

3 Colorization Process Based on Quaternion Algebra

This section will first introduce how we characterize image pattern using hyperanalytic signal representation and structural priority in scale space. Then we will focus on color optimization process to propagate color across similar image patches using polar quaternion color formulation.

3.1 Image Structure Analysis Using Quaternion Gabor Phases

Colors often differ in different structural parts in image. Luminance distribution does carry some of the image structure information; however, it cannot act as

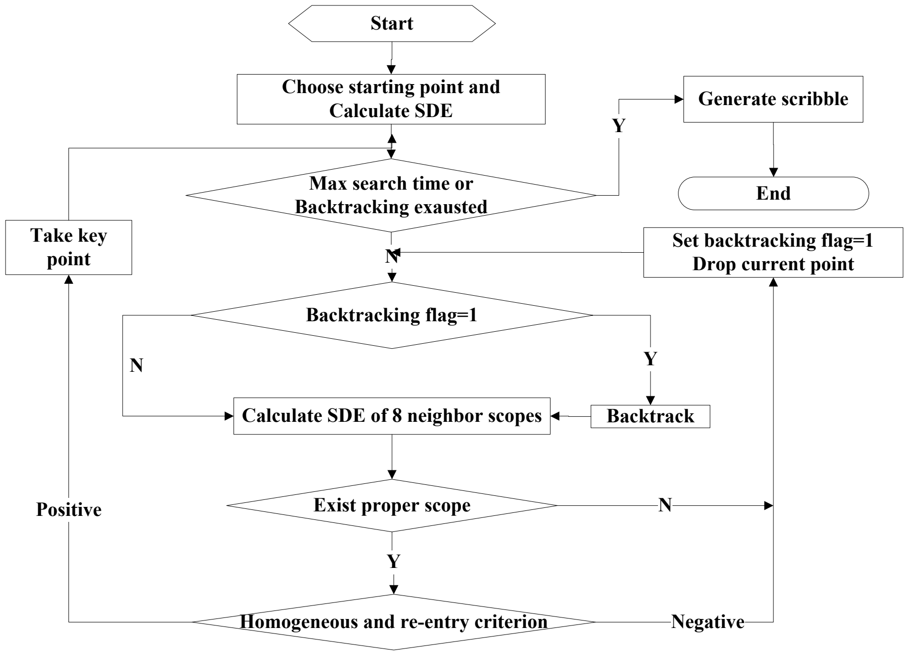


Fig. 3. Scribble searching strategy flow diagram

a useful tool for image analysis under irregular illumination variations. Meanwhile, complex Gabor phases have been extensively exploited to represent image pattern due to the robustness to illumination changes and scale disturbance. In recent work, quaternion Gabor phase is found to provide a better representation of image pattern since it can realize the analysis of intrinsically 2D features (corner-like)[8]. In contrast, complex Gabor phase can only provides a powerful tool for intrinsically 1D features (edge-like) analysis [6]. Considering that (quaternion) Gabor is a windowed (quaternion) Fourier transformation, we can use the Fourier shift theorem to explain it,

$$\begin{aligned} &\mathbf{F}\{f(x-x_0, y-y_0)\} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x-x_0, y-y_0)e^{-i2\pi ux}e^{-i2\pi vy}dxdy \\ &= \mathbf{F}\{f(x,y)\}e^{-i2\pi(ux_0+vy_0)} = \mathbf{F}\{f(x,y)\}e^{-i\Delta\Phi} \end{aligned} \tag{2}$$

$$\begin{aligned} &\mathbf{F}^q\{f(x-x_0, y-y_0)\} = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} f(x-x_0, y-y_0)e^{-i2\pi ux}e^{-j2\pi vy}dxdy \\ &= \mathbf{F}^q\{f(x,y)\}e^{-i2\pi ux_0}e^{-j2\pi vy_0} = \mathbf{F}^q\{f(x,y)\}e^{-i\Delta\varphi}e^{-j\Delta\theta} \end{aligned} \tag{3}$$

where $\mathbf{F}\{\cdot\}$ and $\mathbf{F}^q\{\cdot\}$ denote complex Fourier transform and quaternion Fourier transform, separately. It is noted that quaternion Gabor can encode the relative location shifts x_0 and y_0 using two separate phases $\Delta\varphi$ and $\Delta\theta$, while complex Gabor can provide only one phase $\Delta\phi$ and merges this important information.

In this section, we use quaternion Gabor phases to measure structural similarity between two neighborhoods. Firstly, we establish octave-band quaternion Gabors,

$$G^q_{\sigma\alpha}(\mathbf{x},\mathbf{u}) = \frac{1}{2\pi\sigma^2} e^{\frac{-(x^2+y^2)}{2\sigma^2}} e^{-i2\pi u x'} e^{-j2\pi v y'} \tag{4}$$

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix} \begin{pmatrix} \cos\alpha & \sin\alpha \\ -\sin\alpha & \cos\alpha \end{pmatrix} \tag{5}$$

where $G^q_{\sigma\alpha}$ represents the quaternion Gabor kernel with scale σ and orientation α. Then we get three quaternion Gabor phases φ, ψ, θ following the quaternion algebra rule in [6] strictly,

$$\Phi_{\sigma\alpha}\{\varphi, \psi, \theta\} = arg(I * G^q_{\sigma\alpha}) \tag{6}$$

where $\Phi_{\sigma\alpha}$ denotes quaternion phase vector and I represent the monochrome image.

Now we define a metric to estimate the structural homogeneity between two neighboring pixels $\mathbf{p_a}$, $\mathbf{p_b}$ under scale σ and orientation α,

$$\begin{aligned} H_{\sigma\alpha}(\mathbf{p_a}, \mathbf{p_b}) = &\frac{1}{4}(|\rho(\mathbf{p_a})\varphi^q_{\sigma\alpha}(\mathbf{p_a}) - \rho(\mathbf{p_b})\varphi^q_{\sigma\alpha}(\mathbf{p_b})|_{2\pi}) \\ &+\frac{1}{2}(|\rho(\mathbf{p_a})\theta^q_{\sigma\alpha}(\mathbf{p_a}) - \rho(\mathbf{p_b})\theta^q_{\sigma\alpha}(\mathbf{p_b})|_{\pi}) \\ &+(|\rho(\mathbf{p_a})\psi^q_{\sigma\alpha}(\mathbf{p_a}) - \rho(\mathbf{p_b})\psi^q_{\sigma\alpha}(\mathbf{p_b})|_{\frac{\pi}{2}}) \end{aligned} \tag{7}$$

where ρ is the amplitude of polar quaternion representation, the scalars $\frac{1}{4}$ and $\frac{1}{2}$ are used to keep the same value range of three quaternion phases. It should be noted that mod operator $|\cdot|_\beta(\beta = 2\pi, \pi, \frac{\pi}{2})$ is used to deal with phase wrapping problem when we conduct subtraction operation between two phases. In section 5, we will demonstrate in experiments that quaternion Gabor phase surpass complex Gabor phase in colorization application.

3.2 Scale-Space Contour 'Strength'

In this section, we will introduce a structural priority framework based on scale-space contour 'strength' in order to measure the structural importance of global contours and local contours. In our algorithm, contours are extracted to characterize the structures of objects in an image and act as a compensation of quaternion Gabor phase pattern. Furthermore, global contours and local ones would be differently treated since contours in larger scale scope stand for more critical structures.

We define the importance of a contour as strength and propose an algorithm to calculate strength value. Our algorithm is based on multi-scale canny edge representation [7], where scale factor is denoted as σ_i and the successive scales ranging from $\sigma_1(\sigma_{min})$ to $\sigma_n(\sigma_{max})$ satisfy $\sigma_i = \sigma_1 + (i-1)\Delta\sigma$. As a result, the contour image $C(\sigma_i)$ is expected to contain coarse structures and less fine

Table 1. Algorithm for calculation of Contour 'Strength' of an image

Calculate 'Strength' based on Multi-Scale Canny Edge Detection
Initialize: $Strength = C(\sigma_1)$
for $i = 2$ **to** n
dilate $C(\sigma_i)$ with radius $i-1$ obtaining a dilated image d_i
for each edge pixel $p(x,y)$ in d_i
if $p(x,y)$ is also the edge pixel of $C(\sigma_{i-1})$
$Strength(x,y) = Strength(x,y) + 1$
end if
end for
dilate d_i with radius $i-1$ obtaining another dilated image d_i'
for each non-edge pixel $p(x,y)$ in d_i'
if $p(x,y)$ is the edge pixel of $C(\sigma_i)$
$Strength(x,y) = Strength(x,y) + i$
end if
end for
end for

details. Further, a contour should be assigned a large strength value if it has a long lifetime in scale space since these contours always represent the most significant structures of an image. This means that it should have higher priority in colorization process. Since multi-scale contours might not exactly match at the same location throughout the scale space, we use morphology algorithm dilating to achieve more reasonable strength estimation. The proposed algorithm is illustrated in table 1.

3.3 Color Diffusion

The final step in colorization process is color diffusion based on quaternion phase pattern and contour strength designed above. We implement the color diffusion process by minimizing a cost function in which variables are quaternion color phases. In quaternion color space, each pixel is represented by a pure quaternion $q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$ where $a = 0$ and b, c, d are RGB values. In polar coordinates, the quaternion can be formed as amplitude and phases which can be calculated as described in [6].

Vector operation in color space allows different color channels to be treated as a unity rather than channel being independently manipulated. Illumination variation has little impact on quaternion phases of color image. Hence, it is more reasonable to perform colorization by reconstructing quaternion phases vector in the minimization process.

Inspired by Levin's work [1], We build the cost function to be minimized as follows.

$$\Phi^{opt} = Arg \ \min_{\Phi} \sum_{p} \left| \Phi(p) - \frac{\sum\limits_{q \in N(p)} W_{pq} \Phi(q)}{\sum\limits_{q \in N(p)} W_{pq}} \right|, \quad \Phi = \{\varphi, \psi, \theta\} \tag{8}$$

As a matter of fact, the optimization problem is treated as three independent minimization problems about three phase components ϕ, ψ, and γ.

$$W_{pq} = W_{pq}^{Strength} \times W_{pq}^{QGabor} \tag{9}$$

$$W_{pq}^{Strength} = \exp(-|strength(\mathbf{p}) - strength(\mathbf{q})|) \quad (10)$$

$$W_{pq}^{QGabor} = \exp(-\sum_{\sigma}\sum_{\alpha} H_{\sigma\alpha}(\mathbf{p}, \mathbf{q})) \quad (11)$$

Where $N(\mathbf{p})$ is the 3 by 3 neighborhood of pixel $\mathbf{p}$. W_{pq}^{QGabor} is designed to restrict the neighboring pixels in one homogeneous area to have similar color. Value of $W_{pq}^{Strength}$ tend to emphasize the colorization of contours with high strength. It keeps constant in non-contour area as defined in Section 3.2. This weight leads to promising color propagation along the edge. We use Matlab's built- in least squares solver for sparse linear systems.

4 Experimental Results

Here we show our colorization results, where scribbles are generated automatically. Performance comparison with other colorization algorithms is also provided in this section. It should be noted that our algorithm only needs a rough over-segmentation of the input image in automatic scribble generation process. Hence, we simply set the area of the smallest segment part as $M \times N \times 5\%$,

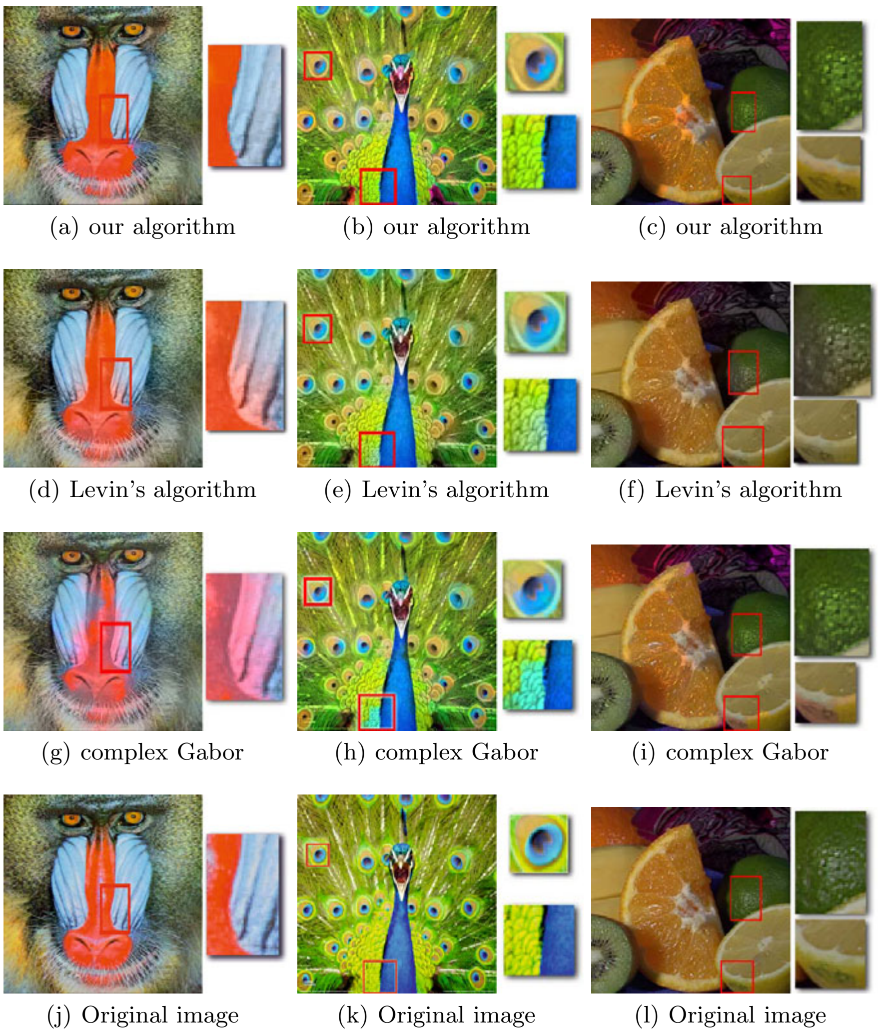
(a) (b) (c)

(d) (e) (f)

Fig. 4. Colorization results compared with the original color images: (a) automatically-produced scribbles (house); (b) colorization result (house); (c) original color image (house); (d) automatically-produced scribbles (Lena); (e) our colorization result (Lena); (f) original color image (Lena)

where $M \times N$ stands for the image resolution. In our simulation, the correlation threshold (see Section 2.2) for the M^{th} scope on the searching path has the value $T_{M+1} = \frac{0.6}{M}\sum_{i=2}^{M} r_i$, where r_i represents the correlation between the i^{th} scope and the $(i-1)^{th}$ one.

Figure 4 gives two colorization results on house image and Lena image, where automatically-produced scribbles are marked using white curves. As shown in Figure 4, scribbles are automatically located in those regions which are supposed to contain critical color structures. No scribble strides across different colors.



(a) our algorithm (b) our algorithm (c) our algorithm

(d) Levin's algorithm (e) Levin's algorithm (f) Levin's algorithm

(g) complex Gabor (h) complex Gabor (i) complex Gabor

(j) Original image (k) Original image (l) Original image

Fig. 5. Comparison with other algorithms; (a), (b), (c): Our algorithm; (d), (e), (f): Levin's algorithm; (g), (h), (i): Algorithm using complex Gabor; (j), (k), (l): Original color image

The original images in the third column are listed to provide visual evaluation of colorization performance. We can see that our algorithm reconstructs the missing color information very well.

In figure 5, we compare our algorithm with those methods depending on intensity similarity or complex Gabor phases, proving that quaternion Gabor phases and scale-space contour strength model can represent fundamental image structures. The quality of the colorization results in figure 5 are measured by PSNR. PSNR of our results: baboon(a) 23.81dB, peacock(b) 20.88dB, fruits(c) 21.73dB. PSNR of Levin's results: baboon(d) 23.58dB, peacock(e) 21.59dB, fruits(f) 21.94dB. PSNR of results using complex Gabor phases: baboon(g) 22.78dB, peacock(h) 19.85dB, fruits(i) 20.80dB. Levin's results and our results have very small differences in PSNR. Results by algorithm using complex Gabor phases are about 1dB lower than ours and Levin's, which leads to unsatisfying visual effects. Comparing with Levin's algorithm in PSNR, our algorithm performs better on baboon image(a), while a little bit worse on peacock image(b) and fruits image(c). However, our colorization results promisingly recover the complex color texture in baboon's villus, peacock plumage and citrus peel. We also achieve almost zero color leakage around object contours, which cannot be achieved by other two algorithms. From the close-up image in Figure 5, we can see that our results have better visual effects than algorithm based on intensity similarity(Levin's algorithm) under similar PSNR. Human visual perception is highly adapted for extracting structural information from a scene [15], which explains why our structure-preserving colorization results are more visually satisfying.

It takes about 11s in average to generate high quality scribbles automatically for each of these input images on PC. Manual scribbles usually take at least several minutes to sketch with professional guidance. Automating this time-consuming process in colorization is user friendly and greatly shortens the overall time of colorization.

5 Conclusion

The evaluation criterion for an outstanding colorization algorithm is whether it can achieve prospective effect requiring as little manual input as possible. Our work achieves this goal by proposing a colorization algorithm based on quaternion algebra with automatic scribble generation. Users are expected to enjoy the convenience of choosing colors for scribbles that have already been created. At the same time, colorization effects are improved in complex color texture regions under quaternion color representation and hyperanalytic signal analysis. Concept of structure priority based on scale-space contour strength together with quaternion phases forms a reliable structure-preserving colorization foundation.

Acknowledgments. This work was supported in part by Research Fund for the Doctoral Program of Higher Education of China (200802481006), NSFC-60902073, NSFC-60932006.

References

1. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. ACM Trans. Graph. 3, 689–694 (2004)
2. Welsh, T., Ashikhmin, M., Mueller, K.: Transferring color to grayscale images. ACM Transaction on Graphics 21(3), 277–280 (2002)
3. Irony, R., Cohen-Or, D., Lischinski, D.: Colorization by example. In: Eurographics Symposium on Rendering, pp. 201–210 (2005)
4. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient Graph-Based Image Segmentation. International Journal of Computer Vision, 167–181 (2004)
5. Rao, A., Srihari, R.K., Zhang, Z.: Spatial Color Histograms for Content-Based Image Retrieval. In: ICTAI, pp. 183–186 (1999)
6. Bayro-Corrochano, E.: The Theory and Use of the Quaternion Wavelet Transform. Journal of Mathematical Imaging and Vision 24, 19–35 (2006)
7. Canny, J.: A Computational Approach To Edge Detection. IEEE Trans. Pattern Analysis and Machine Intelligence 8(6), 679–698 (1986)
8. Xu, Y., Song, L., Yang, X., Traversoni, L., Lu, W.: QWT: retrospective and new Applications. Geometric Algebra Computing for Engineering and Computer Science. Springer, London (2010)
9. Drew, M.S., Finlayson, G.D.: Realistic colorization via the structure tensor. In: ICIP, pp. 457–460 (2008)
10. Sykora, D., Burinek, J., Zra, J.: Unsupervised colorization of black-and-white cartoons. In: 3rd Int. Symp. NPAR (2004)
11. Kim, T.H., Lee, K.M., Lee, S.U.: Edge-preserving colorization using data-driven random walks with restart. In: ICIP, pp. 1641–1644 (2009)
12. Lezoray, O., Ta, V., Elmoataz, A.: Nonlocal graph regularization for image colorization. In: ICPR, pp. 1–4 (2008)
13. Heu, J., Hyun, D., Kim, C., Lee, S.: Image and video colorization based on prioritized source propagation. In: ICIP, pp. 465–468 (2009)
14. Liu, X., Wan, L., Qu, Y., Wong, T., Lin, S., Leung, C., Heng, P.: Intrinsic colorization. ACM Trans. Graph. 27(5), 152:1–152:9 (2008)
15. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image Quality Assessment: From Error Visibility to Structural Similarity. IEEE Trans. Image Processing. 13(4), 600–612 (2004)