Grayscale Portrait Colorization using Optimization and MTCNN Face Detector

Naimul Haque, Dept. of CSE, Manarat International University, naimul@mamarat.ac.bd
Samin Shahriar Tokey, Dept. of CSE, Southeast University, samin.shahriar@seu.edu.bd

Abstract: Colorization of a grayscale image requires human interaction since perceiving different colors associated with different objects is intrinsic to human experience. In the scribble-based method, a user typically scribbles on a grayscale image with color hints and then the colors are spread across the borders of different regions throughout the image. This paper presents a way to reduce the human interaction in colorization of portrait grayscale image by detecting the interiors of the facial regions using Multi-task Cascaded Convolutional Networks (MTCNN) and putting predetermined color hints on those regions. The colors are then spread throughout the image using Optimization. This paper presents automatic colorization by combining these two technologies. Though the method is not fully automated as the user still have to pick colors for the color hints, the method has an artistic application and can be used in real time colorization for facial regions in scenes.

Keywords: Colorization, CNNs, MTCNN, Optimization, YUV

I. Introduction

Colorization is the addition of colors to a grayscale image which has never been seen in colors before. Coloring the image requires segmenting the image into regions, detecting the regions as objects and putting appropriate colors for them.

Traditionally, the task requires human-labeled color scribbles [1] [2] [3] on the grayscale image to propagate the colors across the different regions and throughout the image using optimization techniques [2]. This is still a tedious process since the artists have to put different color scribbles based on the type of objects they see on the grayscale image before inundating it with colors using optimization application.

This paper presents a simple method to put color scribbles on a grayscale portrait image automatically by detecting the facial landscape and facial regions before coloring the portrait. The scribbles with predetermined colors are instantaneously set to appropriate facial regions, which are detected using Multi-task Cascaded Convolutional Networks (MTCNN) [4]. Impressive performance in face detection and alignment is only the fruits of recent success in deep learning. The Convolutional Neural Networks (CNNs) [5] can be used to predict face and landmark locations in an unconstrained environment.

The reason for colorization being a daunting task is that it is expensive and time-consuming. Typically, an artist needs to identify different objects, segments the image into regions accordingly, and then assign colors to different regions.

The segmentation algorithms often fail to identify intertwined or complex regions, such as face and hairs where it is difficult to detect the boundary of such regions. In a later attempt, Colorization using Optimization [2] eliminates the need for precise segmentation for colorization. The system requires the user to scribbles color hints in the interior of the regions and set a constraint on the neighboring grayscale pixels. The constraint is based on the premise that neighboring pixels in space-time that have similar intensities should have similar colors. Then using standard optimization technique, the colors are propagated across the regions based on the constraints.

The technique works flawlessly not just on still images but also on motion pictures. Nevertheless, users are still required to choose which colors to put in the facial regions. The method tries to minimize this user interaction by automatically predicting the locations of the interiors of facial regions and putting appropriate color hints on them. In **section 2,** we have discussed the basics ideas and tried to show the methodology works. In **section 3**, we show the result and how the method works in practice.

II. Approach

A. YUV Color Space

YUV [6] is a color encoding system that represents a color image or video taking human perception into consideration. In YUV color space, the Y component gives the monochromatic luminance while U and V are the chrominance channels.

B. Colorization using Optimization

Initially, we start with Y which is the input grayscale image and can be assumed as the monochromatic luminance of our final output colored image in YUV color space. The problem is to find the U and V chrominance components for the information in Y.

Let r = (x, y, t) be the position of a pixel that needs to be colored where x and y are the spatial location and t is the time in the frame location for the motion image. The neighboring pixel is that pixel which is in the neighbor pixel N(r) set of r.

The important part is to set the constraint on neighboring pixels r, s such that these two pixels get the same color if Y(r) and Y(s) are similar [2]. We need to minimize the error function based on the weighted average colors of the pixels neighboring to the pixel r. The colors $u(r_i)$ and $v(r_i)$ are set by users for the locations r_i in the grayscale image. The problem is to find $u(r_i)$ and $v(r_i)$ colors without the need for users in the portrait colorization. The images we are interested in are the images with human faces and we need to predict the locations of different facial regions automatically.

C. Face detection and Alignment

Deep learning [5], especially Convolutional Neural Networks (CNNs), have achieved much success in tasks such as image classification and face detection. Before deep learning techniques were available, the researchers ignored the apparent correlation between face detection and face alignment and considered them separate tasks.

In the paper, Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks [4], the authors combined these two tasks using unified cascaded CNNs by multi-task learning. The proposed framework is completed in three stages. Firstly, the framework produces candidate windows quickly through shallow CNN. Next, a large number of windows are eliminated which might not contain any faces using a more complex CNN. Lastly, a more powerful CNN refines the result again and outputs five facial landmarks positions. Namely, we get the positions of interiors of eyes, nose and two ends of mouth.

D. Portrait Colorization using Optimization and MTCNNs

We obtain the positions of five landmarks positions using the pre-trained MTCNN Face Descriptor and take the positions of eyes and find the middle position in between them, which is assumed to be the forehead of the subject, also find the middle position of the two ends of the mouth which is presumably the interior position of the lips. Color hints are added automatically to the pixel locations with predetermined colors. After the preprocessing, the grayscale portrait with color hints is colorized using Optimization and get the components U(r) and V(r).

III. Results

In our first attempt, we demonstrate a classical way to color a grayscale portrait using human labeled color hints. We take a black and white portrait from the Helen dataset [7] and it is shown in **Fig.1.**

Fig.1. A grayscale portrait from Helen dataset.

Using a coloring tool we manually put color hints on the portrait in **Fig.1** and run the optimization algorithm and colored image in **Fig.2**.

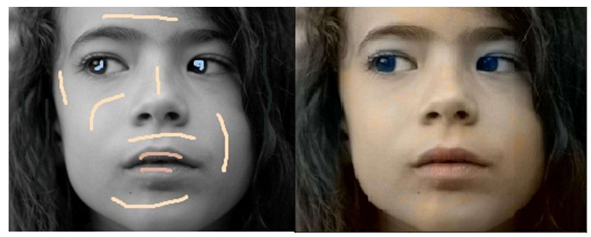

Fig.2. On the left is the scribbled image and on the right is the colored image using optimization.

Next, we run the image through the pre-trained MTCNN face detector and get the locations of the facial landmarks. With some preprocessing we get two more landmarks and add appropriate color hints to each of the locations to obtain hinted image in **Fig.3**.

Fig.3. The scribbled grayscale portrait using pre-trained MTCNN face detector.

Finally, using the optimization technique the image is colored again and the final result shown in the **Fig.4**.

Fig.4. On the left is the hinted image using MTCNN face detector and on the right is the colored image using optimization.

The MTCNN also detects the facial region and outputs a rectangular box around the subject's face which can be used to add color hints outside the facial boundary. IN **Fig.5** and **Fig.6** such an attempt is taken. In Fig.5 ten color hints with the color of average hair color are added outside the facial region and in **Fig.6** random colors are added to have an artistic look.

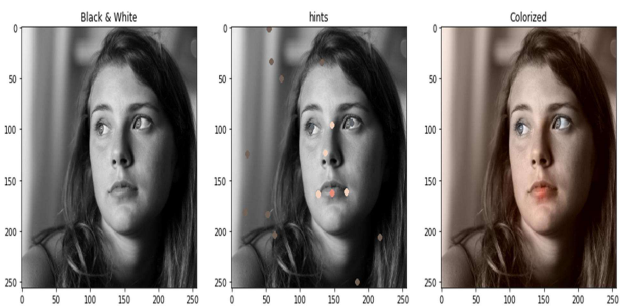

Fig.5. Result when hair color hints are added outside the facial region.

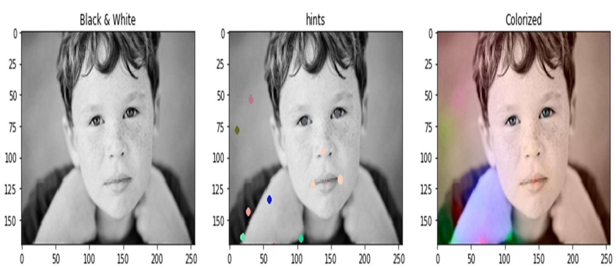

Fig.6. Result when random colors are added outside the facial region for an artistic look.

IV. Conclusions

The union of the two previous technologies leads us to automate colorization after putting color hints inside the interiors of the facial regions without human-interaction. Although complete automation can only be possible if the color hints could have been predicted. The idea that different subjects may have different eye color, skin color, etc. hence the color hints should be predicted based on the facial information, which can be achieved by further adding a separate CNN network to predict those phenotypes from the portrait. A face descriptor can be used to get the vector representation of the subject's face and using this to train the CNN network that can predict the colors of these four distinct colors. Even though the method doesn't provide robust results, the method opens up new idea to color an image. The method can be applied to manga or color comics arts where if MTCNN is trained to detect manga characters, may allow to color them instantly.

V. References

[1] Y.-C. Huang, Y.-S. Tung, J.-C. Chen, S.-W. Wang, and J.-L. Wu, "An adaptive edge detection based colorization algorithm and its applications," in Proceedings of the 13th Annual ACM International Conference on Multimedia, ser. MULTIMEDIA '05, 2005, pp. 351–354.

[2] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," in ACM SIGGRAPH 2004 Papers, 2004, pp. 689–694.

[3] Q. Luan, F. Wen, D. Cohen-Or, L. Liang, Y.-Q. Xu, and H.-Y. Shum, "Natural image colorization," in Proceedings of the 18th Eurographics Conference on Rendering Techniques, ser: EGSR'07, 2007, pp. 309–320.

[4] K. Zhang and Z. Zhang and Z. Li and Qiao, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," in IEEE Signal Processing Letters, 2016, pp. 1499-1503.

[5] Ian Goodfellow, YoshuaBengio, Aaron Courville, "Deep Learning", MIT Press,http://www.deeplearningbook.org, 2016.

[6] YUV, url: https://en.wikipedia.org/wiki/YUV, accessed: 26.1.2018 .[7] Helen dataset, url:https:ifp.illinois.edu/~vuongle2/helen/:,accessed: 29.10.2018 date).