Image-to-Image translation from a sketch to a potrait using Cycle-consistent adversarial networks

Dhruvam Panchal Sai Deepak Siddharth Deshpande Hitarth Bharad

Abstract

This report presents an application of image-to-image translation with Cycle-consistent Adversarial Network (CycleGAN). Image-to-Image translation is learning of the mapping between two specific images from source domain t o the target domain and has many applications including the image in-painting, image colouring and style transfer. In the recent times, GANs have made much progress in cross-domain image-to-image translation. This paper demonstrates an application of converting sketch to a real life portrait using CycleGAN.

1. Introduction

What do humans' see when they look at a portrait sketch? Our mind tries to generate a coloured image or a real life presentation of the individual in focus. From the sketch, we can imagine everything from the texture of face to the skin tone to the contours despite never having seen a side by side comparison of the images from both domains. We instead have the knowledge of the features of the real life images and sketch images. We can reason the differences between the two domains and thereby imagine the portrait in one domain given the other domain.

We aim to convert the sketch portrait to a real life portrait of a person using a machine learning model that can accurately learn the differences between both the domains. This problem accurately signifies the need for an image-to-image translation network. Recently there has been a lot of research in the area of image-to-image translation which have produced significant results.

Recent methods in image generation include methods such as Autoencoders[1], GANs[2], Conditional GANs[7]. However, almost all the researched models require a paired dataset for it to learn the difference between the two domains. We aim to use a model which can train itself of an unpaired image dataset by learning the difference between the two domains. Autoencoders and CycleGAN[11] are two methods that use unpaired image-to-image dataset to convert the image from one domain to the other domain. We use CycleGAN as the results of it are much better than any other comparative models.

2. Related work

2.1. Generative Adversarial Networks

Generative Adversarial Network or GANs have achieved impressive results in image generation and image editing. The key reason behind the success of GANs is it's use of adversarial loss which makes the generated images indistinguishable from the real images. This loss is particularly substantial for image generation tasks. CycleGANs particularly use this loss for the task of image generation.

GANs contain a generator and a discriminator that work against each other to produce better results. Generator is a convolutional neural network that takes a random noise and outputs the desired image. Discriminator is also a convolutional neural network that is used for classification between fake and real dataset.

GANs use adversarial loss for training the model. The loss function is shown below:

$$\mathcal{L}_{\mathrm{GAN}}\left(G, D_Y, X, Y\right) = \mathbb{E}_{y \sim p_{\text{tatu}}\left(y\right)}\left[\log D_Y(y)\right] + \mathbb{E}_{x \sim p_{\text{data}}\left(x\right)}\left[\log\left(1 - D_Y(G(x))\right)\right]$$

Generator(G) aims to minimize this loss and discriminator(D) tries to maximize this loss.

$$G = \arg\min_G \max_D \mathcal{L}\left(G, D\right)$$

2.2. Context Encoders

Developed in April 2016, context encoders[8] provided one of the most impressive results of that time. The aim of the research paper was to input an image with a missing region and in-painting it in the image. It used a convolutional neural network or CNN to regress the missing values in the given input image. Context encoders share a similar structure to that of autoencoders as they use the same encoder-decoder architecture.

Researchers of the context encoders believe that context encoders need to understand the context of the image as well

as the plausible hypothesis of the content in the missing region of the image. So, context encoders use an encoder-decoder pipeline for the problem where the images are first encoded using a network derived from AlexNet architecture, in which given a 227 x 227 input image, where they use five convolutional layers followed by pooling layer to compute an abstract 6x6x256 dimensional feature representation.

Following the Encoder part of the pipeline, the feature representation is then passed through a channel-wise fully-connected layer. This part of the fully-connected layers is an essential part as its intention is to propagate the information within the activations of different feature maps. Thus, if the input layer has m feature maps, the output layer will also have m feature maps. Following this, there is decoder part of the context encoder where it uses the encoder generated features passed through the channel wise fully-connected layer to generate the pixels of the image. Decoder uses a series of five up-convolutional layers with learned filters, each with rectified linear unit (ReLU) activation function to generate the target image.

The network uses normalized masked L2 distance as the reconstruction loss function to reduce the difference between the generated and desired output image. And it also uses adversarial loss based on Generative Adversarial Networks to optimize the model.

2.3. Conditional GANs

Conditional GANs also represents a method of image-to-image translation. As the name suggests, the base behind the working of Conditional GANs is the derivation from generative adversarial networks or GANs. As GANs learn from normal data, conditional GANs learn from conditional data. It uses class labels along with the input data and inputs class labels along with noise as an input to generator to generate class specific results. Along with the generator, discriminator also receives class input to discriminate specifically for a particular class.

Conditional GANs use a U-net[9] based architecture with skip connections in the generator model for better results due to the involvement of inputs from layers earlier to the preceding layer in the current layer. In the discriminator part, the researchers have used discriminator used in PatchGAN which are highly efficient and produce a sharper result as opposed to just using L1 or L2 loss for discriminator.

2.4. Unsupervised Image-to-Image Translation Networks

Liu et al. proposed a method for image to image translation between two different data domains. It was built upon the frameworks of variational autoencoders with generative adversarial networks and a weight sharing strategy. A Variational Autoencoder[6] has an encoder and a generator. Encoder obtains a latent space probability distribution of useful attributes of an input image that can be used as input to generator to obtain the original image back. GAN is used to learn to generate an image of the output set given the latent space variables that were generated by the VAE of the input set. The weight sharing strategy ensures that the latent variables formed for the two datasets are the same and a common representation is obtained. This promotes a better mapping of the images from one set to another.

The model improved upon Cycle GAN, providing slightly better results. Cycle GAN achieved a pixel accuracy of 0.569 whereas the UNIT model achieved pixel accuracy of 0.600.

3. CycleGAN

Jun-Yan Zhu et al. proposed an improvement to already existing Generator Adversarial Network to improve the existing level of models available for image-to-image translation method. Researchers used a very basic concept of cyclic consistency, similar to the concept of back translation and reconciliation technique used in language domain by human translators for verifying and improving translations. According to the idea of cyclic consistency, if there exists two domains X and Y and function F:X $\rightarrow$ Y and G:Y $\rightarrow$ X, G(F(X)) should be equal to X.

CycleGAN consists of two generators and two discriminators in total. The generators work as per the functions F and G mentioned above, and discriminators in the network work towards classifying the images.

3.1. Architecture

The network uses two different architectures for generator and discriminator.

Generators comprise of 3 sections: encoder, transformer and decoder. Encoder uses a 3 layer architecture which down-samples the input image to capture the features of the image, transformers use residual blocks which extract the information from the features extracted by the encoder block and decoder in the end up-samples the features processed by the transformers block.

Discriminators uses a PatchGAN discriminator architecture where instead of outputting one single value that is used to classify the image, they use a 32 $\times$ 32 $\times$ 1 output which helps classify the image much more accurately than a normal classifier with single neuron output.

3.2. Loss Functions

The network uses two types of loss functions to train the network.

Adversarial Loss is applied to both the mapping functions F : X$\rightarrow$Y and G : Y$\rightarrow$X and its corresponding discriminator D_X and D_Y, expressed by the objective function:

$$\mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) = \mathbb{E}_{y \sim p_{\text{tatu}}(y)}[\log D_Y(y)] + \mathbb{E}_{x \sim p_{\text{data}}(x)}[\log(1 - D_Y(G(x))]$$

Cycle Consistency Loss is used to make the model cycle consistent. Adversarial training can map the GAN model to produce output, however, it can output any random permutation of image which might not be the result we desire. Hence, we use the following cycle-consistency loss to get the desired output from all the possible permutations.

$$\mathcal{L}_{\text{cyc}}(G, F) = \mathbb{E}_{x \sim p\,\text{dat}(x)}[\|F(G(x)) - x\|_1] + \mathbb{E}_{y \sim p_{\text{data}}(y)}[\|G(F(y)) - y\|_1]$$

3.3. Overall Objective

The overall objective of our network is to minimize the loss mentioned below:

$$\mathcal{L}(G, F, D_X, D_Y) = \mathcal{L}_{\text{GAN}}(G, D_Y, X, Y) + \mathcal{L}_{\text{GAN}}(F, D_X, Y, X) + \lambda \mathcal{L}_{\text{cyc}}(G, F)$$

And the particular objective of G and F is given by the function below.

$$G^*, F^* = \arg\min_{G,F} \max_{D_x, D_Y} \mathcal{L}(G, F, D_X, D_Y)$$

Generator attempts to minimize the loss function as much as possible to produce the desired output by learning the features of the output domain. At the same time, discriminator attempts to maximize the overall loss by learning the difference between the two domains to help generator train better.

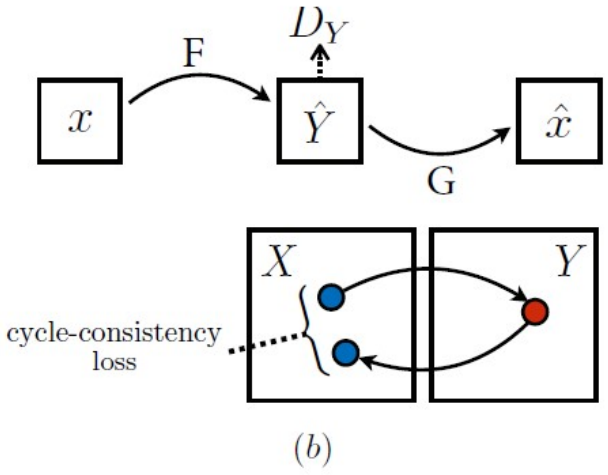


Figure 1: Cycle Consistency Loss

4. Dataset

Cycle GAN can be trained on paired as well as unpaired dataset. So 2 different models were trained and compared, one on a paired dataset and one on an unpaired dataset. Both the datasets were collected in a controlled setting with the same background and the position of face relatively same in all the photos.

For training on a paired dataset, CUHK student data set from CUHK Face Sketch Database[10] was used. The dataset has 188 photos and sketches with a blue background. The dataset comprises students from only CUHK, thus there is no variation in ethnicity of the subjects. The controlled setting makes the dataset ideal for paired image translation, as it reduces the features the variation in faces and the model can learn to generate output specific to this dataset faster. This is ideal only if the output is also required in the controlled setting as well. A paired dataset with controlled setting would ensure a better quality in the output images.

For training on an unpaired dataset, The Tufts Face Database[4] was used. The dataset itself is paired but for training on unpaired dataset, half of the photos and the other half of the sketches were used so that there were no pairings available. The model would learn to generate images from sketches without knowledge of what the output would look like. The dataset has a wider range of ethnicities, facial features, colour tones and hair styles. A bigger dataset would be ideal for such training although TUFT dataset is limited to 113 images.

5. Results

We used the results of the paper Convolutional Sketch Inversion[3] as a baseline as we could visually compare the results of our model to the images generated by the DNN based approach for qualitative analysis. For quantitative analysis we calculated the MAE score for the images generated from the sketches in CUHK as this was a paired dataset and we could use the original images as the desired output, heatmaps in figure 2 were also generated from the MAE scores.

Some outputs are shown in figure 3 and figure 4.

5.1. Limitations

One inherent limitation that is encountered in all GANs, when dealing with an unpaired dataset or with data that was not collected in a controlled setting is the quantity. The datasets used in this paper had a maximum of 188 images. With fewer images, a model will try to learn everything but will fail to differentiate between features if enough examples are not given of the different diverse inputs. When trained with the paired dataset, a range of skin tone is not established. The input sketch is always converted to a skin

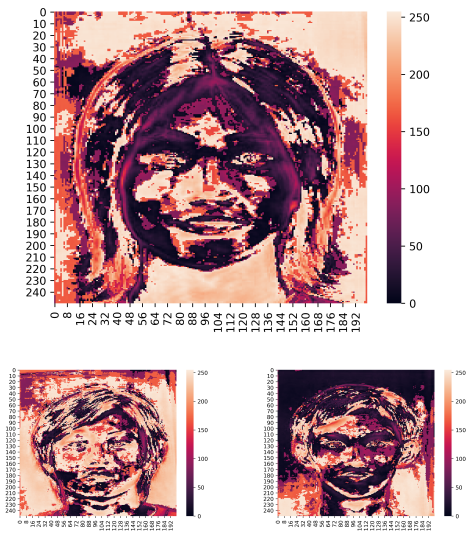


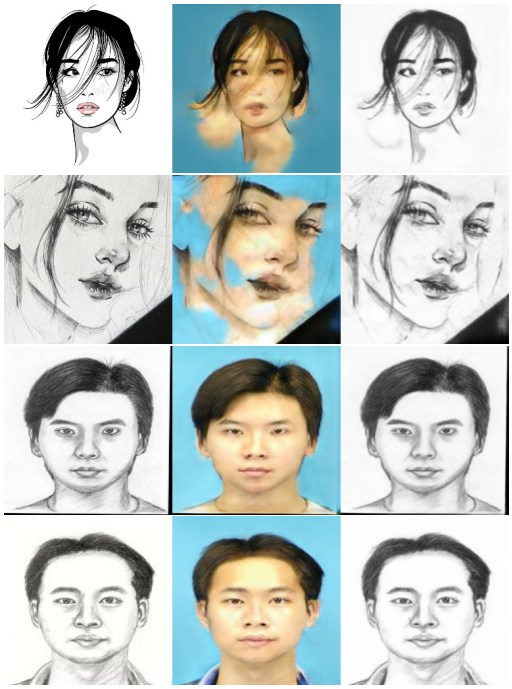
Figure 2: Heatmaps of MAE of colour images.

tone that is closer to the ski tones of the participants of the CUHK dataset, than the original photo corresponding to the input sketch. Lighter and darker skin tones can only be predicted if the input sketch is shaded to some extent to resemble the skin tone.

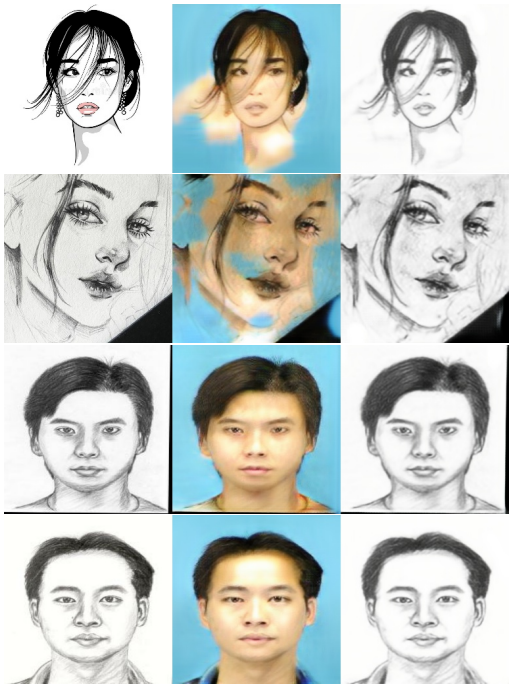
Cycle GANs also show poor performance when it comes to learning to change the structure of a sketch. Cycle GANs show substantially better performance for texture or tone change in image to image translation than structural changes. In the model trained with paired dataset, the change is minimal but it is apparent in the output of the model trained on unpaired dataset where the output matches the face structure in the sketch more than the corresponding input photo.

The heatmaps in figure 6 were generated from the pixel accuracy of the grayscale output of the model trained on the paired dataset and the corresponding images. Since the tones are almost similar at the same pixels in the image, converting image to grayscale significantly reduces the error that may occur due to colour difference and instead structural change will be more significantly visible. The darker regions in the figure represent difference between the generated output and the given output.

A common problem with Cycle GAN is the dissimilarity in the learning rate of generator and discriminator. Ideally generator and discriminator should run in junction so that generator loss is minimized while discriminator loss is maximized. When training a model, the discriminator learns much faster than the generator. This hinders the growth of the generator so that it cannot learn fast enough to match the accuracy of the discriminator and its losses keep increasing as shown in figure 7.



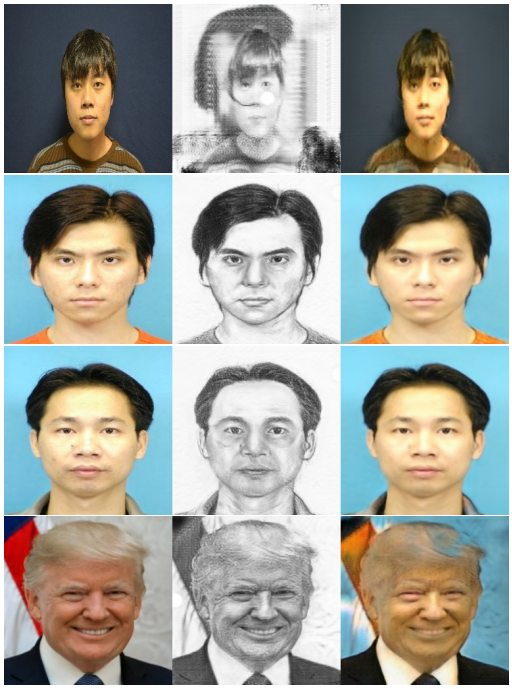
(a) Sketch to face to sketch results of model trained on unpaired data.



(b) Sketch to face to sketch results of model trained on paired data.

Figure 3: Sketch to face to sketch results.

Cycle GAN also may not work exactly as it is intended to work in an unsupervised setting. The generator will prioritise learning to reduce overall loss even if it is at the cost of



(a) Face to sketch to face results of model trained on unpaired data.

(b) Face to sketch to face results of model trained on paired data.

Figure 4: Face to sketch to face results.

(a) Model trained on paired data.

(b) Model trained on unpaired data.

Figure 5: Examples of difference in skin tone. Ignoring the colour leak, a clear difference is visible in the skin tone generated by the model trained on CUHK dataset and the skin tone generated by model trained on Tufts and CUHK dataset.

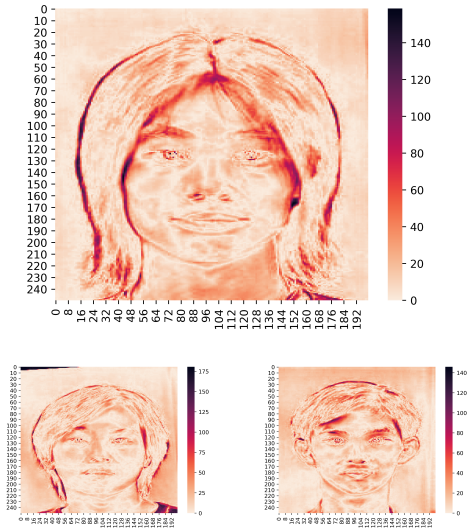
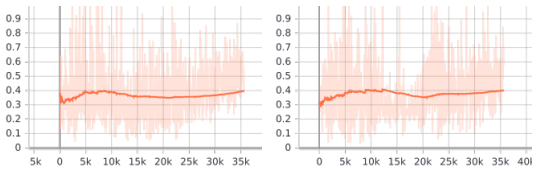


Figure 6: Heatmaps of MAE of grayscale images generated by model trained on paired dataset and the corresponding output in CUHK face sketch dataset.

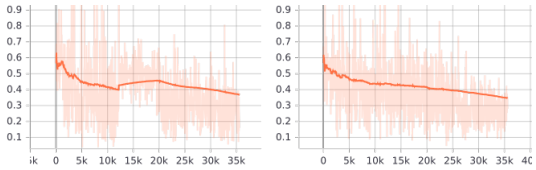
suffering from a significantly greater loss from some one part of the output. This results in what is known as the droplet artifact. The droplet artifact is a result of the generator intentionally sneaking signal strength information past instance normalization: by creating a strong, localized spike so that the generator can effectively scale the signal as it likes elsewhere[5]. This makes the output closer to the desired output on whole but leaves in some splotches (See figure 8).

6. Observations

The model after training could produce impressive results. Upon training the model twice, once with paired



(a) A2B Generator Loss (b) B2A Generator Loss



(c) A Discriminator Loss (d) B Discriminator Loss

Figure 7: Loss graphs of generator and discriminator of a Cycle GAN training between image sets A and B

Figure 8: Examples of droplet artifact.

dataset from CUHK and once with unpaired dataset from CUHK and TUFTS database, a few interesting observations were made.

1. With unpaired dataset, the model was able to capture the relevance of skin tone with sketch much better as the paired dataset only contained white skin toned faces.

2. At the end of the training, it was observed that even though the model can train itself on the unpaired dataset, it still requires structural similarity. In the case of tufts database, it was observed that the position of faces in the image space varied between different images, and this created confusion between the images.

3. When trained on paired image dataset consisting of Chinese volunteers, the model could not produce good results for portraits of people from other ethnic groups due to differences like skin tone and texture. However, this particular difference was not observed in terms of hair. We assume that hair has same texture and overall look across all groups and hence, the model does not have difficulty when rendering the hair of the subject.

4. It was also noticed that since the images where black and white, even after training on a diverse dataset with the help of TUFTS dataset, the model could not appropriately determine the right skin colour of the subject based on the sketch.

5. Intermediate results during training and final results showed that since the CUHK dataset and TUFTS dataset had light blue and dark blue backgrounds respectively, it created confusions in the model too.

Figure 9: Images from TUFT dataset.

7. Possible Changes for Better Results

There are two changes which we think could possibly give better results:

1. The paper mentions the use of identity loss for one of its applications to preserve the colour of the input images. We believe that training with identity loss can help bring out better output.

2. TUFTs database contains non-uniform dataset, the images can be edited or cropped to create uniformity between the positions of face in the image and help get better results.

References

[1] Dor Bank, Noam Koenigstein, and Raja Giryes. "Autoencoders". In: (2020). arXiv: 2003.05991 [cs.LG].

[2] Ian J. Goodfellow et al. "Generative Adversarial Networks". In: *arXiv e-prints*, arXiv:1406.2661 (June 2014), arXiv:1406.2661. arXiv: 1406.2661 [stat.ML].

[3] Yağmur Güçlütürk et al. "Convolutional Sketch Inversion". In: *arXiv e-prints*, arXiv:1606.03073 (June 2016), arXiv:1606.03073. arXiv: 1606.03073 [cs.CV].

[4] Panetta K. et al. *A comprehensive database for benchmarking imaging systems*. http://tdface.ece.tufts.edu/. 2020.

[5] Tero Karras et al. “Analyzing and Improving the Image Quality of StyleGAN”. In: *arXiv e-prints*, arXiv:1912.04958 (Dec. 2019), arXiv:1912.04958. arXiv: 1912.04958 [cs.CV].

[6] Diederik P Kingma and Max Welling. “Auto-Encoding Variational Bayes”. In: (2014). arXiv: 1312.6114 [stat.ML].

[7] Mehdi Mirza and Simon Osindero. “Conditional Generative Adversarial Nets”. In: (2014). arXiv: 1411.1784 [cs.LG].

[8] Deepak Pathak et al. “Context Encoders: Feature Learning by Inpainting”. In: *arXiv e-prints*, arXiv:1604.07379 (Apr. 2016), arXiv:1604.07379. arXiv: 1604.07379 [cs.CV].

[9] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. “U-Net: Convolutional Networks for Biomedical Image Segmentation”. In: *arXiv e-prints*, arXiv:1505.04597 (May 2015), arXiv:1505.04597. arXiv: 1505.04597 [cs.CV].

[10] X. Wang and X. Tang. *Face Photo-Sketch Synthesis and Recognition*. http://mmlab.ie.cuhk.edu.hk/archive/facesketch.html. 2009.

[11] Jun-Yan Zhu et al. “Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks”. In: *arXiv e-prints*, arXiv:1703.10593 (Mar. 2017), arXiv:1703.10593. arXiv: 1703.10593 [cs.CV].