Conversion of Gray-scale image to Color Image with and without Texture Synthesis

Mrs.V. Karthikeyani[1] Dr.K.Duraiswamy[2] Mr.P.Kamalakkannan[3]
[1,3] Research Scholar, K.S.Rangasamy College of Technology, Tiruchengode, TamilNadu, INDIA.

[2] Dean(Academic), K.S.Rangasamy College of Technology, Tiruchengode, TamilNadu, INDIA.

Summary
We introduce a general technique for coloring the gray-scale images into the color images with and without texture synthesis. The general problem of adding chromatic values to a gray-scale image has no exact solution, the current approach attempts to provide a method to minimize the amount of human labor required for this task, rather than choosing RGB colors to individual components, we convert the entire color to the gray-scale image by matching luminance and texture information between the images. We choose only chromatic information and retain the original luminance values of the gray-scale image. Further, the procedure is enhanced by allowing the user to match areas of the two images with rectangular samples. We show that this simple technique can be successfully applied to a variety of images, provided that texture and luminance are sufficiently distinct. The images generated the potential utility of our technique in a diverse set of application domains.
Keywords: Image Processing, Color, Texture Synthesis.

1. Introduction

The color fundamental process followed by the human brain in perceiving color is a physiopsychological phenomenon that is not yet fully understood, the physical nature of color can be expressed on a formal basis supported by experimental and theoretical results. Basically, the colors we perceive in an object are determined by the nature of the light reflected from the object. Due to the structure of human eye, all colors are seen as variable combinations of the three so-called Primary colors Red, Green and Blue (RGB).

The characteristics generally used to distinguish one color from another are brightness, hue and saturation. Brightness refers to intensity. Hue is an attribute associated with the dominant wavelength in a mixture of light waves. Saturation refers to relative purity or the amount of white light mixed with a hue. The Hue and saturation taken together are called chromaticity, and therefore a color may be characterized by its brightness and chromaticity. The amounts of RGB needed to form any given color are called the tri-stimulus value. The chromaticity is useful for color mixing because a straight line segment joining any two points, and define all the different color variations that can be obtained by combining these two colors additively.

Color can be added to gray-scale images in order to increase the visual appeal of images such as black and white photos, classic movies and scientific illustrations. In addition, the information content of some scientific images can be perceptually enhanced with color by exploiting variations in chromaticity as well as luminance.

The task of coloring a gray-scale image involves assigning RGB values to an image which varies along only the luminance value. Since different colors may have the same luminance value but vary in hue or saturation, the problem of coloring gray-scale images has no correct solution. Due to these ambiguities, human interaction usually plays a large role in the coloring process.

Here, the gray-scale image is represented by only the luminance values that can be matched between the two images. Because a single luminance value could represent entirely different parts of an image, the remaining values within the pixel's neighborhood are used to guide the matching process. Once a pixel is matched, the color information is transferred but the original luminance value is retained. In difficult cases, a few sample blocks can be used to aid the matching process between the source (color) and the target (gray-scale) image. After color is transferred between the source and the target sample blocks, the final colors are assigned to each pixel in the gray-scale image by matching each gray-scale image pixel to a pixel in the target sample blocks using the distance metric. Thus, each pixel match is determined by matching it only to other pixels within the same image. We have found that this simple procedure works well for a wide range of image types.

2. Literature Survey

Even in the case of pseudocoloring, [4] where the mapping of luminance values to color values is automatic, the choice of the colormap is commonly determined by human decision. However, a few web articles describe software in which humans must meticulously hand-color each of the individual image regions. For example, one software package is described in which the image is first polygonalized so the user can color individual components much like a coloring book. Then the system tracks polygons between frames and transfers colors in order to reduce the number of frames that the user must color manually [7].

Manuscript received April 5, 2007
Manuscript revised April 25, 2007

Alternatively, photographs can be colorized using photo-editing software to manually or automatically select components of a scene. The area is then painted over with a color selected from a palette using a low opacity level.

There also exist a number of applications for the use of color in information visualization. For example, [4] describe a simple approach for pseudocoloring gray-scale images of luggage acquired by X-ray equipment at an airport. The method uses separate transformations for each color channel which results in coloring objects with the density of explosives in bright orange and other objects with a blue tone. Further, color can be added to a range of scientific images for illustrative and educational purposes.

An image enhancement technique describes in [8] because it can be used to enhance the detectability of detail within the image. In its most basic form, pseudocoloring is a transformation T, such that, $c(x,y) = T(f(x,y))$ where $f(x,y)$ is the original gray-scale image and $c(x,y)$ is the resulting color vector for the RGB. A simplified example of this method is the application of an arbitrary color map to the data where a single, global color vector is assigned to each gray-scale value. The strength of this approach is that it does not alter the information content of the original data since no extra information is introduced.

However, by using a colormap which does not increase monotonically in luminance, pseudocolored images may introduce perceptual distortions. Studies have found a strong correlation of the perceived “naturalness” of face images and the degree to which the luminance values increase monotonically in the colormap.

Colorization technique is to exploits textural information. The work of Welsh et al [32], which is inspired by the color transfer [30] and by image analogies [31], examines the luminance values in the neighborhood of each pixel in the target image and adds to its luminance the chromatic information of a pixel from a source image with best neighborhoods matching.

Our concept of converting the color from one image to another is inspired by work by [9] in which color is transferred between two color images. In their work, colors from a source image are transferred to a second colored image using a simple but surprisingly successful procedure. The basic method matches the distribution of color values between the images and then converting the gray-scale image to color image. Further, sample blocks can be employed to match similar areas between the two images.

3. Algorithm

In this section, we describe the general algorithm for transferring color, the basic idea is then extended to use sample blocks. The general procedure for converting the gray-scale image to color image requires a few simple steps.
Step 1: Each image is converted into the lxy color space. We use jittered sampling to select a small subset of pixels in the color image as samples.
Step 2: Next, we go through each pixel in the gray-scale image in scan-line order and select the best matching sample in the color image using neighborhood statistics.
Step 3: The best match is determined by using a weighted average of pixel luminance and the neighborhood statistics.
Step 4: The chromaticity values (x,y) of the best matching pixel are then transferred to the gray-scale image to form the final image.
Step 5: Color transfer using sample blocks involves the same for the whole image matching procedure but only between the source and target sample blocks.
The colored pixels in the target sample block regions are then used as the source pixels for conversion of color to the remaining non-colored pixels using a texture synthesis approach, is explained in [1].

3.1 Matching Procedure

Both color (source) and gray-scale (target) RGB images are converted to the decorrelated lxy space [Ruderman et al. 1998] for subsequent analysis. lxy space was developed to minimize correlation between the coordinate of the color space. The color space provides the decorrelation, corresponding to an achromatic luminance, l and two chromatic values x and y, which roughly correspond to yellow-blue and red-green opponent channels. Thus, changes made in one color should minimally affect values in the other. The reason the lxy color space is selected in the current procedure is, because it provides a decorrelated achromatic value for color images. This allows us to selectively transfer the chromatic x and y values from the color image to the gray-scale image without cross-channel artifacts. The transformation procedure follows directly from [9].

In order to transfer chromaticity values from the source to the target, each pixel in the gray-scale image must be matched to a pixel in the color image. The comparison is based on the luminance value and neighborhood values of that pixel. The luminance value is determined by the l in lxy space. In order to account for whole differences in luminance between the two images we perform luminance remapping [5] to linearly shift and scale the luminance histogram of the source image to fit the histogram of the target image. This helps create a better correspondence in the luminance range between the two images but does not alter the luminance values of the target image.

The neighborhood values are precomputed over the image and consist of the standard deviation of the luminance values of the pixel neighborhood. We have found that a neighborhood size of 5x5 pixels works well for most images. For some problematic images we use a larger neighborhood size.

Since most of the visually significant variation between pixel values is attributed to luminance differences, we can limit the number of samples we use as source color pixels and still obtain a significant range of color variation in the image. This allows us to reduce the number of comparisons made for each pixel in the grey-scale image and decrease computation time. We have found that approximately 200 samples taken on a randomly jittered grid is sufficient. Then for each pixel in the gray-scale image in scan-line order the best matching color sample is selected based on the weighted average of luminance (50%) and standard deviation (50%). We have also included the neighborhood mean and varied the ratio of these weights but have not found significant differences in the results. Once the best matching pixel is found, the x and y chromaticity values are transferred to the target pixel while the original luminance value is retained.

This automatic, matching procedure works reasonably well on images when corresponding color regions between the two images also correspond in luminance values. However, regions in the target image which do not have a close luminance value to an appropriate structure in the source image will not appear correct.

3.2 Sample blocks

In order to allow more user interaction in the color transfer procedure and to improve results, sample blocks are used between corresponding regions in the two images.

The first step is to use the general procedure described above to transfer color, but now only between the corresponding sample blocks. This allows the user to selectively transfer colors between the source and target sample blocks. We also expect the results to be good for individual sample blocks because there should be less overlap of luminance levels between different color regions within the same block. We perform luminance remapping as in the whole image block procedure but only between corresponding sample blocks. Again, we use random jittered sampling with approximately 50 samples per block.

The second step is similar to texture synthesis algorithms [1,2,3] in which the distance is used to find texture matches. We define the error distance E using the distance metric between neighborhood N_g in the gray-scale image and N_c in the colored block neighborhood as:
$E(N_g,N_c) = \sum [G(p) - L(p)]^2$, $p \in N$ where G is the gray-scale image, L is the luminance channel of the colored block and p are the neighborhood pixels.

Note, at this stage we no longer search the color image for texture matches but only search for matches within the colored sample blocks in the target image. The advantage of the approach is that in the first stage we transfer colors to the sample blocks selectively which prevents pixels with similar neighborhood statistics but from the wrong part of the image from corrupting the target block colors. It also allows the user to transfer colors from any part of image to a select region even if the two corresponding regions vary largely from one another in texture and luminance levels. Secondly, since we expect there to be more texture coherence within an image than between two different images, we expect pixels which are similar in texture to the colored target sample blocks to be colored similarly.

4. Results

The technique works well on scenes where the image is divided into distinct luminance clusters or where each of the regions has distinct textures. Although we show that the algorithm works well in a number of image domains, we do not claim that the technique will work on most images. It should be clear that when one considers only a small neighborhood size around a pixel it is often impossible to determine whether that neighborhood belongs to one texture or another. However, by using high resolution images and larger neighborhoods we can obtain improved results. Further, we believe that more images can be colorized using the basic method provided but with better texture classification methods at the expense of simplicity and computation time.

Result of [9] is, applying the method directly in a corrected image with too much major color in it. Reducing the standard deviation in the red–green channel by a factor of 10 produced the image, which more closely resembles the old photograph's appearance.


(a)


(b)


©

Fig. 1. (a), (b) & (c) are the Results of [9]

Fig.1.(a), (b) & (c) shows the result of [9] that leaves the average for the achromatic channel unchanged, because this would affect the overall luminance level. The standard deviations should also remain unaltered. The computation time may large compare with our computation time.

The result of [11] works well only on scenes where the image is divided into distinct luminance clusters or where each of the regions has distinct textures. The limitation in [11] it does not work very well for faces and medical image data.

4.1 Conversion from gray to color with synthesis

Here, using [2] the input image is a gray-scale image, that image has been synthesized. After the synthesis, color can be extracted from other image and applied to our synthesized image. Here we are mainly concentrating the Chrominance and the Luminance values.

Fig.2 Input gray-scale image

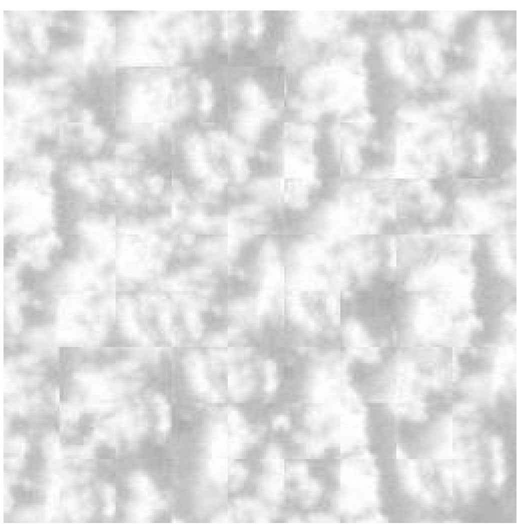

Fig.3 shows a output synthesized gray-scale image

Fig.2 shows the input gray-scale image, Fig.3 shows an output synthesized gray-scale image, Fig.4 shows the color image (for color extraction) and the Fig.5 shows the colored result of the synthesized image.

4.2 Conversion from gray-scale to color image without synthesis

Here, color is extracting from one image and applied to some other gray-scale image. Without synthesis, directly the gray scale image is converted to color image. Fig.6 shows the input gray scale image, Fig. 7 is the color image from which the color is going to extract and the colored output is shown in the Fig.8.

Fig. 4 Sample color image

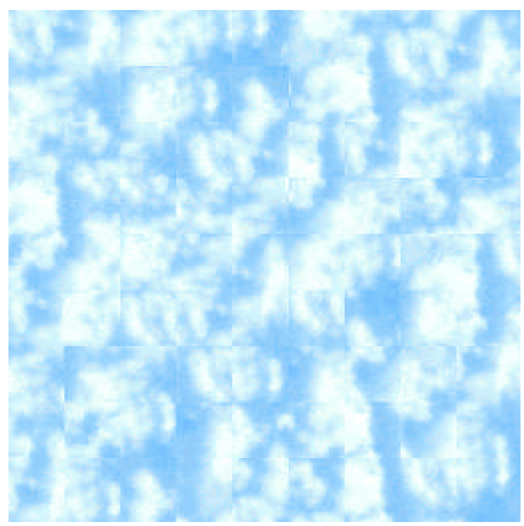

Fig. 5 colored result for Fig. 4 (for color extraction)

The running time of the algorithm for one image can range from 15 seconds to 4 minutes on a Pentium III 900 Mhz CPU using optimized MATLAB code. Running time will vary depending on the number of samples used for comparison, the number of sample blocks, neighborhood size and the size of the images. Most images can be colored reasonably well in under a minute.

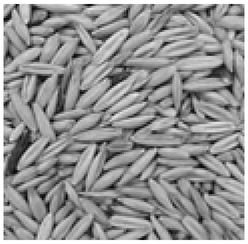

Fig. 6 Input gray image

Fig. 7 Input color image

Fig. 8 Colored Output image

5. Conclusion

In this paper we have formulated a new, general, fast, and user-friendly approach to the problem of coloring gray-scale images. While standard methods accomplish this task by assigning pixel colors via a color palette, our technique empowers the user to first select a suitable color image and then transfer the color of this image to the gray-scale image at hand. We have intentionally kept the basic technique simple and general by not requiring registration between the images or incorporating spatial information. Our technique can be made applicable to a larger class of images by adding a small amount of user guidance. In this mode, the user first transfers the desired color from a set of specified block regions in the color image to a set of corresponding blocks in the gray-scale image. Then, in the second and final stage of the coloring process, the colored sample blocks are employed, using a texture synthesis-like method, to color the remaining pixels in the gray-scale image. Currently, the distance is used to measure texture similarity within the image. With the help of the texture synthesis algorithm, we developed the result with synthesis and without texture synthesis. In the future we believe the technique can be substantially improved by using a more sophisticated measure of texture animation.

References

[1] Mrs.V.Karthikeyani, Dr.K.Duraiswamy & P.Kamalakkannan, "Single and Multi – Pixel based Texture Synthesis", GESTS International Transactions on Computer Science and Engineering, ISSN 1738-6438, Vol. 34, Nov. 2006, pp. 142 – 150.

[2] Mrs.V.Karthikeyani, Dr.K.Duraiswamy & P.Kamalakkannan, "Evaluation of Texture Methods for Image Analysis", National Conference on Advanced Image Processing and Networking, National Engineering College, Kovilpatti, Feb.,13-14,2004.

[3] A. A. Efros and T. K. Leung, "Texture synthesis by non-parametric sampling", In *International Conference on Computer Vision*, pages 1033–1038, Corfu, Greece, September 1999.

[4] Rafael C. Gonzalez & Paul Wintz, "Digital Image Processing", Addison-Wesley Publications, May, 1987.

[5] Madhuri A. Joshi, "Digital Image Processing: An Algorithmic Approach".

[6] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D.H. Salesin, "Image analogies", In *Proceedings of ACM SIGGRAPH,* pp. 341- 346, 2002.

[7]SILBERG, J., Cinesite Press Article, 1998. http://www.cinesite.com/core/press/articles/1998/10_00_98-team.html

[8] Pratt, W. K. "*Digital Image Processing"*, John Wiley & Sons, 1991.

[9] Reinhard, E. Ashikhmin, M., Gooch B. and Shirley, P., "Color Transfer between Images", *IEEE Computer Graphics and Applications*, September/October 2001, 34-40.

[10] Ruderman, D. L., Cronin, T. W. and Chiao, C. C., "Statistics of Cone Responses to Natural Images: Implications for Visual Coding", *J. Optical Soc. Of America*, vol 15, no. 8, 2036-2045, 1998.

[11] T.Welsh, M.Ashikhmin & K. Mueller, "Transferring Color to Greyscale Images", Siggraph, 2002.

[12] Mrs.V.Karthikeyani, Dr.K.Duraiswamy & P.Kamalakkannan, "Texture Analysis Experiments with Meastex and Vistex Benchmarks", National Level Conference, Tech Fete 2004, KSR College of Technology, Feb.,24,2004.

Name : Karthikeyani Vajravel
Address : Research Scholar, K.S.Rangasamy College of Technology, Tiruchengode, TamilNadu, INDIA.
Education & Work experience: B.Sc. (Mathematics)., MCA., & 11 years of teaching and 6 years of research experience.
Other information: Area of Interest are in the field of Image Processing, Computer Graphics, Computer Networks.

Name : Dr.K.Duraiswamy
Address : Dean (Acadamic), K.S.Rangasamy College of Technology, Tiruchengode, TamilNadu, INDIA.
Education & Work experience: B.E., M.Sc., (Engg)., Ph.D. & 36 years of teaching and 20 years of Research experience.
Other information: Area of Interest are in the field of Image

Processing, Computer Graphics, Computer Architecture, Parallel Processing, Computer Networks. 40 Papers Published in National/International Conferences, 25 Papers Published in National/International Journals, 7 Books are published.

Name: Kamakakkannan Palaniappan
Address : Research Scholar, K.S.Rangasamy College of Technology, Tiruchengode, TamilNadu, INDIA.
Education & Work experience: B.Sc. (Physics)., MCA., & 13 years of teaching and 6 years of research experience.

Other information: Area of Interest are in the field of design and evaluation of multicast routing, Medium Access Control and Quality of Service for Mobile Ad Hoc Networks, Sensor Networks, and Image Processing.