ROBUST AND AUTOMATIC VIDEO COLORIZATION VIA MULTIFRAME REORDERING REFINEMENT

Sifeng Xia[1], Jiaying Liu[1,], Yuming Fang[1,2], Wenhan Yang[1] and Zongming Guo[1,3]*

[1]Institute of Computer Science and Technology, Peking University, Beijing, China
[2]School of Information Technology, Jiangxi University of Finance and Economics, Nanchang, China
[3]Cooperative Medianet Innovation Center, Shanghai, China

ABSTRACT

In this paper, we propose a robust video colorization method automatically through limited color references in a video sequence. The proposed method first estimates motion vectors between a monochrome frame and colored reference frames for initial matching by optical flow. Then it transfers color information to matched points in the monochrome frame and further propagates color information of matched points to other parts of the monochrome frame. Furthermore, we design a multiframe reordering refinement to colorize video sequences robustly. Experimental results demonstrate that the proposed method achieves much better performance in video colorization than state-of-the-art methods.

***Index Terms*—** Video colorization, optical flow, color propagation, multiframe reordering

1. INTRODUCTION

Video colorization aims to add color information to a set of monochrome frames in the video sequence. In recent years, it has been an important research topic and attracted much attention in the research community due to its wide applications in video restoration of old black-and-white films, video compression *etc*.

Generally, video colorization methods can be classified into two categories based on information transferred ways among video frames. The first category transfers color information of *scribbles* drawn by users to other frames. Levin *et al.* [1] transferred scribbles of the marked frame among video frames and propagated color information of the scribbles by solving a quadratic function. Yatziv *et al.* [2] exploited geodesic distances as weights to transfer color information of the scribbles. Heu *et al.* [3] transferred scribbles to other video frames by calculating motion vectors with initialized weights. Sheng *et al.* [4] designed a video colorization method in a spatiotemporal manner to preserve temporal coherence. In these studies, the video colorization method is designed by simply transferring color information of scribbles to other frames. However, there might be some artifacts in these methods because scribbles drawn for a particular frame may not be suitable for others. Moreover, with the limited color information provided by sparse scribbles, monochrome frames can not obtain a promising colorized result similar to the ground truth. Additionally, these video colorization methods require user interaction to obtain scribbles in video sequences.

The second category of video colorization methods transfers *exemplar* color information of the colored frame to other frames. Welsh *et al.* [5] and Kawulok *et al.* [6] transferred color information to other monochrome frames based on luminance values and texture features of the pixels respectively. Jacob *et al.* [7] took color information into the other frames in the video using a color transfer technique with motion estimation. Veeravasarapu *et al.* [8] utilized motion estimation calculated by [9] to transfer color information of the whole reference frame to others. Otani *et al.* [10] transferred color information of pixels around corner feature points to neighboring frames by pyramidal implementation of the Lucas-Kanade feature tracker. Zrihem *et al.* [11] used a colored frame as a reference to colorize other frames based on a real-time patch matching algorithm called RIANN.

In order to apply the video colorization technique in the practical applications such as color video compression, where only the first color frame of a video sequence is preserved, it is obvious that there is no user interaction and the colorization results should be similar to the ground truth. However, the existing video colorization methods introduced above can not satisfy the requirements in practical applications, since they cannot colorize monochrome frames to obtain the promising colorful video due to limited color information of scribbles or artifacts caused by motion estimation and some of them even require user interaction.

In this paper, we propose an effective automatic video colorization method. We first use optical flow to estimate motion vectors for initial matching between the monochrome and colored reference frames. Then an effective colorization method is implemented based on matching results. Besides, we in-

∗Corresponding author
This work was supported by National High-tech Technology R&D Program (863 Program) of China under Grant 2014AA015205, National Natural Science Foundation of China under contract No. 61472011 and Beijing Natural Science Foundation under contract No.4142021.

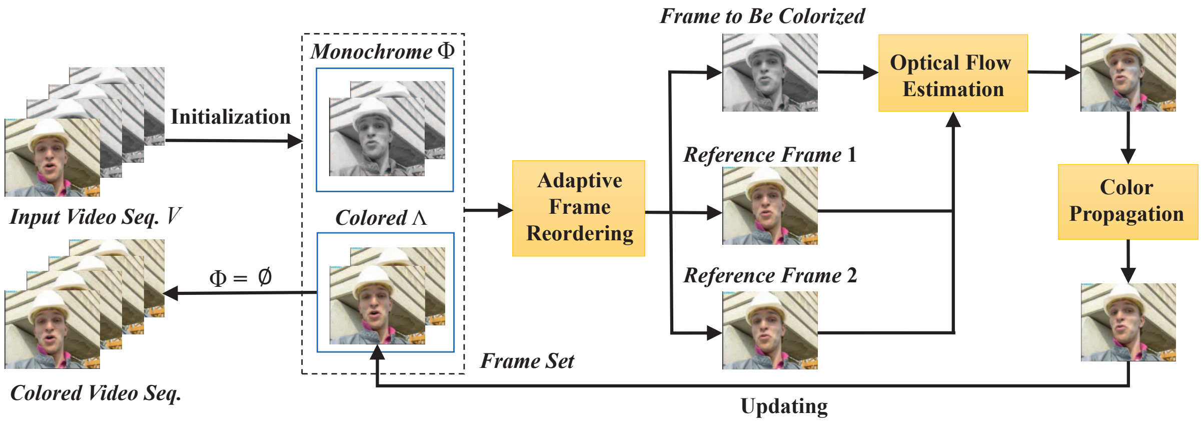


Fig. 1. Framework of the proposed adaptive frame reordering based automatic video colorization method.

troduce a novel adaptive multi-frame reordering method to obtain the robust colorization results of the whole video sequence.

The rest of this article is organized as follows. Sec.2 describes the proposed video colorization method. Experimental results are shown in Sec.3 and concluding remarks are given in Sec.4.

2. PROPOSED VIDEO COLORIZATION METHOD

Given a video sequence V, its colored part is denoted as Λ, and the monochrome part is denoted as $\Phi = V - \Lambda$ as shown in Fig.1. Instead of colorizing frames in sequential order, we iteratively calculate the similarity distance between the colored reference frame and monochrome frames, in order to choose a candidate frame to be colorized and simultaneously select the reference frames of it. Since only the 1^{st} frame has color information at first, there is only one frame (*i.e.* the 1^{st} frame) in Λ. As the colorization process proceeds, more colorized frames will be added to Λ. The proposed method is designed in the YC_bC_r color space. Let $Y_\mathbf{p}$ and $\mathbf{C_p} = (C_{b_\mathbf{p}}, C_{r_\mathbf{p}})$ denote the luminance and chrominance vector of pixel $\mathbf{p}(x, y)$, respectively.

2.1. Optical Flow Estimation

In order to make good use of the spatial similarity of the video sequence, we utilize a state-of-the-art semi-dense optical flow method [12] to estimate the motion vector field between video frames based on their grey information.

Given the reference frame $\mathbf{R}$ and the target frame $\mathbf{I}$, discrete labels $l_\mathbf{r} \in \{1, \dots, L\}$ representing the pixel motion between $\mathbf{R}$ and $\mathbf{I}$ are obtained by minimizing the following energy function,

$$E(\mathbf{L}) = \lambda \sum_{\mathbf{r}\in\mathbf{R}} \varphi_\mathbf{r}(l_\mathbf{r}) + \sum_{\mathbf{r}\sim\mathbf{r}'} \psi_{\mathbf{r},\mathbf{r}'}(l_\mathbf{r}, l_{\mathbf{r}'}), \tag{1}$$

where $\mathbf{L} = \{l_\mathbf{r} | \mathbf{r} \in R\}$ represents the set of labels associated to pixels in frame $\mathbf{R}$. $\mathbf{r} \sim \mathbf{r}'$ indicates that $\mathbf{r}$ and $\mathbf{r}'$ are neighboring pixels in the four-connected neighbors position. $\varphi_\mathbf{r}(l_\mathbf{r})$ is a data term measuring data fidelity based labels associated to the pixels and $\psi_{\mathbf{r},\mathbf{r}'}(l_\mathbf{r}, l_{\mathbf{r}'})$ is the smooth term for smooth flow fields. λ is a weighting parameter to determine the relative importance of these two terms.

2.2. Inter-Frame Color Propagation

According to the optical flow estimation results computed in Sec.2.1, we can match most pixels of the reference frame $\mathbf{R}$ to pixels in the target frame $\mathbf{I}$.

For colorizing the monochrome frame, we first initialize the monochrome frame based on matching results. Assuming that Γ is the set of colored reference frames, which includes frame $\mathbf{R}$. For each pixel $\mathbf{r} \in \Gamma$, its matched pixel in the monochrome frame $\mathbf{I}$ is defined as $\phi(\mathbf{r}, l_\mathbf{r})$, where $l_\mathbf{r}$ is the motion label of $\mathbf{r}$. Then for each pixel $\mathbf{p}$ in frame $\mathbf{I}$, its accuracy $a^0_\mathbf{p}$ is initialized as:

$$a^0_\mathbf{p} = \begin{cases} \frac{1}{|\Delta_\mathbf{p}|} \cdot \sum\limits_{\mathbf{r}\in\Delta_\mathbf{p}} e^{-|Y_\mathbf{r} - Y_\mathbf{p}|}, & |\Delta_\mathbf{p}| \neq 0, \\ 0, & |\Delta_\mathbf{p}| = 0, \end{cases} \tag{2}$$

where the set $\Delta_\mathbf{p}$ is denoted as $\{\mathbf{r} \in \Gamma | \phi(\mathbf{r}, l_\mathbf{r}) = \mathbf{p}\}$ and $|\Delta_\mathbf{p}|$ represents the number of pixels in it, and $Y_\mathbf{r}$, $Y_\mathbf{p}$ are luminance value at the pixel $\mathbf{r}$ and $\mathbf{p}$. We initialize the accuracy based on luminance differences so that it can be more robust to matching errors.

Then, the color of $\mathbf{p}$ is calculated by,

$$\mathbf{C}^0_\mathbf{p} = \begin{cases} \frac{\sum\limits_{\mathbf{r}\in\Delta_\mathbf{p}} \mathbf{C_r} \cdot e^{-|Y_\mathbf{r} - Y_\mathbf{p}|}}{\sum\limits_{\mathbf{r}\in\Delta_\mathbf{p}} e^{-|Y_\mathbf{r} - Y_\mathbf{p}|}}, & |\Delta_\mathbf{p}| \neq 0, \\ 0, & |\Delta_\mathbf{p}| = 0. \end{cases} \tag{3}$$

Here, $\mathbf{C}^0_\mathbf{p} = \mathbf{C_r}$, when there is one single reference frame matched with frame $\mathbf{I}$.

After the initialization, we update accuracies and color iteratively, In the k^{th} iteration, if $a^{k-1}_\mathbf{p} = 0$, we update the accuracy of $\mathbf{p}$ by the colored pixels in its 4-connect neighbors:

$$a^k_\mathbf{p} = \frac{1}{\pi} \cdot \sum_{\mathbf{q}\sim\mathbf{p}} a^{k-1}_\mathbf{q} \cdot e^{-|Y_\mathbf{q} - Y_\mathbf{p}|}, \ \mathbf{q} \in \Omega, \tag{4}$$

where Ω is the set of colored pixels in the target frame $\mathbf{I}$ and

$$\pi = \sum_{\mathbf{q} \sim \mathbf{p}} e^{-|Y_{\mathbf{p}} - Y_{\mathbf{q}}|}, \ \mathbf{q} \in \Omega. \tag{5}$$

By updating accuracies in this way, the weight of a colored pixel in neighboring pixels decrease with luminance differences, which makes our color propagation method robust.

Utilizing accuracies obtained above as the weights, we update the color of $\mathbf{p}$ by,

$$\mathbf{C}_{\mathbf{p}}^{k} = \frac{\sum\limits_{\mathbf{q} \sim \mathbf{p}} \mathbf{C}_{\mathbf{q}}^{k-1} \cdot a_{\mathbf{q}}^{k-1} \cdot e^{-|Y_{\mathbf{q}} - Y_{\mathbf{p}}|}}{\sum\limits_{\mathbf{q} \sim \mathbf{p}} a_{\mathbf{q}}^{k-1} \cdot e^{-|Y_{\mathbf{q}} - Y_{\mathbf{p}}|}}, \ \mathbf{q} \in \Omega. \tag{6}$$

For the pixel $\mathbf{q}$, which has already been colorized, $a_{\mathbf{q}}^{k} = a_{\mathbf{q}}^{k-1}$ and $\mathbf{C}_{\mathbf{q}}^{k} = \mathbf{C}_{\mathbf{q}}^{k-1}$. Following the method proposed above, we propagate color information iteratively in the frame $\mathbf{I}$ until all of the pixels are colored.

2.3. Multi-Frame and Adaptive Reordering Colorization Refinement

In Sec.2.2, a simple and direct way to colorize a set of video frames is to propagate color information among frames in-order between adjacent frames. However, as shown in Fig.2(a), this simple method might bring into artifacts. Since there is certain error for motion estimation by optical flow, the colorization would propagate the error within following video frames. To avoid these artifacts, we further improve the video colorization process by refining the multiframe prediction and colorizing order.

First, we increase the number of reference frames which are chosen form colored frames set Λ to make color propagation more robust. Once obtained more colored frames, we could add them into the reference set Γ described. That is, multiframes are utilized into predicting the motion estimation and refining the process of inter-frame color propagation.

Then, in general, more similar and small motion between two video frames, the error made by optical flow estimation and color propagation is less. Along this way, we calculate the similarity distance among frames in order to refine the order of the video frames to be colorized.

(a) In-order (b) Refinement

Fig. 2. Colorization results of the 90^{th} frame in *Foreman* by in-order method and our refinement method, respectively.

Let $\rho(\mathbf{S}, \mathbf{T})$ be the distance between frames $\mathbf{S}$ and $\mathbf{T}$,

$$\rho(\mathbf{S}, \mathbf{T}) = \frac{\sum\limits_{s \in \mathbf{S}} m(\mathbf{s}, \mathbf{T})}{|\mathbf{S}|}, \tag{7}$$

$m(\mathbf{s}, \mathbf{T})$ is the length of the motion vector of pixel $\mathbf{s}$ in $\mathbf{S}$ to $\mathbf{T}$. $|\cdot|$ represents the number of the set.

Then, we set

$$\xi(\mathbf{S}) = \frac{1}{|\Lambda|} \cdot \sum_{\mathbf{R} \in \Lambda} \rho(\mathbf{S}, \mathbf{R}), \tag{8}$$

$$\delta(\mathbf{S}) = \frac{1}{|\Phi|} \cdot \sum_{\mathbf{I} \in \Phi} \rho(\mathbf{S}, \mathbf{I}), \tag{9}$$

and choose the frame to be colorized by optimally solving:

$$\hat{\mathbf{S}} = \arg\min_{\mathbf{S}} \left\{ \xi(\mathbf{S}) - \alpha \cdot \delta(\mathbf{S}) \right\}, \tag{10}$$

where α is a weighting parameter for relative importance for these two terms. After choosing the frame $\hat{\mathbf{S}}$, we add its two nearest frames in Λ into the reference set Γ.

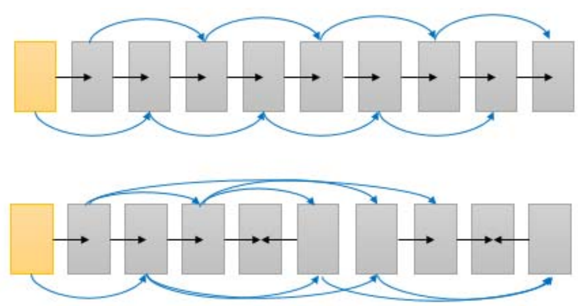


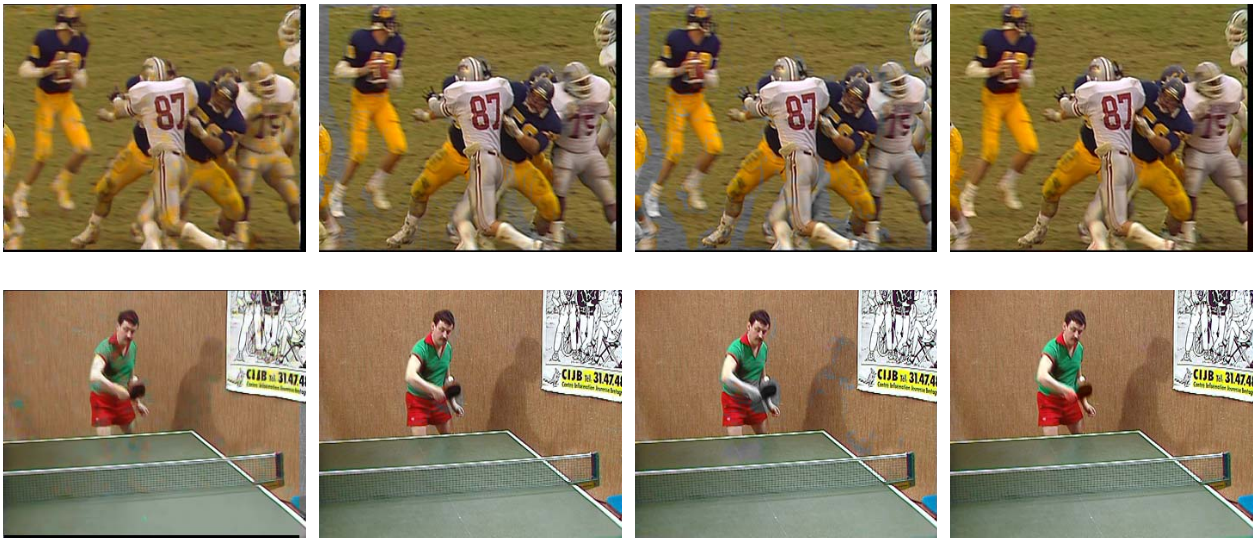
Fig. 3. New orders of two sequences generated by our reordering method.

Eq.(10) guarantees that we can colorize a frame earlier, if it is more similar to colored frames and more different from other uncolored frames. Fig.3 illustrates the different reordering for different sequences colorization. It leads to reducing the error propagation to latter processed frames.

3. EXPERIMENTAL RESULTS

The proposed method is implemented by Matlab R2014a platform. In the experiment, we set $\alpha = 1$ in Eq.(10). Our video colorization method is tested in two ways. One is to use frames randomly selected from some standard video sequences as references to color their next frames in the sequences. Another way is to use one selected frame as a reference and colorize the following few frames. PSNR is computed for objective quality evaluation by our colorization method. More experimental results can be found on our website[1].

[1]http://www.icst.pku.edu.cn/course/icb/Projects/videocolorization.html



(a) Zrihem [11] (b) Veeravasarapu [8] (c) Baseline (d) Proposed

Fig. 4. Two colorization samples from different methods. In the first row, we use 61^{st} frame in *Football* as a reference frame to colorize the 62^{nd} frame. And in the second row we use the 91^{st} frame as a reference to colorize the next frame.

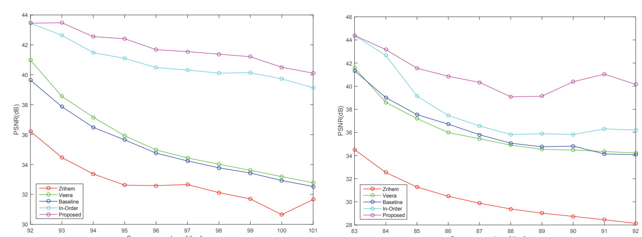
We compare our method with Zrihem's method [11], Veeravasarapu's method [8] and the baseline made by the optical flow method [12] where color information of the reference frame is directed transferred to the monochrome frame based on the matching results. Fig.4 shows some comparison samples of neighboring frames. From this figure, we can see that some regions of the results by Veeravasarapu's method and baseline have not been colored, which are caused by mistaken motion estimation. Due to mistaken patch match results, there are also some artifacts in Zrihem's method. On the opposite, there is no apparent artifact in the colorization results by the proposed method.

Table 1. PSNR(dB) comparison of some colorization results from different methods.

Sequence	Zrihem [11]	Veera [8]	Baseline	In-Order	Proposed
Silent	30.99	36.59	40.51	44.48	**44.86**
Soccer	31.33	37.11	37.33	41.42	**41.52**
Foreman	30.73	37.65	37.99	41.85	**42.45**
Tennis	30.48	33.89	34.21	36.99	**37.33**
MomDaughter	31.14	37.13	36.42	43.82	**44.33**
Average	30.99	36.47	37.29	41.71	**42.10**

In Table 1, we provide average PSNR values of results by different compared methods. In the experiment, we choose 22 frames in each sequence and preserve color information every 11 frames. Moreover, results of colorizing sequences in order by our color propagation method have also been listed to demonstrate the effectiveness of our reordering refinement. From this table, we can see that average PSNR values of video sequences by the proposed method are higher than those from other existing ones, which demonstrate that the proposed method can obtain better performance of video colorization than other existing ones.

In Fig.5, we provide PSNR values for the colorization results of two video sequences of *Foreman* and *Soccer* in the type of a line chart. New orders decided by our method are shown in Fig.3. From this figure, we can see that the proposed refinement results outperform other methods greatly, which also demonstrates that the proposed method can obtain better performance in video colorization than other existing ones.



(a) *Soccer* (b) *Foreman*

Fig. 5. Video colorization results in *Soccer* and *Foreman* sequences frame by frame.

4. CONCLUSION

In this study, we propose a new automatic video colorization method for video sequences. The optical flow estimation method is first adopted to initially match reference colored frames with monochrome frames which are to be colorized. Then we initialize monochrome frames and propagate color to uncolored regions. Furthermore, in order to avoid the artifacts from motion estimation and color propagation, we design a multiframe and adaptive reordering colorization method. Experimental results show that the proposed method can obtain the effective results of video colorization.

5. REFERENCES

[1] Anat Levin, Dani Lischinski, and Yair Weiss, "Colorization using optimization," in *Proc. ACM Int'l Conf. and Exhibition on Computer Graphics and Interactive Techniques*, August 2004, pp. 689–694.

[2] Liron Yatziv and Guillermo Sapiro, "Fast image and video colorization using chrominance blending," *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1120–1129, May 2006.

[3] Jun-Hee Heu, Dae-Young Hyun, Chang-Su Kim, and Sang-Uk Lee, "Image and video colorization based on prioritized source propagation," in *Proc. IEEE Int'l Conf. Image Processing*, November 2009, pp. 465–668.

[4] Bin Sheng, Hanqiu Sun, Marcus Magnor, and Ping Li, "Video colorization using parallel optimization in feature space," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 3, pp. 407–417, March 2014.

[5] Tomihisa Welsh, Michael Ashikhmin, and Michael Ashikhmin, "Transferring color to greyscale images," in *Proc. ACM Int'l Conf. and Exhibition on Computer Graphics and Interactive Techniques*, June 2002, pp. 277–280.

[6] Kawulok, Michal, Jolanta Kawulok, and Bogdan Smolka, "Discriminative textural features for image and video colorization," *IEICE Trans. Commun.*, , no. 7, pp. 1722–1730, 2012.

[7] Vivek George Jacob and Sumana Gupta, "Colorization of grayscale images and videos using a semiautomatic approach," in *Proc. IEEE Int'l Conf. Image Processing*, November 2009, pp. 1653–1656.

[8] V S Rao Veeravasarapu and Jayanthi Sivaswamy, "Fast and fully automated video colorization," in *Proc. SPIE Int'l Conf. on Signal Processing and Communications*, July 2012, pp. 1–5.

[9] Thomas Brox, Andres Bruhn, Nils Papenberg, and Joachim Weickert, "High accuracy optical flow estimation based on a theory for warping," in *Proc. European Conference on Computer Vision*, May 2004, pp. 25–36.

[10] Mayu Otani and Hirohisa Hioki, "Video colorization based on optical flow and edge-oriented color propagation," in *Proc. SPIE*, March 2014, pp. 902002–902002–9.

[11] Nir Ben-Zrihem and Lihi Zelnik-Manor, "Approximate nearest neighbor fields in video," in *Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition*, June 2015, pp. 5233–5242.

[12] Moritz Menze, Christian Heipke, and Andreas Geiger, "Discrete optimization for optical flow," in *Proc. German Conference on Pattern Recognition*, October 2015.