Automatic Image Colorization based on Feature Lines

Van Nguyen[1], Vicky Sintunata[1] and Terumasa Aoki[1,2]

[1]*Graduate School of Information Sciences (GSIS), Tohoku University, Aramaki Aza Aoba 6-3-9, Aoba-ku, Sendai, Japan*

[2]*New Industry Hatchery Center (NICHe), Tohoku University, Aramaki Aza Aoba 6-6-10, Aoba-ku, Sendai, Japan*

Keywords: Automatic Colorization, Color Lines, Feature Lines.

Abstract: Automatic image colorization is one of the attractive research topics in image processing. The most crucial task in this field is how to design an algorithm to define appropriate color from the reference image(s) for propagating to the target image. In other words, we need to determine whether two pixels in reference and target images have similar color. In previous methods, many approaches have been introduced mostly based on local feature matching algorithms. However, they still have some defects as well as time-consuming.

In this paper, we will present a novel automatic image colorization method based on Feature Lines. Feature Lines is our new concept, which enhances the concept of Color Lines. It represents the distribution of each pixel feature vector as being elongated around the lines so that we are able to assemble the similar feature pixels into one feature line. By introducing this new technique, pixel matching between reference and target images performs precisely. The experimental achievements show our proposed method achieves smoother, evener and more natural color assignment than the previous methods.

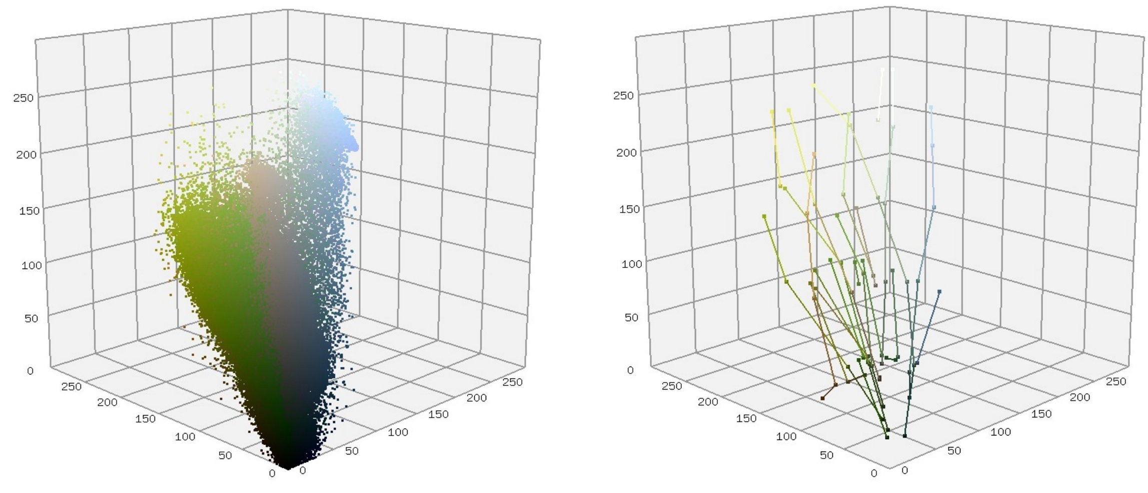
1 INTRODUCTION

Image colorization works on finding a worth solution for adding colors to mono-chroma images. Approaching a novel solution in this field contributes the strong principle for colorizing large amount of old images and videos. Depending on the contribution of user influence in colorization process, existing image colorization methods can be classified into two main categories: interactive (or manual) colorization (Pang et al., 2014), (Levin et al., 2004), (Marki et al., 2014) and automatic colorization techniques (Gupta et al., 2012), (Yang et al., 2014), (Irony et al., 2005), (Chia et al., 2011). All interactive methods require dozens of color feeding from users, (Pang et al., 2014) expands provided scribbles by self-similarity algorithm where similar patches with provided color cues are identified by looking inside a pre-defined search window. (Levin et al., 2004) works on inferring color of gray pixels from provided clues by optimizing the difference between known-color pixels around particular gray scale pixel within a window. The work of (Marki et al., 2014) uses geodesic distance to transfer color from user-provided strokes to other pixels in image and concentrates on creating a simulation of water painting application which produces smooth and artistic colorized images. Although the gray images are impressively colorized, these user-assisted methods demand plentiful color scribbles feeding from users. The process of colorizing images requires strong and careful efforts from users. In the situation of automatic colorization, recent approaches require robust feature vectors to achieve high precision of matching algorithm between reference and target pixels, along with that, high computational cost is needed. Our goal is to focus only on automatic colorization technique and demand the standard features in pixel matching schemes to overcome this obstacle.

In RGB color space, it is non-trivial problem to determine whether two pixels have the similar color. The concept of Color Lines (Omer and Werman, 2004) exploits the information of the pixels in RGB spaces to build a Color Lines model in which the pixels having similar color will be elongated around their representative color lines. It means that if we know two pixels belonging to the same color line they are probably similar in RGB channels. In other words, the line which the pixels elongate to, is their representation in RGB color space and we can project color from the color lines to them with trivial discrepancy.

Expanding the concept of Color Lines, we intend to introduce a method of visualizing pixel feature as a vector of three feature components. Following this approach, each of input images (including reference and target image) can be converted into "feature image" where the feature components are considered as

Nguyen, V., Sintunata, V. and Aoki, T.
Automatic Image Colorization based on Feature Lines.
DOI: 10.5220/0005676401260133
In *Proceedings of the 11th Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2016) - Volume 4: VISAPP*, pages 126-133
ISBN: 978-989-758-175-5



(a) (b)

Figure 1: (a) Pixel RGB histogram, (b) Color Lines model.

the RGB channels corresponding to each pixel. When applying Color Lines concept to feature images, we can get the Feature Lines model of each input image. These feature lines are plausibly precise feature representation for the pixels which elongate to them.

In this paper, we will introduce a new method of automatic colorization based on the concept of Feature Lines. Our method uses very simple features including pixel intensity, standard deviation and neighbor standard deviation to classify pixels and match them up between reference and target images. By applying Feature Lines concept to target and reference feature images, we can get their corresponding Feature Lines models to use as the inputs for our matching scheme. The following sections will discuss in detail about the algorithm to construct Feature Lines models from input images and its application on automatic colorization method.

2 PREVIOUS WORK

2.1 Existing Automatic Image Colorization Algorithms

Automatic image colorization is the rich fields where there are many works have been introduced. (Irony et al., 2005) proposed a method for colorizing gray images using image and feature space voting where Discrete Cosine Transform (DCT) Coefficients of a k by k neighborhood around each pixel was used as its feature vector. The authors go further by switching to low dimensional subspace of feature using Principal Component Analysis (PCA) and projections to overcome the problem of mismatching referent and target segments. The proposed algorithm performs well in the case of input images with low number of textures and might fail when images contain diverse details.

Another work in this field is the internet images search based colorizing system (Chia et al., 2011) where authors concentrate on introducing a software system for colorizing mono-chroma images with less user efforts. The proposed system required user to provide a label for each target image to use as searching keyword and a segmented image as filter to obtain most appropriate reference image from internet. This method exploits the vibrant resource from the internet. Though, it might not work well when users can not provide a concrete keyword for searching step.

In the state-of-the-art automatic colorization method (Gupta et al., 2012), the authors use a bundle of feature vectors corresponding to each superpixel which includes Gabor, SURF, standard deviation and intensity feature in a cascade feature matching scheme. However, feature vectors are extracted as the average all of the pixels in target and reference image superpixels respectively. Then these vectors are used as the representation for pixels belonging to a specific superpixel. Therefore, the results of feature extracting process will be affected by the accurracy of superpixel extraction algorithm. Superpixel is the group of square-shaped neighboring pixels with a specific size so that there will exist many superpixels assembling dozen of stray pixels especially in the case of images containing many different small details or the pixels near by edges of objects. Moreover, when extracting SURF feature for arbitrary pixel, the prerequired parameters of the keypoints are missing or they are left to default values. These shortages will influence the precision of the superpixel feature computation leading to the inaccuracy in feature matching algorithm.

In the most recent automatic colorization method (Yang et al., 2014), reference and target images are condensed to epitomes using hidden mapping scheme. This approach can perform efficiently in the case of images with few number of textures and become less productive when input images contains a large amount of details. Beside that, the learning process for epitomic image generation uses only single type of feature, however, robust feature vectors including YIQ chanels, dense SIFT feature and the rotation invariant Local Binary Pattern (LBP) are still demanded for matching algorithm.

These approaches demand a bundle of robust feature vectors to achieve high precision matching results, however, the high computational cost is also required to implement these algorithms.

2.2 Color Lines Representation

Color Lines has been introduced as the ideal model for pixel classification in RGB color space. Based on the observation that two pixels having similar color should be closed to each other when being plotted in RGB coordinate system. By exploiting the geometrical properties of pixel RGB components, (Omer and Werman, 2004) builds a concrete clustering scheme to classify image pixels into color clusters. Each of cluster is represented by two connected pixels which creating a line segment as its skeleton so-called ”color line”.

Color Lines algorithm firstly slices RGB histogram using the hemispheres of equal radius distances centered at the origin O. Each *histogram slice* is the collection of all pixels with RGB-norms in between two upper and lower hemisphere surface boundaries. The maxima points are determined as the pixels which intersect with higher hemisphere surface. To define the color points of each color line in corresponding histogram slice, simply picks up the pixels with maxima RGB norms. Then, the Euclidean distances and a threshold are used as the parameters to joint pair of color points from neighbor histogram slices in to color line skeleton. A Gaussian is fitted to each skeleton and used as the classifying model to distribute pixels into corresponding color line cluster. Finally, from the RGB histogram shown in figure 1a we can get the image Color Lines representation model depicted in figure 1b. By using this model, RGB coordinates of pixels can be recovered by projecting color from their belonging color lines.

Figure 2: Feature image.

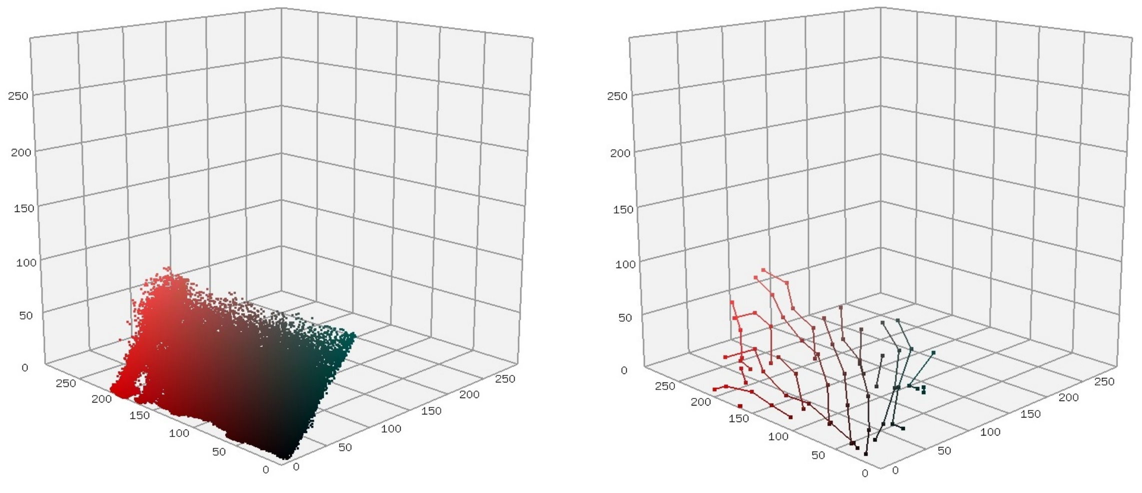
3 FEATURE LINES

3.1 Feature Lines Concept Intuition

Color Lines representation performs its advantages on pixel color classification by introducing a concrete clustering algorithm based on only RGB components of image pixels. Since our implementation is involving the problem of image colorization, the idea of exploiting the achievements of Color Lines come up to us intuitively. Target and reference image are semantically chosen so that they should have similar Color Lines model. In other word, if we can determine Color Lines model of reference image, then the colorized target Color Lines model will be alike. Beside that, target and reference image are similar which means there exists the corresponding image areas in each of them having similar characteristics or feature vectors.

The method of using pre-extracted image segments for feature matching scheme have been introduced in many previous approaches. (Gupta et al., 2012) method extracts feature vectors of superpixels to feed them to a cascade feature matching process and (Chia et al., 2011), (Irony et al., 2005) requires segmented images before performing further steps. Although there are many efficient methods for extracting superpixels (Achanta et al., 2012), (Levinshtein et al., 2009) or segments (Comaniciu et al., 2002), all of them are affected by spatial constraint of pixel in image matrix which might be the weakness in feature extracting work since pixels in different positions of an image can have similar neighbor and texture characteristics.

Our approach moves in the opposite direction of those familiar processes. We directly segment image based on the local features of its pixels to avoid the double erroneous short-coming in clustering and fea-



(a) (b)

Figure 3: (a) Pixel feature coordinates, (b) Feature Lines representation.

ture extracting process. Unearthing this motivation, we think about applying the concept of Color Lines for feature components instead of three RGB color channels. We construct three-dimensional feature coordinates for each pixel by using one dimensional feature vectors. With three types of pixel feature, we can exactly mimic RGB color elements as in Color Lines model. Consequently, the output model of this process is the feature model corresponding to each input image so-called "Feature Lines" model.

3.2 Feature Coordinates Generation

Feature Lines is the extension of Color Lines concept in three-dimensional feature space. For this purpose, each pixel need to be constructed with three-dimensional feature coordinate. In this paper, we use intensity, standard deviation and neighbor standard deviation as three components of pixel feature coordinates.

Intensity Feature. We use gray scale value as the first component of pixel feature coordinate.

Standard Deviation Feature. The second feature component is pixel standard deviation, this feature can be calculated by using below expression.

$$f_{deviation} = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2} \quad (1)$$

where n is the total number of pixels in neighboring window of current pixel, x_i, $\bar{x}$ are the intensity of current pixel and mean of neighboring pixel intensities respectively.

Neighbor Standard Deviation Feature. Neighbor deviation is the mean of pixel standard deviations in a square window around each pixel.

$$f_{deviation\_neighbor} = \frac{d+\bar{d}}{2} \quad (2)$$

where d is the deviation of current pixel and $\bar{d}$ denotes the mean of neighboring pixel deviations.

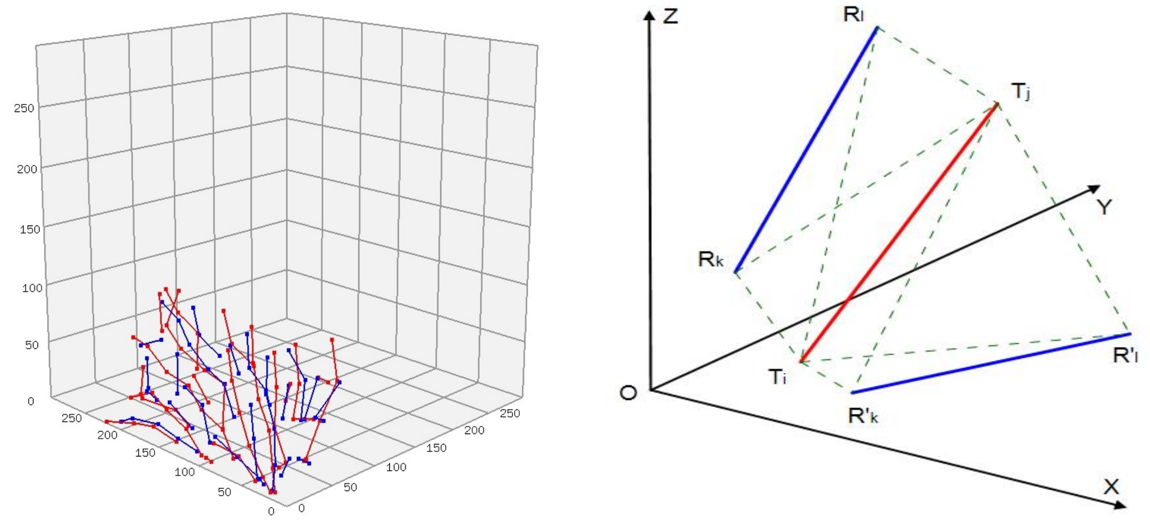
3.3 Feature Lines Construction

After computing feature elements, we construct three-dimensional coordinates for each pixel of reference and target image. The outputs of this step are the images where feature coordinates are visualized as RGB components of the pixels yielding feature images. To define intensity for feature images, we simply compute the mean of three-dimensional coordinates of each pixel. Figure 3 depicts three-dimensional feature coordinates 3a and Feature Lines model 3b in RGB color space. When we apply the concept of Color Lines on feature images, the obtained models are feature-based pixels classification where pixels having similar features will elongate around the feature lines. In the following section, we will discuss further the algorithm of applying Feature Lines model to solve the problem of automatic image colorization.

4 AUTOMATIC COLORIZATION USING FEATURE LINES

4.1 Feature Lines Matching

The most crucial task of automatic image colorization is to determine the corresponding color between ref-



(a) (b)

Figure 4: (a) Reference (red) and target (blue) Feature Lines models, (b) Feature Lines segment matching algorithm.

(a) (b)

Figure 5: (a) Mean color transfering for matched feature lines, (b) Mean color transfering for matched superpixels in (Gupta et al., 2012).

erence and target image. The obvious methods are to use the Euclidean distances between feature vectors of reference and target pixels. However, this method will suffer from the weakness of Euclidean distance. Althought Euclidean measurement can preserve the difference or similarity of magnitude between two given vectors, it is vunerable for extracting the geometrical relation between them. Our method will exploit the advantages of Feature Lines models to strengthen matching scheme.

As the inducements of Feature Lines which are inherited from the Color Lines concept, Feature Lines model of an image is the feature representation for the pixels belonging to them. Since reference and target image are similar, their feature line models should also be alike. Figure 4b shows the Feature Lines models of reference and target images in single three-dimensional space. It is obvious to see that when we consider each pair of reference and target feature line segments, the best candidate of referent feature line should be the line closest target feature line. Beside that, Feature Lines models are the spatial-based feature clustering themself so that they are the concrete inputs for feeding to feature matching scheme. Figure 4b demonstrates our technique to define corresponding feature lines between input images. We simply compute the Euclidean distances between their feature points coordinates and use them as the decisive cost in matching process. Below is the computational formula.

$$m_{cost} = d_{T_iR_k} + d_{T_iR_l} + d_{T_jR_k} + d_{T_jR_l} \tag{3}$$

where d is the Euclidean distance between target and reference feature line segment points, T_i, T_j denote target feature line segment points and R_k, R_l are reference feature line segment points. The matched ref-

(a) (b)

Figure 6: (a) Zoom in Feature Lines based colorized image, (b) zoom in colorized image using (Gupta et al., 2012) method.

erence feature line segment should be the one which has the lowest m_{cost}.

Since Color Lines concept shows their advantage in color preservation, we will perform further steps by using Color Lines as the model to propagate color from reference to target pixels.

4.2 Color Projection

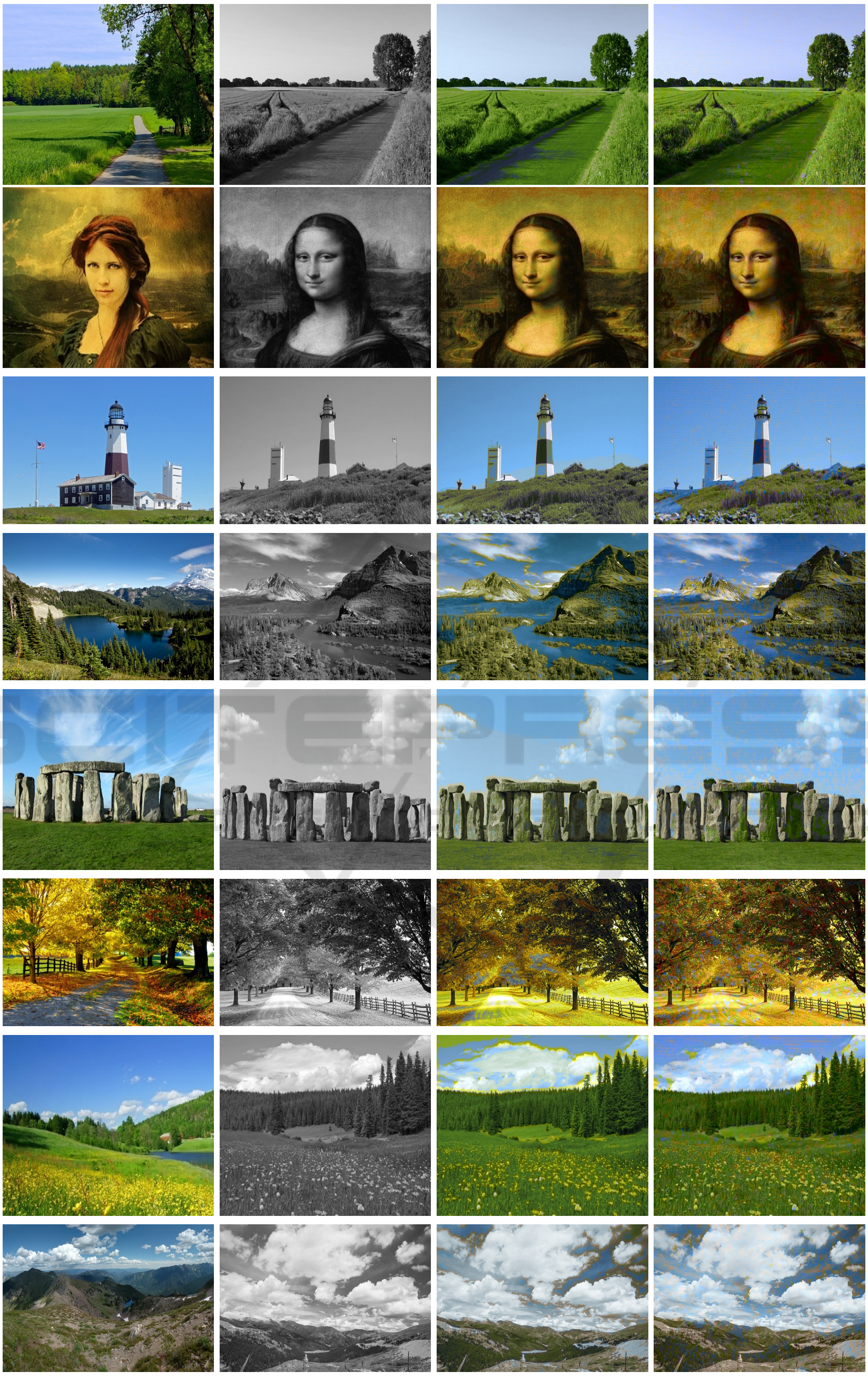
We will firstly, construct Color Lines models for reference and target image. We consider target image as a ”color” image with R, G, B channels are equal to gray scale of each pixel. The step to construct Color Lines for target image is actually to classify its pixels based on their gray intensities. However, by applying Color Lines concept, the output is expected to be more precise.

After getting Color Lines models of input images, we map the corresponding color line for each feature line segment by simply defining common pixels between them and keeping only the color line which contains most of pixels belonging to current feature line. Since we have 1:1 correspondence between color line and feature line, we can then directly define the corresponding reference color line of target pixels. The final step is to project color from reference color line to target pixels based on their belonging reference color line and gray scale level.

5 EXPERIMENTAL RESULTS

In the previous sections, we discussed our proposed algorithm based on Feature Lines concept to tackle the problems of automatic image colorization. This section will show the achievements of our algorithm implementation to some input images and the comparisons with (Gupta et al., 2012)'s method.

Figure 5 depicts the results of our Feature Lines based matching scheme comparing with cascade feature matching in the state-of-the-art superpixel based method (Gupta et al., 2012). It is clearly to see that, by only transfering mean color of corresponding patches, our matching result is more uniform and evener than superpixel based scheme. Moreover, the color propagation process is performed by exploiting Color Lines representation model which can smoothy and evenly projects color from color lines skeleton to belonging pixels. Our Feature Lines based algorithm returns smoother and more natural color assignment without any color jerky which might occur in superpixel-based approach as shown in figure 6. Figure 7 shows our experimental results and the comparison with superpixel based method. While superpixel based algorithm requires robust feature vectors such as SURF and Gabor features to achieve highly precise colorization, our experiments use only three basic pixel feature vectors: Pixel intensity, standard deviation feature within the 3x3 square window around each pixel and neighbor standard deviation feature with the 9x9 window size. Despite those very limited input conditions, our method generates better results as dipicted in figure 7c compared with the superpixel based method shown in figure 7d. Our method classifies pixels based on their feature vectors without any constraint to spatial position in image matrix. Therefore pixel characteristics are completely preserved and it is guaranteed that pixels having similar features gather in the same feature line.



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

Figure 7: (a) Reference image, (b) target image, (c) our proposed method, (d) (Gupta et al., 2012) method.

6 CONCLUSION AND FUTURE WORK

In this paper, we proposed a new feature matching scheme using Feature Lines model which exploits the advantages of Color Lines representation concept and its application on solving the problem of automatic image colorization. By constructing three-dimensional feature vectors, we considered them as coordinates of pixels in color space. Reference image is semantically similar to target image, intuitively, they should have the similar Feature Lines models which are the outcomes when applying Color Lines concept to feature images. Following this theory, we were able to match up feature lines and consequently pixels from reference and target image. To propagate color from matched reference to target pixels, we represented reference image as a set of color lines and defined corresponding color lines of feature lines in reference image. Color transfering process could be done accordingly by projecting color from corresponding color lines to harmonized target pixels.

Since, the results of automatic colorization algorithms strongly depend on how semantically equivalent between reference and target image, we might get imprecise results when input images are not satisfying. Our method exploits the advantage of Color Lines concept in feature space. Pixels are classified based on the distribution of their feature vectors in three-dimensional space. However, feature vectors arrangement is not persistently similar with RGB color. Feature points gather in a denser and more crowded area than color pixels in RGB space. For future work, we would like to explore more robust pixel features to strengthen matching scheme and work on improving clustering algorithm to overcome the obstacle of feature distribution. Additionally, we also intend to extend our method for any dimensional vectors since our current approach only dedicates for three-dimensional features.

REFERENCES

Achanta, R., Shaji, A., Smith, K., Lucchi, A., Fua, P., and Susstrunk, S. (2012). SLIC superpixels compared to state-of-the-art superpixel methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 34(11):2274–2281.

Chia, A. Y.-S., Zhuo, S., Gupta, R. K., Tai, Y.-W., Cho, S.-Y., Tan, P., and Lin, S. (2011). Semantic colorization with internet images. *ACM Transactions on Graphics*, 30(6):1.

Comaniciu, D., Meer, P., and Member, S. (2002). Mean Shift: A Robust Approach Toward Feature Space Analysis. 24(5):603–619.

Gupta, R., Chia, A., and Rajan, D. (2012). Image colorization using similar images. *Proceedings of the 20th ACM international conference on Multimedia*, pages 369–378.

Irony, R., Cohen-Or, D., and Lischinski, D. (2005). Colorization by Example. *Symposium A Quarterly Journal In Modern Foreign Literatures*, pages 201–210.

Levin, A., Lischinski, D., and Weiss, Y. (2004). Colorization using optimization. *ACM Transactions on Graphics*, 23(3):689.

Levinshtein, A., Stere, A., Kutulakos, K. N., Fleet, D. J., Dickinson, S. J., and Siddiqi, K. (2009). TurboPixels: Fast superpixels using geometric flows. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(12):2290–2297.

Marki, N., Wang, O., Gross, M., and Smoli, A. (2014). *COLORBRUSH : Animated Diffusion for Intuitive Colorization Simulating Water Painting*. pages 4652–4656.

Omer, I. and Werman, M. (2004). Color lines: image specific color representation. *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, 2.

Pang, J., Au, O. C., Yamashita, Y., Ling, Y., Guo, Y., and Zeng, J. (2014). *Self-Similarity-Based Image Colorization* The Hong Kong University of Science and Technology Tokyo Institute of Technology. pages 4687–4691.

Yang, Y., Chu, X., Ng, T. T., Chia, A. Y.-s., Yang, J., Jin, H., Huang, T. S., Avenue, N. M., Star, a., and Way, F. (2014). *Epitomic Image Colorization* Department of Electrical and Computer Engineering , University of Illinois at Urbana-Champaign Adobe Research , San Jose , CA 95110 , USA. pages 2489–2493.