

Automatic Colorization of Grayscale Images

Team Members Group-15:

- Çağlar Çankaya
- Duygu Nur Yıldız
- İsmail Şahal
- Sena Korkut
- Yavuz Faruk Bakman

The Problem: In our project, we want to achieve automatic colorization of grayscale images. Colorization of grayscale images is an important problem because the photos before the invention of color photograph are grayscale and converting these photos into colorized photos without using machine learning algorithms might be very hard. Therefore, we want to propose a model or models that might solve this problem easily. We will try to convert some grayscale landscape images of Ankara city to colorized version. We will select a category of images intentionally because in the training, we will use only grayscale images and colorized versions of these images as input. We will not use text description of image or class labels in the image. As the colorized version of a grayscale image is not unique, for instance, training the model with landscape photographs and testing it with people's selfies might be a disaster. Also, other automatic colorization works use the same category technique for that problem.

The Dataset: For this project, it is relatively easy to find dataset because we don't need any label at all. All we have to do is to find landscape photos of Ankara city and convert them into grayscale format by existing libraries in Python such as Pillow. To find landscape photos of Ankara, we will use Flickr website which is a platform for sharing photos. In that website, we will search #landscape and #Ankara tags together and collect these images. Although number of images we will collect is not determined yet, when other works about this problem are examined, approximately 200 images will be enough for the project and there are 2000 tagged images in Flickr.

Planned Milestones: In the project, there are various algorithms for this task. For instance, CNN, GAN, WGAN, SVM techniques can be used for this problem. We might use different algorithms that we can compare their performance. Also, we can extract some hints from the input and use it for better prediction. For instance, some colors might be more frequently used in the landscape photos such as green. If we get this type of information from the training data, we can get better performance in the test by using that information wisely. We want to create a program having graphical user interface for the users. A user can upload a grayscale image of Ankara landscape photo and get colorized version of it. Also, another user can train a model from scratch by different category of images and get the trained model. With this model, a user can upload grayscale images in the category user trained and get the colorized versions of them.

