University of Toronto

Faculty of Applied Science and Engineering

Final Report

Date	Aug 15, 2019
Project Title	Image Colourization
Teaching Assistant	Farzaneh
Prepared By (Name and Student #)	Zijian Wang Cheng Peng Yiming Li
Word Count	1856 words

Table of Contents:

1.0 INTRODUCTION

Colourization is the process of adding plausible colours to grayscale images. Many graphic designers can colourize grayscale images using Photoshop, yet this process is time-consuming and requires skills and artistic talents to create appealing images with accurately colour schemes.

This project's motivation is to create an automatic method that colourizes grayscale images in a short period of time. This problem is challenging since a single grayscale pixel value may correspond to many possible colours, for instance, red and green balloons have similar shades in a grayscale picture (Figure 1.0.1). As a result, traditional methods rely on user inputs alongside a grayscale image and these methods are considered to be semi-automatic. In recent years, machine learning techniques, such as deep learning, has been proven to be effective in solving image colourization problems; it can colourize images without user inputs, thus creating a true "automatic" approach.

Machine learning is an appropriate tool for this task because this problem can be framed into either a regression task or classification task; given the grayscale image and predict colour channel values or compute the pixel-wise probability distribution over a set of different colours. We propose to use a Convolutional Neural Network to implement both methods, and the models and results will be presented in subsequent sections.

The goal of the project is to implement a model to automatically colourize grayscale images, and the output should have an appropriate and accurate colour scheme without compromising the image quality.

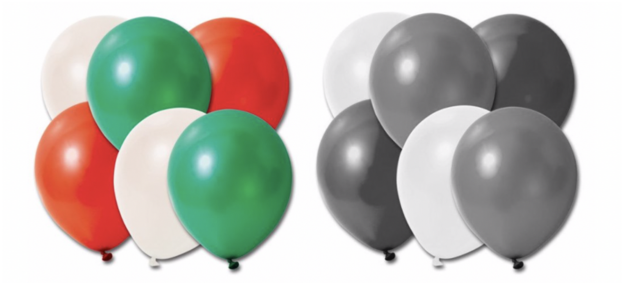


Figure 1.0.1: Coloured and grayscale Pictures of Red and Green Balloons

2.0 MODEL ILLUSTRATION

The basic model for the project is a convolutional autoencoder (Figure 2.0.1), which consists of a convolutional network and a deconvolutional network. The regression model and the classification model are variations of the basic model.

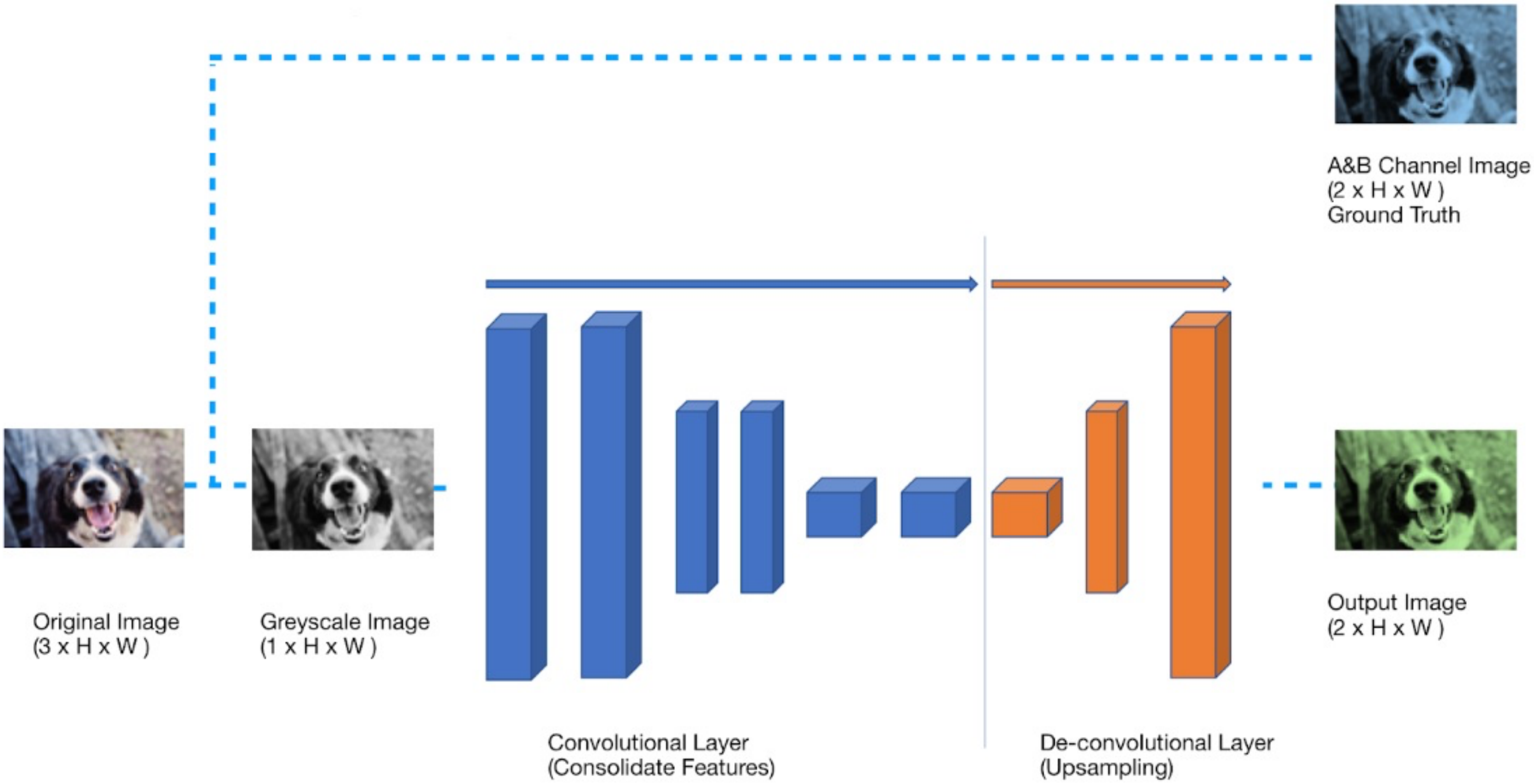


Figure 2.0.1: Illustration of the Basic Machine Learning Model

3.0 BACKGROUND AND RELATED WORK

There are several successful machine learning models that can automatically colourize grayscale images. These methods differ in how the problem is framed and the way of encoding image data [1]. Those algorithms can be classified into parametric and non-parametric[1]; parametric methods use regression to predict pixels' colour channel values as proposed by *Deshpande et al* [4]. In addition, the classification of the pixel's colour as proposed by *Zhang et al* [1,5] also generates satisfying results.

4.0 DATA PROCESSING

Data is selected from a subset of MIT Places dataset called "*The full-sized images for the test set of Places205*" [6]. The dataset consists of 41,000 images of size 256x256 pixels. The dataset is then divided into training, validation and test sets containing 30,000, 10,000 and 1,000 images respectively. Data augmentation is applied to the training and validation sets, and the images are cropped to 224x224 pixels and randomly flipped.

4.1 DATA PROCESSING FOR REGRESSION MODEL

The regression model manipulates the image data in the LAB colour space [8] which consists of 3 colour channels; L channel encodes illuminance (lightness), and A & B channels encode chrominance. The LAB colour space is preferred over the RGB colour space as it preserves the texture of the grayscale image and it is ideal for machine vision implementation and computer processing [8]. Finally, a customized image data loader is generated and later used in training. Usage of the customized image folder allows batching the colour-space tensor beforehand and it also saves training time compared to slicing the image tensor in the training loop.

4.2 DATA PROCESSING FOR CLASSIFICATION MODEL

The input to the classification model is the same as that of the regression model, but the ground truth has to be changed to a probability distribution in order to implement the classification model. Therefore, the training data has to be labelled with colour categories through colour quantization which is a process of representing images with a limited number of colours. For instance, the image in Figure 4.2.1 can be represented by a tensor of size 3xWxH using RGB values, and it can be further encoded using indices of colour categories, thus reducing the tensor size to 1xWxH. This project uses files from CSC421 Assignment 2 [12] which contain information on 16, 24 and 32 colour categories. These colour categories are chosen using K-mean clustering [17] over colours and selecting cluster centers [12]. All the images in the training set and validation set are encoded with indices of colour categories and the classification model compute the pixel-wise probability over them.

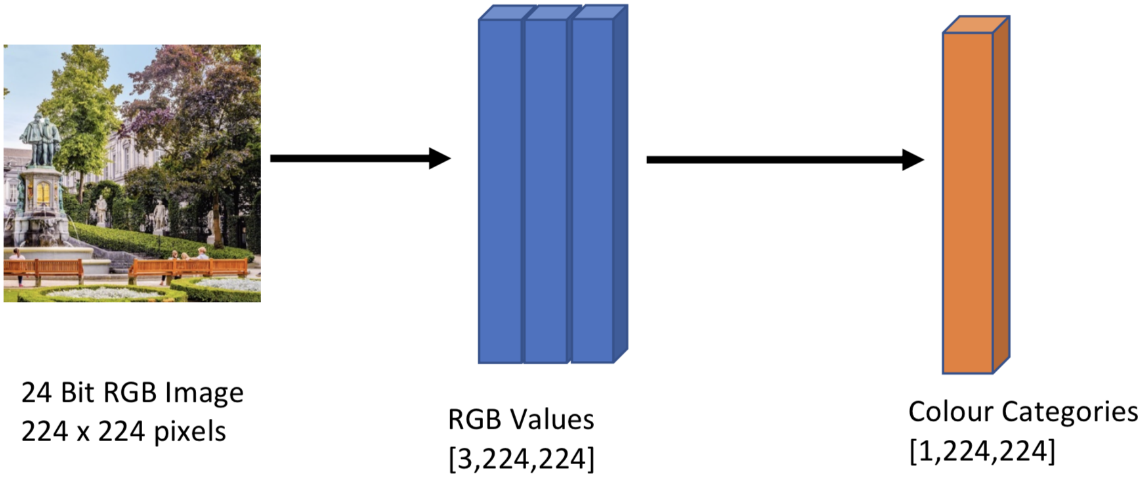


Figure 4.2.1: Illustration of Colour Quantization

5.0 ARCHITECTURE

This section will present the models used for regression and classification. Both models are based on the basic models in section *2.0 Model Illustration*.

5.1 REGRESSION MODEL

The Regression Model consists of two parts: the first 10 layers from a pre-trained ResNet18[10] and 4 convolutional layers. The input is a grayscale image with one L channel, and the output is tensor of size 2 x 224 x 224 containing A&B channel information. A pre-trained ResNet18[10] is adapted to accelerate the training and improve accuracy, and it extracts mid-low level features from the input. The convolutional layers use kernels of size 3, and each layer is followed by an upsampling layer with a scale factor of 2 and a batch normalization layer [15]. The model uses ReLU [14] as the activation function.

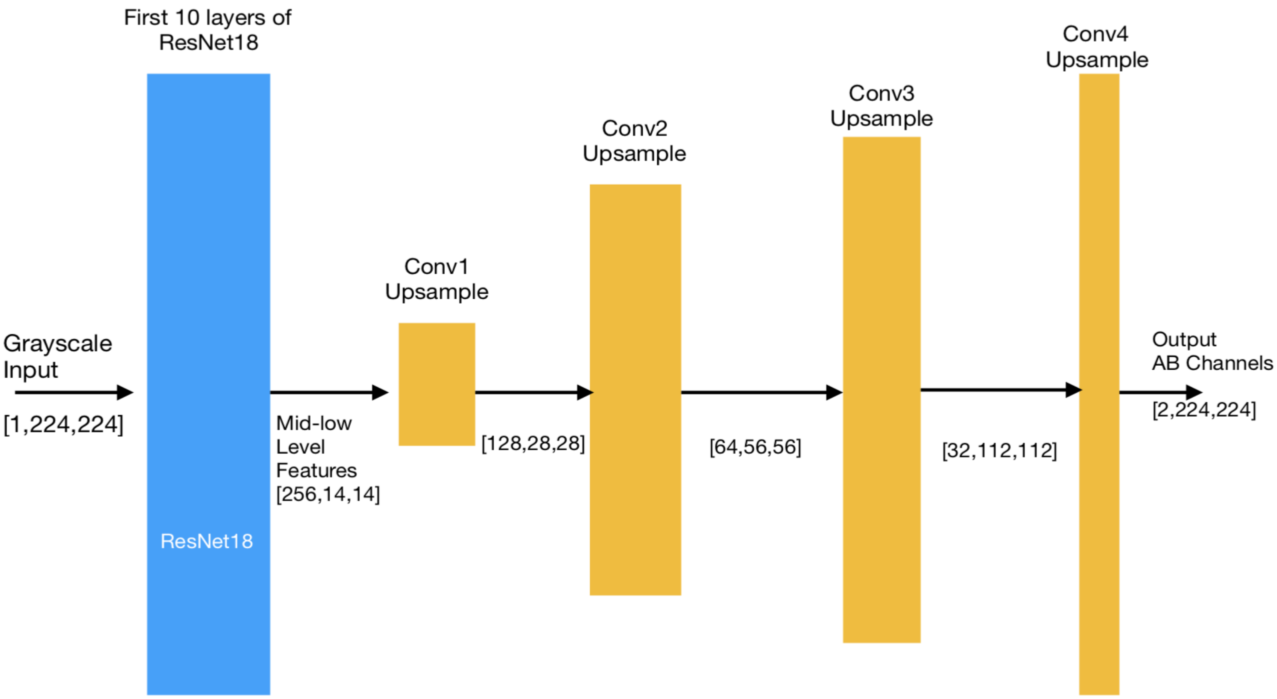


Figure 5.1.0: Regression Model Using Transfer Learning

5.2 CLASSIFICATION MODEL

The classification model utilizes a U-Net [13] architecture where skip connections are added to the convolutional network and the deconvolutional network. The dimensions of the input and output tensor are shown in the image, and *num_colours* represents the number of colour categories (num_colours can take on values of 16,24, or 32). In addition, there is batch normalization after each convolutional layer, and finally, the model uses Leaky ReLU [14] as the activation function.

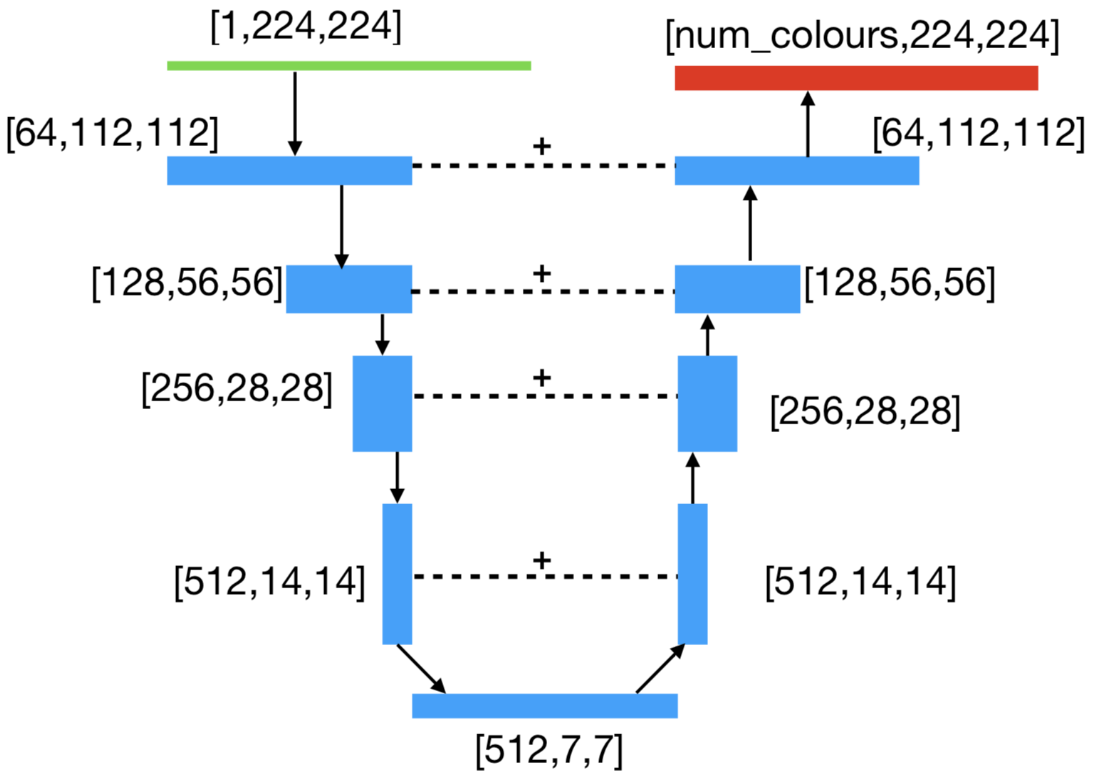


Figure 5.2.0: Classification Model with U-Net

6.0 BASELINE MODEL

The baseline model is a simplified version of the basic model proposed in section *2.0 Model Illustration*, and it has 1 convolutional layer and 1 deconvolutional layer (Figure 6.0.1). The model predicts AB channels using regression. The sample result is shown in Figure6.0.2, and it serves as the minimal requirement to compare the proposed models with.

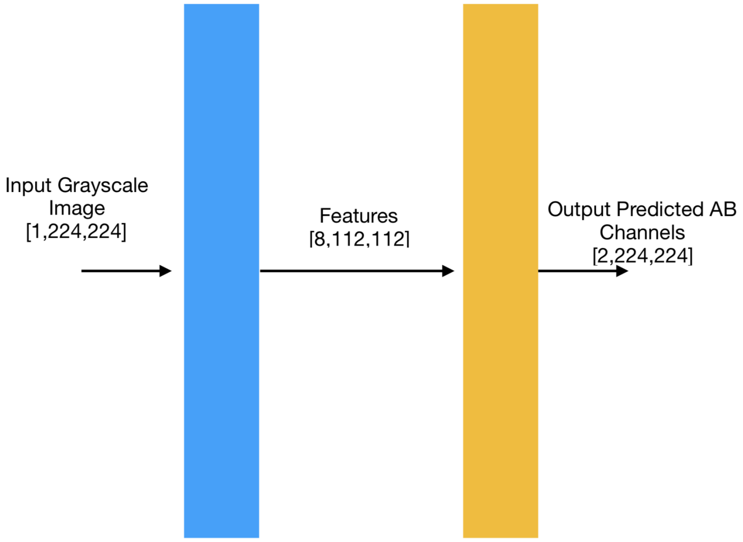


Figure 6.0.1: Baseline Model

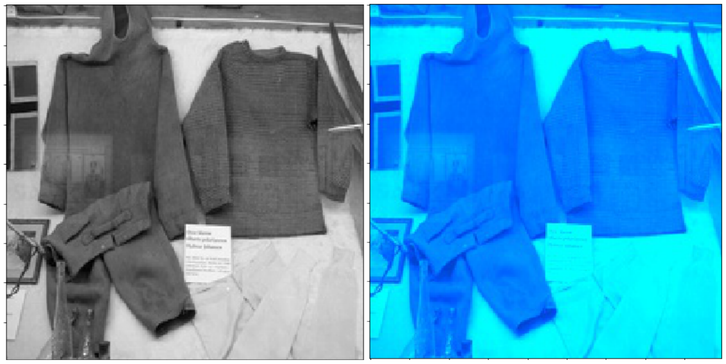


Figure 6.0.2: Sample Output From the Baseline Model

7.0 QUANTITATIVE RESULTS

Quantitative measurements that are appropriate for this project should measure the pixel-wise colour similarities between the original image and the colourized image. The group introduced two measurements, Image Similarity and Image Difference, as percentage correlation indicators. Image Similarity can be derived from Image Difference, which is equal to the average of pixel-wise normalized RGB value differences. Table7.0.1 shows the image similarities between the output images produced by both models and the original images.

$$ImgDif = \forall Pixel(\sum \frac{\|\Delta RGB\| /255}{3 \times W \times H})$$

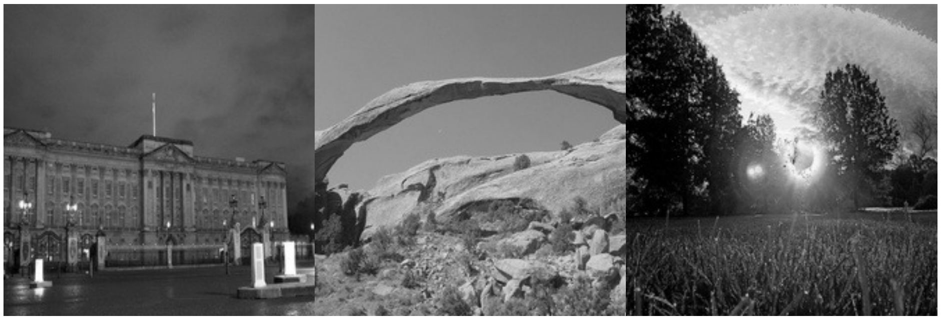
$$ImgSim = 1 - ImgDif$$

Number of Epochs	ImgSim-Regression Model	ImgSim-Classification Model
100 Epochs	65.39%	52.45%
200 Epochs	66.88%	64.31%

Table 7.0.1: Image Similarities of Regression Model and Classification Model

8.0 QUALITATIVE RESULTS

The following images in Figure 8.0.1 are randomly chosen from the test set to showcase the performance of the regression model and the classification model. In addition, Figure 8.0.2 shows some of the best outputs from the classification model.



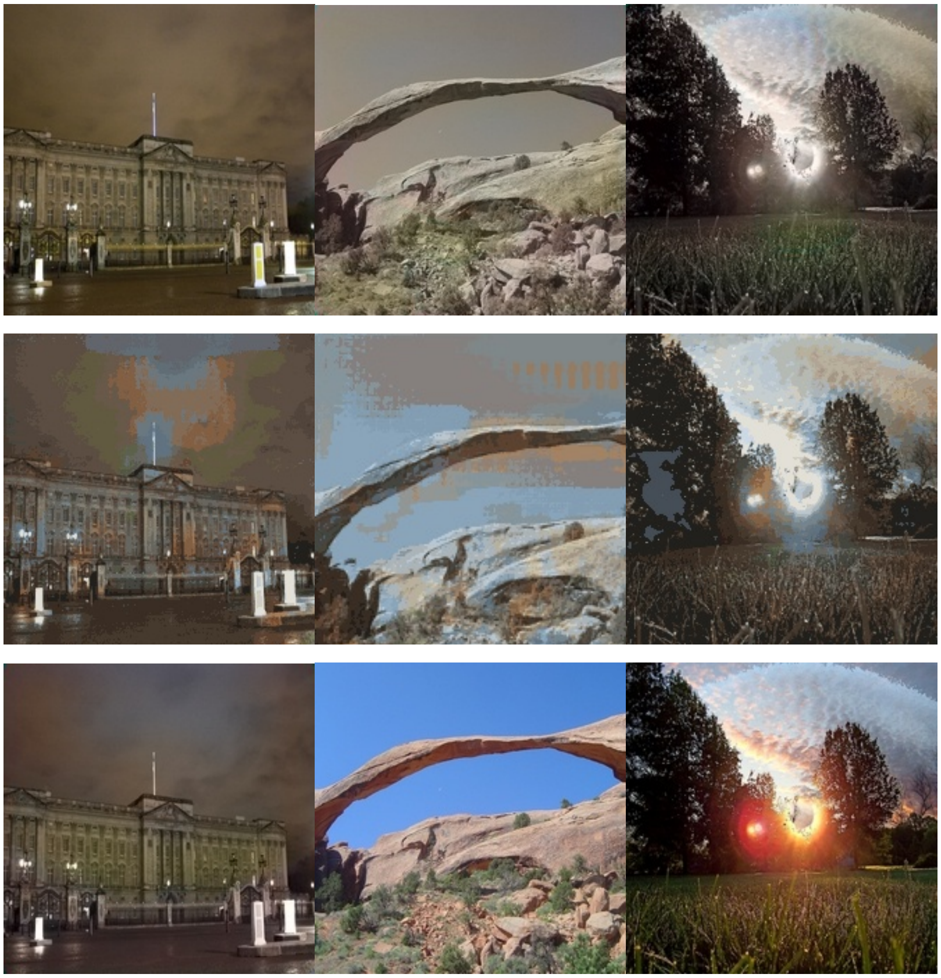
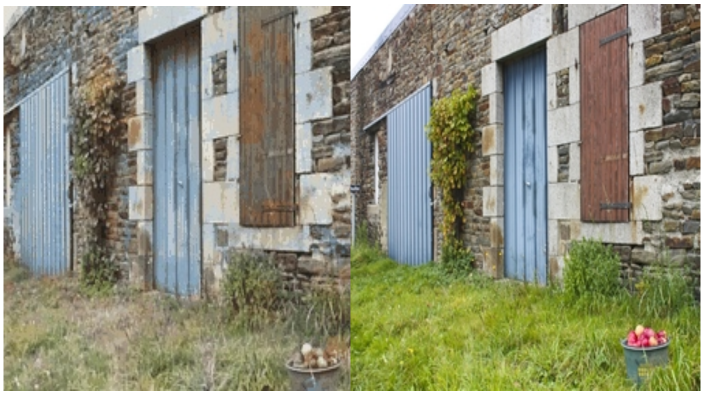


Figure 8.0.1: Sample Outputs, From Top: Grayscale Image, Regression Model Output, Classification Model Output, Original Image



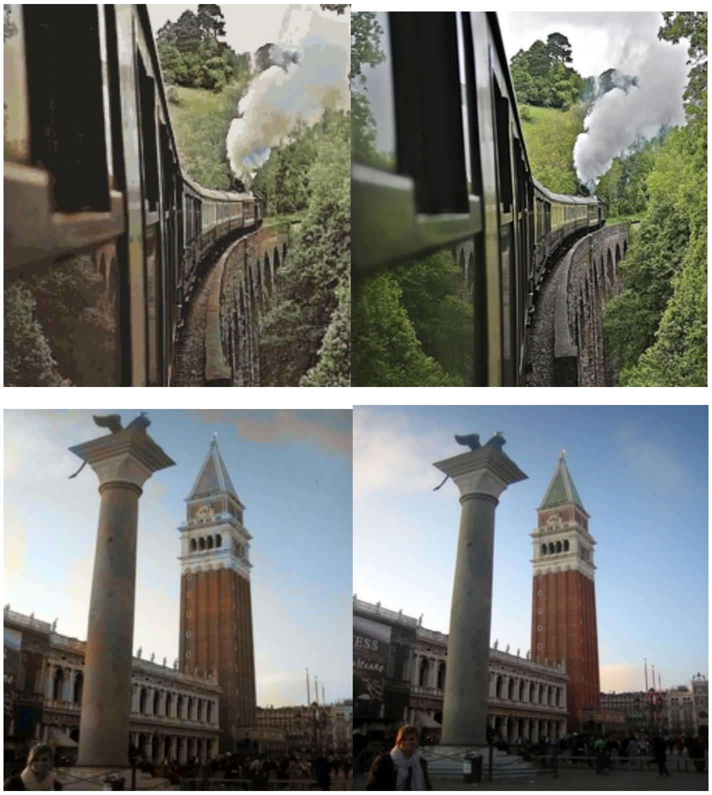


Figure 8.0.2: Best Sample Output from Classification Model, From Left: Colourized Image, Original Image

9.0 DISCUSSION

The results we obtain from the two proposed models are satisfactory, as both methods are able to generate plausible colours for different objects in the images. As shown in Table 7.0.1, the regression model achieves 66.88% similarity after 200 epochs compared to 64.31% of the classification model. However, the quantitative results do not provide a comprehensive justification, there are some differences in terms of colour scheme and output quality between these models.

9.1 REGRESSION MODEL

The output of the regression model is synthesized from the predicted AB channels and the input L channel (the grayscale image), thus the model is able to preserve the texture of the input image well, and capture the changes in colours. However, the regression model is trained using mean square error loss function, thus the output suffers from the averaging effects. The loss function would penalize harshly on vibrant colours (extreme values of AB channels), so the model tends to predict colours conservatively.

9.2 CLASSIFICATION MODEL

The classification model is developed in an attempt to reduce the averaging effects and produce vibrant colours. As shown in the best case output (Figure 8.0.1), the classification model learns green and blue colours well, and it performs better on natural scenery. The model also learns how to colourize buildings and other objects after more training. The downside of this model, however, is that there is inconsistency in the colour palette. For instance, the colour of the sky in the image of the parliament building (left column of Figure 8.0.1) gradually changes from black to blue toward the centre of the image. The classification model is not able to recognize the subtle changes, instead, it maps different colours in the same colour category, resulting in the ring pattern in the centre of the output image.

In conclusion, even though the regression model performs better in quantitative tests, the coloured images produced by the classification model are more vibrant and appealing.

9.3 CHALLENGES AND DIFFICULTIES

The number of colour categories used in colour quantization plays a critical role in the training of the classification model. As the number of colour categories increases, the model could potentially learn more colours thus produces more accurate results, but it also requires more computational power. We tuned this parameter when we trained the classification model, and the results can be seen in Figure9.3.1.

Additionally, as mentioned in section *1.0 Introduction*, a single grayscale value could potentially correspond to multiple colours, and this is called the multi-modality effect [8]. It creates difficulties when examining the colourized output images since the model output's colours can be different from the ground truth, but they are not necessarily "wrong" colours. In order to produce the colour as close as the ground truth, we use post-processing by manipulating the classification model output to take the top 3 predictions and average their RGB values. The results are shown in Figure 8.0.2.

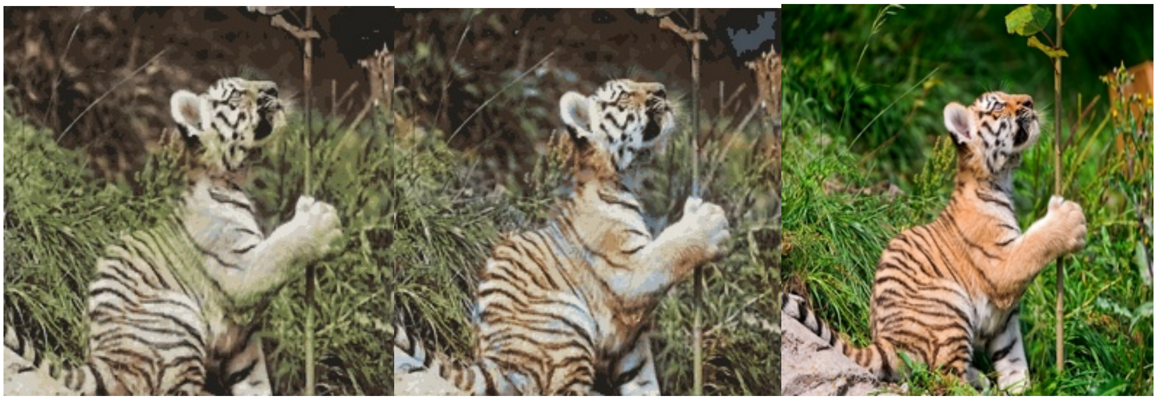


Figure9.3.1: Effect of Different Number of Colour Categories, From Left: 16 Categories, 24 Categories, 24-Bit RGB Image

10.0 ETHICAL CONSIDERATIONS

The model's limitations exist in terms of Fairness as Equalized Odds [16] for people of different racial backgrounds. Our training data contains mostly natural scenery, and it is not representative of other objects, such as people's faces. Therefore different people's images(e.g. different skin colours) will be coloured inaccurately. Pre-processing techniques can be applied to the dataset to balance the number of people from different racial backgrounds and reduce the effects of bias.

Reference

[1] Zhang, R., Isola, P. and Efros, A. (2016). *https://arxiv.org/pdf/1603.08511.pdf*

[2] Cheng, Z., Yang, Q., Sheng, B.: Deep colourization. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 415–423

[3] Dahl, R. (2016). *Automatic colourization*. [online] Tinyclouds.org. Available at: http://tinyclouds.org/colorize/

[4] Deshpande, A., Rock, J., Forsyth, D.: Learning large-scale automatic image colourization. In: Proceedings of the IEEE International Conference on Computer Vision. (2015) 567–575

[5] Charpiat, G., Hofmann, M., Schölkopf, B.: Automatic image colourization via multimodal predictions. In: Computer Vision–ECCV 2008. Springer (2008) 126–139

[6] Places.csail.mit.edu. (2015). *MIT Places Database for Scene Recognition*. [online] Available at: http://places.csail.mit.edu/index.html [Accessed 29 Jun. 2019].

[7] Hwang, J. and Zhou, Y. (2016). *Image colourization with Deep Convolutional Neural Networks*. [online] Cs231n.stanford.edu. Available at: http://cs231n.stanford.edu/reports/2016/pdfs/219_Report.pdf [Accessed 29 Jun. 2019].

[8] Wiki.ubc.ca. (2019). *Image Colourization using Deep Learning - UBC Wiki*. [online] Available at: https://wiki.ubc.ca/Image_Colourization_using_Deep_Learning [Accessed 29 Jun. 2019].

[9] Mouw, T. (2018). *LAB Color Space and Values | X-Rite Color Blog*. [online] X-Rite. Available at: https://www.xrite.com/blog/lab-color-space [Accessed 29 Jun. 2019].

[10] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition", *arXiv.org*, 2019. [Online]. Available: https://arxiv.org/abs/1512.03385. [Accessed: 23- Jul- 2019].

[11] 2019. [Online]. Available: https://www.researchgate.net/figure/Proposed-Modified-ResNet-18-architecture-for-Bangla-HCR-In-the-diagram-conv-stands-for_fig1_323063171. [Accessed: 23- Jul- 2019].

[12] L. Zhang, Cs.toronto.edu, 2019. [Online]. Available: http://www.cs.toronto.edu/~rgrosse/courses/csc421_2019/assignments/assignment2.pdf. [Accessed: 11- Aug- 2019].

[13] Ronneberger, Olaf, Philipp Fischer and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation.” ArXiv abs/1505.04597 (2015): n. Pag.

[14] J. Brownlee, "A Gentle Introduction to the Rectified Linear Unit (ReLU)", *Machine Learning Mastery*, 2019. [Online]. Available: https://machinelearningmastery.com/rectified-linear-activation-function-for-deep-learning-neural-networks/. [Accessed: 15- Aug- 2019].

[15] "Batch normalization in Neural Networks", *Medium*, 2019. [Online]. Available: https://towardsdatascience.com/batch-normalization-in-neural-networks-1ac91516821c. [Accessed: 15- Aug- 2019].

[16] "A Tutorial on Fairness in Machine Learning", *Medium*, 2019. [Online]. Available: https://towardsdatascience.com/a-tutorial-on-fairness-in-machine-learning-3ff8ba1040cb. [Accessed: 15- Aug- 2019].

[17] A. Trevino, "Introduction to K-means Clustering", *Datascience.com*, 2019. [Online]. Available: https://www.datascience.com/blog/k-means-clustering. [Accessed: 15- Aug- 2019].