Stanford University

Changing people’s hair color in images

Clemens Macho

cmacho@stanford.edu

Video: https://youtu.be/uJSWYr131aQ

Introduction

Many popular apps provide functionality for changing the hair color of people in images. We designed and implemented a variation of the cycleGAN architecture [2] that learns this task from a dataset of real photos of people with various hair colors. The photos are labeled with RGB values corresponding to the (average) hair color in the image. Our generator takes an image and a target color as input and outputs the same image but with the hair in the image changed to the target color.

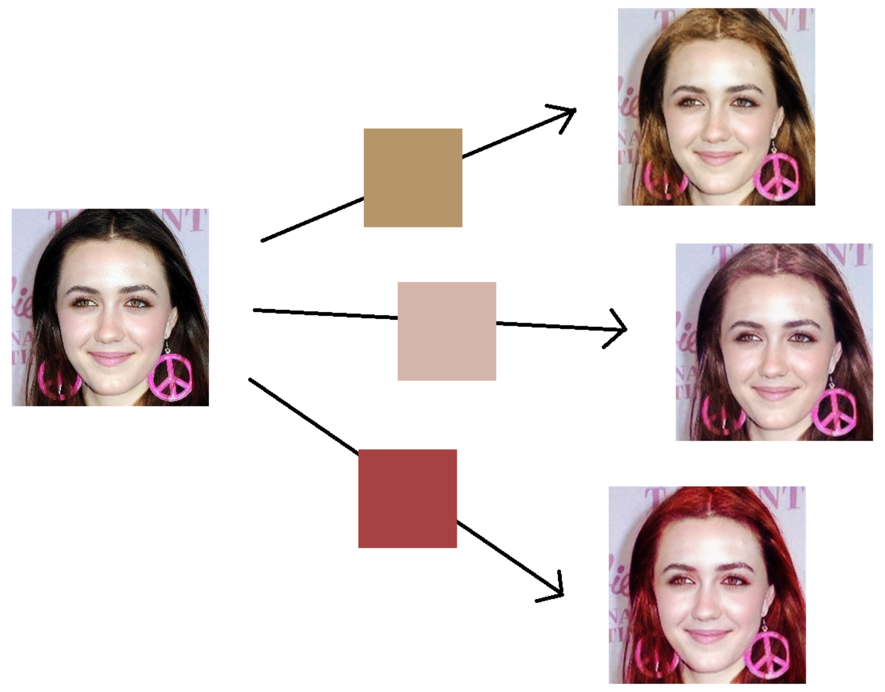

Figure 1: examples of hair color transformation with haircolorGAN

GAN training

Our haircolorGAN architecture is shown in figure 3 to the right. In contrast to cycleGAN, it uses just one generator and one discriminator, but both of them take hair color labels as input in addition to images. The discriminator must learn to distinguish between real images labeled with their actual hair color and generated images labeled with the target hair color. In order to successfully fool the discriminator, the generator must learn to match the target hair color in the generated image. To complete the cycle as in cycleGAN, generated images are fed back to the generator with the original hair color as target hair color and the output is compared to the original image.

The "identity loss" shown in the red square in figure 3 is an additional way of training the generator: When the target color is the same as the original hair color, then the image should remain unchanged.

An architecture similiar to haircolorGAN is described in starGAN [1] which inspired my method. However, the input/output behavior of the discriminator is different.

Results

After training, the model does a good job of changing the hair color. Consider the examples in figure 4 on the bottom right of the poster. These were hand picked from the examples generated during training.

Unfortunately, the transformation does not always work that well. In some cases it only changes the color of parts of the hair. Some of these examples are shown in figure 2.

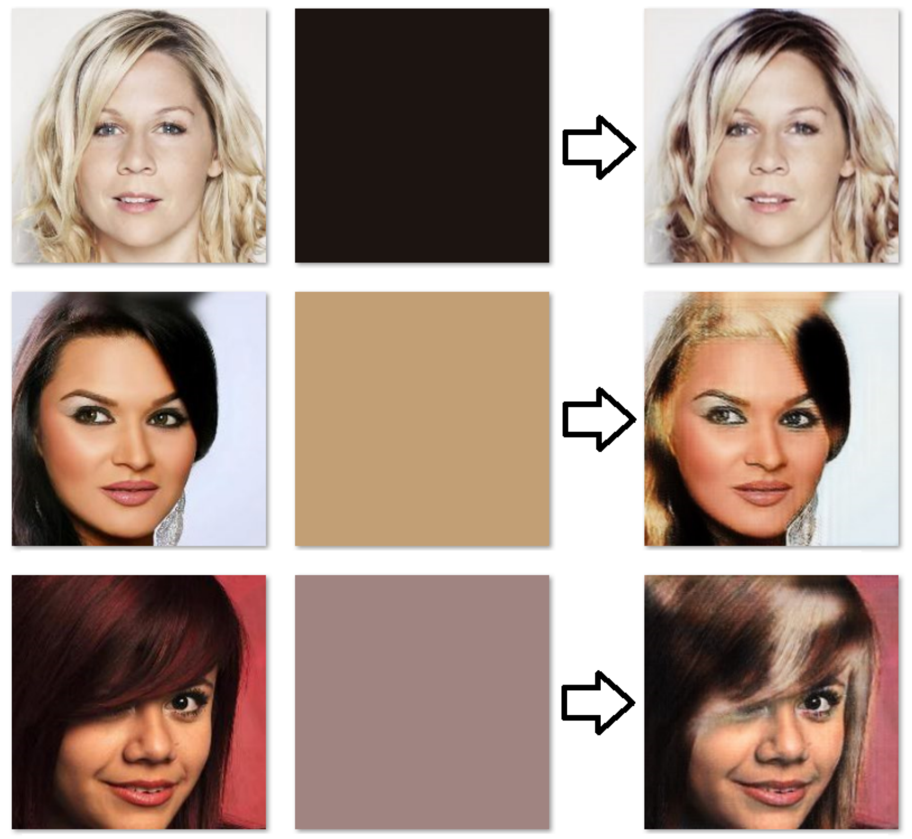

Figure 2: Some examples where only parts of the hair were changed.

In other cases the hair color remains completely or almost completely unchanged. In order to give a sense of how well the model works or doesn’t work, I counted for 100 random examples, how often the transformation worked and how often it did not. The results are summarized in the following table.

transformation worked well	55
only parts of hair changed	6
no change necessary because target hair color ≈ color in input image	9
transformation failed	30

References

[1] Y. Choi, M. Choi, M. Kim, J. Ha, S. Kim, and J. Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. *CoRR*, abs/1711.09020, 2017.

[2] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *CoRR*, abs/1703.10593, 2017.

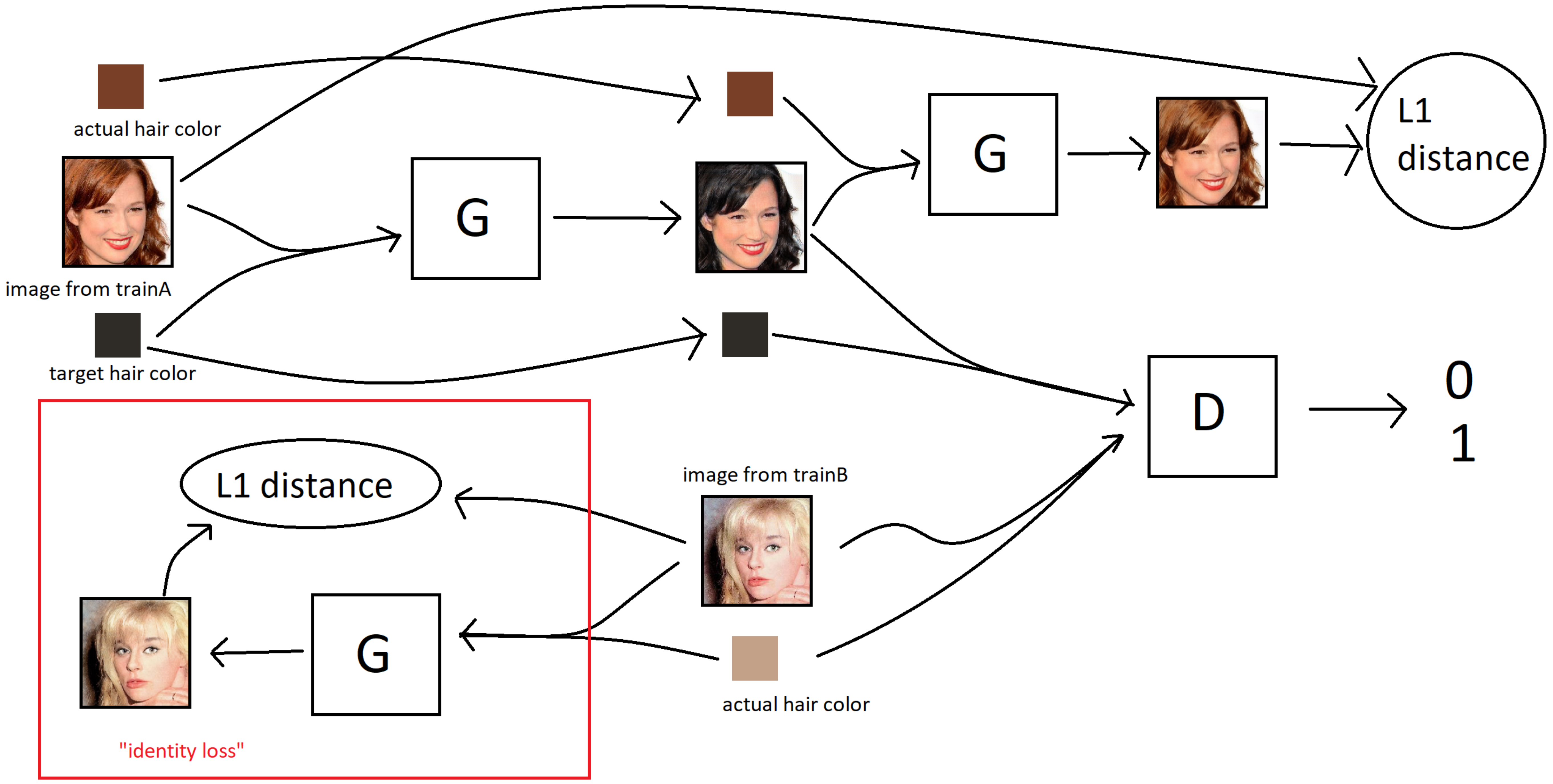

Figure 3: haircolorGAN architecture

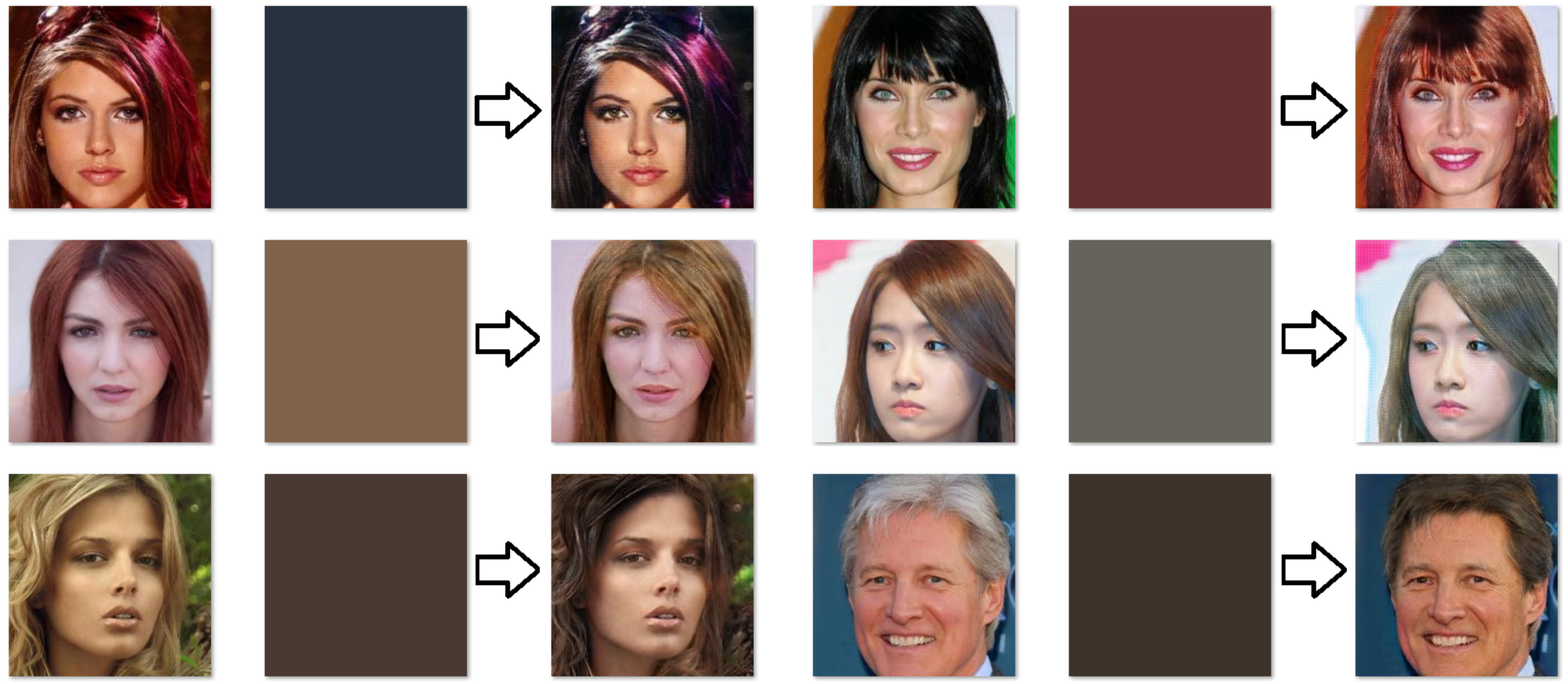

Figure 4: some examples generated during training (hand picked)