**Autoencoder**

**Introduction**

The second method of image colorization is mainly centered on an autoencoder system. While the model primarily constitutes an encoder and a decoder, it also comprises of a very powerful image classification network ‘InceptionResNetv2’ used for transfer learning. The Inception model has been trained on over a million images from the ImageNet database and helps in classifying a particular image into one of 1000 categories. The combined model, as such comprises of the Inception network, the encoder model, their merged model, and a decoder model.

**Architecture**

The encoder model starts with an input layer of (,256,256,1) and further contains eight two-dimensional convolution layers. Each layer has a 3-by-3 kernel, with strides to reduce dimensions in alternative layers. A ‘category’ model takes in an input of size (,1000), after which it has a repeat vector and a reshape layer with arguments (1024) and (32,32,1000) respectively.

A merge model is then constructed, which receives its input in the form of combined outputs from the two previous models. It is followed by a convolution layer with a (1x1) kernel. The next model is that of the decoder, which comprises of six convolution layers with (3x3) kernels along with three up-sampling layers. Every padding argument used in these models is ‘same’, while every convolution layer except the final layer uses ‘relu’ as its activation function. The final layer utilizes ‘tanh’ as its activation function. The final model combines the inputs of the encoder and categorize models with the outputs of the decoder.

**Methodology**

The encoder model takes in this input and extracts few important features through its convolution layers. Simultaneously, the inception model is used to classify the underlying image as one of 1000 categories after converting the image into the required shape of (299, 299, 3) and then preprocessing this input to the scale of [-1,1]. The predicted feature vector is then sent into a ‘categorize’ model which repeats this input (32x32) times and reshapes it into a format having the same dimensions of features as the encoder model.

The resultant outputs from the encoder and categorize models are (32,32,256) and (32,32,1000) respectively. These are then combined in the merge model with the help of a concatenate layer, after which a convolution layer is used to convert it back into the shape of the encoder output. The output from this model is consequently sent into a decoder model, which up-samples the received input into producing the predicted ‘a’ and ‘b’ values, which contain information on the colors of the image. The combined model takes as its input the shape of the encoder input (256,256,1)

An ‘Image Data Generator’ is used to provide numerous iterations of the same image. Each generated batch is then used to obtain the categorical features predicted by the inception model. After converting each batch into the ‘lab’ format, the obtained feature vectors are merged with ‘l’ value of the image to be provided as its input. The output is the combination of ‘a’ and ‘b’ values of the image. The respective inputs and outputs are then yielded to train the combined model.

To test the model, test images are converted from ‘RGB’ to ‘lab’, after which their ‘l’ values and predicted feature vectors are passed to the model to predict their ‘a’ and ‘b’ values. The collection of ‘l’, ‘a’ and ‘b’ values are converted into ‘RGB’ to obtain the predicted color image.

**Results and Limitations**

The model predicts a good estimate of the input image and its colors based on the classification. However, it tends to associate specific elements to a specific color, while it may not always be the case. For example, if a photo of a canopy of trees is passed as for prediction, the obtained result will most likely be a set of green trees, although there are many instances of yellow or orange being the predominant color in the original image. Moreover, the encoder also has an issue with identifying the right boundaries of each value, which may lead to some undesirable overlap.

Szegedy, Christian, et al. "Inception-v4, inception-resnet and the impact of residual connections on learning." *Thirty-first AAAI conference on artificial intelligence*. 2017.

Baldassarre, Federico, Diego González Morín, and Lucas Rodés-Guirao. "Deep koalarization: Image colorization using cnns and inception-resnet-v2." *arXiv preprint arXiv:1712.03400* (2017).

|  |  |
| --- | --- |
| An, Jiancheng, Koffi Gagnon Kpeyiton, and Qingnan Shi. "Grayscale images colorization with convolutional neural networks." *Soft Computing* 24.7 (2020): 4751-4758. |  |

Hwang, Jeff, and You Zhou. "Image colorization with deep convolutional neural networks." *Stanford University, Tech. Rep.*. 2016.