DemoFusion: Democratising High-Resolution Image Generation With No $$$

CVPR 2024

Ruoyi Du   Dongliang Chang   Timothy M. Hospedales   Yi-Zhe Song   Zhanyu Ma
PRIS, Beijing University of Posts and Telecommunications   Tsinghua University
University of Edinburgh SketchX   University of Surrey

STRUCT Group Seminar
Presenter: Lan Xicheng
2024.11.24

Outline

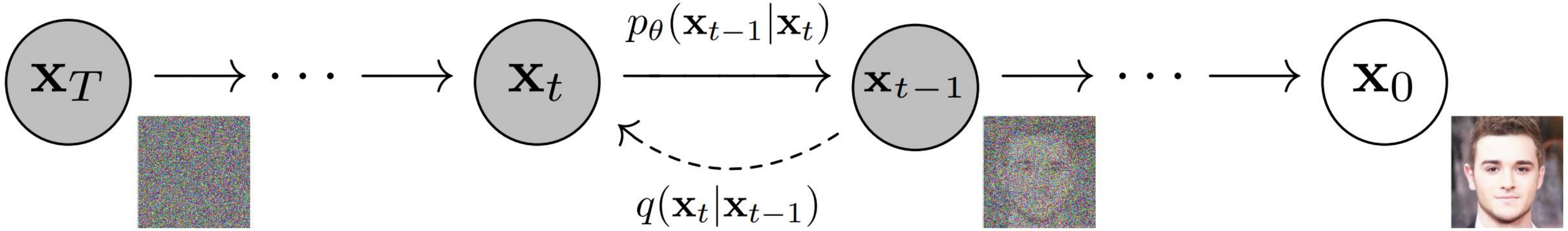
- Authors
- Background
- Methods
- Experiments
- Conclusion

Outline

- Authors
- Background
- Methods
- Experiments
- Conclusion

Background

Diffusion Models



Algorithm 1 Training

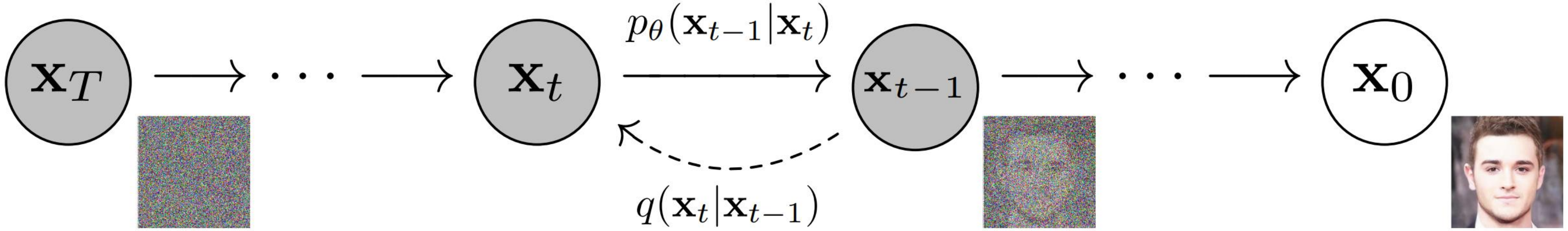
1: **repeat**
2: $\mathbf{x}_0 \sim q(\mathbf{x}_0)$
3: $t \sim \mathrm{Uniform}(\{1, \ldots, T\})$
4: $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
5: Take gradient descent step on
$$\nabla_\theta \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}, t) \right\|^2$$
6: **until** converged

Algorithm 2 Sampling

1: $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
2: **for** $t = T, \ldots, 1$ **do**
3: $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
4: $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$
5: **end for**
6: **return** $\mathbf{x}_0$

Background

Diffusion Models



$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) := \mathcal{N}(\mathbf{x}_{t-1}; \boldsymbol{\mu}_\theta(\mathbf{x}_t, t), \boldsymbol{\Sigma}_\theta(\mathbf{x}_t, t)) \qquad (1)$$

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

Background

Diffusion Models

Training Requirement ⟶ Paywalls

Model	Training GPUs	Training Time (Estimated)	Number of Parameters	Base Generation Resolution
SD1.5	A100	256*20d	1B	512^2
SDXL	A100	256*50d	2.3B	1024^2
FLUX	A100	1000*120d	12B	1024^2-2048^2

Background

Super-resolution Models

Real-ESRGAN

Low-resolution

High-resolution

Background

Super-resolution Models

SD-x2-latent-upscaler

Low-resolution

High-resolution

Background

Super-resolution Models

- Faithfully enhance the resolution according to the original image
- It is difficult to add corresponding details at higher resolutions

(a) Low-resolution (b) Upscaler (c) ReLife

Background

Objective: Generate Higher-resolution Images

- Directly prompting SDXL to generate images at a resolution of 2048^2 failed
  - The base model of SDXL lacks the ability to directly sample from a higher-resolution latent space
- The base SDXL has learned details at higher resolutions

Background

Objective: Generate Higher-resolution Images

- Directly prompting SDXL to generate images at a resolution of 2048^2 failed
- The base SDXL has learned details at higher resolutions
    - Observing the results of SDXL image generation experiments, occasional incomplete images may appear in some regions
    - The presence of partial images in the training set, or some training samples being cropped from complete higher-resolution images

Background

MultiDiffusion

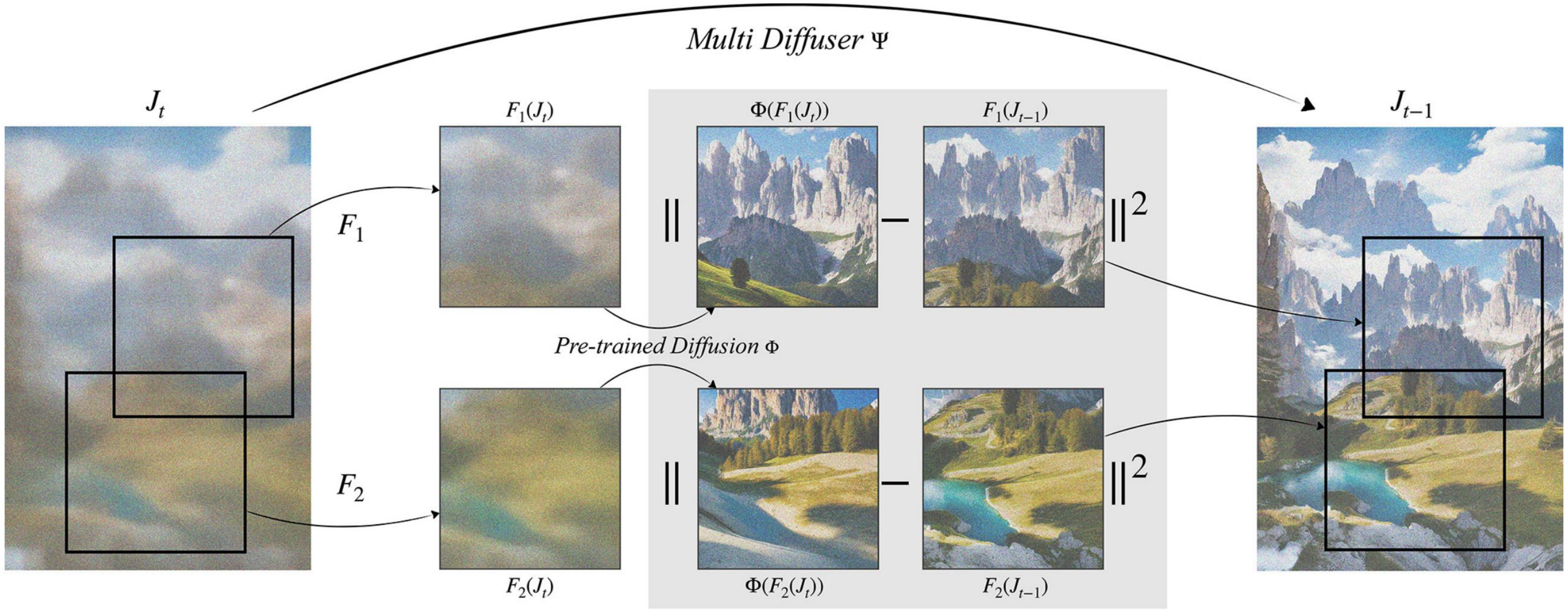
MultiDiffusion: Fusing Diffusion Paths for Controlled Image Generation

Omer Bar-Tal [*1] Lior Yariv [*1] Yaron Lipman [1 2] Tali Dekel [1]

Background

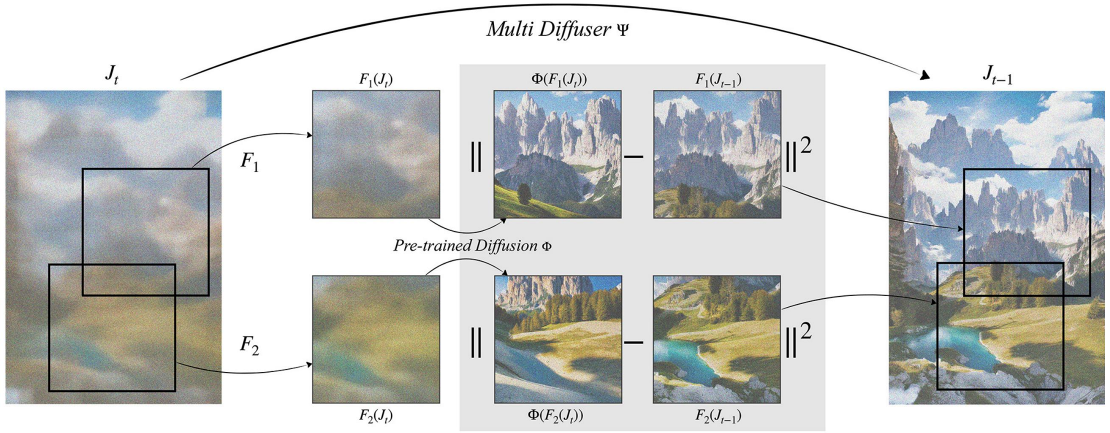
MultiDiffusion

- Fusion of multiple denoising processes
- Generate images of arbitrary size and resolution



Background

MultiDiffusion



Reference Model

$$\Phi : \mathcal{I} \times \mathcal{Y} \rightarrow \mathcal{I}$$

$$I_T, I_{T-1}, \ldots, I_0 \quad \text{s.t.} \quad I_{t-1} = \Phi(I_t|y)$$

$\mathcal{I} = \mathbb{R}^{H \times W \times C}$ Image Space

$y \in \mathcal{Y}$ Condition Space

MultiDiffusion

$$\Psi : \mathcal{J} \times \mathcal{Z} \rightarrow \mathcal{J}$$

$$J_T, J_{T-1}, \ldots, J_0 \quad \text{s.t.} \quad J_{t-1} = \Psi(J_t|z)$$

$\mathcal{J} = \mathbb{R}^{H' \times W' \times C}$ Image Space

$\mathcal{Z}$ Condition Space

$$F_i : \mathcal{J} \rightarrow \mathcal{I}, \lambda_i : \mathcal{Z} \rightarrow \mathcal{Y},\ i \in [n] = \{1, \ldots, n\}$$ **(Application Depended)**

$$I_t^i = F_i(J_t), \quad y_i = \lambda_i(z)$$

Background

MultiDiffusion

$F_i : \mathcal{J} \rightarrow \mathcal{I}, \lambda_i : \mathcal{Z} \rightarrow \mathcal{Y}, \; i \in [n] = \{1, \ldots, n\}$ **(Application Depended)**

$I_t^i = F_i(J_t), \quad y_i = \lambda_i(z)$

$$\Psi(J_t|z) = \arg\min_{J \in \mathcal{J}} \; \mathcal{L}_{\text{FTD}}(J|J_t, z)$$

$$\mathcal{L}_{\text{FTD}}(J|J_t, z) = \sum_{i=1}^{n} \left\| W_i \otimes \left[F_i(J) - \Phi(I_t^i|y_i) \right] \right\|^2$$

Algorithm 1 MultiDiffusion sampling.

Input : Φ ▷ pre-trained Diffusion Model
$\{F_i\}_{i=1}^n$ ▷ image space mappings
$\{y_i\}_{i=1}^n$ ▷ text-prompts conditioning
$\{W_i\}_{i=1}^n$ ▷ per-pixel weights
$J_T \sim P_{\mathcal{J}}$ ▷ noise initialization
for $t = T, ..., 1$ **do**
  $I_{t-1}^i \leftarrow \Phi(F_i(J_t), y_i) \;\; \forall i \in [n]$ ▷ diffusion updates
  $J_{t-1} \leftarrow \texttt{MultiDiffuser}(\{I_{t-1}^i\}_{i=1}^n)$ ▷ Eq. 5
Output : J_0

F_i consist of direct pixel samples, thus L is a quadratic Least-Squares:

$$\Psi(J_t|z) = \sum_{i=1}^{n} \frac{F_i^{-1}(W_i)}{\sum_{j=1}^{n} F_j^{-1}(W_j)} \otimes F_i^{-1}(\Phi(I_t^i|y_i))$$

$W_i \in \mathbb{R}_{\geq 0}^{H \times W}$	*Per Pixel Weights Application Depended*
$\otimes$	*Hadamard product*

Background

MultiDiffusion

(a) Generation with per-crop independent diffusion paths.

(b) Generation with fused diffusion paths using MultiDiffusion.

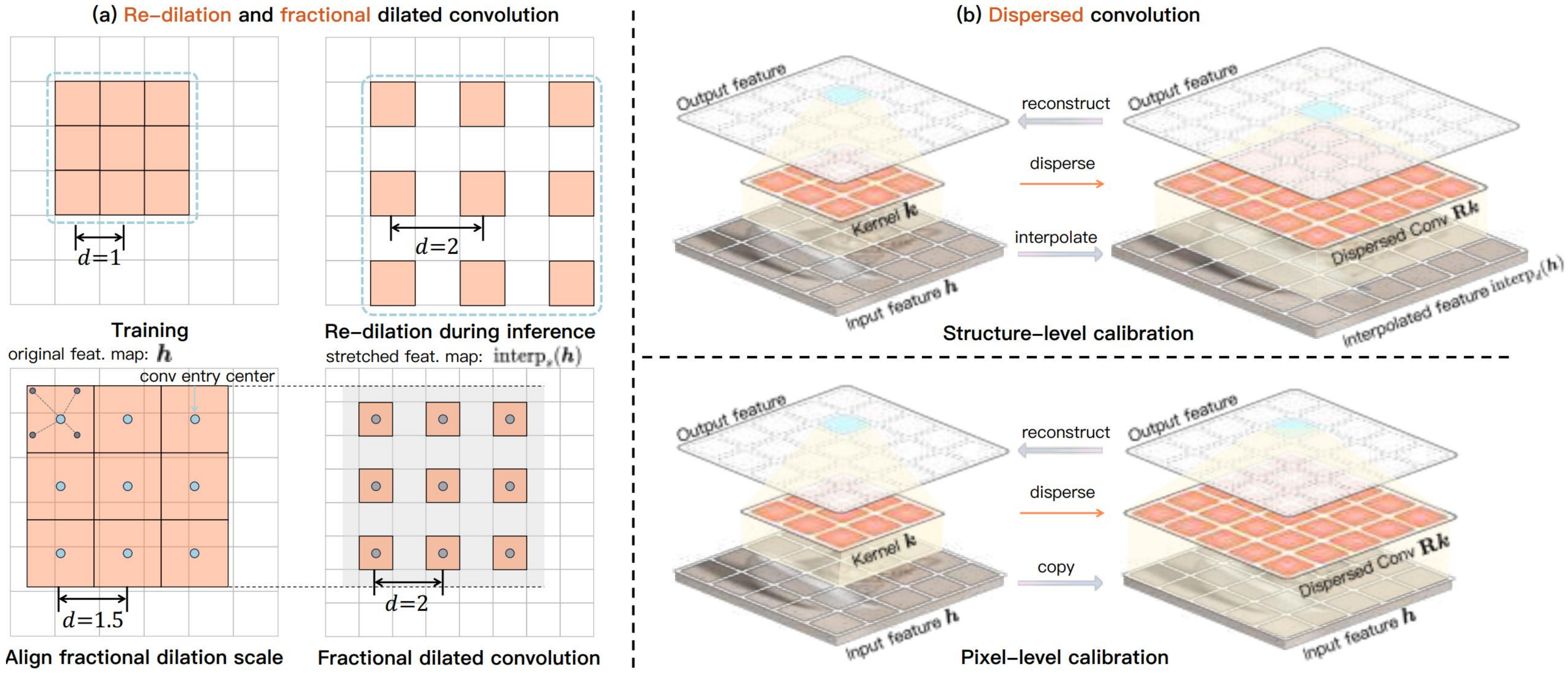
Background

MultiDiffusion

- Used for generating larger-sized images, with the central regions of each part being almost independently sampled
- For generating a single target object, the correlation between paths is weak, making it difficult to consider global semantics

Background

SCALECRAFTER



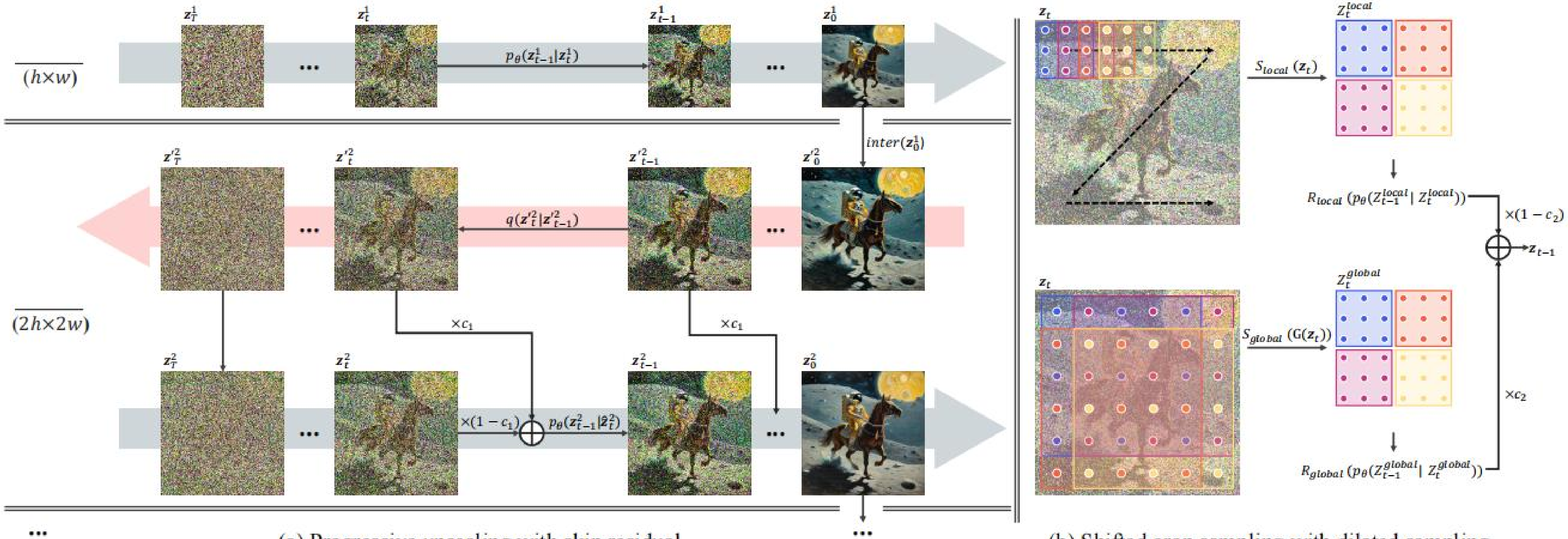
Outline

- Authors
- Background
- Methods
- Experiments
- Conclusion

Methods

Framework

·



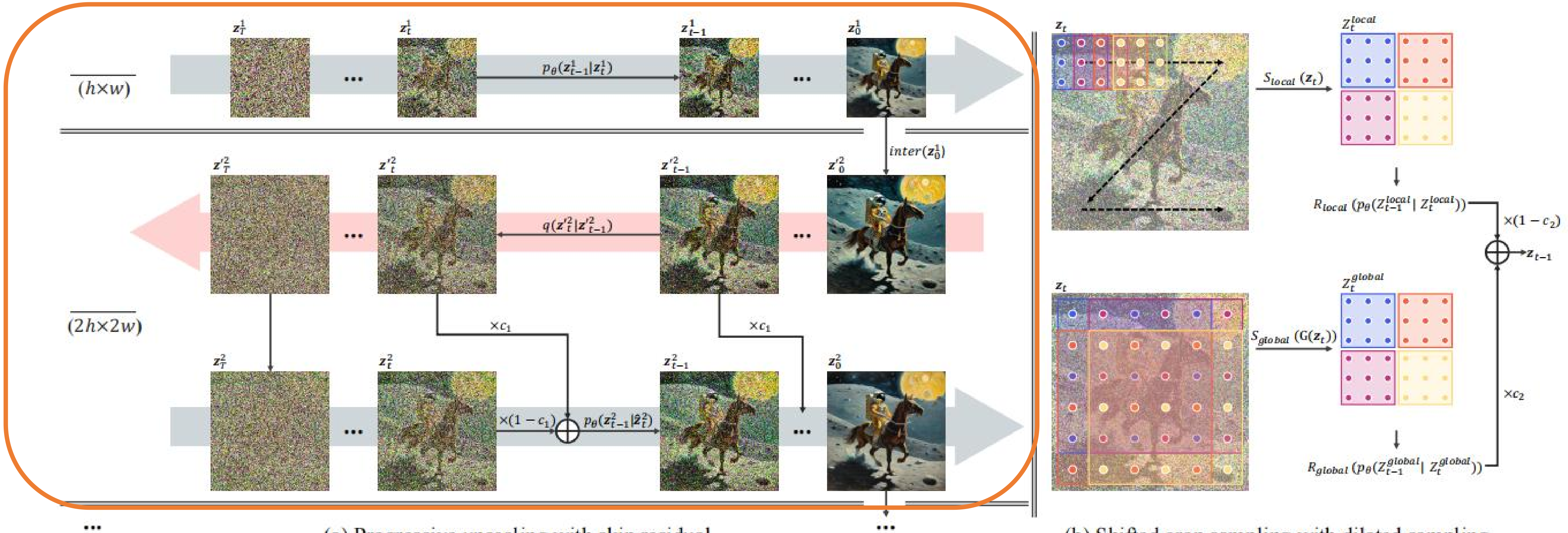
(a) Progressive upscaling with skip residual.

(b) Shifted crop sampling with dilated sampling.

Methods

Framework

Progressive Upscaling

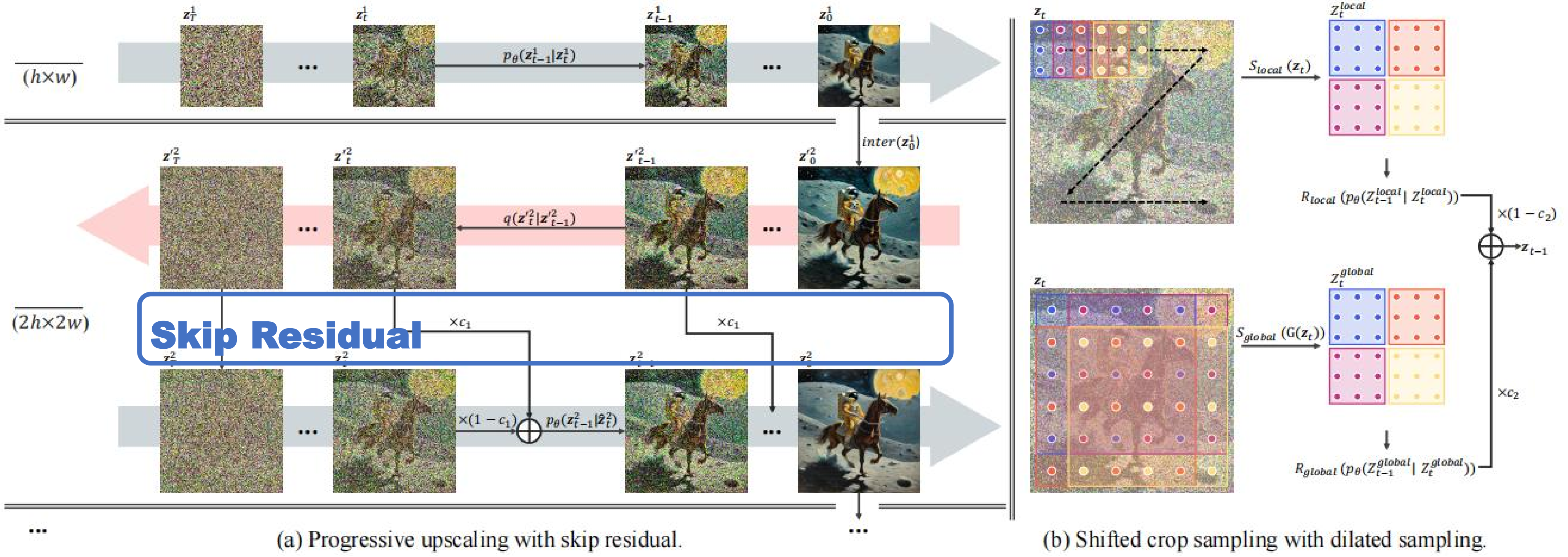


(a) Progressive upscaling with skip residual.

(b) Shifted crop sampling with dilated sampling.

Methods

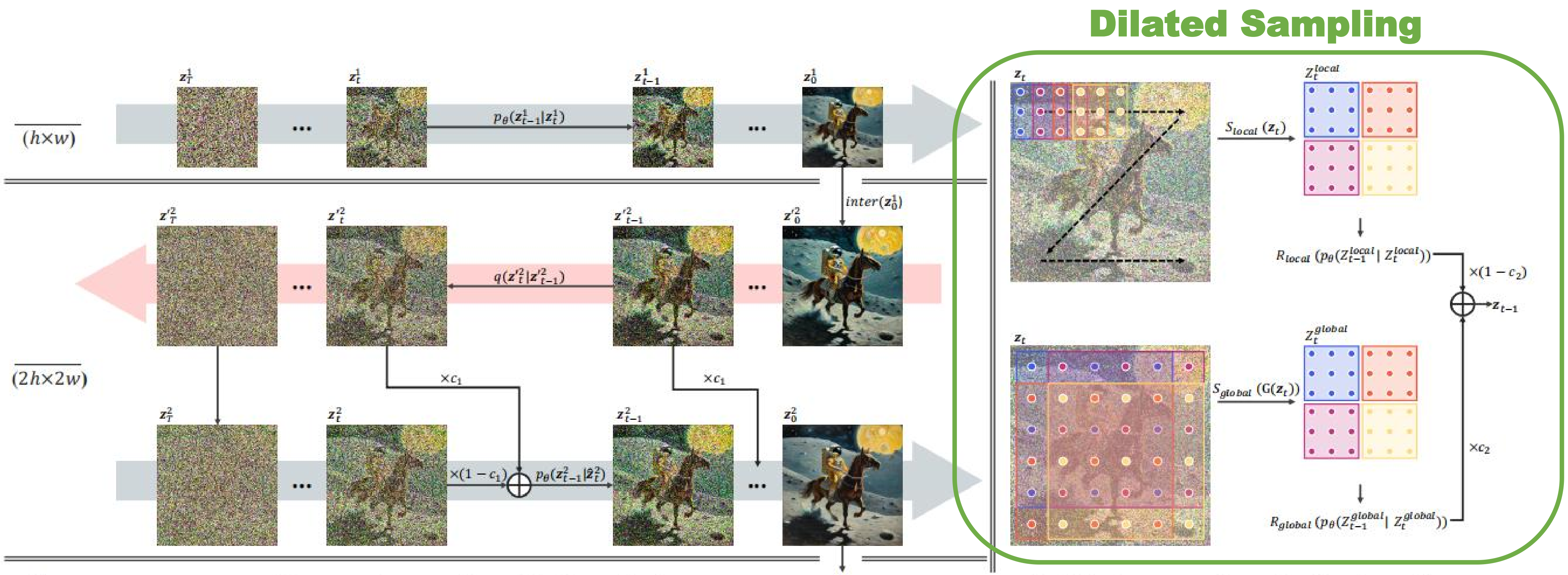
Framework



(a) Progressive upscaling with skip residual.
(b) Shifted crop sampling with dilated sampling.

Methods

Framework



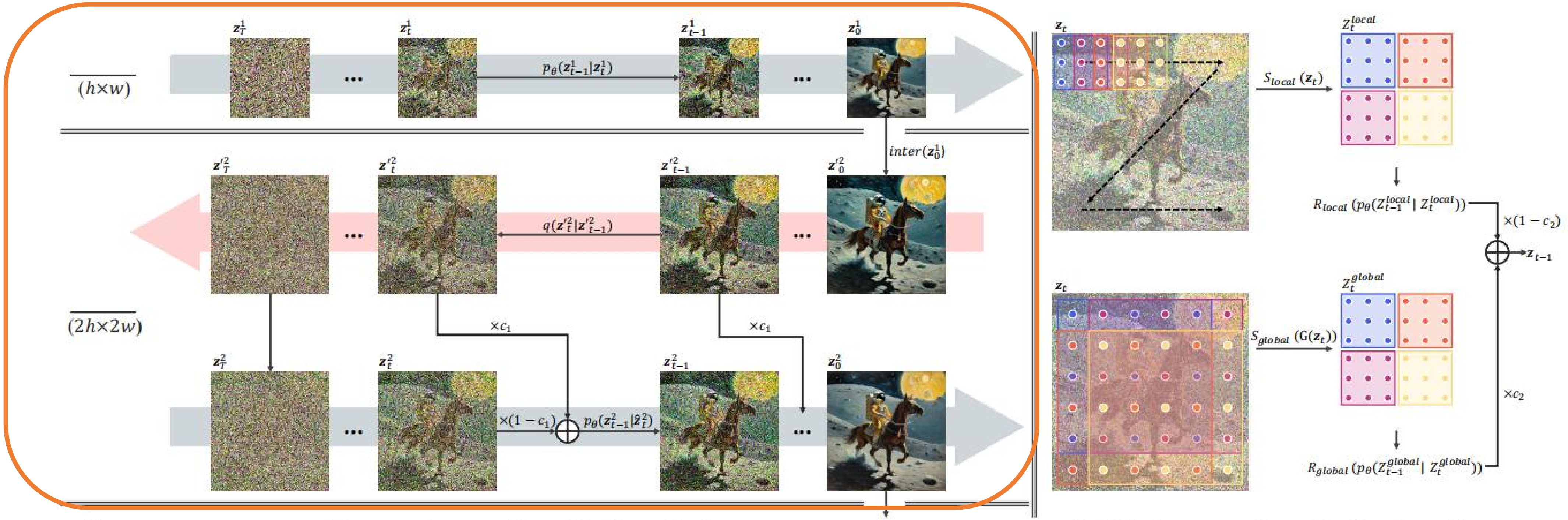
(a) Progressive upscaling with skip residual.

(b) Shifted crop sampling with dilated sampling.

Methods

Progressive Upscaling

.



(a) Progressive upscaling with skip residual.

(b) Shifted crop sampling with dilated sampling.

Methods

Progressive Upscaling

Generate images with progressively higher resolutions in steps

$\mathbf{z}_T^1$ → $\mathbf{z}_0^s$

$\mathbb{R}^{c\times h\times w}$ → $\mathbb{R}^{c\times H\times W}$

K: Factor Magnified

$S = \sqrt{K}$

$H = Sh$ and $W = Sw$

as $q(\mathbf{z}_T|\mathbf{z}_0) = \prod_{t=1}^{T} q(\mathbf{z}_t|\mathbf{z}_{t-1})$ and $p_\theta(\mathbf{z}_0|\mathbf{z}_T) = \prod_{t=T}^{1} p_\theta(\mathbf{z}_{t-1}|\mathbf{z}_t)$

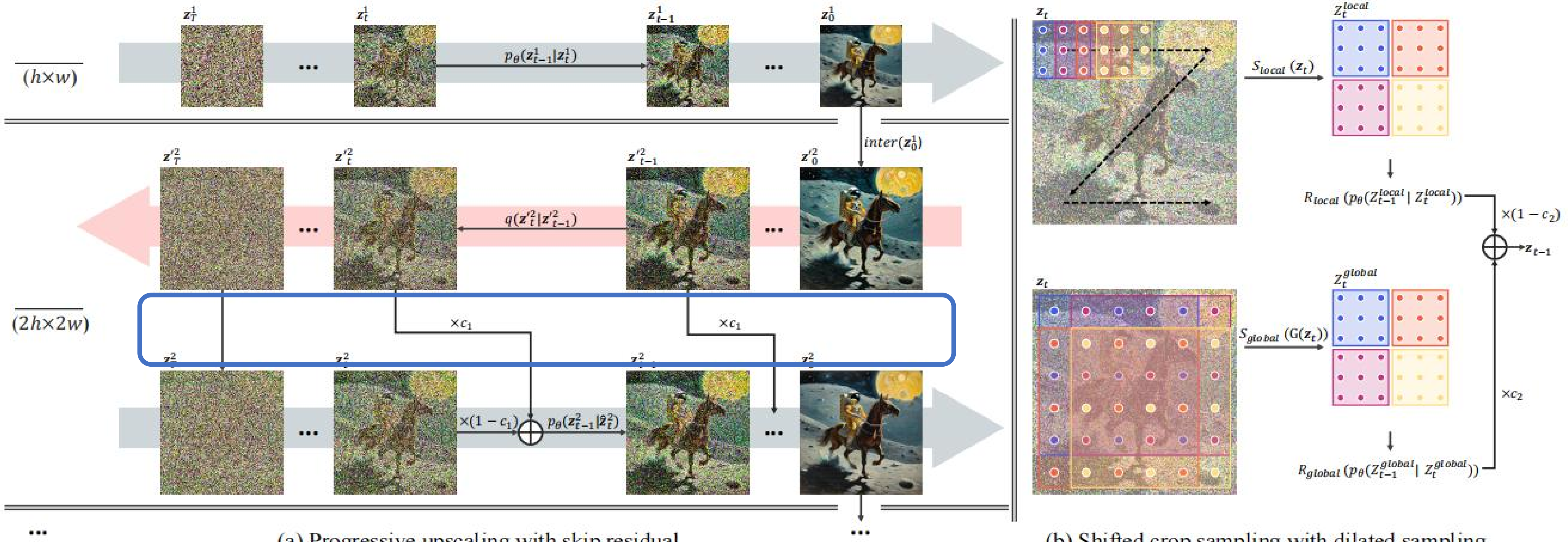
$$p_\theta(\mathbf{z}_0^S|\mathbf{z}_T^1) = p_\theta(\mathbf{z}_0^1|\mathbf{z}_T^1)\prod_{s=2}^{S}(q(\mathbf{z'}_T^s|\mathbf{z'}_0^s)p_\theta(\mathbf{z}_0^s|\mathbf{z'}_T^s))$$

$\mathbf{z'}_0^s = inter(\mathbf{z}_0^{s-1})$ $inter(\cdot)$ is an arbitrary interpolation algorithm

Methods

Skip Residual

.



(a) Progressive upscaling with skip residual.

(b) Shifted crop sampling with dilated sampling.

Methods

Skip Residual -as an optimization of SDEdit

Why use edit in such scenarios

- To obtain more image details
- Without changing the original structure of the image

Issues with edit

Methods

Skip Residual -as an optimization of SDEdit

Why use edit in such scenarios

Issues with edit: Intersection Time-step

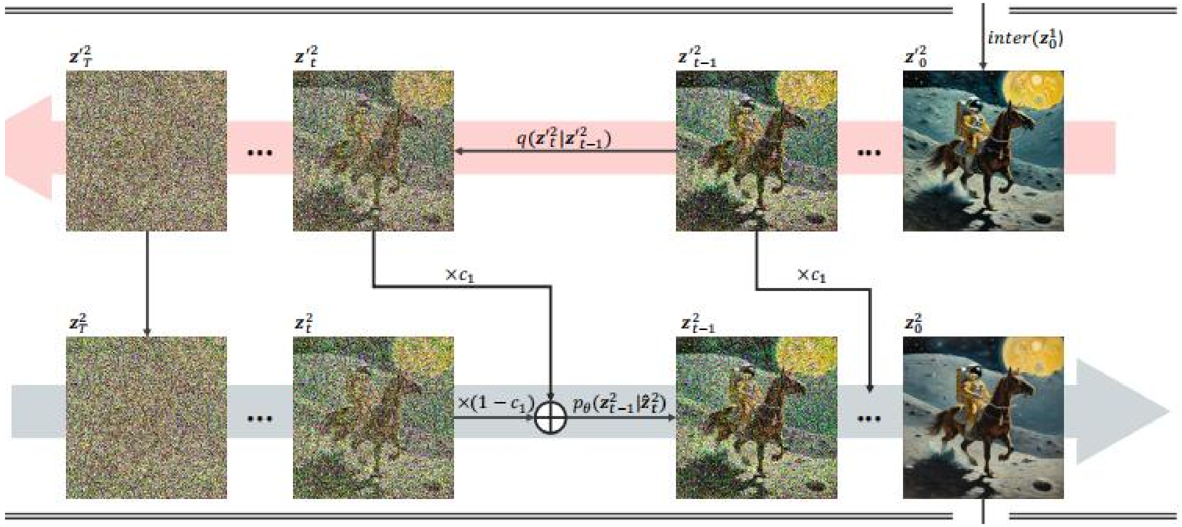
- Attempting to reverse-engineer the initial noise, but facing challenges, so Gaussian noise is directly added
- Too low noise intensity leads to insignificant effects
- Too high noise intensity causes loss of key information

Methods

Skip Residual -as an optimization of SDEdit

$$\hat{\mathbf{z}}_t^s = c_1 \times \mathbf{z'}_t^s + (1 - c_1) \times \mathbf{z}_t^s$$

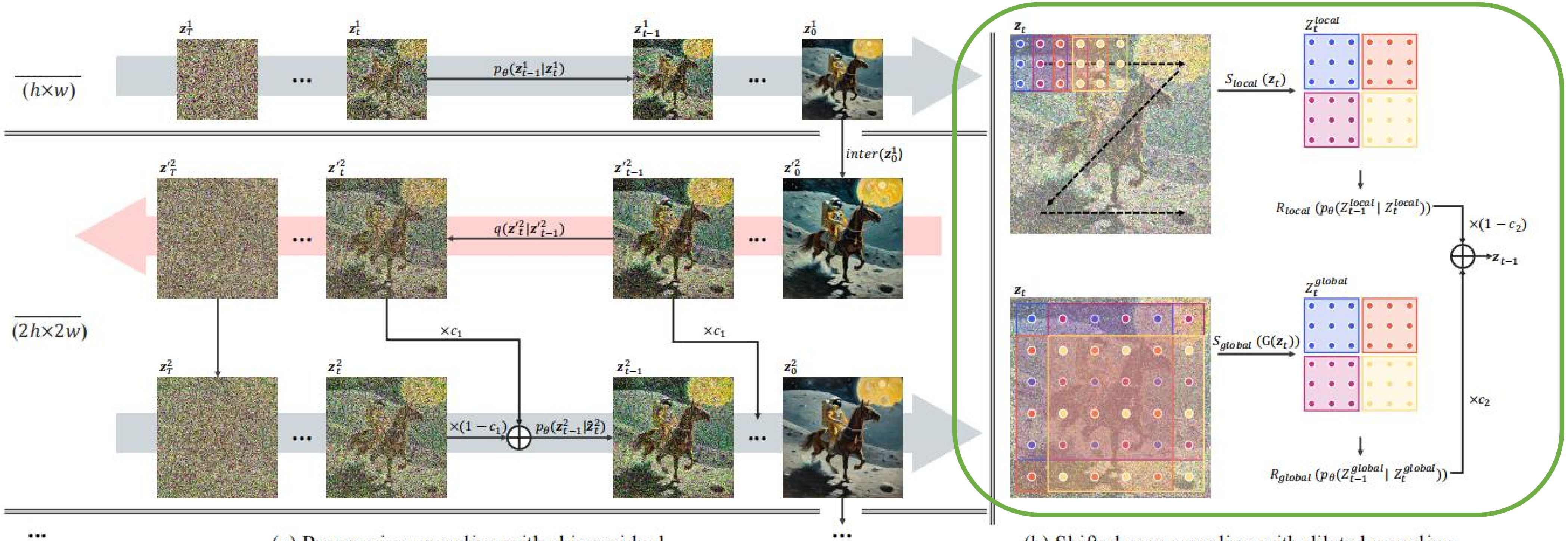
$$c_1 = ((1 + \cos\left(\frac{T-t}{T} \times \pi\right))/2)^{\alpha_1}$$



Methods

Dilated Sampling

.



(a) Progressive upscaling with skip residual.

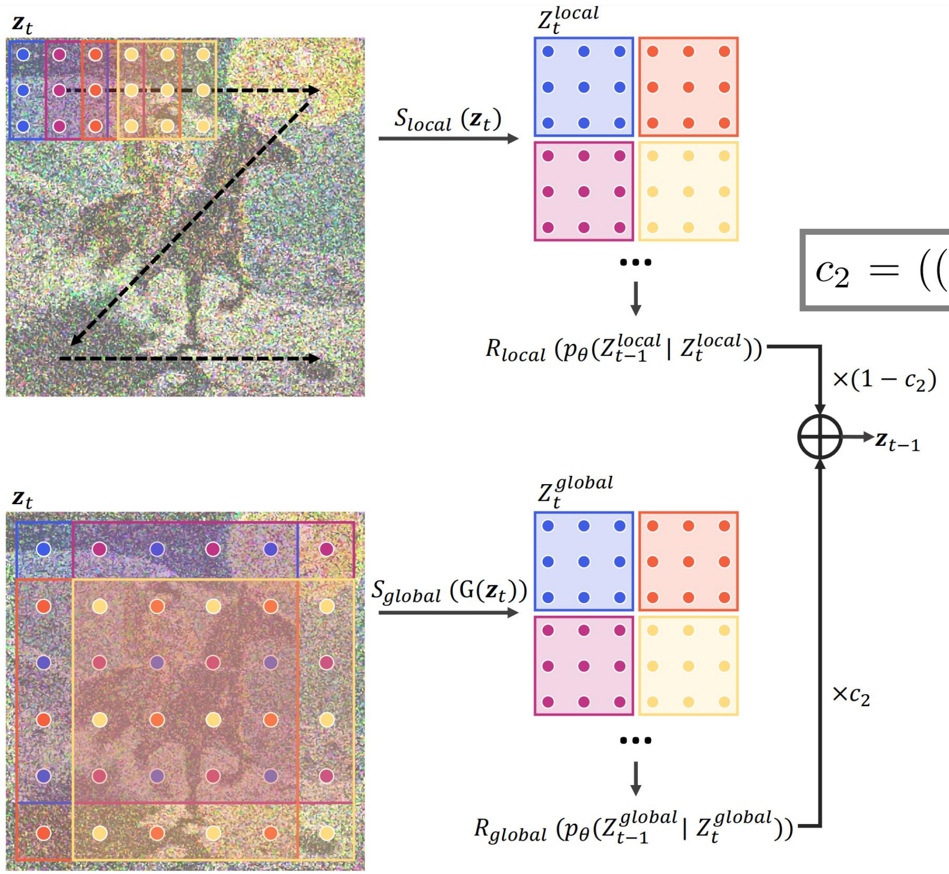
(b) Shifted crop sampling with dilated sampling.

Methods

Dilated Sampling

.

Shifted Sampling



$$c_2 = ((1 + \cos(\frac{T-t}{T} \times \pi))/2)^{\alpha_2}$$

Dilated Sampling

$$Z_t^{global} = [\mathbf{z}_{0,t}, \cdots, \mathbf{z}_{m,t}, \cdots, \mathbf{z}_{M,t}] = \mathcal{S}_{global}(\mathbf{z}_t)$$

$$\mathbf{z}_{m,t} \in \mathbb{R}^{c \times h \times w}$$

$$M = s^2$$

Methods

Dilated Sampling

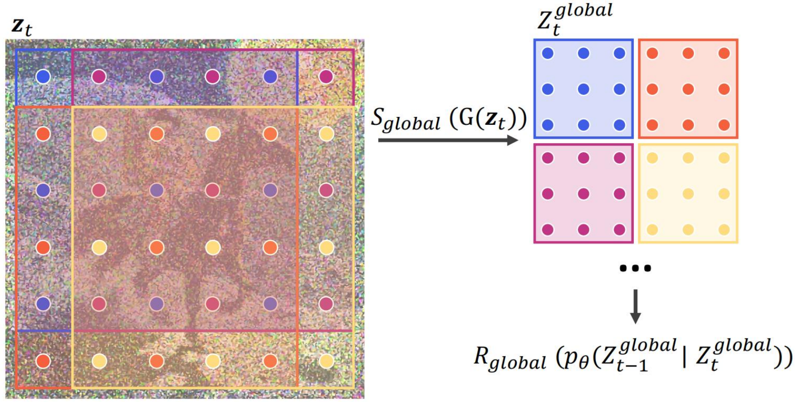
.

Dilated Sampling

$$Z_t^{global} = [\mathbf{z}_{0,t}, \cdots, \mathbf{z}_{m,t}, \cdots, \mathbf{z}_{M,t}] = \mathcal{S}_{global}(\mathbf{z}_t)$$

$$\mathbf{z}_{m,t} \in \mathbb{R}^{c \times h \times w}$$

$$M = s^2$$



- *No overlapping regions between different samples*
- *Introduce a Gaussian filter:*

$$Z_t^{global} = \mathcal{S}_{global}(\mathcal{G}(\mathbf{z}_t))$$

$$\text{kernel size} = 4s - 3$$

Outline

- Authors
- Background
- Methods
- Experiments
- Conclusion

Experiments

Baselines

- SDXL
- MultiDiffusion: Baseline method based on overlapped local patch denoising
- SDXL+BSRGAN: Directly upscale SDXL results
- SCALECRAFTER: Dilate convolutional kernels at specific layers

Experiments

Quantitative Results

Method	2048 × 2048					
	FID ↓	IS ↑	FID_{crop} ↓	IS_{crop} ↑	CLIP ↑	Time
SDXL Direct Inference [24]	79.66	13.47	73.91	17.38	28.12	1 min
MultiDiffusion [2]	75.93	14.56	70.93	17.85	28.97	3 min
SDXL + BSRGAN [39]	66.41	16.22	67.42	21.11	29.61	1 min
SCALECRAFTER [7]	69.91	15.72	68.36	19.44	29.51	1 min
DemoFusion (Ours)	**65.73**	**16.41**	**64.81**	**21.40**	**29.68**	3 min

2048 × 4096						4096 × 4096					
FID ↓	IS ↑	FID_{crop} ↓	IS_{crop} ↑	CLIP ↑	Time	FID ↓	IS ↑	FID_{crop} ↓	IS_{crop} ↑	CLIP ↑	Time
97.08	14.12	96.41	18.01	27.29	3 min	105.65	14.01	98.59	19.47	25.64	8 min
89.38	14.17	82.78	18.87	28.66	6 min	97.98	13.84	79.45	19.73	28.62	15 min
68.70	16.29	75.03	21.76	29.01	1 min	**66.44**	**16.21**	77.20	22.42	**29.63**	1 min
80.16	15.29	83.08	19.56	28.87	6 min	87.50	15.20	84.36	20.32	29.04	19 min
73.15	**16.37**	**71.35**	**23.55**	**29.05**	11 min	74.11	16.11	**70.34**	**24.28**	29.57	25 min

Experiments

Qualitative Results

Prompt: ***Emma Watson as a powerful mysterious sorceress, casting lightning magic, detailed clothing.***

SDXL

Experiments

Qualitative Results

MultiDiffusion

Experiments

Qualitative Results

SDXL+BSRGAN

Experiments

Qualitative Results

SCALECRAFTER

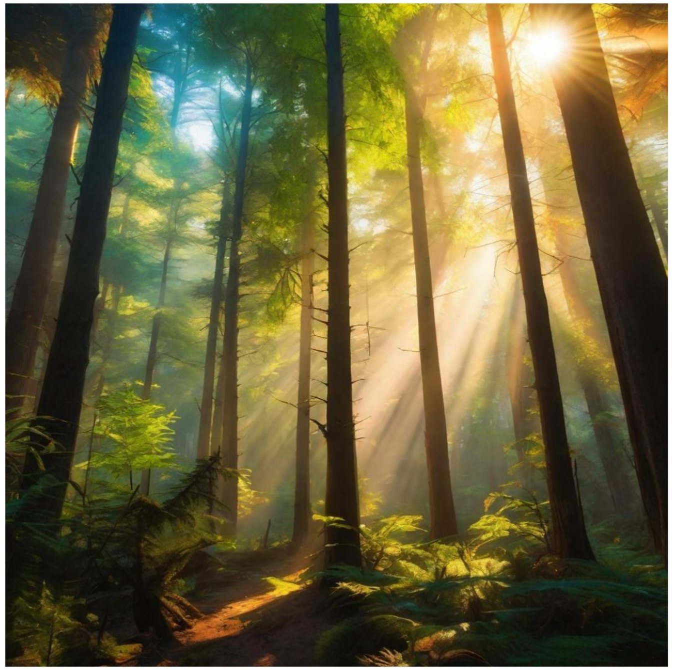
Experiments

Qualitative Results

DemoFusion

Experiments

Qualitative Results



Prompt: ***Primitive forest, towering trees, sunlight falling, vivid colors*****.**

SDXL

Experiments

Qualitative Results

MultiDiffusion

Experiments

Qualitative Results

SDXL+BSRGAN

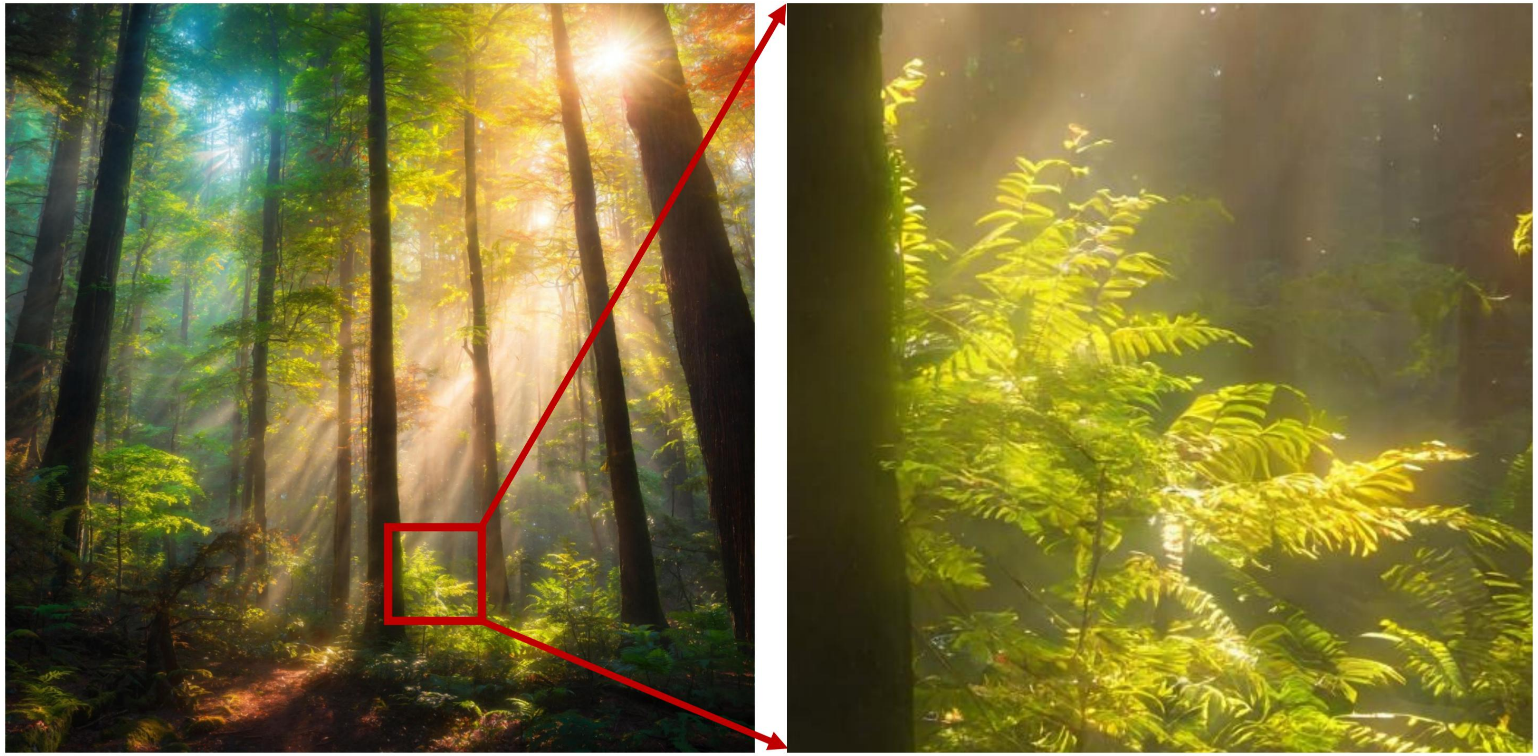
Experiments

Qualitative Results

SCALECRAFTER

Experiments

Qualitative Results



DemoFusion

Experiments

Ablations

Progressive Upscaling (PU) Skip Residual (SR) Dilated Upsampling (DS)

Experiments

Ablations

Progressive Upscaling (PU) Skip Residual (SR) Dilated Upsampling (DS)

Outline

- Authors
- Background
- Methods
- Experiments
- Conclusion

Conclusions

- Introduce a tuning-free framework that achieve higher-resolution image generation
- Enable generation with both global semantic coherence and rich local details
- Demonstrates the possibility of LDMs generating images at higher resolutions and the untapped potential of existing open-source GenAI models.

- Sampling takes a long time, heavily depends on the capabilities of the base model

Methods

Dilated Sampling

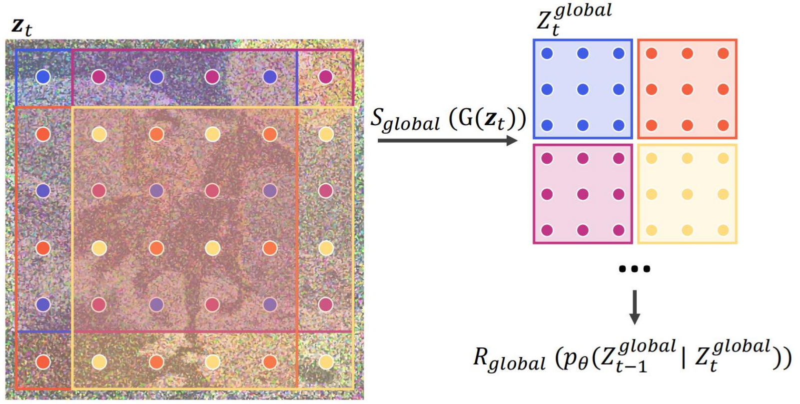
.

Dilated Sampling

$$Z_t^{global} = [\mathbf{z}_{0,t}, \cdots, \mathbf{z}_{m,t}, \cdots, \mathbf{z}_{M,t}] = \mathcal{S}_{global}(\mathbf{z}_t)$$

$$\mathbf{z}_{m,t} \in \mathbb{R}^{c \times h \times w}$$

$$M = s^2$$

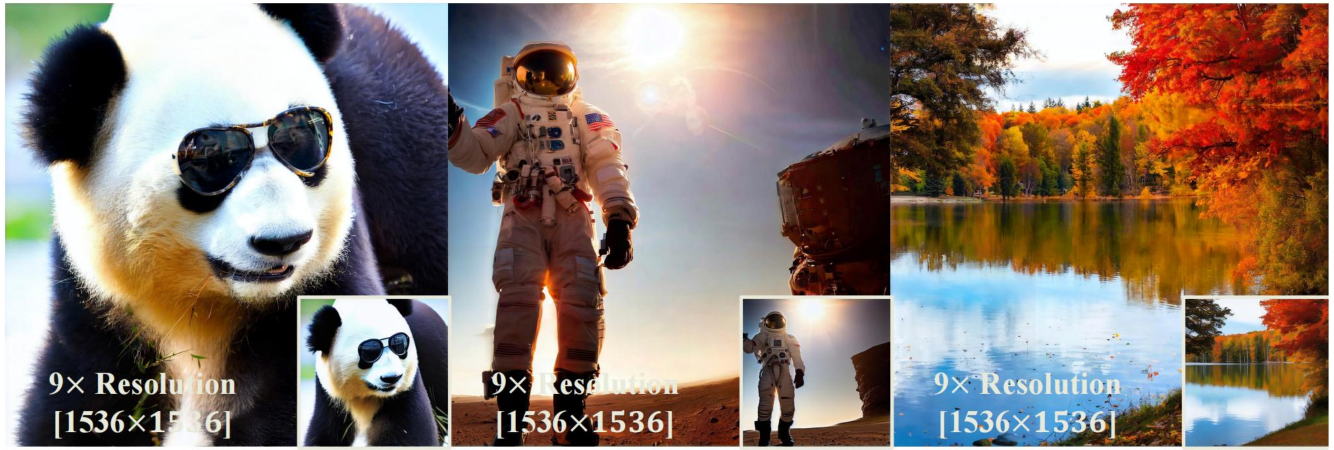


$$Z_t^{global} = \mathcal{S}_{global}(\mathcal{G}(\mathbf{z}_t))$$

$$\sigma_1 \text{ to } c_3 \times (\sigma_1 - \sigma_2) + \sigma_2$$

$$c_3 = ((1 + \cos\left(\frac{T-t}{T} \times \pi\right))/2)^{\alpha_3}$$

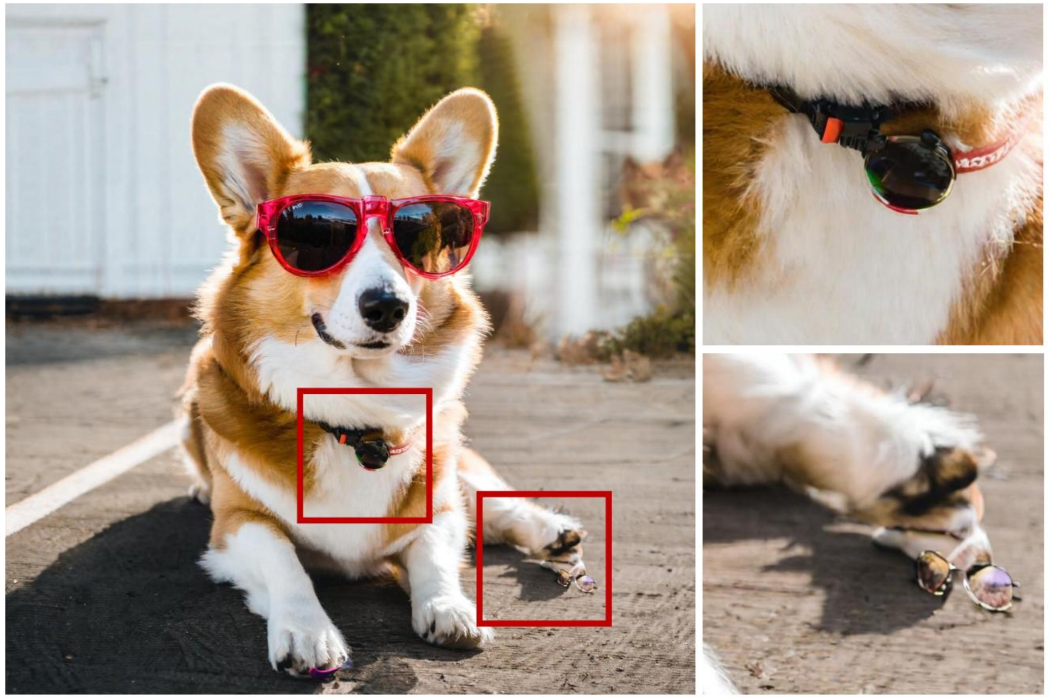
Other Results



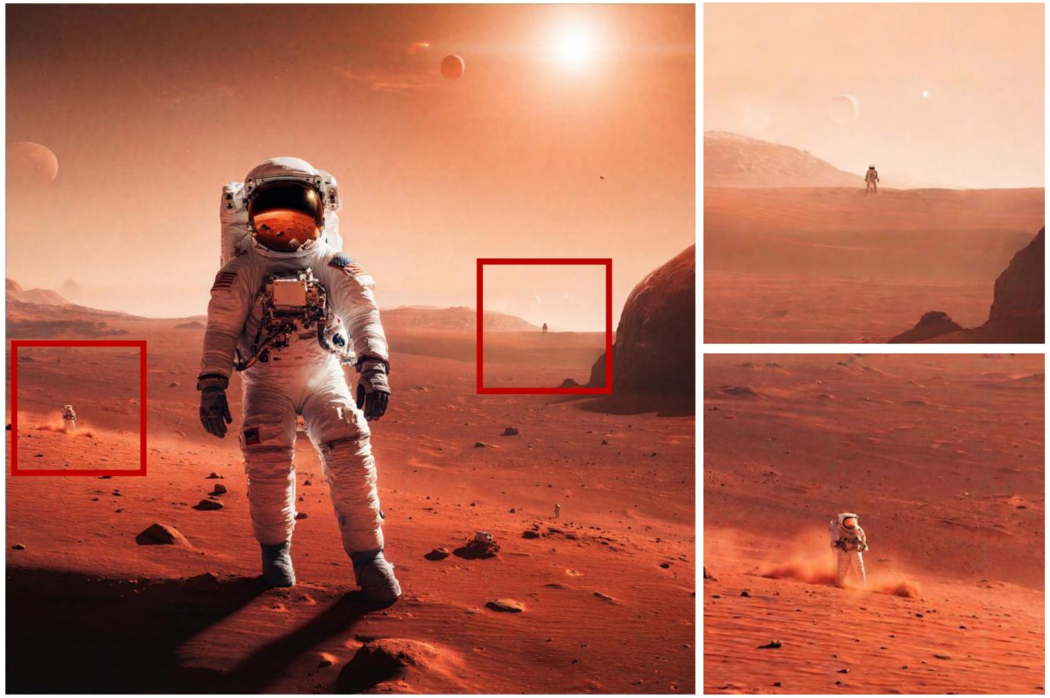
(a) Stable Diffusion 1.5

(b) Stable Diffusion 2.1

Failures



(a) Locally Unreasonable



(b) Small Object Repetition

Thank you for Listening!