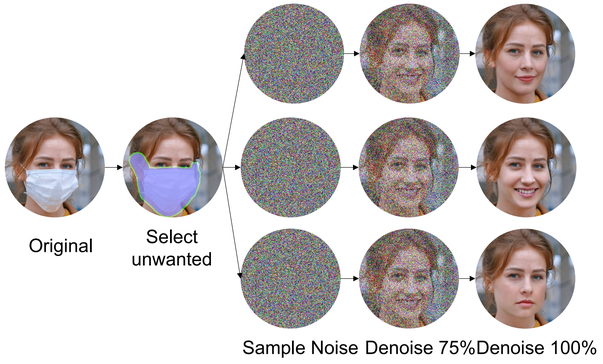
RePaint: Inpainting using Denoising Diffusion Probabilistic Models

Andreas Lugmayr, Martin Danelljan, Andres Romero, Fisher Yu, Radu Timofte, Luc Van Gool
Computer Vision Lab, ETH Zurich

GitHub
git.io/RePaint

Introduction



Motivation

- Current Autoregressive and GAN Inpainting Methods:
  - Limited generative capabilities lead to failure for large masks.
  - Design for specific masks lead to fail on sparse masks.
- Diffusion Models showed good generative capabilities.
- The conditioning process for Diffusion Model Inpainting lacked harmonization of the known and generated part.

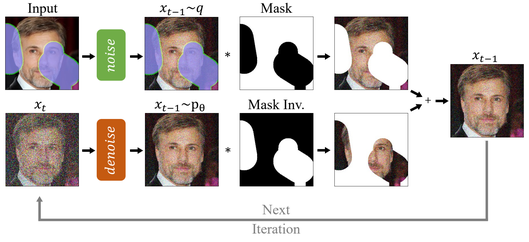
Contribution

- Method to condition an unconditionally trained Diffusion Models.
- Inference schedule generalizes to any inpainting mask.
- Generate semantically meaningful image completions.
- Harmonize generated and known part for inpainting.
- Analysis of inpainting algorithms on six different masks.

Method

Overview

- Condition inference of Diffusion Model
- No training or finetuning of the model
- Harmonization of known and generated content
  - Go forward and backward in diffusion time
  - Using larger jumps improve perceptual quality



Conditioning

$$x_{t-1}^{\text{known}} \sim \mathcal{N}(\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I})$$
$$x_{t-1}^{\text{unknown}} \sim \mathcal{N}(\mu_\theta(x_t,t), \Sigma_\theta(x_t,t))$$
$$x_{t-1} = m \odot x_{t-1}^{\text{known}} + (1-m) \odot x_{t-1}^{\text{unknown}}$$

Resampling

$x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $t = T, \dots, 1$ **do**
  for $u = 1, \dots, U$ **do**
    $\epsilon \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\epsilon = \mathbf{0}$
    $x_{t-1}^{\text{known}} = \sqrt{\bar{\alpha}_t}x_0 + (1-\bar{\alpha}_t)\epsilon$
    $z \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
    $x_{t-1}^{\text{unknown}} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t)\right) + \sigma_t z$
    $x_{t-1} = m \odot x_{t-1}^{\text{known}} + (1-m) \odot x_{t-1}^{\text{unknown}}$
    if $u < U$ and $t > 1$ **then**
      $x_t \sim \mathcal{N}(\sqrt{1-\beta_{t-1}}x_{t-1}, \beta_{t-1}\mathbf{I})$
    end if
  end for
end for
return x_0

Ablation Study

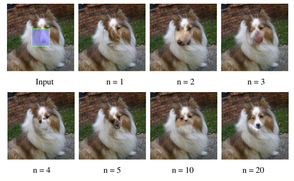
Resampling vs Slowing Down

	T	r	LPIPS	T	r	LPIPS	T	r	LPIPS	T	r	LPIPS
Slowing down	250	1	0.168	500	1	0.167	750	1	0.179	1000	1	0.161
Resampling	250	1	0.168	250	2	0.148	250	3	0.142	250	4	0.134

Jump Length

	$j=1$		$j=5$		$j=10$	
r	LPIPS	Votes [%]	LPIPS	Votes [%]	LPIPS	Votes [%]
5	0.075	42.50±7.7	0.072	46.88±7.8	0.073	53.12±7.8
10	0.088	42.50±7.7	0.073	45.62±7.8	0.068	56.25±7.8
15	0.065	46.25±7.8	0.063	53.12±5.5	0.065	53.75±7.8

Number of Resampling



Experiments

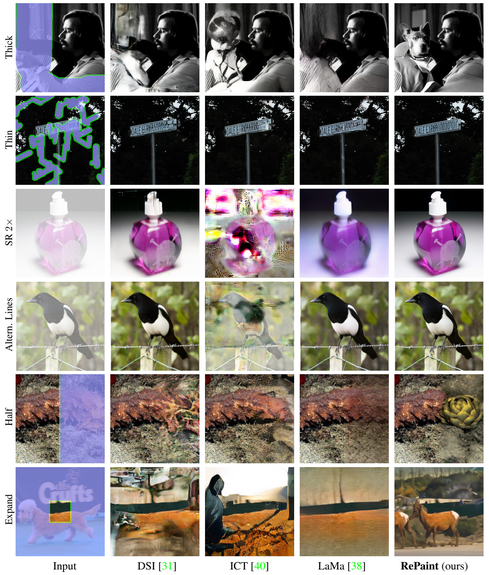
Experiments

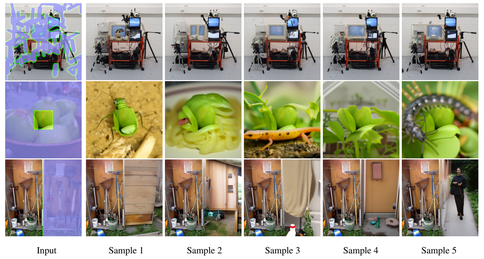
- Datasets: CelebA-HQ, ImageNet, Places2
- Masks: Thin, Thick, Generate Half, Expand, Every Second Line, Super-Resolution
- Class Conditional Inpainting
- Extensive use study

SOTA Comparison

ImageNet	Wide		Narrow		Super-Resolve 2×		Altern. Lines		Half		Expand	
Methods	LPIPS↓	Votes [%]	LPIPS↓	Votes [%]	LPIPS↓	Votes [%]	LPIPS↓	Votes [%]	LPIPS↓	Votes [%]	LPIPS↓	Votes [%]
DSI [31]	0.117	31.7 ± 2.9	0.072	28.6 ± 2.8	0.153	26.9 ± 2.8	0.069	23.6 ± 2.6	0.283	31.4 ± 2.9	0.583	9.2 ± 1.8
ICT [40]	0.107	42.9 ± 3.1	0.073	33.0 ± 2.9	0.708	1.1 ± 0.6	0.620	6.6 ± 1.5	0.255	51.5 ± 3.1	0.544	25.6 ± 2.7
LaMa [38]	0.105	42.4 ± 3.1	0.061	33.6 ± 2.9	0.272	13.0 ± 2.1	0.121	9.6 ± 1.8	0.254	41.1 ± 3.1	0.534	20.3 ± 2.5
RePaint	0.134	*Reference*	0.064	*Reference*	0.183	*Reference*	0.089	*Reference*	0.304	*Reference*	0.629	*Reference*

Visual Examples





Class Conditional Inpainting