Comparative Analysis of Image Colorization Models

Team 3

Matt Corrigan (undergraduate), Chan Hen (undergraduate), Pritam Kayal (undergraduate)

Abstract

The challenge of colorizing black and white images has spurred the development of various techniques, among which the application of Generative Adversarial Networks (GANs) and Linear Encoder-Decoder models stand out. In this research, we conduct a comparative analysis of a generalized GAN-based model and an encoder-decoder-based model for image colorization, exploring the strengths and weaknesses inherent in the two distinct approaches within the overarching domains of generative learning and supervised learning. Model 1, a Linear Encoder-Decoder, integrates a Convolutional Neural Network (CNN) with features from the pre-trained Inception ResNet-v2 model. Model 2, a conditional Deep Convolutional Generative Adversarial Network (DCGAN) with a U-Net generator, employs a modified GAN framework. Our evaluation spans three diverse datasets, incorporating varied hyperparameters and employing key metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), Peak Signal to Noise Ratio (PSNR), and accuracy for comprehensive comparison. Visual analysis, comparing predictions against original images and referencing a state-of-the-art model

B. Introduction

We performed a comparative analysis of a generalized GAN-based model to an encoder-decoder-based model for coloring black and white images. Based on our metrics, we determined which of these models is better at image colorization.

The challenge of colorization of black-and-white images has been a difficult task. The problem involves using a lower-dimensional representation of an image to reconstruct its original higher-dimensional colors. However, it has been done before with varying degrees of success. In our project, we followed the outlines and models from https://arxiv.org/pdf/1803.05400.pdf and https://arxiv.org/pdf/1712.03400.pdf. To elaborate, the first model will be a Linear Encoder-Decoder model: a model that combines a deep Convolutional Neural Network (CNN) trained from scratch with high-level features extracted from the Inception ResNet-v2 pre-trained model (Baldassarre et. al). The second model is a Generative Adversarial Network (GAN): a conditional Deep Convolutional Generative Adversarial Network (DCGAN) with a U-Net generator (Nazeri et. al).

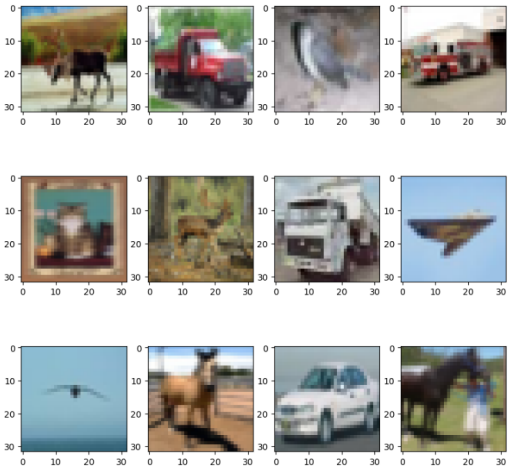
In the present day, numerous state-of-the-art models, such as ColTran, ToVivid, BigColor, InstColor, and others, have made remarkable advancements in image colorization. When we individually assessed our models in comparison to these advanced counterparts, it became evident that our models may lag in several respects. Nevertheless, our primary objective revolved around gaining a comprehensive insight into the core approaches to the challenge of image colorization, as well as recognizing the strengths and weaknesses of both methodologies.

C. Data

Model 1 Dataset: CIFAR-10
https://www.tensorflow.org/datasets/catalog/cifar10

The CIFAR-10 [4] dataset is a collection of 60,000 color images, each measuring 32x32 pixels, divided into 10 distinct classes, namely airplanes, cars, birds, cats, deer, dogs, frogs, horses, ships, and trucks. It is widely used for training and evaluating machine learning and computer vision models, particularly for image classification tasks. The dataset is split into 50,000 training images and 10,000 testing images, and it has become a standard benchmark for assessing the performance of various deep learning algorithms, including convolutional neural networks (CNNs). Due to its diversity and relatively low image resolution, CIFAR-10 poses challenges that reflect real-world scenarios, making it a valuable resource for researchers and practitioners in the field of computer vision.

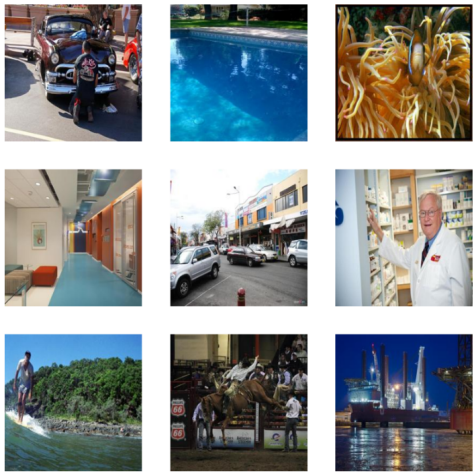
We used a subset of 3600 images from the CIFAR-10 dataset because the resources needed to train on a larger dataset would have exceeded the scope of this project. We partitioned these images into a 83/17 split with 3000 images for training and 600 for testing. Before feeding the images into our model we also performed some preprocessing, starting with upscaling the image size to 299x299 in order to fit the input shape of ResNet. Next, we normalized the pixel values of the images to center and scale them between -1 and 1. Lastly, we converted the images from RGB encoding to L*a*b* encoding which allowed us to use only one component (L) for the grayscale image encoder input.



Model 2 Dataset: Places365
http://places.csail.mit.edu/

The Places365 [4] dataset, also known as Places2, is a massive and diverse collection of over 1.8 million labeled images, encompassing 365 distinct scene categories that represent a wide array of indoor and outdoor environments, from "beaches" to "kitchens" and more. These images, in RGB color, are utilized for a variety of computer vision tasks, including scene recognition, object detection, and image segmentation. Places365 is highly valued in the computer vision community as it enables the development and evaluation of models capable of understanding and interpreting complex and diverse visual scenes. With its hierarchical organization of scene categories, it is particularly useful for fine-grained scene recognition, and its scale and diversity make it a valuable resource for assessing the robustness and generalization abilities of deep learning models in real-world scenarios.

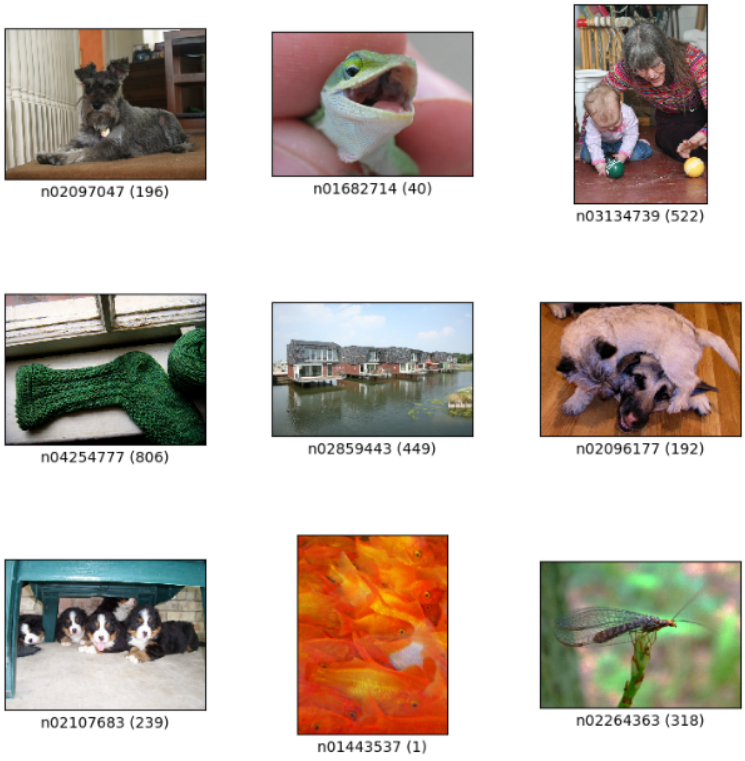
For this model we took a subset of 2500 images from the Place365 dataset. We partitioned them using an 80/20 split, 2000 for training and 500 for testing. The images were then rescaled to pixel dimensions of 256x256. We performed preprocessing similar to the CIFAR-10 dataset - pixel values were normalized and converted to L*a*b* encoding.



Testing Dataset: ImageNet
https://www.image-net.org/

The ImageNet [3] dataset was created primarily as a large-scale image dataset for use in computer vision research and the development of image recognition and classification algorithms. The dataset's images are organized according to the WordNet hierarchy, a lexical database that groups words and phrases into sets of synonyms (synsets) and describes their relationships. ImageNet contains a wide range of images covering a vast array of everyday objects, animals, scenes, and more, with the goal of providing a diverse and comprehensive set of visual data for machine learning tasks. During its creation and maintenance, ImageNet's developers aimed to provide, on average, 1000 high-quality images for each synset. These images are carefully curated, quality-controlled, and human-annotated to ensure that they accurately represent the corresponding concept within the WordNet hierarchy.

In order to effectively evaluate the two models, we needed to use the same test data to compare each model's loss and accuracy. For this reason we chose a subset of 1800 images from the ImageNet dataset - a dataset that neither model has been trained on before - to evaluate the model with. Each model performed the same preprocessing on the ImageNet data that it did on the training data. Note that the average image size of images in this dataset is 469x387, however all inputs were rescaled to 299x299 or 250x250 depending on the model.



D. Tasks Performed

Dataset Collection and Preprocessing

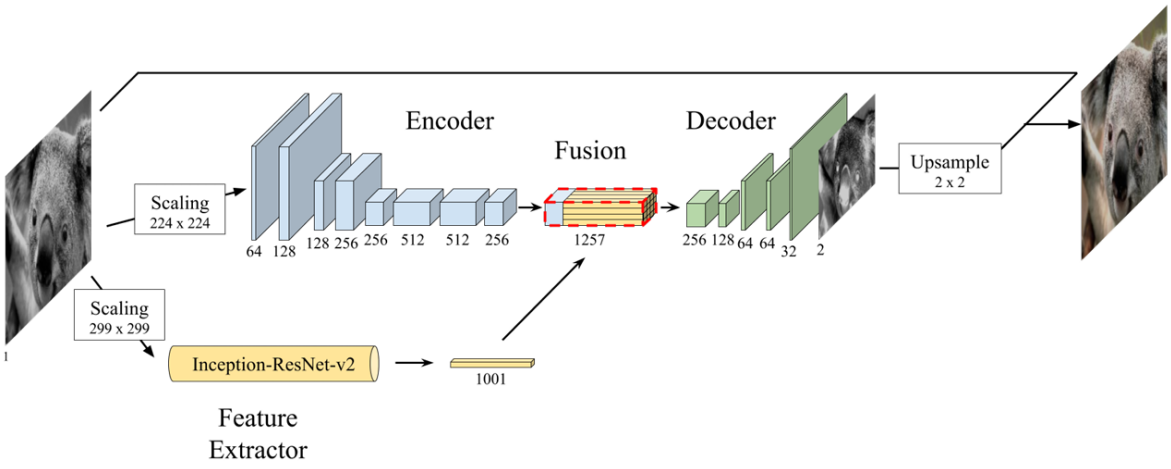
For training our two models, we obtain a random subset of images from the aforementioned databases and convert the images into the CIELAB color space. That is, each pixel has three coordinates L, a, and b. We use the luminescence (L) component of the images as the representation of the "black and white" format of the image. The goal of the two models is to predict the remaining two components a and b to reconstruct the colored image.

We use the CIELAB color space for two reasons:

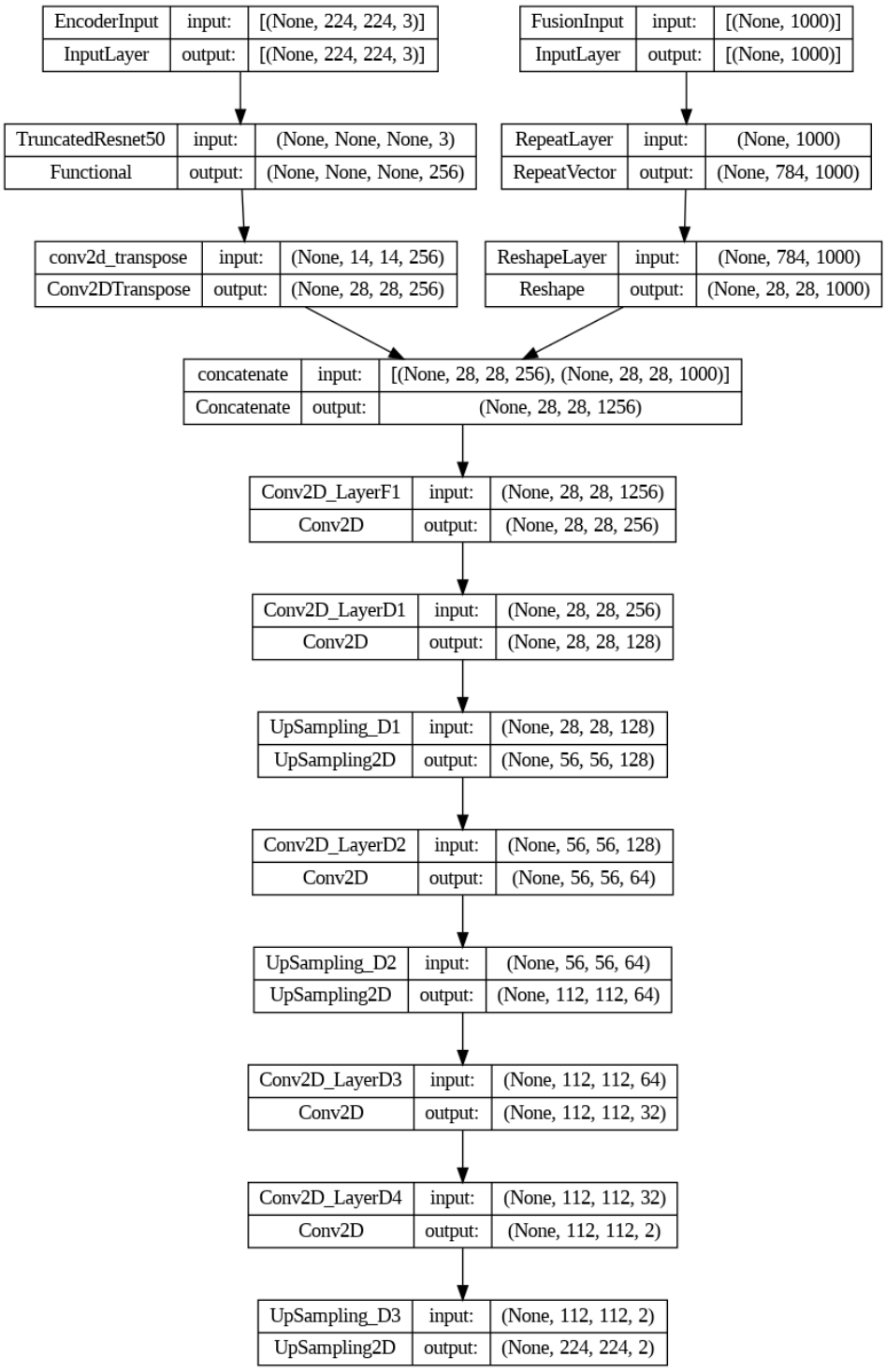
- CIELAB has a dedicated channel for representing image brightness, while the color information is entirely encoded within the remaining two channels. This configuration effectively mitigates abrupt changes in both color and brightness resulting from minor intensity value perturbations experienced in the RGB color space.
- The models have to just predict two components a and b instead of three; if the RGB configuration were used, all three components would have to be learned

Building and Training Model 1

Our first model was based on the "Koalarizer" model described in the paper by Baldassarre et al. This paper makes use of an encoder-decoder model along with an Inception-ResNet-v2 model for feature extraction. We modified the Koalarizer model to use a truncated ResNet50 instead for our encoder component (instead of building it from scratch) and then used transfer learning to fine tune the model. The original architecture is as follows:



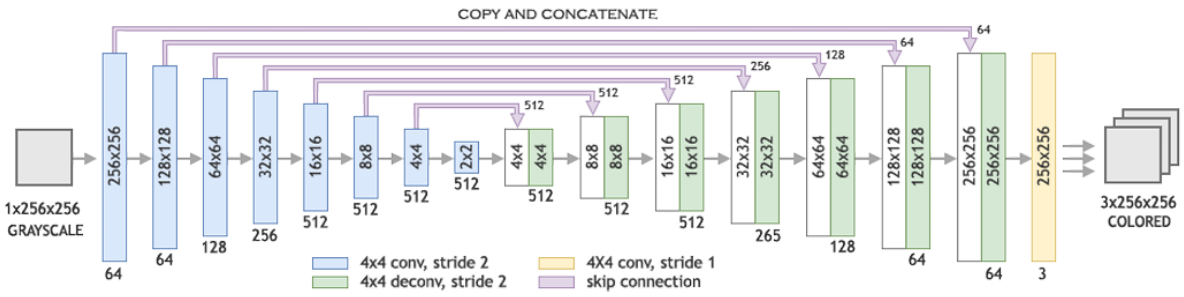
Our modified architecture is depicted below (with everything the same except the encoder as the truncated Resnet50).



Our model was trained for 400 epochs with a batch size of 64.

Building and Training Model 2

The second model was based on Nazeri et al's deep conditional generative adversarial network. The generator comprises a U-net architecture. The motivation for the U-net is that in a linear encoder "there is an information bottleneck that prevents flow of the low level information the network in the encoder-decoder architecture.To fix this problem, features from the contracting path are concatenated with the up sampled output in the expansive path within the network." The overview of the generator (in the original paper) is as follows:



We modified the architecture to have 2 filters at the end instead of 3. This is because we use the L component of an image as input to obtain the A and B components as output and simply pad these three dimensions to obtain the full image. This not only optimizes the memory but also makes training faster as the model just has to learn two dimensions of the full image. The discriminator has a similar architecture as the encoder part of the generator.

A traditional generator is trained to minimize the probability that the discriminator makes a correct prediction in generated data:

$$min_{\theta_G} J^{(G)}(\theta_D, \theta_G) = min_{\theta_G} E_z[\log(1 - D(G(z)))]$$

As recommended by Nazeri et al, we modified to maximize the probability of the discriminator being mistaken instead:

$$min_{\theta_G} - J^{(G)}(\theta_D, \theta_G) = min_{\theta_G} - E_z[log(D(G(z)))]$$

The cost function was further modified using l^1 regularization:

$$min_{\theta_G} J^{(G)^*}(\theta_D, \theta_G) = min_{\theta_G} - E_z[log(D(G(z)))] + \lambda \|G(z) - y\|_1$$

The model was trained for a total of 400 epochs with a batch size of 32 (for memory optimization).

Comparing Both Models With the ImageNet Dataset

After training each model with their respective datasets for similar amounts of time, we compared them with the ImageNet dataset. By using a dataset that neither of the models have seen before, we can evaluate their metrics in hopefully an unbiased manner. Each model was tested using the same 1839 images to be consistent. Each model was scored on four different metrics, listed in section E.

E. Results and Discussion

The two models have been evaluated on a subset (~2000 images) of Imagenet using the following four metrics:

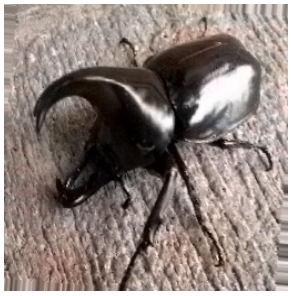
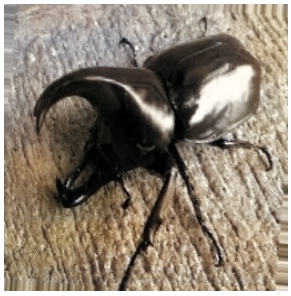
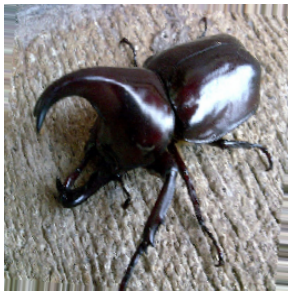
- Accuracy
- Mean Absolute Error (MAE)
- Mean Square Error (MSE)
- Peak Signal-to-Noise Ratio (PSNR)

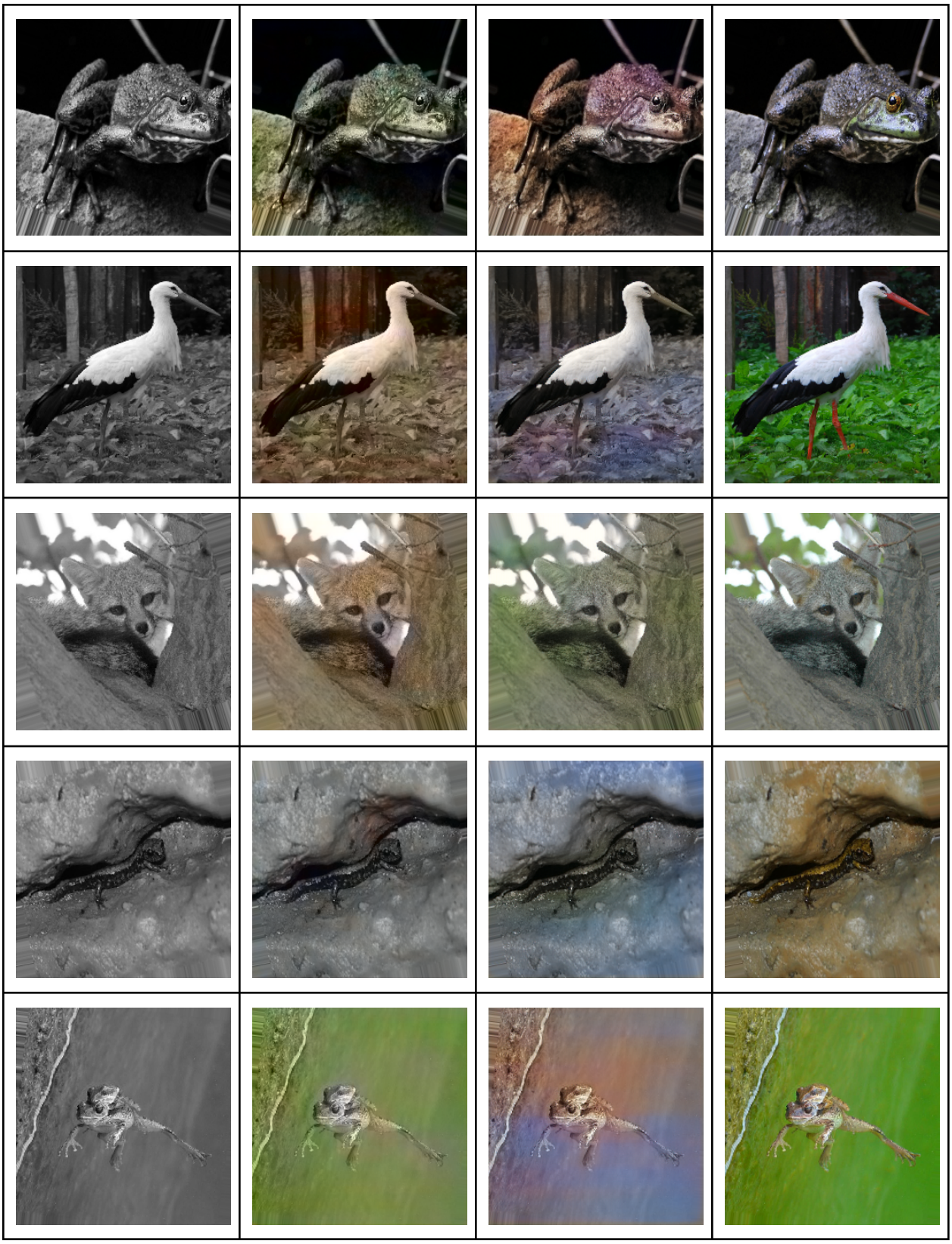
The results are as follows:

Model Number	Model Name	Accuracy	MAE	MSE	PSNR
1	Modified Koalarizer	**64.61**%	0.094655	0.018983	19.289463
2	DCGAN	64.15%	**0.084736**	**0.016054**	**19.91638**

In terms of accuracy, Model 1 outperforms Model 2 but only by a very slight margin. This is perhaps because Model 1 tends to color many images brown which averages the accuracy to a slightly higher margin. Model 2 gets punished for coloring some images with highly diverse colors (like coloring white surfaces blue, purple, and green). In terms of the other metrics, Model 2 outperforms Model 1 by a small margin.

The visual comparison of the performance of the two models is given below. The predictions are based on a random subset of Imagenet from different classes.

Black and White	Koalarizer (Modified)	DCGAN	Actual Image
			



Both models perform well when evaluated with nature images but fail to predict accurate results when evaluated with human images; this is perhaps because the CIFAR-10 dataset and the Places365 dataset both have very few human images. This problem will vanish on training the models on human images. In the table below, we see that both models are able to color the water surrounding the face very well, but leave the image of the face gray with no change. BigColor (state-of-the-art model) does not face the aforementioned issue, however.

Black and White	Koalarizer (Modified)	DCGAN
		

BigColor	Actual Image
	

Issues unique to the modified Koalarizer:

For images which the model could not learn properly (like cityscapes, water, etc), the model tends to color them by either leaving it uncolored (gray) (issue 1) or by coloring it with the most frequently occurring color in the training dataset (green) (issue 2).

Issue 1 (Uncolored image)	Issue 2 (Colored with most common color)
	

Issues unique to the DCGAN:

Some predicted images experience th[1]e "sepia effect." This hue appears mainly in images with clear skies. There is a strange color gradient between light blue and yellow (issue 1). Further, some images have small circular spots of bright colors (pink, green, etc.) which we suspect is caused due to insufficient training (issue 2).

Issue 1 (Sepia effect)	Issue 2 (Small bright circular spots)
	

[1] https://github.com/PritamLemon/cs571Project

F. References

Federico Baldassarre, Diego González Morín, and Lucas Rodés-Guirao. "Deep Koalarization: Image Colorization using CNNs and Inception-ResNet-v2." (2017).

Kamyar Nazeri, & Eric Ng. Image Colorization with Generative Adversarial Networks. CoRR, abs/1803.05400. (2018).

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In 2009 IEEE conference on computer vision and pattern recognition (pp. 248–255). (2009).

Krizhevsky, A., Nair, V. and Hinton, G. The CIFAR-10 Dataset. https://www.cs.toronto.edu/~kriz/cifar.html. (2014).

Places365-Standard. (n.d.). MIT Scene Parsing Benchmark. Retrieved from http://places2.csail.mit.edu

Geonung Kim, Kyoungkook Kang, Seongtae Kim, Hwayoon Lee, Sehoon Kim, Jonghyun Kim, Seung-Hwan Baek, and Sunghyun Cho. "BigColor: Colorization using a Generative Color Prior for Natural Images." (2022).