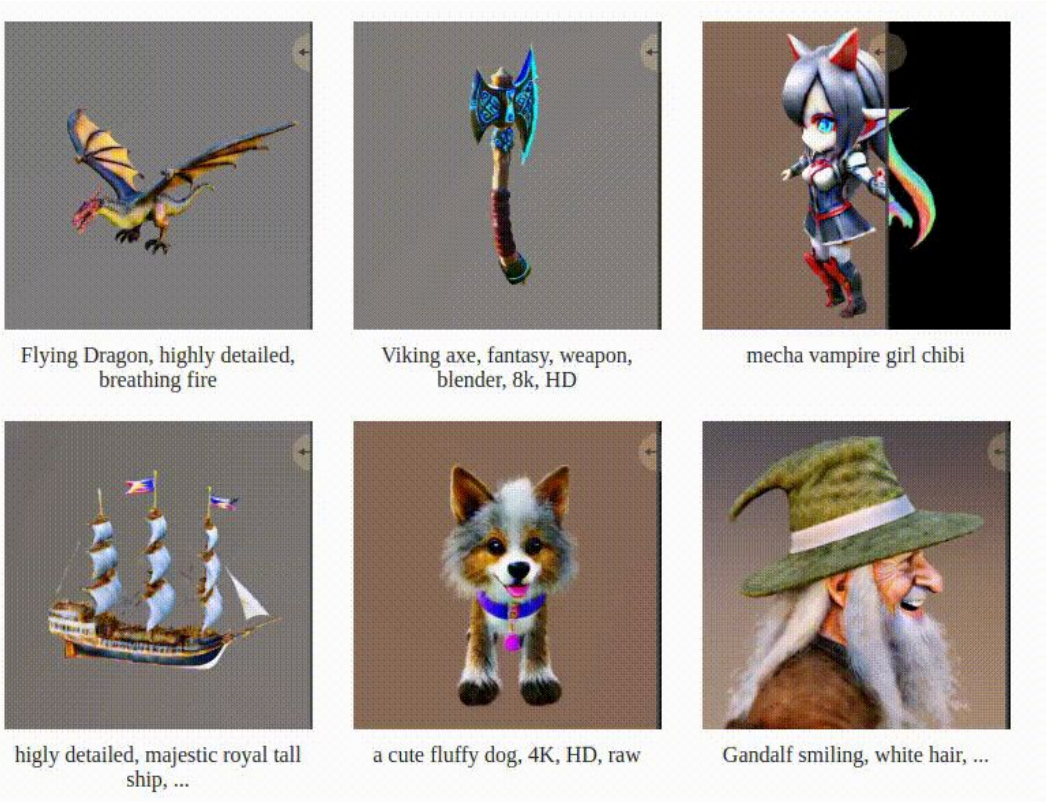
MVDream — text-to-3D оверлорд

(вроде SOTA)



Диффузионки

Все эти изображения сгенерированы нейронной сетью по текстовому запросу
[https://techcrunch.com/2022/10/17/stability-ai-the-startup-behind-stable-diffusion-raises-101m/]

Диффузионные модели могут использоваться для:

- Генерация изображения по тексту (+ по сегментации / карте глубины / человеческой позе)
- Изменение содержания изображения текстом
- Генерация видео по тексту
- Inpainting / Outpainting
- Image Restoration

Диффузионные модели для изображений — одна из самых активных областей в ML последние 1-2 года

DreamFusion — начало бума text-to-3D

На CLIP и Stable Diffusion до этого уже делали 3D Synthesis

Но **DreamFusion** — статья от Google Research 1 года давности — представила гораздо более гибкий метод.

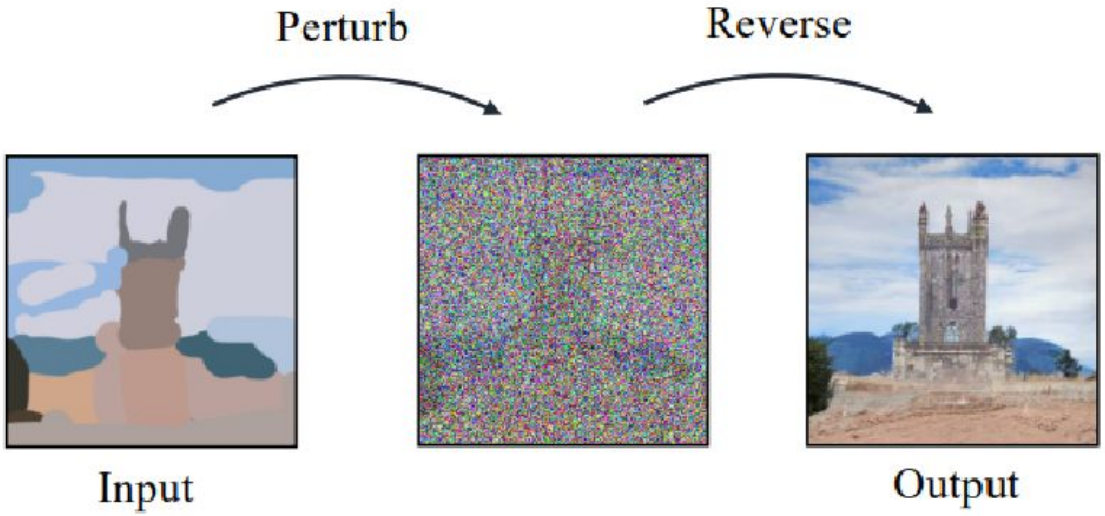
Называется **Score Distillation Sampling** — буквально дистилляция знаний диффузионной модели в объект.

Все результаты справа — 3D-объекты, сгенерированные по **text prompt'у**.

Важно, что это **не 3D Mesh'и**

https://dreamfusion3d.github.io/

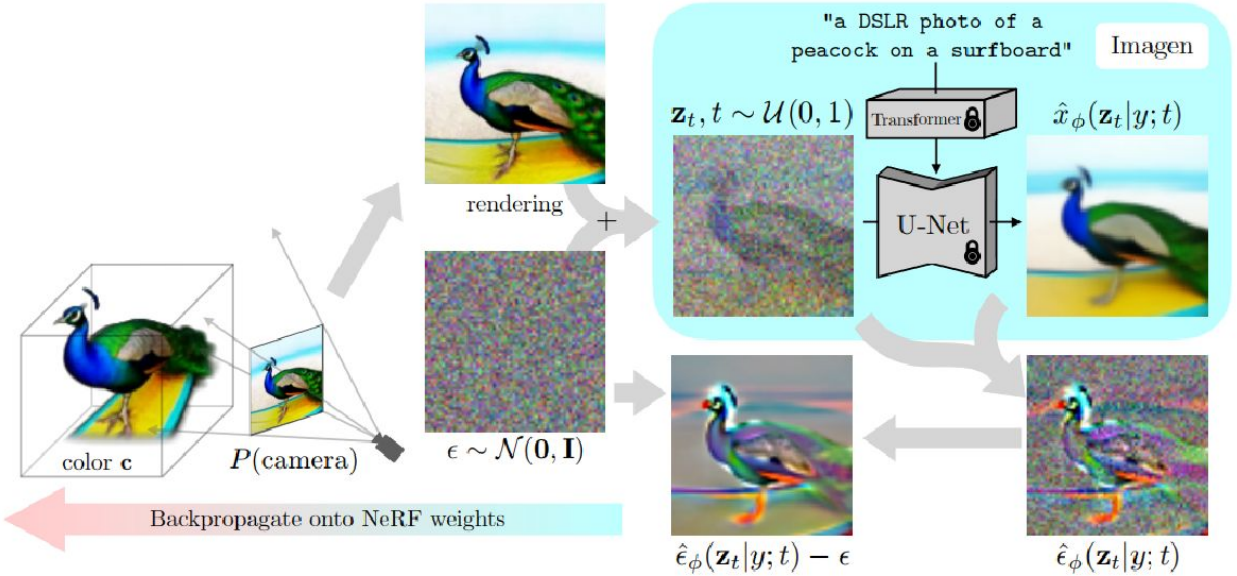
Диффузионки могут одним шагом менять картинку



https://sde-image-editing.github.io/

Reparameterization trick на диффузионных моделях описанный в DDIM позволяет рассматривать их как **Denoising модели**
Это можно использовать для **зашумления + расшумления** картинки диффузионкой

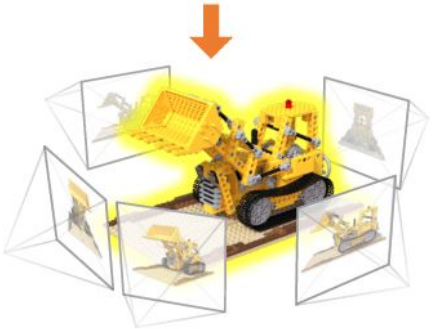
Как это работает на рендерах



Алгоритм Score Distillation Sampling (SDS) из DreamFusion:

1. Sample camera
2. Render object from the camera
3. Get a change that needs to happen to the render
4. Back-propagate the change to the 3D-object (as a gradient)

Images + accurate camera poses

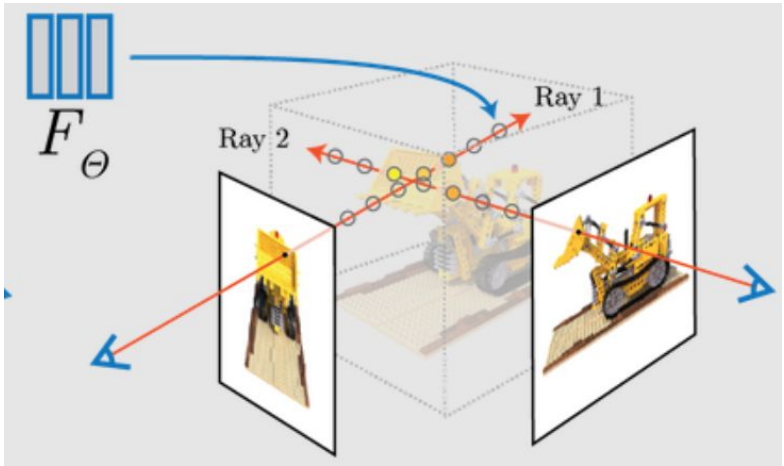


3D scene representation

$$(x,y,z,\theta,\phi) \rightarrow \boxed{\quad} \rightarrow (RGB\sigma)$$

$$F_\Theta$$

Весь объект — одна нейронная сеть



Можно рендерить дифференцируемо

Для представления генерируемого объекта используется **NeRF**, который изначально создан для 3D-реконструкции

Проблемы DreamFusion:

1. “The Janus Problem”
   2D диффузионки по запросу обычно генерируют объект **спереди**.
   Получается объект у которого **каждая сторона — передняя.**

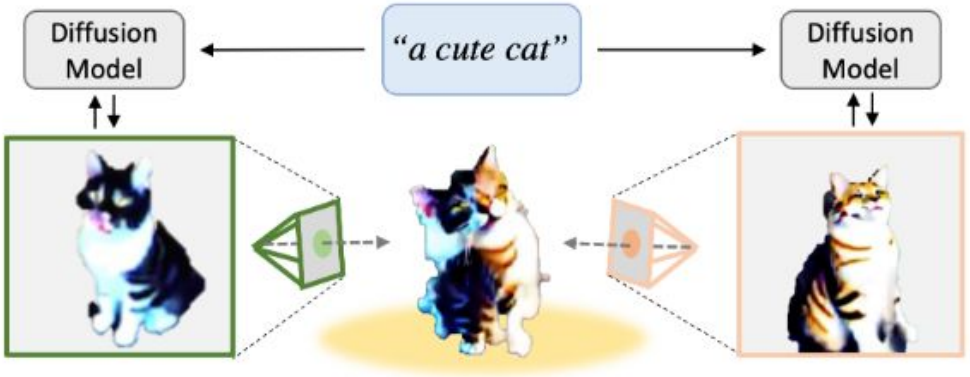
2. Качество
   a. Размытость
   b. Полу-прозрачность
   c. Перенасыщение цвета

3. Неконсистентность

“HDR photo of an elephant”

“a cute little kitten”

"a stack of pancakes with maple serum and butter"



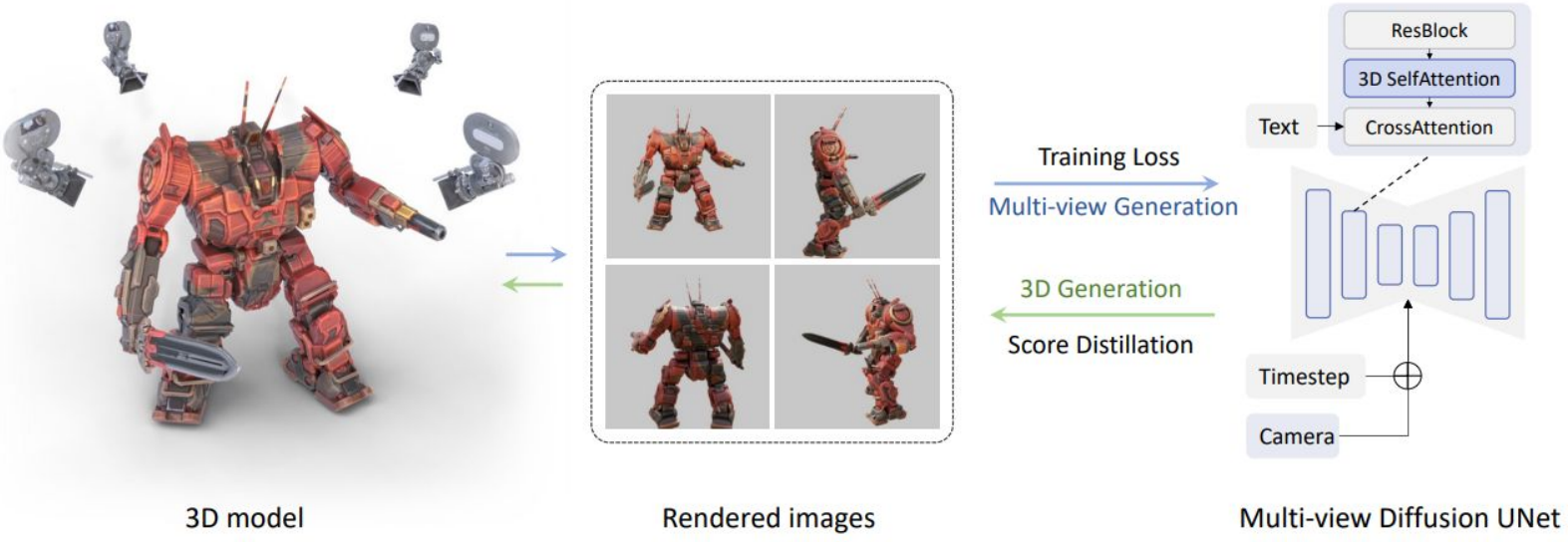
Визуализация Janus и неконсистентности

DreamFusion не раскрыл код, поэтому смотрим на [stable-dreamfusion]

MVDream — примеры

https://mv-dream.github.io/

MVDream — 4 равномерно расположенные камеры

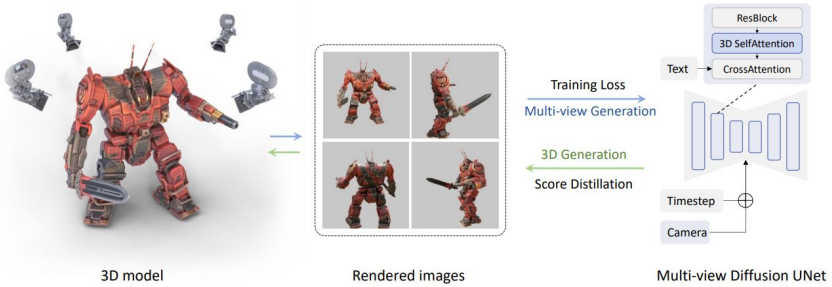


https://mv-dream.github.io/

Вместо случайных камер, берём 4 расположенные равномерно

1. **Camera parameters** теперь кормятся в диффузионку
2. **3D Attention** для обмена информацией между 4мя предсказаниями

MVDream — дообучение на 3D-датасете



https://objaverse.allenai.org/

Обучение:

- 2D-объект: Loss Function как обычно. (с camera=0)
- 3D-объект: один sample — 4 рендера одного объекта — зашумляется одним уровнем шума, потом обычный Loss

Это Stable Diffusion v2.1 дообученный на 3D-датасете **Objaverse**
Плюс **LAION** — 400M image-text пар
Обучается на **32 A100** с **batch_size=1024** 😭

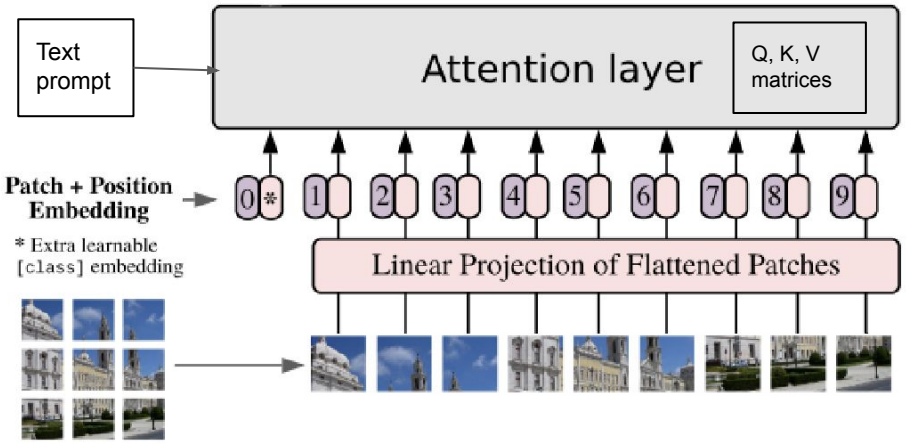
MVDream — дообучение на 3D-датасете

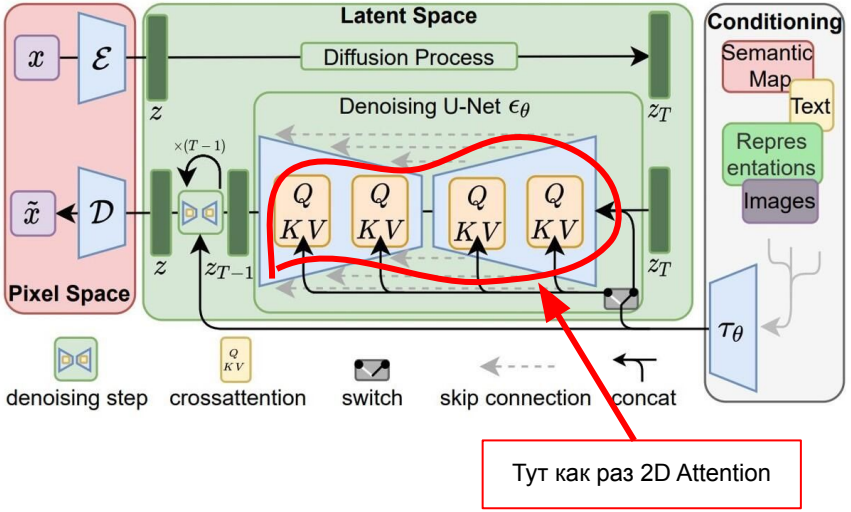
Model	Batch Size	FID↓	IS↑	CLIP↑
training data	N/A	N/A	14.75 ± 0.81	31.31 ± 3.34
Multi-view Diffusion				
- no 2D data	256	33.41	12.76 ± 0.70	30.60 ± 3.14
- proposed	256	32.57	13.72 ± 0.91	31.40 ± 3.05
- proposed	1024	32.06	13.68 ± 0.41	31.31 ± 3.12

Table 1: Quantitative evaluation on image synthesis quality. DDIM sampler is used for testing.

Важно — дообучение на 2D+3D сильно лучше чем просто на 3D

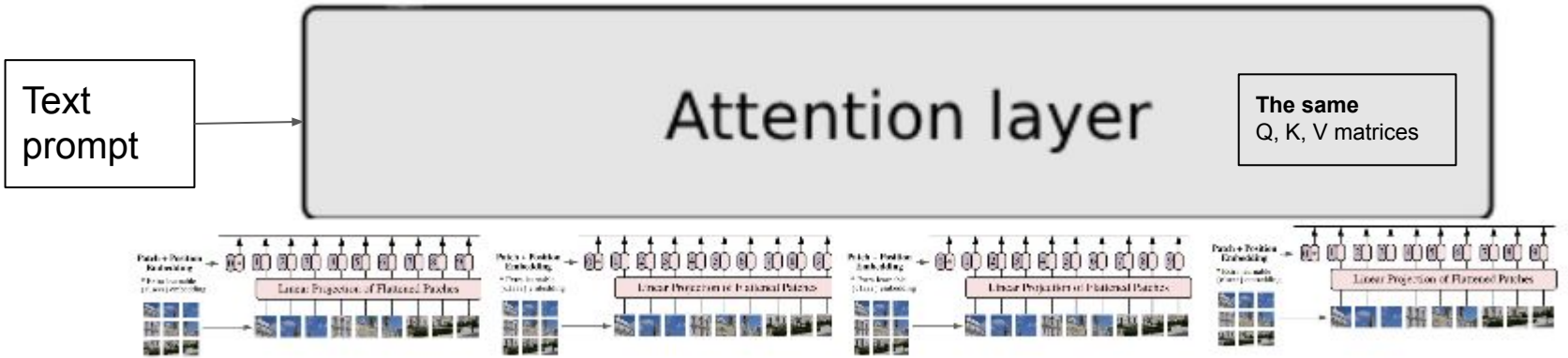
MVDream — 3D Attention





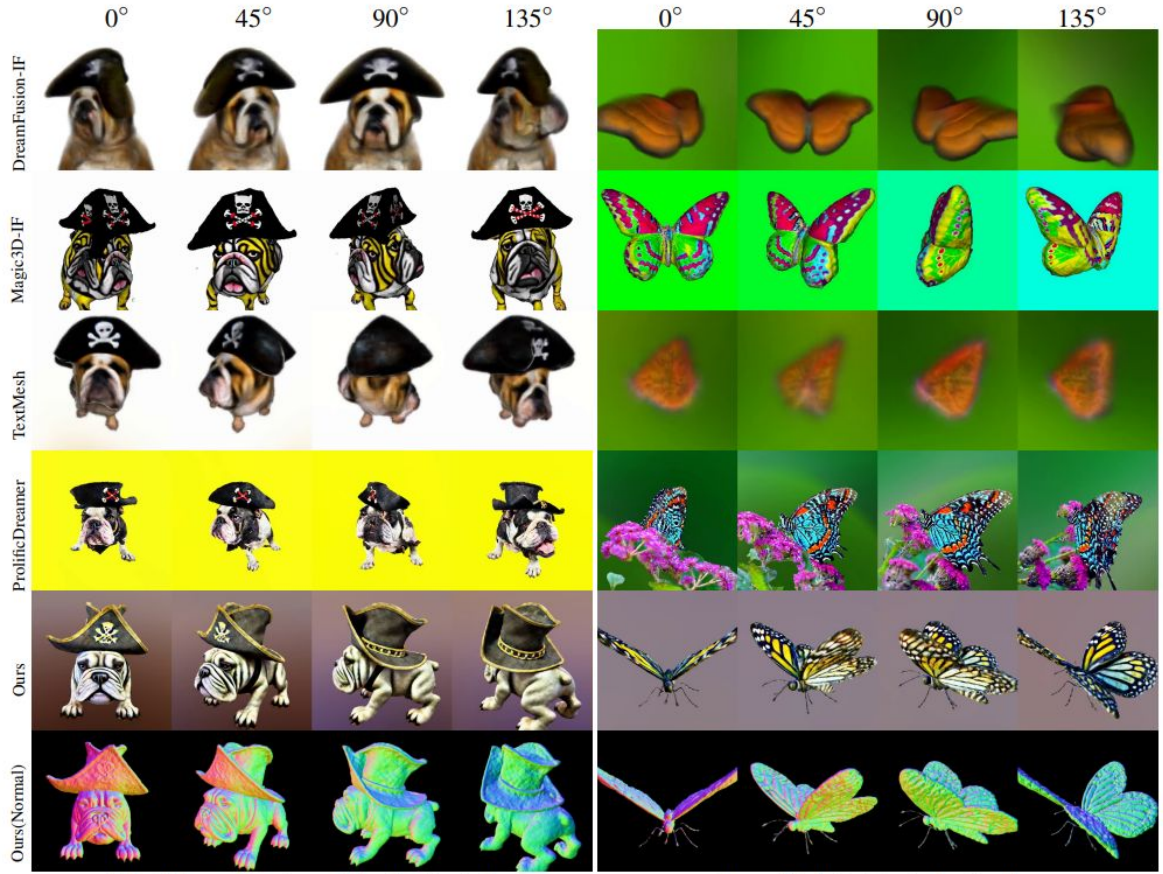
Простой **2D Attention** — эмбеддим патчики, потом в Self-Attention слой с Positional Encoding
Справа **Stable Diffusion**, на котором основывается MVDream

MVDream — 3D Attention



3D Attention из MVDream — патчики **из всех 4 картинок** участвуют в Self-Attention слое
И **переиспользуются** те же Q,K,V

MVDream — Сравнение



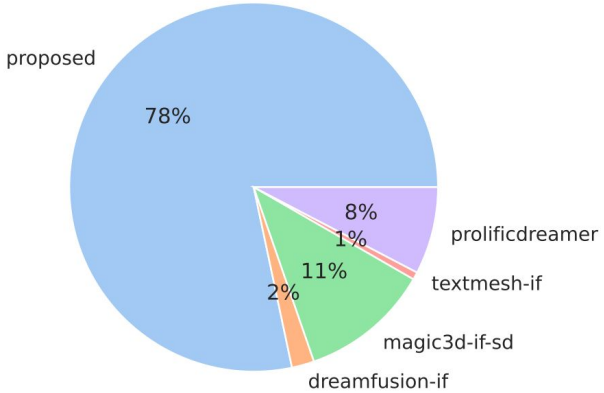


Figure 8: User study.

MVDream — Выводы

+ 3D Attention и дообучение на 3D объектах дало **консистентность**, **избавилось от Janus** (multi-face) и значительно **улучшило качество**
+ Неупомянутые трюки поверх SDS loss **исправляют перенасыщение цвета**
+ Продемонстрировали как делать **DreamBooth** (персонализация) в 3D генерации

- **1.5 часа** для генерации на Tesla V100
- Стиль в основном “игрушечный” из-за специфики датасета