**The University of Hong Kong**
**Faculty of Engineering**
**Department of Computer Science**


COMP4801 Final Year Project

# 3D reconstruction with single/multi-view images

Interim Report

**Title:** 3D reconstruction with single/multi-view images

**Supervisor:** Zhao Hengshuang

**Student:** Jiang Zeyu (3035639056)

**Date of Submission:** January 21, 2024

# Abstract

3D reconstruction from images is an important and challenging problem in computer vision. This progress report outlines research on a framework for reconstructing 3D models from single or multi-view images. A combination of diffusion models [4] and neural radiance fields [13] is proposed to enable high-quality reconstruction from sparse inputs. The diffusion model leverages strong priors to synthesize novel views and refine geometry. The neural radiance field [13] reconstructs an implicit 3D representation that can be rendered from any viewpoint. A preliminary version of the framework has been implemented. While the multi-view diffusion model gives a relatively satisfying result, reconstruction result is not satisfactory. We may investigate adopting 3D Gaussians Splatting to reconstruction process with only 1 forward pass in the future. Meanwhile, further evaluation of tradeoffs in reconstruction quality and efficiency compared to state-of-the-art methods will also be done in the future.

# Contents

# List of Figures



# List of Tables



# List of Abbreviations

**DDPM** Denoising Diffusion Probabilistic Models

**MLP** Multi Layer Perceptron

**MVS** Multi-view Stereo

**NeRF** Neural Radiance Field

**VAE** Variational Autoencoders

**SSIM** Structural Similarity Index, higher the better

**PSNR** Peak Signal-to-Noise Ratio, higher the better

**FPS** Frames Per Second, higher the better

**3DGS** 3D Gaussians Splatting

# 1 Introduction

## 1.1 Background

Humans can effortlessly interpret the 3D structure of an object from a mere collection of 2D images. This marvelous ability rooted in our cognitive faculties results from imagination and strong prior knowledge from our visual experiences. However, enabling machines to perform 3D reconstructions just like humans remains an open challenge, while there are many substantial applications across medicine, games, augmented reality and more.

## 1.2 Motivation

In 3D reconstruction, classic multi-view stereo (MVS) [3, 21] methods can reconstruct 3D models from multiple images taken from known viewpoints. However, they struggle with textureless surfaces, lighting variations, and thin structures. MVS also requires many input views captured in a controlled setting.

Lately, numerous methods have adopted neural radiance field (NeRF) to model 3D scenes, demonstrating strong reconstruction performance and high fidelity results [13, 23, 14, 8]. They use implicit neural representations, directly leveraging the learnable neural networks, despite traditional voxel, mesh or point cloud methods. However, they normally require considerable input images with known poses and may produce blurry output when facing unseen areas.

More recently, 3D Gaussian Splatting achieves real-time rendering of a trained 3D scene using 3D Gaussians as an explicit representation [5]. It still adopts the learning techniques as in NeRF [13], allowing faster backpropagation and reconstruction of a 3D scene, though it still needs enough images with pose inputs.

On the track of image generation, denoising diffusion probabilistic models (DDPM) [4, 17] have shown remarkable performances. Techniques based on DDPM [4] primarily learn a noise predictor from the forward process, which continuously adds noise to natural images. This learned noise predictor is then used in the reverse process to generate images from the Gaussian noise, through a series of diffusion steps.

Thus, this project hypothesizes that the confluence of NeRF [13], 3D Gaussians Splatting [5] and diffusion models may represent a new paradigm in 3D reconstructions, especially under the sparse-views condition. Furthermore, reconstructing 3D models from merely a few images taken from your phone may have vast applications in various domains like gaming and augmented reality.

## 1.3 Project objectives

The key objectives of this project are to develop a framework for 3D reconstruction with sparse input images, which should be capable of:

- **Single-view 3D reconstruction:** This framework should be capable of reconstructing 3D models from single image inputs and synthesizing consistent hidden views using the prior knowledge from diffusion models.

- **Incremental multi-view enhancement:** This framework should be capable of refining the quality of 3D models generated from single-view inputs. Enhancements may include correcting the textures and geometry of the model with additional inputs.
- **Flexable input handling:** This framework should be capable of directly handling inputs from the open world without needing categorical priors, masks, or predefined poses. This flexibility will make the framework more adaptable to real-world, uncontrolled scenarios.
- **Evaluation and analysis:** Evaluate the reconstruction quality compared to SOTA methods in closed-world benchmarks quantitively and qualitatively for open-world inputs. Analyze tradeoffs between single vs multi-view reconstruction in terms of quality, reconstruction time, and other available metrics.

### 1.4 Outline of the report

The remaining parts of this report proceed as follows. Section 2 analyzes the current research state and identifies literature review gaps. Section 3 offers a discussion on methodology and describes the framework's construction. Section 4 presents the work that has been accomplished, what remains to be done, plans for the future, and problems encountered. We round off with a conclusion restating objectives and progress.

## 2 Literature Review

The literature review summarizes related current research. We will first go through the basics of diffusion models, nerual radiance field [13] and 3D Gaussians Splatting [5]. Afterwards, we'll look into some view-conditioned diffusion models that are able to generate images from novel viewpoints leveraging the prior knowledge of large pre-trained 2D diffusion models. Finally, we'll see how diffusion models reforms the 3D generation and reconstruction tasks.

### 2.1 Diffusion Models

#### 2.1.1 Denoising Diffusion Probabilistic Model

The denoising diffusion probabilistic model (DDPM) [4] marks a new paradigm for image generation. The orginal DDPM [4] models a Markov chain as shown in the figure below.

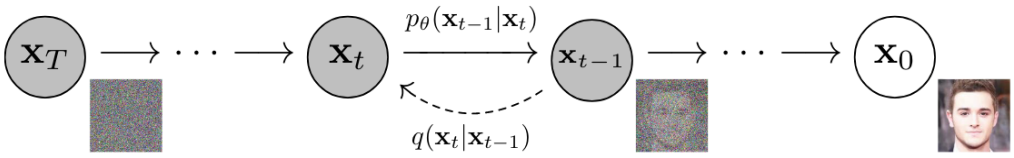


Figure 1: The Markov chain of DDPM [4]

Given $\mathbf{x}_0$ d representing the observed distribution of an arbitrary natural image and $\mathbf{x}_T$ representing the pure Gaussian noise, DDPM [4] learns a noise predictor from the forward process $q(\mathbf{x}_t|\mathbf{x}_{t-1})$ of adding

noise from $\mathbf{x}_0$ to $\mathbf{x}_T$. Since we are modeling a Markovian process, the forward process can be modeled as:

$$q(\mathbf{x}_{1:T}|\mathbf{x}_0) = \prod_{t=1}^{T} q(\mathbf{x}_t|\mathbf{x}_{t-1}) \tag{1}$$

Also, DDPM [4] assumes all latent variables in the encoder are a Gaussian distribution centered around the previous one and sets the mean and variance of the Gaussian encoder as follows, with $\alpha_t$ as a coefficient that may vary.

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\mathbf{x}_t; \sqrt{\alpha_t}\mathbf{x}_{t-1}, (1-\alpha_t)\mathbf{I}) \tag{2}$$

With $\alpha_t$ evolving over steps $t$ and $p(\mathbf{x}_T)$ being a standard Gaussian distribution, i.e. $p(\mathbf{x}_T) = \mathcal{N}(\mathbf{x}_T; \mathbf{0}, \mathbf{I})$:

$$p(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=1}^{T} p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \tag{3}$$

DDPM [4] then defines its trainable noise predictor by minimizing the KL divergence. $\epsilon$ is a random variable sampled from standard Gaussian. The final loss function used to train the noise predictor evolves to:

$$L_\theta = \mathbb{E}_{t,\S+0,\epsilon}\left[\left|\left|\epsilon - \epsilon_\theta\left(\sqrt{\overline{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\overline{\alpha}_t}\epsilon, t\right)\right|\right|^2\right] \tag{4}$$

### 2.1.2 Latent Diffusion Models

Due to the sequential and repeated nature of the DDPM [4], training and inference of the model should be performed step by step. Meanwhile, the DDPM [4] directly operates on the pixel space, it requires a large amount of memory during training and generating high resolution images. To resolve the above issues, the latent diffusion model [17] is trained to generate the latent representations of images, which applies the denoising process in the latent space. It utilizes a variational autoencoder (VAE) to encode the image into latent space during training while decoding the latent representations into images during inference process.

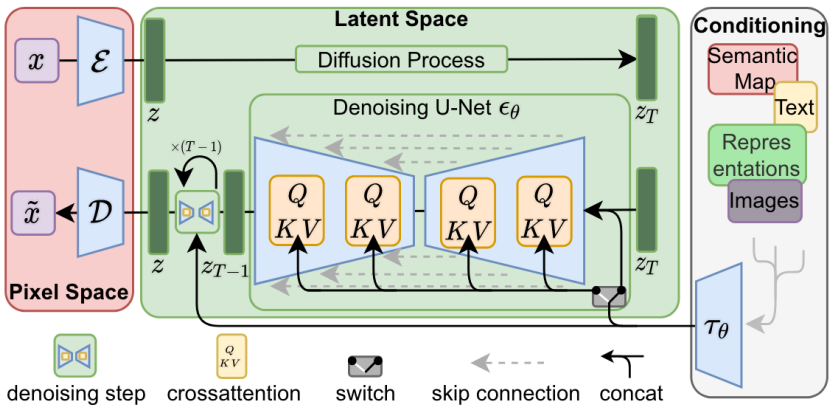


Figure 2: Architecture of latent diffusion model [17]

In the latent space, a U-Net [18] structure is used to predict the noise. This U-Net consists an encoder and a decoder. While the encoder downsamples the image's latent representation, the decoder upsamples

it back to the original size with less noise, making this U-Net output predicting the noise residual which is used to denoise the image's latent representation. Moreover, this model is also capable of conditioning on the additional input, such as text, image or semantic map via a cross-attention layer.

The cross-attention layer works with a pre-trained embedding model. Taking Stable Diffusion [17] as an example. It utilizes CLIP [16] to encode text prompt into a text embedding vector that could be fed into the cross attention layer. This makes it possible for the image generation processes to focus on the input text prompts.

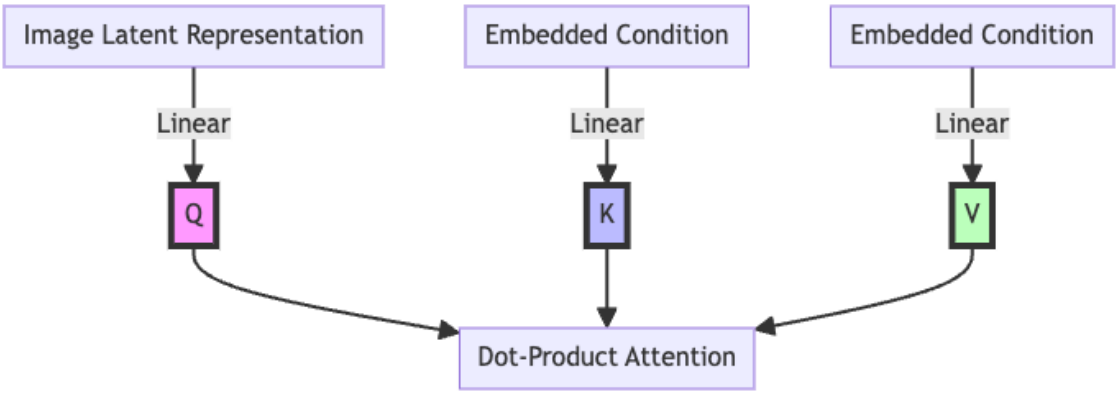


Figure 3: Cross Attention Mechanism

## 2.2 Neural Radiance Field

In the vanilla NeRF [13], a scene could be represented using a function taken in coordinate $\mathbf{x} = (x, y, z)$ in the 3D space, along with a viewpoint $\mathbf{d} = (\theta, \phi)$ to reconstruct colors $\mathbf{c} = (r, g, b)$ and densities $\sigma$ along the ray.

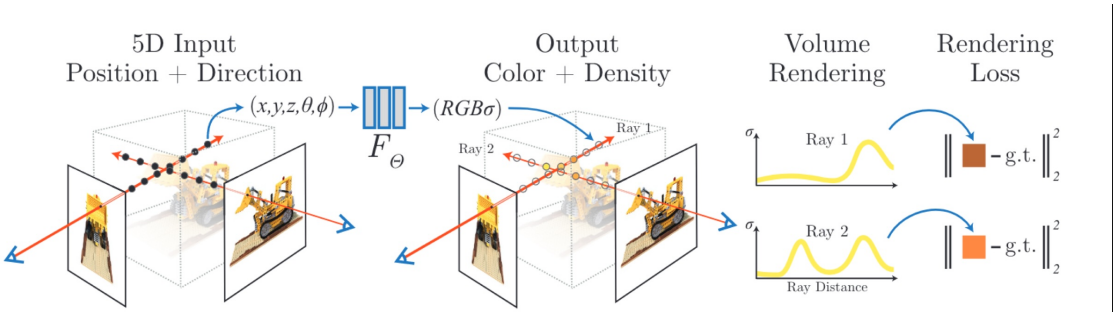


Figure 4: Neural Radiance Field [13] scene representation

The vanilla design of NeRF [13] uses sets of MLPs to model the radiance and density functions with trainable parameters. These MLPs are then trained using multi-view images of a scene to learn the implicit neural representation. Afterward, the novel view can be synthesized by querying these MLPs and rendering using volume renderings. Given the camera ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$, the color in the bounds $t_n$ to $t_f$ can be derived as [13]:

$$C(\mathbf{r}) = \int_{t_n}^{t_f} T(t)\sigma(\mathbf{r}(t))\mathbf{c}(\mathbf{r}(t), \mathbf{d})dt \tag{5}$$

In addition, the author found that directly operating on coordinate $\mathbf{x} = (x, y, z)$ and viewpoint $\mathbf{d} =$

$(\theta, \phi)$ results in underfitting problem, that the network cannot perform well when there's high variation of color and geometry in the scene. Therefore, it uses a positional encoding to embed the 5D vector into a higher dimensional space utilizing the sine and cosine functions, which is similar to that in the Transformer [22] architecture. Meanwhile, it introduces the volume rendering method with hierarchical volume sampling to avoid useless repeated sampling.

However, under our scenario of 3D reconstruction with sparse input images, the original design of NeRF [13] often produces blurry outputs, since itself does not have the ability to make prediction on the unseen areas. Meanwhile, due to its implicit representation design, it's slow to train and render. These downsides have been emphasized in future works. Instant-NGP [14] used a multi-resolution hash encoding, prominently improves the speed in training and rendering. NeuS [23] and Neuralangelo [8] used implicit signed distance function and novel volume rendering method to reconstruct smooth surface. Mip-NeRF 360 [1] focuses on resolve the blurry boundary issue by using online distillation and distortion-based regularizers.

## 2.3 3D Gaussians Splatting

3D Gaussians Splatting [5] is a more recent study adopting the key idea of training in NeRF [13] while using a relatively explicit 3D scene representation. It outperforms most NeRF-based method in terms of quality (SSIM, PNSR) and efficiency (FPS, Training Time).

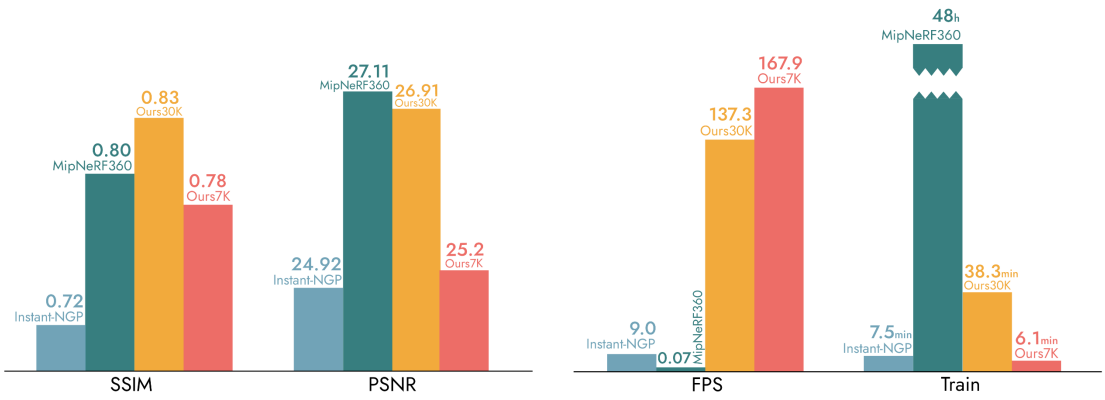


Figure 5: Performance Evaluation of 3D Gaussians Splatting [5]

3D Gaussians Splatting [5] uses anistropic 3D Gaussians as an explicit representation of a 3D scene. Each Gaussian can carry the feature vector like opacity and spherical harmonics to fit the directional appearance of the radiance field and itself carries position information naturally by centering itself at a specifc position. Then, by projecting relavant Gaussians onto a 2D plane using a differentiable tile rasterizer, it can render the image from a specific camera viewpoint. Compared to Point-NeRF [24], which also stores features in points but using volume rendering and linear interpolation, this process effectively leverages the explicit representation of the point cloud, the scene could be rendered way more faster than some NeRF-based methods which need to make inference through multiple MLPs.

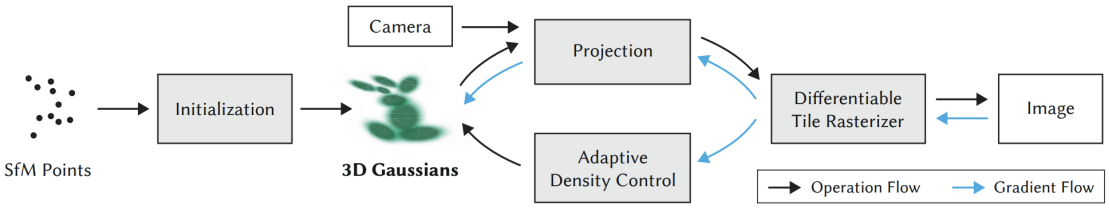


Figure 6: Flow of 3D Gaussians Splatting [5]

Another major contribution of the 3D Gaussian Splatting is its adaptive density control method. The control method basically checks on the gradient applies to each Gaussian. When the gradient is too large, proving that one Gaussian may not be enough to represent the scene at its position, then this Gaussian will be cloned or splitted as in the following diagram. Similarly, after some interations, remove the Gaussians that have opacity below the threshold, which means these Gaussians have nearly no impact on the rendering.

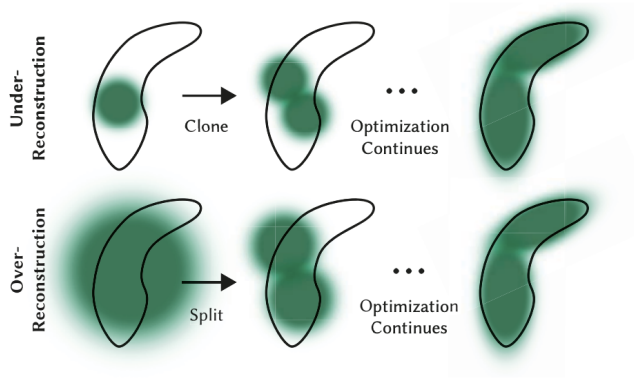


Figure 7: Densify (Clone/Split) Scheme of 3D Gaussians Splatting [5]

However, same as NeRF [13] methods, 3D Gaussian splatting still requires a large amount of images input, with camera poses. It will produce blurry outputs for unseen areas as well due to lack of prediction ability. Meanwhile, we found that 3D Gaussians Splatting cannot produce a sharp output when there's a large variance in shape and color, or in the boundary position.

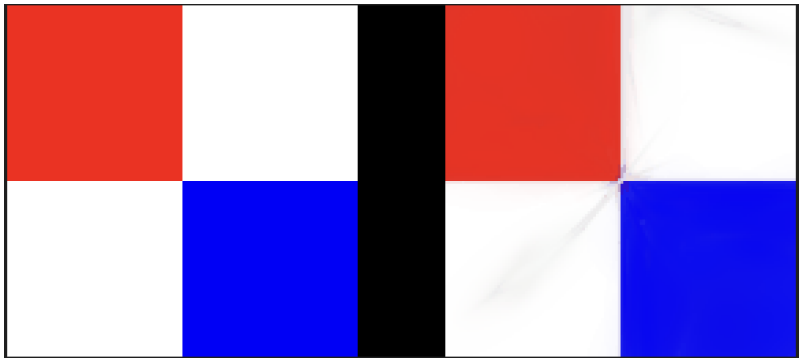

Figure 8: Simulation on recovering the left-side ground truth using 3DGS [5]

## 2.4 View-conditioned diffusion models

The pioneer work Zero-1-to-3 [10] make use of the large pre-trained 2D diffusion models, Stable Diffusion [17], to learn a control mechanism that could manipulate camera viewpoint during the image generation process, enabling the zero-shot ability.

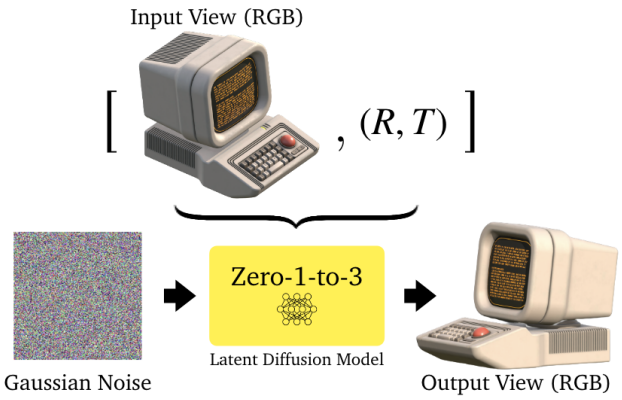


Figure 9: Model of Zero-1-to-3 [10]

It uses the Objaverse [2] dataset, which contains a large amount of 3D objects, processing it into viewpoint and image pairs to finetune the Stable Diffusion [17] model. However, since it's generating only one image at a time, even using the same view prompt, due to the probabilistic nature of the diffusion model, this may lead to inconsistency problems. One is the multi-face problem that the diffusion model repeatedly generates content that might be invisible in some angle, and the other is the content drift problem that some content in the image might be loss or gradually become other things.

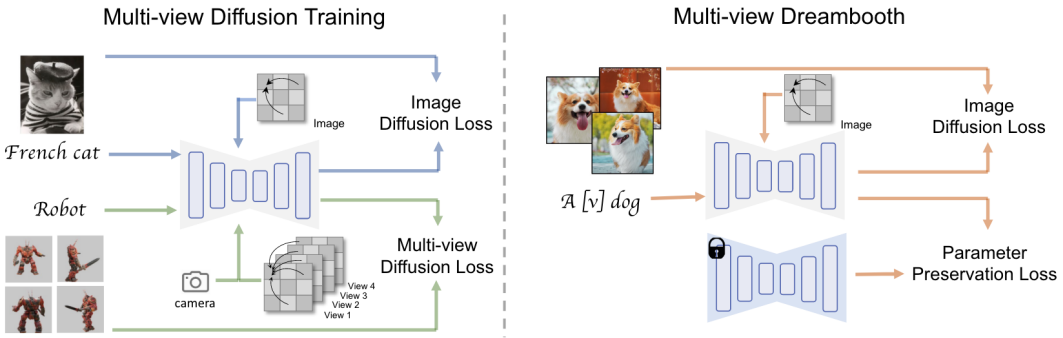


Figure 10: Pipeline of MVDream [20]

To resolve the above mentioned issue and keep the consistency cross views, SyncDreamer [11] and MVDream [20] both propose to generate multiple views at the same time. While SyncDreamer [11] finetunes from Zero-1-to-3 [10] and aims to model a joint distribution across different views, using a synced noise predictor for views from different angles. MVDream [20] finetunes from pre-trained 2D diffusion models and uses a 3D attention mechanism across the views to maintain consistency. Future work Wonder3D [12] also utilizes this 3D self-attention mechanism to ensure multi-view consistency, as well as generating paired

normal images for 3D reconstruction using SDF method.

## 2.5 Diffusion Guided 3D Generation and Reconstruction

DreamFusion [15], as a pioneer work, introduces the score distillation sampling technique that allows the 3D reconstruction process to be guided by the pre-trained diffusion models, as its pipeline shown in the following figure.

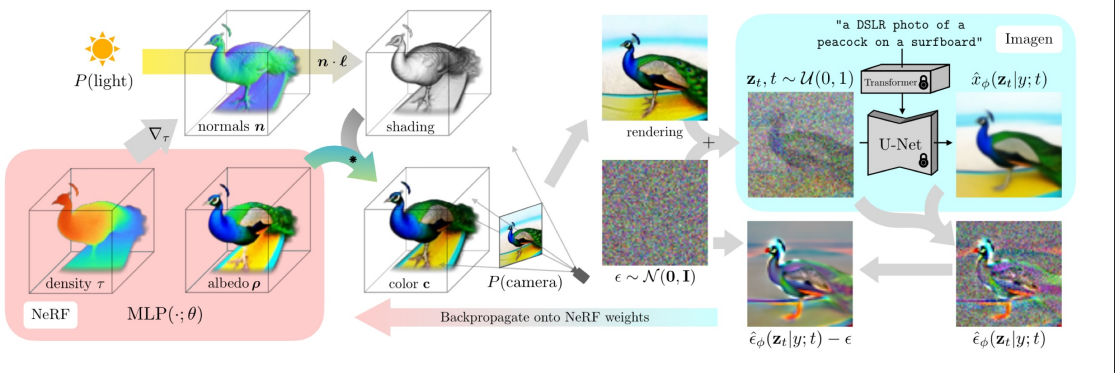


Figure 11: Pipeline of DreamFusion [15]

DreamFusion [15] firstly initialize the NeRF [13] model with parameters $\theta$. Then, given a specific camera viewpoint, it uses the NeRF [13] model to render the 2D image and adds noise to this rendered image. Afterwards, using the text prompt describing the viewpoint and the object that should be generated and the noised rendered image as the input to the pre-trained diffusion model. The diffusion model should be able predict the noise we just added to the rendered image. The difference between this prediction and the noise we add will be backpropagate onto the weights of the NeRF [13] model. Under this setting, DreamFusion proposes the following SDS loss, which is derived from the loss of the diffusion model and let the gradient to flow outside:

$$\nabla_\theta \mathcal{L}_{SDS}(\phi, \mathbf{x} = g(\theta)) = \mathbb{E}_{t,\epsilon}\left[w(t)(\hat{\epsilon}_\phi(\mathbf{z}_t; y, t) - \epsilon)\frac{\partial \mathbf{x}}{\partial \theta}\right] \tag{6}$$

However, the drawbacks of this method is obvious. Firstly, the vanilla DreamFusion directly uses a text-to-image diffusion model, Imagen [19], to serve as the guidance. As the training procedure of these kind of models did not focus on the view angle information, directly using text prompts to control the angle may be hard. Though this can be resolved by using fine-tuned view-conditioned diffusion model like Zero-1-to-3 [10], its need of per instance optimization still leads to ineffiency. Problem of inaccurate colors and textures also happens from time to time.

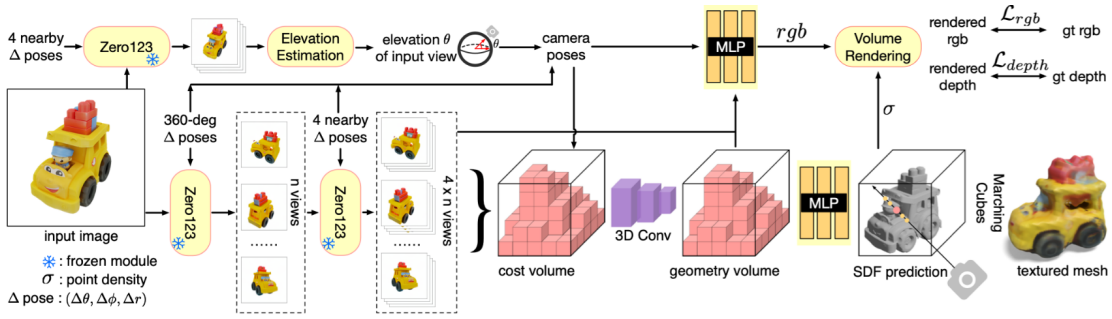


Figure 12: Pipeline of One-2-3-45 [9]

Under this scenario, One-2-3-45 [9] directly utilizing the Zero-1-to-3 [10] to generate novel views from multiple angles in and reconstruct them using SDF-based method NeuS [23]. Though it could generate the mesh in one forward pass, it still suffers from the inconsistency problem from Zero-1-to-3. Also, it needs to retrain on the 3D datasets for the latter part of depth prediction using SDF method, which may affect the generalization ability.

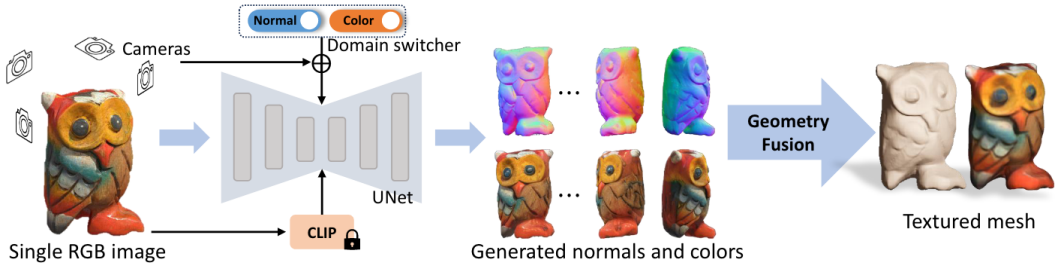


Figure 13: Pipeline of Wonder3D [12]

Wonder3D [12], a more recent work, directly fine-tunes the diffusion model to be able to output the paired normal images. In that case, there's no need to estimate depth information from colored image, and this output could be directly feed into SDF-based reconstruction method, achieving smooth surface while maintaining the efficiency by reconstruction in one forward pass as well.

# 3 Methodology

We design the prelinminary version of the framework to have three modules, a preprocessing module, a multi-view diffusion model conditioning on the input image for novel view synthesize and a 3D reconstruction module to do the reconstruction under the guidance of the multi-view diffusion model.

## 3.1 Preprocessing Module

The preprocessing is mainly the masking procedure that separate the object out from the environment and resize the image into the resolution of 256×256. We simply uses the out-of-box Segment Anything [6] model to create the alpha channel for the input images.

### 3.2 Multi-view Diffusion Model

We finetune our multi-view diffusion on the Stable Diffusion Variations model [7] on a small subset of Objaverse [2] dataset to generate 4 views of an object given the single image input.

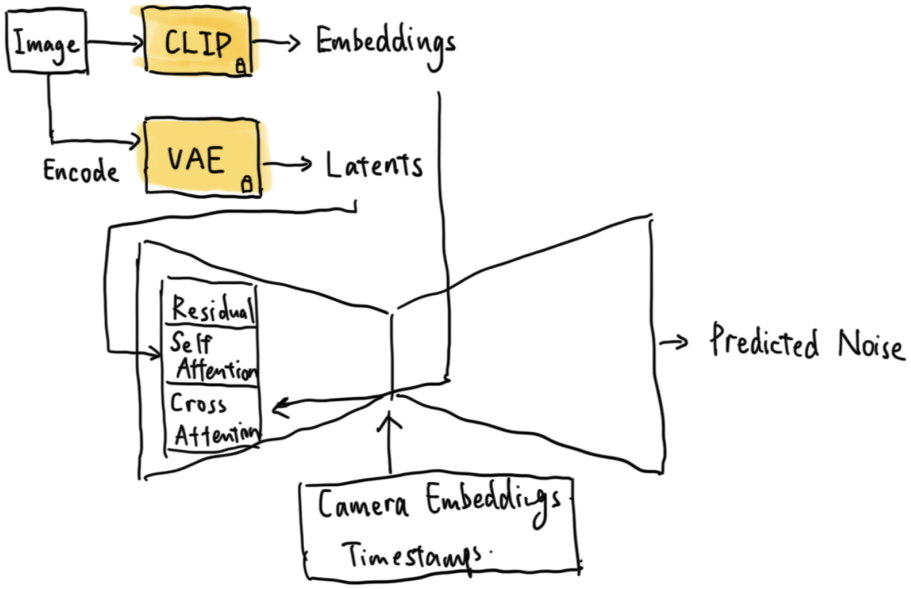


Figure 14: Pipeline of our multi-view diffusion model

Similar to text-conditioning in Stable Diffusion [17], we use CLIP [16] to embed the conditioning image and feed it into the cross attention block. Meanwhile, the original input image is also be encoded using the VAE to feed into the self attention block which is used to maintain multi-view consistency.

For camera embeddings, we adopt the world camera system and utilize a 2-layer MLP to encode the 4 camera postions of front, back, and sides. The camera embeddings are then directly added to the embedded timestamps like residual.

### 3.3 3D Reconstruction Module

We directly insert our multi-view diffusion model into the DreamFusion [15] pipeline and uses the SDS loss to do the reconstruction.

## 4 Progress and Prelinminary Results

### 4.1 Project schedule

The project schedule is shown in the Appendix A.

### 4.2 Work accomplished to date

Currently, the project is in its last phase. All work in previous thress phases are completed in time with satisfactory, including:

- A complete and comprehensive literature review.

- Reproduction of some studies like Zero-1-to-3 [10], DreamFusion [15], SyncDreamer [11].
- Implementation and training of the multi-view diffusion model.
- Integration of multi-view diffusion model with SDS method for 3D reconstruction.
- Draft of the interim report
- Preliminary implementation of novel view synthesis module

## 4.3 Prelinminary results

We've already fine-tuned the multi-view diffusion model and created an interface for generating novel views from specific angles. This demo can be directly run on a Macbook Pro with M2 Max chip in around 30 seconds, setting denoising steps to be 50.

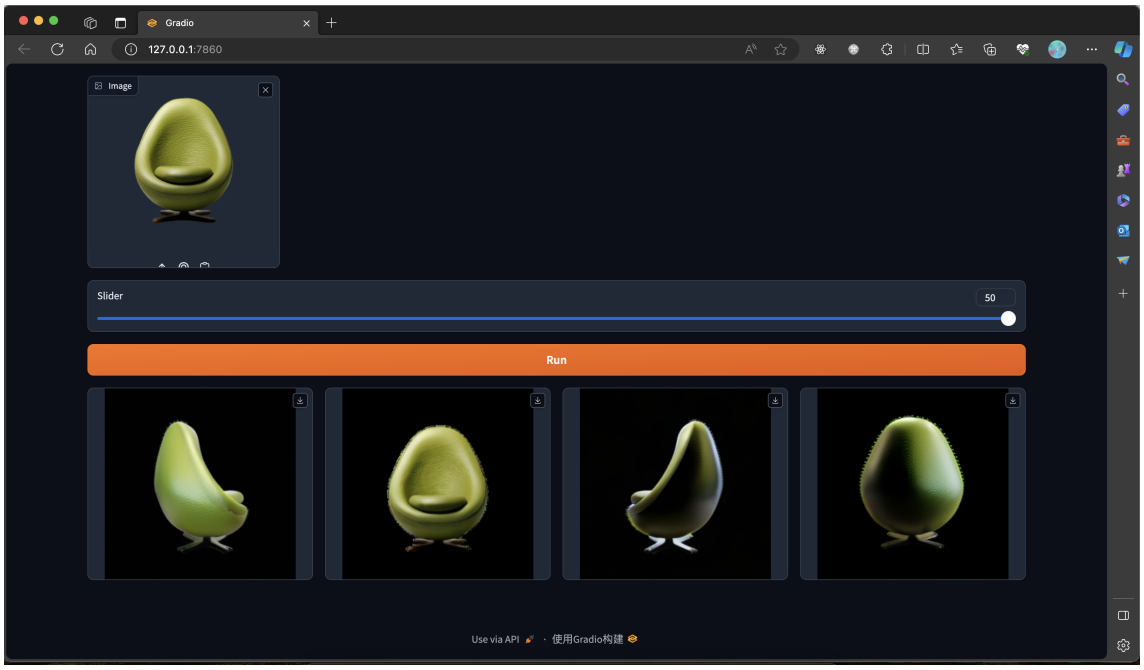


Figure 15: Interface of our multi-view diffusion model

Following are some images generated using our multi-view diffusion model.

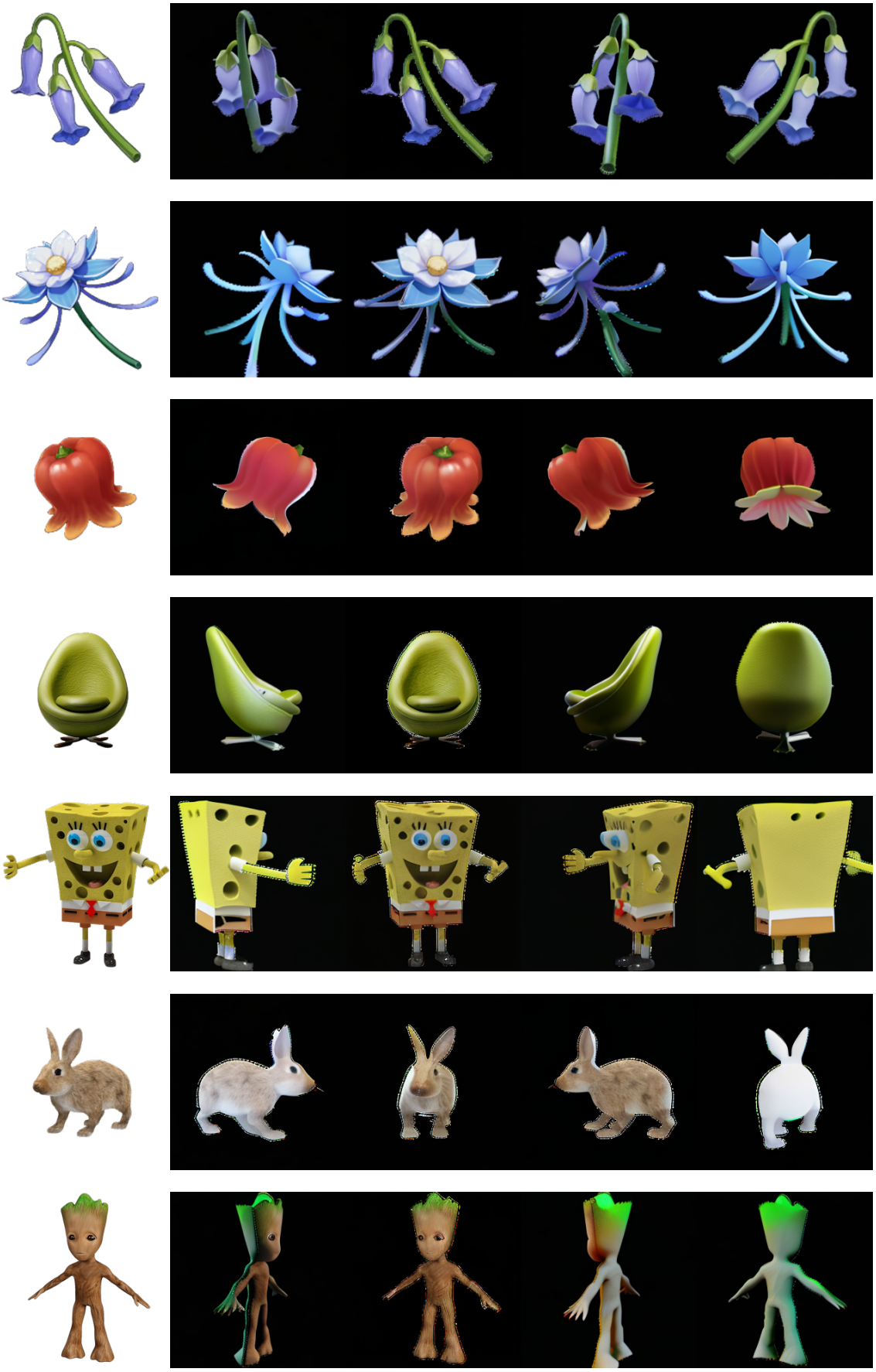

Figure 16: Images generated by multi-view diffusion models (gt, left, front, right, back)

Afterwards, we've tested to use DreamFusion [15] pipeline to do 3D reconstruction with our multi-view diffusion model. We run it for 15000 steps and this process takes  1 hour using 2 RTX3090s.

Figure 17: Reconstruction result of a chair

The surface isn't smooth, while the texture is also not satisfying as well. When we analyze the log during optimization, NeRF [13] rendering and propagation consumed 3/4 of the total time, which is truly time-consuming.

### 4.4 Future plan

In the last phase, we will focus on improving the performance and provide a comprehensive quantitive and quantitive analysis. We may focus on the adoption of 3D Gaussian Splatting, abandoning the SDS method and provide a one-stop interface for 3D reconstruction.

## 5 Conclusion

In conclusion, this interim report outlines progress on developing a framework for 3D reconstruction from sparse image inputs using diffusion models and neural radiance fields. The key objectives are to enable high-quality 3D reconstruction from single images, allow incremental enhancement with additional views, handle real-world inputs flexibly, and analyze tradeoffs compared to state-of-the-art methods.

The literature review summarizes relevant research on diffusion models, neural radiance fields, 3D Gaussian splatting, and recent works combining these approaches for novel view synthesis and 3D reconstruction.

The proposed methodology combines a preprocessing module, a multi-view diffusion model for novel

view synthesis, and a 3D reconstruction module guided by the multi-view diffusion model. Preliminary multi-view generation results are quite satisfying, though 3D reconstruction results are still coarse.

Future work will focus on improving reconstruction quality, efficiency, and flexibility. This includes adopting 3D Gaussians Splatting [5] to avoid slow volumetric rendering in NeRF [13]. A more comprehensive analysis will be performed in the future.

# References


[1] Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5470–5479, 2022. 5

[2] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. *arXiv preprint arXiv:2212.08051*, 2022. 7, 10

[3] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(8):1362–1376, 2010. 1

[4] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc., 2020. i, iii, 1, 2, 3

[5] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), July 2023. iii, 1, 2, 5, 6, 14

[6] Alexander M. Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C. Berg, Wan-Yen Lo, Piotr Dollár, and Ross Girshick. Segment anything. *arXiv (Cornell University)*, 4 2023. 9

[7] lambdalabs. Stable diffusion image variations. https://huggingface.co/spaces/lambdalabs/stable-diffusion-image-variations. 10

[8] Zhaoshuo Li, Thomas Müller, Alex Evans, Russell H. Taylor, Mathias Unberath, Ming-Yu Liu, and Chen-Hsuan Lin. Neuralangelo: High-fidelity neural surface reconstruction. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 8456–8465, 2023. 1, 5

[9] Minghua Liu, Chuan Xu, Haian Jin, Linghao Chen, Mukund Varma T, Zexiang Xu, and Hao Su. One-2-3-45: Any Single Image to 3D Mesh in 45 Seconds without Per-Shape Optimization. *arXiv (Cornell University)*, 6 2023. iii, 9

[10] 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. iii, 7, 8, 9, 11

[11] Yuan Liu, Lin Cheng, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. SyncDreamer: Generating Multiview-consistent Images from a Single-view Image. *arXiv (Cornell University)*, 9 2023. 7, 11

[12] Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. *arXiv preprint arXiv:2310.15008*, 2023. iii, 7, 9

[13] 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. *arXiv (Cornell University)*, 3 2020. i, iii, 1, 2, 4, 5, 6, 8, 13, 14

[14] Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM Trans. Graph.*, 41(4):102:1–102:15, July 2022. 1, 5

[15] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D Diffusion. *arXiv (Cornell University)*, 9 2022. iii, 8, 10, 11, 13

[16] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PMLR, 2021. 4, 10

[17] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10674–10685, 2022. iii, 1, 3, 4, 7, 10

[18] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-NET: Convolutional Networks for Biomedical Image Segmentation. *arXiv (Cornell University)*, 5 2015. 3

[19] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S. Sara Mahdavi, Rapha Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic Text-to-Image Diffusion Models with Deep Language Understanding. *arXiv (Cornell University)*, 5 2022. 8

[20] Yichun Shi, Peng Wang, Jianglong Ye, Mai Long, Kejie Li, and Xiao Yang. Mvdream: Multi-view diffusion for 3d generation. *arXiv preprint arXiv:2308.16512*, 2023. iii, 7

[21] Engin Tola, Christoph Strecha, and Pascal Fua. Efficient large-scale multi-view stereo for ultra high-resolution image sets. *Machine Vision and Applications*, 23(5):903–920, 5 2011. 1

[22] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017. 5

[23] Peng Wang, Lingjie Liu, Yuan Liu, Christian Theobalt, Taku Komura, and Wenping Wang. NEUS: Learning Neural implicit surfaces by volume rendering for multi-view reconstruction. *arXiv (Cornell University)*, 6 2021. 1, 5, 9

[24] Qiangeng Xu, Zexiang Xu, Julien Philip, Sai Bi, Zhixin Shu, Kalyan Sunkavalli, and Ulrich Neumann. Point-nerf: Point-based neural radiance fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5438–5448, 2022. 5

# A Project Schedule

| Phases | Duration |
|---|---|
| **Preparation and Literature Review**<br>• Focus on literature review<br>• Reproduce results from relevant areas<br>• Finalized plan and methodologies<br>• Setup environment and framework (PyTorch) | Sep 1, 2023 - Oct 15, 2023 |
| **Prelinminary implementation of the framework**<br>• Implementation of a prelinminary version<br>• interim<br>• Assume input images have masks and camera positions<br>• Fine-tune the diffusion and reconstruction network | Oct 16, 2023 - Dec 30, 2023 |
| **Improvement of Implementation**<br>• Performance improvement<br>• Interim report | Jan 1, 2023 - Jan 15, 2024 |
| **Finalization and Future Work**<br>• Finalize all implementation and report<br>• Future work includes deploying an online demo<br>• Invesigating on 3D Gaussians Splatting method, without SDS | Jan 16, 2024 - Apr 15, 2024 |

Table 1: Project Timeline