Patch-driven colorization

Chen Yao
Xiaokang Yang
Jia Wang
Li Song
Guangtao Zhai
Shanghai Jiao Tong University
Institute of Image Communication and Information Processing
800 Dong Chuan Road
5 Dian Xin Building, Room 321
Shang Hai 200240, China
E-mail: yaochen_mm@sjtu.edu.cn

Abstract. Colorization is the process of adding colors to monochrome images. In this paper we present a scribble-based colorization, which uses nearest neighborhood propagation and mixed weighing. The major assumption of our method is that chrominance is similar if luminance is similar in a natural image. We introduce the definition of the nearest neighborhood and mixed weighting. Non-local-mean-based patch weight and point weight are used in the mixed weighting. Experimental results show the benefits of our method. [DOI: 10.1117/1.3281666]

Subject terms: colorization; propagation; mixed weighting.

Paper 090521RR received Jul. 13, 2009; revised manuscript received Nov. 10, 2009; accepted for publication Nov. 12, 2009; published online Jan. 7, 2010.

1 Introduction

Colorization is the process of adding colors to monochrome images. It is usually done manually by professionals such as color experts or artists. A major difficulty in colorization is its labor intensity. For example, in order to colorize an image an artist typically begins by dividing it into regions, and next assigns a color to each region. This approach, also known as the segmentation method, is very time consuming.[1] The process of dividing the picture into correct segments is painstaking. Computer-aided colorization techniques can improve the efficiency of colorization.

State-of-the-art colorization can be divided into two categories. One is the scribble-based method. Colorization is performed by propagating user-provided color scribbles on the grayscale image to the rest of the image. Levin et al.[2] proposed a simple user-guided colorization algorithm in which the scribbles' colors are propagated to the rest of the image with a least-squares constraint. However, it may produce color pervasion. Yatziv and Sapiro[3] make use of image geodesic distances for colorization, where the distance weights between marked and unmarked points are computed in order to obtain the final chrominance of the unmarked points. Colorization by sowing color pixels has been proposed by Horiuchi and Hirano.[4] However, it is not good for complex, detailed images. To avoid the burden of scribbling over images with complex textures, Qu et al.[5] Qing et al.[6] both employed texture grouping techniques to colorize cartoon and natural images, respectively.

Another kind of colorization technique is example-based colorization, which transfers colors in an example image to the target image. This method does not depend on the user's skill or experience to choose color scribbles. A simple and efficient method is proposed by Reinhard et al.[7] based on color space mapping. Welsh et al.[8] proposed a pixel-based approach to colorize an image by matching swatches between the target grayscale and a color reference image. References 9 and 10 describe two kinds of colorization by using segmentation. Intrinsic colorization[11] colorizes target images by decomposing the grayscale target image's intrinsic reflectance and illumination components. However, example-based colorization methods rely heavily on reference images, and it is usually quite difficult to select the candidates for those images.

The current work is partially inspired by the work of Ref. 2, which uses a simple premise that neighboring pixels that have similar intensities should have similar colors. We also introduce the non-local-mean algorithm[12] in our method. As a kind of neighborhood filter,[13] the non-local-mean algorithm examines not only the similarity of grayscale level of a single pixel, but the geometrical structure in the neighborhood, thus making full use of the self-similarities of an image (i.e., the extent to which whole image has the same features as one or more of its parts).

In this paper, we also exploit the assumption that the luminance channel faithfully represents the geometrical structure of an image. We use an effective approach similar to the non-local-mean method to compute the characteristics of the local geometry of the monochrome image. Nearest neighborhood propagation (NNP) is used. Next we apply the luminance's self-similarity to the chrominance, using mixed weighting. The proposed method uses the color scribbles as linear constraints to solve a quadratic programming problem. The mixed weighting consists of two parts: the patch weight, which is the non-local-mean weight, and the point similarity weight, which is described in the following.

Our contribution is a new, simple yet effective interactive colorization technique. In addition to grayscale image colorization, our technique is also applicable to recolorization of still images.

2 Problem Setup

In Ref. 2, the accuracy of optimization solving depends on the number of scribbles. In our case the scribbles of the test image in Fig. 1 contains 16,822 pixels, which is nearly one-third of the total pixels. If we have fewer scribbles, we obtain Fig. 4(b) and Fig. 5(b). It can be seen that the colors have been pervaded to noncoherent areas in these two images. So the problem is how to obtain better results when the number of scribbles is very limited.

In the quadratic programming problem, the number of

0091-3286/2010/$25.00 © 2010 SPIE

Fig. 1 Scribbled image.

constraints is reduced when the number of scribbles is reduced. The constraint condition is related to the chrominance weight and scribble number. So we can improve the constraint condition by means of a better chrominance weight.

3 Proposed Approach

In the first subsection, the NNP algorithm is presented. After that, we propose a mixed weighting algorithm for grayscale image colorization.

3.1 *Nearest Neighborhood Propagation*

We first mark the nearest neighborhood points. Next we use Euclidean distance to propagate these marked points. Let the pixels within a two-dimensional image $f(x,y)$, where (x,y) are the coordinates of the pixel, be divided into two classes—the marked pixel set Ω_s and the unmarked pixel set Ω_r:

$$\Omega_s \cup \Omega_r = \{(x,y): x=1,\ldots,W,\ y=1,\ldots,H\}, \tag{1}$$

where W and H stand for the width and height of the image, respectively.

We use the following equation as the image mask for nearest neighborhood propagation:

$$\text{mask}(x,y) = \begin{cases} 0 & (x,y) \in \Omega_s, \\ 1 & (x,y) \in \Omega_r. \end{cases} \tag{2}$$

We set $\text{mask}(x,y)=0$ if the distance from the current scribbled pixel to the scribbled point is less than a threshold, which can be determined empirically. Next we obtain a new image mask containing nearest neighborhood propagation pixel information.

3.2 *Mixed Weighting Color Propagation*

Our algorithm works in the YUV color space. For a given grayscale image, we use Y directly as the luminance channel in a YUV color space to construct a target image. We first compute neighbor points' non-local-mean weights in a search window about an unmarked point. Next we compute points' similarity weights within the same search window. After that, we compute the chrominance channels, assuming that they have the same local pixel neighborhood relations as the Y channel. The target chrominance is obtained by solving a linearly constrained quadratic optimization problem with the following cost function:

$$\hat{U}_s = \arg\min\left\{\left\| U_s - \sum W_* \cdot U_r \right\|^2\right\}, \tag{3}$$

where U_s is the chrominance of unmarked points, U_r is the neighbor chrominance of unmarked points in a search window, and W_* is given by

$$W_* = W_{\text{patch}} \times W_{\text{point}}. \tag{4}$$

The weight W_* consists of two parts: the patch weight and the point weight. The first weight W_{patch} measures the similarity between neighboring patches. The second weight W_{point} measures the similarity between current point and neighboring points, and Euclidean distance is applied here. Note that $0 \leqslant W_{\text{patch}} \leqslant 1$, $0 \leqslant W_{\text{point}} \leqslant 1$, and we have

$$\sum_{i\in\mathcal{N}_i} W_{\text{patch}}(i) = \sum_{i\in\mathcal{N}_i} W_{\text{point}}(i) = 1, \tag{5}$$

where $\mathcal{N}_i$ is the neighborhood pixel index, and

$$W_{\text{patch}}(i) = \frac{1}{Z(i)} \exp\left\{-\frac{d_f^2(x,y)}{h^2}\right\}, \tag{6}$$

where

$$d_f^2(x,y) = \|f(\mathcal{N}_i) - f(\mathcal{N}_j)\|^2_{G_\sigma} \tag{7}$$

is the L_2 norm of the difference of $f(\mathcal{N}_i)$ and $f(\mathcal{N}_j)$, weighted against a Gaussian G_σ with standard deviation σ. The $d_f(x,y)$ measures how similar two patches centered at x and y are. If two patches are similar, the corresponding weight coefficient will be high. If the patches are dissimilar, the weight coefficient will be small. Here $f(\mathcal{N}_j)$ denotes the pixel intensity located in the square neighborhood $(\mathcal{N}_i)$ of fixed size, and $Z(i)$ is the normalizing constant given by

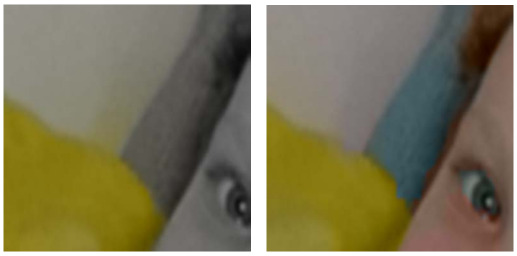
$$Z(i) = \sum_i \exp\left\{-\frac{\|f(\mathcal{N}_i) - f(\mathcal{N}_j)\|_2^2}{h^2}\right\}, \tag{8}$$

where the parameter h acts as a degree of filtering.

Let $f:\mathbb{R}^2 \to \mathbb{R}^+$ be the function on an image that maps the coordinates of a pixel (i,j) to the pixel intensity. Next for any given pixel φ at (i,j) within a neighborhood of size n, which has φ_0 as its center, W_{point} is determined by the following function:

$$W_{\text{point}} = \exp\left\{-\frac{\|f(\varphi_i) - f(\varphi_0)\|_2^2}{2\sigma_r^2}\right\}. \tag{9}$$

For the point weight of each pixel in the image, the central pixel φ_0 contributes more significantly than its neighboring pixels. Greater point weights are assigned to those neighboring pixels with pixel intensity more similar to the central pixel's



(a) (b)

Fig. 2 (a) Colored image with fixed window. (b) Colored image with adaptive window. (Color online only.)

3.3 *Adaptive Window*

In the patch weight computation, we adopt a square fixed window for the neighborhood. Actually, there is color pervasion in the restored chrominance when the square window overlaps a discontinuity. Of course, the color pervasion can be reduced by increasing the number of scribbles. We use an adaptive window for weight computation, instead of increasing the number of scribbles, because of the anisotropy of images. That means the patch size will change according to the distance from the current pixel to the nearest edge. The closer the current pixel to the nearest edge, the smaller the patch size is. Figure 2(a) and 2(b) show the different results using fixed windows and adaptive windows.

3.4 *Algorithm Pseudocode*

We provide pseudocode that summarizes the proposed method:

- **Input:**
  grayscale image
  scribbles.
- **Output:**
  colored image.
- **Definition:**
  $\Omega_s^n \leftarrow$ {new scribbled image}
  $W_* \leftarrow$ {mixed weight}
  $U_s \leftarrow$ {final chrominance}.
- **Algorithm:**
  For all scribbles in UV channel:

(1) Propagate scribbles and record new scribble location.
(2) $\Omega_s^n \leftarrow$ {new scribbles}.
(3) $W_{\text{patch}} \leftarrow$ {compute patch weight}.

(a) (b) (c)

Fig. 4 Test example for one-line color propagation: (a) one-line scribble in a grayscale image, (b) Levin et al.'s approach for color propagation, (c) the result of our algorithm. (Color online only.)

(4) $W_{\text{point}} \leftarrow$ {compute point weight}.
(5) $W_* \leftarrow \{W_{\text{patch}} \times W_{\text{point}}\}$.
(6) U_s
$\leftarrow$ {solve quadratic optimization by Mosek5.0}.
(7) Obtain colored image by Y and U_s.

4 Experimental Results

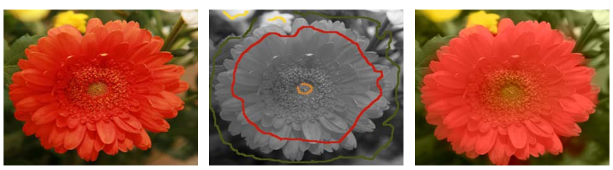
We now present experiment results of our colorization method. The resolution of the grayscale image adopted for the colorization test is 320×265. The size of initial search window in computation of the mixed weights is 7×7.

Figure 3 shows a colorization example using our algorithm. Figure 3(a) is the original color image. Figure 3(b) is a grayscale image with color strokes, which were added with Adobe Photoshop, and the scribble points' minimum size is 3×3. Figure 3(c) is the test result, which demonstrates that our method performs better than the original color image in Fig. 3(a).

Figure 4 is an example of one-line color propagation. Figure 4(a) is the original grayscale image with a yellow line stroke. Figure 4(b) is the propagation result of Ref. 2. Figure 4(c) is the result of our algorithm, which illustrates that our method can well satisfy the constraints on color propagation within regions of similar texture. There is no color pervasion over texture boundaries. In contrast, Ref. 2 leads to obvious color pervasion.

Figure 5(a) is the original grayscale image scribbled with some different color strokes. Figure 5(b) is the propagation result of Ref. 2. Figure 5(c) shows that our method does better than Levin et al.'s on local colorization.

Recolorization is a process that modifies the regional colors of an image. It can be applied, for example, to image color correction. Our colorization algorithm can also be applied to the recolorization problem, in which we need to replace original chrominance with propagated chrominance. Figure 6 is an example of recolorization. Figure 6(a)



(a) (b) (c)

Fig. 3 Colorization example: (a) original color image, (b) the scribbles in grayscale image, (c) image after colorization. (Color online only.)

(a) (b) (c)

Fig. 5 Regional color propagation test example: (a) local scribbled grayscale image, (b) Levin et al.'s approach for color propagation, (c) the result of our algorithm. (Color online only.)

(a) (b) (c)

Fig. 6 Recolorization example: (a) original color image, (b) scribbled color image, (c) recolored result. (Color online only.)

(a) (b) (c)

Fig. 7 Recolorization example: (a) original color image, (b) scribbled color image, (c) recolored result. (Color online only.)

(a) (b) (c)

Fig. 8 Recolorization example: (a) original color image, (b) scribbled color image, (c) recolored result. (Color online only.)

(a) (b) (c)

Fig. 9 Global colorization example: (a) grayscale image, (b) color image with scribbles, (c) the result of our algorithm. (Color online only.)

(a) (b) (c)

Fig. 10 Global colorization example: (a) grayscale image, (b) color image with scribbles, (c) the result of our algorithm. (Color online only.)

(a) (b) (c)

Fig. 11 Global colorization example: (a) grayscale image, (b) color image with scribbles, (c) the result of our algorithm. (Color online only.)

is the original color image. Figure 6(b) is the color image with color scribbles. Figure 6(c) is the result of recolorization by our method.

Figure 7 is another example of recolorization. Figure 7(a) is the original color image. Figure 7(b) is the color image with color scribbles. Figure 7(c) is final recolorization result, which demonstrates that our method can be used to propagate red color until an intensity boundary is found. Figure 8 also shows recolorization. Figure 8(b) is a color image with more color scribbles than in Fig. 7(b), and Fig. 8(c) demonstrates the final recolored result.

Figures 9–11 show our colorization results over different kinds of images. Figures 9(a), 10(a), and 11(a) are the original grayscale images Figures 9(b), 10(b), and 11(b) are the scribbled color images; and Figures 9(c), 10(c), and 11(c) are our colorization results. The experiments demonstrate that our colorization algorithm gets favorable results.

5 Conclusion and Future Work

In this paper, we have presented a scribble-based colorization method driven by patch-based non-local-mean weighting. Experimental results demonstrate the benefits of mixed weighting colorization. In addition to colorization, the method is also applicable to recolorization. The major novelty and improvement over the previous algorithms is the introduction of a mixed weighting strategy, which efficiently controls chrominance pervasion. Besides the mixed weighting method, NNP and an adaptive window searching algorithm are also introduced for improving colorization efficiency. Although satisfactory experiments are demonstrated, labeling scribbles in grayscale image is still a problem. In future work, we intend to focus on evaluation of the scribble position and how to obtain best labeling.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (60632040 and 60932006), the 111 Project (B07022), and the Program for the Professor of Special Appointment (Eastern Scholar) at Shanghai Institutions of Higher Learning.

References

1. "Film colorization," http://en.wikipedia.org/wiki/Film_colorization, accessed June 2009.
2. A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *ACM Trans. Graphics* **23**(3), 689–694 (2004).
3. L. Yatziv and G. Sapiro, "Fast image and video colorization using chrominance blending," *IEEE Trans. Image Process.* **15**(5), 1120–1129 (2006).
4. T. Horiuchi and S. Hirano, "Colorization algorithm for grayscale image by propagating seed pixels," in *IEEE Int. Conf. on Image, Processing*, pp. 457–460 (2003).
5. Y. Qu, T.-T. Wong, and P.-A. Heng, "Manga colorization," *ACM Trans. Graphics* **25**(3), 1214–1220 (2006).
6. L. Qing, F. Wen, D. Cohen-Or, L. Liang, Y.-Q. Xu, and H. Shum, "Natural image colorization," in *Rendering Techniques 2007 (Proc. Eurographics Symp. on Rendering)*, pp. 309–320 (2007).
7. E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color transfer between images," *IEEE Comput. Graphics Appl.* **21**(5), 34–40 (2001).
8. T. Welsh, M. Ashikhmin, and K. Mueller, "Transferring color to grayscale images," *ACM Trans. Graphics* **21**(3), 277–280 (2002).
9. R. Irony, D. Cohen-Or, and D. Lischinski, "Colorization by example," in *Rendering Techniques 2005*, pp. 201–210, IEEE Computer Soc. Press (2005).
10. Y.-W. Tai, J. Jia, and C.-K. Tang, "Local color transfer via probabilistic segmentation by expectation-maximization," in *IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 747–754 (2005).
11. X. Liu, L. Wan, Y. Qu, T.-T. Wong, S. Lin, C.-S. Leung, and P.-A.

Heng, "Intrinsic colorization," *ACM Trans. Graphics* **27**(5), 11–13 (2002).
12. A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Int. Conf. on Computer Vision and Pattern Recognition*, vol. **2**, pp. 60–65 (2005).
13. C. Kervrann and J. Boulanger, "Local adaptivity to variable smoothness for exemplar-based image denoising and representation," *Int. J. Comput. Vis.* **79**(1), 45–69 (2008).

Chen Yao received the BE and ME degrees in electronics engineering from East China University of Science and Technology, Shanghai, China. He is currently pursuing the PhD degree at the Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai, China.

Xiaokang Yang received the BSci degree from Xiamen University, Xiamen, China, in 1994, the MEng degree from the Chinese Academy of Sciences, Beijing, China, in 1997, and the PhD degree from Shanghai Jiaotong University, Shanghai, China, in 2000. He is currently a professor at the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China. From April 2002 to October 2004, he was a research scientist at the Institute for Infocomm Research, Singapore. His current research interests include scalable video coding, video transmission over networks, video quality assessment, digital television, and pattern recognition.

Jia Wang received his BSc, MSc, and PhD degrees from Shanghai Jiao Tong University, Shanghai, China. He is currently an associate professor at the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China. His research interests include information theory and video processing and transmission.

Li Song received the BS and MS degrees in electronic engineering from Nanjin University of Science and Technology in 1997 and 2000, and the PhD degree from Shanghai Jiao Tong University, Shanghai, China, in 2005. He is currently a lecturer at the Institute of Image Communication and Information Processing, Department of Electronic Engineering Shanghai Jiaotong University, Shanghai, China. He has published more than 30 research papers and filed 15 patents. His research interests are in the areas of signal processing, image and video coding, computer vision, and machine learning.

Guangtao Zhai received the BE and ME degrees in electronics engineering from Shan Dong University, Jinan, China, in 2001 and 2004, respectively. He is currently pursuing the PhD degree at the Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai, China.