# Dream Fusion
# Text-to-3D
# Using 2D Diffusion

Fung Ho Kit 3035779105
Zhang Yuhan 3035771672
Jing Jingderong 3035771775

# Table of **contents**

# 01

# Introduction

# Introduction

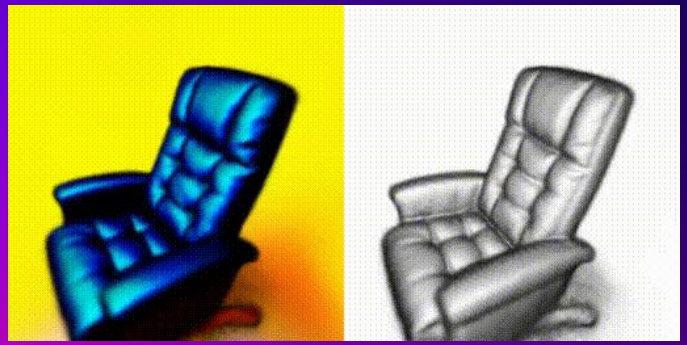
01.1

- Replicate and examine "DreamFusion: Text-to-3D Using 2D Diffusion"
- Novel approach for text-to-3D synthesis
- No need for 3D training data or image diffusion model modifications

# DreamFusion Overview

01.2

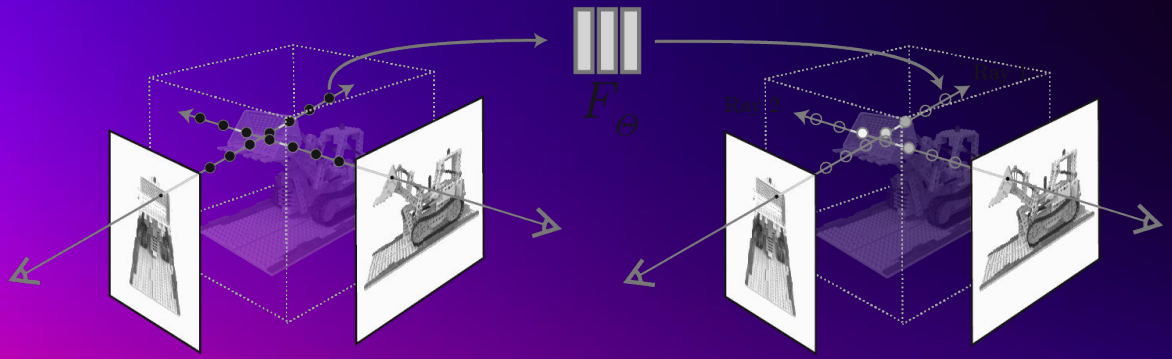
- Probability density distillation-based loss functions
- Generates 3D models from Neural Radiance Fields (NeRF)
- Optimizes via gradient descent for photorealistic results
- Potential applications: gaming, entertainment, architecture



# Model Description: mip-NeRF 360

01.3

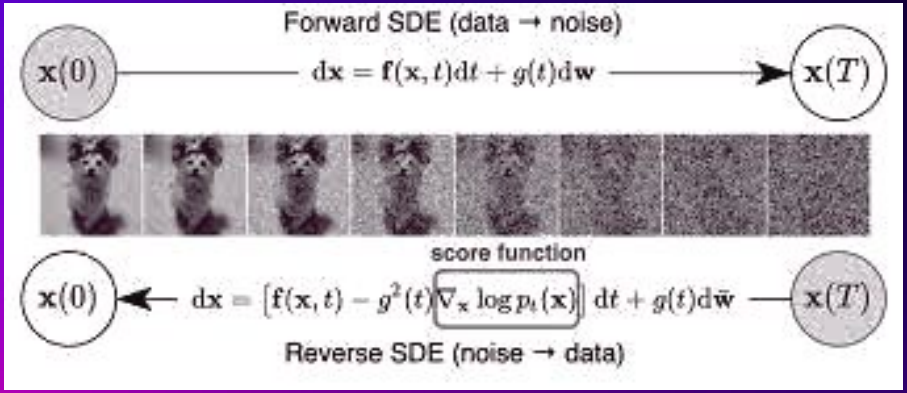
- Neural Radiance Field (NeRF) algorithm
- Randomly initialized for each caption
- Trained using images to learn radiance values
- Tailored to specific scene descriptions



# Model Description: Diffusion Model

01.4

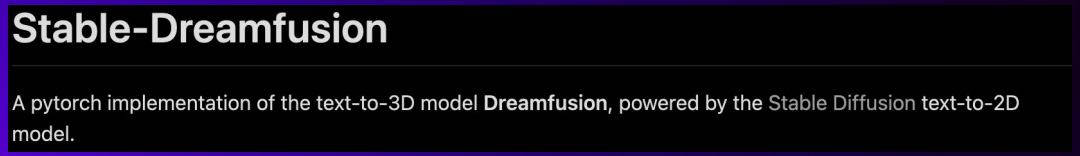
- Utilizes latent variables to generate new data samples
- Gradually transforms samples from noise distribution to data distribution
- Reversible transformations



# Third-Party Reproduction Differences

01.5

- Code contributor: ashawky from GitHub




| | DreamFusion: text-to-3D Using 2D Diffusion | Stable-DreamFusion |
|---|---|---|
| Main Model | Conditional Imagen Model | Stable-Diffusion Model |
| Operation Method | Operates on the image space | Operates on the latent space, requires extra time for propagating the loss back from VAE |

- Reproduction backbone choose: Instant-NGP NeRF backbone instead of Vanilla NeRF backbone to achieve faster rendering speed (around 10 frames per second at 800 x 800 resolution).and less GPU memory use (around 16GB GPU memory)

# Why DreamFusion?

01.6

- DreamFusion's promising approach to text-to-3D synthesis
- Extra dimension compared to traditional text-to-2D synthesis
- Expands potential applications of text-to-image synthesis
- Aims to explore possibilities and boundaries of 3D modeling

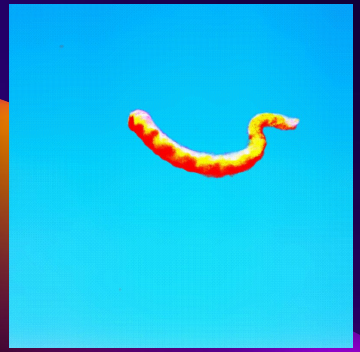
# 02

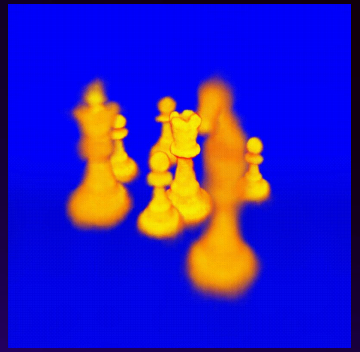
# Reproduction & Analysis

# Demo Trials
# Coherence Reproduction
# Ablation Test Reproduction

# Demo Trials

| Prompts | Correct output | Completeness |
|---|---|---|
| i) A rex rabbit | T | T |
| ii) A snake flying in the sky | T | F |
| iii) Four dogs are playing chess | F | F |
| iv) A Spiderman is on top of a car | Not Finished Yet | Not Finished Yet |





# Coherence Reproduction

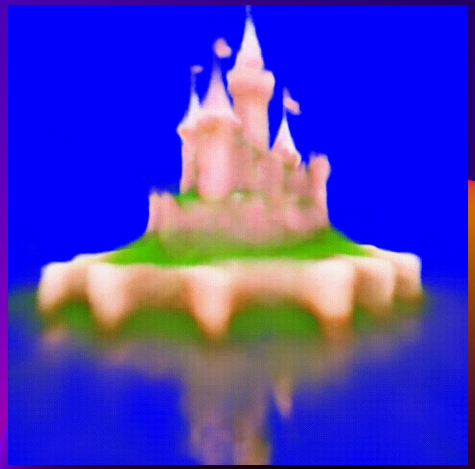
**4. Optimization.** Our 3D scenes are optimized on a TPUv4 machine with 4 chips. Each chip renders a separate view and evaluates the diffusion U-Net with per-device batch size of 1. We optimize for 15,000 iterations which takes around 1.5 hours. Compute time is split evenly between rendering the NeRF and evaluating the diffusion model. Parameters are optimized using the Distributed Shampoo optimizer (Anil et al., 2020). See Appendix A.2 for optimization settings.

| | R-Precision | | | | | |
|---|---|---|---|---|---|---|
| | CLIP B/32 | | CLIP B/16 | | CLIP L/14 | |
| Method | Color | Geo | Color | Geo | Color | Geo |
| GT Images | 77.1 | - | 79.1 | - | - | - |
| Dream Fields | 68.3 | - | 74.2 | - | - | - |
| (Reimpl.) | 78.6 | 1.3 | -99.9 | -0.8 | 82.9 | 1.4 |
| CLIP-Mesh | 67.8 | - | 75.8 | - | 74.5 | - |
| DreamFusion | 75.1 | 42.5 | 77.5 | 46.6 | 79.7 | 58.5 |
| **Stable dream fusion** | | | | | | |
| matte painting of a castle made of cheesecake surrounded by a moat made of ice cream | **30.9** | **22.62** | **35.57** | **24.85** | **27.96** | **21.69** |
| a vase with a hamburger pink flower | **27.51** | **24.84** | **30.04** | **24.49** | **28.6** | **20.08** |
| a hamburger | **33.98** | **23.68** | **33.32** | **27.87** | **28.98** | **25.04** |

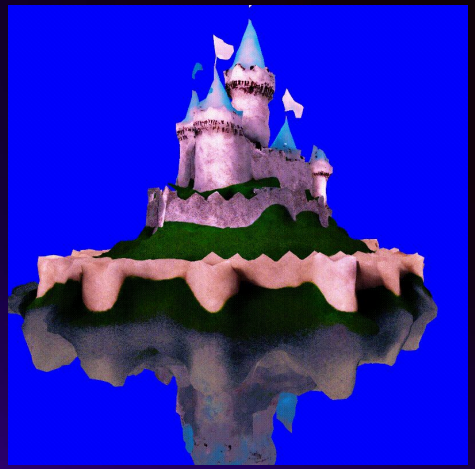
We only run 10000 iteration with 1 hours on GPU A3070

# Coherence Reproduction

Text:
matte painting of a castle made of cheesecake surrounded by a moat made of ice cream



DMTet finetuning



# Ablation Test Reproduction

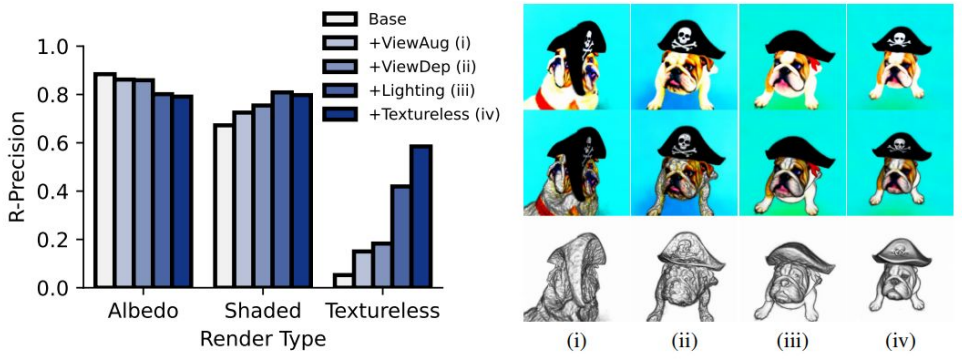



Figure 6: An ablation study of DreamFusion. **Left**: We evaluate components of our unlit renderings on albedo, full shaded and illuminated renderings and textureless illuminated geometry using CLIP L/14 on object-centric COCO. **Right**: visualizations of the impact of each ablation for *"A bulldog is wearing a black pirate hat."* on albedo (top), shaded (middle), and textureless renderings (bottom). The base method (i) without view-dependent prompts results in a multi-faced dog with flat geometry. Adding in view-dependent prompts (ii) improves geometry, but the surfaces are highly non-smooth and result in poor shaded renders. Introducing lighting (iii) improves geometry but darker areas (e.g. the hat) remain non-smooth. Rendering without color (iv) helps to smooth the geometry, but also causes some color details like the skull and crossbones to be "carved" into the geometry.

(i) ViewAug:
Without view-dependent prompts

(ii) ViewDeep:
Improve geometry

(iii) Lighting:
Improve geometry

(iv) Texttureless:
Smooth the geometry

# CUDA_OUT_OF_MEMOTY
# &
# Long Fitting Time

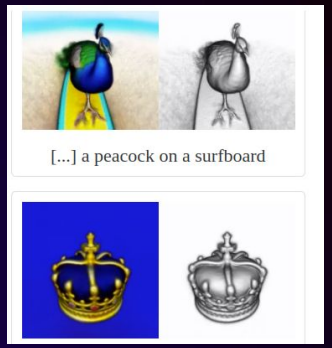
# 03

# Improvement

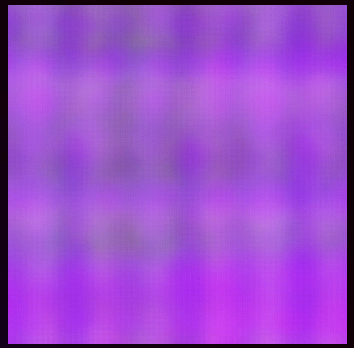
# **Limitation**
# of Dream Fusion

03

1. Fail to generate multiple objects
2. GPU Resources Occupation
3. Omit a significant amount of information




**Dream Fusion Art Gallery**



**Our generation**

# Our **Pipeline**

**Keyword Extraction**

1

2

**Relationship Mapping**

3

**3D mesh Editing**

# 1. Prompt - **Keyword Extraction**

Original input : dog is on the left hand side of the rabbit

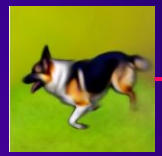
1
Vlt5 / XLNet / Bert

Dog

Rabbit

Left

2



Stable
Dream Fusion

```
(dreamfusion) [ImMusic@dslab Train]$
(dreamfusion) [ImMusic@dslab Train]$
(dreamfusion) [ImMusic@dslab Train]$ python prompt.py --text "the dog is on the left hand side of the rabbit" --model vlt5
Asking to truncate to max_length but no maximum length is provided and the model has no predefined maximum length. Default to no truncation.
+-----------------------------------------------------+
| dog, rabbit, dog is on the left hand side,          |
+-----------------------------------------------------+
(dreamfusion) [ImMusic@dslab Train]$
(dreamfusion) [ImMusic@dslab Train]$
(dreamfusion) [ImMusic@dslab Train]$
```

# 2. Relationship - **RelNet (GNN)**

Initial thought

Dog → Rabbit → Left → ...

Left

Above

Away

...

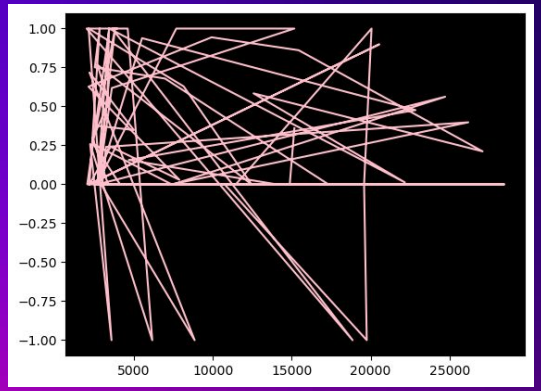
Facing

...

# 2. Relationship - **RelNet (FFN)**

**Second Trial**

**Left**




```
df = pd.read_csv("Relnet_data2.csv")
df
```

| | words | x | y | z |
|---|---|---|---|---|
| 0 | approach | 1 | 0 | 0 |
| 1 | ascend | 1 | 0 | 0 |
| 2 | descend | -1 | 0 | 0 |
| 3 | direct | 1 | 0 | 0 |
| 4 | enter | 1 | 0 | 0 |
| ... | ... | ... | ... | ... |
| 67 | straight | 1 | 0 | 0 |
| 68 | there | 0 | 0 | 0 |
| 69 | underfoot | 0 | 0 | 0 |
| 70 | upwards | 0 | 0 | 0 |

```
df_tokenized
```

| | words | x | y | z |
|---|---|---|---|---|
| 0 | 3921 | 1.00000 | 0.0 | 0.0 |
| 1 | 2004 | 1.00000 | 0.0 | 0.0 |
| 2 | 18855 | -1.00000 | 0.0 | 0.0 |
| 3 | 3622 | 1.00000 | 0.0 | 0.0 |
| 4 | 4607 | 1.00000 | 0.0 | 0.0 |
| ... | ... | ... | ... | ... |
| 67 | 3442 | 1.00000 | 0.0 | 0.0 |
| 68 | 2045 | 0.00000 | 0.0 | 0.0 |
| 69 | 2104 | 0.00000 | 0.0 | 0.0 |
| 70 | 14873 | 0.00000 | 0.0 | 0.0 |
| 71 | 15165 | 0.88147 | 1.0 | 0.0 |

72 rows × 4 columns

```
X_train, X_test, y_train, y_test = train_test_split(X, x,random_state=42)
regr = MLPRegressor(random_state=42, max_iter=500).fit(X_train, y_train)
regr.predict(X_test[:2])
regr.score(X_test, y_test)
```

```
-2760.521410364419
```

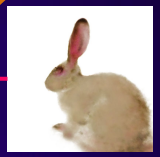
```
class TextClassifier(nn.Module):
    def __init__(self, vocab_size, embedding_dim, hidden_dim, num_classes)
        super(TextClassifier, self).__init__()
        self.embedding = nn.Embedding(vocab_size, embedding_dim)
        self.gru = nn.GRU(embedding_dim, hidden_dim, batch_first=True)
        self.fc = nn.Linear(hidden_dim, num_classes)

    def forward(self, x):
        x = self.embedding(x)
        h, _ = self.gru(x)
        h = h[:, -1, :]
        out = self.fc(h)
        return out
```

# 2. Relationship - **RelNet (Rule base)**

**Flexible solve -**
**Rule Base Method**

**Left**



**Transformat vector**
**(1,0,0)**

| words | x | y | z |
|---|---|---|---|
| approach | 0 | 1 | 0 |
| left | 1 | 0 | 0 |
| ascend | 0 | 1 | 0 |
| descend | 0 | -1 | 0 |
| direct | 0 | 1 | 0 |
| enter | 0 | 1 | 0 |
| exit | 0 | -1 | 0 |
| face | 99 | 99 | 0 |
| follow | 0 | -1 | 0 |
| head | 0 | 1 | 0 |
| lean | 99 | 99 | 0 |

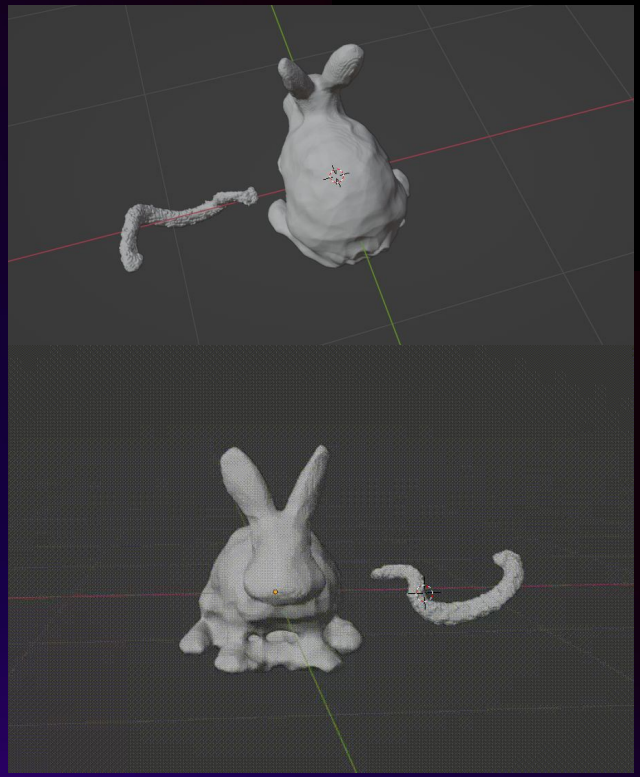
```
def randomised(x):
    if x == 99:
        x = np.random.rand()
    return x
```

# 3D model - **Editing and rendering**

**Trimesh + (pyvista)**

```
python mesh_to_video.py
--center_obj 'mesh_whiterabbit/mesh.obj'
--surround_obj 'mesh_snake/mesh.obj'
--transform_vector [1,0,0]
```

```
(dreamfusion) [ImMusic@dslab Train]$ python mesh_to_video.py --center_obj 'mesh_whiterabbit/me
sh.obj' --surround_obj 'mesh_snake/mesh.obj' --transform_vector [1,0,0]
[0.03760918 0.24613992 0.21480175] [1.0, 0.0, 0.0] 1.0653440554936726
----> merge mesh done
Generating frames ...
 21%|                                                    | 21/100 [00:04<00:15,  5.07it/s]
channel 3: open failed: connect failed: Connection refused
channel 4: open failed: connect failed: Connection refused
100%|                                                   | 100/100 [00:19<00:00,  5.03it/s]
----> rotation done
Rendering video ...
ffmpeg version 4.3 Copyright (c) 2000-2020 the FFmpeg developers
  built with gcc 7.3.0 (crosstool-NG 1.23.0.449-a04d0)
  configuration: --prefix=/home/ImMusic/.conda/envs/dreamfusion --cc=/opt/conda/conda-bld/ffmp
eg_1597178665428/_build_env/bin/x86_64-conda_cos6-linux-gnu-cc --disable-doc --disable-openssl
 --enable-avresample --enable-gnutls --enable-hardcoded-tables --enable-libfreetype --enable-l
ibopenh264 --enable-pic --enable-pthreads --enable-shared --disable-static --enable-version3 -
-enable-zlib --enable-libmp3lame
  libavutil      56. 51.100 / 56. 51.100
  libavcodec     58. 91.100 / 58. 91.100
  libavformat    58. 45.100 / 58. 45.100
  libavdevice    58. 10.100 / 58. 10.100
  libavfilter     7. 85.100 /  7. 85.100
  libavresample   4.  0.  0 /  4.  0.  0
  libswscale      5.  7.100 /  5.  7.100
  libswresample   3.  7.100 /  3.  7.100
```



# **Future** Work

1. Automate the pipeline
2. Enhance the generation quality
3. Enable more DOF

# Thanks!

Any question?