Journal of Imaging Science and Technology® 50(3): 243–250, 2006.
© Society for Imaging Science and Technology 2006

Colorization Algorithm for Monochrome Video by Sowing Color Seeds

Takahiko Horiuchi▲ and Hiroaki Kotera▲
Department of Information and Image Sciences, Faculty of Engineering, Chiba University, 1-33 Yayoi-cho, Inage-ku, Chiba 263-8522, Japan
E-mail: horiuchi@faculty.chiba-u.jp

Abstract. *Recently, the study of the colorization algorithm for monochrome still images has prospered. Considering actual applications, an extension to monochrome video becomes important. This paper proposes two colorization algorithms for monochrome image sequence without scene changing in video. In our approach, key frames in the image sequence are firstly colorized by a conventional colorization technique for still image. In the technique, a small number of color seeds are sown on the monochrome key frame, and color seeds are propagated spatially to remaining monochrome pixels. Then, each color in the colorized frame propagates to the next monochrome frame by extracting a displacement vector for each pixel between two frames. In this paper, the displacement vector is calculated by a simple block-matching algorithm, and color propagation to temporary direction is performed. So, in the proposed algorithm, video colorization can be realized by setting only some color seeds on key frames. Experimental results suggest a success of colorization for video by setting key frames for each 10 frames.*
© 2006 Society for Imaging Science and Technology.
[DOI: 10.2352/J.ImagingSci.Technol.(2006)50:3(243)]

INTRODUCTION

Colorization is a computerized process that adds color to a black and white print, movie, and TV program. Hand coloring was practiced in motion pictures in the early 1900's. Wilson Markle introduced a computer-assisted colorization process in 1970.[1] The demand of adding color to monochrome images such as old films, BW movies and BW photos has been increasing. For example, in the amusement field, many movies and video clips have been colorized by human labor, and many monochrome images have been distributed as vivid color images. In other fields such as archaeology dealing with historical monochrome data and security dealing with monochrome images by a crime prevention camera, we can imagine easily that colorization techniques are useful.

Mapping between intensity or luminance and color is not unique, and colorization is an ill-posed problem. Due to these ambiguous, human interaction usually plays a large role in the colorization process. The correspondence between a color and a luminance value is determined through common sense (green for grass, blue for the ocean) or by investigation. Even in the case of pseudo-coloring,[2] where the mapping of luminance values to an RGB vector is automatic, the choice of the color-map is purely subjective. Since there are a few industrial software products, those technical algorithms are generally not available. However, by operating those software products, it turns out that humans must meticulously hand-color each of the individual image subjectivity. There also exist a few patents for colorization.[3,4] However, those approaches depend on heavy human operation.

Recently, a few effective colorization algorithms have been proposed. Horiuchi et al. proposed a coloring method by sowing color seeds and propagating colors spatially to the remaining pixels in the still image.[5,6] In Refs. 7–9, a fast algorithm for Horiuchi's technique was proposed. Those techniques work well by considering spatial continuity. Welsh et al., inspired by work in Ref. 11, proposed another idea by transferring color from a reference color image.[10] A fast algorithm for Welsh's technique was also developed.[12] Though a user must prepare a reference image containing the desired colors with similar textures, the technique can transfer a *mood* in the reference image. In 2004, Levin et al. solved an optimization problem that minimizes a quadratic cost function of the difference of color between a pixel and neighborhood colors.[13] In the algorithm, a user indicates colors with a few color scribbles, and they are propagated to remaining pixels.

Those approaches work well for colorizing monochrome still images. However, there are many problems with applying them to moving pictures. In Horiuchi's algorithm, a user has to set color seeds on all frames for colorizing image sequence. This is very heavy human operation. Welsh's algorithm does not use temporal information for colorization to image sequences. So, it works well only for the limited image. Levin's algorithm can solve the problem by considering corresponding pixels in the next frame as neighbors. However, it is required to annotate image with complicated scribbles for every few frames.[14]

This study aims to develop a colorization algorithm for monochrome video with easy interactive processing. As the first step, this paper proposes two colorization algorithms for monochrome video, which is defined as continuous frames without scene changing. In our algorithm, only a small number of color seeds are set on key frames for each 10 monochrome frames. Then the remaining pixels in all

▲IS&T Member

Received Jan. 26, 2005; accepted for publication Jun. 16, 2005.
1062-3701/2006/50(3)/243/8/$20.00.

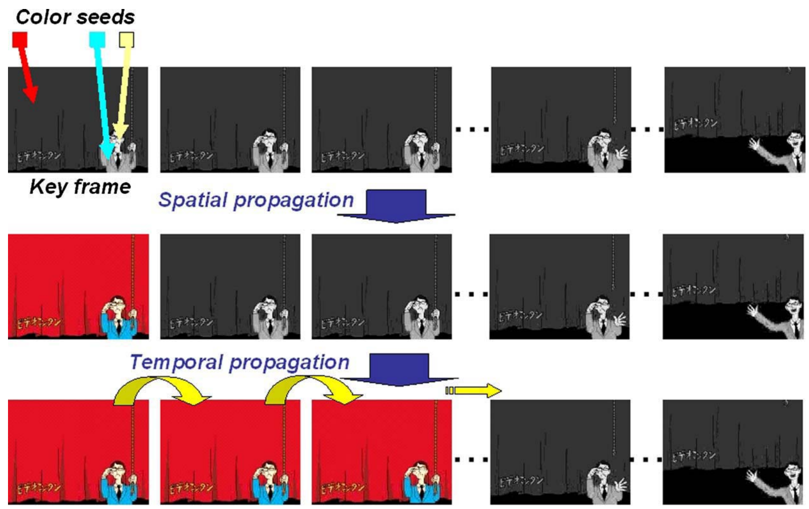


Figure 1. Colorization process of the proposed algorithm. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

frames are colorized automatically by considering both space and temporal relations.

PROBLEM SETTING

Consider an image sequence including N frames and a luminance value at (x,y) in nth frame is expressed as $f^{(n)} \times (x,y): 1 \leqslant n \leqslant N$. Here, it is assumed that the image sequence must be composed of continuous scenes without scene changing. Scene changing problems may be solved by backward colorization in the future. Each frame is $M_1 \times M_2$ monochrome image. Let $\boldsymbol{I}^{(n)}(x,y)$ be a correct RGB color vector $(R(x,y), G(x,y), B(x,y))$ in nth frame. Each element is quantized by L_1 bits. It is known that a color vector $\boldsymbol{I}^{(n)}(x,y)$ and the luminance value $f^{(n)}(x,y)$ have the following relation:

$$f^{(n)} = [0.299, 0.587, 0.114]\boldsymbol{I}^{(n)T}, \tag{1}$$

where symbol T expresses transposition of a matrix. By Eq. (1), $f^{(n)}(x,y)$ can be given uniquely from $\boldsymbol{I}^{(n)}(x,y)$. Meanwhile, $\boldsymbol{I}^{(n)}(x,y)$ cannot be given uniquely from $f^{(n)}(x,y)$. Let $\{\boldsymbol{I}_i^{(n)}(x,y)\}$ be a set of solutions, which is called *color candidates*, for a luminance value $f^{(n)}(x,y)$ which satisfies Eq. (1). The main problem of colorization is how to determine the most suitable color $\boldsymbol{I}^{*(n)}(x,y)$ out of the color candidates $\{\boldsymbol{I}_i^{(n)}(x,y)\}$ for each monochrome pixel $f^{(n)}(x,y)$.

Let n^* be nth frame which is chosen as key frame. In our algorithm, some color seeds $\boldsymbol{I}^{*(n*)}(x,y)$ are sown on (x,y) in the key frame beforehand, and the color will be propagated spatially in the remaining monochrome pixels in the key frame. Then the colorization in the key frame is completed. Next, the color will be propagated to the next frame temporally. So, the problem can be divided into two parts.

(PROBLEM 1) Colorization in the key frame

We have to determine how to propagate the color spatially from color seeds.

(PROBLEM 2) Temporal color propagation

We have to determine how to propagate the color temporally from the previous colored frame. The concept of the proposed idea is shown in Fig. 1 (Available in color as Supplemental Material on the IS&T website, www.imaging.org).

KEY FRAME COLORIZATION

In this section, a key frame colorization algorithm is described. Easy interactive process is very important for colorization. In this paper, we use a colorization algorithm in Ref. 7 for key frame colorization, which gives only small number of color seeds on a monochrome image. The following is the outline of the algorithm.

(STEP 1) Sowing color seed pixels

Color seeds are set on a monochrome key frame. Here, the key frame is assumed as the first frame in the image sequence. Many seeds can be set in parallel, but the behavior of one seed pixel is explained here. Sowing a color seed means to choose the optimum color $\boldsymbol{I}^{*(1)}(x,y)$ out of the color candidates $\{\boldsymbol{I}_i^{(1)}(x,y)\}$ for each monochrome pixel $f^{(1)}(x,y)$ by a user. Figure 2(a) (Available in color as Supplemental Material on the IS&T website, www.imaging.org) shows a given seed pixel.

(STEP 2) Initial propagation

The color seed $\boldsymbol{I}^{*(1)}(x,y)$ propagates to adjacent four-connected pixels. A few propagation algorithms have been proposed by the authors.[7–9]. For example, the optimum color in a pixel $(x-1,y)$ is determined by the following Eq. (2) out of the color candidates $\{\boldsymbol{I}_i^{(1)}(x-1,y)\}$ in Ref. 7.

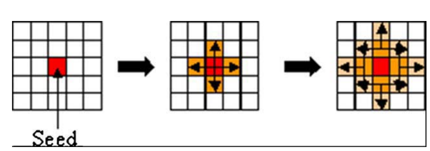


Figure 2. Spatial propagation process: (a) A seed, (b) Initial propagation, (c) 2nd propagation. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

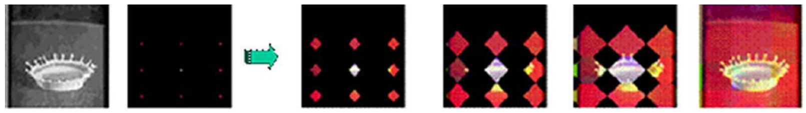


Figure 3. Colorization process for a key frame with 9 seeds. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

$$I^{*(1)}(x-1,y) = \{I_i^{(1)}(x-1,y) \big| \min_i \|I_i^{(1)}(x-1,y) - I^{*(1)}(x,y)\|\}. \quad (2)$$

Here, symbol $\|\cdot\|$ shows the Euclidean distance in RGB color space. In our experiment, more accurate colorized results can be obtained by using the color difference as distance in CIELAB color space.[8] However, it requires heavy calculation time for converting color space. The selection of color space may depend on the application. Figure 2(b) shows an image after the initial propagation.

(STEP 3) Iterative propagation

The propagation process proceeds to the next four connected components. In the case that colors are propagated from many adjacent pixels at the same time as shown in Fig. 2(c), a suitable color is selected from color candidates to minimize the average of color differences for each adjacent propagated pixel. In Ref. 8, a partitioning algorithm was proposed for preventing error propagation at edge. For complex image, the partition may work well.

(STEP 4) End conditions

(STEP 3) is continued until all pixels are colorized. If a propagation wave conflicts with other waves, the propagation process stops at the point.

Figure 3 (Available in color as Supplemental Material on the IS&T website, www.imaging.org) shows an example of propagation behavior for an actual image. Figure 3(a) shows a monochrome image. Nine color seeds were sown in the shape of lattice as shown in Fig. 3(b). Figure 3(c) to Fig. 3(e) shows the process of colorization. Finally, we can get the colorized image as shown in Fig. 3(f). Figure 4 (Available in color as Supplemental Material on the IS&T website, www.imaging.org) shows other examples of still image colorization. By sowing seven color seeds on the monochrome image as shown in Fig. 4(a), the colorized image in Fig. 4(b) can be obtained. Of course, the colorized result depends on the location and color of seeds. Figure 4(c) is a colorized result by sowing a blue color seed on the clothes. Figure 4(d) shows that the lowest seeds were sown at the different part from Fig. 4(a). In this case, possible "failure" results are obtained as shown in Fig. 4(e). Note also as we comment with this figure that a common scenario for this application includes user interaction and thereby there is no "real failure" since the user can quickly correct and improve the results if needed. Another solution is to adopt a partitioning algorithm in Ref. 8. By using the algorithm, colorization will be performed in each region which is segmented at edges.

TEMPORAL COLOR PROPAGATION

The second problem in this paper is how to propagate the color from the colored frame to the next monochrome frame. The proposed algorithm consists of two steps. The first step is displacement vector extraction and the second step is color estimation. Each step is described in the following subsections.

(STEP 1) Displacement Vector Extraction

In order to propagate colors in a colorized frame, displacement vector between adjacent frames is calculated for each pixel. In this paper, we use a typical block-matching algorithm[15] in the optical flow scheme. The optical flow of an image sequence is a set of vector fields, relating each frame to the next. Each vector field represents the apparent displacement of each pixel from frame to frame. If we assume the pixels conserve their luminance value, we arrive at the "brightness conservation equation,"

$$f^{(n)}(x,y) = f^{(n-dn)}(x+dx, y+dy), \quad (3)$$

where (dx,dy) is the displacement vector for the pixel at coordinate (x,y) and n and dn are the frame and temporal displacements of the image sequence.

The obvious solution to Eq. (3) is to use template-based search strategies. A template of a certain size around each pixel is created and the best match is searched for in the next frame. The best match is usually found using correlation, sum of absolute difference or sum of squared difference metrics. In this paper, we use the following criterion:

$$\sum_p \sum_q |f^{(n)}(x+p,y+q) - f^{(n-1)}(x+dx+p,y+dy+q)|, \quad (4)$$

where (p,q) shows an coordinate in template block. A vector (dx,dy) with the minimum Eq. (4) is determined as the displacement vector for $f^{(n)}(x,y)$. Such a search strategy is computationally costly and generally does not represent subpixel displacements.

(STEP 2) Color Estimation

By the algorithm in the previous subsection, a corresponding color pixel $I^{*(n-1)}(x+dx,y+dy)$ in a colorized $(n-1)$th frame is determined for each pixel $f^{(n)}(x,y)$ in nth monochrome frame. In this subsection, we propose two estimation algorithms.

(Algorithm 1)

$$I^{*(n)}(x,y) = I^{*(n-1)}(x+dx,y+dy). \quad (5)$$

Equation (5) means that each pixel (x,y) in nth frame is colorized by the corresponding color $I^{*(n-1)}(x+dx,y+dy)$ in $(n-1)$th frame.

(Algorithm 2)

$I^{*(n)}(x,y) = I_i^{(n)}(x,y)$ which minimizes the difference be-

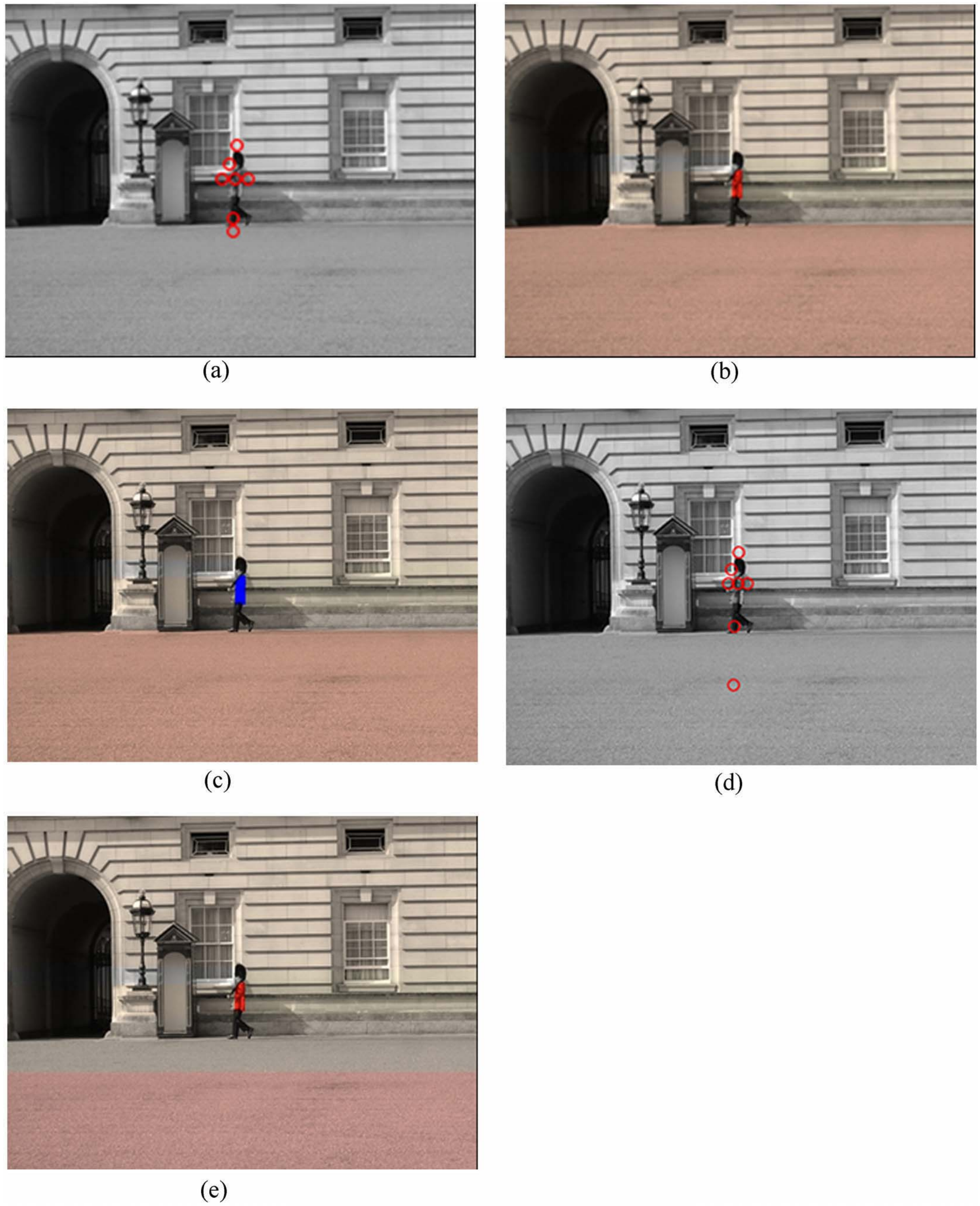


Figure 4. Differences of colorization by changing the location and color of seeds. (a) A monochrome image. Seven color seeds were sown at the center of red circles. (b) Colorized result from (a). (c) Colorized result by sowing a blue color at the center seed instead of red. (d) The location of the lowest seed is different from (a). (e) Colorized result from (d). (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

tween a *color candidate* $I_i^{(n)}(x,y)$ and $I^{*(n-1)}(x+dx,y+dy)$ at corresponding pixel in the previous frame:

$$J(I_i^{(n)}) = \|I^{*(n-1)}(x+dx,y+dy) - I_i^{(n)}(x,y)\| \rightarrow \min. \quad (6)$$

Equation (6) means that each pixel (x,y) in nth frame is colorized by selecting the most suitable color $I_i^{(n)}(x,y)$ from color candidates $\{I_i^{(n)}(x,y)\}$ for the luminance value $f^{(n)}(x,y)$. Here, the color $I_i^{(n)}(x,y)$ has the minimum RGB difference between the corresponding color $I^{*(n-1)}(x+dx,y+dy)$ in $(n-1)$th frame and *color candidates* $\{I_i^{(n)}(x,y)\}$. Algorithm 2 has a property to keep the luminance value for the colorized image.

EXPERIMENTS

In order to verify the proposed algorithms, we carried out colorization for monochrome image sequences. We have collected two kinds of images which are animated images and

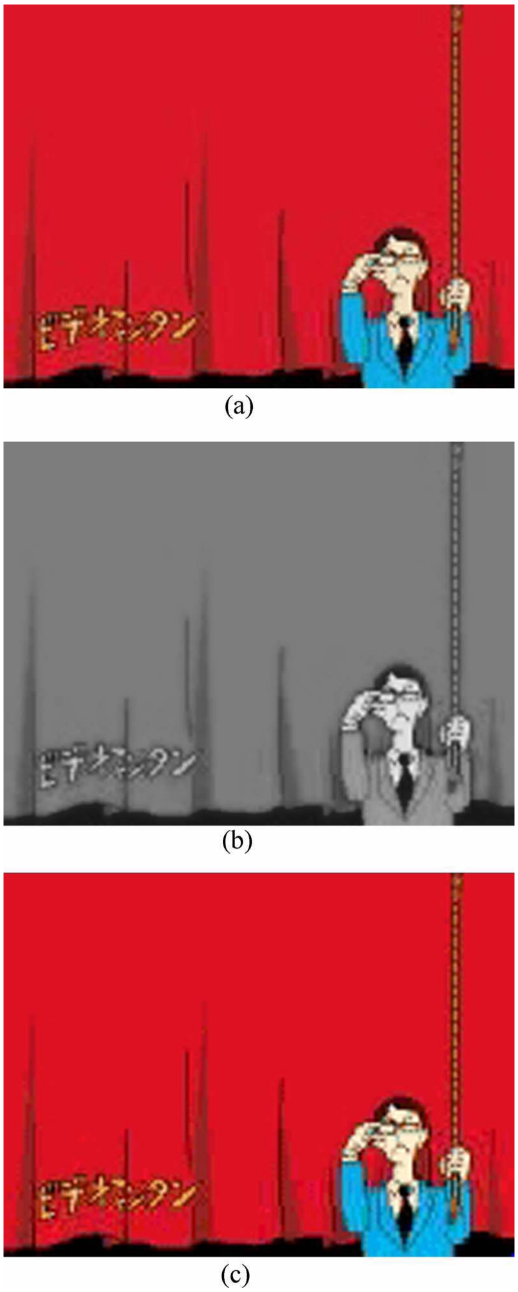


Figure 5. A result of key frame colorization. (a) Original color image (160×120) (Frame #1). (b) Monochrome image. (c) Colorized image by 20 color seeds. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

natural images. All original images have color and they are converted into monochrome images by using Eq. (1). Only the first frame is given as original color and other monochrome frames are colorized in order by the proposed methods. We set the template block size in Eq. (4) as 7×7 empirically.

Colorization for Animated Images

Since animated images have little change of the color between frames, it seems that displacement vectors can be detected stably. At first, we show the result of the key frame

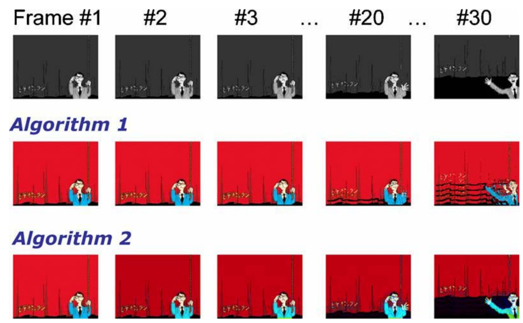


Figure 6. Colorized frames from frame Nos. 1–30. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

colorization. Figure 5(a) shows the original color frame with 160×120 pixels and Fig. 5(b) shows its monochrome image. Figure 5(c) shows the colorized results when 20 color seeds are sown on Fig. 5(b). Color seeds were sown on different color regions by user. PSNR between Figs. 5(a) and 5(c) is 35[dB]. PSNR is calculated by the following definition:

$$(\mathrm{PSNR}) = 20\log_{10}\frac{2^{L_1}-1}{\mathrm{RMSE}} = 10\log_{10}\frac{(2^{L_1}-1)^2}{\mathrm{MSE}} = 10\log_{10}\frac{3M_1M_2(2^{L_1}-1)^2}{\sum_{x,y}|I-I^*|^2}, \quad (7)$$

where RMSE means the root mean square error between an original color image and its colorized image, and MSE means the mean square error between them.

Next, we verify the performance of the temporal colorization. Figure 6 (Available in color as Supplemental Material on the IS&T website, www.imaging.org) (upper) shows a partial monochrome image sequence. The frame rate is 16[frames/s]. We used 48 frames for the test. Figure 6 (middle) shows the colorized results by Algorithm 1. Algorithm 1 kept the color. But, the structure was broken by accumulated error of optical flow detection. Meanwhile, in Algorithm 2 as shown in Fig. 6 (lower), although there were some mis-colorized parts, better results were obtained. The significant difference is the colorization of the curtain. In Algorithm 2, the thirty frames show a red curtain being

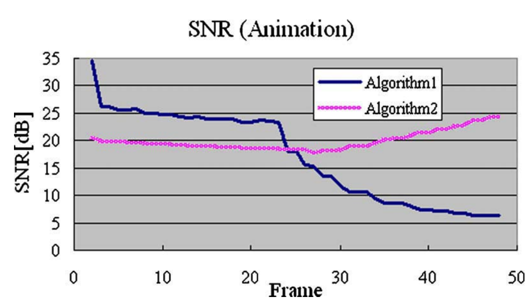


Figure 7. PSNR for each frame (animation). (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

(a)

(b)

(c)

Figure 8. A result of key frame colorization. (a) Original color image (320×240) (Frame #1). (b) Monochrome image (seeds were sown at the center of circles). (c) Colorized image by 30 color seeds. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

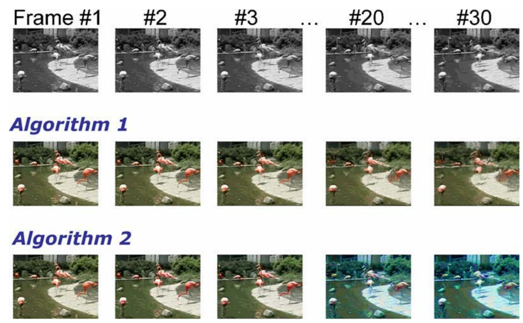
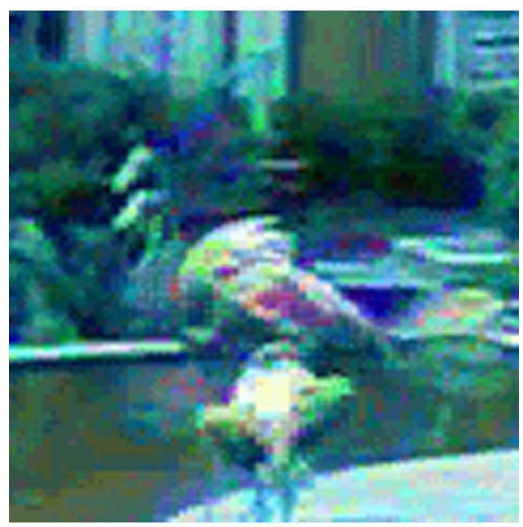


Figure 9. Colorized frames from frame #1 to #30. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

(a)



(b)

Figure 10. Enlarged colorized images for frame #30. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

raised behind a man in a blue suit. Since the detected displacement vectors are the same, the difference depends on the proposed color estimation algorithm. Algorithm 2 colorizes the image while keeping the luminance values, dark luminance can be colorized by a dark color.

In order to verify the results objectively, PSNR graph between an original color image and the colorized one is shown in Fig. 7 (Available in color as Supplemental Material on the IS&T website, www.imaging.org). As shown in Fig. 7, Algorithm 1 works well until the 25th frame. From the 25th frame, the red curtain goes up. Since the black part is increasing, PSNR of Algorithm 2 is going up from the 25th frame. The difference in coloring to a motion of curtain becomes the difference of the PSNR.

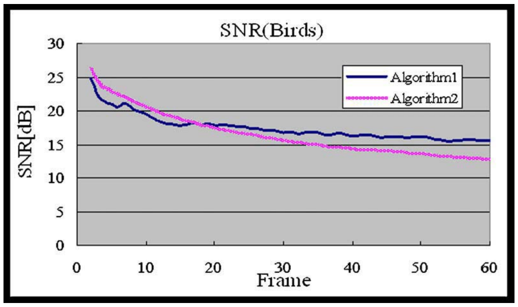


Figure 11. PSNR for each frame (birds). (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

Colorization for Natural Scenes

The proposed algorithms were also applied to the several natural scenes. Figure 8 (Available in color as Supplemental Material on the IS&T website, www.imaging.org) shows a colorized result for the first frame of a test data "bird." Figure 8(a) shows the original color image with the size of 320×240. Figure 8(b) shows its monochrome image and Fig. 8(c) shows the colorized results by sowing 30 arbitrary color seeds. Since the natural image includes more complicated regions than the animated image, we set 30 color seeds. PSNR between Figs. 8(a) and 8(c) is 28[dB].

The performance of the temporal sequence is verified using the image sequence in Fig. 9(a) (Available in color as Supplemental Material on the IS&T website, www.imaging.org). Frame rate is 12[frames/s]. We used 60 frames for the test. Figures 9(b) and 9(c) show the colorized results. Figure 10 (Available in color as Supplemental Material on the IS&T website, www.imaging.org) shows a part of an enlarged image of colorized frame #30. In the case of Algorithm 1, the preservability of a color was recognized as the same as the experimental results. But, the structure was also broken as shown in Fig. 10(a). On the other hand, in the case of Algorithm 2, the preservability of a structure was accepted. However, faded color appeared more remarkably than colorized results for animated images as shown in Fig. 10(b).

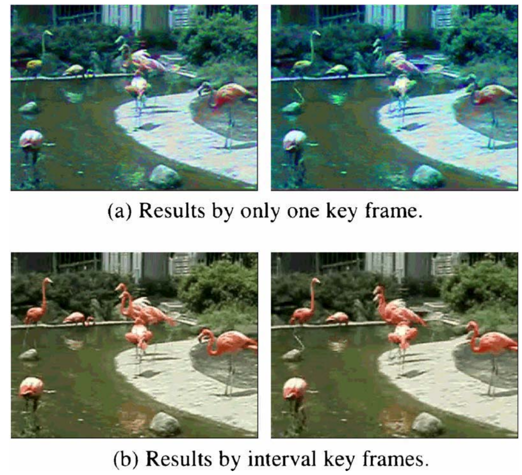


Figure 13. Colorized results. (Left) Frame #20, (Right) frame #30. (a) Results by only one key frame. (b) Results by interval key frames. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

PSNR graph is shown in Fig. 11 (Available in color as Supplemental Material on the IS&T website, www.imaging.org). Although both PSNR were decreased gently, the slope of Algorithm 2 is more sudden. The same results were obtained for other test images. These results are in agreement with a subjective evaluation.

In the above experiments, only the first frame was colorized as key frame beforehand. In order to avoid the influ-

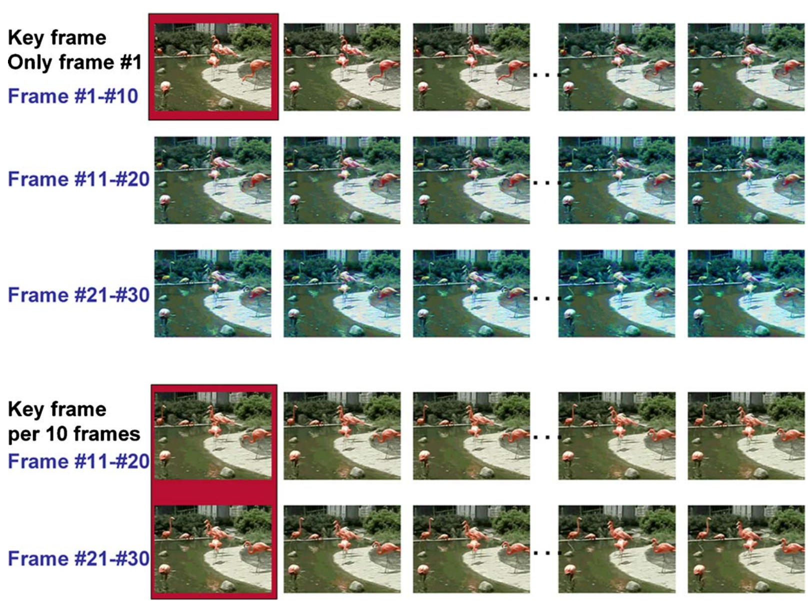


Figure 12. Colorization with only one key frame (upper) and with key frames per 10 frames (lower). (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

Figure 14. Colorized frames from frame Nos. #1–#30. (Available in color as Supplemental Material on the IS&T website, www.imaging.org)

ence of an accumulation error, we should increase the number of key frames. We also conducted the experiment which set up the key frame at the fixed interval. If the interval is too large, boundary frames were recognized. In our experiments, a better interval was 10 [frame]. Figure 12 (Available in color as Supplemental Material on the IS&T website, www.imaging.org) shows the comparison results. The upper part in Fig. 12 shows the colorized results using Algorithm 2 with only one key frame. The lower part in Fig. 12 shows the results with 3 key frames that are frames #1, #11, and #21 as shown by red squares. In order to confirm the effectiveness, the colorized results for frames #20 and #30 are shown in Fig. 13 (Available in color as Supplemental Material on the IS&T website, www.imaging.org).

Colorized results for other test data are shown in Fig. 14 (Available in color as Supplemental Material on the IS&T website, www.imaging.org). The experiments were performed by Algorithm 2, and key frames were set for each 10 frames. We obtained good results just by sowing a few color seeds on key frames.

CONCLUSION

This paper proposed algorithms for colorizing monochrome image sequences in video. In our algorithms, key frames are colorized by sowing color seeds using a conventional algorithm for still images, and colors are propagated to remaining frames by a block-matching algorithm. For color estimation, we proposed two kinds of algorithms. Experimental results showed that one algorithm can keep color and another algorithm can keep the structure of scenes. In order to avoid error accumulation, we confirmed that setting key frames for each 10 frames are effective.

It is a challenging problem to decide the number of most suitable color seeds and the number of frames theoretically. In this paper, we decided them through the interactive process, but have to find a rule for deciding them. Moreover, error of displacement vector detection accumulates from frame to frame. In actual applications, flicker causes a detection error of corresponding pixels. Therefore, we must develop a more accurate and robust displacement vector detection algorithm. Moreover, it may be necessary to use the information on a frame bi-directionally to the temporary axis.

REFERENCES

[1] G. Burns, *Colorization*, Museum of Broadcast Communication: Encyclopedia of Television (http://www.museum.tv/archives/etv/C/htmlC/colorization/colorization.htm).

[2] R. C. Gonzales and P. Wintz, *Digital Image Processing* (Addison-Wesley, Reading, MA, 1987).

[3] M. Wilson and H. Brian, "Coloring a black and white signal using motion detection", Canadian Patent, No. CA 01291260 (1991).

[4] P. V. Roy, "Designing, drawing, and colorizing generated images by computer", U.S. Patent, No. 5,831,633 (1998).

[5] T. Horiuchi, "Estimation of color for gray-level image by probabilistic relaxation", *Proc. IEEE International Conference on Pattern Recognition* (IEEE, Piscataway, NJ, 2002) pp. 867–870.

[6] T. Horiuchi, "Colorization algorithm using probabilistic relaxation", *Proc. IEEE International Conference on Pattern Recognition* (IEEE, Piscataway, NJ, 2002).

[7] T. Horiuchi and S. Hirano, "Colorization algorithm for grayscale image by propagating seed pixels", *Proc. IEEE International Conference on Image Processing* (IEEE, Piscataway, NJ, 2003) pp. 457–460.

[8] T. Takahama, T. Horiuchi, and H. Kotera, "Improvement on colorization accuracy by partitioning algorithm in CIELAB color space", *Lecture Notes in Computer Science, Advances in Multimedia Information Processing - PCM2004, LNCS3332, Part II* (Springer, Heidelberg, Germany, 2004), pp. 794–801.

[9] T. Horiuchi and H. Kotera, "Colorization for monochrome image based on diffuse-only reflection model", *Proc. Congress of the International Colour Association (AIC 05), Part I*, (2005), pp. 353–356.

[10] T. Welsh, M. Ashikhmin, and K. Mueller, "Transferring color to grayscale image", Proc. ACM SIGGRAPH 2002 **20(3)**, 277–280 (2002).

[11] E. Reinhard, M. Ashikhmin, B. Gooch, and P. Shirley, "Color transfer between images", IEEE Comput. Graphics Appl. **21**, 34–40, (2001).

[12] G. D. Blasi and D.-R. Recupero, "Fast colorization of gray images", *Proc. Eurographics Italian Chapter 2003*, Milan, Italy (2003).

[13] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization", ACM Trans. Graphics **23**(3), 689–694 (2004).

[14] http://www.cs.huji.ac.il/~yweiss/Colorization/index.html#video.

[15] J. Little and A. Verri, "Analysis of differential and matching methods for optical flow", *Proc. IEEE Workshop on Visual Motion* (IEEE, Piscataway, NJ, 1989) pp. 173–180.