COLORIZATION OF GRAYSCALE IMAGES AND VIDEOS USING A SEMI-AUTOMATIC APPROACH

Vivek George Jacob and Sumana Gupta

Department of Electrical Engineering, Indian Institute of Technology Kanpur

ABSTRACT

Colorization is a computer-aided process of adding color to a grayscale image or video. The task of colorizing a grayscale image involves assigning three dimensional (RGB) pixel values to an image which varies along only one dimension (luminance or intensity). Since different colors may have the same luminance value but vary in hue or saturation, mapping between intensity and color is not unique, and colorization is ambiguous in nature, requiring some amount of human interaction or external information. In this paper we propose a semi-automatic process for colorization where the user indicates how each region should be colored by putting the desired color marker in the interior of the region. The algorithm based on the position and color of the markers, segments the image and colors it. In order to colorize videos, few reference frames are chosen manually from a set of automatically generated key frames and colorized using the above marker approach and their chrominance information is then transferred to the other frames in the video using a color transfer technique making use of motion estimation. The colorization results obtained are visually very good. In addition the amount of manual intervention is reduced since the user only has to apply color markers on few selected reference frames and the proposed algorithm colors the entire video sequence.

Index Terms— Colorization, Watershed segmentation, Rain simulation technique, Motion estimation, Key frames

1. INTRODUCTION

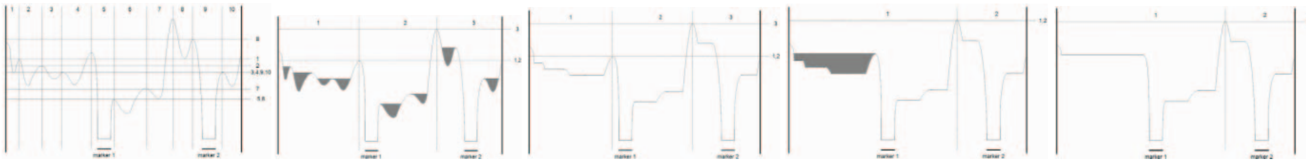
Colorization is the art of adding color to a monochrome image or movie. This is done in order to increase the visual appeal of images such as old black and white photos, classic movies or scientific illustrations. Various semi-automatic colorization approaches have been published previously. They all involve some form of partial human intervention in order to make a mapping between the color and the intensity. Luminance keying also known as pseudo-coloring[1] is a basic colorization technique which utilizes a user-defined look-up table to transform each level of grayscale intensity into a specified hue, saturation and brightness, i.e a global color vector is assigned to each grayscale value. Welsh et al.[2] proposed techniques where rather than choosing colors from a palette to color individual components, the color is transferred from a source color image to a target grayscale image by matching luminance and texture information between the images. This approach is inspired by a method of color transfer between images described in Reinhard et al.[3] and image analogies by Hertzmann et al.[4]. Another well known approach to colorization[5] assumes that small changes take place between two consecutive frames, therefore, it is possible to use optical flow to estimate dense pixel-to-pixel correspondences. Chromatic information can then be transferred directly between the corresponding pixels. There are some approaches [6], [7], [8] which make use of the assumption that the homogeneity in the gray-scale domain indicates homogeneity in the color domain and vice versa. This assumption provides a possibility to propagate color from several user-defined seed pixels to the rest of the image. In [9], colorization is done through luminance-weighted chrominance blending and fast intrinsic distance computations. Shi et al.[10] color the grayscale images by segmentation and color filling method, where an image is first segmented into regions and then the desired colors are used to fill each region. Since the existing automatic image segmentation algorithms usually cannot segment the image into meaningful regions, only color filling of each segmented region cannot produce natural colorized results. Sykora et al.[11] suggested using unsupervised image segmentation in cartoons colorization. However the method usually cannot get ideal results for other types of images and is restricted to only cartoons.

The proposed method makes use of semi-automatic segmentation to segment the image with the aid of colored markers provided by the user and then use the color filling method. The proposed method overcomes the drawbacks of Shi et al.[10] and Sykora[11] and provides better quality results.

2. IMAGE COLORIZATION

We propose a semi-automatic segmentation approach, where the user provides some color hints in the form of color markers which aid in the segmentation process and also contain information as to which segment gets what color. We segment the image using the rain water simulation technique of watershed segmentation[12] where every pixel will belong to a unique segment or what we call catchment basin in watershed terminology. We then construct the watershed lines i.e, the set of pixels which form the boundaries to these basins. Watershed segmentation results in a problem called oversegmentation where many unwanted segments are obtained. To remove oversegmentation, we perform a merging operation through flooding. The steps of the proposed merging operation is as follows:

1. Compute the gradient of the input grayscale image.
2. At the position of the markers, change the pixel values of the gradient image to the minimum value of the gradient image, i.e, we are introducing a global minima at the position of the markers. It is this modified gradient image that is input to the flooding procedure for oversegmentation removal.



(a) (b) (c) (d) (e)

Figure 1: Progression of the algorithm on a cross section of a topographical representation of an image. The vertical lines indicate the watershed lines and the numbers at the top of the image indicate the catchment basin number. The overflow altitude of each catchment basin is indicated by a horizontal line with the numbers on the right indicating the corresponding catchment basin numbers. There are two markers indicated by thick bold horizontal lines at the bottom of the image. (a) 10 catchment basins : basins 5 and 9 non-floodable. (b) Flooding operation in the first iteration. (c) Image structure after first iteration having 3 catchment basins : basins 2,3 non-floodable. (d) Flooding operation in the second iteration. (e) Image structure after second iteration having 2 catchment basins both non-floodable.

(a) (b) (c) (d) (e) (f) (g) (h) (i)

Figure 2: Progression of algorithm on the jeans image. (a) Gray scale image. (b) Image with 3 color markers: markers 1 and 3 are white dots shown enclosed in circles while marker 2 is a blue line shown enclosed in a box. (c) Watershed transformation of modified gradient image :2896 regions. (d) Iteration1 : 489 regions. (e) Iteration 2 : 90 regions. (f) Iteration 3 : 22 regions. (g) Iteration 4 : 5 regions. (h) Iteration 5 : 3 regions. (i) Output colorized image

The following steps represent the flooding algorithm.

3. Determine the catchment basins of the current image through watershed segmentation and identify which of these basins entirely enclose a marker. Such basins are considered non-floodable while the other basins are floodable.
4. Count the number of floodable basins.
5. If the number of floodable basins is zero then STOP, else go to 6.
6. In the current image replace all the pixels belonging to a floodable basin to their overflow altitude if their value is lower than it. Overflow altitude of a basin is defined as the minimum gray level value among the pixels belonging to the watershed line surrounding that basin. The non-floodable basins are left untouched.
7. Go to step 3.

Figure 1 shows the progression of the algorithm over the cross section of an image. The cross section shows the image structure after applying step 2 of the algorithm. Figures 2(c) to 2(h) show the progression of our algorithm on the input image in Figure 2(a) with markers shown in Figure 2(b). On implementing the segmentation algorithm, we get a final segmented image having as many regions as the number of markers. Each pixel in the image is assigned to a unique region which has a unique color marker inside it. We then flood the colors in their respective regions retaining their intensity values. The output of the color flooding process of the jeans image is shown in Figure 2(i). The markers method usually gives correct segmentation results unless the user places the markers incorrectly i.e, for example if the user's marker lies in two different regions which are supposed to be segmented as two separate regions. In such cases the user will have to interactively modify the position and shape of the markers to obtain the correct segmentation results.

3. VIDEO COLORIZATION

In order to colorize videos, few reference frames are chosen manually from a set of automatically generated key frames and colorized using the above marker approach. Their chrominance information is then transferred to the other frames in the video using a color transfer technique making use of motion estimation.

3.1. Key frame generation

The reason for generating key frames is that it provides a small list of significant frames from which the user can select reference frames for the color transfer technique. So instead of looking at the entire video, the user has to look at only the key frames. The reason why we do not select all the key frames as reference frames is that the key frame set may have a number of frames which are similar to each other and they can be manually discarded. We use the method proposed in [13] for identifying key frames. In many shots, key frames are identified by stillness, i.e, either the camera stops on a new position or the characters hold gestures to emphasize their importance. Motion is used to identify key frames. We use block motion estimation to measure the motion in a shot and select key frames at the local minima of motion.

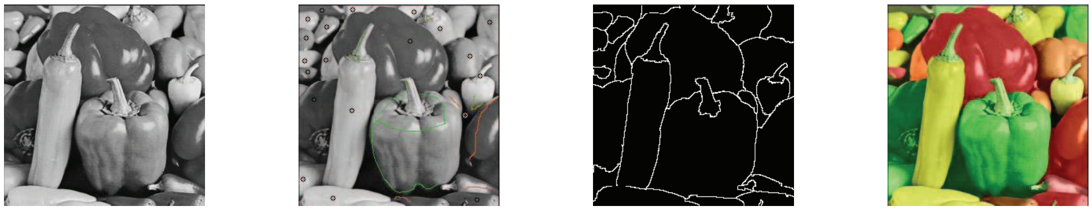
3.2. Color transfer technique

After selecting the reference frames we use a color transfer technique to automatically transfer the color from these colorized frames to the uncolored frames of the video. When a frame is colorized, it will be used as the reference color frame for the next or previous grayscale frame in the video sequence depending on whether the color transfer is taking place in the forward or backward direction, as reference frames may be selected in the middle of the sequence. The color transfer technique makes use of motion estimation and is summarized below:

1. Using the motion estimation algorithm, compute the motion vectors of the current grayscale frame with the previous colorized frame as the reference frame.

(a) (b) (c) (d) (e) (f)

Figure 3. (a) Grayscale image. (b) Grayscale image with color markers. (c) Automatic segmented image. (d) Segmented image using our method. (e) Colorized output using automatic segmentation. (f) Colorized output using our method.



(a) (b) (c) (d)

Figure 4. (a) Grayscale image. (b) Grayscale image with color markers. (c) Segmented image. (d) Output colorized image

2. For each pixel in the target image displace its search space according to the motion vector computed in 1.
3. Find the pixel in the displaced search space of the previous colored source image which gives the minimum error distance for the closest texture match. The error distance between target N_t and the source neighborhood N_s is defined as $E(\mathrm{Nt}, \mathrm{Ns}) = \sum_{x \in \mathrm{Nt,Ns}}[G(x) - C(x)]2$, where G is the grayscale image and C is the luminance channel of the colored image and x are the pixels in these neighborhoods.
4. Transfer the chromatic values of this best match pixel to the target pixel retaining the intensity component of the target pixel.
5. If all pixels have been processed then STOP else go to 2.

4. RESULTS

Figure 3 makes a comparison of our results with the results obtained using automatic segmentation[10]. Figure 3(c) shows the result of automatic segmentation computed using a version of the normalized cuts algorithm[10] while Figure 3(e) shows the colorized output by flooding each segment with the desired color. It is noted that, even state of the art methods in segmentation fail to automatically obtain all the correct boundaries, such as the intricate boundary between the hair and the forehead, or the low contrast boundary between the face and the lips. The segmentation and colorization results obtained by the proposed method and shown in Figures 3(d), 3(f) is better compared to that shown in Figures 3(c), 3(e). Figure 4 shows our image colorization result on the standard pepper image where color markers as dots are shown enclosed in a circle for visibility purpose.

Figures 5 and 6 shows two video sequences colorized using our approach. Figure 5(a) shows 4 frames of the input grayscale tennis sequence which consists of 20 frames. The key frame generation algorithm generates four key frames (Frames 4, 13, 16 and 19). Frames 4 and 13 are picked as reference frames for colorizing using our marker method. Figure 5(c) and 5(d) shows the reference frames 4 and 13 respectively with the color markers along with their respective colorized outputs using the image colorization algorithm. The figures also show the original color image from which the colors were extracted. Here the markers which are just dots are shown enclosed in circles for visibility purpose. Figure 5(b) shows 4 frames of the output colored image sequence obtained using the color transfer technique. Figure 6 shows colorization of a 158 frame sequence of the Charlie Chaplin movie "City Lights". Frames are colorized using the proposed image colorization algorithm along with color markers. This sequence has 4 reference frames which have been picked from a key frame set consisting of 31 frames. Two of those reference frames are shown in Figure 6(c) and 6(d). We have used a fast block motion estimation algorithm equivalent to exhaustive search using a multi resolution approach[14] in our color transfer technique. For the motion estimation algorithm we have used a block size of 16×16, a search area of 33×33. In order to further speed up the process, the color is transferred 2×2 pixel blocks at a time. The 20 frame tennis sequence with frame size 352×240 takes 209.919 sec (28.812 sec for the image colorization of the two reference images + 181.107sec for color transfer) while the 158 frame Charlie Chaplin sequence with frame size 352×240 takes 348.547 sec (20.735sec for the four reference frames + 225sec for color transfer) to colorize on a 3 GHz CPU and 512 MB RAM. The reason why the tennis sequence takes more time is that it is a faster motion sequence compared to the Charlie Chaplin sequence. The results obtained are visually very good and the amount of manual labor required is reduced where all the user has to do is apply some color markers on some selected frames and our colorization algorithm colors the entire video sequence.

5. CONCLUSION

In this paper we have proposed a semi-automatic algorithm for colorizing still images through image segmentation and color markers. The results we have obtained are visually very good and the algorithm works on a wide range of images, unlike the segmentation technique used by Sykora[11] where the method works only on cartoons. The results we have obtained are better than that obtained through automatic segmentation[10]. We also propose a video colorization algorithm using a color transfer technique. The results obtained are visually very good. In addition the amount of manual intervention required is reduced since the user only has to apply some color markers on some selected

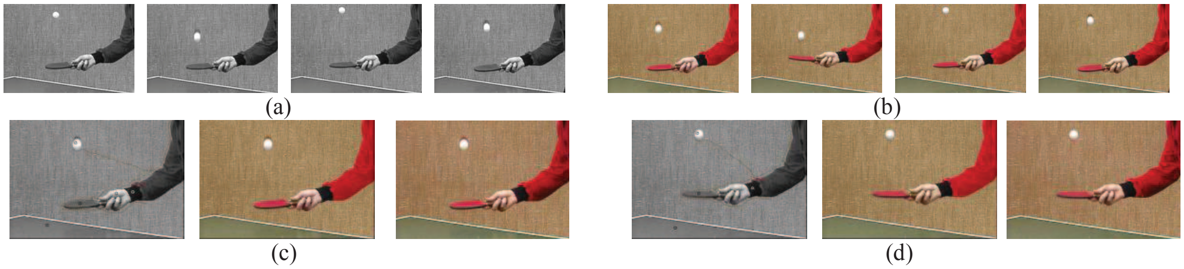


Figure 5. 20 frame Tennis sequence from which 4 frames are shown. (a) Frames 1,6,11 and 20 of the 20 frame input grayscale sequence. (b) Corresponding colorized outputs of Figure 5(a). (c) and (d) show the colorization of reference frames 4 and 13 using the marker approach along with the original colorized image from which the color was extracted for the markers-left:Input grayscale image with color markers,middle:Output colorized image using the marker approach, right:original colorized image

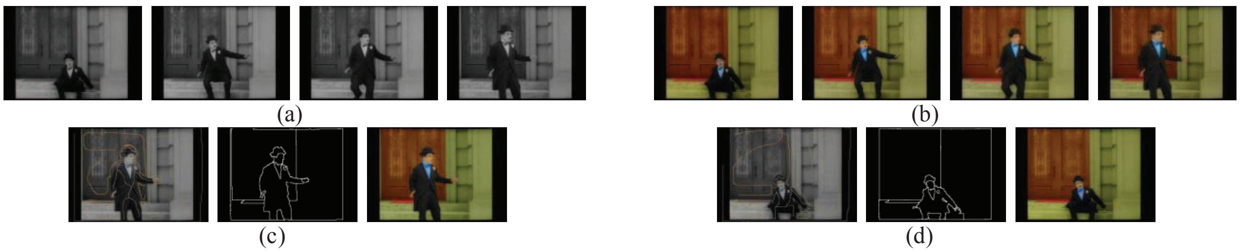


Figure 6. Charlie Chaplin sequence. (a) 4 frames from the 158 frame input grayscale video sequence. (b) Corresponding 4 frames from the output colorized sequence . (c) and (d) show colorization of reference frames 21 and 136 respectively using the marker approach

reference frames, and the colorization algorithm colors the entire video sequence.

11. REFERENCES

[1] R.C. Gonzalez and R.E. Woods, *Digital Image Processing* (second ed.), AddisonWesley Publishing, Reading, MA (1987).

[2] T. Welsh, M. Ashikhmin and K. Mueller, Transferring color to greyscale images, in: *ACM SIGGRAPH 2002 Conference Proceedings (2002)* pp. 277-280.

[3] E. Reinhard, M. Ashikhmin, B. Gooch and P. Shirley, Color transfer between images, *IEEE Transactions on Computer Graphics and Applications 21 (2001) (5)*, pp. 34-41.

[4] A. Hertzmann, C.E. Jacobs, N. Oliver, B. Curless and D.H. Salesin, "Image analogies", *ACM SIGGRAPH 2001 Conference Proceedings*, 327-340(2001).

[5] Z. Pan, Z. Dong and M. Zhang, A new algorithm for adding color to video or animation clips, in: *Proceedings of WSCG—International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision (2004)* pp. 515-519.

[6] T. Horiuchi, Estimation of color for gray-level image by probabilistic relaxation, in: *Proceedings of IEEE International Conference on Pattern Recognition (2002)* pp. 867-870.

[7] T. Horiuchi and S. Hirano, Colorization algorithm for grayscale image by propagating seed pixels, in: *Proceedings of IEEE International Conference on Pattern Recognition (2003)* pp. 457-460.

[8] A. Levin, D. Lischinski and Y. Weiss, Colorization using optimization, in: *ACM SIGGRAPH 2004 Conference Proceedings (2004)* pp. 689-694.

[9] Liron Yatziv and Guillermo Sapiro, Fast image and video colorization using chrominance blending, in: *IEEE Transactions on Image Processing*, Vol. 15, No. 5, May 2006, pp. 1120-1129.
[10] Shi, J., Malik, J., 1997. Normalized cuts and image segmentation. In: *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, pp. 731-737.

[11] Sy´kora, D., Buria´nek, J.,Zara, J., 2003. Segmentation of Black and White Cartoons, In: *Proceedings of Spring Conference on Computer Graphics*, pp. 245-254.

[12] Victor Osma-Ruiz, Juan I. Godino-Llorente, Nicolas Saenz-Lechon, and Pedro Gomez-Vilda, "An improved watershed algorithm based on efficient computation of shortest paths", *Pattern Recogn*., 40(3), 1078-1090(2007).

[13] W. Wolf, Key frame selection by motion analysis, in *Proc. IEEE Int. Conf. Acoust., Speech, and Signal Proc.*, 1996, vol. 2, pp. 1228-1231.

[14] M. Gharavi-Alkhansari, "A fast motion estimation algorithm equivalent to exhaustive search", *Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001)*, vol. 2, 1201-1204, Salt Lake City, Utah, May 7-11, 2001.