IJSTE - International Journal of Science Technology & Engineering | Volume 4 | Issue 3 | September 2017
ISSN (online): 2349-784X

Monochrome to Multichrome Image Automatization

Vishakha Ranjane
UG Student
Department of Computer Engineering
Rajiv Gandhi Institute of Technology Mumbai University, India

Shweta Salve
UG Student
Department of Computer Engineering
Rajiv Gandhi Institute of Technology Mumbai University, India

Twinkle Shah
UG Student
Department of Computer Engineering
Rajiv Gandhi Institute of Technology Mumbai University, India

Prof. Sumitra Sadhukhan
Professor
Department of Computer Engineering
Rajiv Gandhi Institute of Technology Mumbai University, India

Abstract

The paper is about Coloring a Grayscale Image automatically, colorization of images has many applications such as in CCTV Cameras, Astronomical photography, and electron microscopy, photographs of ancient history. In this proposed system, we will de-sign and implement system which color the images automatically with the help of training images. There consists of two types of images one is black and white image that is source image and second one is colored image that is training image then by doing appropriate color space conversion techniques, we get the required target image. In this col-or space conversion, we use machine learning techniques and image processing methods. Our objective is to build use friendly system which colorize the system in faster and in an efficient way.
Keywords: Image Processing, Machine Learning, K-Means Algorithm, SVM

I. Introduction

Image colorization consists in adding colors to grayscale images. Two approaches are mainly used: the first one consists in using manually pre-defined color inputs while the second consider an entire colored image as color example to transfer. The work presented here is in the second category. In this paper we have addressed one of the challenging problem from antique photography to low-resolution video, which lack color information. Assigning color to black and white image is a difficult problem.

- Example: Differentiating colors like red, blue, green, etc. along with adding textures and removing noises so that the image looks realistic in nature.

There are several vintage photographs that are colorized on the basis of colorization process require great financial and time cost. Here we present a method using machine learning techniques and image processing methods.

II. Literature review

Coloring a black and white image requires a lot of human interaction. To save our time we are presenting a machine learning method of automatic colorization of grayscale images. The resultant image obtained has aesthetically believable colors, using colormap selected from a similar training image.

We compared papers “Patch-based image colorization” by Aurelie Bugeau and Vinh-Thong Ta of 2012 [2] and “Variational Exemplar-based Image Colorization” by Aurelie Bugeau, Vinh-Thong Ta and Nicolas Papadakis of March 2013 [3] with our existing system [1]. In the first research paper of 2012 [2], a simple patch based image colorization method and general distance selection framework for color prediction have been used, using a training image. But the resultant image is not that realistic and image is desaturated. In the paper of 2013 [3], the input color data comes from a source image considered as a reference image. They have used a variational formulation, energy minimization along with patch based method to automatically select set of best possible colors. Now drawbacks of this paper is it does not properly work with old medias, it gives unwanted colors and also gives us desaturation effect. Now in the existing system [1], they have analyzed the color content of a provided training image and then attempt to predict colorization for a single target grayscale image on a per-pixel basis. Along with this they have used many methods such as K-means, Graph cut algorithm, SVM (Support Vector Machine) to make result accurate. The drawbacks in the first two paper does not seem to have in this system. That is the system deals well with ill posed nature of the problem and we get realistic result. So this system is better.

But there is just one problem, it does not distinguish objects globally. For example, the algorithm has difficulty in differentiation a mountain and sky.

To overcome this drawback of the existing system, we have proposed a better system.

III. Proposed system

In the existing system aside from graph cut minimization, there is no explicit feature used, that contain global information about the image. The algorithm thus cannot distinguish between pixels that locally look similar but globally are from different objects.

So to remove the disadvantage from existing system we will incorporate global features we will segment the image into broad regions such as foreground and background, and then include this assignment as an additional feature. Alternatively, each segmented region of the image can be colorized independently, recombined, and processed via graph cut.

IV. Hardware and software

A. Requirements

1) *Hardware requirements*
- 2 GB Ram
- Dual Core Processor
- INTEL core i3 processor and above.

2) *Software Requirements*
- WINDOWS 7 and above or Linux
- Python
- scikit-learn
- OpenCV libraries

V. System design

In order to execute our main goal, we use the following two processes:
- Training
- Colorization

Training consists of:
1) Taking Training image as input.
2) Transforming and discretizing the color space.
3) Extracting a set of features from each pixel.
4) Training a set of binary classifiers.

Colorization consists of:
1) Selecting the input image
2) Extracting the same set of features from each pixel.
3) Estimating the probability for each color at each pixel.
4) Optimizing the probabilities using Graph Cut algorithm.
5) Selecting the color with the highest probability.
6) Finally obtain the colorized image.

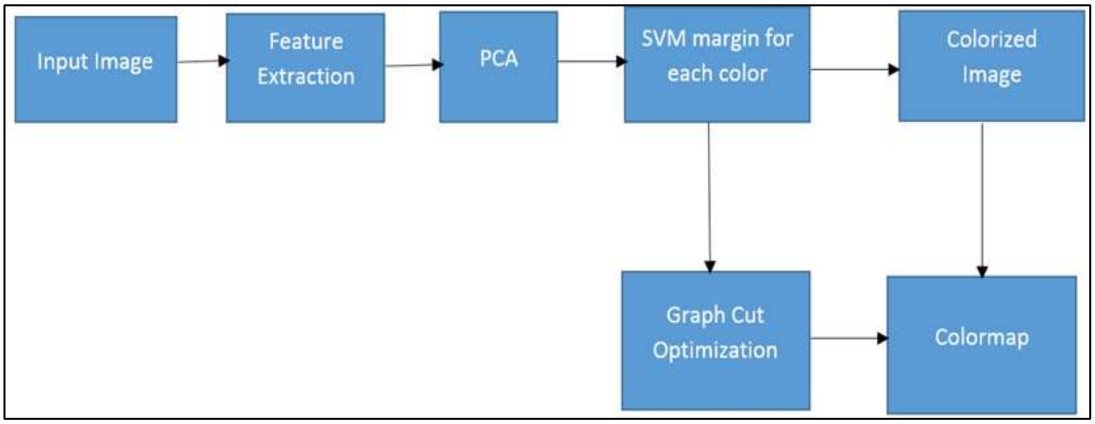
A. Color Space Discretization

Color space means the n-dimensional representation of a color. Usually we use 3 dimensional space and its coordinates are used to encode a color. There are many color spaces and the most commonly used is RGB color space. But we are using Lab color space in our system where L stands for luminance and a, b are just letters. Lab is used over RGB because Euclidean distance in Lab is more similar to how humans perceive color differences. Its Euclidean distance is calculated as given below:

$$\Delta E^*_{76} = \sqrt{(L^*_1 - L^*_2)^2 + (a^*_1 - a^*_2)^2 + (b^*_1 - b^*_2)^2}$$

Also it allows very fast workflow. L channel describes how dark or light a color is. L varies between 0 and 100. 0 means total darkness and 100 means maximum light. a describes whether a color is towards green or magenta, b describes whether a color is towards blue or yellow. Color information is then represented as the two 8 bit values (a, b). a and b can vary in the range of -128 and 127. So total possible colors will be 256^2. So in order to reduce this large number we are going to use K means algorithm. Now we obtain a reduced color map. Then next we select the reduced color space and then we quantize the training image and randomly select a number of pixels.

(a) Training



(b) Colorization

Fig. 1: Here we show flowcharts illustrating the steps taken in our training and colorization

B. Feature Extraction

Feature extraction is a term used with computers and machine learning. Feature extraction is a type of dimensionality reduction that efficiently represents interesting parts of an image as a compact feature vector. There are techniques for feature extraction. We are going to use 3 parameters:

- SURF (Speeded Up Robust Features)
- FFT (Fast Fourier Transform)
- Localized mean and variance

SURF descriptors for each pixel are extracted over three octaves, using a 20x20 window. Octave refers to a series of response maps of covering a doubling of scale. To this we add the magnitude of the 2D FFT over the same 20x20 grid. Finally, we calculate the localized mean and variance.

C. Classification

We use a set of Support Vector Machines (SVMs) for classification. Support vector machines (SVMs) are a set of supervised learning methods used for classification, regression and detection. This set of SVMs will perform a binary classification for each color bin and will estimate whether a pixel is of the corresponding color or not.

This type of classification is called One Vs all Multilabel classification. We use the margin of each SVM as a proxy for the probability of each color being the true color. These probabilities are then used in post-processing by the graph cut algorithm.

D. Colorization Process

Colorization begins by computing the 786-point feature vector for each pixel in the greyscale image. Feature vectors are used in machine learning because of the effectiveness and practicality of representing objects in a numerical way to help with many kinds of analyses. Feature can be gradient magnitude color, grayscale intensity, edges, area. These features are then projected onto the lower-dimensional basis computed using PCA in the training step. PCA is the principal component Analysis. PCA helps us to reduce the dimensionality of a data set consisting of many variables correlated with each other. Each reduced feature vector then passed to the set of SVM's from which we obtain the geometric margin of the feature vector for each possible color. SVMs are more commonly used in classification problem. The distance between the hyper plane and the nearest data point from either set is known as margin.

The graph-cut algorithm is used to find the minimum energy.

$$E_i = -\sum_j s_{c_j}(v_i) + \sum_{j \in \mathcal{N}(i)} g_j ||c_i - c_j||_2$$

Where,

Scj(vi) is the geometric margin of the feature vector at pixel i, vi for color classifier Cj

Second summation is over all neighbour pixels if i and includes the Euclidean distance in color space and gradient magnitude gj.

VI. Conclusion

As stated above, this project solves the prob-lems presented by the previous two papers making this system if not completely, yet full proof and able to differentiate between two different objects or surroundings in a picture. Furthermore, this algorithm per-forms well with given ill posed nature of the problem and shows promise that realistic re-sults are available.

REFERENCES

[1] http://cs229.stanford.edu/proj2013/KabirzadehSousaBlaes-AutomaticColorizationOfGrayscaleImag-es.pdf
[2] https://pdfs.semanticscholar.org/bee2/839f7cc501521681506883d8a663fdace459.pdf
[3] https://hal.inria.fr/hal-00803219/document