Exploration on Novel View Synthesis with Generative Models

Yan Zeng, Yifan Qin, Zheng Chen, Bingnan Li, Chongyu Wang, Yucen Peng
{zengyan,qinyf1,chenzheng,libn,wangchy5,pengyc}@shanghaitech.edu.cn

Abstract

Novel view synthesis involves the acquisition of a model's ability to learn 3D structures from 2D images and generate corresponding images based on the acquired structure. It also has broad applications in areas such as 3D reconstruction, Virtual or Augmented Reality. Despite the recent surge in the development of Neural Radiance Field (NeRF) techniques, which employ an implicit expression to learn 3D structure and color, generative models such as diffusion have also demonstrated their capacity for learning 3D structure. In this research, we thoroughly investigate the capabilities of three prominent generative models, namely Variational Autoencoder (VAE), Generative Adversarial Network (GAN), and Denoising Diffusion Probabilistic Model (DDPM), in the context of novel view synthesis. We present an analysis of the strengths and weaknesses of each model based on their respective results.

1. Introduction

The advancement of generative models has revolutionized the field of computer vision and opened up new possibilities for various applications. One such application is novel view synthesis, which involves the generation of images from previously unseen viewpoints based on the learned 3D structure. This capability has significant implications for areas such as 3D reconstruction, virtual reality, and augmented reality.

In recent years, Neural Radiance Field (NeRF) [17] techniques have gained considerable attention for their ability to learn 3D structure and color using implicit representations. However, it is essential to explore other generative models to fully understand their potential in novel view synthesis. In this research project, we delve into the capabilities of three prominent generative models: Variational Autoencoder (VAE), Generative Adversarial Network (GAN), and Denoising Diffusion Probabilistic Model (DDPM).

In contrast to the NeRF training setting, the training of the generative models in our research project follows a different approach. In this setting, the model is provided with an input image along with its corresponding pose, and the objective is to generate the predicted image from a different pose. The general pipeline is shown in Figure 1.

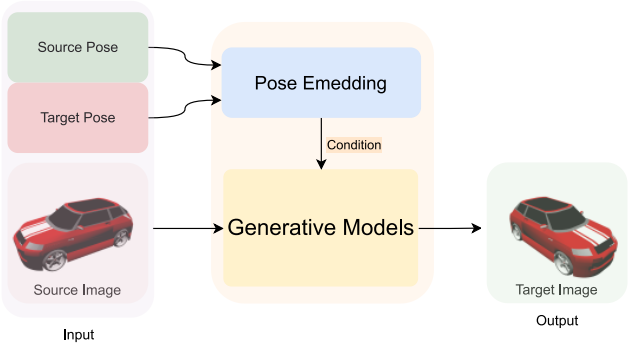

Figure 1. The general pipeline

The VAE is a popular generative model that can learn latent representations of data and generate new samples by sampling from the learned latent space. VAEs have shown promise in various tasks, including image synthesis, and we aim to investigate their effectiveness in novel view synthesis.

GANs have been widely utilized for image-generation tasks and have demonstrated their ability to capture complex distributions. By training a generator network against a discriminator network, GANs can produce realistic and diverse samples. We aim to explore the potential of GANs in generating novel views by leveraging their adversarial training framework.

Another generative model that we consider is the DDPM. DDPM is a recently proposed framework that learns a diffusion process to generate images. By gradually adding noise to an initial image, DDPM can model complex distributions and generate high-quality samples. We investigate the utility of DDPM in the context of novel view synthesis and assess its performance against other generative models.

2. Related Work

Novel view synthesis is a challenging task in computer vision that involves generating new viewpoints of a scene or object given a limited set of observed views. This field has gained significant attention in recent years, and

researchers have proposed various approaches to tackle this problem. In this section, we provide a brief overview of some notable related works in the novel view synthesis field.

Convolutional Neural Network With the advancement of powerful Convolutional Neural Networks (CNN), many researchers [3, 26] have attempted to use it to directly generate the final image in the target view without explicitly estimating its 3D structure. Therefore, view synthesis is achieved through a mapping function between the source view and the target view, which is related to its camera pose [32]. Since there is no need to estimate the 3D model, it is suitable for a wider range of scenarios. However, these solutions find it difficult to separate pose invariant factors from a single view. In order to improve the quality of the results, Zhou et al. [32] predicted the appearance flow instead of synthesizing pixels from scratch. It does not handle areas in the input that do not contain pixels. Park et al. [18] connected another generator to such a network for enhancement, which requires 3D annotations for training.

Multi-View Stereo Multi-view Stereo (MVS) techniques aim to reconstruct 3D scenes from multiple 2D views and can be leveraged for novel view synthesis [5]. Traditional MVS methods use techniques like structure from motion [20] and dense reconstruction [13] to estimate the 3D geometry and texture of the scene. These 3D representations can then be used to synthesize novel views. However, MVS approaches often struggle with handling occlusions and require accurate camera calibration, limiting their applicability to specific scenarios [21].

Neural Scene Representation An alternative approach to synthesizing novel views involves the pursuit of an effective neural representation of the scene. In scenarios where dense capture of scene images is feasible, simple techniques like light field interpolation [7, 12] have proven effective in producing high-fidelity novel views. However, exhaustive sampling of the light field is often impractical, necessitating methods that reconstruct the 3D scene geometry from sparsely-captured images to enable the projection of observed images into new viewpoints. Recent advancements have employed volumetric representations, such as voxel grids [16] or multiplane images [4, 24, 31], which are better suited for gradient-based optimization. While these discrete volumetric representations can be effective for view synthesis, their scalability is limited when it comes to representing large or high-resolution scenes. A recent paradigm shift in view synthesis involves the adoption of coordinate-based neural representations, where a Multi-Layer Perceptron (MLP) maps continuous input 3D coordinates to the geometry and appearance of the scene at that specific location. NeRF [17] has emerged as an effective coordinate-based neural representation for photorealistic view synthesis, modeling a scene as a field of particles that obstruct and emit view-dependent light. In the above context, the generation of images through volume rendering of a single comprehensive 3D representation, commonly referred to as "geometry-aware" models, ensures the inherent 3D consistency during the construction process.

Generative Models Prior to the recent advancements in the field [17, 23], the prevailing state-of-the-art methods for novel view synthesis mainly relied on generative models [25]. In contemporary scholarly literature, there has been a growing interest in geometry-free methodologies, which refer to techniques that lack explicit geometric inductive biases, such as those employed in volume rendering. Among these, [29] employs a deformable conditional VAE architecture to accomplish view synthesis in unpaired data, thus introducing a novel approach to this task and [30] introduces a novel architecture for enhancing U-Net performance, achieved through iterative source-to-target deformation and consideration of image flow. Also, due to the recent advancements in DDPM [15, 27], the task of novel view synthesis has attained competitive outcomes that are comparable to those achieved by 3D-aware methods in the "few-shot" scenario.

3. Dataset

We benchmark this study on the SRN ShapeNet dataset [23], which is a widely recognized and extensively utilized dataset in the field of 3D shape analysis and reconstruction. It serves as a fundamental benchmark dataset, supporting research and development in various aspects of shape understanding and synthesis. The dataset is a product of the ShapeNet project, which aims to create a vast collection of 3D models representing a diverse range of object categories. These categories encompass everyday objects like chairs, tables, cars, airplanes, and many more, providing a comprehensive and versatile dataset for researchers. In our study, the primary focus lies on utilizing the cars category [2] within the ShapeNet dataset. This particular category comprises a total of 2.5k instances, and for each instance, we gather 50 observations at a resolution of 128 × 128 pixels. To generate a diverse set of viewpoints, camera poses are randomly generated on a sphere, with the object located at the origin. These camera poses are then saved in the form of rotation and translation matrices, allowing for precise specification of the viewpoint for each observation.

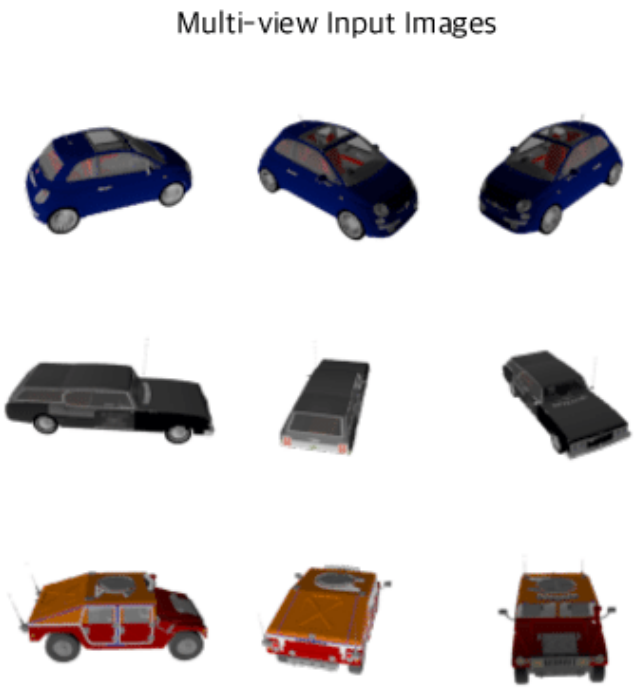

Figure 2. Cars in SRN ShapeNet dataset

4. Our experiment

4.1. VAE

The first generative network that we have considered is the Variational Autoencoder (VAE). VAE is one of the simplest generative models. Its generative ability is limited, so we can see the result of VAE as kind of baseline.

4.1.1 VAE Introduction

The VAE has a similar structure to the Autoencoder (AE), where the observation x is encoded into a latent code z using an encoder, and then the latent code z is decoded back into the observation $\hat{x}$ using a decoder. While both architectures share similarities, the AE is primarily focused on representation learning, where the network is trained to capture the deterministic relationship between the observation x and the latent variable z. On the other hand, the VAE utilizes probabilistic modeling to capture the latent variable z.

The VAE employs a latent variable model that can represent complex data distributions and conditional dependencies. Unlike the AE, which directly maps the observation x to the latent variable z, the VAE's encoder maps the observation x to the mean μ and variance σ of the latent variable z. By sampling from the predicted mean μ and variance, we obtain a latent vector. This sampled latent vector is then passed to the decoder, which maps it to a generated observation $\hat{x}$.

The loss function of the VAE consists of two terms: the regularization term and the reconstruction term. The regularization term is determined by the Kullback-Leibler (KL) divergence between the distribution of the latent variable and the prior distribution. On the other hand, the reconstruction term is approximated.

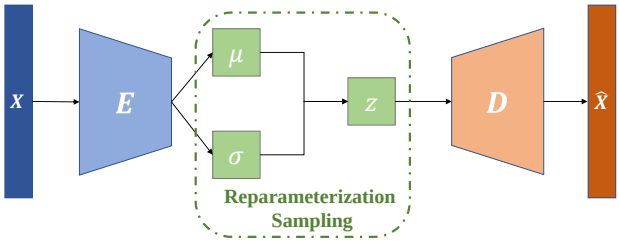

Figure 3. Traditional VAE pipeline.

The task we aim to tackle is novel view synthesis, which involves generating an image from a given pose. In this task, the input consists of an image and its corresponding pose. The goal is to reconstruct an image from a different pose using a neural network. Originally, the generation process of the VAE is unconstrained and relies on sampling in the latent space. To adapt the VAE for novel view synthesis, where maintaining 3D consistency is crucial, additional conditions are imposed on the VAE. These conditions specifically involve incorporating the poses from the images into the network. With the help of the integration, the network is able to generate images that adhere to the desired 3D consistency principles.

4.1.2 Methodology

As mentioned in the related works, when conditions are integrated into the VAE structure, most of the structure also fuses other networks to improve the whole performance, including the UNet structure [30] and GAN-based structure [1, 29]. Here we want to know the capability of VAE alone when it is integrated with the condition information, so only VAE is used and no other networks are fused with VAE.

There are several ways to incorporate pose information into the VAE. One approach is to integrate pose information both in the encoder stage and the decoder stage. However, through experiments, we observed that during training and inference, the generated image consistently resembled the input image. This indicates that the model may not fully utilize the pose information we intend to generate, or it may have overfit on the pose information of the given image. To address this issue and enable the model to generate images

of novel views, we modified the VAE by removing the pose information at the encoder stage and including it only at the decoder stage. During VAE training, we have access to the input image and its corresponding pose, as well as the pose of the target image we want to generate.

However, when considering either the encoder stage or the decoder stage alone, we can only utilize pose information from one of the two images. In order to fully leverage the pose information from both images, we designed a branching mechanism in the decoder stage. This branching mechanism incorporates shared weights in the network, ensuring that the decoder branches can benefit from each other's learning. One branch takes the pose information of the input image and aims to reconstruct the input image that was passed to the network. The other branch takes the pose information of the target image we want to generate and focuses on generating the desired image. By sharing weights between the branches, the network can effectively utilize the pose information from both images, enhancing its capability to generate images while considering the desired pose.

The branching mechanism in the decoder D is based on our hypothesis that the encoder should capture the object information which is independent of the pose information. the disentangled object information is passed to the decoder as input, with the integration of the pose information, the decoder can generate the correct image of the required pose of the object.

This modification allows our model to overcome the limitations of solely relying on the desired image's pose and keep the fidelity between the reconstructed image and the input image. The complete pipeline of our model, including the shared weight branching mechanism in the decoder, will be described in detail below.

4.1.3 Pipeline

Given the training pairs in the training set, which can be represented as $\{P_i\}_{i=1}^{N}$, where the P_i can be represented as the set composed of the input image X_{in}^i, the required generation X_{gen}^i, the pose of the input image and the required generation image, ϕ_{in}^i and ϕ_{gen}^i. So we have $P_i = \{X_{in}^i, X_{gen}^i, \phi_{in}^i, \phi_{gen}^i\}$. And the pipeline of our VAE with conditioning is illustrated in the image Figure 4.

As shown in the pipeline, the input image X_{in}^i is passed to the encoder E, the mean μ and variance σ are generated by the encoder E. With the reparameterization trick, a latent variable z is generated and passed to the two branches in the decoder D.

In one branch of the decoder D, which can be seen as the decoder for reconstruction D_{rec}, the pose of the input image ϕ_{in}^i is imposed on the decoder and the result of this branch of decoder D_{rec} should be the reconstruction of the input image $\hat{X}_{rec}^i$. In the other branch of the decoder D,

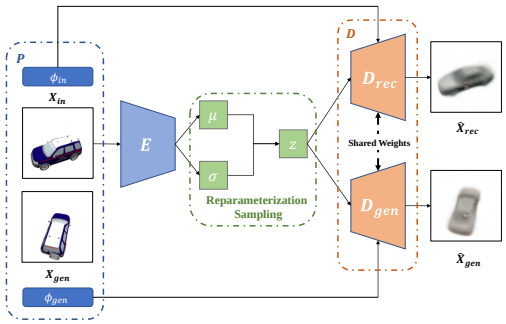

Figure 4. Our VAE with conditioning

which can be seen as the decoder for generation D_{gen}, the pose of the input image ϕ_{gen}^i is imposed on the decoder and the result of this branch of decoder D_{gen} should be the generation of the required image $\hat{X}_{gen}^i$.

4.1.4 Training Loss

The training loss that we have used to train the VAE with conditioning is the image loss and the KL divergence. The image loss is given by the L2 distance between the pixels on the image, which is given by the Equation 1. Because of the branching mechanism in the decoder D, we have two images, the reconstruction image $\hat{X}_{rec}$ and the generated image $\hat{X}_{gen}$. The image loss between the reconstructed image $\hat{X}_{rec}$ and the input image X_{in} is calculated, which is denoted as L_{rec}^{img}. The image loss between the generated image $\hat{X}_{gen}$ and the required image X_{gen} is calculated, which is denoted as L_{gen}^{img}.

$$L^{img} = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{m}(X_{ij} - \hat{X}_{ij})^2 \tag{1}$$

We also see the KL divergence as the loss, which can regularize the distribution of the latent variable z, which is given by the Equation 2.

$$L^{kld} = D_{KL}\left(q_\theta\left(z|x\right)\|p(z)\right) \tag{2}$$

where we want to push the approximate posterior close to the prior distribution.

So the loss in the training of the VAE with conditioning is given by the summation of three parts, image loss of reconstruction L_{rec}^{img}, image loss of generation L_{gen}^{img}, and the KL Divergence L^{kld}. The equation of the whole loss is shown in 3.

$$L = L_{rec}^{img} + L_{gen}^{img} + L^{kld} \tag{3}$$

4.1.5 Results

As shown in the image Figure 5, it is worth noting that the blurriness and lack of clarity in the images generated by the VAE persist even when generating novel views with conditions. This suggests that the VAE's inherent limitations in capturing intricate details and fine textures continue to manifest in the synthesized images.

Figure 5. Result from VAE generation

Moreover, upon closer examination of the image presented in Figure 6, it becomes evident that the VAE structure primarily grasps the overall shape and general characteristics of the car class it was trained on. However, it struggles to effectively capture and reproduce vehicles with distinctive or specialized shapes, such as the Formula One car depicted in the image. This highlights the VAE's inability to learn and generate accurate representations of objects that deviate significantly from the prototypical examples it has been exposed to during training.

These findings shed light on the challenges associated with using VAEs for image generation tasks that require high levels of detail and specific object variations. The blurred and generalized nature of the generated images, combined with the limited capacity to capture unique shape attributes, may restrict the VAE's applicability in domains where precise visual fidelity and fine-grained distinctions are crucial.

In addition to the original configuration of the VAE with conditioning, we conducted further experiments by augmenting the encoder and decoder components of the VAE. Our objective was to assess whether an expanded VAE architecture could more effectively capture the intricate structure of the car class and yield images with enhanced levels of detail. Regrettably, the results obtained from this endeavor, as illustrated in Figure 7, demonstrated no discernible distinction between the generated images produced

Figure 6. F1 car from VAE generation

by the enlarged VAE and those generated by the VAE with the original structure, as depicted in Figure 5. This disappointing outcome indicates that the scale or dimensions of the VAE architecture do not significantly influence the level of detail and visual clarity manifested in the generated images.

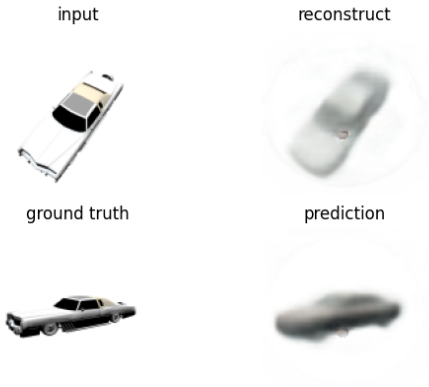

Figure 7. Result from enlarged VAE

The finding suggests that simply enlarging the VAE's encoder and decoder components does not inherently facilitate the acquisition and representation of finer nuances within the generated images. Despite the augmented capacity of the VAE, the inherent limitations of the model in capturing intricate features and high-resolution textures persist. This observation underscores the notion that solely increasing the size of the VAE structure may not be a viable approach for achieving the desired improvements in image generation quality. Further investigations and alternative methodologies should be explored to overcome the challenges as-

sociated with capturing fine-grained details and enhancing clarity in the generated images. The metric evaluation of the different structures is shown in Table 1.

Models	PSNR ↑	SSIM↑	MSE↓
Original structure	18.3168	**0.9852**	1141.1855
enlarged encoder&decoder	**18.6180**	0.9849	**1032.6435**

Table 1. Different structure of our VAE with condition

4.2. GAN

4.2.1 Motivation

Generative Adversarial Networks (GAN) is one of the most popular generation task algorithm at present, proposed by [6]. It consists of two neural networks: a generator network and a discriminator network.

The generator network is responsible for generating data, while the discriminator one is responsible for determining whether the data is real or false (synthetic). In the training process, the generator tries to generate real pictures to deceive the discriminator, and the discriminator tries to distinguish the pictures generated by the generator from the real ones. They constitute a dynamic game process, and the relationship between the two forms a confrontation.

The biggest advantage of GAN is that it is unsupervised, which means there is no need for labeled data to train the network. At the same time, in theory, GAN can train any kind of generator and it can have better model data distribution (sharper, clearer images). So, we can choose the GAN model to realize the novel view synthesis.

4.2.2 Overview of the network

Our network is an encoder-decoder based generative adversarial network and the whole structure is shown in the Figure 8 which is motivated from the VI-GAN [28]. The general idea is that any 2D image is a projection of the 3D world. Therefore, if features with invariant viewpoints can be found, which are important intrinsic properties of the 3D world, they can be used to generate a new target view.

4.2.3 Encoder

The input of the encoder E is the 2D image(I_A) and the camera pose (P_A). Through the encoder, view-independent features F_A will be extracted.

$$F_A = E(I_A \oplus P_A) \tag{4}$$

The encoder is a coordconv network [14]. The structure of CoordConv is shown in the Figure 9: compared to traditional convolution, CoordConv adds two channels after the input feature map. One representing the x coordinate and the other representing the y coordinate, and the rest is the same as normal convolution process. By modifying traditional convolution, which is characterised by translation invariance, coordconv realizes the perception of spatial information, which is essential for multi-perspective generation.

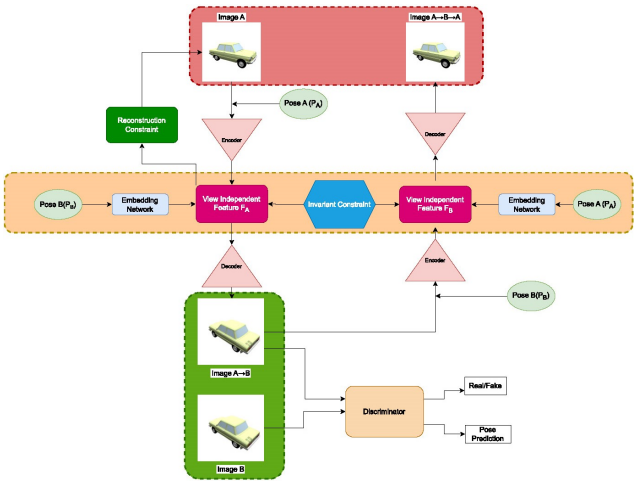

Figure 8. Overall structure of our GAN network.

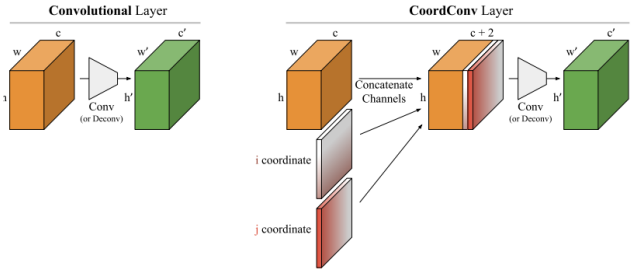

Figure 9. Comparison of 2D convolutional and CoordConv layers [14].

4.2.4 Decoder

The decoder D takes the view independent feature F_A and the target camera pose (P_B), and it generates the target image ($I_{A\to B}$). More specifically, an embedding network M_D is used to accommodate the channel numbers of P_B and F_A.

$$I_{A\to B} = D(F_A \oplus M_D(P_B)) \tag{5}$$

The adaptive instance normalization (AdaIN) [9] is used in the decoder. It aligns channel-wise mean and variance of the input (x) and the style (y). Meanwhile, the mean and variance of the instance normalization layer are inferred by the target pose P_B instead of the feature map.

$$AdaIN(x, y) = \sigma(y)(\frac{x - \mu(x)}{\sigma(x)}) + \mu(y) \tag{6}$$

where $\mu(x)$ is the mean of x and $\sigma(x)$ is the variance of x.

This makes it easier for objects with similar attitude sharing characteristic statistics to present the result of the target attitude for the decoder.

4.2.5 Loss

4.2.5.1 View-independent Loss

To ensure that the model can effectively learn view-independent features with specfic pose, the model utilizes a view independent loss for invariant constraint. We define F_A as the latent feature of I_A with camera pose A. Next, we define F_B as the latent feature of $I_{A\to B}$ with camera pose B. We desire A and B to be more similar, and when they are equal, we can consider them as view-independent. Next is the equation representation:

$$\mathcal{L}_{VI} = \mathbb{E}(||F_A - F_B||) \tag{7}$$

4.2.5.2 Image-reconstruction Loss

GAN has three kinds of image-reconstruction losses. The first term of the image-reconstruction loss is derived from generating the target view I_B. To ensure the accuracy of the synthesized view, we combine pixel-level and perceptual loss to effectively map latent features back to the image space.

$$\begin{aligned} \mathcal{L}_{pixel} &= \mathbb{E}(||I_{A\to B} - I_B||), \\ \mathcal{L}_{per} &= \mathbb{E}((\mathcal{V}(I_{A\to B}) - \mathcal{V}(I_B))^2). \end{aligned} \tag{8}$$

$\mathcal{L}_{pixel}$ is the pixel-level loss [10] and $\mathcal{V}(I_B)$ is the perspective loss [11]. $\mathcal{V}$ represents feature extraction from the VGG16 [22] network.

We aim for the decoder to have the ability to reconstruct the original appearance with given camera parameters. Therefore, this GAN introduces a reconstruction constraint and its corresponding reconstruction loss.

$$\mathcal{L}_{rec} = \mathbb{E}(||I_A - I_{A\to A}||) \tag{9}$$

$I_{A\to A} = D(F_A \oplus M_D(P_A))$ is the reconstruction of image A with camera pose P_A.

To further enhance the understanding of the generated images by the GAN, we employ a cycle constraint that enables the model to reconstruct the images ($I_{A\to B\to A}$).

$$\begin{aligned} \mathcal{L}_{cycle\ pixel} &= \mathbb{E}(||I_{A\to B\to A} - I_A||), \\ \mathcal{L}_{cycle\ per} &= \mathbb{E}((\mathcal{V}(I_{A\to B\to A}) - \mathcal{V}(I_A))^2), \end{aligned} \tag{10}$$

where $I_{A\to B\to A} = D(E(I_{A\to B} \oplus P_B) \oplus M_D(P_A))$.

4.2.5.3 Pose Prediction Loss

To better capture pose information, this model incorporates an additional discriminator $\mathcal{D}q$ that specifically analyzes the pose of the generated images.

$$\begin{aligned} \mathcal{L}_{GAN_{dis}} &= \mathbb{E}((\mathcal{D}q(I_B) - P_B)^2), \\ \mathcal{L}_{GAN_{gen}} &= \mathbb{E}((\mathcal{D}q(I_{A\to B}) - P_B)^2), \end{aligned} \tag{11}$$

where $\mathcal{D}q(\mathcal{X})$ is the output of this discriminator with input $\mathcal{X}$. The two pose prediction losses serve as constraints for the pose in both the generation and discrimination processes.

4.2.6 Result

From the following picture 10, we can find that compared to the real perspective image, the designated perspective image generated by the network has a similar perspective to the original image, and the approximate shape of the object is preserved. However, the generated images have obvious flaws, such as the significant loss of details in the original object and the fact that we can see from the last two images that all the colors of the original object have been lost.

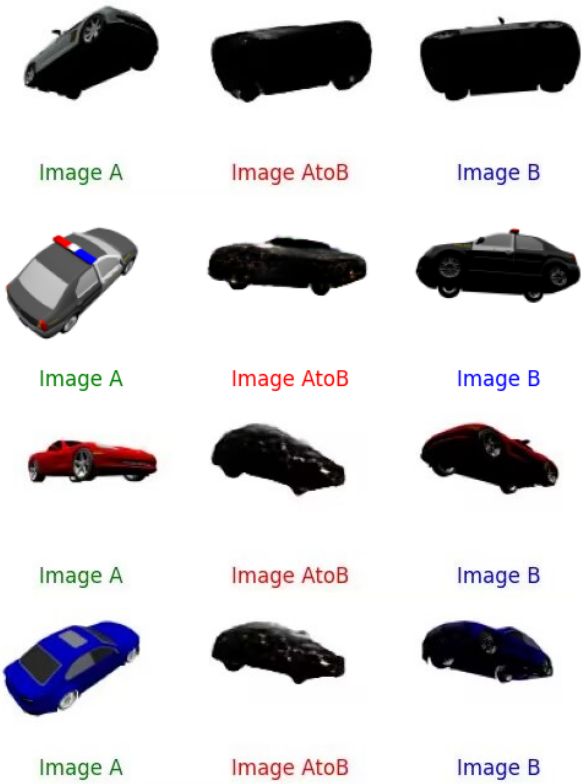

Figure 10. The output of our GAN model

4.3. Diffusion Model

4.3.1 Research motivation

The diffusion model, as demonstrated in the notable work of [8], has already exhibited its remarkable generation capabilities and a remarkable extent of realism. This model surpasses the expressive capacity of Variational Autoencoders (VAEs) while being comparatively easier to train than the two-player model Generative Adversarial Networks (GANs). Given its exceptional performance, the diffusion model holds immense promise for the field of novel view synthesis.

In this section, we delve into the exploration of the diffusion model based on the groundbreaking research conducted by [27]. Our objective is to reproduce their results while also experimenting with two distinct loss functions designed to ensure color consistency. By rigorously investigating the diffusion model and introducing these novel approaches, we aim to further enhance its capabilities and contribute to the advancements in novel view synthesis techniques.

4.3.2 Preliminary: Diffusion model

Different from the conventional diffusion model first proposed by [8], we follows the work of [19] by using signal-to-noise-ratio to represent the level of noise added into the clean image. More specifically, we replaced the series of hyperparameters α_t and β_t with a series of signal-to-noise-ratio λ. Thus, the forward process of diffusion model can be express as follows:

$$q(z^\lambda|x) := \mathcal{N}(z^\lambda; \delta(\lambda)^{\frac{1}{2}}x, \delta(-\lambda)I) \tag{12}$$

$$z^\lambda = \delta(\lambda)^{\frac{1}{2}}x + \delta(-\lambda)^{\frac{1}{2}}\epsilon, \quad \epsilon \sim \mathcal{N}(0, I) \tag{13}$$

where z is the image bound with noise, x is the original clean image and λ is the signal-to-noise-ratio controlling the extent of noise.

For the backward process, the diffusion model is based on a set of known views $\mathcal{X} = \{x_1, \cdots, x_k\}$ and poses $\mathcal{P} = \{p_1, \cdots, p_k\}$ (for novel view synthesis with only one image, the cardinality of $\mathcal{X}$ and $\mathcal{P}$ should both be 1). Then for each step t, there is a corresponding signal-to-noise-ratio λ_t and the backward process can be written as follows:

$$\hat{x}_{k+1} = \frac{1}{\delta(\lambda_t)^{\frac{1}{2}}}(z_{k+1}^{\lambda_t} - \delta(-\lambda_t)^{\frac{1}{2}}\epsilon_\theta(z_{k+1}^{\lambda_t}, x_i, \lambda_t, p_i, p_{k+1})) \tag{14}$$

$$z_{k+1}^{\lambda_{t-1}} \sim q(z_{k+1}^{\lambda_{t-1}}|z_{k+1}^{\lambda+t}, \hat{x}_{k+1}) \tag{15}$$

where i is uniformly sampled from 1 to k which means for each step, the backward process is based on only one random known view, then $z_{k+1}^{\lambda_t}$ is the noisy image of x_{k+1} at time step t with signal-to-noise-ratio λ_t. Crucially, $\epsilon_\theta(z_{k+1}^{\lambda_t}, x_i, \lambda_t, p_i, p_{k+1})$ represents the network that predicts the noise.

Then, as mentioned in [8], the loss function of diffusion model is derived by ELBO which has the following form:

$$L = \mathbb{E}_{q(x_1,x_2)}\mathbb{E}_{\lambda,\epsilon}||\epsilon(z_2^\lambda, x_1, \lambda, p_1, p_2) - \epsilon||_2^2 \tag{16}$$

4.3.3 Network Architecture

The core of the diffusion model is the UNet-like model that predicts the noise that is added into x. Instead of utilizing the vanilla UNet implementation, we leverage a modification version named XUNet proposed by [27]. The pipeline of training process is shown in Figure 11 and corresponding architecture of XUNet is shown in Figure 12.

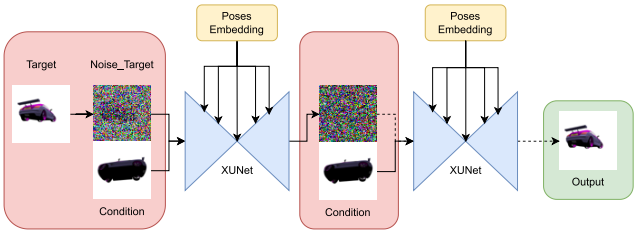

Figure 11. The pipeline of diffusion model for training process

the Network architecture of unet in diffusion is shown below.

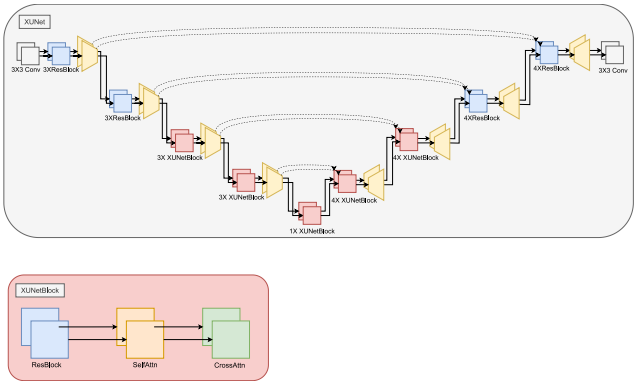

Figure 12. The detail architecture of XUNet and XUNet Block

The same UNet weight is shared between the clean condition view and the noisy target view in both input frames. The XUNet utilizes cross-attention layer to bind information between the input and output views which has been proved to be efficient than the vanilla implementation [27].

4.3.4 Experimental Results

In this section, we will experiment with diffusion model and the XUNet model. Given that UNet has also been widely

used for image generation since its classical encoder-decoder structure, we modified the XUNet from predicting noises to generating images directly based on the poses information and known pose images. The corresponding generation results are shown in Figure 13 and Figure 14.

Figure 13. Results of diffusion model, the three rows represents input, groundtruth and prediction respectively

Figure 14. Results of XUNet model, the three rows represents input, groundtruth and prediction respectively

Based on our findings, the XUNet model demonstrates the ability to grasp a general association between image view and pose information. However, its output image tends to exhibit a blurry representation. On the other hand, the diffusion model showcases remarkable aptitude in comprehending the intricate relationship between image view and pose information. Notably, the output image produced by the diffusion model possesses enhanced clarity and realism, surpassing that of the XUNet model. These results emphasize the superior capability of the diffusion model in capturing and representing the nuances of the image view and pose connection, thereby yielding more visually appealing and authentic outputs.

The metric evaluation results is shown in Table 2.Upon comparing the results presented in Table 1, it becomes evident that the diffusion model outperforms the VAE in terms of Structural Similarity Index (SSIM). This observation indicates that the diffusion model has the ability to generate sharper and clearer ship images. However, it is worth noting that the diffusion model achieves slightly lower values for Peak Signal-to-Noise Ratio (PSNR) and Mean Squared Error (MSE) when compared to the VAE. We attribute this discrepancy to a potential mismatch between the colors of the real car body and those generated by the diffusion model, we will discuss this in the following section.

4.3.5 Limitation of Diffusion Model

Although diffusion model captured the complex relationship between different views, we discovered during our experiments that the diffusion model is struggled to recover the real color from noise. We hypothesised that it's difficult for diffusion model to learn the proper color range since we can not bound the color range simply from predicted noise or even simply constrain the model output like VAE and GAN. During the whole training process, we can barely manipulate the generated image by any mean since diffusion model takes hundreds of step to iteratively generate the result. To prove our assumption, we tried to add constrains between $\hat{x}_{k+1}^{\lambda_t}$ in Equtation 14 and x_{k+1} with three ways:

Models	PSNR ↑	SSIM↑	MSE↓
Diffusion Model	**17.3431**	**0.9872**	**1198.8768**
XUNet only	7.3652	0.8843	2537.1254

Table 2. Final results on SRN cars

- Constrain the mean values of three color channels to be the same between $\hat{x}_{k+1}^{\lambda_t}$ and x_{k+1}.
- Use SSIM loss to encourage $\hat{x}_{k+1}^{\lambda_t}$ as similar as x_{k+1}.
- Use random hue-adjust for input image to improve the model's sensitivity to color.

Unfortunately, all of the methods mentioned above failed to generate even "real" image. Shown in Figure 15, we found that the constrain imposed on $\hat{x}_{k+1}^{\lambda_t}$ gives extremely bad result, and the ssim loss severely damaged the normal training process of diffusion model.

Figure 15. Results of diffusion model by adding ssim loss, the three rows represents input, groundtruth and prediction respectively

5. Conclusion

The aim of this project was to investigate various generative models with regard to their efficacy in the task of novel view synthesis. The examined models include VAE, GAN, and Diffusion models. As anticipated, the quality and clarity of the generated results demonstrated a discernible improvement as we progressed from VAEs to GANs and ultimately to Diffusion models. Notably, all three models exhibited internal consistency, as evidenced by the absence of sudden color shifts within the generated images. However, a significant inconsistency in color was observed when comparing the generated images to the corresponding ground truth image. Consequently, further investigation

and explore alternative methodologies can be undertaken in order to address this prevalent challenge of discordant coloration between the generated results and the ground truth image. By doing so, we can strive to enhance the fidelity and realism of the synthesized views with generative models in novel view synthesis tasks.

References

[1] Jianmin Bao, Dong Chen, Fang Wen, Houqiang Li, and Gang Hua. Cvae-gan: fine-grained image generation through asymmetric training. In *Proceedings of the IEEE international conference on computer vision*, pages 2745–2754, 2017. 3

[2] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 2

[3] Alexey Dosovitskiy, Jost Tobias Springenberg, and Thomas Brox. Learning to generate chairs with convolutional neural networks. *CoRR*, abs/1411.5928, 2014. 2

[4] John Flynn, Michael Broxton, Paul E. Debevec, Matthew DuVall, Graham Fyffe, Ryan S. Overbeck, Noah Snavely, and Richard Tucker. Deepview: View synthesis with learned gradient descent. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2362–2371, 2019. 2

[5] Michael Goesele, Brian Curless, and Steven M. Seitz. Multi-view stereo revisited. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2:2402–2409, 2006. 2

[6] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014. 6

[7] Steven J. Gortler, Radek Grzeszczuk, Richard Szeliski, and Michael F. Cohen. The lumigraph. *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996. 2

[8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems*, 33:6840–6851, 2020. 8

[9] Xun Huang and Serge J. Belongie. Arbitrary style transfer in real-time with adaptive instance normalization. *CoRR*, abs/1703.06868, 2017. 6

[10] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 694–711. Springer, 2016. 7

[11] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. Autoencoding beyond pixels using a learned similarity metric. In *International conference on machine learning*, pages 1558–1566. PMLR, 2016. 7

[12] Marc Levoy and Pat Hanrahan. Light field rendering. *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, 1996. 2

[13] Yangming Li. 3 dimensional dense reconstruction: A review of algorithms and dataset. *ArXiv*, abs/2304.09371, 2023. 2

[14] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. *CoRR*, abs/1807.03247, 2018. 6

[15] Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object, 2023. 2

[16] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *arXiv preprint arXiv:1906.07751*, 2019. 2

[17] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 1, 2

[18] Eunbyung Park, Jimei Yang, Ersin Yumer, Duygu Ceylan, and Alexander C. Berg. Transformation-grounded image generation network for novel 3d view synthesis. *CoRR*, abs/1703.02921, 2017. 2

[19] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022. 8

[20] Johannes L. Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016. 2

[21] Steven M. Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 1:519–528, 2006. 2

[22] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 7

[23] Vincent Sitzmann, Michael Zollhöfer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *Advances in Neural Information Processing Systems*, 32, 2019. 2

[24] Pratul P. Srinivasan, Richard Tucker, Jonathan T. Barron, Ravi Ramamoorthi, Ren Ng, and Noah Snavely. Pushing the boundaries of view extrapolation with multiplane images. *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 175–184, 2019. 2

[25] Shao-Hua Sun, Minyoung Huh, Yuan-Hong Liao, Ning Zhang, and Joseph J. Lim. Multi-view to novel view: Synthesizing novel views with self-learned confidence. In *European Conference on Computer Vision*, 2018. 2

[26] Maxim Tatarchenko, Alexey Dosovitskiy, and Thomas Brox. Single-view to multi-view: Reconstructing unseen views with a convolutional network. *CoRR*, abs/1511.06702, 2015. 2

[27] Daniel Watson, William Chan, Ricardo Martin-Brualla, Jonathan Ho, Andrea Tagliasacchi, and Mohammad Norouzi. Novel view synthesis with diffusion models. *arXiv preprint arXiv:2210.04628*, 2022. 2, 8

[28] Xiaogang Xu, Yingcong Chen, and Jiaya Jia. View independent generative adversarial network for novel view synthesis. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7790–7799, 2019. 6

[29] Mingyu Yin, Li Sun, and Qingli Li. Novel view synthesis on unpaired data by conditional deformable variational auto-encoder. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXVIII 16*, pages 87–103. Springer, 2020. 2, 3

[30] Mingyu Yin, Li Sun, and Qingli Li. Id-unet: Iterative soft and hard deformation for view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7220–7229, 2021. 2, 3

[31] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *ArXiv*, abs/1805.09817, 2018. 2

[32] Tinghui Zhou, Shubham Tulsiani, Weilun Sun, Jitendra Malik, and Alexei A. Efros. View synthesis by appearance flow. *CoRR*, abs/1605.03557, 2016. 2