Case Study: Supervised Binary Domain Image-to-Image Translation with Pix2Pix Conditional GAN

Tan Yu Hoe
Diploma in Applied Artificial
Intelligence and Analytics
School of Computing
Singapore Polytechnic

This technical paper aims to conduct a case study on supervised Image-to-Image Translation on a paired/two-domain dataset using a conditional generative adversarial, formally called Pix2Pix. The network architecture is reproduced using the proposed model by Philip Isola, then implemented on a cityscape paired dataset. The network is trained with an objective to perform image to image translation from a semantic annotated urban street image to a realistic urban street image.

Keywords—Image-to-Image Translation, I2I, Supervised Learning, Two-Domain Image-to-Image Translation, Generative Adversarial Network, Conditional Generative Adversarial Network, Pix2Pix, Convolutional Neural Network, cityscape, U-Net, PatchGAN, Image Synthesis

I. INTRODUCTION

With the tremendous increase of research on GAN (Generative Adversarial Networks) in recent years, GAN has seen successes in numerous applications such as image synthesis, segmentation, style transfer and restoration. This is likely due to the immense potential of GAN, an unsupervised/semi-supervised learning method requiring little-to-no labelled data to show results. A particular subset of the GAN research field which piques my interest is Image-to-Image Translation. Hence the focal domain of this paper. Image-to-Image Translation, or I2I in short – the aim of transferring an image content from a source domain to a target domain while preserving the content representation. With the concept of transferring contents between two domains, I2I have been seen in many use-cases such as image colourization, wildlife habitat analysis, domain adaptation, and facial geometry reconstruction. [1] In this technical paper, my aim would be to take on a simple case study – to train a network architecture to perform I2I on a paired dataset, formally called Supervised I2I. The network will be following the Pix2Pix architecture, a conditional GAN, which follows the concept of mapping an image to another image.

Fig. 1. An example of a two domain supervised Image to Image Translation – Edge to Shoes

II. RELATED WORKS

Since the theme of this technical paper will be on Image-to-Image Translation, I focused on looking for related works that had the following properties:

1. Current state of Image-to-Image Translation
2. Supervised Image-to-Image Translation using the Pix2Pix architecture
3. The origination of a U-Net, which is used in the Pix2Pix architecture

For property 1, I found a paper published in 2021 on "Image-to-Image Translation: Methods and Applications" by Yingxue Pang. The paper provides the state of I2I developments over the past years to 2021, describing the existing techniques in the community. The paper mentioned that there are two main sets of tasks – two-domain I2I and multi-domain I2I. These tasks are then split into supervised, unsupervised, and semi-supervised. Focusing on supervised binary domain I2I, the paper describes training an architecture to learn a mapping function on a paired dataset, usually leveraging on deep convolutional neural networks. This brought I2I to using conditional GAN Pix2Pix architecture. Some common issues that arise from Pix2Pix are blurry results, unstable training, and failure-prone to large resolution images. These issues led to the invention of Pix2Pix variants such as Pix2PixHD, SelectionGAN, and SPADE. [1]

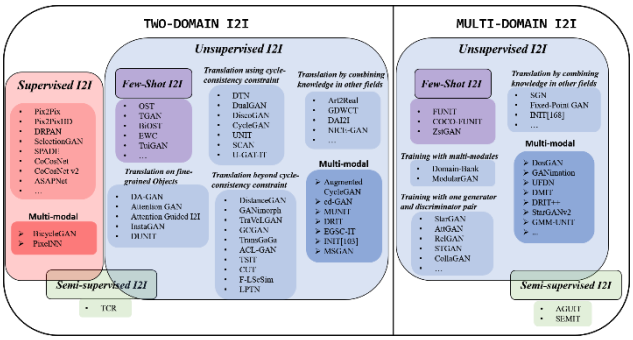

Fig. 2. An overview of image-to-image translation methods [1]

For property 2, I found a paper published in 2017 on "Image-to-Image Translation using Conditional Adversarial Networks" by Phillip Isola, the original implementation of the Pix2Pix architecture. The paper mainly describes the methodology and implementation of Pix2Pix as a general-purpose solution to image-to-image problems translation problems. The implementation used a general-purpose architecture that has been proven effective when applied on various datasets, mainly using a paired dataset. The network utilises a custom autoencoder "U-Net" as a generator and PatchGAN as a discriminator. The structure and training loss functions of both generator and discriminator are quite different from a Deep Convolutional Neural Network, more details will be specified later on. This complex architecture can adapt onto

multiple binary domains datasets, like Figure 1 Edge to Shoe, and have proven to be a successful general-purpose architecture for supervised binary I2I tasks. [2]

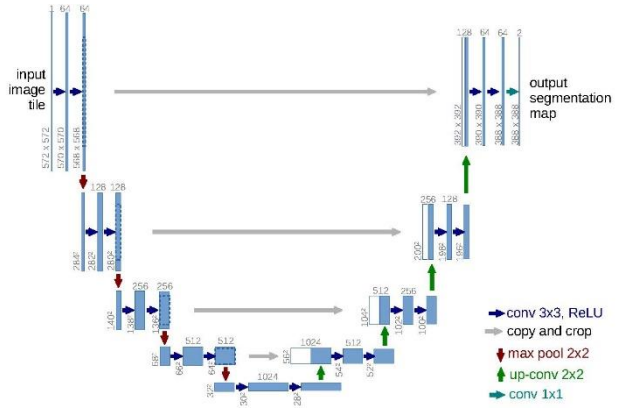

Fig. 3. U-Net architecture provided by the paper [4]

For property 3, I found a paper published in 2015 on "U-Net: Convolutional Networks for Biomedical Image Segmentation" by Olaf Ronneberger. [4] The paper describes the large demand for image segmentation in biomedical image segmentation and the model that was built to accomplish this task. The model used consists of a contracting path to capture the data distribution and an expanding path with skip connections to enable precise localization for their segmentation task. Using this neural network, the author won the ISBI (International Symposium on Biomedical Imaging) cell tracking challenge 2015. Another point to take note, the author made strong use of data augmentation to achieve very strong results. The network is later found to be implemented in the widely renowned Pix2Pix architecture.

III. NETWORK ARCHITECTURE

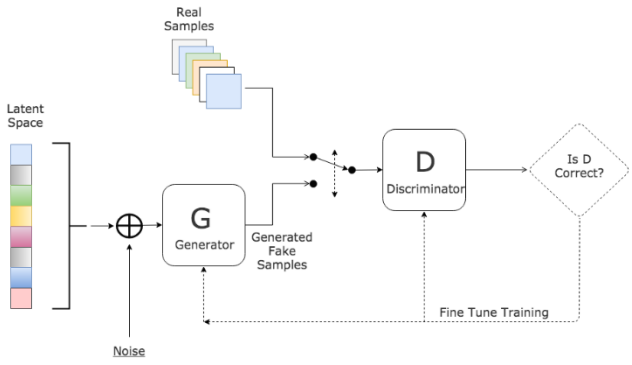

Fig. 4. Structure of Generative Adversarial Network [3]

A typical generative adversarial network contains two neural networks – a generative model G and a discriminative model D. The generative network G learns to map from a latent space z to data distribution x, while the discriminative network distinguishes samples produced from the generative network from the true data distribution. The training objective is to get the generative network to fool the discriminative network, such that the samples produced by the generative network is similar to the true data distribution, also known as image synthesis. In a sense, the intuition of GAN is similar to the Turing Test, "a test of a machine's ability to exhibit intelligent behaviour equivalent to that of a human", is renowned today as the imitation test for artificial intelligence. The Pix2Pix architecture, used in this paper, slightly differs from the original GAN intuition. Instead of having a latent space, the Pix2Pix generator takes in an input image (source domain). It uses a conditional generative adversarial network implementation that the generator learns to map an input image (3D Tensor) to an output image (3D Tensor), hence called Pix2Pix.

A. Generative Model – U-Net

The generator of the Pix2Pix architecture uses a convolutional autoencoder (encoder-decoder) structure that takes in an input image (source domain) and outputs a target image (target domain), modelling a mapping function for translation. The network structure is further extended to a "U-Net" architecture, where skip connections are implemented. According to the U-Net paper, the network consists of two main structures – a contracting path (the encoder) and an expansive path (the decoder). [4] The contracting path consists of strided convolution layers that progressively downsample an image to the smallest form while doubling the number of channels at each layer. Batch Normalization is added for reducing the number of covariates shift to induce stable training; Leaky Rectified Linear Unit is used as the activation function for the network to learn non-linearities representation while producing a non-sparse output. The expansive path consists of strided transposed which progressively up samples the output, from the contracting path, while halving the number of channels of the image. Batch Normalization is added for reducing the number of covariates shift to induce stable training; Rectified Linear Unit is used as the activation function for the network to learn non-linear representations in the data distribution. Between the contracting path and expansive path, skip connections are added between layer i and layer $n - i$. The intuition of the skip connection is that the expansive path would receive two inputs from the previous upsampling layer and the contracting path, increasing precision in the next upsampling layer. The symmetry between the contracting path and expansive yields a U-shaped structure, hence called "U-Net". The U-Net convolutional network came about around 2015, first proven successful for biomedical image segmentation tasks, which is then implemented in many generative neural networks involving an encoder-decoder mapping network. [4]

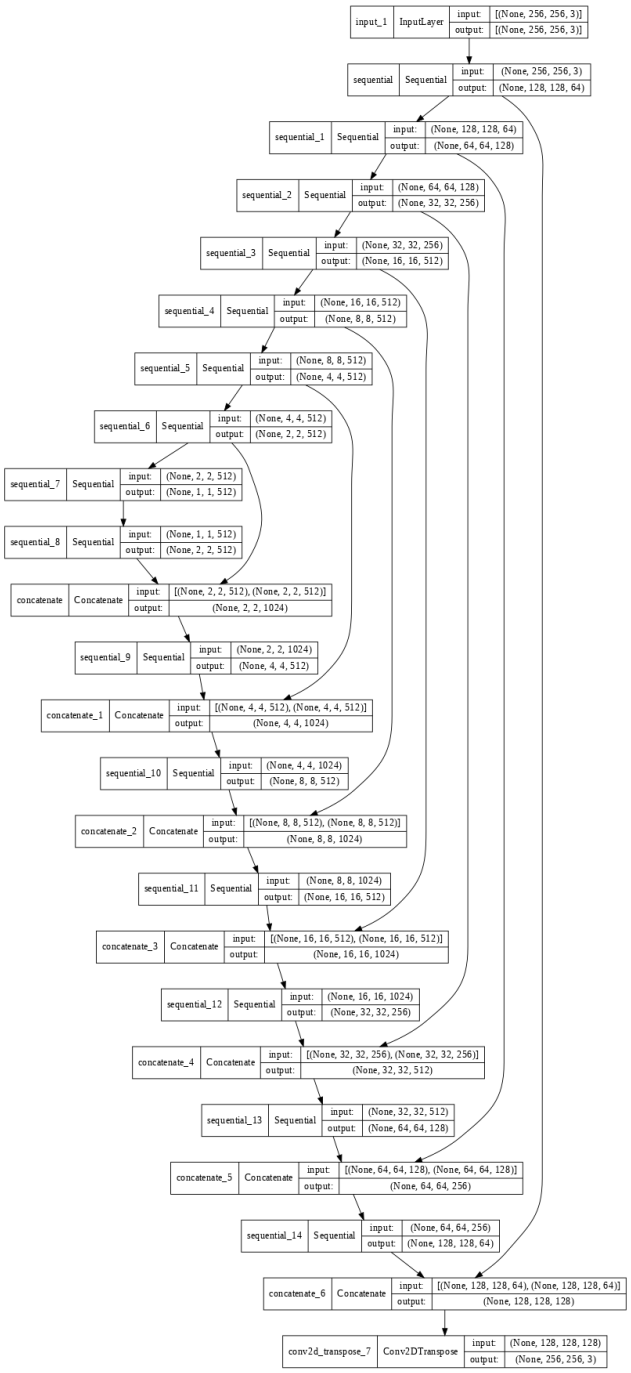

Fig. 5. Illustration of "U-Net" encoder-decoder in Pix2Pix architecture

The U-Net generator trains using two loss functions. First, a binary cross-entropy loss between an array of 1 and generated images discrimination $D\big(G(x)\big)$, to describe the probability similarity error between a generated image and a real image. Secondly, an L1 distance, using Mean Absolute Error, is used between the generated image and the target image. These two terms are added up as total generator loss, with a constant $\lambda = 100$ regularization term.

$$\mathcal{L}_{Generator}(G, D) = BCE\big(D\big(x, G(x)\big), 1\big) + \lambda \mathcal{L}_{L1}(G)$$

B. Discriminative Model

The discriminator in Pix2Pix architecture takes the form of a Convolutional PatchGAN classifier. [2] The discriminator takes in two the inputs – an input image (source domain; ground truth) and a target image (target domain; generated image or training images). PatchGAN takes in input image and target image, then effectively downsamples the image using convolutions like a regular CNN classifier. Each downsampling layer also consists of Batch Normalization and Leaky Rectified Linear Unit as activation function. A typical discriminator, it produce the discrimination of the generated image, to determine whether the image is real or synthetic. PatchGAN slightly alters this concept, it produces discrimination of each $N \times N$ patch or a spatial region of the downsampled image, producing a 2D feature map of discrimination. It then averages the responses and provides the ultimate value of D. [2] The novelty of PatchGAN is that it assumes the independence between each patch, thus it models based the texture and style. Therefore, it can be understood as a form of texture/style loss.

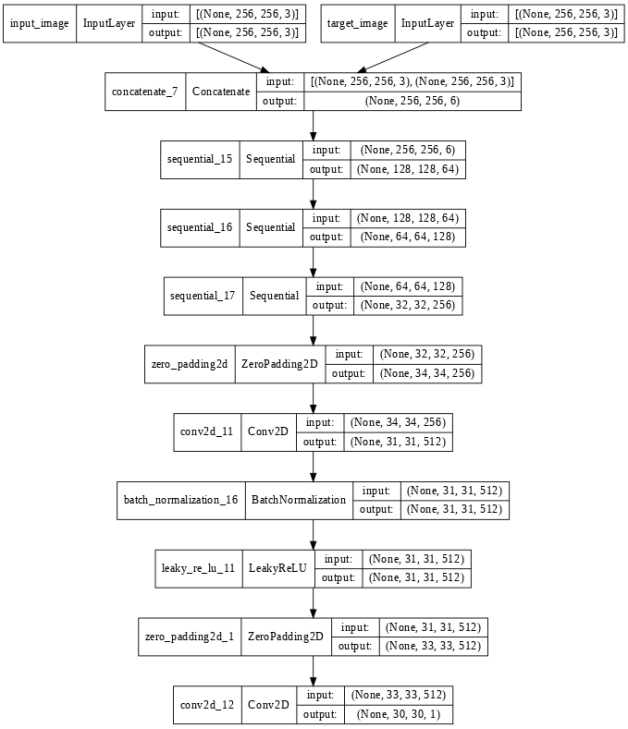

Fig. 6. Illustration of PatchGAN discriminator in Pix2Pix architecture

The PatchGAN discriminator uses two loss functions, similar intuition to the original GAN architecture. First, a binary cross-entropy loss between real images and an array of ones used to represent real images. Second, a binary cross-entropy loss between an array of zeros and generated images, is used to represent fake/synthetic images. These two loss functions are then added up into total discriminator loss.

$$\mathcal{L}_{Discriminator}(G, D) = BCE\big(D\big(x, G(x)\big), 0\big) + BCE(D(x, y), 1)$$

IV. Methodology

Fig. 7. Sample paired images from Cityscapes dataset

To demonstrate the prowess of Supervised I2I, I will be using the cityscape dataset sourced from the UC Berkley data repository. [5] The dataset features a repository of 2975 paired images – a high-quality pixel annotated image and an image of an urban street scene. Annotation refers to using overlaying colours to encode semantic classes; using Figure 6 as an example, blue is used to encode cars, purple is used to encode, and pink is used to encode pedestrian pavements. Each image is of sized 256×256 .The objective supervised I2I task is to translate a semantic annotated image into a realistic photograph.

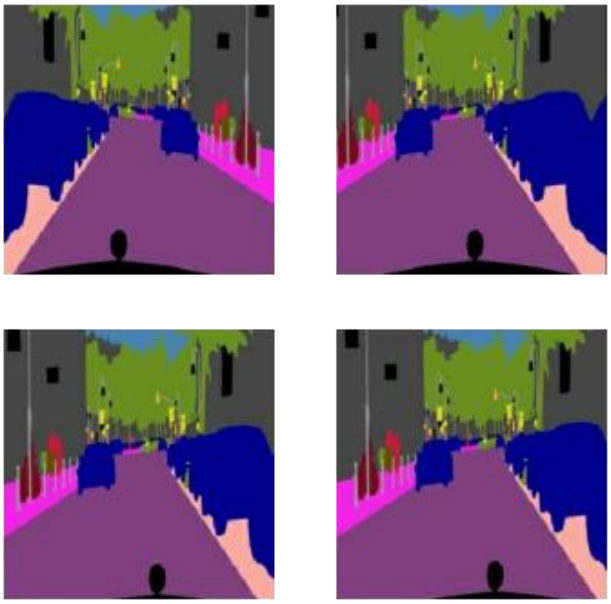

Fig. 8. Annotated images after random jitter and random mirroring

For feature engineering, the authors of Pix2Pix applied slight augmentation onto the input image, random jitter, and random mirroring. Random jitter refers to resizing an image to a slightly large resolution then cropping back to its original dimensions; Random mirroring refers to randomly inverting the horizontal landscape of the image. The intuition is to expose multiple permutations of image variants, such that the generative network would not be fixated on learning from the same set of images, hence also acts as a form of regularization. Unity-based normalization is also done to scale down the model to a range [-1, 1] for easier training.

Fig. 9. Generated Image without Training

For model training, I did my model implementation in Keras and training implementation on TensorFlow. Trainable weights are initialized from a Gaussian weights initialization $N(\mu = 0, \sigma = 0.02)$. For training optimization, I used the Adam (Adaptive Momentum) optimizer with a learning rate of $\alpha = 0.0002$, and momentum parameters $\beta_1 = 0.5$ and $\beta_2 = 0.999$. A batch size of 1 is used for training, as the authors of Pix2Pix mentioned that they found better results for U-Net as compared to a batch size of 10. [2] I used 50,000 training iterations, approximately around 16 epochs, is set to train the Pix2Pix architecture, approximately 40 minutes to train on a Tesla P100 GPU.

V. Discussion

Fig. 10. Generated Image at Train Iteration 0

Fig. 11. Generated Image at Train Iteration 1000

Fig. 12. Generated Image at Train Iteration 10,000

Fig. 13. Generated Image at Train Iteration 20,000

Fig. 14. Generated Image at Train Iteration 30,000

Fig. 15. Generated Image at Train Iteration 40,000

Fig. 16. Generated Image at Train Iteration 49,000

Generally, we could see a steep training improvement between Figure 9 (Train Iteration 0) to Figure 11 (Train Iteration 1000), given the improvement of image quality. This is an indication of the process where the Generator progressively learns to capture and map the data distribution of the source domain to the target domain. However, from Figure 10 to Figure 15, we could see there are is little to no change in image quality, the first-hand indication for training saturation. At train iteration 49,000, we could see a collapse problem might be occurring, given that generated image quality beginning to diminish.

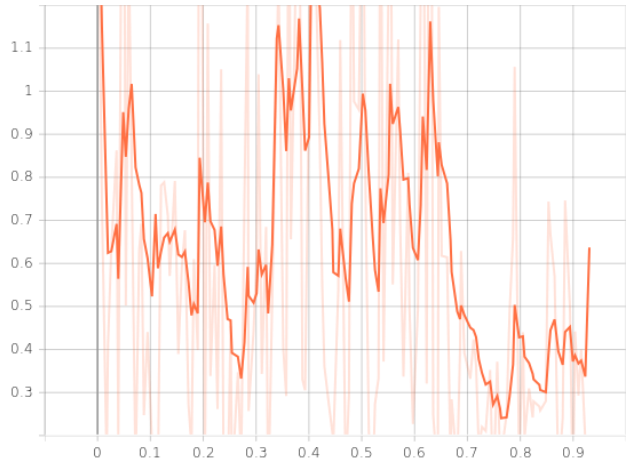

Fig. 17. Discriminator Loss Learning Curve with Smoothing

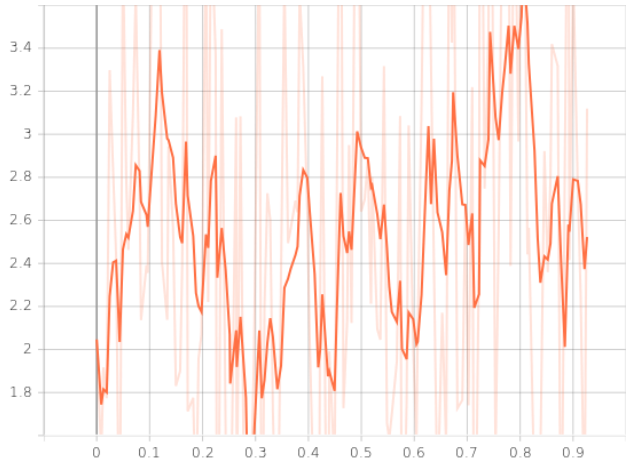

Fig. 18. Generator Loss Learning Curve with Smoothing

The learning curves from Figure 16 and Figure 17 also shows an issue of unequal training, it seems that the discriminator is overpowering over the generator, a common issue while training adversarial networks. The learning curves also affirms our judgement which Generative and Discriminative loss begins to saturate at the end of the training, around 37,500 training iterations.

Fig. 19. Test Generated Image 1

Fig. 20. Test Generated Image 2

Fig. 21. Test Generated Image 3

Fig. 22. Test Generated Image 4

Relatively, the Pix2Pix architecture can roughly perform Image-to-Image translation from the semantically annotated images to recreate a realistic photograph based on Figure 18, Figure 19, Figure 20, and Figure 21. However, inferred from the generated image, we could see that the network is prone to commonly noisy features such as windows, doors and road lines, thus attempting to badly recreate these noises. An example shown in Figure 19, the network translated a black encoding into a stop sign when it should be a lamp pole or fire hydrant. Another issue is Mode Collapse, Figure 20 shows the cars could be seen being disoriented in the same way and only displays red cars. This could mean that the model only learns cars are only red and have the same orientation.

VI. CONCLUSION

In this technical paper, the Pix2Pix architecture is implemented to do supervised binary domain Image-to-Image translation, demonstrated using the cityscapes semantic annotated dataset. The architecture comprises a U-Net generator and PatchGAN discriminator adapted to use for general-purpose supervised I2I. The Pix2Pix architecture has shown to be able to translate semantically annotated images to recreate realistic photographs using the cityscape dataset, however, it is prone to training failures and mode collapse, which is a common problem while training Generative Adversarial Networks. The Pix2Pix architecture used in this case study is not just limited to the cityscape dataset, it can also be applied on other binary domains and has potential in image segmentation tasks.

VII. REFERENCES

1. Y. Pang, J. Lin, T. Qin, en Z. Chen, “Image-to-Image Translation: Methods and Applications”, arXiv [cs.CV]. 2021.

2. P. Isola, J.-Y. Zhu, T. Zhou, en A. A. Efros, “Image-to-Image Translation with Conditional Adversarial Networks”, arXiv [cs.CV]. 2018.

3. S. Hitawala, “Comparative Study on Generative Adversarial Networks”, arXiv [cs.LG]. 2018.

4. O. Ronneberger, P. Fischer, en T. Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation”, arXiv [cs.CV]. 2015.

5. M. Cordts et al., “The Cityscapes Dataset for Semantic Urban Scene Understanding”, arXiv [cs.CV]. 2016.