ScaleDreamer: Scalable Text-to-3D Synthesis with Asynchronous Score Distillation

Zhiyuan Ma, Yuxiang Wei, Yabin Zhang, Xiangyu Zhu,
Zhen Lei, and Lei Zhang

ECCV 2024

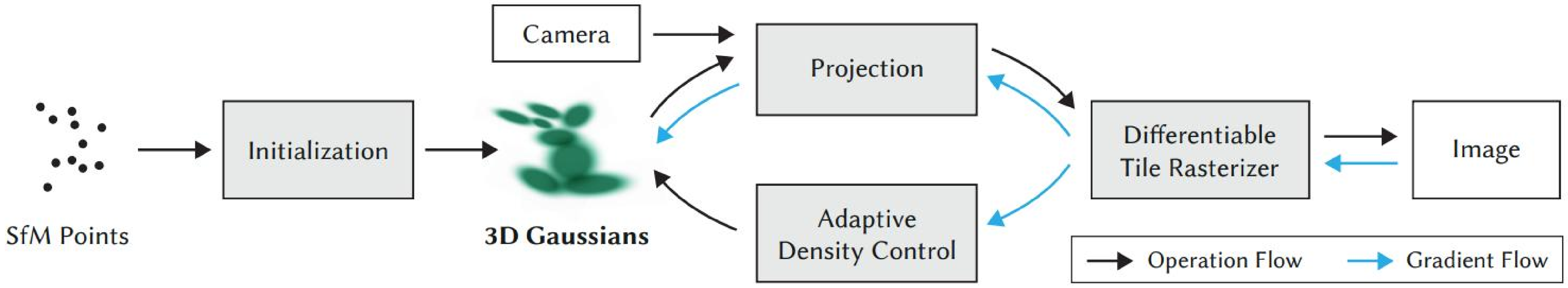
STRUCT Group Seminar
Presenter: Yifan Li
2024.7.20

Outline

- Background
- Method
- Experiments
- Conclusion

Background

3D Reconstruction



Explicit access of constraints and prior knowledge

Background

3D Generation

a tiger dressed as a doctor*

a baby bunny sitting on top of a stack of pancakes†

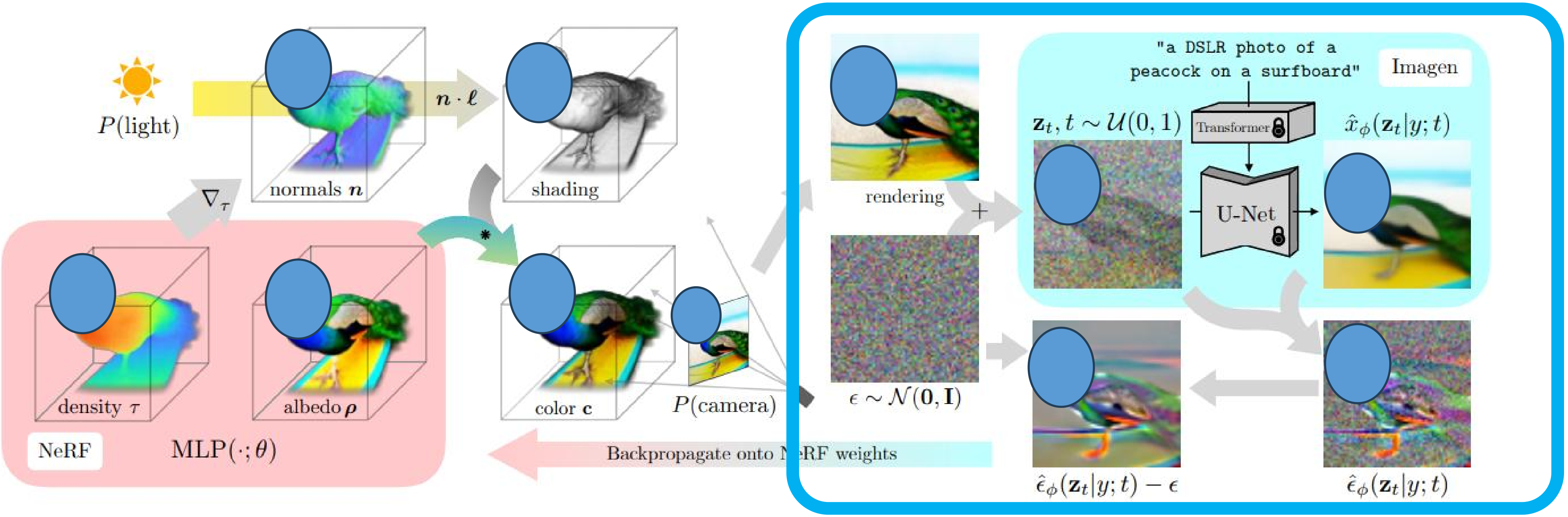
Lack of enough external priors to generate a high quality 3D object

High quality huge scale 3D dataset is hard to collect

How to utilize strong generative power of **2D Diffusion Models** to 3D?

Background

DreamFusion: Diffusion Model as a loss



Score Distillation Sampling (SDS)

“DreamFusion: Text-to-3D using 2D Diffusion”, Ben Poole, Ajay Jain, Jonathan T. Barron, Ben Mildenhall, ICLR23 outstanding paper

Background: DreamFusion

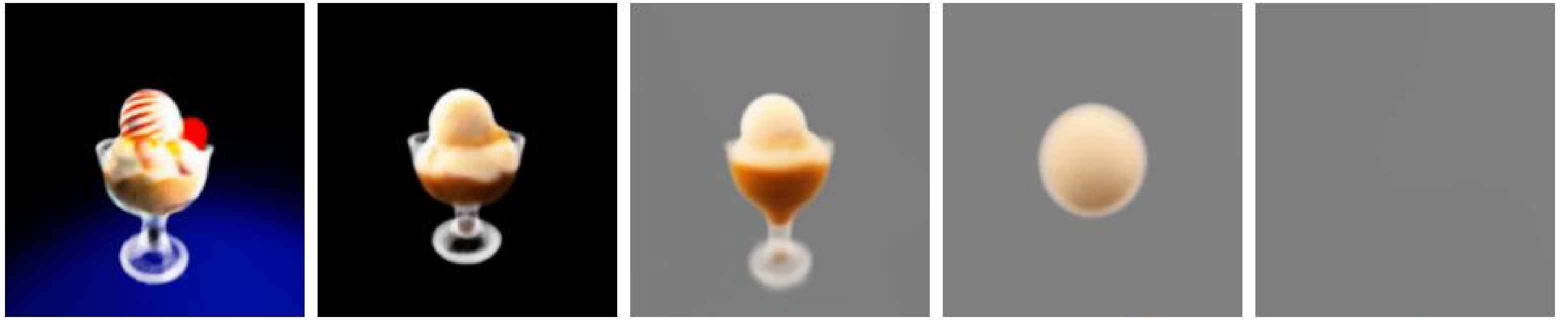
- Pros:
    - Do not need to backpropagate through the diffusion model
    - DM simply acts like an efficient, frozen critic predicts image-space edits
    - Effectively insert 2D DM’s generative priors to produce 3D objects
- Cons:
    - Need to set the Classifier Free Guidance as high as 100 for convergence
    - Produce excessively large gradients and lead to unstable optimization
    - High-saturation results

Background

Classifier Score Distillation (CSD)

Classifier score is the true essential component that drives the optimization

$$\delta_x(\mathbf{x}_t; y, t) = \underbrace{[\epsilon_\phi(\mathbf{x}_t; y, t) - \epsilon]}_{\delta_x^{\text{gen}}} + \omega \cdot \underbrace{[\epsilon_\phi(\mathbf{x}_t; y, t) - \epsilon_\phi(\mathbf{x}_t; t)]}_{\delta_x^{\text{cls}}}$$



$\omega = 40$ $\omega = 20$ $\omega = 10$ $\omega = 5$ $\omega = 0\ (\delta_{gen})$

ω: classifier free guidance intensity

"Text-to-3D with Classifier Score Distillation", Xin Yu, Yuan-Chen Guo, Yangguang Li, Ding Liang, Song-Hai Zhang, Xiaojuan Qi, arXiv 23.10

Background

Classifier Score Distillation (CSD)

Replace “ground truth noise”

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

$$\nabla_\theta \mathcal{L}_{\text{CSD}} = \mathbb{E}_{t,\epsilon,\mathbf{c}} \left[w(t)(\epsilon_\phi(\mathbf{x}_t; y, t) - \epsilon_\phi(\mathbf{x}_t; t)) \frac{\partial \mathbf{x}}{\partial \theta} \right]$$

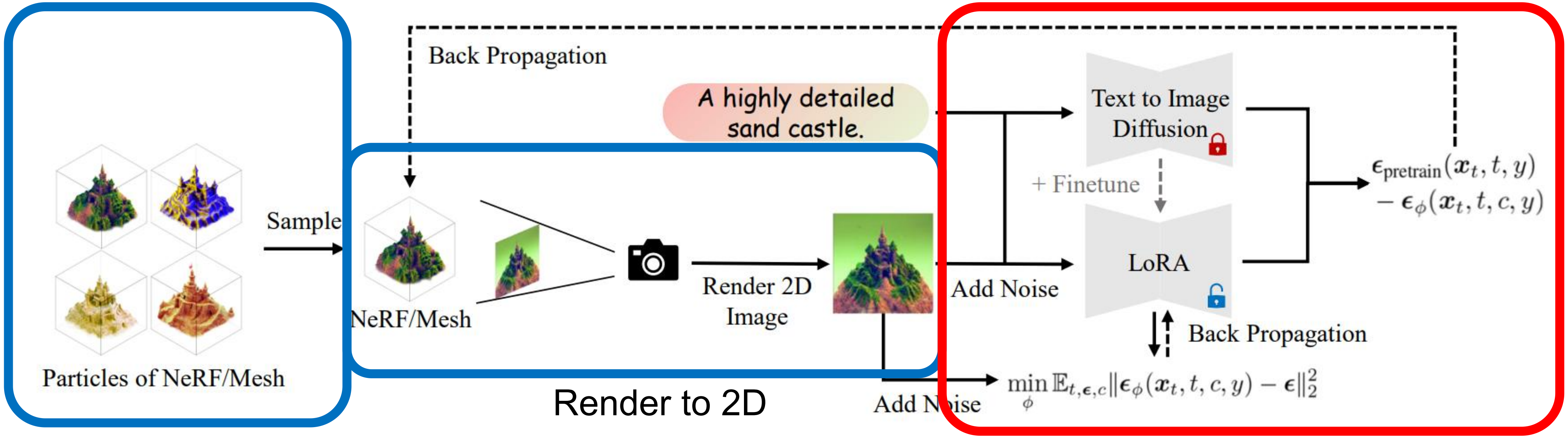
θ: parameters of 3D model (NeRF, …), used to generate a rendered 2D image
ϕ: parameters of diffusion model
y: text prompt

Background

Variational Score Distillation (VSD)

Predict noise adaptively and more accurately

Find a better alignment to rendered images distribution



3D generation

Main contribution

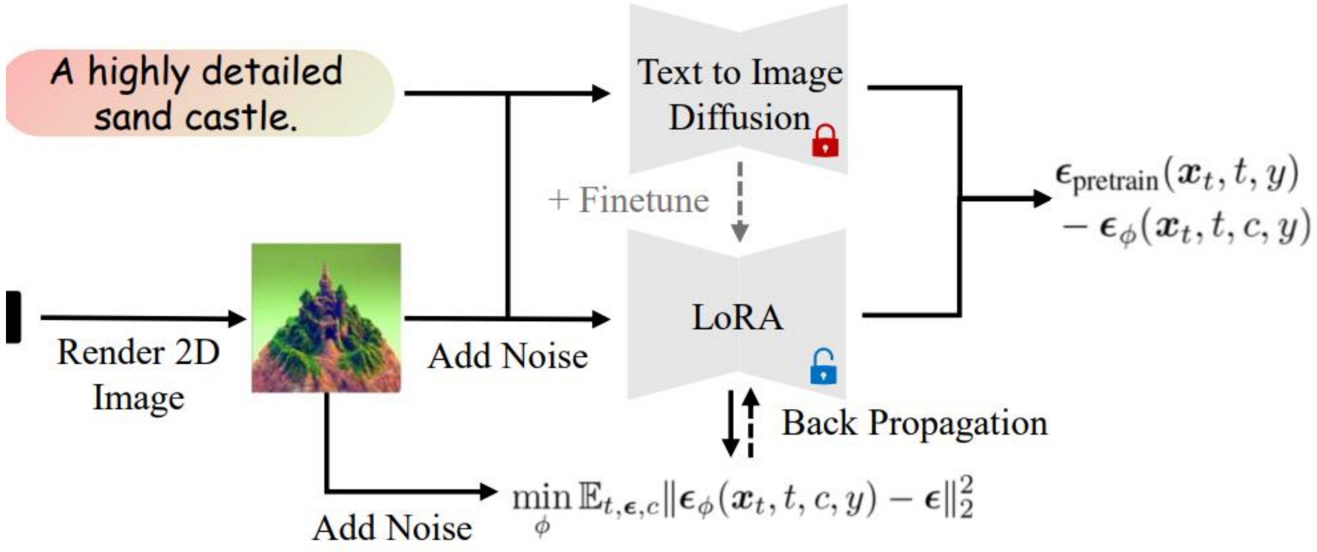
“ProlificDreamer: High-Fidelity and Diverse Text-to-3D Generation with Variational Score Distillation”, Zhengyi Wang et al., NeurIPS 23.

Background

Variational Score Distillation (VSD)

A better alignment achieves more accurate noise prediction

- Train a LoRA to predict noise
- Form a bi-level optimization
  - Finetune LoRA first, then predict noise to optimize 3D generation model iteratively



Background

Variational Score Distillation (VSD)

Algorithm 1 Variational Score Distillation

Input: Number of particles n (≥ 1). Large text-to-image diffusion model $\epsilon_{\text{pretrain}}$. Learning rate η_1 and η_2 for 3D structures and diffusion model parameters, respectively. A prompt y.

1: **initialize** n 3D structures $\{\theta^{(i)}\}_{i=1}^n$, a noise prediction model ϵ_ϕ parameterized by ϕ.
2: **while** not converged **do**
3: Randomly sample $\theta \sim \{\theta^{(i)}\}_{i=1}^n$ and a camera pose c.
4: Render the 3D structure θ at pose c to get a 2D image $\boldsymbol{x}_0 = \boldsymbol{g}(\theta, c)$.
5: $\theta \leftarrow \theta - \eta_1 \mathbb{E}_{t,\boldsymbol{\epsilon},c}\left[\omega(t)\left(\boldsymbol{\epsilon}_{\text{pretrain}}(\boldsymbol{x}_t, t, y^c) - \boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t, t, c, y)\right)\frac{\partial \boldsymbol{g}(\theta,c)}{\partial \theta}\right]$
6: $\phi \leftarrow \phi - \eta_2 \nabla_\phi \mathbb{E}_{t,\epsilon}||\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t, t, c, y) - \boldsymbol{\epsilon}||_2^2$.
7: **end while**
8: **return**

Outline

- Background
- Method
- Experiments
- Conclusion

Method

Asynchronous Score Distillation (ASD)

Improve VSD which limited to:

- problematic optimization
- sacrificed comprehension ability to diverse prompts

Assumption:

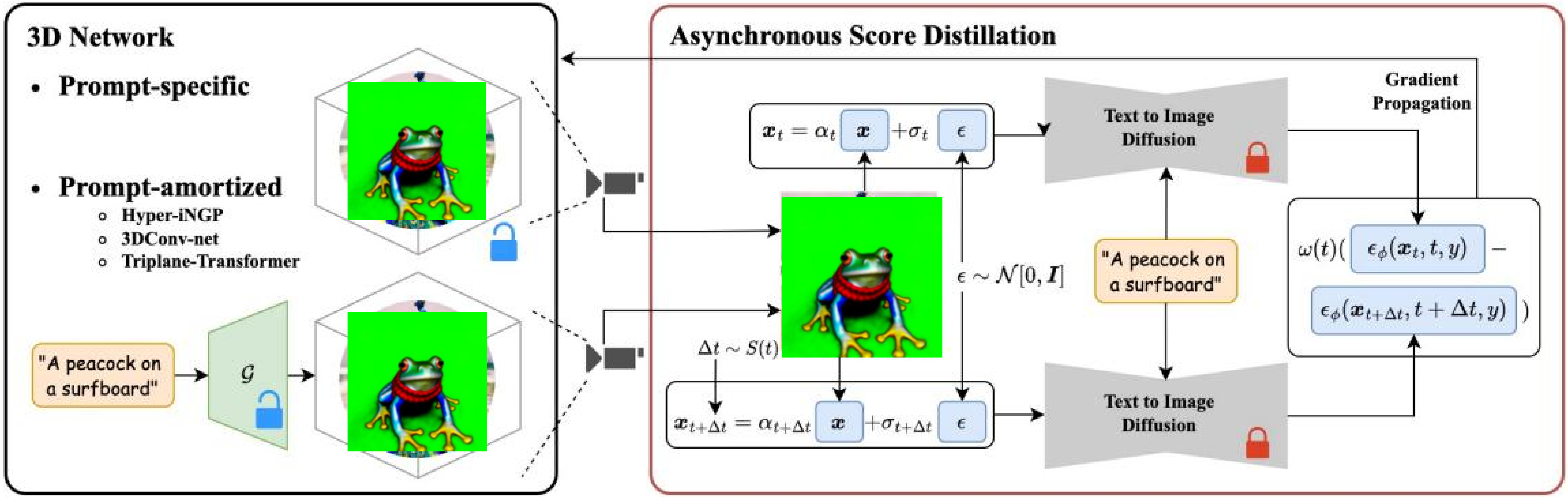
better alignment with rendered image distribution, will lead to:

- more accurate noise prediction
- more effective loss gradient for optimization
- better 3D generation results

Method

Asynchronous Score Distillation (ASD)

- Predict noises on different timesteps, use discrepancy as gradient



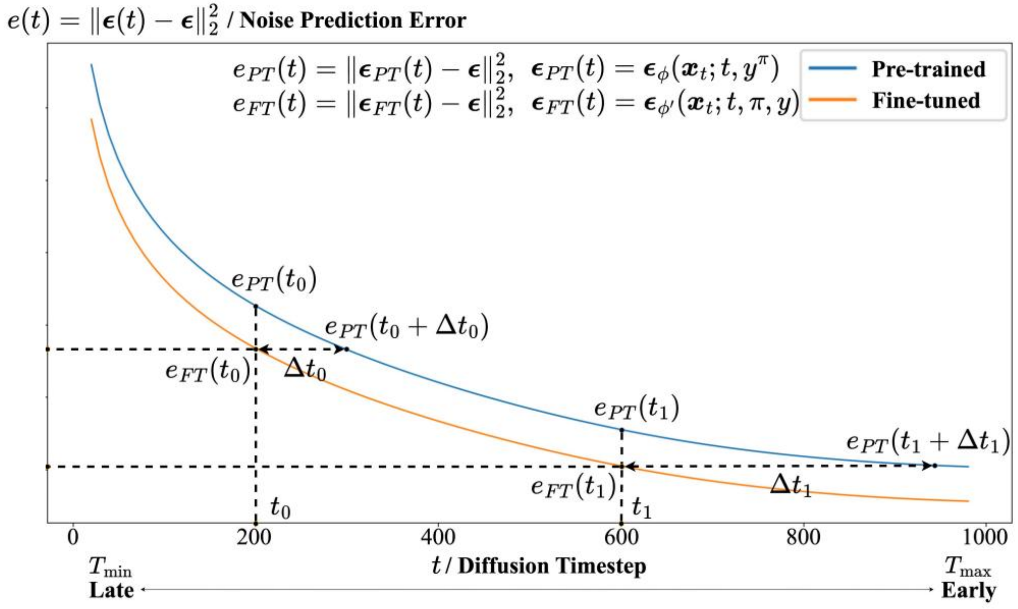
Method

Observation

- Finetuned Diffusion model predict more accurate noise

- Noise prediction error will decrease as timestep increase, both on original UNet and finetuned UNet

- The speed of decrease becomes slower and slower as timestep increase

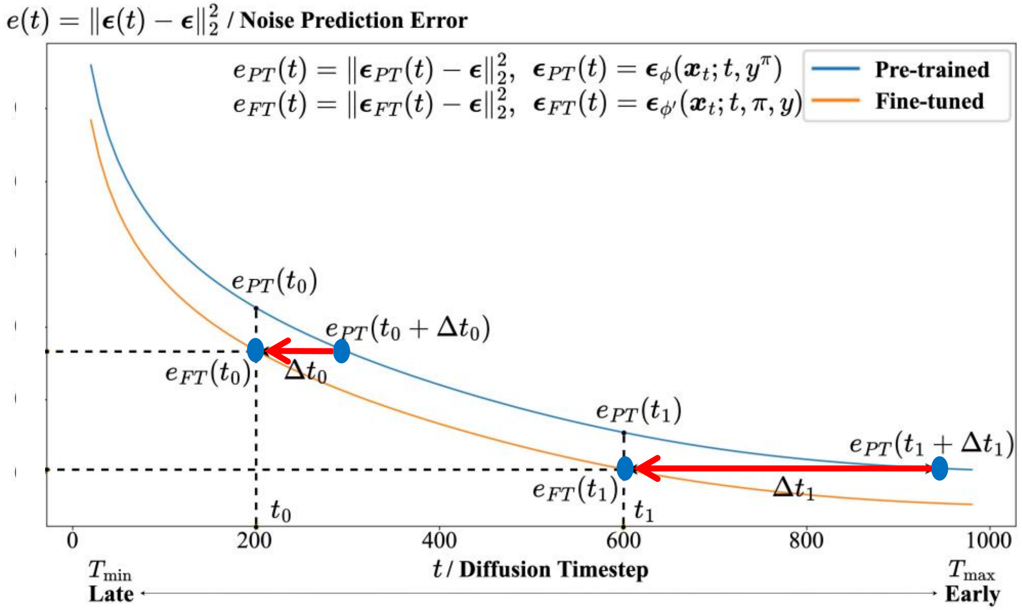


Method

Goal: Find a more accurate noise prediction on rendered images

Replace “ground truth noise” with noise prediction at larger timestep

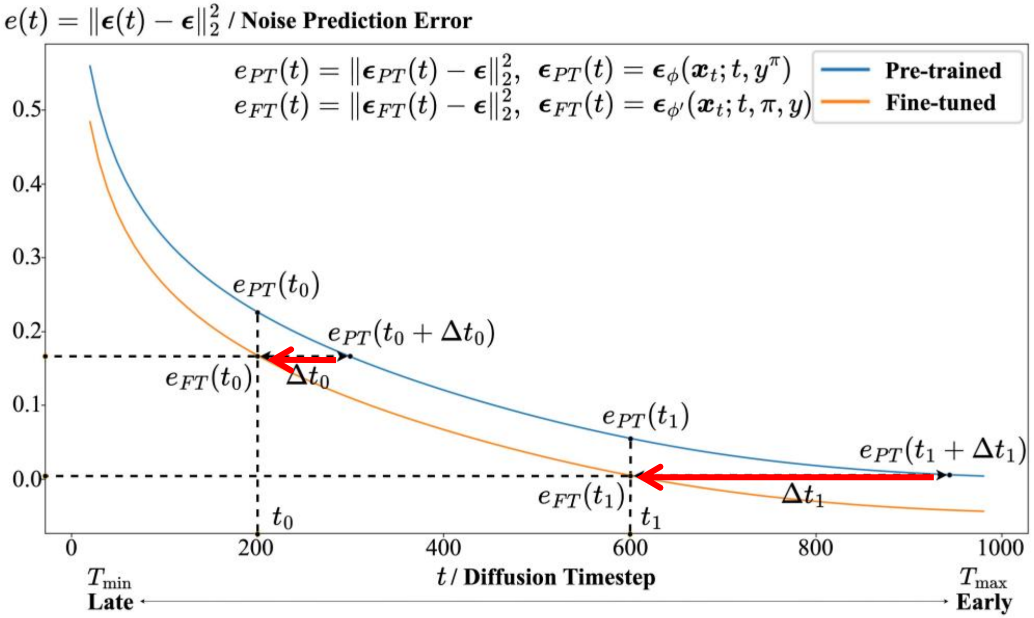
- More accurate prediction: $e_{PT}(t) \rightarrow e_{FT}(t)$
- Use $e_{PT}(t + \Delta t)$ approximate $e_{FT}(t)$



Method

A heuristic strategy to increase Δt with larger timestep:

- if $t_0 < t_1$, then $\Delta t_0 < \Delta t_1$
- Sample a timestep shift $\Delta t \sim S(t) = \mathcal{U}\left[0, \eta\left(t - T_{\min}\right)\right]$



Method

Comparison with VSD:

- No LoRA need to be trained, a single training objective
- Maintain most generative prior of pretrained diffusion model

Algorithm 1 Variational Score Distillation

Input: Number of particles n (≥ 1). Large text-to-image diffusion model $\epsilon_{\text{pretrain}}$. Learning rate η_1 and η_2 for 3D structures and diffusion model parameters, respectively. A prompt y.

initialize n 3D structures $\{\theta^{(i)}\}_{i=1}^{n}$, a noise prediction model ϵ_ϕ parameterized by ϕ.
while not converged **do**
  Randomly sample $\theta \sim \{\theta^{(i)}\}_{i=1}^{n}$ and a camera pose c.
  Render the 3D structure θ at pose c to get a 2D image $\boldsymbol{x}_0 = \boldsymbol{g}(\theta, c)$.
  $\theta \leftarrow \theta - \eta_1 \mathbb{E}_{t,\boldsymbol{\epsilon},c}\left[\omega(t)\left(\boldsymbol{\epsilon}_{\text{pretrain}}(\boldsymbol{x}_t, t, y^c) - \boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t, t, c, y)\right)\frac{\partial \boldsymbol{g}(\theta, c)}{\partial \theta}\right]$
  $\phi \leftarrow \phi - \eta_2 \nabla_\phi \mathbb{E}_{t,\epsilon}||\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t, t, c, y) - \boldsymbol{\epsilon}||_2^2$.
end while
return

Algorithm 1 Asynchronous Score Distillation (ASD)

Input: 3D representation θ; Text prompt y; Hyperparamter η; 2D diffusion prior $\boldsymbol{\epsilon}_\phi$
while *not converged* **do**
  Sample a camera pose π
  Render an image $\boldsymbol{x} = g(\theta, \pi)$
  Sample a timestep $t \sim \mathcal{U}[T_{\min}, T_{\max}]$, Gaussian noise $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \boldsymbol{I})$
  Sample a timestep shift $\Delta t \sim S(t) = \mathcal{U}\left[0, \eta\left(t - T_{\min}\right)\right]$
  $\boldsymbol{x}_t \leftarrow \alpha_t \boldsymbol{x} + \sigma_t \boldsymbol{\epsilon}$, $\boldsymbol{x}_{t+\Delta t} \leftarrow \alpha_{t+\Delta t}\boldsymbol{x} + \sigma_{t+\Delta t}\boldsymbol{\epsilon}$
  Update θ with $\Delta\theta \leftarrow \omega(t)\left(\boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_t; t, y^\pi\right) - \boldsymbol{\epsilon}_\phi\left(\boldsymbol{x}_{t+\Delta t}; t + \Delta t, y^\pi\right)\right)\frac{\partial \boldsymbol{x}}{\partial \theta}$
end

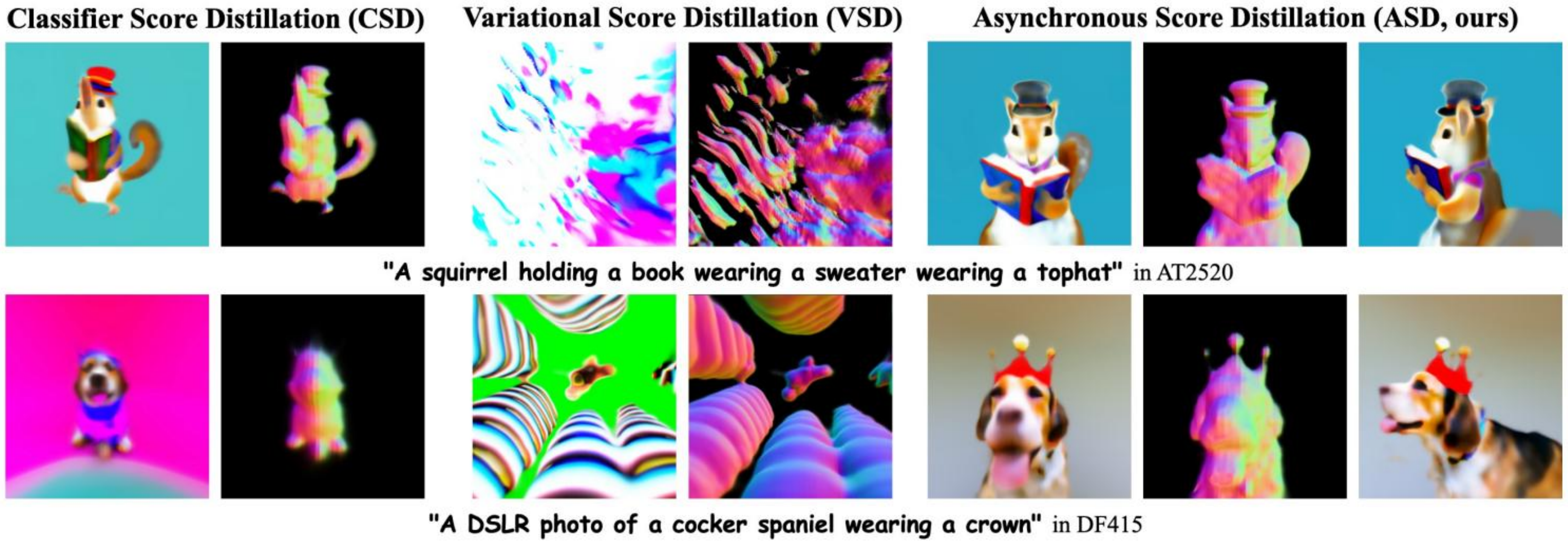
Outline

- Background
- Method
- Experiments
- Conclusion

Experiments

Prompt amortized: optimize a general 3D generator to produce 3D objects given different prompts

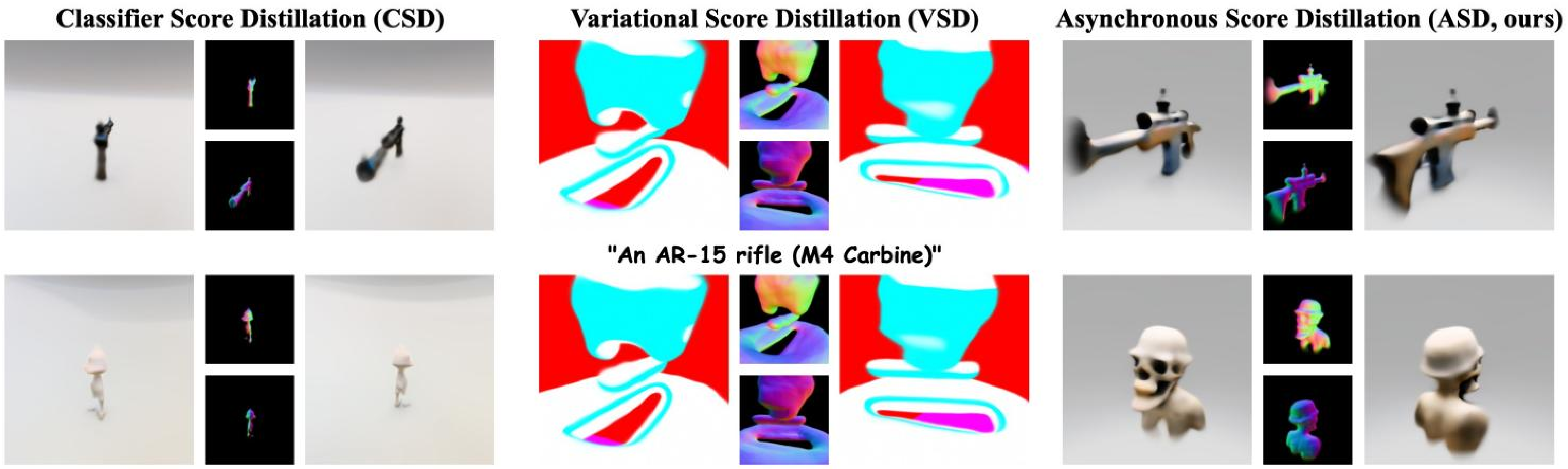
- More strict to prompt comprehensive ability



Experiments

Scalability: train a 3D generator on 100k prompts

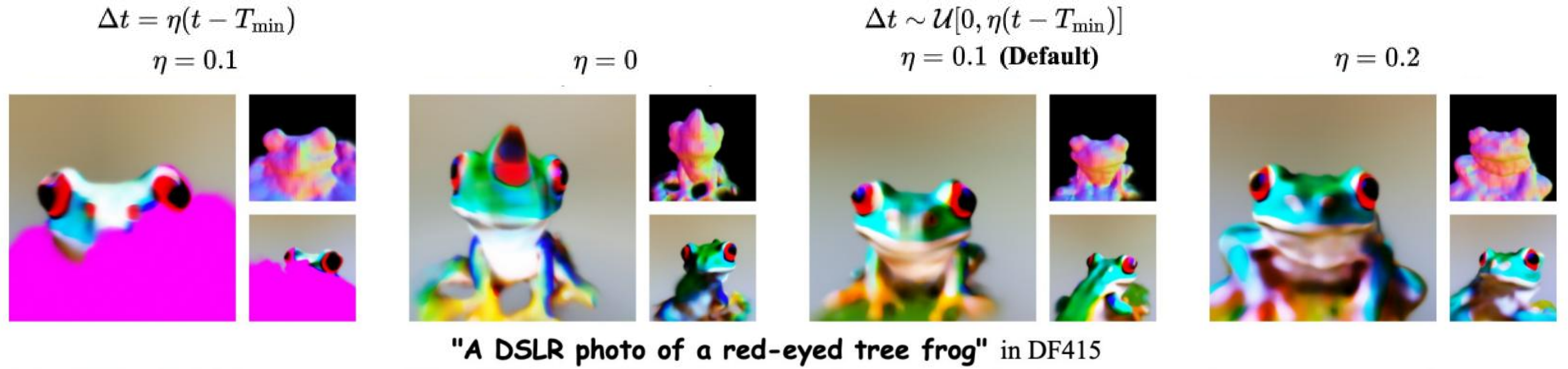
- VSD will be crashed
- ASD can generator more vivid results compared with CSD



Experiments: Ablation

Different setting of Δt

- No random sampling: not work
- Too big η: $\boldsymbol{\epsilon}_\phi(\boldsymbol{x}_t; t, y^\pi) \approx \boldsymbol{\epsilon}$, degrade to SDS, which is not suitable with CFG=7.5

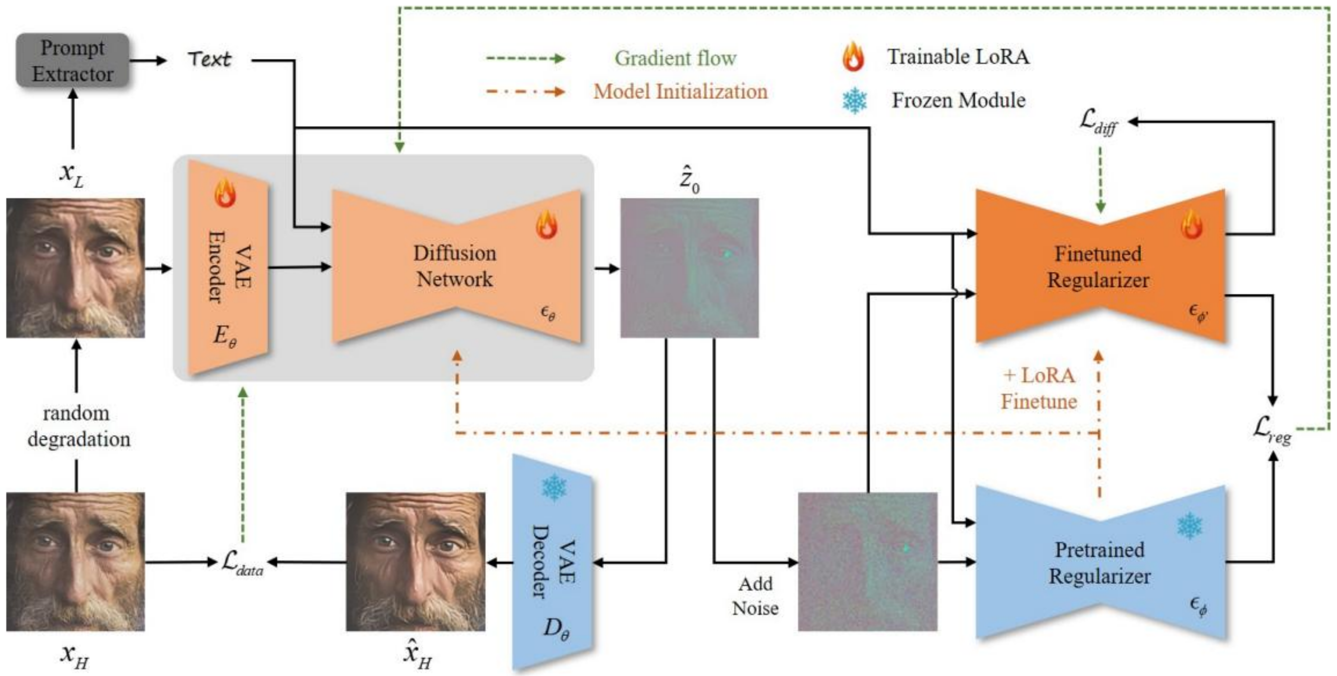


Outline

- Background
- Method
- Experiments
- Conclusion

Conclusion: Potential Extension

OSEDiff (arXiv 24.06): utilize VSD Loss on **one step** RealSR task
(Comes from the same team)



No noise input: stable, high fidelity

Conclusion

- Regularize 3D generation by 2D pretrained Diffusion models effectively and efficiently
- Improve comprehension ability with scalable prompts
- Potential applications to more vision tasks

Thanks for listening!