Neural style conversion with Generative models

Adway Kanhere, Aravind Kumar, Deeksha Shama

10 May 2022

***Abstract*—Image translation for style conversion has been an active area of research in deep learning finding a wide variety of applications from medical imaging to speech processing. It aims to find an apt mapping from one signal domain to another, by learning the underlying distributions or by conditioning on given cues like text. Generative Adversarial Networks and their variants are the most popular architecture for this purpose, with CycleGAN being the most simple and intuitive approach. However, GANs are generally much harder to train with unstable convergence guarantee. They also require large computational resources and more representative databases in both domains. In this project, we implement the cycleGAN and show its effectiveness in learning the conversion along with the accompanying challenges. In an attempt to overcome them, the recent DiffusionCLIP model is modified and finetuned to fit the dataset. The models are trained on Wikiarts-Impressionism dataset for paintings with CelebA and Intel Image Classification datasets for real-world photos. We achieve FID score of 5.941 for CycleGANs and 5.535 with DiffusionCLIP, hence proving the superiority of diffusion-based generative models in learning the transformation while being practically efficient.**

***Index Terms*—Image translation, CycleGAN, Diffusion models, Image processing, Deep learning, Generative modeling**

I. Introduction

Image translation is a subset of deep learning problems where the models are trained to learn mapping from one image domain to another. It finds applications in a wide variety of areas including but not limited to art generation [1], [2], image inpainting [3], medical imaging modality conversion (MRI T1 to T2 contrast or CT images) [4] [5], and emotion conversion [6]. This can be solved in a supervised discriminative manner when provided with paired images from source and target domains. However, it is not practically possible to have paired data in all applications. Suppose the subjects in imaging modality conversion problem undergo both CT and MRI under controlled conditions, the images from two scanners need to be fused to simulate simultaneous acquisition without more expensive technology [7]. On the other hand, consider the experiment of generating Monet-style impressionist art from real photos. As demonstrated in the figure 1, the database of target domain here only contains the experiences of the 19th century artists. Monet could never have captured the sky-scrappers or the fashion accessories of the present day world. Hence, it is necessary to employ generative models to learn the mapping between the underlying distributions of the domains.

In this project, the problem of art generation from real-world photos is tackled using deep generative models. This can be broadly conceptualized as building a parameterized model to extract relevant features from the source domain (X) and then manipulating them accordingly to map to the target domain (Y). It can be modelled as an informed encoder-decoder architecture. Generative Adversarial Networks (GANs) are one such state-of-art models for image generation. Meanwhile, the mapped image must still contain features like edges and shapes from original photo while containing distinguishable features of the art style like brush strokes. This can be achieved by ensuring cycle consistency in the architecture as first introduced by Zhu et al in 2017 [8]. CycleGANs consist of two generators with paired discriminators to learn the X to Y as well as Y to X mappings. While CycleGAN is simple and intuitive, training the large network with images from both domains can be complicated. Secondly, they also lack in preserving the highly variable image contents. In an alternative approach, we explore art generation by neural style transfer which informs the model with a representation guided by textual cues. This is called zero-shot image generation which has been very successful with Contrastive Language-Image Pretraining (CLIP) [9]. Diffusion models overcome the second challenge by designing a paired forward-reverse process using a parameterized Markov chain [10].

This report is structured to include the theory of two models:

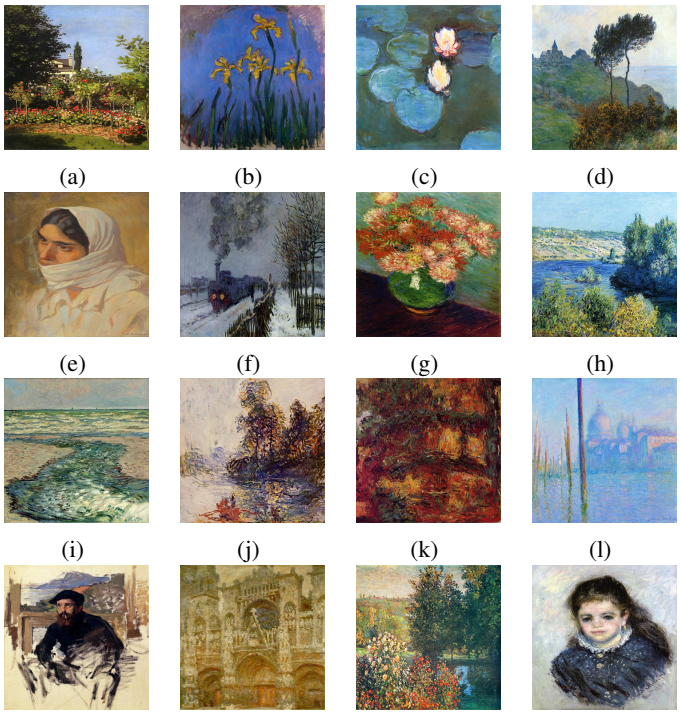


Fig. 1: Sample impressionist paintings by Claude Monet

CycleGAN and DiffusionCLIP and present our findings on implementing the two models for generating impressionist art images from real world photos of landscapes, cityscapes, portraits and miscellaneous objects. We compare the generative ability of models visually and quantitatively and report on their computational efficiency, robustness, and generalizability.

II. Related Work

Generative Adversarial Networks are widely popular generative models that have shown state-of-art performance in several tasks like image generation [11], signal conversion [6], and representation learning for dimensionality reduction [12]. Adversarial loss sets a coupled generative and discriminative model in a two-player game to improve the quality of performance without the need for maximum likelihood estimations. Image translation has been extensively studied with traditional and deep learning based methods. In the paired image problem, a non-parametric approach is employed in Image Analogies method to build a texture model [13].

More recently, CNNs have outperformed such models by training the model to learn the mapping in a supervised fashion [14] which are shown to perform better with perceptual loss [15]. GAN-based image translation such as Pix2Pix learn more acceptable mapping than a vanilla autoencoder [16] for art generation, sketch to photo conversion [2], or from semantic attributes [17]. Other variations include weight sharing, coupled VAE-GAN, predefined metric and feature space guided training. CycleGANs build on this idea for unpaired image tasks by combining two generators to learn mappings to and from both domains. [**?**].

Diffusion probabilistic models are more recent score-based generative models that make use of parameterized Markov chains to infer forward-reverse diffusion process of image generation from noise input [18]. Each step in the process i s a modelled as a Gaussian transition. ALternatively, it can be modified to be a non-markovian process with an alternate sampling process [19]. DDIM makes the whole process faster by setting the noise to zero which makes the sampling process deterministic and inferred through full inversion [20]. CLIP is a pretrained text encoder decoder architecture with losses at multiple resolutions to improve overall performance. CLIP-guided diffusion models that build on U-Net like architecture have shown promising results neural style transfer [10]

III. Methodology

A. CycleGAN

1) Theory and motivation: We aim to develop a generative adversarial network that can learn to translate between the domains of renaissance styled paintings and realistic photographic images. For all the experiments we setup, we assume that there is an underlying relationship between these two domains where each image in one domain is assumed to have different rendering in the other domain. Hence for every set of images belonging to one particular domain X and their Y corresponding rendering in the other domain, we seek to train a generative mapping such that

$$G : X \rightarrow Y$$

for the output

$$\hat{y} = G(x); x \in X$$

Assuming the mapping to be stochastic, the objective that we train on can induce an output distribution over $\hat{y}$ to match the empirical distribution. The main assumption being that the mapping that we generate will translate from domain X to a domain which is identically distributed to the target domain Y. While the above assumption is justified, it is not certain that only a single mapping from the two domains will be generated as there are likely infinite mappings from domain X to Y.

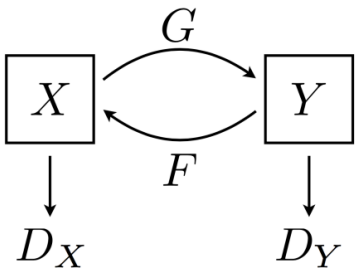


Fig. 2: GAN network with the two mapping functions [21]

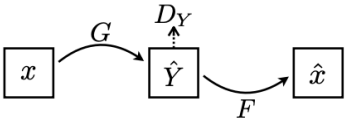


Fig. 3: Forward cycle consistency

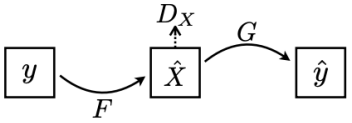


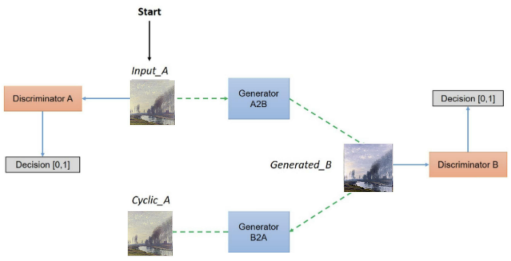
Fig. 4: Backward cycle consistency

While mode collapse is a common problem when tuning GANs, where the generator fails to produce a new set of images each time, this makes the use of GANs for style transfer more difficult as the adversarial objective cannot be successfully optimized in isolation.
Hence, the need to develop a ”cycle consistent” structure was employed by [16]. This concept ensured that translations from one domain to another and back would yield consistent results. This means that if we have a mapping F from one domain to another and G back from the second to first domain, then F and G should be bijections in nature.

In order to maintain this mapping, we have to modify our optimization strategy to accommodate for the bijection mapping by modifying the loss function. This can be done by designing a cyclically consistent loss function that enables the training of mappings in both directions i.e. $F(G(x)) \approx x$ and $G(F(y)) \approx y$. By combining this with the adversarial loss function of the GAN, we can complete our design of the transformation mapping objective between image-to-image domains.

Thus the problem that we propose to solve is an unpaired Image-to-image translation problem where the goal is to build a mapping between two domains. Our work is inspired by previous work in the field of unpaired image-to-image translation and neural style transfer. The idea of image-to-image translation was first proposed by Hertzmann et.al [22] using a non-paramteric model on a single pair of images. Rosales et.al [23] had also proposed a bayesian framework for a prior based patch-based Markov random field computed from a source image and a likelihood term sourced from several style images. In recent years, Liu et.al [24] used a weight sharing strategy to build a representational mapping between two domains which was then extended to using variational auto-encoders and GANs. The idea of using a cycle consistent loss function was also proposed by Zhou et. al [25] using CNNs. Finally we also derive inspiration from the works in neural style transfer but differ in the concept of learning mapping between two different image domains rather than just two different image pairs.

2) Network architecture:: We propose a GAN network architecture based on the work of Johnson et.al [26]. In unpaired image to image translation, we have no pre-defined pathway to learn the necessary mapping to train our network. Hence we need to allow the generator network to map an input image from domain A to the target domain B but by ensuring that we can map the generated output image back to the original image domain. This allows us to essentially learn the mapping between the input image and the generated image.



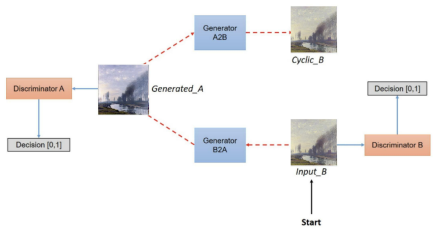


Fig. 5: Complete cycleGAN model. Top: Forward cycle Bottom: Reverse cycle

Thus, each discriminator will receive two inputs, one original image and the other generated image from the generator. The discriminator is tasked to distinguish between the two images and identify if the image is fake or not thus defying the generator. In this process, as the generator learns, it will learn to generate more realistic images that are as close looking to the original images. At the Nash equilibrium, the distribution of the generator and the discriminator becomes the same.

In detail, the GAN generator architecture can be divided into:

- Encoder
- Transformer
- Decoder

together, there are three convolutional layers, 9 residual blocks, two fractionally strided convolutions with stride 1/2 and lastly, a convolution layer to map to RGB. Instance normalization is used for regularization.

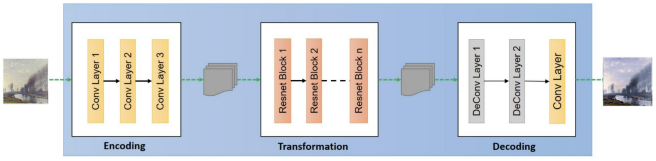


Fig. 6: Generator architecture [27]

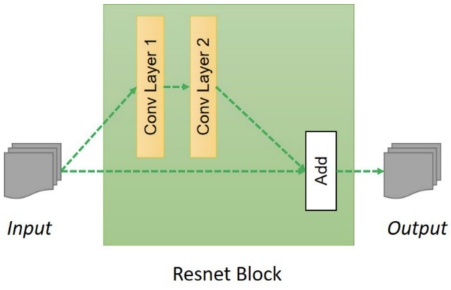


Fig. 7: Intermediate Residual blocks used [27]

Similarly, the discriminator network consists of a PatchGAN model that has a field of 70×70. Such an approach was

employed to reduce the parameter size compared to a full size discriminator.

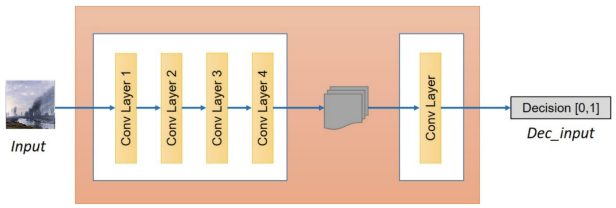


Fig. 8: Discriminator architecture [27]

Also, in order to stabilize the learning, similar to the original cycleGAN paper, we have replaced the negative log-likelihood by a least-square loss. Adam was used as the optimizer and a batch size of 5 was used. The networks were trained from scratch and using a pretrained network later to compare the performance.

3) Loss function definition:: To learn the mapping from domain X to domain Y, we denote the data distribution as $x \sim p_{data}(x)$ and $y \sim p_{data}(y)$
Further, as proposed in the original cycleGAN paper, we use two additional adversarial discriminators. Thus, the final objective contains a combination of two losses, adversarial loss and cycle consistency loss. In detail, each is defined as:
Adversarial loss: The native GAN's adversarial loss is applied to both mapping functions. For example, for the mapping for the generator, $G : X \rightarrow Y$ and it's discriminator D_Y, the objective is defined as:

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

where the generator G tries to generate images that look similar to the target Y domain and the discriminator D_Y does the opposite. The generator aims to minimize this loss while the discriminator tries to maximize this.

$$\min_{G} \max_{D_Y} \mathcal{L}_{\mathrm{GAN}}\left(G, D_Y, X, Y\right)$$

Cycle Consistency loss: Adversarial loss alone cannot guarantee bijection mapping between the input and target domains. While adversarial training can usually produce outputs that match the target distribution, this requires that the mapping are stochastic in nature. If given a large enough compute, any of the learnt mappings can induce an output distribution that matches the target distribution, and a network can map the same set of input photos to any random permutation of images in the target domain. This gives rise to the formulation of the cycle consistency loss given by:

$$\begin{aligned}\mathcal{L}_{\text{cyc}}\left(G, F\right) = & \mathbb{E}_{x \sim p_{\text{data}}(x)}\left[\|F(G(x)) - x\|_1\right] \\ & + \mathbb{E}_{y \sim p_{\text{data}}(y)}\left[\|G(F(y)) - y\|_1\right]\end{aligned}$$

Hence the full objective to optimize now becomes:

$$\begin{aligned}\mathcal{L}\left(G, F, D_X, D_Y\right) = & \mathcal{L}_{\mathrm{GAN}}\left(G, D_Y, X, Y\right) \\ & + \mathcal{L}_{\mathrm{GAN}}\left(F, D_X, Y, X\right) \\ & + \lambda \mathcal{L}_{\text{cyc}}(G, F)\end{aligned}$$

with the focus to solve the following optimization -

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

B. DiffusionCLIP

1) Theory and motivation: GAN inversion methods have recently been combined with Contrastive Language-Picture Pretraining (CLIP), allowing zero-shot image alteration driven by text instructions. However, due to the restricted GAN inversion capabilities, its application to a variety of real-world images remains tough [28]. These methods frequently struggle to reconstruct photos with unique positions, viewpoints, and highly changeable contents when compared to the training data, modify object identity, or produce undesired image artifacts. To address these issues and enable accurate alteration of real images, DiffusionCLIP [29], a text-driven image manipulation method based on diffusion models were recently developed.

Diffusion probabilistic models are a class of latent variable models that is inspired by nonequilibrium thermodynamics for high-quality image synthesis metho. It brings a new class of generative models called Diffusion models [30]. Diffusion models have been shown to achieve image sample quality superior to the current state-of-the-art generative models. This is achieved on unconditional image synthesis by finding a better architecture through a series of ablations.

We explore a CLIP-guided robust image alteration approach based on diffusion models. Through forward diffusion, an input image is first converted to latent noises. If the score function for the reverse diffusion is kept the same as the score function for the forward diffusion, the latent noises can be inverted nearly precisely to the original picture using DDIM. DiffusionCLIP's main idea is to use a CLIP loss to fine-tune the score function in the reverse diffusion process, which controls the properties of the created image based on the text prompts. DiffusionCLIP can thus successfully manipulate images in both the trained and unseen domains and can even translate an image from one domain to another.

The forward process consists of a Markov chain where noise is gradually added to the data when sequentially sampling the latent variables x_t for $t = 1, , T$ [30].

Each step in the forward process is a Gaussian transition:

$$q(x_t|x_{t1}) := N(\sqrt[2]{1-\beta_t}x_t - 1, \beta_t I)$$

where $\{\beta_t\}_{t=0}^{T}$ are fixed or learned variance schedule. The resulting x_t can be shown as:

$$x_t = \sqrt{\alpha_t}x_0 + (1 - \alpha_t)\omega$$

where ω is a normal distribution and $\alpha_t := \prod(1 - \beta_s)$. The reverse process $q(x_t - 1|x_t)$ is also parameterized using another Gaussian transition

$$p_\theta(x_{t1}|x_t) := N(x_t - 1; \mu_\theta(x_t, t), \sigma_\theta(x_t, t)I)$$

$\mu_\theta(x_t, t)$ can be decomposed into the linear combination of x_t and a noise approximation model $\epsilon_\theta(x_t, t)$, which can be learned by optimizing:

$$\min_\theta E_{x_0\ q(x_0),\omega\ N(0,I),t}||\omega - \epsilon_\theta(x_t, t)||_2^2$$

After training $\epsilon_\theta(x_t, t)$, the data is sampled using reverse diffusion:

$$x_t = \frac{1}{\sqrt{1-\beta_t}}(x_t - \frac{\beta_t}{\sqrt{1-\alpha_t}}\epsilon_\theta(x_t, t)) + \sigma_t z$$

where z is sampled from a normal distribution. The DDIM sampling [31] process where noise is set to 0 can be modeled as a deterministic process, enabling full inversion of the latent variables into the original images. This can be considered as an Euler method to solve an ODE:

$$\sqrt{\frac{1}{\alpha_{t-1}}}x_{t-1} - \sqrt{\frac{1}{\alpha_t}}x_t = (\sqrt{\frac{1}{\alpha_{t-1}-1}} - \sqrt{\frac{1}{\alpha_t - 1}})\epsilon_\theta(x_t, t)$$

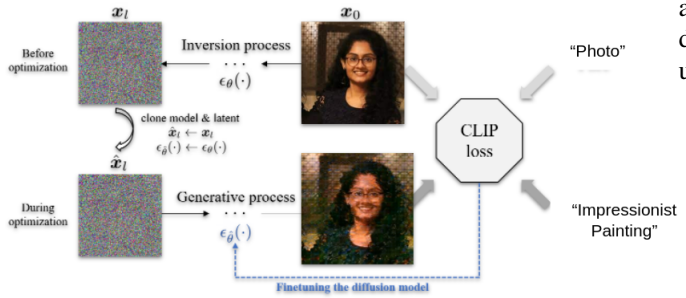


Fig. 9: The input image is first converted to the latent via diffusion models. Then, guided by directional CLIP loss, the diffusion model is fine-tuned

2) Loss definition - CLIP Loss: CLIP [32] was developed as a way to acquire visual concepts quickly with natural language guidance. CLIP uses a text encoder and an image encoder to identify which texts correspond to which photos in the dataset. As a result, for text-driven image alteration, a pretrained CLIP model was used. Two distinct losses have been proposed to effectively extract knowledge from CLIP: a global target loss and a local directional loss. The global CLIP loss aims to reduce the cosine distance between the generated image and a given target text in the CLIP space as follows:

$$L_{global}(x_{gen}, y_{tar}) = D_{CLIP}(x_{gen}, y_{tar})$$

y_{tar} is a target's text description, x_{gen} is the generated picture, and D_{CLIP} returns the cosine distance between their encoded vectors in CLIP space. Local directional loss, on the other hand, is intended to address the problems associated with global CLIP loss, such as poor variety and vulnerability to adversarial attacks. In the CLIP space, the local directional CLIP loss causes the direction between the embeddings of the reference and generated pictures to align [33] with the direction between the embeddings of a pair of reference and target as follows:

$$L_{direction}(x_{gen}, y_{tar}; x_{ref}, y_{ref}) = 1 - \frac{< \Delta I, \Delta T >}{||\Delta I||||\Delta T||}$$

$$\Delta T = E_T(y_{tar}) - E_T(y_{ref}); \Delta I = E_I(x_{gen}) - E_I(x_{ref})$$

CLIP's image and text encoders are E_I and E_T, respectively, while the source domain text and image are y_{ref} and x_{ref}. The altered images guided by the directional CLIP loss are known to be resistant to mode-collapse difficulties because separate images should be generated by aligning the direction of the image representations with the direction of the reference text and the target text. It's also more resistant to adversarial attempts because the perturbation will vary based on the photos.

3) Model architecture and training: Figure 10 depicts the overall flow of the proposed DiffusionCLIP for image translation. It follows a U-Net architecture with multiple attention layers in between. Using a pretrained diffusion model, the input image is first transformed to the latent. The diffusion model at the reverse process is then fine-tuned to generate samples driven by the target text, guided by the CLIP loss. DDIM is used to create deterministic forward-reverse processes.

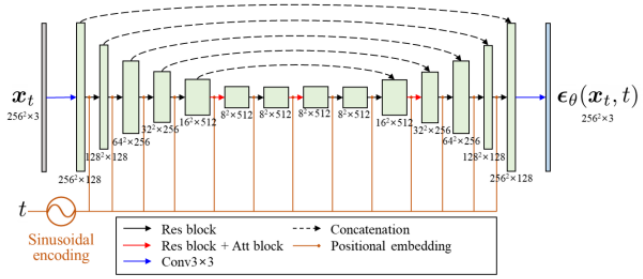


Fig. 10: The shared U-Net architecture across t of the diffusion model that generates 256 × 256 images. The model receives xt and t as inputs and outputs $\epsilon_\theta(xt, t)$.1001[29]

The CLIP loss is a crucial component in the optimization process. Because of the appealing qualities indicated ,we use directional CLIP loss as a guidance among the two types of CLIP losses discussed above [29]. Directional CLIP loss requires a reference text and a target text during training for the text prompt. For example, we may use 'photo' as a reference text and 'monet painting' as a target text to change the expression of a given photo into an impressionist painting.

The majority of existing diffusion models take x_t and t as network inputs. In the CelebA-HQ, DDPM models that have been pre-trained on 256 x 256 pictures which uses the U-Net architecture, which is based on Wide- ResNet. The model

is made up of four parts: an encoder, a middle component, a decoder, and a time embedding part. The 8 x 8 feature is created in the encoder part from the 256 x 256 input picture using 1 input convolution and 5 Res blocks. One Res block is made up of two convolutional blocks: Group normalization and Swish activation, with a residual link as shown in Fig. []. Self-attention blocks are added to the Res block at the 16 x 16 resolution. Three Res blocks make up the middle section, and the second block includes a self-attention block. After the middle section of the decoder, the feature produces an output with the same resolution as the input through 5 Res blocks and 1 output convolution with skip connections from the features. After the Transformer sinusoidal encoding, the diffusion time t is embedded into each Res block in the time embedding section.

IV. Results

A. Dataset information

We form out training image classes using the following:

- Source photo domain: We combine landscape images from Intel Image Classification dataset [34] downloaded from Kaggle and human portrait images from CelebA dataset [35] with a total of 5000 images.
- Target art domain: The Wikipaintings dataset [36], which contains images from the WikiArt website together with their style names. In total, this dataset contains 80,000 images from 25 styles, out of which we only use the Impressionism subset with 2000 images.

We reiterate that we only have unpaired images as shown in the right image in figure 11 Test set is generated by choosing 100 unseen real world photos from the same dataset. All images are resized to (512, 512) with RGB channels, normalized to the range of (-1,1), and injected with a Gaussian noise of mean 0 and standard deviation of 0.05.

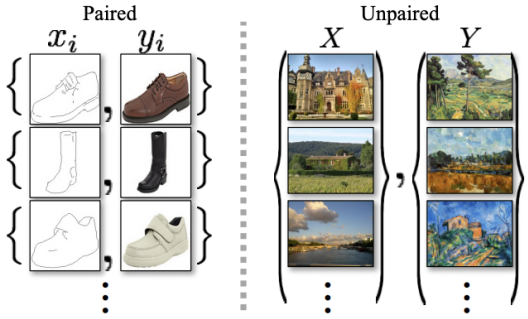


Fig. 11: Paired and unpaired images [21]

B. CycleGAN

Result: We experiment with several hyper-parameters such as regularization (dropout, normalization), size of generator and discriminator networks and try to build a stable GAN model that provides a strong mapping between the two image domains. The final results are shown from a model trained using Adam optimizer with 0.0002 and batch size of 2. We suppose the problem to be an unpaired image-to-image problem thus allowing us to demonstrate whether our model would work in the absence of paired data. The curves of the loss function during training are as shown in figure 15. It is typical of CycleGANs to have spiky training curves as observed here.

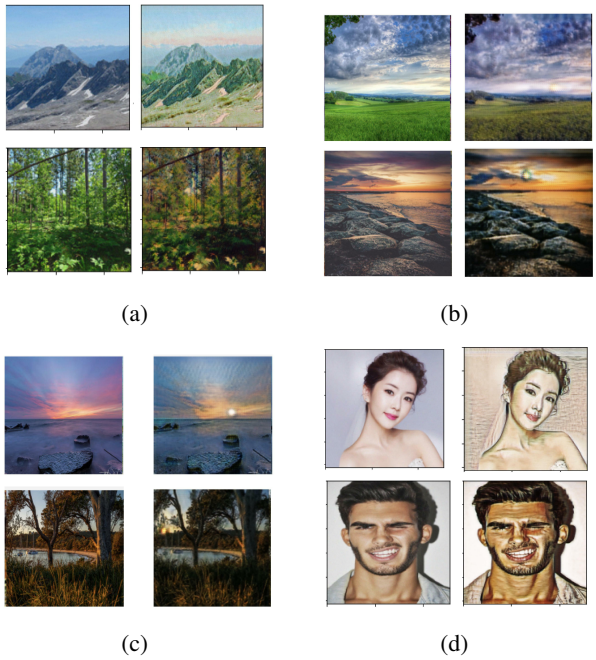


Fig. 13: Reconstructed images *(right)* compared to original source images *(left)* using the GAN model. Outputs are good for simpler images but need multiple passes into the trained model for complex images

C. DiffusionCLIP

The process focuses on zero shot translation from the image domain to the impressioninst art domain. To precompute latents and fine-tune the Diffusion models, about 30+ images in the source domain are used. For the process, the model requires a VRAM of 24 GB+ is required for 256x256 images and this can be brought down to 6GB of VRAM for smaller image sizes. The model was trained on 50 images from the celebA-HQ dataset for 5 iterations with a clip learning rate of 8e-6 and tested on custom face images.

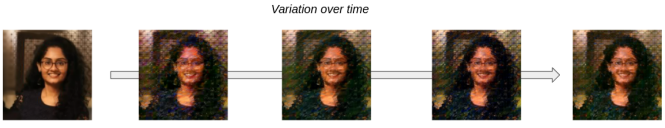
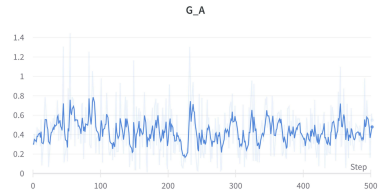
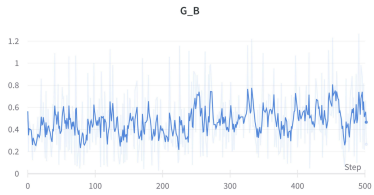


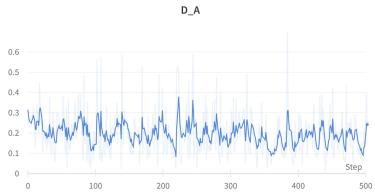
Fig. 14: Diffusion: Variation over time



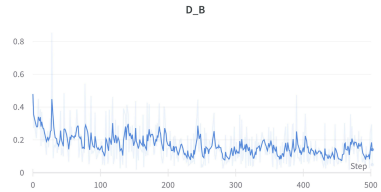
(a) Generator A loss curve



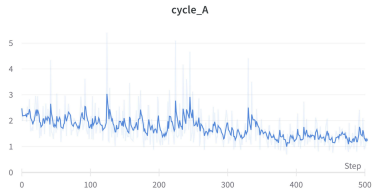
(b) Generator B loss curve



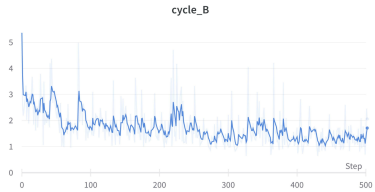
(c) Discriminator A loss curve



(d) Discriminator B loss curve



(e) Loss of cycleGAN A



(f) Loss of cycleGAN B

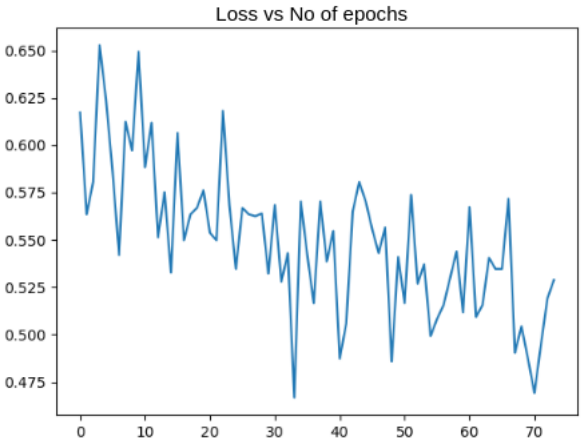


Fig. 15: Diffusion: Training Loss Curve

Fig. 16: Impressionist Images generated using diffusion model *(lower)* compared to original source images *(upper)*

D. Comparison

Due to the lack of paired images, it is not possible to use the general metrics used for image comparison such as Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). Alternatively, Fréchet inception distance (FID) score was first introduced by Heusal et al in 2017 [37] to compare the distribution mappings learnt by the GANs. FID score is given by

$$FID = ||\mu - \mu_w||_2^2 + tr(\Sigma + \Sigma_w - 2(\Sigma^{1/2}\Sigma_w\Sigma^{1/2})^{1/2})$$

where μ, Σ and μ_w, Σ_w are mean and covariances of the two model generated and real world images respectively. The parameters are empirically computed on the activations of final layer of the Inception-v3 model [38]. We use the function in predefined library Pytorch-fid to implement the computation. Specifically we use dimensionality of 64 (first max pooling features) for calculating FID score. We use this score to compare the generated art from 100 test images with a subset of our dataset in impressionism. [39]. If identical, FID score would be zero. Hence, lower the FID score, better is the quality of generated images.

Model	FID score
CycleGAN	5.941
DiffusionCLIP	5.535

TABLE I: FID scores of two models on 100 test images of landscapes and portraits

As reported in the table I, cycleGAN and DiffusionCLIP models generate comparable FID scores, although the latter resulted in visually more appeasing images. We suppose this is because of the Gaussian perturbations introduced in the diffusion process, while the CycleGAN still entails good sharp features which is unlikely to be found in impressionist paintings.

V. Discussion

A. Strengths

With the combined experiments on transforming real world images into art using CycleGAN and CLIP-guided diffusion models, we have showed the effectiveness of deep generative models in learning the mapping from one distribution to another. CycleGAN does this by learning the parameters for both forward and backward processes using two CNN bottleneck autoencoders paired with discriminators. The identity loss and cycle loss ensure that the mapped images belong to the target distribution while still withholding the characteristic features of the original image. This process is simple and intuitive, hence easily implementable even for a beginner. As shown in the results, the generated art images contain the necessary features. The use of instance normalization over batch normalization is a better choice in image translation tasks as it is more important to maintain the same contrast information in the mapped images, more than maintain the same intensities.

On the other hand, CLIP-guided diffusion model not only resulted in more acceptable art images with distinct impressionist features, they were also trained in a more stable fashion. The physics-based diffusion process intuits the forward-backward process with parameterized Gaussian transition. This whole tractable process was much easier to train until convergence. This model also generalized very well outside the training dataset unlike GAN models. The unseen images looked more like paintings irrespective of its structural and textural complexities.

Hence, we compared the state-of-art generative models for the task of image translation. This can be easily extended to other applications like medical imaging modality conversion using fine-tuning on the required dataset. Generative models like this perform better than overly parameterized discriminative models for two reasons: One, the losses are defined to learn the underlying distribution and hence does not need pair wise information. Two, they can easily outperform discriminative models on unseen images of the same distribution for same reasons. Hence, they are more practical and efficient.

B. Weaknesses

Generative models are known to be prone to unstable training process. Particularly, our CycleGAN loss curve was spiky with minute reduce in loss, hence, we suppose that it needs at least 10 times the current training steps i.e., about 50000 iterations taking several days of training which is computationally impossible given the resources at our disposal. This resulted in teh CycleGAN to learn an intermediate mapping between domains. As a solution, we passed the output of the model iteratively back into the model make it more art-like as shown in the figure, but this is not the best strategy in long term. GANs are trained with images from both source and target domain in the cyclic fashion which requires a lot of data to enure the representation from all possible cases. This limits the model to learn within the available data and hence can not generalize to unseen images. As seen in the results, GANs fail in transforming images with higher complexities as it is not very well represented in the target domain.

Diffusion model overcomes many of the challenges faced by CycleGAN, however, presents a new set of issues given its recency within the research area. The tractable training process is faster but still requires a top-end GPU to be trained. It requires about 2 min to generate an art image from the input.

Both these models however need to be trained separately for each genre of art, for example, three separate models for transforming photos to impressionism, pointillism, and cubism. This makes the strategy computationally inefficient in a resource-constrained setting. A possible solution would be to learn multi-way transfers using StarGANs or learning with conditional losses in diffusion model using latent representations. This would require a huge corpus of data from all genre, hence, necessitating a better planning.

VI. Conclusion

In this project, we implemented two state-of-art generative models, namely CycleGAN and DiffusionCLIP for the task of image translation from photos to impressionist art. We established the superiority of diffusion-based models in learning a better transformation without need for large databases in both image domains. They are easy to train to convergence but necessitate long hours of training. Meanwhile, cycleGANs are entitled to more unstable training learning only an intermediate transformation.

In the future, this work can be extended to learn multi-way neural style transfers. It also requires better usage of resources to facilitate faster training. These models can thus be extended to more practical scenarios of medical imaging and emotion conversion.

REFERENCES

[1] K. Wołk, E. Zawadzka-Gosk, and W. Czarnowski, "Deep learning and sub-word-unit approach in written art generation," in *New Knowledge in Information Systems and Technologies*, Á. Rocha, H. Adeli, L. P. Reis, and S. Costanzo, Eds. Cham: Springer International Publishing, 2019, pp. 303–315.

[2] H. Liu, P. N. Michelini, and D. Zhu, "Artsy-gan: A style transfer system with improved quality, diversity and performance," in *2018 24th International Conference on Pattern Recognition (ICPR)*, 2018, pp. 79–84.

[3] X. Zhang, X. Wang, C. Shi, Z. Yan, X. Li, B. Kong, S. Lyu, B. Zhu, J. Lv, Y. Yin, Q. Song, X. Wu, and I. Mumtaz, "De-gan: Domain embedded gan for high quality face image inpainting," *Pattern Recognition*, vol. 124, p. 108415, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0031320321005914

[4] N. Olliverre, G. Yang, G. Slabaugh, C. C. Reyes-Aldasoro, and E. Alonso, "Generating magnetic resonance spectroscopy imaging data of brain tumours from linear, non-linear and deep learning models," in *Simulation and Synthesis in Medical Imaging*, A. Gooya, O. Goksel, I. Oguz, and N. Burgos, Eds. Cham: Springer International Publishing, 2018, pp. 130–138.

[5] G. Kwon, C. Han, and D.-s. Kim, "Generation of 3d brain mri using auto-encoding generative adversarial networks," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, Eds. Cham: Springer International Publishing, 2019, pp. 118–126.

[6] R. Shankar, H.-W. Hsieh, N. Charon, and A. Venkataraman, "Multi-speaker emotion conversion via latent variable regularization and a chained encoder-decoder-predictor network," 2020. [Online]. Available: https://arxiv.org/abs/2007.12937

[7] Y. Xi, J. Zhao, J. R. Bennett, M. R. Stacy, A. J. Sinusas, and G. Wang, "Simultaneous ct-mri reconstruction for constrained imaging geometries using structural coupling and compressive sensing," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 6, pp. 1301–1309, 2016.

[8] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2223–2232.

[9] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[10] G. Kim, T. Kwon, and J. C. Ye, "Diffusionclip: Text-guided diffusion models for robust image manipulation," 2021. [Online]. Available: https://arxiv.org/abs/2110.02711

[11] E. L. Denton, S. Chintala, a. szlam, and R. Fergus, "Deep generative image models using a laplacian pyramid of adversarial networks," in *Advances in Neural Information Processing Systems*, C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, and R. Garnett, Eds., vol. 28. Curran Associates, Inc., 2015. [Online]. Available: https://proceedings.neurips.cc/paper/2015/file/aa169b49b583a2b5af89203c2b78c67c-Paper.pdf

[12] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, X. Chen, and X. Chen, "Improved techniques for training gans," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper/2016/file/8a3363abe792db2d8761d6403605aeb7-Paper.pdf

[13] S. Zelinka and M. Garland, "Mesh modelling with curve analogies," in *ACM SIGGRAPH 2003 Sketches Applications*, ser. SIGGRAPH '03. New York, NY, USA: Association for Computing Machinery, 2003, p. 1. [Online]. Available: https://doi.org/10.1145/965400.965412

[14] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," 2014. [Online]. Available: https://arxiv.org/abs/1411.4038

[15] I. Ananthabhotla, S. Ewert, and J. A. Paradiso, "Towards a perceptual loss: Using a neural network codec approximation as a loss for generative audio models," in *Proceedings of the 27th ACM International Conference on Multimedia*, ser. MM '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 1518–1525. [Online]. Available: https://doi.org/10.1145/3343031.3351148

[16] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[17] Y. Shu, R. Yi, M. Xia, Z. Ye, W. Zhao, Y. Chen, Y.-K. Lai, and Y.-J. Liu, "Gan-based multi-style photo cartoonization," *IEEE Transactions on Visualization and Computer Graphics*, pp. 1–1, 2021.

[18] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," 2020. [Online]. Available: https://arxiv.org/abs/2006.11239

[19] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," 2020. [Online]. Available: https://arxiv.org/abs/2010.02502

[20] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," 2021. [Online]. Available: https://arxiv.org/abs/2105.05233

[21] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 1125–1134.

[22] A. Hertzmann, C. E. Jacobs, N. Oliver, B. Curless, and D. H. Salesin, "Image analogies," in *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, 2001, pp. 327–340.

[23] R. Rosales, K. Achan, and B. J. Frey, "Unsupervised image translation." in *iccv*, 2003, pp. 472–478.

[24] M.-Y. Liu, T. Breuel, and J. Kautz, "Unsupervised image-to-image translation networks," *Advances in neural information processing systems*, vol. 30, 2017.

[25] T. Zhou, P. Krahenbuhl, M. Aubry, Q. Huang, and A. A. Efros, "Learning dense correspondence via 3d-guided cycle consistency," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 117–126.

[26] J. Johnson, A. Alahi, and L. Fei-Fei, "Perceptual losses for real-time style transfer and super-resolution," in *European conference on computer vision*. Springer, 2016, pp. 694–711.

[27] H. Bansal, "https://hardikbansal.github.io/cycleganblog/." [Online]. Available: https://hardikbansal.github.io/cycleganblog/

[28] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *CoRR*, vol. abs/2105.05233, 2021. [Online]. Available: https://arxiv.org/abs/2105.05233

[29] G. Kim and J. C. Ye, "Diffusionclip: Text-guided image manipulation using diffusion models," *CoRR*, vol. abs/2110.02711, 2021. [Online]. Available: https://arxiv.org/abs/2110.02711

[30] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," *CoRR*, vol. abs/2006.11239, 2020. [Online]. Available: https://arxiv.org/abs/2006.11239

[31] A. Nichol and P. Dhariwal, "Improved denoising diffusion probabilistic models," *CoRR*, vol. abs/2102.09672, 2021. [Online]. Available: https://arxiv.org/abs/2102.09672

[32] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, "Learning transferable visual models from natural language supervision," *CoRR*, vol. abs/2103.00020, 2021. [Online]. Available: https://arxiv.org/abs/2103.00020

[33] R. Gal, O. Patashnik, H. Maron, G. Chechik, and D. Cohen-Or, "Stylegan-nada: Clip-guided domain adaptation of image generators," *CoRR*, vol. abs/2108.00946, 2021. [Online]. Available: https://arxiv.org/abs/2108.00946

[34] "Intel image classification." [Online]. Available: https://www.kaggle.com/datasets/puneet6060/intel-image-classification

[35] "Celeba." [Online]. Available: https://mmlab.ie.cuhk.edu.hk/projects/CelebA.html

[36] "Visual art encyclopedia." [Online]. Available: http://www.wikiart.org/

[37] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "Gans trained by a two time-scale update rule converge to a local nash equilibrium," 2017. [Online]. Available: https://arxiv.org/abs/1706.08500

[38] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," 2015. [Online]. Available: https://arxiv.org/abs/1512.00567

[39] M. Seitzer, "pytorch-fid: FID Score for PyTorch," https://github.com/mseitzer/pytorch-fid, August 2020, version 0.2.1.