UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Escola Tècnica Superior d'Enginyeria
de Telecomunicació de Barcelona

GAN-based Image Colourisation with Feature Reconstruction Loss

Master Thesis
submitted to the Faculty of the
Escola Tècnica d'Enginyeria de Telecomunicació de Barcelona
Universitat Politècnica de Catalunya
by

Laia Tarrés Benet

In partial fulfillment
of the requirements for the master in
Master in Advanced Telecommunication Technologies **ENGINEERING**

Advisor at UPC: Dr. Xavier Giró i Nieto
Advisor at BBC: Dr. Marta Mrak
Barcelona, Date 19 May 2020

Contents

List of Figures

List of Tables

Abstract

Automatic image colourisation is a complex and ambiguous task due to having multiple correct solutions. Previous approaches have resulted in desaturated results unless relying on significant user interaction.

In this thesis we study the state of the art for colourisation and we propose an automatic colourisation approaches based on generative adversarial networks that incorporates a feature reconstruction loss during training. The generative network is framed in an adversarial model that learns how to colourise by incorporating a perceptual understanding of the colour. Qualitative and quantitative results show the capacity of the proposed method to colourise images in a realistic way, boosting the colourfulness and perceptual realism of previous GAN-based methodologies.

We also study and propose a second approach that incorporates segmentation information in the GAN framework and obtain quantitative and qualitative results.

Acknowledgements

First of all, I wanted to thank Marta Mrak for accepting me to join their Research and Development team at the BBC. I enjoyed and learned a lot from this project. I want to specially thank Marc Górriz for all the advice and dedication during the whole project. It has been a pleasure getting to know you even if it had to be virtually.

Thanks also to Xavi Giró for your constant motivation and support during the whole experience, and for always pushing me to perform my best.

Finally, I want to thank my family and close friends, who always supported me in an unconditional way.

1 Introduction

1.1 Motivation

The British Broadcasting Corporation (BBC) owns a historical archive of grayscale media. These contents can be nowadays reproduced in displays that allow much richer user experiences: most screens nowadays support colour images and much higher spatial and temporal definition than those available in the past.

This master thesis addresses the task of media adaptation to richer representations so that contents captured with deprecated technology can exploit the latest advances in display technology. In particular, we focus in the basic image enhancement of adding colour to grayscale images.

The computer vision field has studied different ways to reproduce how the human brain processes visual information. One of the many visual tasks that humans are capable of is imagining realistic colours for a grayscale image. Although grayscale images do not explicitly contain colour information, there are clues coded in them, like the type of object, the texture or the lighting. A variety of techniques have been proposed to produce perceptually salient and colorful images from their grayscale counterparts. Recently, deep convolutional networks have shown potential in different image-to-image translation tasks, and among them, automatic image colourisation.

This work focuses on image colourisation by training deep neural networks with different loss terms: colourisation, adversarial, perceptual and segmentation. In our work, we consider as a baseline solutions based on the adversarial loss, popularly known as generative adversarial networks (GANs). This training paradigm aims at producing novel and realistic samples from a learned data distribution. However, GANs also present some limitations in terms of being able to generate images with the conditions specified by the user. This is particularly true for the task of colourisation since different colours are plausible for the same grayscale image. Our main contributions focus in reviewing the limitations and possible solutions to image colourisation with GANs.

1.2 Contributions

This master thesis builds on top of an existing solution from BBC [7] to colourize images.In particular, our contributions are the following:

- Improvement of the BBC colourization solution by adding a loss term of feature reconstruction.
- New implementation of the colourization technique in the PyTorch deep learning framework, complementary to the existing Keras implementation.
- Introduction of the image segmentation maps as an additional cue to guide the colourization process.

The improvement of the existing solution has been accepted after peer-reviewing as a poster in the CVPR 2021 Women in Computer Vision Workshop, which will be hold

online on June 19, 2021. The submitted accepted abstract is included as an annex to this report.

1.3 Work Plan

This project was funded by the Research & Development department at the British Broadcast Company (BBC) through the the Image Processing Group (GPI) at the Universitat Politècnica de Catalunya (UPC). Between February 2020 and May 2021, we had regular video calls every week to develop the work plan. The work plan originally considered an internship at the BBC premises in London starting Summer 2020 but, due to the COVID-19 global pandemic and derived mobility restrictions, the whole work was developed remotely.

1.3.1 Tasks

The different work packages for the project are defined as follows:

- WP 1: Definition of project
- WP 2: Research about state of the art
- WP 3: Datasets
- WP 4: Adaption of software to the GPI computational service
- WP 5: Adaption of software to Pytorch
- WP 6: Research and experimentation on improvements
- WP 7: Participation in a WiCV
- WP 8: Research and experimentation on further improvements
- WP 9: Final Documentation

1.3.2 Milestones

The milestones of the project are defined in Table 1:

WP	Milestone	Date
1	Definition of the project	15/03/2020
3	Dataset ready to be used	21/03/2020
4	Run the original code in Keras	28/03/2020
4	Generate a baseline for our task with the Original code	05/04/2020
6	Research state-of-the-art for improvements	15/04/2020
6	Define the strategy to improve the baseline model	31/05/2020
6	Run the code with our first improvement	01/07/2020
5	Run the original code + our first improvement in Pytorch	10/09/2020
8	Further research for state-of-the-art improvements	10/02/2021
8	Define the strategy to improve out latest model	10/03/2021
8	Run the code with our second improvement	10/05/2021
9	Submit to CVPR 2021 Women in Computer Vision Workshop	10/05/2021
9	Deliver report to BBC & UPC	23/05/2021
9	Oral defense at UPC	28/05/2021

Table 1: Table with the milestones of the project.

1.3.3 Gantt diagram

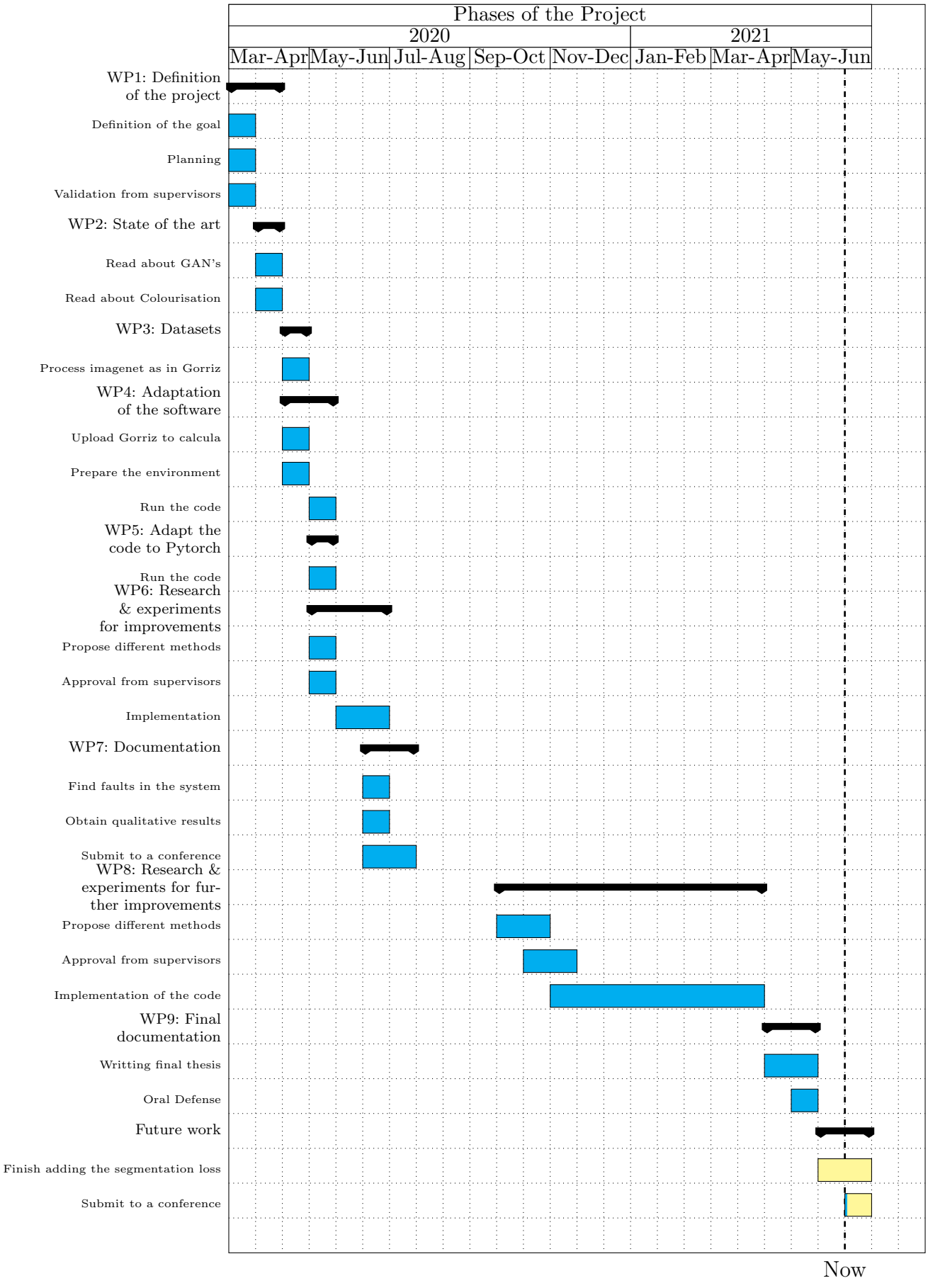


Figure 1: Gantt diagram of the project

1.3.4 Deviations from the original plan

Throughout the project there have been deviations from the original plan, mainly on tasks that took longer than expected to develop.

The main issues were related to replicating the implementations that we had in Keras with Pytorch. Both of this software development frameworks can be used to work with deep learning. However, Keras is a higher level framework, so even if some simple implementations are much easier to code and run, when it comes to custom code, it is much harder to debug. When we were trying to replicate the baseline results with Pytorch, it was almost impossible to obtain the same results. However, after some time we were able to duplicate them. Another time-consuming issue that occurred during the development of the project was the development of the segmentation addition with Keras, we had some issues with the implementation that we were not able to solve so we want back to Pytorch. In the end there was a lot of effort put in the implementation of all the solutions proposed for this paper even if some of them were not fruitful.

All of this different problems affected the final submission date, and it was extended from February 2021 to May 2021.

2 Theoretic Background

This section presents the main concepts and required background knowledge to understand the models, methodology and details that will be approached. We expect the reader to have minimum knowledge of the basic concepts in Deep Learning [1].

2.1 Image colourisation definition

Image colourisation is the task of adding plausible colour to grayscale images. This transformation requires predicting a three dimensional colour-valued mapping from a real-valued grayscale image, which leads to an undetermined problem. However, the semantics and texture provide cues for many regions. The goal of image colourisation in not to recover the ground truth colour but to potentially fool a human observer.

Adding chromatic information to gray-scale visual data is a widely used technique in commercial applications and a researched topic in the academia world due to its various applications. Recently, deep learning has enabled new algorithms for colourisation that can generalise better the distribution of colours. However, existing methods still suffer ambiguity when trying to predict realistic colours and often result in de-saturated results.

There are multiple works that implement a semi-supervised or supervised colourization, such as adding scribbles to an image and propagating that colour though the same textures. Although this approach provide plausible results, we will focus on automatic image colourisation, where the network learns plausible distribution for the colours.

2.2 Colour Representation

Colour representation is a fundamental problem in the computer vision field. For the Colourisation task, we are using CIE Lab colour space to represent the colourised images. For this specific colour representation we have three channels. The first one L is the lightness channel and it is defined in the range 0-100, representing black at L=0 and white at L=100. This channel is equivalent to a grayscale image of the given colour picture, and it contains most of the visual information, such as object edges and lighting effects. Colour information is stored in the two last channels: a - green to red and B- blue o yellow. The range of the values goes from -110 to +110.

It is important to note that with the CIE Lab colour representation we already have the colour channels represented independently from the luminance channel. In this thesis, we will predict the two colour channels, a and b given a luminance channel L of an image.

The CIE Lab colour space is chosen similarly to others in the literature, because it is designed to maintain perceptual uniformity and is more perceptually linear than other colour spaces [57].

2.3 Generative Adversarial Networks (GAN)

Since 2012, Deep Convolutional Neural Networks have revolutionized the computer vision field, where they have surpassed previous baselines in most tasks. Specially when it comes to generative models, GANs [3] have become one of the most use architectures due to its realistic results and adaptability to many tasks. Some real-world applications include image generation [5, 6, 8], image to image translation [4], [9], text to image [10], super-resolution [11] and photo inpanting [12] between others.

2.3.1 GAN architecture

The principle behind GAN is that we have a joint model where two distinct networks are trained together. The first network, which we call generator, is trained to generate similar images than the input. The second network, the discriminator, is trained to distinguish the original input images from the generated ones from the generator.

In the task of colourisation, the generator is used to produce plausible colour channels while the discriminator learns to distinguish between real colourised images and the ones generated by the discriminator.

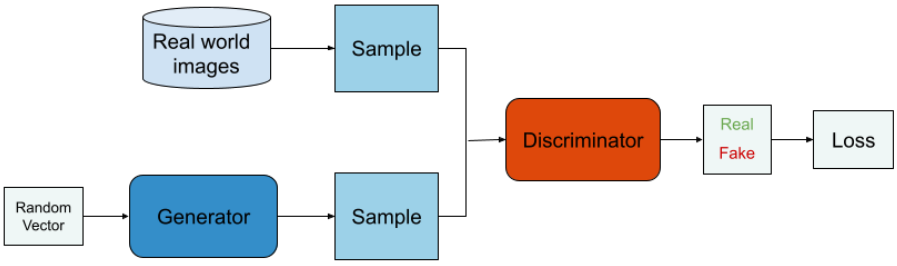


Figure 2: Architecture of a vanilla GAN.

2.3.2 Objective function

The GAN scheme can be understood as a min-max game where the generator and discriminator compete between them to achieve the Nash equilibrium. At the same time, the discriminator is trying to minimize the loss while the generator tries the maximize the loss.

The objective function for the vanilla GAN is defined in the Equation 1,

$$\min_G \max_D V(D,G) = E_{xP_{data}(x)}[\log D(x)] + E_{xP_z(z)}[\log(1 - D(G(z)))] \quad (1)$$

,

where we have a vector of random noise, z that follows P_z distribution, and a real sample x extracted from the dataset that follows P_{data}. G and D are respectively the generator and discriminator. Where the output of G is the generated sample and the output of D is the probability of authenticity. Finally E_x and E_z represent the expected log likelihood.

During training the network must perform two steps:

1. The discriminator is frozen while we train the generator to get generated data that "fools" the discriminator. During this step the generator is minimizing the likelihood of D predicting that the generated $G(z)$ is not genuine.
2. The generator is frozen while we train the discriminator to be able to distinguish the generated data from the real input data. During this step, discriminator maximises the expected log likelihood of D predicting that real world data is genuine.

This min-max game requires many real samples, since the discriminator needs to learn the distribution of the real data in order to distinguish it from the fake fake samples. But not only that, the generator needs "creativity" to generate samples that will fool the discriminator. This creativity comes from the random noise vector that contains information that affects the generated sample.

However, in the case of colourisation, the generator will not take a point from the latent space as the input. The source of randomness comes with the addition of dropout layers both during training and prediction. Similarily, batch normalization is also used in the same way, meaning that statistics are calculated for each batch, and are not fixated in the training process so they add randomness, that allows the generator to have this source of "creativity".

2.3.3 Conditional GAN

Conditional GANs (cGAN) [2] provide a control mechanism over the generated data by adding extra input information that acts as a condition. This conditional data forces the generator to produce a more specific output instead of a generic sample.

The min-max game is still similar to the one presented to train the GAN but adding the condition and with a few peculiarities. It was found that using the traditional vanilla GAN had a slow convergence rate for the generator [3], and a new objective function is depicted in Equation 2,

$$\min_G L(D,G) = -E_{xP_z(z)}[\log D(G(z))] \tag{2}$$

,

which can be interpreted as the maximization of the probability of the discriminator being mistaken and rewritten as a minimization problem. cGAN also proposes adding an extra term in the objective function to generate images closer to the ground truth $L1 = \lambda||G(z) - y||$, where $G(z)$ is the prediction, y is the ground truth and λ is a parameter set to 100.

The final objective function for the cGAN is defined as follows:

$$\min_G \max_D V(D,G) \tag{3}$$

$$V(D,G) = L(D) + L(G) \tag{4}$$

$$L(D) = E_{xP_{data}(x)}[\log D(x)] + E_{xP_z(z)}[\log(1 - D(G(z)))] \tag{5}$$

$$L(G) = -E_{xP_z(z)}[\log D(G(z))] + \lambda||G(z) - y|| \tag{6}$$

For colourisation, the model only predicts the A and B channels, so we are keeping the grayscale from the input and concatenating it with the output. This way, we are conditioning the network to have the same grayscale information at the output, and predicting the colour channels.

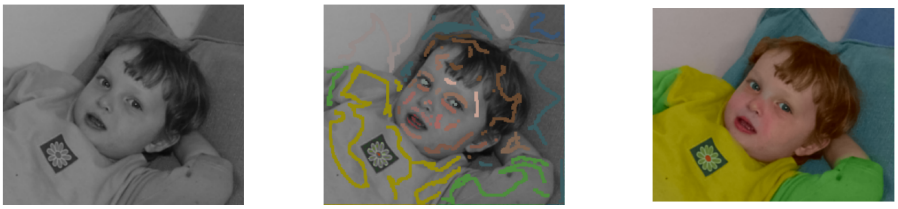
3 State of the art for colourisation

This sections reviews other works in the field of image colourisation, with particular focus on the contributions that have inspired this thesis.

We can group these techniques in two main approaches: semi-supervised and supervised ones.

3.1 Semi-Supervised Colourisation

In semi-supervised case colourisation, the user provides hints to the algorithm about how the final result should look like. These hints can come in the form of scribbles - small patches of colour in specific areas of the image- or a coloured reference image [53] - from which the network will match the texture and statistical data. Colourisation methods based on colour-scribbles generally use an optimization framework without explicit parameter learning to propagate the colour from the colour patches onto the whole image. An example of methods based on colour-scribbles can be seen in fig. 3.



(a) Target image (b) User created scribbles (c) Resulting image

Figure 3: Example of a method based on colour-scribbles.

The first method was proposed by Levin et al. [41] and later improved by Huang et al. [42] that exploited edge detection in order to remove colour bleeding over object boundaries. Luan et al. [43] method automatically labels pixels that should share similar colours and they extend the scribbles into those similar pixels.

Colourisation based on a coloured reference image, also known as style transfer, has also been explored by a few researchers. Welsh et al. [44] and Gupta et al. [45] proposed feature matching to transfer the colour information.

In Deep Colorization by Cheng et al. [46], they introduce the use of deep features and deep feature matching to both extract and obtain the per-pixel colourisation.

Combining both methods, Liu et al. [47] automatically generates scribbles from the reference and propagates it through the target image using [41].

While both methods are able to obtain visually plausible results, it is important to note that they need user interaction. And as in most fields within the machine-learning field, we are always prioritizing having end-to-end automatic algorithms.

3.2 Unupervised Colourisation

Unsupervised image colourisation is a fully automated solution, since the user does not provide any hint to algorithm regarding the expected colour output.

Unsupervised image colourisation is in particular trained in a self-supervised set up [19] where the colour images are converted to grayscale. This allows quick collection of training data suitable for training deep neural networks. A first approach to image colourisation with deep learning was proposed by Cheng et al. [16]. They approached the problem by formulating a least square minimization equation solved with deep neural networks. In their work, they grouped reference images by clusters, and trained a dense neural network that was able to extract pixel-wise chrominance values for each similar clustered image.

The capabilities of Generative Adversarial Networks (GANs) [21] for producing realistic samples was firstly applied for image colourisation in *Pix2Pix* by Isola et al. [4]. In particular, *Pix2Pix* applies the concept of Conditional GAN (cGAN) [35] in an image-to-image translation framework implemented on top of a U-Net [34] convolutional architecture. Some training improvements to his set up were proposed by Nazeri et al. [27] to allow the colourisation of high definition images.

Gorriz et al. [7] increased the colour saturation obtained by an off-the-shelf *pix2pix* model by adding batch and instance normalization, spectral normalization normalization as a regularization step to the training, as well as multiple discriminators. This is the existing approach at BBC we built upon.

ChromaGAN [30] obtained more lively and plausible colours by adding a module to the generator for predicting the distribution of semantic classes. They use a generator that contains two submodules, one predicts the chrominance values and the second one the class distribution, obtaining extra segmentation information.

A similar idea, to improve the quality and diversity of the generated images was implemented in the OASIS model [40]. OASIS introduced a novel discriminator with segmented pixel-wise loss instead of the typical adversarial loss model. By providing stronger supervision during training, they achieved synthetic images of higher fidelity with better alignment to their input.

InstColorization [39] proposed a new deep learning approach to achieve instance-aware colourisation. They obtained lively and plausible results without the use of generative networks by a first step of object detection, which allowed a later figure-background separation, a second step of colourising the foreground instances through a colourisation network, and a final step of combining all instances within the image with a final fusion module.

Our work explores the benefits and challenges of this approaches by reproducing the results of *Pix2Pix* and *Gorriz*. In addition, we propose novel approaches to exploit the semantic segmentation maps, inspired by *OASIS* and *InstColorization*.

4 Methodology

This section presents the methods we tested ad proposed to achieve automatic image colourisation.The first part of our project consisted on improving the baseline provided by the BBC, by adding a feature reconstruction loss. The second part tried to exploit the pixel-wise labels provided in the semantic segmentation maps to further improve the results.

4.1 Baselines

While our main goal was improving the existing baseline by BBC [7], we also reproduced other more basic methods to fully understand the challenges before proposing new solutions.

4.1.1 Colourisation loss only

The most basic baseline consists on a simple U-Net that is able to predict the a and b channels given the luminance channel. We chose this network because it would later be adapted in an adversarial framework, so we would be able to compare the results with and without the adversarial loss.

The U-Net architecture [34] is a well-known symmetric encoder-decoder based architecture. The encoder consists of n=7 downsampling layers with an increasing number of filters. The decoder has the corresponding symmetric filters, but adding skip connections to allow the flow of the low-level information in the network. This type of network is usually used for image to image translation, as it allows to take a $n \times n$ dimensional input and transform it into another $n \times n$ dimensional output.

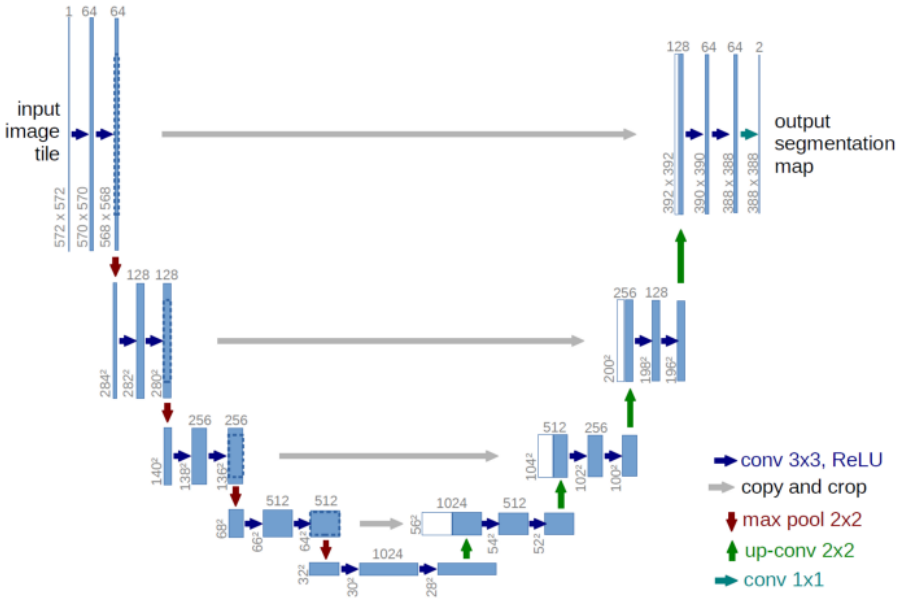


Figure 4: Architecture the U-Net.

The encoder part of the model is comprised of convolutional layers that use a 2×2 stride to downsample the input source image down to a bottleneck layer. The decoder part of the model reads the bottleneck output and uses transpose convolutional layers to upsample to the required output image size.

Skip connections are added between the layers with the same sized feature maps so that the first downsampling layer is connected with the last upsampling layer, the second downsampling layer is connected with the second last upsampling layer, and so on. The connections concatenate the channels of the feature map in the downsampling layer with the feature map in the upsampling layer. The concatenating operation ensures that the features that are learned while contracting the image will be used to reconstruct it. For the loss term, we use a type of regression loss, L1.

Although U-Net does not provide state-of-the-art results for segmentation, it is a robust, fast and easy-to-implement baseline that has been the inspiration for many posterior works.

4.1.2 Colourisation and adversarial losses

Pix2Pix [4] is popular solution for image-to-image translation that is based on a U-Net architecture that adds an adversarial loss term to the loss term of the downstream task, such as colourisation, converting maps to realistic satellite photographs, from sketches of products to photographs, etc.

The image-to-image translation task with the adversarial loss actually corresponds to the conditional GAN (cGAN) presented in Section 2.3.3. In the Pix2Pix model, we have a generation of a target image that is conditional on a source image. The Discriminator in this case is provided with the source image and the target image and must determine weather the target is a plausible transformation of the source image.

The architecture of the discriminator is a *PatchGAN*, which is a patch-based fully convolutional Network [52]. The discriminator performs conditional-image classification, it takes the source and the target and predicts the likelihood of whether the target image is real or generated. It is based on the effective receptive field of the model, and it is designed to do a classification of a patch of the input image, providing an activation map of classification values. Since we have an output for each patch, the labels will have the same shape accordingly. The authors reason that this enforces more constraints that encourage sharp high-frequency detail. Additionally, the PatchGAN has fewer parameters and runs faster than classifying the entire image.

4.1.3 BBC Górriz baseline

The baseline from BBC proposed by Górriz et al. [7] improved the *Pix2Pix* by introducing batch [48] and instance [50] normalization in both the generator and discriminator layers. These techniques improved the stability of the adversarial loss during training, leading to better colourisation of a variety of images from large multi-class datasets.

Applying mini-batch normalisation such as Batch Normalization [48] has become a popular practice to accelerate the training of deep neural networks. For the cGAN architecture,

it was proven that adding batch normalization in the generator and discriminator can stabilise the GAN training and prevent the mode collapse due to poor initialisation [49]. The authors noticed that batch normalization preserves content-related information by reducing the covariance shift within the mini-batch during training as it uses the internal mean and variance of the batch to normalise each feature channel

Another concept that the introduce in this work is Instance Normalisation (IN) [50]. Which benefited the task of stylisation. This technique uses the statistics of an individual sample instead of the whole mini-batch to normalise features. They noted that similar to stylisation, image colourisation aims to capture information by learning features that are invariant to appearance changes, with the goal to colourise within a mini-batch of variable content.

Their proposal was inspired by IBN-Net [51], where batch normalisation and instance normalisation were combined to exploit both normalisation capabilities in style transfer, achieving a more stable training that resulted in an improved capacity of the GAN model. Górriz et al [7] adapt the IBN-Net architecture to the pix2pix model.

Following the discussion presented in IBN-Net, where the shallow layers usually contain the appearance variance while the deeper layers have higher feature discrimination content, they avoided IN in deep layers to preserve content discrimination, while keeping BN in the whole architecture to preserve content-related features.

To improve generalisation of the network, they also introduced some changes in the use of weight regularisation. Weight regularisation proportionally penalises the weights of the network based on their size. To avoid small changes in the input leading to large changes in the output. Also the activation functions as sigmoid in the discriminator can lead to unbounded gradients. To prevent this particular anomalies, they introduced Spectral Normalisation [56] to control the Lipsichtz constant of the discriminator.

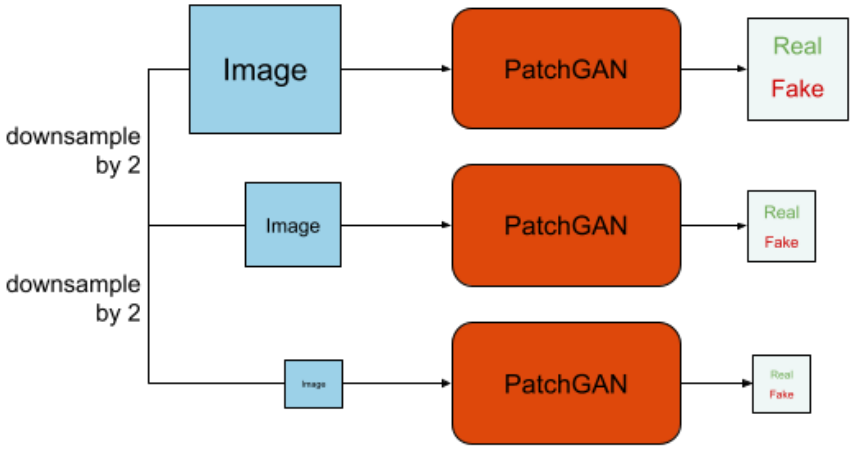


Figure 5: Schema from the multi-resolution discriminator used in [7].

Their final contribution was adding multiple discriminators, as represented in Figure 5. The idea comes from the analysis of the PatchGAN discriminator used in the *pix2pix*

framework, which tries to increase the receptive field of the discriminator, but usually leading to blurry effects and tilling artifacts for the colourisation task. They proposed using multi-scale discrimination [54], by using N discriminators that share architecture but have different scale inputs. This allows the discriminator to keep the original architecture and to obtain variable receptive fields that are larger at the coarsest levels.

4.2 Colourisation, adversarial and feature reconstruction losses

In our first proposal we add a new feature (or perceptual) reconstruction loss and include it in the objective function used in *Pix2Pix* [4]. This loss was proposed by Johnson et al. [28] for image translation tasks, defined as the squared and normalized Euclidean distance between activations produced in the early layers of the network for the output image and the target image.

Mahendran et al. [29] showed that using a feature reconstruction loss for training image transformation networks encourages the output image to be perceptually similar to the target image, but does not force them to match exactly. In our implementation, rather than using only per-pixel loss functions depending only on low-level pixel information, we train our networks using an additional perceptual loss function that depend on high-level features from a pretrained loss network. During training, perceptual losses measure image similarities more robustly than per-pixel losses.

The main idea is that when we feed an image to a pretrained network for image classification, the network has already learned the perceptual and semantic information that we would like to measure. So comparing the network's activations from the ground truth and the generated image provides perceptual information.

For our experiments, we tested different pretrained neural networks to extract the features to be compared. The first group of experiments consisted on image classification networks pretrained with ImageNet: a VGG16 network [32], and ResNet50 network [33].

The second group of experiments are based on segmentation networks: U-net [34] and FPN [36], that had been pretrained with the COCO dataset for semantic segmentation, in addition to an even earlier pretraining for classification with ImageNet.

4.3 Colourisation and segmentation losses

Training GANs is a challenging task in terms of optimization given the delicate balance needed between the generator and the discriminator. Inspired by *InstColorization* [39], a framework that avoids using an adversarial loss for instance colourisation, we explored this venue when colourising the whole image.

In our case, we added segmentation cues when training our model, so that it could learn to discern between objects at the same time as colourising them. Our hypothesis was that new loss term would prevent the colour bleedding at the object contours that we had observed in some results from the baselines.

We adopt a U-Net [34] similar to the one used as a generator in the previous approach, but now the model would predict the segmentation maps in addition to the colour *ab*

channels. We considered two possible architectures: with a shared decoder for both tasks, or with separate decoders for the segmentation and colourization tasks.

4.3.1 Shared decoder

The solution with a shared decoder reused the Generator model from the previous implementations but changes the last convolutional layer to output a dimension of $n_classes * 2$. Therefore, in the input we have a grayscale image of shape [1,height, width], and at the output we have an image of shape $[2 * nclasses$, height, width], that will represent the a and b channels for each class contained in the original data.

This architecture is depicted in Figure 6.

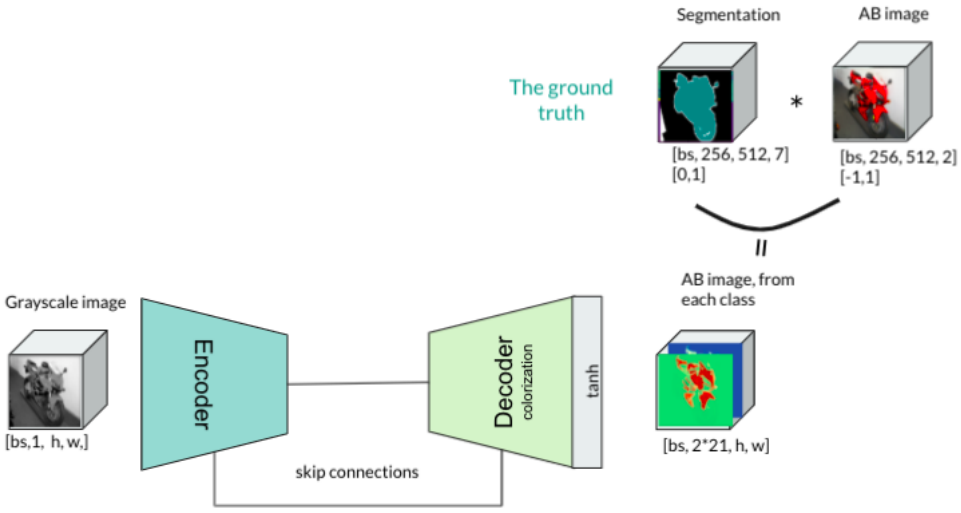


Figure 6: Colourization and segmentation with a shared decoder.

To train the network, we match the original a and b channels for each class with the generated a and b channels. It is in this novel training process that we infer the segmentation cues into the model. As in the baseline models, we have used the regression L1 loss to compare the different channels for each class.

4.3.2 Separate decoders

As an alternative, we also considered the adding a new branch to the U-Net decoder, so that the weights for the colourization and segmentation tasks would not be shared.

This solution is shown in Figure 7.

The model was firstly trained to segment the images, and once it achieved acceptable results, the encoder + colourising branch was trained using both the segmentation losses and colourisation losses.

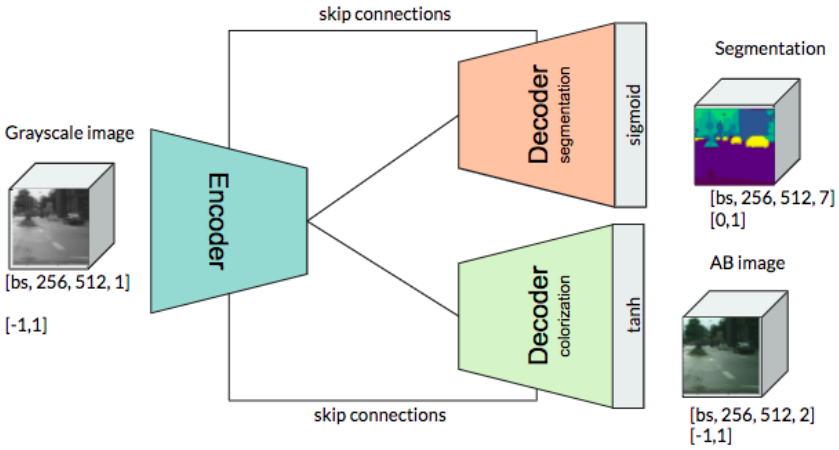


Figure 7: Colourization and segmentation with separate decoders.

5 Experiments

During this project we developed a few experiments to achieve the best plausible coloured images. However, evaluating the quality of a colourised image in a quantitative way is a challenging task given the high subjectivity it implies.

Therefore, quantitative measures reflecting how close the outputs are to the ground truth data may not characterise the human perception of the problem. Nevertheless, we have used quantitative measures used in the literature in order to compare the results of the proposed methods.

The most used metrics for this task are PSNR (Peak Signal-to-Noise Ratio) and SSIM (Structural Similarity Index Measure). The Kullback Liebler Divergence and Jenson Shannon Divergence have also been used for this task, which are divergences computed between the logarithmic colour histograms generated by the test images for the real dataset and the test images generated by the model.

For the qualitative results, ChromaGAN [30] proposed a perceptual test, where participants are shown a picture and later asked to choose between if the colours are real or generated. The perceptual test must be done in a very concise way, and it is subject to many mishaps that can happen if it is not adequately planned.

5.1 Colourisation, adversarial and feature reconstruction losses

5.1.1 Implementation details

We defined as baselines a U-net [34] architecture as generator, and PatchGAN as the discriminator, the same approach as in *pix2pix* [4].

Training examples were sampled from the ImageNet dataset [31], particularly from the 1,000 synsets selected for the ImageNet Large Scale Visual Recognition Challenge (ILSVRC) 2012. Samples were selected from the reduced validation set, containing 50.000 RGB images uniformly distributed as 50 images per class. The test dataset wsa created by randomly selecting 10 images per class from the training set, generating up to 10,000 examples. All images were resized to 256 x 256 pixels and converted to the CIE Lab colour space.

The best results were obtained with the VGG16 model. For our final model, we trained computed the feature reconstruction loss on *block3 conv3* layer from the VGG16 network. This configuration was trained for 23 epochs, during 36 hours. The weight of the feature reconstruction loss in the generator loss was 0.00001.

5.1.2 Quantitative results with Keras

The first set of experiments were developed with the Keras software framework for deep learning. In them, we implemented the baselines presented in Section 4.1 and added the perceptual loss proposed in Section 4.2.

The plots Figure 8 show the colour histogram of the real coloured images, *pix2pix* baseline [4], Górriz et al. [7], and our model with different backbones. Quantitative metrics are

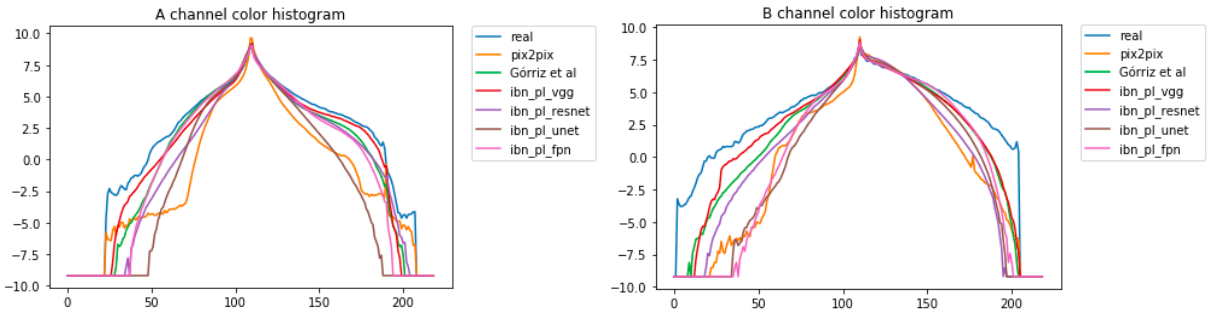


Figure 8: Comparison of logarithmic colour histograms for both *ab* channels for CIE lab colourspace

represented in Table 2. The results are coherent with the qualitative results in Figure 9. The model that includes the feature reconstruction loss with the VGG backbone has the most similar histogram to the real histogram, denoting more vivid colours in the image. PSNR is a measure that does not penalize desaturated results, so *pix2pix* performs better for this metric.

Table 2: Quantitative metrics for the models

Models	Backbone	JS divergence ↓		KL divergence ↓		PSNR ↑
		a	b	a	b	
pix2pix [4]	-	0.13	0.13	79.52	82.94	**26.70**
Górriz et al. [7]	-	**0.06**	0.06	59.51	22.28	25.14
Our model	VGG	0.09	**0.05**	**50.33**	**20.15**	25.13
	ResNet	0.12	0.13	94.82	109.70	25.23
	U-Net	0.23	0.19	282.60	205.80	25.19
	FPN	0.15	0.19	139	197.90	25.24

5.1.3 Quantitative Results with Pytorch

We replicated the results using the Pytorch software framework instead of Keras. For this part, we only have the quantitative results as we did not have enough time to study the qualitative results. Note that even if we have different results, our goal was to extract the same conclusions as we did with the Keras framework implementation.

The plots in Figure 10 represent the colour histogram of the real coloured images, the U-net baseline, pix2pix, Górriz et al and the best performing model from our Keras Experiments, which was adding feature reconstruction loss using the VGG architecture. As we have seen in the Keras implementation, the feature reconstruction loss aids the colourisation by making it more colourful, represented in the width of the logarithmic histograms. We were also able to compare the Unet baseline and corroborate our expectations for this baseline: it provides desaturated results, shown in figure 12.

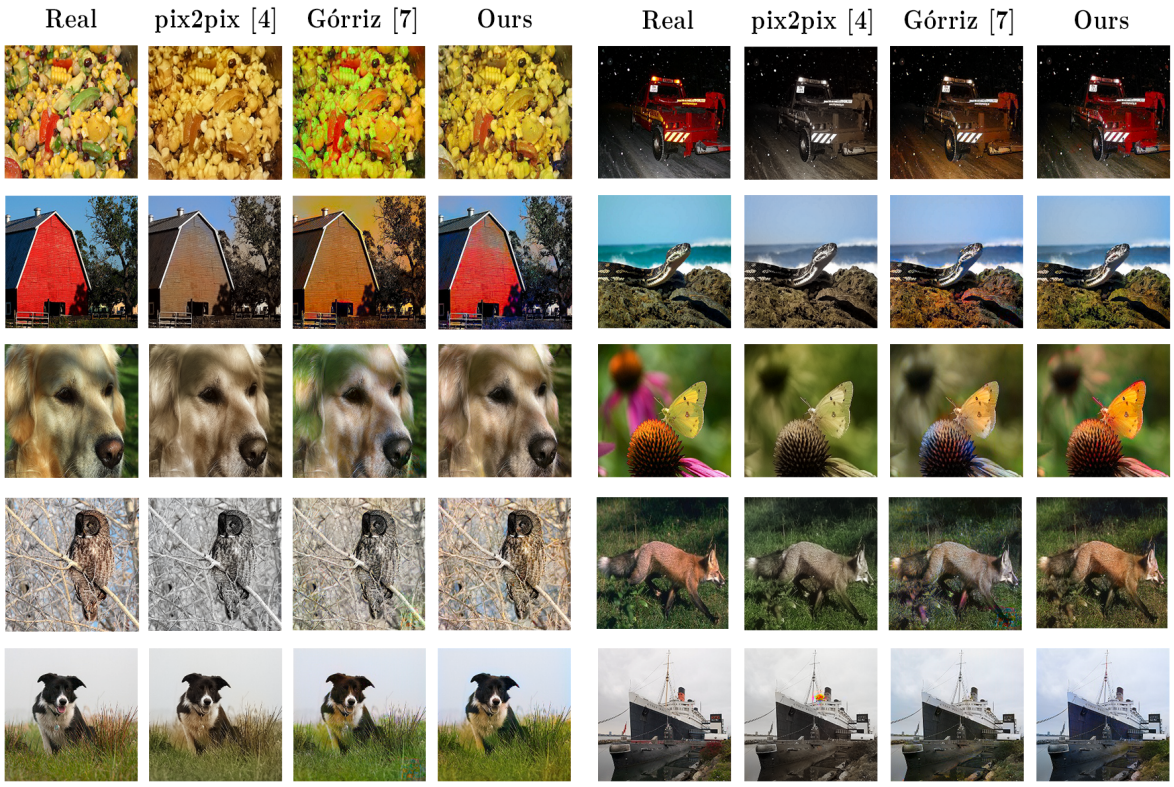


Figure 9: Example of coloured images using our model compared to the state-of-the-art approaches in [4] and [7]

Table 3 contains the evaluation metrics obtained with the PyTorch implementation. Although the values are different than in the Keras implementation, we can extract similar conclusions than in Pytorch:

Table 3: Quantitative metrics for the models

Models	Backbone	JS divergence ↓		KL divergence ↓		PSNR ↑	SSIM ↑
		a	b	a	b		
U-Net	-	0.33	0.51	372.75	123.49	24.32	1.13
pix2pix	-	**0.15**	0.28	48.18	250.41	**24.38**	**1.31**
Górriz et al.	-	0.16	0.16	11.36	83.12	23.05	1.06
Our model with feat. rec. loss	VGG	0.19	**0.13**	**10.24**	**22.28**	23.04	1.08

We plotted the loss curves for the different implementations to be able to compare the training behaviour.

In the basic U-Net implementation (Figure 12), we can see the clear tendency of the model to optimize the loss, decreasing it during training time. We can see from the results that even at the minimum point of the loss, the results in Figure **??** look desaturated, due to the nature of the regression (or colourisation) loss.

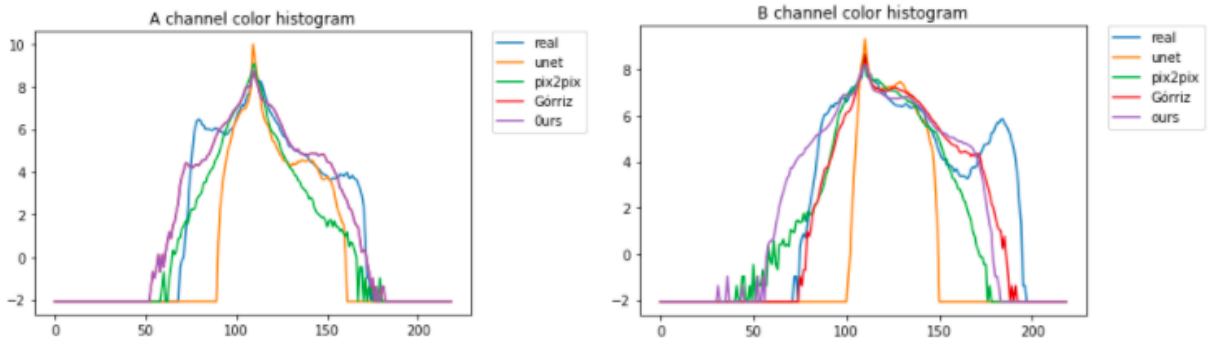


Figure 10: Logarithmic colour histograms for both a and b channels of CIE Lab colourspace

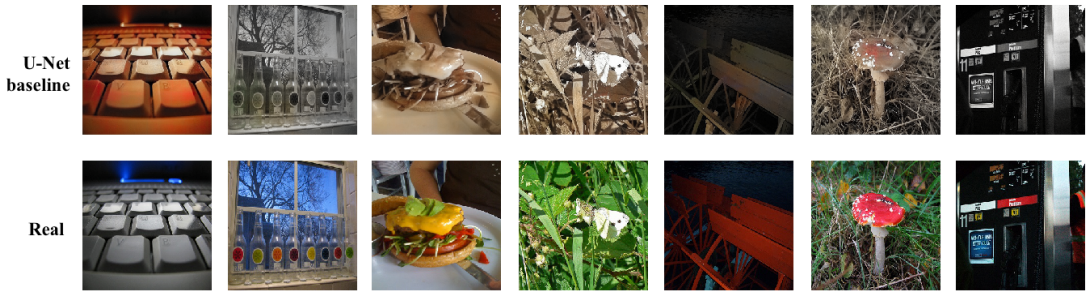


Figure 11: A few examples of coloured images using the basic U-Net baseline.

In the pix2pix implementation (Figure 13), we can see that a stabilization where the discriminator and generator losses are fighting each other, but at around iteration 120k we achieve what is known as mode collapse. The discriminator is able to distinguish every time the difference between the real and fake image, and the generator is not able to learn through the GAN loss. This mode collapse produces the famous desaturated results noted in [54] and [55] due to the reduction of the contribution of the adversarial loss.

In the Górriz baseline (Figure 14), we can see a clear avoidance of the previously mentioned mode collapse.

5.1.4 Qualitative Results

A perceptual realism study was performed, similarly to the one presented in Chroma-GAN [30]. Images were shown to non-expert participants, where some are coloured using ground-truth and others the results of a colourisation method. The colourisation methods included were: pix2pix [4] and BBC Górriz [7] implementation and ours. For each image shown, the participant indicates if the image has real or generated colours.

The qualitative study was run for 150 ground truth images and 150 images for each model. Each participant had 50 images to label, and the study was performed 35 times. Perceptual realism corresponds to the % of pictures noted as real from each model.

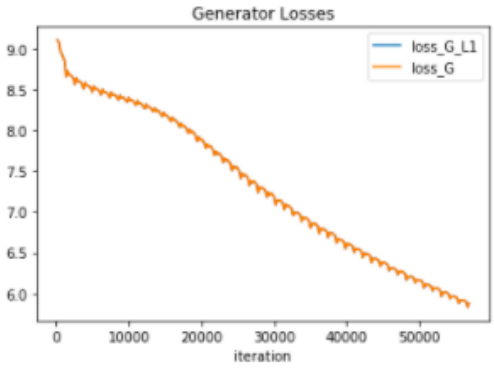


Figure 12: Loss for the U-Net baseline.

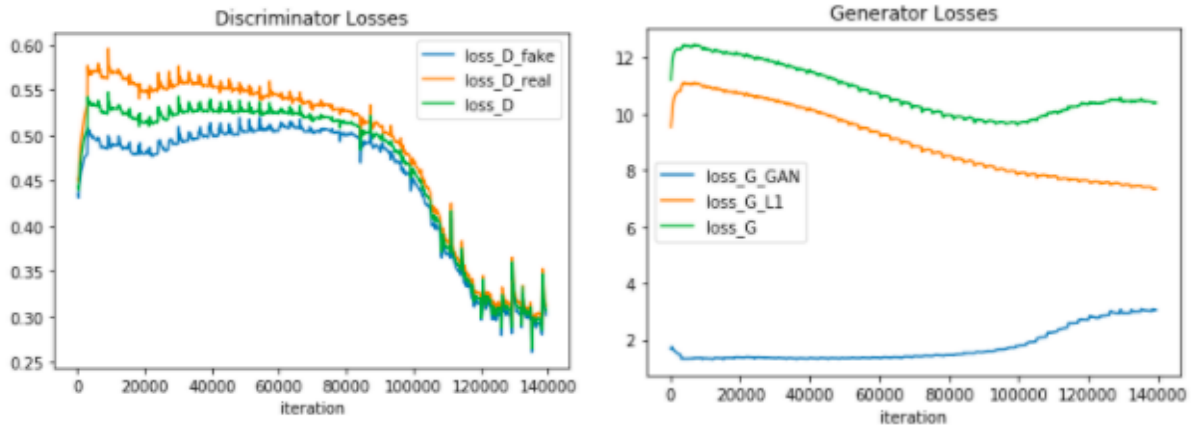


Figure 13: Losses for the Pix2pix baseline.

The results presented in Table 4 show how *pix2pix* obtained better perceptual realism. After a study of the reason behind it, we found that participants tend to classify desaturated images as real. However, even if out method produces more colourful results that are perceptually coherent, the participants were able to pick up on artifacts appearing in the images and were able to distinguish them as generated images.

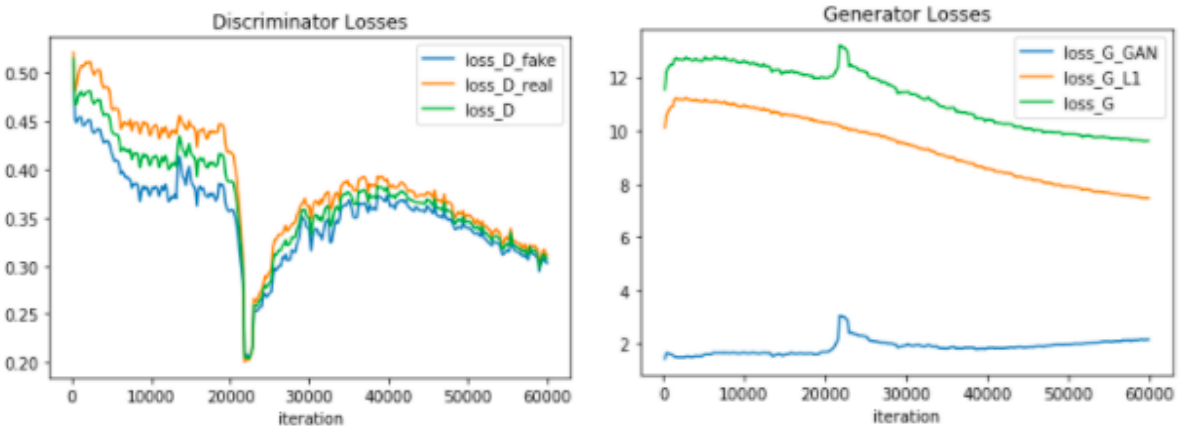


Figure 14: Losses for the BBC Górriz baseline.

Table 4: Qualitative metrics for the models

Model	**Naturalness**
Ground Truth	0.87
pix2pix [4]	0.53
Górriz et al. [7]	0.26
Ours with VGG	0.38

5.2 Image colourisation with segmentation maps

5.2.1 Implementation details

We designed two solutions to exploit semantic segmentation maps for colourisation: one with a shared decoder an another one with separate decoder (see Section 4.2 for more details). However, we could only implement the solution with the shared decoder during the timeframe of this master thesis.

This solution adds channels to the baseline U-net [34], a pair of a and b channels for each considered semantic class.

Our training data came from the Pascal dataset [58], which contains 21 classes, including background. We use the standard partitions of 1,464 samples for training and 1,449 for test. All images were resized to 256 x 256 pixels and converted to the CIE Lab colour space. This configuration was trained for 48 epochs, during 10 hours.

We trained the network from scratch, as the pretrained networks for Imagenet are pretrained with RGB images and normalised between [0,1] and we are using a network that has as an input a luminance image that is normalised between [-1,1] we were afraid that the initialisation would not work. However, it would be another improvement that we could add to use a pretrained network and study the compatibility with our colourspace.

5.2.2 Quantitative Results

Table 5 shows the different quantitative metrics for the different experiments we did with feature reconstruction loss. Since we are using the segmentation Pascal Dataset, we are unable to compare the results with the previously defined models.

We provide a set of test images 15 to visualise the results and hypothesise about the lack of colour in the results.

Table 5: Results with a segmentation loss in a shared decoder

Models	PSNR ↑	SSIM ↑
U-Net with segmentation loss	24.20	1.11

Figures 16 and 17 show the loss curves of these two configurations.

For our architecture without an adversarial loss, we can clearly see that the model quickly learns a distribution but does not converge to a minimum. This could be because the network is trying to predict the a and b channels for every class, augmenting the output data complexity. It is possible then, that the best possible outcome for the network is to simply output the grayscale images with almost none activation for the chromatic channels. For future results we suggest only computing the loss over the present classes in the ground truth and ignore the rest.

For the experiment where we implement our segmentation network and add it to the adversarial framework, the model collapses because the losses from the discriminator are

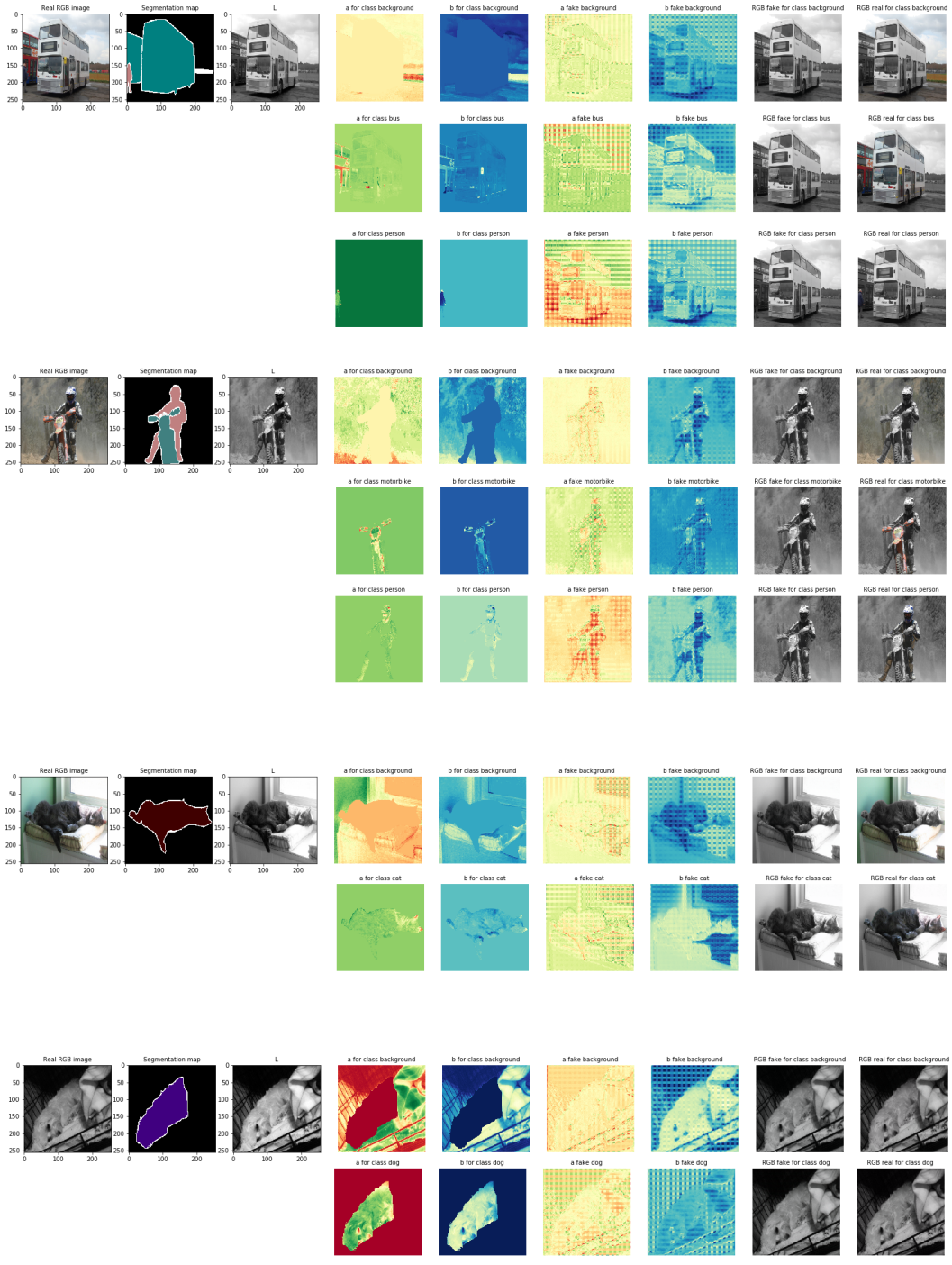


Figure 15: Results for the Image colourisation with segmentation maps without generative loss.

small, meaning that it is able to identify almost always whether the image is real or generated, and the GAN for the generator is high, meaning that it is not able to trick the discriminator. This may be due to the previously mentioned problem, where the loss is pushing the model to learn the average value, which might be easy to identify by the discriminator. Similarly to the previous experiment, we can observe that the L1 is not decreasing.

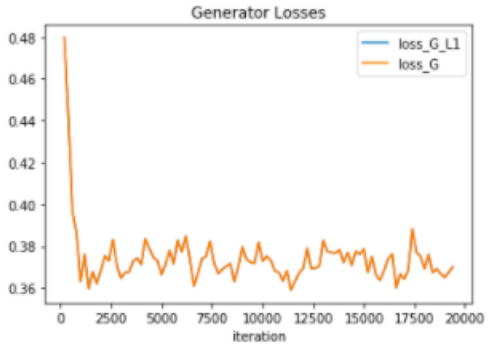


Figure 16: Loss for the U-Net baseline plus the segmentation information.

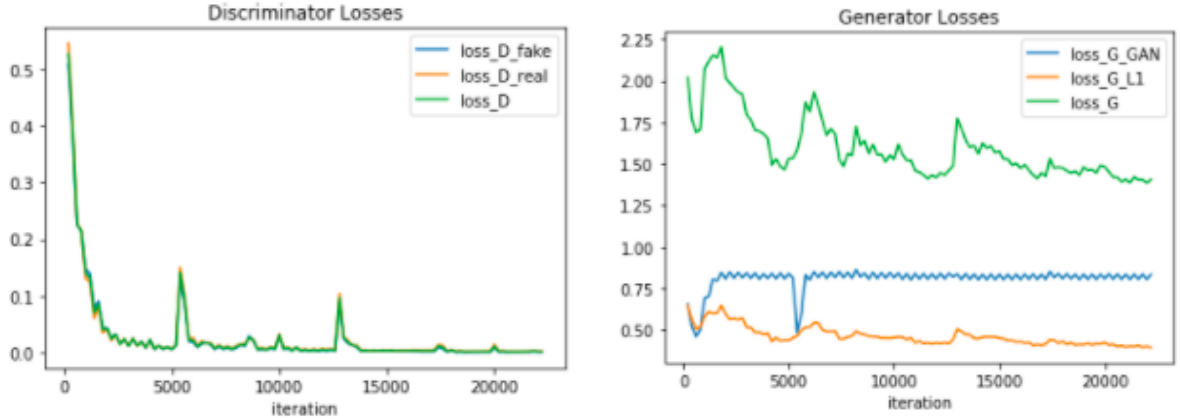


Figure 17: Losses for the segmentation + gan model.

5.3 Summary

This section sums up all the findings from our implementations and the results achieved.

Colourisation loss only: The histogram from both chroma channels is very narrow compared to the real channel histograms. This is due that the loss does not encourage the results to be colourful and it predicts either the mean values or the values that are most prevalent in the training data (mostly blues and greens), since there are a lot of pixel values from the sky and greenery in the Imagenet dataset.

Colourisation and adversarial losses: The pix2pix baseline improves the results by adding the generative adversarial loss. We can see that the histograms are wider than before and from the results we can perceptually notice the difference in the appearance of some vibrant colours.

BBC Gorriz baseline: We observed another boost in the width of the histograms, plus the added benefit of stabilization of the networks during training.

Colourisation, adversarial and feature reconstruction losses: The addition of the feature reconstruction loss further boosts the colourisation while keeping the images perceptually salient.

Colourisation and segmentation losses: We did not succeed in training a neural model capable of exploiting the segmentation maps. We plan to keep working on this research direction after the completion of this master thesis.

6 Budget

This section estimates the necessary budget to develop the work presented in this report. It is important to remark that this project is software-based, so there is not a final physical product created. Moreover, there is no aim in selling the final outcome of this thesis, so we do not include any analysis in this matters.

One important part of the budget comes from the personnel costs. The author of the thesis is counted as a junior engineer working as a full-time worker for the first part of the master thesis, and half-time for the second part of the thesis, which corresponds to an extension agreed between BBC and UPC beyond the original contract. Weekly meetings were held with both BBC and UPC supervisors, that will be counted as managers. In addition, an engineer at BBC also supported the work and participated in the weekly meetings. In Table 6 we can see the total personal costs. It is relevant to note that the personal costs regarding of the junior engineer were funded by BBC following the scholarship regulations defined at UPC.

	Number	Wage	Hours/Week	Total Weeks	Total
Junior Engineer	1	5€/hours	40/20	35/20	9000€
Engineer	1	20€/hour	1	55	1100 €
Manager	2	40€/hour	2	55	8800 €
					18900€

Table 6: Total personal costs

The software has been developed in Python which is open-source. However, we have used Pycharm as an integrated development environment which requires a license. In order to develop this project we used the GPU cluster from the Image Processing Group at UPC. To quantify how much it costs to use this service by comparing with how much it would cost the usage of such service from Google Cloud services. We have been running jobs in the cluster for approximately 10 months, on average we have been running a job a day for 12h, each of them with a GPU with 16G of RAM. The most similar resource that Google provides is the Nvidia T4, which costs $178.85 a month. The total cost, estimating that we had one job continuously running during this 10 months would be of $1788.5. The equivalent is 1487.18€ (with the conversion 1 USD = 0,83 EUR, at the date of 04/05/2021). In Table 7 we can see the total software costs.

Software	Number	Price	Total Usage	Total
IDE License	1	199€/year	1 year	199 €
GPU Cluster	1	178.85$/month	10 months	1478.18 €
				1677.18 €

Table 7: Software licences and GPU costs

Adding the different costs, the total budget for the project results in 20,577.18 €.

7 Conclusions and future work

In this master thesis we have proposed an algorithm for automatic colourisation of grayscale images with multiple approaches. Based on the presented state of the art, we proposed different improving methodologies to increment the liveliness of the results.

We validated that the feature reconstruction loss improves the results of the existing baseline at BBC, as it promotes a colour distribution more similar to the ground truth than any of the previous baselines. We experimented both on qualitative and quantitative results, but it was proven that there is still room for improvement to do the analysis for the colourisation task.

We lacked time to study in detail the potential of segmentation maps to improve colourisation. The model we implemented was not properly train despite a sounding theoretical approach. Next steps beyond this master thesis will focus in visualizing the activation maps and gradients during training to understand which part of the pipeline is not working as expected.

As we have seen in the results, quantitative metrics are not able to correctly determine the performance of the network that is doing colourisation. We believe this is a very interesting field that needs further development and should be studied.

Beyond colourisation, the task of adapting old audiovisual content to modern displays presents several challenging tasks, such as increasing the spatial definition of the images or exploiting the temporal redundancies when enhancing videos. We predict that this field will continue growing, as there is still much to be explored, and the task of colourisation still offers opportunities for innovation.

References

[1] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Machine learning basics. *Deep learning*, 1:98–164, 2016.

[2] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *CoRR*, abs/1411.1784, 2014.

[3] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[4] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017.

[5] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

[6] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation, 2018.

[7] Marc Gorriz Blanch, Marta Mrak, Alan F Smeaton, and Noel E O'Connor. End-to-end conditional gan-based architectures for image colourisation. In *2019 IEEE 21st International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6. IEEE, 2019.

[8] Yanghua Jin, Jiakai Zhang, Minjun Li, Yingtao Tian, Huachun Zhu, and Zhihao Fang. Towards the automatic anime characters creation with generative adversarial networks, 2017.

[9] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks, 2020.

[10] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks, 2017.

[11] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network, 2017.

[12] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting, 2016.

[13] D. Lischinski A. Levin and Y. Weiss. Colorization using optimization. *In ACM transactions on graphics (tog)*, 23:689–694, 2004.

[14] M. Ashikhmin T. Welsh and K. Mueller. Transferring color to greyscale images. *In ACM transactions on graphics (tog)*, 21:227–280, 2002.

[15] P. Isola X. Geng A. S. Lin T. Yu R. Zhang, J.-Y. Zhu and A. A. Efros. Real-time user-guided image colorization with learned deep priors. *ACM Transactions on Graphics*, 36, 05 2017.

[16] Q. Yang Z. Cheng and B. Sheng. Deep colorization. *In Proceedings of the IEEE International Conference on Computer Vision*, page 415–423, 2015.

[17] A. Deshpande, J. Rock, and D. Forsyth. Learning large-scale automatic image colorization. pages 567–575, 2015.

[18] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. *ACM Trans. Graph.*, 35(4), 2016.

[19] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016.

[20] M. Maire G. Larsson and G. Shakhnarovich. Learning representations for automatic colorization. 9908:577–593, 10 2016.

[21] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014.

[22] C. Chan, S. Ginosar, T. Zhou, and A. Efros. Everybody dance now. *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5932–5941, 2019.

[23] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. 12 2018.

[24] X. He Z. Zhang K. Lin, D. Li and M.-T. Sun. Adversarial ranking for language generation. 12 2017.

[25] A. Martinez A. Sanfeliu A. Pumarola, A. Agudo and F. Moreno-Noguer. Ganimation: Anatomically-aware facial animation from a single image. *ECCV*, 2018.

[26] J. Zhu, T. Park, P. Isola, and A. A. Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. *ICCV*, 2017.

[27] Kamyar Nazeri, Eric Ng, and Mehran Ebrahimi. Image colorization using generative adversarial networks. pages 85–94, 2018.

[28] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. *European Conference on Computer Vision*, 2016.

[29] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. *In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[30] Patricia Vitoria, Lara Raad, and Coloma Ballester. Chromagan: An adversarial approach for picture colorization. In *WACV*, 2019.

[31] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[32] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014.

[33] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *LNCS*, 9351:234–241, 2015.

[35] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*, 2014.

[36] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017.

[37] Keras. Keras Applications. *£https://keras.io/api/applications/£*.

[38] P. Yakubovskiy. Segmentation models. *£https://github.com/qubvel/segmentation_models£*.

[39] S. Jheng-Wei, C. Hung-Kuo, and H. Jia-Bin. Instance-aware image colorization. *2020 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[40] V. Sushko, E. Schonfeld, D. Zhang, J. Gall, B. Schiele, and A. Khoreva. You only need adversarial supervision for semantic image synthesis. In *ICLR*, 2021.

[41] A. Levin, D. Lischinski, and Y. Weiss. Colorization using optimization. *ACM Trans. Graph.*, 23(3):689–694, 2004.

[42] Yi-Chin Huang, Yi-Shin Tung, Jun-Cheng Chen, Sung-Wen Wang, and Ja-Ling Wu. An adaptive edge detection based colorization algorithm and its applications. In *ACM Multimedia*, pages 351–354. ACM, 2005.

[43] Qing Luan, Fang Wen, Daniel Cohen-Or, Lin Liang, Ying-Qing Xu, and Heung-Yeung Shum. Natural image colorization. pages 309–320, 01 2007.

[44] Tomihisa Welsh, Michael Ashikhmin, and Klaus Mueller. Transferring color to greyscale images. *ACM Trans. Graph.*, 21(3):277–280, July 2002.

[45] Raj Kumar Gupta, A. Chia, D. Rajan, E. S. Ng, and Z. Huang. Image colorization using similar images. In *ACM Multimedia*, 2012.

[46] Zezhou Cheng, Q. Yang, and Bin Sheng. Deep colorization. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 415–423, 2015.

[47] Xiaopei Liu, Liang Wan, Yingge Qu, Tien-Tsin Wong, Stephen Lin, Chi-Sing Leung, and Pheng-Ann Heng. Intrinsic colorization. *ACM Transactions on Graphics (SIGGRAPH Asia 2008 issue)*, 27(5):152:1–152:9, December 2008.

[48] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015.

[49] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks, 2016.

[50] Dmitry Ulyanov, Andrea Vedaldi, and Victor Lempitsky. Instance normalization: The missing ingredient for fast stylization, 2017.

[51] Xingang Pan, Ping Luo, Jianping Shi, and Xiaoou Tang. Two at once: Enhancing learning and generalization capacities via ibn-net, 07 2018.

[52] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation, 2015.

[53] Marc Gorriz Blanch, Issa Khalifeh, Alan Smeaton, Noel O'Connor, and Marta Mrak. Attention-based stylisation for exemplar image colourisation, 2021.

[54] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. High-resolution image synthesis and semantic manipulation with conditional gans, 2018.

[55] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans, 2016.

[56] Takeru Miyato, Toshiki Kataoka, Masanori Koyama, and Yuichi Yoshida. Spectral normalization for generative adversarial networks, 2018.

[57] C. Connolly and T. Fleiss. A study of efficiency and accuracy in the transformation from rgb to cielab color space. *IEEE Transactions on Image Processing*, 6(7):1046–1048, 1997.

[58] Mark Everingham, S. M. Ali Eslami, Luc Van Gool, Christopher K. I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes challenge: A retrospective. *International Journal of Computer Vision*, 111(1):98–136, January 2015.

Appendices

We have appended the Extended Abstract accepted for the Woman in Computer Vision workshop at CVPR, which will have a poster session on June 19th 2021.

GAN-based Image Colourisation with Feature Reconstruction Loss

Laia Tarrés[1] Marc Gorriz[3] Xavier Giro-i-Nieto[1,2] Marta Mrak[3]
[1]*Universitat Politècnica de Catalunya*
[2]*Institut de Robòtica i Informàtica Industrial, CSIC-UPC*
[3]*BBC Research & Development*
laia.tarres@upc.edu

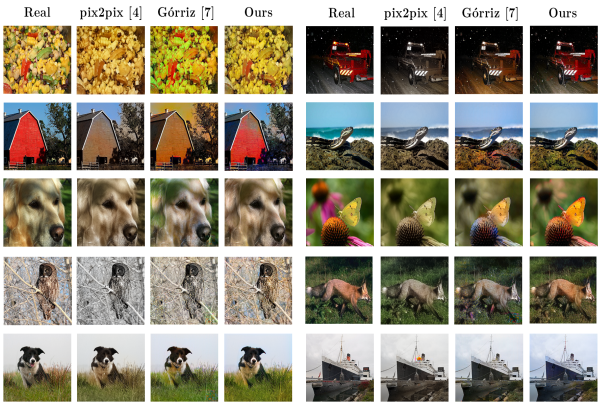


Figure 1: Example of coloured images using our GAN model compared to the state-of-the-art approaches in *Pix2Pix* [5] and Górriz et al[3]

.

1. Introduction

Image colourisation is the task of adding plausible colour to grayscale images. This transformation requires obtaining a three dimensional colour-valued mapping from a real-valued grayscale image, which leads to an undetermined problem because the gray-scale semantics and texture provide cues for multiple possible colour mappings. The goal of image colourisation in not to recover the ground truth colour in a manner that it is perceived as natural by a human observer.

Our work takes as a baseline a scheme based on an end-to-end trainable convolutional neural network (CNN) trained with a smooth L1 loss to predict the ab channels of a colour image given the L channel. We introduce an extra perceptual reconstruction loss during training to improve the capabilities of a adversarial adversarial model, that we adopt as a baseline. Figure 1 presents some examples of the results achieved by our method.

2. Related Work

Image colourization networks are typically trained in a self-supervised set up in which colour images are converted to grayscale [14]. This allows quickly gathering data suitable for training deep neural networks. A first approach of image colourization with deep learning was proposed by Cheng et al. [13] by formulating a least square minimization problem solved with deep neural networks.

The capabilities of Generative Adversarial Networks (GANs) [2] for producing realistic samples was firstly applied for image colourization in *Pix2Pix* Isola et al. [5]. Some training improvements to his set up were proposed by multiple authors [9, 12, 3]. In particular, Gorriz et al. [3] increased the colour saturation obtained by an off-the-shelf *pix2pix* model by adding batch and instance normalization to the training, as well as multiple discriminators.

3. Methodology

In our work we add the feature (or perceptual) reconstruction loss and include it in the objective function used in *Pix2Pix* [5]. This loss was proposed by Johnson et al. [6] for image translation tasks, defined as the squared and normalized Euclidean distance between activations produced in the early layers of the network for the output image and the target image. Mahendran et al. [8] showed that using a feature reconstruction loss for training image transformation networks encourages the output image to be perceptually similar to the target image, but does not force them to match exactly.

In our implementation, rather than using only per-pixel loss functions depending only on low-level pixel information, we train our networks using added perceptual loss functions that depend on high-level features from a pretrained loss network. During training, perceptual losses measure image similarities more robustly than per-pixel losses. This way, when we feed an image to a pretrained network for image classification, the model has already learned the perceptual and semantic information that we

would like to measure. So comparing the network's activations from the ground truth and the generated image provides perceptual information.

The computation of the feature reconstruction loss corresponds to the squared and normalized Euclidean distance between the activations of a selected layer produced by the real image and generated image, when forwarded through the perceptual loss network.

For our experiments, we have tried different types of pretrained neural networks to extract the features to be compared. The first group of experiments used either a VGG16 network [11] or ResNet50 network [4] classification networks, were both were pretrained for image classification on the ImageNet dataset [1]. The second group of experiments is based on either U-Net [10] or FPN [7] segmentation networks, both composed by classification networks pretrained on ImageNet and COCO dataset, respectively.

4. Experimental Results

4.1. Implementation Details

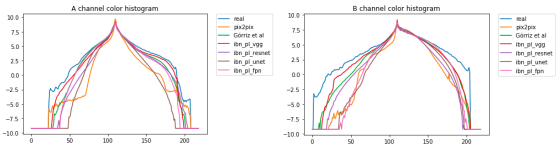
We define as baselines a U-Net [10] architecture as generator, and PatchGAN as the discriminator, the same approach as is *pix2pix* [5]. Training data are extracted from the ImageNet dataset. We select 50,000 RGB images that represent 50 images per class for training, and 10,000 test images selected as 10 images per each class. All classes are converted to CIE Lab colour space. The best results were obtained with the feature reconstruction loss on *block3 conv3* layer from the VGG16 network, with a loss weight of 0.00001. This configuration was trained for 23 epochs, during 36 hours.

4.2. Quantitative results

Evaluating the quality of a colourised image in a quantitative way is a challenging task, and still remains to be solved. Therefore, quantitative measures reflecting how close the outputs are to the ground truth data may not characterise the human perception of the problem. Nevertheless, we have used quantitative measures in order to quantitatively compare the results of the proposed methods to others in the literature.

The plots in Figure 2 represent the colour histogram of the real coloured images, *pix2pix* baseline [5], Górriz et al. [3] and our model with different backbones. As quantitative metrics, we have chosen the Kullback Liebler Divergence. Furthermore, as state-of-the-art methods on colourisation, we have also included peak signal to noise ratio (PSNR). They are represented in Table 1.

Our model with the VGG16 backbone has the most similar histogram to the real histogram, denoting more vivid colours in the image. PSNR is a measure that does not penalize desaturated results, so pix2pix performs better.



(a) A channel (b) B channel

Figure 2: Comparison of logarithmic colour histograms for both AB channels for CIE lab colourspace. The wider the histograms, the more colours they are representing, and more vivid the resulting images are.

Table 1: Quantitative metrics for the models

Models	Backbone	JS divergence		PSNR
		a	b	
pix2pix	-	0.13	0.13	**26.70**
Górriz et al.	-	**0.06**	0.06	25.14
Our model	VGG16	0.009	**0.05**	25.13
	ResNet	0.12	0.13	25.23
	Unet	0.23	0.19	25.19
	FPN	0.15	0.19	25.24

Table 2: Qualitative metrics for the models

Model	Naturalness
Ground Truth	0.87
pix2pix [5]	0.53
Górriz et al. [3]	0.26
Ours with VGG16	0.38

4.3. Qualitative results

A perceptual realism study was performed, similarly to the one presented in ChromaGAN [12]. Images were shown to non-expert participants, where some are ground-truth colourisation and others the results of a colourisation method. The colourisation methods included were: our method with VGG16 backbone, pix2pix [5] and Górriz [3] implementation. For each image shown, the participant indicates if the image has real or generated colours.

The qualitative study was run for 150 ground truth images and 150 images for each model. Each participant had 50 images to label, and the study was performed 35 times. Perceptual realism corresponds to the % of pictures noted as real from each model.

The results presented in Table 2 show how *pix2pix* achieves better perceptual realism, as participants tend to classify desaturated images as real. Our model produces more colourful results that are perceptually coherent, so it is suitable when aiming at equally vibrant results.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 2

[2] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 1

[3] M. Górriz, M. Mrak, A. Smeaton, and N. O'Connor. End-to-end conditional gan-based architectures for image colourisation. 2019. 1, 2

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 2

[5] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 1, 2

[6] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. *European Conference on Computer Vision*, 2016. 1

[7] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 936–944, 2017. 2

[8] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them. *In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2015. 1

[9] Kamyar Nazeri, Eric Ng, and Mehran Ebrahimi. Image colorization using generative adversarial networks. pages 85–94, 2018. 1

[10] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. *LNCS*, 9351:234–241, 2015. 2

[11] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv 1409.1556*, 09 2014. 2

[12] Patricia Vitoria, Lara Raad, and Coloma Ballester. Chromagan: An adversarial approach for picture colorization. In *WACV*, 2019. 1, 2

[13] Q. Yang Z. Cheng and B. Sheng. Deep colorization. *In Proceedings of the IEEE International Conference on Computer Vision*, page 415–423, 2015. 1

[14] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *ECCV*, 2016. 1