Fırat Üniversitesi Deneysel ve Hesaplamalı Mühendislik Dergisi

FUJECE

U-Net ve ResNet Entegrasyonu ile Görüntüden Görüntüye Dönüşüm için Hibrit Koşullu GAN Tasarımı

Khaled AL HARIRI[1*], Muhammet PAŞAOĞLU[2], Erkut ARICAN[3]

[1,2] Bilgisayar Mühendisliği Bölümü, Mühendislik ve Doğa Bilimleri Fakültesi, Bahcesehir Üniversitesi, İstanbul, Türkiye.
[3] Bilgisayar Mühendisliği Bölümü, Mühendislik ve Doğa Bilimleri Fakültesi, Bandırma Onyedi Eylül Üniversitesi, Balıkesir, Türkiye.
[1]alharerekhaled@gmail.com, [2]m.pasaoglu.acad@gmail.com, [3]earican@bandirma.edu.tr

Geliş Tarihi: 07.03.2025
Kabul Tarihi: 31.07.2025

Düzeltme Tarihi: 09.05.2025

doi: https://doi.org/10.62520/fujece.1653548
Araştırma Makalesi

Alıntı: K. Al Hariri, M. Paşaoğlu ve E. Arıcan,, ''U-Net ve ResNet entegrasyonu ile görüntüden görüntüye dönüşüm için hibrit koşullu GAN tasarımı", Fırat Üni. Deny. ve Hes. Müh. Derg., vol. 4, no 3, pp. 557-579, Ekim 2025.

Öz

Görüntüden görüntüye dönüşüm, bilgisayarla görme alanının temel görüntü işleme görevlerinden biri olup, stil transferi, görüntü iyileştirme ve benzeri birçok uygulamada kullanılabilmektedir. Bu çalışmada, koşullu üretici çekişmeli ağlara (Conditional GAN) dayalı ve U-Net ile ResNet mimarilerini bir araya getiren hibrit bir üretici mimarisi üzerine kurulu yeni bir yaklaşım sunulmaktadır. Bu birleşim, yüksek uyumlulukları sayesinde modelin her iki mimarinin avantajlarından yararlanmasına olanak sağlamaktadır. Ayırt edici ağ ise PatchGAN mimarisi üzerine inşa edilerek yama bazlı ayrım yapmaktadır. Modelin performansı, görüntü kalitesi değerlendirmesinde standart kabul edilen SSIM ve PSNR metrikleri kullanılarak ölçülmüş, ayrıca aynı ölçütler ve veri kümeleri üzerinden değerlendirilen önceki çalışmalarla karşılaştırılmıştır. Bunun yanı sıra, katılımcılardan önerilen modelin çıktıları ile başka bir çalışmanın çıktıları arasından hedef görüntüye en çok benzeyeni seçmelerinin istendiği bir kamuoyu anketi de yapılmıştır. Hem değerlendirme metriklerinden hem de kamuoyu anketinden elde edilen bulgular, önerilen görüntüden görüntüye dönüşüm yöntemimizin önceki çalışmalara kıyasla üstün olduğunu açıkça ortaya koymaktadır.

Anahtar kelimeler: Görüntüden görüntüye dönüşüm, Bilgisayarla görme, Derin öğrenme, Koşullu üretici çekişmeli ağlar

[*]Yazışılan yazar

İntihal Kontrol: Yes – Turnitin
Şikayet: fujece@firat.edu.tr

Firat Univ Jour. of Exp. and Comp. Eng., 4(3), 557-579, 2025

Firat University Journal of Experimental and Computational Engineering

A Hybrid Conditional GAN Design for Image-to-Image Translation Integrating U-Net and ResNet

Khaled AL HARIRI[1*], Muhammet PAŞAOĞLU[2], Erkut ARICAN[3]

[1,2]Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Bahcesehir University, Istanbul, Türkiye.
[3]Department of Computer Engineering, Faculty of Engineering and Natural Sciences, Bandirma Onyedi Eylul University, Balikesir, Türkiye.
[1]alharerekhaled@gmail.com, [2]m.pasaoglu.acad@gmail.com, [3]earican@bandirma.edu.tr

Received: 07.03.2025
Accepted: 31.07.2025
Revision: 09.05.2025
doi: https://doi.org/10.62520/fujece.1653548
Research Article

Citation: K. Al Hariri, M. Paşaoğlu and E. Arıcan, "A hybrid conditional GAN design for image-to-image translation integrating U-Net and ResNet", Firat Univ. Jour.of Exper. and Comp. Eng., vol. 4, no 3, pp. 557-579, October 2025.

Abstract

Image-to-image translation is one of the major image processing tasks in the computer vision field that can be utilized in many types of applications such as style transfer, image enhancement, and more. This study introduces a novel approach for image-to-image translation based on a conditional generator adversarial network with a new hybrid generator architecture that combines the U-Net and ResNet architectures. This combination allows the model to benefit from both of their advantages due to their high compatibility. The discriminator uses the PatchGAN architecture for patch-wise discrimination. The model was evaluated by using the SSIM and PSNR which are standard metrics for image quality evaluation. The results are also compared to previous work that uses the same evaluation criteria and datasets. Furthermore, a public survey was conducted in which the participants were asked to choose the image that most closely resembled the target image between the proposed model and another study. The outcome of both the evaluation metrics and the public survey successfully demonstrated that the proposed image-to-image translation method is superior to that of previous studies.

Keywords: Image-to-image translation, Computer vision, Deep learning, Conditional generative adversarial networks

[*]Corresponding author

Plagiarism Checks: Yes – Turnitin
Complaints: fujece@firat.edu.tr

1. Introduction

Computer Vision (CV) is one of the most important sections of Artificial Intelligence (AI) and is focused on the way computers view and perceive visual data. This field has received significant progress in the recent years that is especially centered around how computers can manipulate and interpret visual information such as pictures and videos. This kind of output can only be obtained via training deep learning algorithms on incredibly large datasets. Such training would allow the resultant model to be able to recognize, manipulate, and generate visual data. Image-to-image translation, is a field in CV that has shown great effectiveness in generating an image that is translated from one domain to another domain, while keeping the original semantics of the original image. This approach has been very effective and applied to many areas including style transfer, image enhancement, medical imaging, and map making [1].

Early image-to-image translation approaches [2, 3] mostly utilized Convolutional Neural Networks (CNNs) which were effective in terms of feature extraction but limited when it came to generative tasks due to their reliance on predefined loss functions. Generative Adversarial Networks (GANs) revolutionized this field by adopting the game theoretical approach in which the generator and discriminator competed against each other, leading to the development of highly realistic images. However, GANs are typically used for normal image generation and are not fully suited for a task such as image-to-image translation [4] .Conditional GANs (cGANs), introduced by [5] have shown phenomenal performance in structured image translation if accompanied by supplementary input data such as paired examples.

Although cGANs have achieved remarkable success, challenges still persist, particularly in the balance between preserving the details of the spatial and acquiring high-level semantic understanding. This study proposes a novel hybrid generator that combines the high detail preservation of the U-Net with the strong feature representation of ResNet blocks. This hybrid generator is able to output incredibly realistic images with high fidelity due to it leveraging the advantages of both skip connections that come with U-Net and strong modeling that comes with ResNet blocks. Moreover, the discriminator for this model utilizes the PatchGAN architecture which functions by examining the image as separate patches and discriminating the image patch by patch instead of just once, allowing for a more intricate and effective discrimination process.

In this study, the evaluation of the proposed approach is done on two different datasets with standard metrics for image quality evaluation such as SSIM and PSNR. The experimental results were corroborated further by a user study that was conducted in which participants compared the outputs of the proposed model with those of Pix2Pix. The output of both types of metrics (objective and subjective) has shown that performance of the proposed method is better in terms of both quality and realism of the generated images.

2. Literature Review

Visual data in this digital age is one of the most important types of data which is why having methods to interpret and manipulate is very critical. Due to this significance, CV has risen to become one of the fastest growing fields in AI, primarily due to the abundant amount of visual data in this day and age. Furthermore, it has been the core focus in many recent studies, including ones primarily researching image-to-image translation. Among the many fields associated with CV, image-to-image translation is considered one of the most impactful fields due its great diversity in image analysis and manipulation.

With the growth of CV and visual data analysis, many studies have come out with new architectures and approaches for image-to-image translation and other forms of CV. An example of such a study is [6] in which the authors proposed a method for face de-meshing in images using GANs. Moreover, other such studies have researched and utilized this technology in different ways to solve their own purpose of applications in many different sectors and fields. With how vast the world of CV is, it can extend its uses to many sectors such as security, healthcare, navigation, architecture, and education, showcasing the significance of this field and its high influence on the world around it.

Due to the high demand in this field a great amount of research is actively being done to profit from the potential of visual data analysis, which emphasizes the need for an informative interpretation as well as

efficient methods for manipulating this data. More generally, in the context of image-to-image translation, existing works have used this novelty to achieve different goals. The fast development of this industry demonstrates the continuous demand for research to catch up with the fast-growing technologies and tackle a vast range of needs in various domains. The aim for this section is to survey a broad range of related work for CV and image-to-image translation. More directly, the main goal is to talk about the variety of uses of this field, to show them and to compare them with the proposed model. Furthermore the evolution of image-to-image translation over the past years is discussed, including a wide variety of approaches proposed in the literature and a view on possible future research directions.

GANs were introduced in 2014 [7] and consist of two models competing against one another. The generator model creates images while the discriminator model tries to determine if the image is real or fake. Through this adversarial process, if either model manages to deceive the other, the losing model adapts and improves itself accordingly. The advent of GANs has opened up numerous opportunities for various image generation applications, including image-to-image translation. This led to the development of several studies that explore different implementations of GANs in their image generation applications.

GANs serve as an excellent framework for image generation and other visual data tasks. However, image-to-image translation and similar tasks could achieve improved performance when input image conditions were established to produce corresponding output images with their guidance. This led to the introduction of cGANs by [5] in late 2014, which is a conditional variant of GANs where user-provided data acts as a condition for both models. In 2016, [8] published a study that focused on applying cGANs in image editing, enabling modifications to images by applying a condition based on various arbitrary attributes through inverting the mapping of the cGAN. By employing invertible cGANs (a combination of encoders and GANs), the authors were able to reconstruct and alter real images using image-to-image translation. They also assessed their results with an attribute predicate network to evaluate the cGAN and tested modification variables across different cGAN layers.

In 2017, [9] proposed new methods for image synthesis that enhance GAN training outcomes. By using label conditioning, they created a new variant of GANs with image samples at a resolution of 128x128, maintaining global coherence while comparing it with other image resolutions through two analyses. Their analysis indicated that the discriminability of the 128x128 samples was over 50% greater than that of the down sampled 32x32 samples used in the study. Later in the same year, [2] published an image-to-image translation method utilizing cGANs, in which the generator model employs the U-Net architecture [10] and the discriminator model uses PatchGAN [2]. The U-Net architecture, developed by [10] in 2015, enables the generator to incorporate skip connections that help maintain information between the corresponding layers during up sampling and down sampling.

In 2018, [11] applied cGANs to salience detection to transform it into a salience segmentation task. This study used a specific form of saliency, termed the pair-wise image-to-ground-truth saliency. Additionally, the authors translated the saliency mask into a real image via saliency-to-image translation using cGANs. Despite that, the study achieved limited success in both saliency segmentation and saliency-to-image translation.

In 2018, [12] introduced a method that uses Semantic Invariant GANs with the image-to-image translation as a solution for the problem of managing the image's hierarchical semantics. The study kept the semantic information of the original image via creating constraints at the label and space levels. Furthermore, the study introduced a custom constraint termed as the attention loss, which focuses on the most important sections of the original image that need to be maintained in the generated one. The results of the study showed that the generated images were of high quality while also maintaining the semantic information of the original input and met the target image conditions.

In 2018, [13] published a study dealing with the issue of the mapping function in image-to-image translation in which it had little to no comprehensive data for the generated image. To address this, the authors introduced a novel unpaired generative adversarial networks model called AdGAN, which is able to unify the domain to be learned with a domain that has been augmented. The proposed model of this study was made up of a

multitude of generators and discriminators specifically for enabling the unpaired cross domain learning. The final model included six generators (G1, G2, G3, F1, F2, F3) with three domains (X, Y, Z). As the training began, the model went through a number of adversarial loses and full cycle constraint loses. The results of the study showed that the model performed better than the other methods in terms of unpaired cross domain learning.

In 2018, the study [14] introduced a novel version of GANs referred to as InstaGAN. The goal of this study and the introduction of this new GAN is to provide a solution for the challenges in unsupervised image-to-image translation when it comes to complex tasks. This model was particularly important for those requiring considerable modifications to the image or involving multiple target instances. According to the authors, InstaGAN is not only able translate the image, but can even be used to translate an instance containing the collection of instance attributes while preserving each instance attributes' permutation invariance, resulting in an improved multi-instance image translation. The study also introduced a context preserving loss to keep the context intact amongst different target instances. The model was trained using sequential mini batch training. According to the findings of their research, their approach worked well across different datasets that employed multiple instances of target images.

In 2019, [15] used GANs for image restoration and dehazing by image-to-image translation. In order to understand what makes hazy underwater images different, the authors used unpaired image-to-image translation. Even for images that are highly distorted, the results from the study produced were very impressive for their mode. The study used L1 loss in combination with multiple cyclic consistency losses in order to capture the image properties.

Another study conducted in 2019 was the one by [16] which concentrated on addressing the mode-collapse problem associated with cGANs. The authors pointed out that this problem arises because cGANs typically learn an overly simplistic distribution, which disregards variations in latent codes and consistently maps an input to a single output. This does not utilize the multi-modal distributions that cGANs are meant to work with. This challenge has been tackled by incorporating a regularization technique for the generator which assists in forcing the generator to produce outputs with high diversity and that also considers the latent codes. To achieve this, the study implemented diversity sensitive cGANs, and in doing so generated output images with a high diversity, but of good quality too. They also applied their method in image-to-image translation and evaluated their method and the resulting generated outputs in the study.

In 2019, [17] conducted research focusing on image-to-image translation using only a limited number of examples instead of relying on extensive image datasets. The authors created a few-shot unsupervised approach to image-to-image translation, which employs target classes that the model has never encountered except during testing, utilizing only a small number of example images. The primary objective of the study was to mimic human capability to adapt and comprehend the essential characteristics of an object by viewing only a few examples and forming a generalization for future reference. Their method demonstrated remarkable effectiveness in applying image-to-image translation with minimal examples when a large dataset is not accessible.

During 2019, [18] conducted a study to solve the problem of high predictability in the images generated via image-to-image translation by introducing their own novel version of GANs referred as InjectionGAN. This is a new GAN structure which can learn many-to-many mappings as opposed to regular GAN's ability to only learn one-to-one mappings. The study generated images with both the domain specific variables for the target domain and random variations to increase the uncertainty of the image. According to the authors, the input image would be fed into these variables for low predictability. In addition, these two components will be synchronized according to their outputs by a single framework that creates outputs with high diversity in each domain.

Later in 2019, [19] conducted a study that ran a triple translation GAN to solve their face image synthesis and translation problem after thorough research. The study mentioned that the GAN architecture at hand was inspired by the challenges experienced when applying the CycleGAN architecture. The study implemented L1 norm representation constraint to reproduce the relative amounts of detail in the images, improving the

quality of generated outputs. Moreover, the research presented a triple translation consistency loss, which was proven to be extremely helpful for the model to optimize in a more plausible fashion than ever before. The outcome of the research was excellent and outperformed other similar studies.

In 2020, [20] introduced a framework called StarGAN v2 for image-to-image translation. This framework was able to produce outputs with great diversity while having great scalability with multiple target domains. The study proposed this framework due to the absence of an image-to-image translation model that is able to offer both of these features at the same time. The study experimented with their proposed framework on different datasets such as celebrity portraits and animals. The output has shown that the proposed method of this study is better than others in terms of diversity, scalability, and image quality.

3. Dataset

Image-to-image translation is heavily dependent on the quality of the image datasets that are used during model training and while testing that proposed model. This study focused heavily on the data collection aspect as it played a critical role in the output generated by the proposed method. The datasets selected must be made up of very high-quality images while retaining great diversity. Finding such datasets can be difficult which is why the dataset gathering process is extremely important and sensitive for the success of any such task. While it was necessary to compile high-quality datasets, it was equally important to ensure that the datasets could be used freely for this study to comply with data privacy regulations.

During the data collection phase, important criteria was devised that needed to be met by the selected datasets. These criteria included image quality, dataset size, and having absolute image pairs for every sample. Moreover, datasets used in other studies were reviewed especially those that were implemented using image-to-image translation. Isola et al. [2] provided the dataset used during their review, which are also publicly available from the University of California at [21]. Six datasets were devised, cityscapes, building facades, edges, maps, among others. Since the facades dataset and the maps dataset had relatively small sizes at about 29 MB and 230 MB respectively, we primarily focused on these two datasets as they are more fit for the training part regarding time and computational resources. These datasets included pairs of input images and the corresponding ground truths. For example, in the facades dataset, the input images showed building facades while the ground truths were actual buildings. The pairs were combined into a single image, necessitating separation during the pre-processing stage to ensure proper readability by the model. Table 1 presents details about the available datasets, including the number of samples and their sizes.

Table 1. Comparison of facades and maps datasets

Information	Facades	Maps
Size (in MB)	29 MB	230 MB
Train Samples	400	1096
Test Samples	206	1098

The datasets are organized into three separate folders: training, testing, and validation. Each folder contains pairs that are appropriate for image-to-image translation tasks. These pairs are combined into a single JPEG file that measures 512 pixels in width and 256 pixels in height. After separating the paired image, each individual image was 256 pixels by 256 pixels. All images in the dataset pairs are in color and have a size of around 60 KB. Figure 1 show examples of the datasets used in our implementation.

Figure 1. Samples of training dataset

Overall, data gathering for image-to-image translation is an important part of effective performance, which is high quality datasets are sought after. Furthermore, the chosen datasets were ensured to be public, as well as having paired samples of input and target images, which can be used as conditions in the cGAN model.

4. Methodology

This study's approach is based on two separate neural networks, one as generator and one as discriminator, implemented within a cGAN for image-to-image translation. Motivated by the complementarity of the capabilities of the U-Net model [10] for retaining details and the ResNet model [22] for learning complex mappings, the generator combines the U-Net architecture with a basic Residual Neural Network (ResNet) architecture to exploit both of those strengths. The discriminator employs a PatchGAN architecture focusing on single patch of image rather than the overall structure of it, which allows the discriminator to understand every little fraction of the image for evaluating its realism. This configuration allows the two neural networks to compete iteratively, leading to a very high-quality output generated for image-to-image translation.

To fully implement this work, extensive research was necessary with a particular research strategy for reviewing articles on different academic databases such as ScienceDirect, Google Scholar, and Arxiv. The strategy focused on particular keywords associated with the documents we needed such as "Image to Image Translation", "Generative Adversarial Networks (GANs)", "Deep Learning" and other relevant phrases. Several aspects were addressed in the criteria for inclusion and exclusion such as relevance of the article to the topic of image-to-image translation, publication date and the language of the article. It was decided to limit the articles to those written in English and published between 2014 and 2024. We only chose the most recent articles and those specific to our implementation to make sure they were relevant. Any study that did not follow these criteria was excluded except a handful that were necessary for the base of this study.

4.1. Data pre-processing

Deep learning projects always require data pre-processing due to the raw data needing to be converted to a format ready for model training. In this study, each pair of images in the dataset is contained within a single image file, so the initial step of pre-processing involved splitting each pair into two distinct images: a pair of input image and target image. To achieve this end goal, the width of the images was cut in half, and used the resulted new width on trimming the image matrix, isolating each individual image. After the images were separate, they were then converted into TensorFlow float objects to make them useable in the deep learning framework.

Data augmentation aims to generate a new much diverse dataset without changing the semantic meaning of the original by keeping the structure of the original images. After the images were separated, they were augmented to expand the dataset and make the model more robust. One data augmentation technique used was to enlarge the image dimensions then randomly crop it back to its original size, picking random pieces of the resized images. Additionally, a random horizontal flip of the image was applied, introducing a 50%

probability of mirroring each image in the training data. After pre-processing, the input images were split into four potential results, as shown in Figure 2.

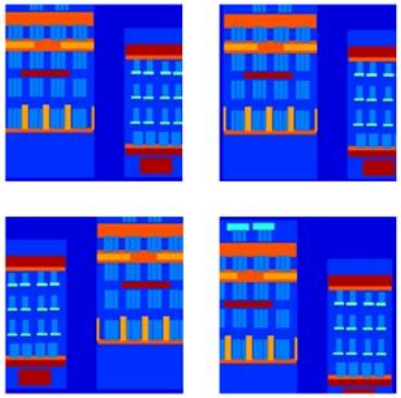

Figure 2. Showcasing dataset after pre-processing

The steps employed are vital in generating a strong and diverse dataset for training image-to-image translation models.

4.2. Model architecture

For image-related tasks in CV, Generative Adversarial Networks (GANs) are often used, which are composed of two linked neural networks: the generator and the discriminator. The generator's objective is to produce data samples that appear as realistic as possible, while the discriminator's aim is to recognize the generated samples and differentiate them from real data. In this context, the generated data samples will be images used for image-to-image translation tasks.

However, standard GANs are not the most suitable option for image-to-image translation tasks because they do not provide any conditional information about the generated samples, which is necessary for generating them in a specific manner, unlike conditional GANs (cGANs). Therefore, this study's implementation employed cGANs as the architecture for image-to-image translation tasks, owing to their capacity to learn mappings between different domains by using paired data samples included in the collected dataset. This capability is vital for the implementation, as it is designed to be adaptable, allowing it to work with various datasets (such as facades, maps, etc.), each considered to belong to a different image domain. In this study, the target images within the paired data will serve as the conditions used in the cGAN model. The subsequent two subsections will provide a detailed explanation of the generator network architecture and the discriminator network architecture.

4.2.1. Generator network architecture

The generator in our proposed cGAN model is designed to perform image-to-image translation by learning a mapping from an input image to a corresponding target image. To enhance both low-level detail preservation and high-level feature transformation, this study constructed the generator using a hybrid architecture that combines U-Net [10] and ResNet [22] structures in a unified and complementary manner.

The U-Net architecture is essentially made up of three sections, downsampling, upsampling, and skip connections. Downsampling blocks are used for reducing the spatial dimensions of the input images while also making sure to extract the intricate details of the image so they can be passed on later. Upsampling blocks are used for increasing the spatial dimensions of the feature maps to around the size of the desired output image. Skip connections are extremely important as they handle the preservation of the details during

the movement from downsampling to upsampling in which each downsampling block is connected with its corresponding upsampling block.

A ResNet block is made up of two convolutional layers in which each layer applies batch normalization and ReLU activation. Furthermore, skip connection is applied where the details of the block's input are added to its output. This configuration allows the generator to keep important semantic information while allowing the discovery of new complex features at the same time. The use of the ResNet architecture within the U-Net bottleneck is important. This approach increases feature extraction in the U-Net without compromising its core role of maintaining spatial resolution. In turn, the generator is able to provide a consistent visual presentation while still allowing minor differences in texture and subtle differences in the target environment.

In the middle of the U-Net architecture, basic ResNet blocks are added in between downsampling and upsampling which are used for capturing the details of the image and in learning complex image transformations. This combination allows the model to be more expressive in which the U-Net will focus on preserving the spatial information while the ResNet will focus on enhancing feature learning. This will result in the model being able to learn much more complex mappings.

In the generator, all layers are initialized with a random normal distribution (mean 0, std 0.02) to stabilize training, and the input layer accepts 256×256 RGB images. The architecture begins with three downsampling blocks using 4×4 convolutions with stride 2 and Leaky ReLU activations, starting with 64 filters (no normalization) and then 128 and 256 filters with layer normalization. This is followed by three ResNet blocks, each with two 3×3 convolutions with 256 filters, stride 1, batch normalization, and ReLU activations for deeper feature extraction. Two upsampling blocks with 4×4 transposed convolutions (stride 2, ReLU) expand spatial dimensions using 128 and 64 filters, each concatenated with its corresponding downsampling block for skip connections. Finally, a 4×4 transposed convolution outputs 3-channel images with a tanh activation to scale pixel values between -1 and 1, producing realistic images. The overall generator consists of downsampling blocks, ResNet blocks, upsampling blocks, and the final output layer.

Figure 3 illustrates this study's generator architecture in detail, showing all the different steps and their interconnections.

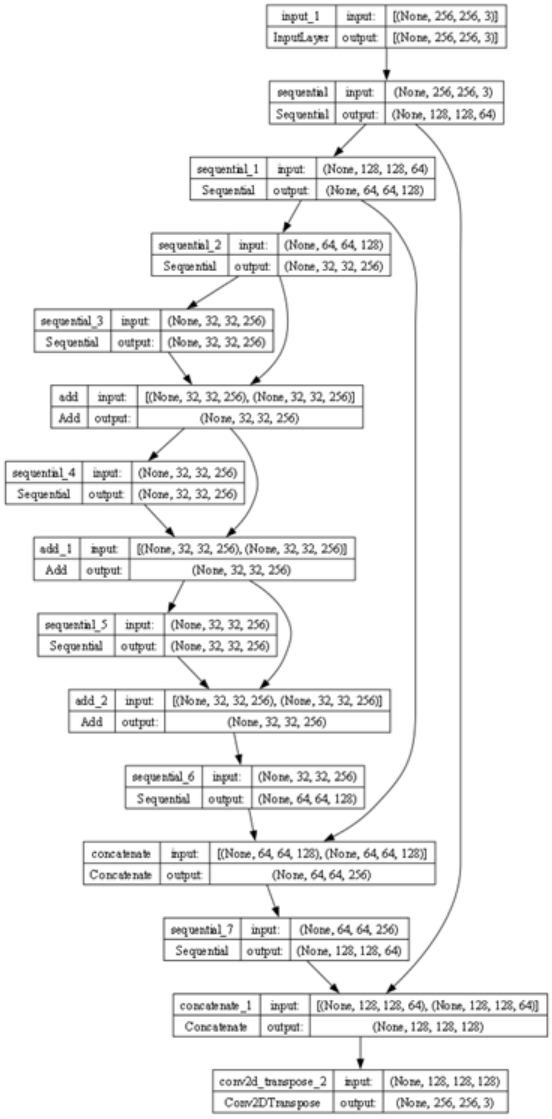

Figure 3. Architecture of our generator with U-Net and ResNet

This study uses two loss functions for calculating the generator loss, the adversarial loss and the L1 loss. The adversarial loss, which is known as the GAN loss, is used for calculating and minimizing the difference between the predictions of the discriminator depending on the output from the generator in terms of authenticity. This will help the generator in generating higher quality images that will be harder to identify as fake and look very similar if not the same as the real images. This is coming from the perspective of the discriminator as the generator will aim to minimize the difference of the discriminator's output between the generated images and real images.

The L1 loss, which is often also referred to as the mean absolute error, is calculated based on the pixel values that differentiate between the generated image and the target image. The main difference between the adversarial loss and the L1 loss is that the L1 loss focuses on the image fidelity rather than the realism of the image. In other words, the important details that make up the core idea of the target image will be preserved and taken into account by the generator. These two loss functions will keep the out of the generator in check when applied and help it in generating realistic images that meet the conditions of the target image.

4.2.2. Discriminator network architecture

The main goal of the discriminator is to successfully differentiate between the fake and real images by pointing out the fake one. The PatchGAN architecture [2] is especially effective in tasks such as image-to-image translation due to it viewing the image as separate patches instead of just one whole image. This discrimination approach enables the discriminator to split the image into patches and evaluate each one separately. After that, the discriminator is able to combine the result of these evaluations and issue a final verdict on the image. In the case the image is judged as fake by the discriminator, it will specifically pinpoint the patches that caused the issue and report it back to the generator. With such detailed discrimination results, the generator will know exactly where it needs to improve its generation.

In this study, all the layers of the discriminator are initialized randomly with a mean of 0 and a standard deviation of 0.002, just as in the generator. Moreover, the input image and the target image are both colored images of size 256x256. Both images are first concatenated into a single tensor for comparison purposes, allowing the discriminator to easily compare the input and target images during training. The discriminator is first made up of 3 downsampling blocks that use convolutional layers alongside the Leaky ReLU action function. The goal of these blocks is to decrease the spatial dimensions of the input and determine the most important features to focus on during discrimination.

The first downsampling block does not use normalization and is made up of 64 filters. The subsequent two blocks contain 128 and 256 filters respectively and incorporate layer normalization. Once downsampling is completed, a zero-padding layer is introduced to maintain spatial information and prevent edge artifacts by surrounding the borders of the feature maps with zeros.

Subsequently, a convolutional layer featuring 512 filters and a stride of 1 is included to compute features from the padded feature maps. Following this, layer normalization is enacted, and a Leaky ReLU activation function is added, succeeded by yet another zero-padding layer. Ultimately, a final convolutional layer is introduced, which contains 1 filter and a kernel size of 4, tasked with generating the output for each patch during the discriminator's operation. Figure 4 illustrates this study's discriminator architecture in detail.

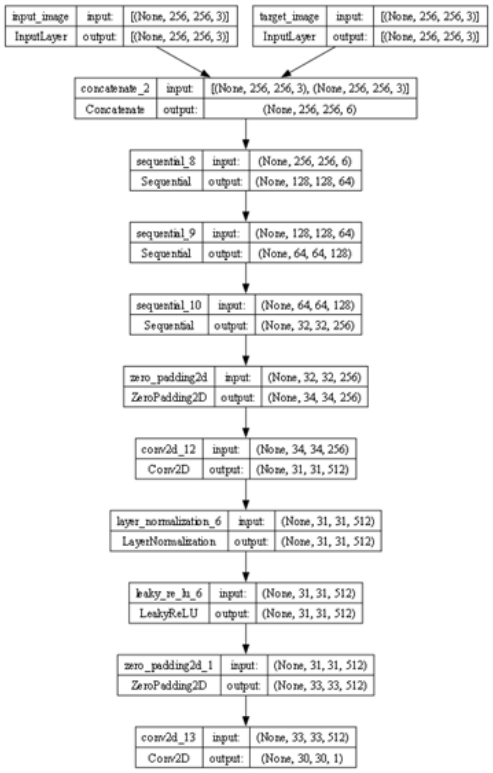

Figure 4. Architecture of our discriminator with PatchGAN

In order for the discriminator to differentiate the real images from the fake ones, a discriminator loss is used that will be part of the training process. This study used two loss functions for calculating the discriminator loss, the real image loss and the generated image loss. The goal of the real image loss is to check the accuracy of the discriminator in classifying real images as real. It is calculating it by doing a comparison between the output of the discriminator when a real image is evaluated with a target label that shows that these images are real. In this case, the target label is a tensor of ones in which they indicate that the evaluated image is a real image and can be used for comparing with the output of the discriminator. The real image loss aims to check the result of the comparison between the output and the target label, and to also minimize the difference between them to encourage the discriminator to correctly classify real images as real.

The generated image loss primarily focuses on evaluating the image in terms of authentic by checking if it is real or fake. This loss is calculated by comparing the output of the discriminator with a label that indicates if the image is real or fake. The target label for fake images is a tensor of zeroes. The generated image loss is extremely important as it aims to minimize the difference between the output of the discriminator and the target label. This will then push the discriminator to correctly classifying the images generated by the generator as fake.

4.2.3. Training process

This section will detail the training procedure employed in our implementation, along with all the configuration values utilized during its setup. This training routine works on both the generator and discriminator by leveraging their respective loss functions to drive minimization.

In the training cycle, steps were used instead of epochs, allowing the loop to traverse the full training dataset while supplying batches of input and target images to our model. During each step, gradient tapes are employed to record the operations for automatic differentiation during backpropagation. Once these are established, the generator is given an input image, which it processes to create an output image corresponding to that input. Subsequently, the discriminator assesses the output image generated by the model alongside the target image to compute the loss functions. The generator loss and discriminator loss are computed after completing each step, with the data used for these calculations depending on the current batch being processed.

This study is using the Adam optimizer for both the generator and the discriminator, with a learning rate of 0.0002 and a beta value of 0.5 for both networks. The optimizers adjust the parameters of the model each iteration depending on the values of the loss functions for both the generator and the discriminator. The parameters are adjusted using methods such as gradient descent with the goal of reducing the loss in every iteration. The evaluation metrics are then calculated for evaluating the quality of the generated images. This study utilized the Structural Similarity Index (SSIM) and Peak Signal-to-Noise Ratio (PSNR) as the metrics for these evaluations and are calculated every step.

Every 1000 steps, an example of the generated images is displayed alongside the generator loss, discriminator loss, and metric values. After every 5000 steps, a checkpoint is created to capture the model's state at those intervals, allowing for future evaluation, testing, or the option to resume training. This study utilized 50,000 steps to train the model as well as the for model used in the comparative study based on Isola et al. [2]. The primary objective of this training loop is to minimize the loss function outcomes for both the generator and the discriminator. Figure 5 shows an example of the training data and the resulting output of this method.

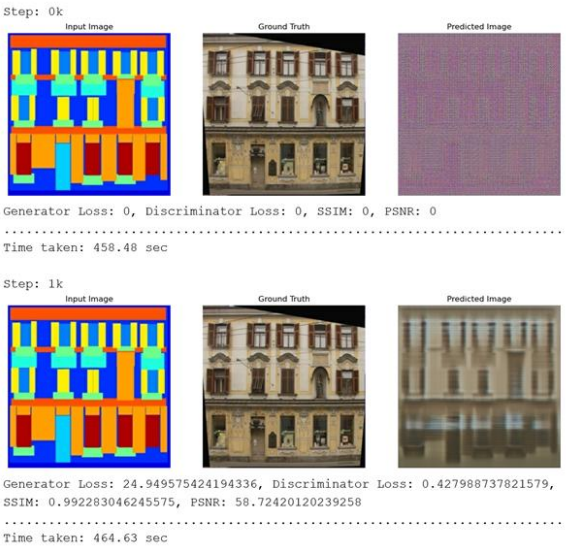

Figure 5. Examples outputs from our model training

5. Findings and Discussion

This section is focused on the results of the proposed model in terms of performance and quality via both objective and subjective metrics. Specifically, the performance of the model is evaluated by comparing it with the Pix2Pix framework [2] by using the same dataset [21] and the same experimental setup. The objective metrics used for evaluating the performance of the model are SSIM and PSNR. These metrics are known for evaluating images in terms of image fidelity and visual resemblance between generated and reference image. The values of the SSIM and PSNR are calculated every step and are logged every 1000 steps within 50,000 steps of training. Figure 7a and Figure 7b show the SSIM and PSNR values respectively during the training process. The rise in both values as the training goes on shows that that the generator is producing outputs that more resemble the target images both in terms of structure and appearance over time. SSIM shows a significant rise during the first 20,000 steps, while PSNR grows steadily.

Additionally, the training loss for both the generator and the discriminator is monitored. This can be in Figure 8a and Figure 8b, the generator's loss declines gradually from 35.1 to 22.2 aligning with the decline in the discriminator's loss from 1.02 to 0.39. The observed inverse trend in the generator and discriminator losses is concurrent with the standard adversarial training. The ability of the generator to deceive the discriminator grows, resulting in this component becoming better at discerning real from false images.

In order to prove or validate the reliability of our results, this study performed a side-by-side analysis with the Pix2Pix framework [2]. To achieve this goal, the Pix2Pix generator and discriminator were reproduced, with the same losses on the same datasets and with the same training configurations. Our method converges better than Pix2Pix as indicated by the Figure 10a–Figure 10d. The generator achieves a lower final error, and this study's SSIM/PSNR figures continue to be superior. This model leads to a mean SSIM value of 0.0822 and PSNR of 6. These results indicate that the use of ResNet in the U-Net architecture improves both generalization and the retention of fine details.

In order to further evaluate perceptual quality, this study performed a user evaluation with 60 participants who chose the images that they believe most resembled the original ground truth image, viewing these from this model and from Pix2Pix. The results have shown that this study's model was chosen by participants in 56.2% of cases, which also supports the previous quantitative findings. Such user input confirms the fact that this solution provides a qualitative advantage, particularly when priority is given to human perception.

5.1. Results

During this section, the results of our model alongside the input and target images are going to be showcased. In the training, there exists 3 images, the input image, the target image, and the generated image. When using the model for testing, the only the input image is provided to it, and it will generate the output image according to the input image and how it gets translated. The training process involves three images: the input image, the target image and generated image.

The objective of this model is to create an image from the input that closely resembles the target image. The target image is provided to the model solely during the training phase to inform it of the results we aim to achieve. Figure 6a illustrates some examples of the results using the facades test dataset, displaying the input image alongside the generated image. This framework is designed to be adaptable, allowing it to be trained on various datasets while still achieving outstanding image-to-image translation. In Figure 6b, several examples of our outcomes using the maps test dataset is presented.

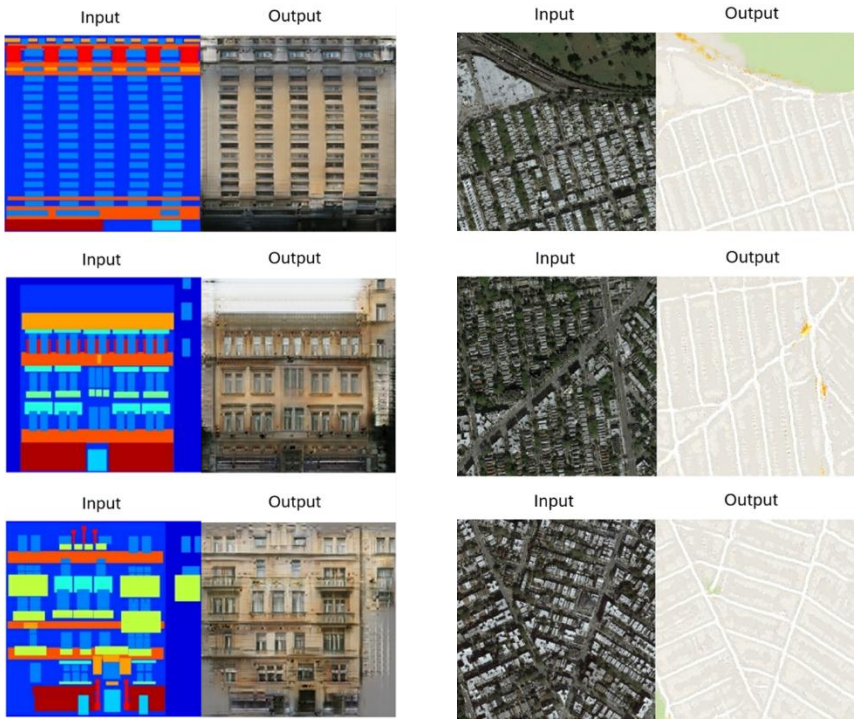

Figure 6. Comparison of our generator results on two different datasets. (a) Example of our generator results with facades dataset, (b) Example of our generator results with maps dataset

5.2. Quantitative evaluation

For the experimental evaluation of the image quality create, this study used SSIM and PSNR as standard metrics. The Structural Similarity Index Measure (SSIM) is based on the measurement of similarities between the distorted image and the reference image considering the luminance, contrast and structure. The SSIM index is computed using these aggregated values forming higher values for worse quality than the reference image, especially when comparing a distorted image to a reference image [23]. The SSIM index value is computed firstly after evaluating each of these elements separately, and then by aggregating these elements. As shown in equation (1), SSIM index is computed by the formula.

$$S\,(x,y) \;=\; f\,(\,l\,(x,y), c(x,y), s(x,y)) \qquad (1)$$

In contrast, PSNR measures the quality of a reconstructed of a reconstructed image by comparing it to the original image, relating to Mean Squared Error (MSE) [24]. The equation for computing both MSE and PSNR is illustrated in equation (2) and (3).

$$MSE = \frac{\sum_{i=1}^{n}(y_i - x_i)^2}{n} \tag{2}$$

$$PSNR = 10\log_{10}\left(\frac{MAX^2}{MSE}\right) \tag{3}$$

To compute SSIM and PSNR, images in the test dataset were processed using TensorFlow. This involved converting images to tensors, changing them to grayscale, and then applying the TensorFlow functions for metrics computation.

During training, average SSIM and PSNR values are calculated every 1000 steps on the testing dataset, tracking the generator and discriminator loss as well. During the course of 50,000 steps, 50 SSIM and 50 PSNR values are obtained and stored for later performance evaluation. After training, the results are visualized with line graphs to illustrate improvements in SSIM and PSNR for the facades dataset, as shown in Figure 7a and Figure 7b.

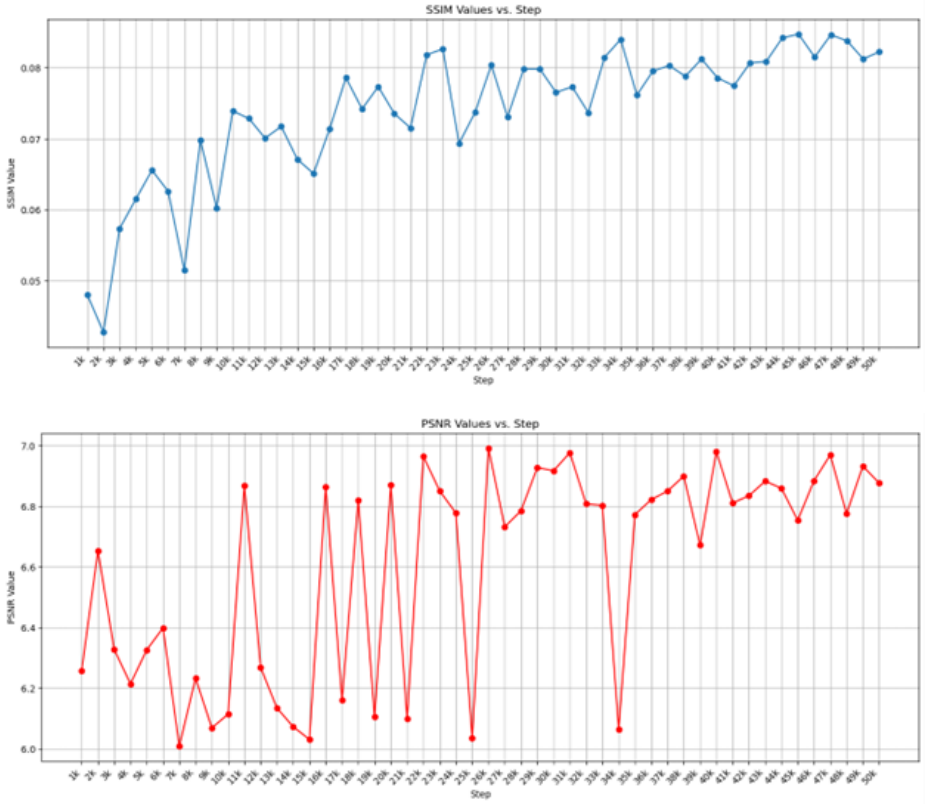

Figure 7. Comparison of SSIM and PSNR values over training steps. (a) Line graph of our SSIM values over training steps, (b) Line graph of our PSNR values over training steps

Once the training was complete, all the image pairs in the test dataset were examined, an output image is generated by the trained generator, and then SSIM and PSNR values were computed for each one against the target image. After all of these steps are completed, the average value is computed for the metrics, in which SSIM was 0.082270745 and PSNR was 6.8772073.

5.3. Training loss results

This section examines the losses of both the generator and the discriminator during the training loop, and how these compare to the loss reported in the comparison research paper.

Throughout the training procedure, this study has employed two distinct loss functions: one for the generator and another for the discriminator. The generator's loss assesses how closely the generated images resemble the target images, while the discriminator's loss evaluates its ability to differentiate between real and fake images. Typically, losses begin at high values and decrease as training progresses. This occurs because, initially, both the generator and the discriminator are inadequately trained, leading to poor performance in achieving their respective goals. As training advances and the networks enhance their capabilities, it is expected that the loss will decrease, reflecting these improvements. In this case, the generator and discriminator losses are computed after every 1000 steps in the training loop and are recorded for performance evaluation. Both Figure 8a and Figure 8b illustrate the generator loss and the discriminator loss, respectively, as a function of the training steps when applied to the facades dataset.

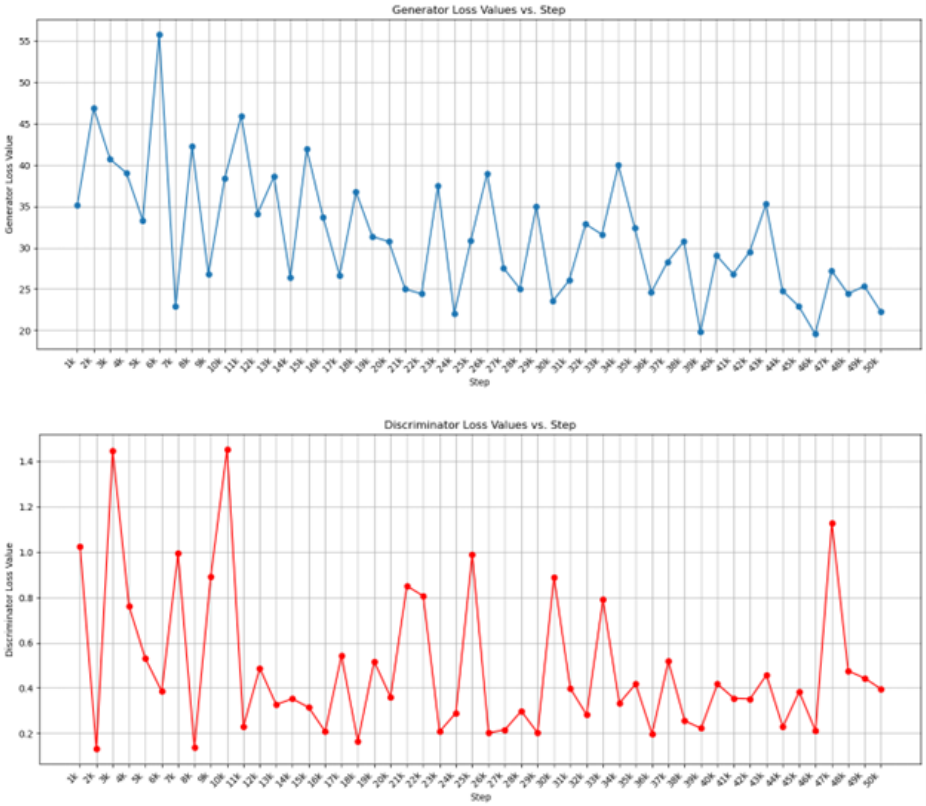

Figure 8. Comparison of generator and discriminator losses over training steps. (a) Line graph of our SSIM values over training steps, (b) Line graph of our PSNR values over training steps

The results illustrated in Figure 8a indicate that the generator loss begins at a value of 35.1 for the initial 1000 steps. The generator loss reduced to 22.2 by the end of the training. Figure 8b shows the discriminator loss, which began with a value of 1.02 during the first 1000 steps and then steadily declined to 0.39 by end of the training. The figures clearly demonstrate how the generator loss and discriminator loss values contrast with each other as the generator and the discriminator compete to determine the authenticity of the generated images, with each network enhancing its performance based on the outcomes.

5.4. Comparing results

This section will create a comparison between the findings of this study regarding presentation and evaluation metrics with the results from the research of Pix2Pix [2]. The datasets used are the same as those employed in their study, which simplifies the comparison between the models.

The comparison focused on quality of the generated images from both generators and how similar they were to the corresponding target images. The input images that led to the generation of these outputs is also shown. This analysis was performed on the test dataset, in which each sample is made up of an input image and a target image. Figure 9a illustrates the comparison between the results of this study's generator and the generator from the [2] study, featuring examples from the facades testing dataset. Figure 9b shows the comparison with using the maps dataset.

Figure 9. Comparison of results with Isola et al. on different datasets. (a) Comparison of results with Isola et al. on facades dataset, (b) Comparison of results with Isola et al. on maps dataset

To compare this study's findings with the Pix2Pix research conducted by [2], the SSIM and PSNR figures were computed using their model in the same manner as in this implementation with 50,000 steps. Additionally, values were plotted in the same format as previously based on the steps, which can be observed in Figure 10a and Figure 10b respectively when trained on the facades dataset.

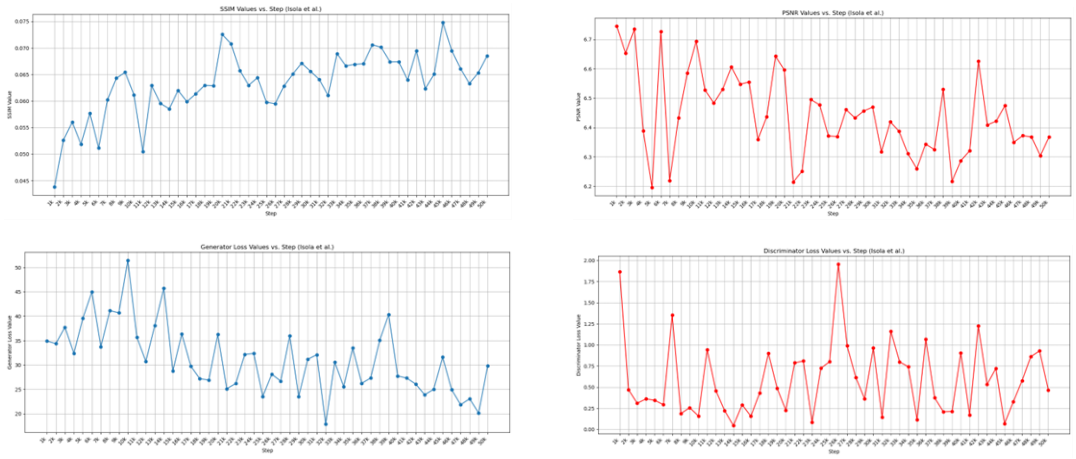

Figure 10. Loss over training steps for Isola et al. [2] (a) Line graph of Isola et al. SSIM over training, (b) Line graph of Isola et al. [2] PSNR over training, (c) Line graph of Isola et al. [2] generator loss over training, (d) Line graph of Isola et al. [2] discriminator

Additionally, the average SSIM and PSNR values were computed on the testing dataset following the completion of training, resulting in 0.068490095 and 6.3685718 respectively. The findings indicate that this study's average values are superior, demonstrating that the proposed model can produce images of higher quality that more closely resemble the target image compared to the [2] study.

To compare this study's generator and discriminator training loss results with those from Pix2Pix research [2], the losses from their model were calculated in a similar manner, which was trained for 50,000 steps, recording the losses every 1000 steps. Afterward, these losses were saved for comparison with the proposed model. Figure 10c and Figure 10d present line graphs illustrating the generator loss and the discriminator loss from the [2] study, based on the step count during their training on the facades dataset.

As illustrated in Figure 10c, the generator loss observed in the [2] study began at 34.9 during the first 1000 steps and decreased to 29.8 by the conclusion of the training process. In contrast, the proposed generator network achieved a loss of 22.2 at the end of its training, indicating superior performance compared to the generator from the research paper [2]. Similarly, Figure 10d depicts the discriminator loss from the research paper [2], which commenced at 1.86 in the initial 1000 steps but fell to 0.46 by the training's end. This also demonstrates that the proposed discriminator outperformed theirs, finishing with a loss of 0.39 at the conclusion of the training process.

5.5. Survey results

In order to also evaluate the visual results of the proposed model, a user study was conducted which compares the proposed model's output to those of [2] in a public survey that was based on similar questions with similar comparison metrics. The human visual evaluation is thus incorporated into the evaluation of this study's results, ensuring the high quality of the generated output.

5.5.1. Structure of the survey

To ensure no bias in treatment to both models, 25 random pairs of inputs and targets were chosen from the facades testing dataset. After that, the corresponding generated images for these inputs were obtained from the proposed model as well as the Pix2Pix model [2]. This process provided three images for each pair: this study's generated image, the ground truth image and the image generated by Pix2Pix [2]. Each group of images was turned into a separate folder with a folder name given by a hashed value of the target image matrix. After all of this was done, the survey was fed all of these images to prevent any bias.

The survey was conducted between 22nd of April 2024 and 12th of May 2024 on Google Forms with 60 total participants that answered the questions. The survey consisted of 25 multi-choice questions in which all of these questions have the same structure. Participants were limited to only being able to select one option for each question. Each question featured a ground truth image, accompanied by the following title: 'Which of the two image options matches the below image more?". Moreover, there were only two image options for each question, one generated by the proposed model and one generated by the original study [2]. Both of these image options are generated according to the ground truth image in the question using an input image. The participants were asked to select the generated image that looks the most similar to the ground truth image given in each question. An example question from the survey can be seen in Figure 11.

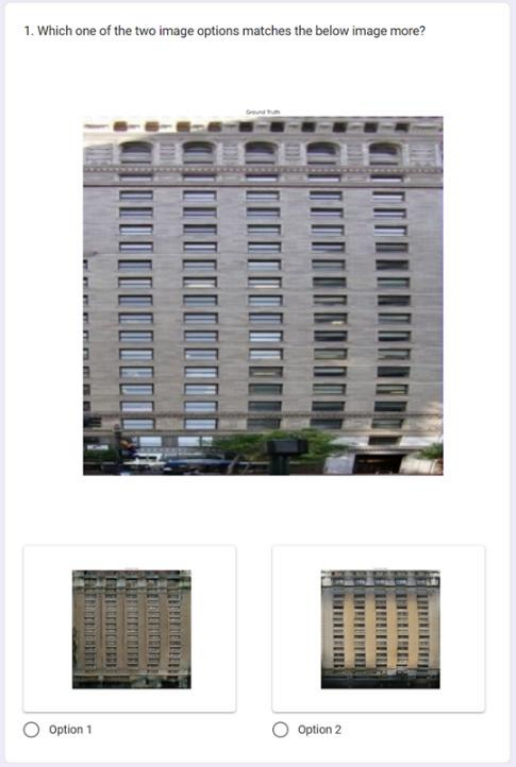

Figure 11. Example showcasing our survey structure

5.5.2. Results of the survey

Throughout the survey, 60 distinct responses were gathered in which each participant answered all questions by choosing images that best represented the ground truth. In general, the survey findings favored the proposed model over the original study [2]. Among the 1496 possible responses to all questions, 841 were attributed to the proposed model, while just 655 were associated with the Pix2Pix model [2]. This indicates that 56.2% of the total responses preferred the proposed model. The results are illustrated in Figure 12a as a pie chart displaying the number of responses.

Additionally, among the 25 questions included in the survey, 15 yielded results that predominantly supported the proposed model, 8 favored the alternative study, and 2 resulted in a tie. These findings are also illustrated in Figure 12b, which presents the data in a horizontal bar chart.

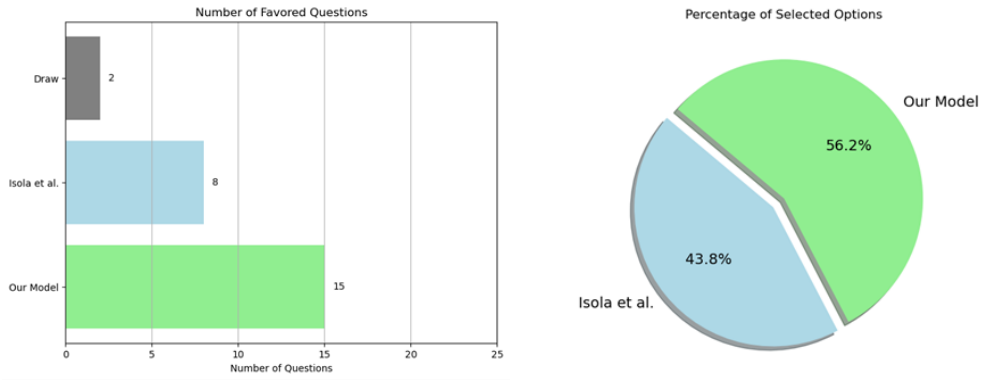

Figure 12. Comparison of Survey results in percentage and total counts. (a) Survey results (percentage of total answers), (b) Survey results (number of favored questions)

5.6. Discussion

The outcome of the proposed image-to-image translation model has already been shown to be promising and fulfilled all our expectations in terms of performance and efficiency. The visual inspection of the generated images that were generated by the proposed model has shown that they had high resemblance to the target images and were of high quality with accurate color representation and semantic consistency. The evaluation metrics used (SSIM, PSNR) also outperformed those described in [2], and the images generated were closer structurally with resolutions from this model rather than the research paper [2]. Moreover, the generator loss and the discriminator loss of the proposed model were lower than those mentioned in the original paper [2], assisting the strong performance of this study's model. Table 2 shows a comparison between the proposed model and the original study's model in terms of evaluation metrics and other variables.

Table 2. Comparison of evaluation metrics with Isola et al. (2017)

Evaluation Metric	Isola et al. (2017)	Our Method
Average SSIM	0.068490095	0.082270745
Average PSNR	6.3685718	6.8772073
Generator Loss	29.8	22.2
Discriminator Loss	0.46	0.39
Survey Selection Percentage	43.8%	56.2%

The positive results of the proposed model proves that this approach is very effective in image-to-image translation and can be applied to various other image processing tasks.

6. Conclusion

This study proposed a new architecture built on top of a cGAN that utilizes a hybrid generator of two architectures which lead to a contribution to the research on image-to-image translation and the CV field as a whole. The goal of the proposed architecture is using image-to-image translation to generate high quality images that retain the important semantic information of the original image while meeting the conditions of the target image. The proposed model was evaluated using SSIM and PSNR which are standard metrics used for image quality evaluation. The values of these metrics were compared with those of another study that used a similar approach. The comparison showed that the proposed method performed better than the other study, validating the effectiveness of this approach. The main feature of the proposed method is a hybrid generator which combines the U-Net architecture with components of a simple ResNet. This combination of the two architecture combines the localization-enhancing feature of U-Net with the deep representations from

the ResNet resulting in high quality images that retain the details of the originals. For the discriminator, this study used the PatchGAN architecture which splits the image into several patches and evaluates them separately, resulting in detailed discrimination results. The model that resulted from these architectures was able to generate images with high quality and accuracy due to the focus keeping the image quality high and preserving the intricate details of the image.

Despite all the success, this implementation suffers from a number of disadvantages that could be improved in future studies. The extremely long period of training is example of such disadvantage which is due to this model being very resource intensive. The proposed model also faces an issue with generating complicated structures depending on the diversity in comparison to the primary data distribution used, as in when there are too many variations. Future studies could explore these drawbacks and work on optimizing the model in such the resource cost is minimized. They could also work on improving the scalability of the model to be more suited to generating complex structures that come with many variations.

In conclusion, this study proposed a new approach for image-to-image translation using a cGAN with a hybrid generator architecture. The study also explored many other papers discussing image-to-image translation and compared the proposed method with those studies. The proposed model was validated through standard metrics with the results compared with similar studies. Furthermore, the model went through a public survey in which the generated outputs were directly compared with other approaches and validated by participants. This contribution is made with the hope that it can push forward future research in this field and inspire further development for image-to-image translation.

7. Acknowledgements

This study is part of a master's thesis at Bahcesehir University.

8. Author Contribution Statement

Author 1 conducted the research as part of his thesis, designed the study, performed the experiments, and analyzed the results under the supervision of Assistant Professor Erkut ARICAN, Author 2 contributed by refining the structure of the manuscript, organizing the results for clarity, and ensuring the manuscript met the standards for journal submission, and Author 3 provided supervision, guidance in conceptualizing the study, and critically reviewed the manuscript for final submission.

9. Ethics Committee Approval and Conflict of Interest

There is no need for an ethics committee approval in the prepared article. There is no conflict of interest with any person/institution in the prepared article.

10. Ethical Statement Regarding the Use of Artificial Intelligence

No artificial intelligence-based tools or applications were used in the preparation of this study. The entire content of the study was produced by the author(s) in accordance with scientific research methods and academic ethical principles.

11. References

[1] H. Hoyez, C. Schockaert, J. Rambach, B. Mirbach, and D. Stricker, "Unsupervised image-to-image translation: A review," *Sensors*, vol. 22, no. 21, p. 8540, 2022.
[2] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Honolulu, HI, USA, 2017.
[3] E. U. R. Mohammed, N. R. Soora, and S. W. Mohammed, "A comprehensive literature review on convolutional neural networks," *Computer Science Publications*, 2022.
[4] A. Kamil, and T. Shaikh, "Literature review of generative models for image-to-image translation problems," in *Proc. Int. Conf. Comput. Intell. Knowl. Economy (ICCIKE)*, Dubai, United Arab Emirates, 2019.
[5] M. Mirza, and S. Osindero, "Conditional generative adversarial nets," *arXiv preprint*, arXiv:1411.1784, 2014.
[6] C. Koç, and F. Özyurt, "An examination of synthetic images produced with DCGAN according to the size of data and epoch," *Firat Univ. J. Exp. Comput. Eng.*, vol. 2, no. 1, pp. 32–37, 2023.
[7] I. Goodfellow, J. Pouget-Abadie, M. Mirza, X. Bing, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 27, pp. 2672–2680, 2014.
[8] G. Perarnau, J. Van De Weijer, B. Raducanu, and J. M. Álvarez, "Invertible conditional GANs for image editing," *arXiv preprint*, arXiv:1611.06355, 2016.
[9] A. Odena, C. Olah, and J. Shlens, "Conditional image synthesis with auxiliary classifier GANs," in *Proc. Int. Conf. Mach. Learn. (ICML)*, Sydney, Australia, 2017.
[10] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Med. Image Comput. Comput.-Assist. Interv. – MICCAI 2015*, Munich, Germany, 2015.
[11] Y. Ji, H. Zhang, and Q. M. J. Wu, "Saliency detection via conditional adversarial image-to-image network," *Neurocomputing*, vol. 316, pp. 357–368, 2018.
[12] X. Mao, S. Wang, L. Zheng, and Q. Huang, "Semantic invariant cross-domain image generation with generative adversarial networks," *Neurocomputing*, vol. 293, pp. 55–63, 2018.
[13] Y. Gan, J. Gong, M. Ye, Y. Qian, and K. Liu, "Unpaired cross-domain image translation with augmented auxiliary domain information," *Neurocomputing*, vol. 316, pp. 112–123, 2018.
[14] S. Mo, M. Cho, and J. Shin, "InstaGAN: Instance-aware image-to-image translation," *arXiv preprint*, arXiv:1812.10889, 2018.
[15] Y. Cho, R. Malav, G. Pandey, and A. Kim, "DehazeGAN: Underwater haze image restoration using unpaired image-to-image translation," *IFAC-PapersOnLine*, vol. 52, no. 21, pp. 82–85, 2019.
[16] D. Yang, S. Hong, Y. Jang, T. Zhao, and H. Lee, "Diversity-sensitive conditional generative adversarial networks," *arXiv preprint*, arXiv:1901.09024, 2019.
[17] M.-Y. Liu, X. Huang, A. Mallya, T. Karras, T. Aila, J. Lehtinen, and J. Kautz, "Few-shot unsupervised image-to-image translation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Seoul, South Korea, 2019.
[18] W. Xu, S. Keshmiri, and G. Wang, "Toward learning a unified many-to-many mapping for diverse image translation," *Pattern Recognit.*, vol. 93, pp. 570–580, 2019.
[19] L. Ye, B. Zhang, M. Yang, and W. Lian, "Triple-translation GAN with multi-layer sparse representation for face image synthesis," *Neurocomputing*, vol. 358, pp. 294–308, 2019.
[20] Y. Choi, Y. Uh, J. Yoo, and J.-W. Ha, "StarGAN v2: Diverse image synthesis for multiple domains," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Seattle, WA, USA, 2020.
[21] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Pix2Pix datasets," UC Berkeley, Feb. 9, 2017. [Online]. Available: https://efrosgans.eecs.berkeley.edu/pix2pix/datasets/. [Accessed: Apr. 2, 2024].
[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Las Vegas, NV, USA, 2016.
[23] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

[24] S. Mallat, “Compression,” in *A Wavelet Tour of Signal Processing*, 3rd ed., Boston, MA, USA: Academic Press, 2009, pp. 481–533.