Attaining Context Awareness in Image Colourisation: An Ensemble of Pix2Pix Models

G054 (s2450044, s2446971, s1925182)

Abstract

This project researches the task of natural image colourisation using Conditional Adversarial Networks (CGANs) in the Colour Spaces of YUV and LAB. We propose a two-stage model for image colourisation which adds contextual information via an initial classification stage followed by a colourisation tailored to the class output (for the queried image) stage. A quantitative and qualitative evaluation of our model is done to compare it with respect to the baseline model: the Pix2Pix CGAN. Due to a combination of factors we were unable to decisively conclude an improvement of our results however believe further research could advance the task of image colourisation.

1. Introduction

Image colourisation is an essential task in computer vision that involves generating a coloured image from a grayscale input image. The range of applications is wide and includes (but is not limited to): medical image colourisation (Liang et al., 2022) where many images are produced in grayscale and colourisation can provide additional information for diagnosis and interpretation of these images, manga and cartoon colourisation (Hensman & Aizawa, 2017) can aid in the time-consuming task of colourising many images or frames by hand, while architectural line drawing colourization (Sun et al., 2022) aids in clarifying and better expressing the full vision of the architect. Our research specialises in the task of natural image colourisation, which has uses including the restoration of old images for historical purposes as well as the research having intrinsic value on it's own. In contrast to some of the other applications, natural image colourisation poses the added tasks of handling complex textures and less defined borders than may otherwise be present in the applications listed above.

Previous research has touched on adding context to image colourisation through some combination of image segmentation and Convolutional Neural Networks. However, our study of previous work identifies and addresses a literature gap (as far as we know) by implementing a novel approach consisting of incorporating context via image classification followed by an application of CGANs. We contemplate the question of how the performance of our proposed ensemble model which takes into account context for image colourisation, consisting of CGANs, a UNet architecture, and a ResNet model, compares to Pix2Pix, a state-of-the-art method in fully automatic natural image colourisation (Kalvankar et al., 2022).

Given our baseline model is the state of the art image colourisation model (Pix2Pix) we contemplate whether we can improve the performance and quality (measured by both quantitative and evaluative metrics) of the colourised images outputted by Pix2Pix through the addition of context. This is done by adding an initial classification stage using a Resnet34 model to a posterior implementation of Pix2Pix consisting of an ensemble of models trained on a per category basis.

Although research in this field has been growing in recent years thanks to advancements in our understanding of colour theory as well as available colourising methods, much work remains as current methods often produce suboptimal results. Colourised images using these methods often suffer from some combination of unsaturation, colour leakage or lack of colour diversity, as well as colour incorrectness. As such image colourisation remains a challenging and insightful area worth conducting further research on.

2. Task and Data Set

2.1. Image Colourisation

In the task of Image Colourisation, the main goal is that of estimating the colour image which corresponds to the input grayscale image.

More formally, Image Colourisation can be depicted as follows. Let us respectively denote by $I_c \in \mathbb{R}^{d\times m\times n}$ and $I_g \in \mathbb{R}^{1\times m\times n}$ the corresponding colour and grayscale images to an arbitrary captured scene. The task of Image Colourisation corresponds to that of finding a function: $f : \mathbb{R}^{1\times m\times n} \longrightarrow \mathbb{R}^{d\times m\times n}$, such that $f(I_g)$ is equal to I_c for each pair of grayscale-colour images (I_g, I_c). In practice, the goal is to learn a map f from data such that, for each pair (I_g, I_c), the output $f(I_g)$ is as close to I_c as possible according to some metric on $\mathbb{R}^{d\times m\times n}$. Another widespread, and more subject to qualitative evaluation, metric to evaluate the map f is whether for any $I_g \in \mathbb{R}^{1\times m\times n}$, $f(I_g)$ is a realistic colourisation.

Image Colourisation is an example of an ill-posed task since the real correspondence between colour images and grayscale images cannot be fully captured by a function. Functions are defined over two sets, X, the domain, and Y

the codomain such that for all $x \in X$ there is a **unique** $y \in Y$ such that $f(x) = y$. In the task of image colourisation, this is not the case. A grayscale image can actually correspond to many colour images, for example, a picture of the same shirt in grayscale may correspond to various different colours in reality.

Due to the nature of the task, Image Colourisation research has no direct consensus on a single 'good performance' metric, instead, work in the field uses a combination of evaluation metrics both in the qualitative and quantitative sense as noted in the review by (Huang et al., 2022). As such for our final evaluation metrics we will parallel related works in the research area. We will use the GAN LOSS as well as the Structural Similarity Metric for our quantitative metrics and visual tests done on participants for our qualitative metrics, all metrics will be further explained in detail in the Section. 4.

We note here this also poses a challenge in defining a single absolute state of the art method, as different methods outperform in different metrics. Nonetheless, in our research we found consistent mentions of Pix2Pix, a method that utilises Conditional GANs, to be a state of the art architecture for image-image translation problems like colourisation (Kalvankar et al., 2022). As such, in this report, we used Pix2Pix as our baseline model to improve upon and compare against.

2.2. Colour Spaces and Image Colourisation

Colour Images are rendered using a wide variety of different models or Colour Spaces. These can be classified, according to Huang et al. (2022), into two main types: primary colour spaces and colour-bright separable colour spaces. Primary Colour Spaces are exemplified by the RGB colour space and are based on representing each pixel's colour by the combination of chroma channels (for the RGB colour space, this is a linear combination of the Red, Green and Blue channels).

On the other hand, colour-bright separable spaces are based on separating brightness and chroma channels, so that any colour in the image is attained by the linear combination of the colour channels while the intensity of the colour is treated separately by it's own brightness channel.

The **LAB Colour Space** was designed so that "the distances between colours in this space correspond to the perceptual distances of colours for a human observer" (Ballester et al., 2022). Here, the L channel dictates the brightness, while a and b channels dictate the colours. The a channel has green and red as opposite colours along its range, while the b channel has blue and yellow at the extremes of its range.

In the **YUV Colour Space**, Y represents the luminance channel, while U and B represent the brightness values of red and blue respectively. Unlike for the LAB colour space, the transformation from the YUV to the RGB colour spaces and vice versa is given by a linear transformation. This is important because it means that, regarding the transformation to RGB and its inverse, YUV is preferable over LAB since errors due to this transformation are linear (Wu et al., 2021).

Colour-bright colour spaces can decrease the complexity of the task of Image Colourisation using Deep Learning. When using colour spaces such as LAB or YUV (separable colour spaces), the luminance or brightness value of the predicted colour image is directly taken as the input grayscale image. In these colour spaces the regression task only requires estimating two values per pixel (AB / UV) which are related to the colour information, as opposed to the estimation of three-dimensional vectors per each pixel for the RGB space. It is for these reasons that in this project we focus only on the LAB and YUV colour spaces.

2.3. The Mirflickr Data set

For our purposes, we choose to work with the MIRFLICKR-25000 data set. This is an open evaluation data set with 25000 images from Flickr, a social photography site (LIACS Medialab, 2008). Its relevance for natural image colourization is rooted in its variety of images. Furthermore, for our purpose of adding contextual information about the image to produce better colourization, we benefit from this dataset's incorporated annotations.

Deleting existent grayscale images was the first preprocessing step required, which resulted in 21599 images. Following this, we split the remaining images into train, validation and test subsets with the proportions 0.75 : 0.15 : 0.1 respectively. In order to train our proposed ensemble of deep learning models to attain context awareness in colourisation, we split further our data set into the categories which are more common according to the description of the data set on the official website. Since image colourisation benefits greatly from large amounts of training images we limited our proposed categories to 6, providing enough training images for our model accurately learn colourisation. This resulted in the following amount of training images for each category - Animals: 1'112, City: 2'210, Nature: 798, People: 3'915, Scenery: 3'182 and Others: 4'984 (which contains the remaining types of images in our varied dataset).

3. Methodology

We start by presenting Generative Adversarial Networks (GANs) and what advantages they provide to the field of Artificial Intelligence. Afterwards, we will introduce the conditional GAN of Pix2pix and present its specifics as well as its benefits for image-to-image translation tasks such as image colourisation. At the end of this section, we will introduce our proposed ensemble to include contextual information about images before colourising them.

3.1. Generative Adversarial Networks

The goal when (Goodfellow et al., 2014) introduced GANs, was to benefit the competition inherent in adversarial training to improve both of the parties involved in it until they

reach equilibrium (specifically, a Nash Equilibrium).

To understand GANs we begin by defining a data space $\mathbb{O}$, with some related distribution along it p. The aim of adversarial training is to typically generate data instances in $\mathbb{O}$ that appear to have been sampled from p but have actually been artificially generated. The generator, G, is the party in charge of artificially generating such data instances and, in its original setting, (Goodfellow et al., 2014), it only receives an input of random noise, $z \sim p_z$, to include randomness in G, a typically deterministic function.

The counterpart of the generator in the adversarial training is the discriminator, $D : \mathbb{O} \longrightarrow [0, 1]$, whose objective is to excel at distinguishing whether a data instance has been generated by the original distribution on the data, p, or not. As such, for any data point, x, its output, $D(x)$, is viewed as the probability that x comes from p. Generally, both D and G are neural networks with parameters α and β respectively.

The novelty of GANs is that the loss function arises from a two-player zero-sum strategic game with D and G as its players and α and β as their respective strategies. As introduced in (Goodfellow et al., 2014), in this game, the payoff of D is defined by:

$$\mathbb{U}(\alpha, \beta) := \mathbb{E}_{x\sim p}[\log(D(x))] + \mathbb{E}_{z\sim p_z}[1 - \log(D(G(z)))].$$

The goal of both parties is to maximise their payoffs. Consequently, during the training process, the parameters α of D are updated to maximise $\mathbb{U}(\alpha, \beta)$ whereas those of G are updated to minimise it. This is done by alternating the updates of the generator and the discriminator generally.

3.2. Pix2Pix: The Baseline Model

Image-to-image translation problems such as image colourisation have been repeatedly labelled as per-pixel regression or classification tasks. Nevertheless, these approaches fault in their underlying assumption that pixels in output images are conditionally independent given the corresponding input image (Isola et al., 2017). Such an assumption is generally erroneous. For instance, in image colourisation, the output values of two adjacent pixels in the forehead of a person, are not conditionally independent given an input grayscale image, as knowing the colour of one will likely give us information about the colour of the other. Due to this, in (Isola et al., 2017) a conditional GAN, Pix2Pix, is introduced for image-to-image translation.

The difference in unconditional GANs is that the input of the generator is not only random noise, $z \sim p_z$, but also a grayscale image, $g \sim p_g$, which we want to colourise. Furthermore, the discriminator in the conditional setting also has access to the input grayscale image, in order to output for a query colour image, x, the probability of coming from the original data distribution: $D(g, x)$. Under this setting, the loss function is equal to the:

$$\mathbb{E}_{x\sim p,\, g\sim p_g}[\log(D(g, x))] + \mathbb{E}_{z\sim p_z,\, g\sim p_g}[1 - \log(D(g, G(g, z)))].$$

Furthermore, another term is added by (Isola et al., 2017) to force our generated image to not only fool the discriminator but also be similar to the real colour image corresponding to g. This term consists of the expected distance using the L1 metric from $G(g, z)$ to x, where x is the true colour image of the grayscale image g. The term is added to the above loss scaled by $\lambda = 100$ as in (Isola et al., 2017). The use of the L1 metric over the L2 metric is justified by (Isola et al., 2017) with the argument that it produces fewer blurred images. Finally, the authors input random noise to the generator in the form of dropout with 0.5 as the rate.

The GAN of Pix2Pix uses a U-Net architecture for the Generator, consisting of a convolutional block which concatenates a convolutional layer with batch normalization and a LeakyReLU (when downsampling) or ReLU (for upsampling) activation function. The U-Net architecture is similar to a symmetric (around the bottleneck) encoder-decoder architecture, but it uses skip connections from all layers in the encoder to the corresponding layers in the decoder at the same distance from the bottleneck. As the bottleneck may hinder the flow of information from the encoder to the decoder, skip connections are used to provide additional information in the upsampling process.

Figure 1, aids in the visualization of the U-Net structure. In our implementation of Pix2Pix based on code by (Shariatnia), the U-Net has 16 convolutional blocks in total with a kernel size of 4, the stride of 2 and amount of padding of 1.

The discriminator, referred to as PatchGAN, evaluates local patches of an image as opposed to focusing on the whole: this leads to better use of high-frequency details to improve the quality of the image and reduce the likelihood of colour overspilling. It consists of downsampling blocks concatenating convolutional layers (with 1 as padding and 2 as stride) with batch normalization and a posterior Leaky ReLU activation function. The last block uses stride 1, omits both batch normalization and substitutes the Leaky ReLU activation function by the Sigmoid activation function. It is built so that the output of the discriminator divides input images into 70×70 patches and then produces for each an output "probability" of how likely it is that each patch is real (the average of them is usually taken for training and evaluation).

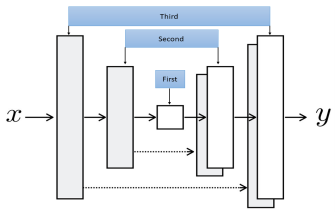

Figure 1. Reduced structure of the U-Net architecture (Shariatnia).

3.3. A Context-Aware Ensemble: Our Proposed Model

In this project we propose a two-stage approach to image colourisation which consists on an initial classification stage performed by a ResNet34 model followed by a pos-

terior colourisation done by a Pix2Pix model trained for colourising images belonging to the same category as the one output by the ResNet model. This proposal is rooted on the observation that the diversity of possible input images is unbounded for the task of natural image colourisation. Therefore, feature extraction in this task is potentially more complicated than for others since inputs are not homogeneous or necessarily restricted to a set of categories. In that sense we believe that by clustering images into categories with more homogeneous features before the colourisation will help in the task by improving the feature extraction capacities of the models.

Firstly, we train a Pix2Pix model for each of the distinct categories in our dataset: city (c), nature (n), animals (a), scenery (s), people (p), and others (o). Let us call the generators of these: G_c, G_n, G_a, G_s, G_p and G_o respectively. Afterwards, we train a ResNet34 model for the classification of images into the aforementioned categories available in our dataset. The trained ResNet model gives us the map:

$$Res : \mathbb{R}^{1\times m\times n} \longrightarrow \{c, n, a, s, p, o\}.$$

Our proposal as explained above then uses the models from the last paragraph as building blocks to build the function:

$$f : \mathbb{R}^{1\times m\times n} \longrightarrow \mathbb{R}^{3\times m\times n};\ f(I_g) := G_{Res(I_g)}(I_g).$$

4. Experiments

4.1. The Baseline Model Experiments.

Initially, for our baseline, we decided to follow the guidelines for training Pix2Pix in the task of image-to-image translation given in (Isola et al., 2017). Starting by training our GAN for 200 epochs in which both the discriminator and generator losses are optimized with the Adam solver by minibatch stochastic gradient descent with learning rate 0.0002 and momentum constants $\beta_1 = 0.5$ and $\beta_2 = 0.999$.

Upon inspection we perceived our Pix2Pix model with LAB colour space might be overfitting the training data, as indicated by the low values in the generator's loss function components at the 200th epoch. The L1 loss gave a value of 5.39, the original loss (without the L1 loss term) a value of 3.44, with the combined total loss (which accounts for the fooling of the discriminator and the pixel-wise similarity) at 8.83. This overfitting was confirmed when examining the colourisations of our training and validation images (Figures 6, 7 and 8 in the Appendix). As similar observations were made using the YUV colour space, we decided to implement generalisation control methods.

To prevent overfitting in our generator, we explored implementing plain Early Stopping for our baselines, such that the training process would be stopped provided the validation loss increased with respect to the previous epoch. Even when implementing a stricter version of early stopping which compared the mean validation loss in the last five epochs with respect to the previous five epochs, our model continued stopping prematurely. Quite possibly this was due to the noisy and abrupt validation and training curves caused by the adversarial training of the generator and discriminator.

As a solution, we decided to implement a version of Early Stopping which accounts not only for the generalization loss but also for the training progress (i.e. the relative convergence of the training loss measurements). As defined in (Montavon et al., 2012), for each epoch e this version first computes the generalization loss:

$$GL(e) := 100\left(\frac{E_{va}(e)}{E_{opt}(e)} - 1\right)$$

where $E_{va}(e)$ the validation loss (in our case the total generator loss) in epoch e, while:

$$E_{opt}(e) := \min\{E_{va}(t) \ :\ t \leq e\}$$

so that $GL(e)$ shows for an $E_{va}(e)$ larger than $E_{opt}(e)$, by how much percentage is the current validation error over the minimum validation error. Afterwards, a measure of how stable is our training error during the last k epochs is also computed by:

$$P_k(t) := 1000\left(\frac{\sum_{e'=e-k+1}^{e} E_{tr}(e')}{k\ \min\{E_{tr}(e') \ :\ e-k+1 \leq e' \leq e\}}\right)$$

as defined in (Montavon et al., 2012). This measures how much bigger is the average training error (from the generator loss in our case) in comparison with the minimum training error during the last k epochs. This is a measure of the incipient convergence of our training progress. For instance, it can be seen that when the average is close to the minimum (always the case when a convergence of the training curve is noticeable), the value of P_k is small indicating the beginning of the phenomenon of convergence.

This version of Early Stopping, (Montavon et al., 2012), then introduces a lower bound μ in:

$$\frac{GL(e)}{P_k(e)} > \mu$$

to give us our stopping criterion which, therefore, considers the validation error and the progress of our training error curve (in both cases, of our generator). We used this version of Early stopping with the following parameters $(k, \mu) = (10, 0.75)$ to train our baseline model once more.

We can now see the training and validation errors that the generators of our baseline model produced at the end of their training for the LAB and YUV colour Space in table 1. The plain loss of the generators (GAN Plain Loss) in the validation data shows that in the YUV colour space, the generator more frequently fools the discriminator in comparison with the ones in the LAB Colour Space. Nevertheless, the performance with respect to the L1 metric was shown to be better for the LAB space meaning that generated images were on average closer pixel-wise to the original images with respect to the chroma channels (the L1 Loss is only calculated for the channels with colour information).

Pix2Pix	LAB	YUV
Train		
L1 Loss	5.923	5.933
GAN Plain Loss	2.560	2.476
Validation		
L1 Loss	11.525	11.686
GAN Plain Loss	1.739	1.400
Last Epoch	119	109

Table 1. Last epoch's training and validation loss values of the generator from Pix2pix which was trained with Early Stopping. The term GAN Plain Loss refers to the loss value of the general GAN loss function without the L1 loss term, while GAN Loss accounts for this term.

4.2. Experiments with our Proposal

Since we propose a two-stage image colourisation method consisting of a classification stage followed by a category-tailored colourisation, we need to undergo the training of the models in both stages.

As a starting point, we begin by training a ResNet34 model for grayscale image classification into our chosen categories. Due to the presence of class imbalances in our dataset we base our model selection criteria on selecting the best Macro F1-Score obtained in the validation set during the training process.

For training the ResNet34 model, we used minibatch Stochastic Gradient Descent with a learning rate 0.00075 and momentum 0.9 during 100 epochs. The batches were of size 16 and our loss function was given by the cross entropy loss function (although the training F1-Score was also recorded). The selected model to be used in the classification stage of our image colourization approach attained a validation F1-score of 0.893.

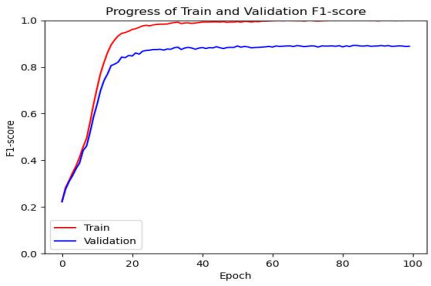

Figure 2. Training and Validation F1-score of ResNet34 during its training process. We see that effectively, our strategy of choosing the model with the best validation score is equivalent to that of Early Stopping.

Once the ResNet34 model was trained, we proceeded to the training of a per-category Pix2Pix Model for colourisation. This was done for both the LAB and YUV colour spaces in order to find the best colourisation method.

For each of the models trained, the discriminator and generator losses were optimized with the Adam solver by minibatch stochastic gradient descent with a learning rate of 0.0002 and momentum constants $\beta_1 = 0.5$ and $\beta_2 = 0.999$. The batch size was chosen to be 16 since it maximised the stability of the training updates given the memory constraints of our used GPU. Furthermore, the hyperparameters (k, μ) used for controlling the implemented Early Stopping were set, after some initial experiments, to:

- $(10, 0.9)$ for the Scenery and People category.
- $(10, 1.1)$ for the City category.
- $(20, 1.15)$ for the rest of the categories.

The results of training each category's colourisation model in the LAB and YUV colour Spaces can be seen in tables 2 and 3 respectively. The results show a better performance of the scenery, people and city models in the LAB colour space, while the nature category shows a better result for YUV. In the case of the categories of animals and others, while the discriminator is more frequently fooled in the LAB colour spaces, the average pixel-wise distance to the real chroma channels from the produced ones is smaller for the YUV colour space.

As a matter of fact, we can see by the results obtained in the validation set that in terms of fooling the adversary discriminators, our proposed model is on average better than the baseline model for the LAB colour space (see Table 6 in the Appendix).

We believe that the hyperparameter tuning done for the k and μ parameters may be one of the causes negatively impacting our models (this can be seen in how it affected the results for the nature, animal and city categories in tables 2 and 3). Initially it was intended to avoid the premature stopping occurring in the training of the Pix2Pix models in several categories such as the nature category. We now believe the premature stoppings observed may relate to our initial choice of using the total GAN Loss (which adds the L1 loss and the GAN Plain Loss) instead of using the plain GAN loss (without accounting the L1 Loss). The reason is that training curves become more unstable if we consider the L1 loss; the pixel-wise distance in the colour channels between images can differ considerably between several produced images and others by chance, and this can cause premature stopping of our model. We believe future research could look instead into the implementation of a version of Early Stopping using the metric of GAN Plain Loss, or explore other methods for preventing overfitting.

Before entering the quantitative and qualitative evaluations of our models, we briefly present some of the output images and briefly discuss notable observations.

Visually we note some discrepancies between our LAB and YUV models and their respective baseline models (Fig. 3 shows the output images for YUV colour space while the appendix contains LAB's). Firstly both our LAB and YUV models tend to output images that are more saturated, as

LAB	Scenery	People	Nature	City	Animal	Others
Train						
L1 Loss	4.844	4.432	7.049	5.165	11.270	5.910
GAN Plain Loss	1.635	1.521	2.354	1.885	1.357	1.682
Validation						
L1 Loss	11.363	**9.644**	16.927	9.881	11.824	13.879
GAN Plain Loss	1.420	**1.076**	1.492	1.080	1.248	1.103

Table 2. Shows for the model trained for each category, the last epoch's training and validation loss values of the generator from Pix2pix which was trained with Early Stopping. The training is done in the LAB Colour Space.

YUV	Scenery	People	Nature	City	Animal	Others
Train						
L1 Loss	4.326	4.405	12.205	6.740	4.818	5.425
GAN Plain Loss	1.397	1.606	1.262	1.610	1.862	1.588
Validation						
L1 Loss	11.718	**9.747**	15.528	10.664	10.315	12.352
GAN Plain Loss	1.520	2.785	0.985	1.562	1.507	**1.199**

Table 3. Shows for the model trained for each category, the last epoch's training and validation loss values of the generator from Pix2pix which was trained with Early Stopping. The Training is done in the YUV Colour Space.

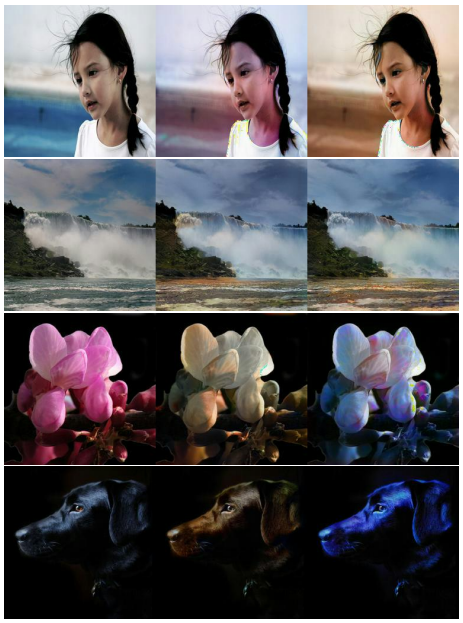

Figure 3. (Left) Ground Truth Images (Center) YUV Baseline Model with Pix2Pix (Right) ResNet34 + Pix2Pix Ensemble in YUV Colour Space

can be seen by example images 1 and 3 in Figure 3, this can be attributed to the baseline relying on a single model for all images and trying to minimise the risk of generating unrealistic colourisations thus opting for more conservative, unsaturated colours. Meanwhile, as each individual model in our ensemble is more accustomed to certain colours (for example, our nature models to bright greens or blues) it is less conservative and thus outputs more saturated colours. While this is beneficial for saturation, it also leads to our models struggling more with colour correctness, indeed we observe that the baseline model (especially in the LAB colour space) is more prone to hallucinating colours. Below we show a limited collection of other output images from our models with some more added in the appendix.

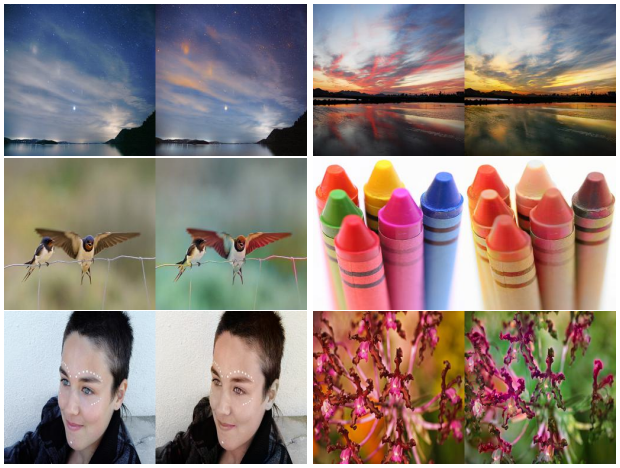

Figure 4. A collection of other images classified by our trained ResNet model in their respective categories and colourised using our colourisation model (either in LAB or YUV) (R) paired with their corresponding ground truth images (L)

4.2.1. Quantitative Evaluation

In order to quantitatively evaluate our proposed ensemble, we will use to metrics. The first metric measures the mean L1 distance between the colour channels of a generated image and its corresponding true colour image. It is a strict measure since it measures dissimilarity pixel-wise requiring an exact colour pixel-wise in order to output a null distance.

In addition, we chose to use the Structural Similarity Met-

ric (SSIM), to measure how generated images differ from the original colour ones with respect to their luminance, contrast and structure. The SSIM for grayscale images is defined by (Wang et al., 2004) as:

$$SSIM(I_1, I_2) := [l(I_1, I_2)]^{\alpha}[c(I_1, I_2)]^{\beta}[s(I_1, I_2)]^{\lambda}$$

where the quantities $l(I_1, I_2)$, $c(I_1, I_2)$ and $s(I_1, I_2)$ measure the discrepancy between images I_1 and I_2 with respect to their luminance, contrast and structure. The range of these functions is $[0, 1]$ and the constants α, β and λ reflect the importance that we attribute to each of these discrepancy measures. In our case, as generally done in image colourization tasks (Huang et al., 2022), $\alpha, \beta, \lambda = 1$. Now we proceed to explain each of the components of the equation above.

The luminance comparison is given by $l(I_1, I_2)$. To compute $l(I_1, I_2)$ we first need the mean of the pixel values μ_h for each $h \in \{1, 2\}$ as an estimate of the luminance of the grayscale image I_h. Then:

$$l(I_1, I_2) := \frac{2\mu_1\mu_2 + c_1}{\mu_1^2\mu_2^2 + c_1}$$

where c_1 is a small constant ensuring numerical stability. As we can see, the closer our estimates of the mean intensities for each image are, the closer is our value to 1. Furthermore, it is clear that $[0, 1]$ is the range of $l(I_1, I_2)$.

The contrast of the input grayscale images, $c(I_1, I_2)$, is calculated by using the standard, σ_h, of each image as an unbiased estimate of the contrast of the corresponding image, I_h. Then:

$$c(I_1, I_2) := \frac{2\sigma_1\sigma_2 + c_2}{\sigma_1^2\sigma_2^2 + c_2}$$

where c_2 is a constant used for numerical stability. The interpretation of this equation is similar to the one given for $l(I_1, I_2)$.

To calculate the structural agreement between I_1 and I_2, (Wang et al., 2004) decided to use the correlation between the both as an estimate:

$$s(I_1, I_2) := \frac{\sigma_{12} + c_3}{\sigma_1\sigma_2 + c_3}$$

where σ_{12} is the covariance between the images and c_3 is another constant to gain numerical stability. Clearly, a correlation of 1 indicates an identical structural similarity while 0 indicates that the images are completely linearly uncorrelated.

Therefore, for grayscale images $SSIM(I_1, I_2)$ close to 0 indicates an absence of resemblance between I_1 and I_2 in either luminance, contrast or structure. On the other hand, $SSIM(I_1, I_2)$ close to 1 indicates the opposite. In the case of colour images, the Structural Similarity Metric is calculated by averaging the $SSIM$ values obtained for each channel.

Table 4 shows that the YUV model performs better colourisations in general with respect to the pixel-wise fidelity and the similarity of luminance, contrast and structure.

Baseline	LAB	YUV
L1 Distance	**12.606**	12.687
SSIM	**0.849**	0.835
Ensemble	**LAB**	**YUV**
L1 Distance	13.505	**12.619**
SSIM	0.832	**0.838**

Table 4. Quantitave Results measuring the L1 distance between the generated image chroma channels and the original ones, and the Structural Similarity Index Measure.

4.2.2. Qualitative Evaluation

For the qualitative evaluation of our model, we follow similar works to previous research (Zhang et al., 2016) in measuring the ability of our output images to 'fool' participants. For this experiment, we asked 104 participants to label which of 24 images presented to them were artificially colourised and which were real (out of 12 Fake and 12 Real images). Below are our results in a contingency table for the YUV colour space (other results follow in the appendix).

		Participant's Judgement		
		Real	AI	Total
Actual Image	Real	715	533	1248
	AI	596	652	1248
	Total	1311	1185	2496

Table 5. Results from our Qualitative Evaluation Survey - YUV

Our model would be performing optimally if the accuracy of participants would be that of 50%, essentially this would mean participants had no way of discerning between if an image was colourised using our method or was natural and was thus forced to guess (50% chance) the labelling of the image. While a 'fooling rate' (FR) of 45.23% may seem high, we perform a one-sample chi-square test to see if the proportion of times participants were fooled by our artificially colourised images is significantly different from chance (50%). The formula used for the Chi-Square test is below:

$$\chi^2 = \Sigma_{df}^2 \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}}$$

Where df denotes our degrees of freedom (in this case 1), O denotes our observed values, E the Expected Values for each cell if labeling by participants was truly by chance, and i, j indexes the entry in the contingency table. We compute our χ^2 statistic of 19.4022 which gives us a p-value $p = 0.000011 p < .00001$ implying the result is significant at $p < .05$. This suggests there is a significant association between the actual image types and the participants' judgements and leads us to statistically conclude our images were

unable to reliably fool participants. Similar results follow for our LAB implementation (χ^2=26.147, $p < .00001$, FR = 44.91%), our baseline model with LAB (χ^2 = 22.7519, $p < .00001$, FR = 45.39%) and our baseline model with YUV (χ^2 = 19.5781, $p < .00001$, FR = 45.59%).

4.3. Interpretation of Results

The above quantitative and qualitative results are decisively similar between models, due to this and the absence of a universally accepted evaluation metric for image colourisation, we cannot conclusively state whether our proposed model which takes into account added context through classification improves on the performance or quality of our baseline model.

The main quantitative comparisons in this project are based on: the proximity of generated images to their real counterparts measured by the average pixel-wise L1 distance in the colour channels, and the luminance, contrast and structural similarity measure which SSIM yields. In general, we see that the results in table 4 give similar values for all models and colour spaces. On the one side, we see that the baseline model performs marginally better than our proposed model in terms of these metrics for the LAB colour space. On the other hand, we can observe that our proposed YUV models performs marginally better in both metrics.

Our qualitative results show a similar conclusion, although participants were fooled a surprising amount of the times across both models, all of the obtained chi-squared tests concluded that there remained a method and structure to participant's choices (they were not purely random). Similarly the 'fooling rates' were much too similar to conclude decisively any improvement was based on the model and not simply by chance of the participant's answers.

We note also that our Macro-F1 Score of ResNet34 being 0.893 mean images would be passed through a different model than the one intended, ocassionally resulting in images with different colour choices than expected. Even further tuning of the ResNet model could also partially improve performance. Another of the reasons potentially underneath this fact is that as image colourisation is data driven, in the process of doing our ensemble we partitioned our data into smaller categories (as opposed to the full dataset which was used for training the baseline) which may have lead to lower generalisation performance.

5. Related work

Image colourisation can initially be split into two sections based on their level of automation, although our work is on automatic methods, semi-automatic methods in image colourisation are also worth pointing out for completion, these include scribble-based methods such as those by (Min et al., 2020) as well as reference-based methods (Sun et al., 2019)(Gupta et al., 2012) these tasks require input from the user and often struggle in many of the same areas that fully automatic tasks struggle in including colour leakage and colour incorrectness.

The first fully automatic colourisation method was proposed by (Cheng et al., 2015) and used a Convolutional Neural Network, most previous methods used only CNNs using Mean Squared Error as a Loss Function (Iizuka et al., 2016) (Baldassarre et al., 2017) (Deshpande et al., 2015) (Zhang et al., 2016) (Cheng et al., 2017), these approaches were taken further and made use of encoders and decoders such as FusionNet(JWA & KANG, 2021) (Quan et al., 2021) while more recent approaches in CNNs have utilised pixel-wise semantic segmentation (Nguyen-Quynh et al., 2020) (Zhao et al., 2018) (Qin et al., 2022) and produced some of the best results in image colourisation. CNNs however are prone to producing smoother images and lacking detail so research with GANs in this field has recently started gaining momentum. Within GANs there exists a wide range of approaches to varying degrees of success, some approaches vary the colour space used such as (Zhou et al., 2020) with their use of a combination of RGB and YCbCr (called a multi-colour space) along with an iterative Generative Model, many use a data-driven approach with Deep GANs to try to improve results (Cao et al., 2017) (Nazeri et al., 2018) (Kiani et al., 2020) while others use variations of GANs such as a modified CycleGAN, CGANs or dense UNet GANs (Huang et al., 2021) (Xu et al., 2021) (Treneska et al., 2022) (Antic, 2019).

Ultimately many of the approaches in fully-automatic image colourisation taken both with GANs and CNNs suffer from similar problems of unsaturation, colour leakage or colour incorrectness. This speaks strongly to the aforementioned statement that image colourisation is not an easy task as it is an ill-posed problem. One of the more recent approaches that may yield promising results is that of vision transformers as proposed by (Kumar et al., 2021) but is not currently at the same level as other models. Some of the work in image colourisation is also being expanded to video colourisation (Zhang et al., 2019) (Lei & Chen, 2019) this is an interesting area for future research as it looks to tackle the challenge of temporal consistency between images and their colours. Further improvements must first be made to the quality of individual images (or in this case frames) simultaneously to fully make use of video colourisation.

6. Conclusions

We were unable to decisively conclude whether our model incorporating context to Pix2Pix through image classification could improve on our baseline. Although some of this may be due to the lack of definite evaluation metrics for the image colourisation task, we note our approach could have been improved through a different early stopping technique as well as changing our data driven approach. Future work could implement a similar idea using larger amounts of data (like ImageNet with adequate category refinement) and a different early stopping method like the one proposed above to decisively conclude if adding this context improved the quality of images.

7. Appendix

7.1. Code

- **Pix2Pix:** The basic files for the Pix2Pix model applied to image colourisation were obtained from (Shariatnia). These were modified to do Early Stopping, to work with the YUV Colour Space, to collect data about the training process, to produce a per-category model, etc.

- **ResNet34:** The original code was taken from (Inkawhich). Since the original code was meant for finetuning models trained in ImageNet, we had to change this. Additionally, the original code only contained ResNet18, so we had to change it to ResNet34. Furthermore, we included the metric of F1-Score for measuring the training and validation performance, and other changes.

Pix2Pix	LAB	YUV
GAN Plain Loss	1.739	**1.400**
Ensemble	LAB	YUV
GAN Plain Loss	**1.237**	1.593

Table 6. Comparison of how the Pix2Pix baseline model performs with respect to its adversary discriminator in comparison with how our proposed ResNet34+Pix2pix ensemble performs on average against the discriminators of all classes.

		Participant's Judgement		
		Real	AI	Total
Actual Image	Real	725	523	1248
	AI	598	650	1248
	Total	1323	1173	2496

Table 7. Results from our Qualitative Evaluation Survey - LAB

		Participant's Judgement		
		Real	AI	Total
Actual Image	Real	694	554	1248
	AI	579	669	1248
	Total	1273	1223	2496

Table 8. Results from our Qualitative Evaluation Survey - Baseline (LAB)

		Participant's Judgement		
		Real	AI	Total
Actual Image	Real	740	508	1248
	AI	630	618	1248
	Total	1370	1126	2496

Table 9. Results from our Qualitative Evaluation Survey - Baseline (YUV)

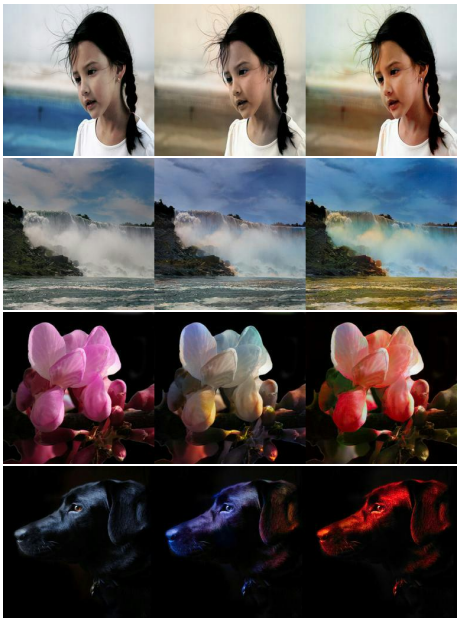

Figure 5. (Left) Ground Truth Images (Center) Baseline Model with Pix2Pix (Right) LAB Colour Space Model

Figure 6. Image produced in the 200^{-th} epoch of the training process of the baseline model. The true colour images are in the left column while the images produced by Pix2Pix using the LAB colour space are those in the right hand side.

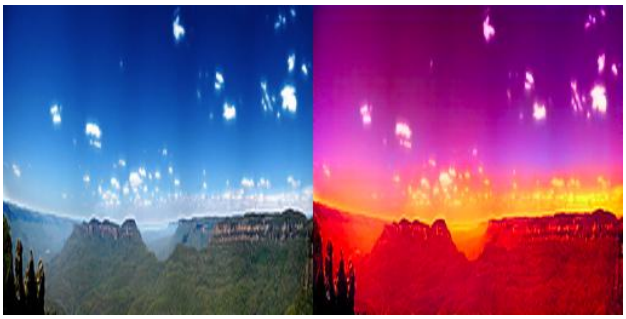

Figure 7. Example 1

Figure 8. Example 2

Figure 9. The left-hand side is the true colour images while those on the right are the ones produced by Pix2Pix after being trained during 200 epochs for LAB. For further images refer to the images in the folder "Validation_LAB_Baseline_no_pretrain".

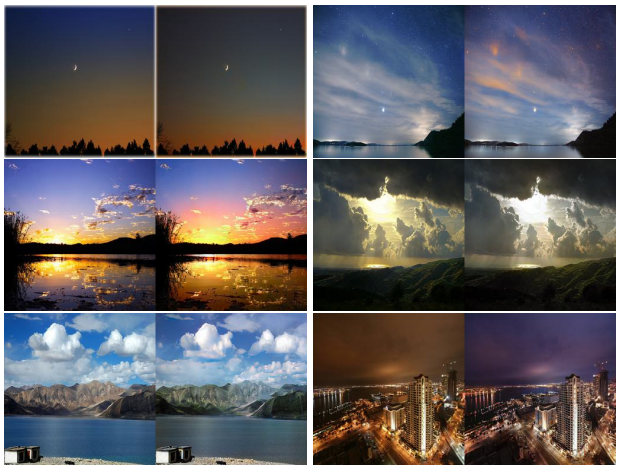

Figure 10. A collection of images classified by our trained ResNet model as 'scenery' and colourised using our colourisation model (R) paired with their corresponding ground truth images (L)

References

Antic, Jason. Deoldify, 2019.

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

Ballester, Coloma, Bugeau, Aurélie, Carrillo, Hernan, Clément, Michaël, Giraud, Rémi, Raad, Lara, and Vitoria, Patricia. Influence of color spaces for deep learning image colorization. *arXiv preprint arXiv:2204.02850*, 2022.

Cao, Yun, Zhou, Zhiming, Zhang, Weinan, and Yu, Yong. Unsupervised diverse colorization via generative adversarial networks. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2017, Skopje, Macedonia, September 18–22, 2017, Proceedings, Part I 10*, pp. 151–166. Springer, 2017.

Cheng, Zezhou, Yang, Qingxiong, and Sheng, Bin. Deep colorization. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 415–423, 2015. doi: 10.1109/ICCV.2015.55.

Cheng, Zezhou, Yang, Qingxiong, and Sheng, Bin. Colorization using neural network ensemble. *IEEE Transactions on Image Processing*, 26(11):5491–5505, 2017. doi: 10.1109/TIP.2017.2740620.

Deshpande, Aditya, Rock, Jason, and Forsyth, David. Learning large-scale automatic image colorization. In *Proceedings of the IEEE international conference on computer vision*, pp. 567–575, 2015.

Goodfellow, I, PougetYAbadie, J, and Mirza, M. B. xu, d. wardeyfarley, s. ozair, a. courville, and y. bengio. generative adversarial nets. *Advances in Neural Information Processing Systems (NIPS)*, pp. 2672–2680, 2014.

Gupta, Raj Kumar, Chia, Alex Yong-Sang, Rajan, Deepu, Ng, Ee Sin, and Zhiyong, Huang. Image colorization

using similar images. In *Proceedings of the 20th ACM international conference on Multimedia*, pp. 369–378, 2012.

Hensman, Paulina and Aizawa, Kiyoharu. cgan-based manga colorization using a single training image. In *2017 14th IAPR International Conference on Document Analysis and Recognition (ICDAR)*, volume 3, pp. 72–77. IEEE, 2017.

Huang, Shanshan, Jin, Xin, Jiang, Qian, Li, Jie, Lee, Shin-Jye, Wang, Puming, and Yao, Shaowen. A fully-automatic image colorization scheme using improved cyclegan with skip connections. *Multimedia Tools and Applications*, 80(17):26465–26492, 2021.

Huang, Shanshan, Jin, Xin, Jiang, Qian, and Liu, Li. Deep learning for image colorization: Current and future prospects. *Engineering Applications of Artificial Intelligence*, 114:105006, 2022.

Iizuka, Satoshi, Simo-Serra, Edgar, and Ishikawa, Hiroshi. Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Transactions on Graphics (ToG)*, 35(4):1–11, 2016.

Inkawhich, Nathan. Finetuning torchvision models. URL https://pytorch.org/tutorials/beginner/finetuning_torchvision_models_tutorial.html#%20sphx-glr-beginner-finetuning-torchvision-models-tutorial-py%20Last%20time%20checked:%2019/11/2022.

Isola, Phillip, Zhu, Jun-Yan, Zhou, Tinghui, and Efros, Alexei A. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1125–1134, 2017.

JWA, MINJE and KANG, MYUNGJOO. Grayscale image colorization using a convolutional neural network. *Journal of the Korean Society for Industrial and Applied Mathematics*, 25(2):26–38, 2021.

Kalvankar, Shreyas, Pandit, Hrushikesh, Parwate, Pranav, Patil, Atharva, and Kamalapur, Snehal. Astronomical image colorization and up-scaling with conditional generative adversarial networks. *INFORMATIK 2022*, 2022.

Kiani, Leila, Saeed, Masoud, and Nezamabadi-pour, Hossein. Image colorization using generative adversarial networks and transfer learning. In *2020 International Conference on Machine Vision and Image Processing (MVIP)*, pp. 1–6. IEEE, 2020.

Kumar, Manoj, Weissenborn, Dirk, and Kalchbrenner, Nal. Colorization transformer. *arXiv preprint arXiv:2102.04432*, 2021.

Lei, Chenyang and Chen, Qifeng. Fully automatic video colorization with self-regularization and diversity. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 3753–3761, 2019.

LIACS Medialab, Leiden University. Mirflickr-25000, 2008. URL https://press.liacs.nl/mirflickr/#sec_copyright.

Liang, Yihuai, Lee, Dongho, Li, Yan, and Shin, Byeong-Seok. Unpaired medical image colorization using generative adversarial network. *Multimedia Tools and Applications*, 81(19):26669–26683, 2022.

Min, Lihua, Li, Zhenhua, Jin, Zhengmeng, and Cui, Qiang. Color edge preserving image colorization with a coupled natural vectorial total variation. *Computer Vision and Image Understanding*, 196:102981, 2020.

Montavon, Grégoire, Orr, Geneviève, and Müller, Klaus-Robert. *Neural networks: tricks of the trade*, volume 7700. springer, 2012.

Nazeri, Kamyar, Ng, Eric, and Ebrahimi, Mehran. Image colorization using generative adversarial networks. In *Articulated Motion and Deformable Objects: 10th International Conference, AMDO 2018, Palma de Mallorca, Spain, July 12-13, 2018, Proceedings 10*, pp. 85–94. Springer, 2018.

Nguyen-Quynh, Tram-Tran, Kim, Soo-Hyung, and Do, Nhu-Tai. Image colorization using the global scene-context style and pixel-wise semantic segmentation. *IEEE Access*, 8:214098–214114, 2020.

Qin, Xujia, Li, Mengjia, Liu, Yuehui, Zheng, Hongbo, Chen, Jiazhou, and Zhang, Meiyu. An efficient coding-based grayscale image automatic colorization method combined with attention mechanism. *IET Image Processing*, 16(7):1765–1777, 2022.

Quan, Tran Minh, Hildebrand, David Grant Colburn, and Jeong, Won-Ki. Fusionnet: A deep fully residual convolutional neural network for image segmentation in connectomics. *Frontiers in Computer Science*, 3:613981, 2021.

Shariatnia, Moein. Image colorization with u-net and gan tutorial. URL https://github.com/moein-shariatnia/Deep-Learning/tree/main/Image%20Colorization%20Tutorial.

Sun, Qian, Chen, Yan, Tao, Wenyuan, Jiang, Han, Zhang, Mu, Chen, Kan, and Erdt, Marius. A gan-based approach toward architectural line drawing colorization prototyping. *The Visual Computer*, pp. 1–18, 2022.

Sun, Tsai-Ho, Lai, Chien-Hsun, Wong, Sai-Keung, and Wang, Yu-Shuen. Adversarial colorization of icons based on contour and color conditions. In *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 683–691, 2019.

Treneska, Sandra, Zdravevski, Eftim, Pires, Ivan Miguel, Lameski, Petre, and Gievska, Sonja. Gan-based image colorization for self-supervised visual feature learning. *Sensors*, 22(4):1599, 2022.

Wang, Zhou, Bovik, Alan C, Sheikh, Hamid R, and Simoncelli, Eero P. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Wu, Min, Jin, Xin, Jiang, Qian, Lee, Shin-jye, Liang, Wentao, Lin, Guo, and Yao, Shaowen. Remote sensing image colorization using symmetrical multi-scale dcgan in yuv color space. *The Visual Computer*, 37:1707–1729, 2021.

Xu, Jiangtao, Lu, Kaige, Shi, Xingping, Qin, Shuzhen, Wang, Han, and Ma, Jianguo. A denseunet generative adversarial network for near-infrared face image colorization. *Signal Processing*, 183:108007, 2021.

Zhang, Bo, He, Mingming, Liao, Jing, Sander, Pedro V, Yuan, Lu, Bermak, Amine, and Chen, Dong. Deep exemplar-based video colorization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8052–8061, 2019.

Zhang, Richard, Isola, Phillip, and Efros, Alexei A. Colorful image colorization. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part III 14*, pp. 649–666. Springer, 2016.

Zhao, Jiaojiao, Liu, Li, Snoek, Cees GM, Han, Jungong, and Shao, Ling. Pixel-level semantics guided image colorization. *arXiv preprint arXiv:1808.01597*, 2018.

Zhou, Jinjie, Hong, Kai, Deng, Tao, Wang, Yuhao, and Liu, Qiegen. Progressive colorization via iterative generative models. *IEEE Signal Processing Letters*, 27:2054–2058, 2020.