# [22CVPR]RePaint Inpainting Using Denoising Diffusion Probabilistic Models

Luo Zhenlin

October 10, 2024

## 1 Overview

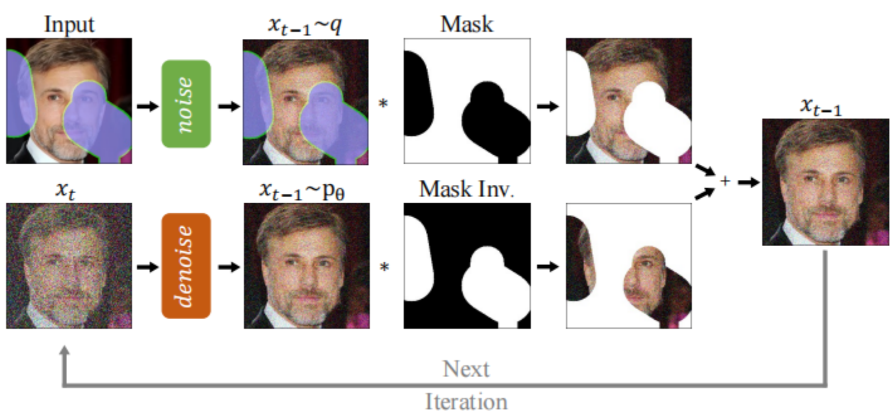


Figure 1: RePaint

In this work, we propose RePaint: A Denoising Diffusion Probabilistic Model (DDPM) based inpainting approach that is applicable to even extreme masks. We employ a pretrained unconditional DDPM as the generative prior. To condition the generation process, we only alter the reverse diffusion iterations by sampling the unmasked regions using the given image information. Since this technique does not modify or condition the original DDPM network itself, the model produces high quality and diverse output images for any inpainting form.

- The forward process is not changed, and this work only changes the reverse process.
- In each step, we sample the known region from the input and the inpainted part from the DDPM output. And combine the two parts to generate new sample.

# 2 Review DDPM

In DDPM. the **forward process** is defined beforehand.

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t\mathbf{I}) \tag{1}$$
$$x_t = \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon \tag{2}$$

But in fact, we construct $x_t$ from $x_0$ as a single step, since

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{3}$$
$$x_t = \sqrt{\bar{\alpha}_t}x_0 + \sqrt{1-\bar{\alpha}_t}\epsilon \tag{4}$$

where $\bar{\alpha}_t = \prod_{s=1}^{t}(1-\beta_s)$.

The DDPM is trained to reverse the process in (1). The reverse process is modeled by a neural network that predicts the parameters $\mu\theta(x_t, t)$ and $\Sigma_\theta(x_t, t)$ of a Gaussian distribution

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{5}$$

Based on the forward process, we can deduce that reverse process mean

$$\tilde{\mu}_t(x_t, x_0) = \frac{1}{\sqrt{\alpha_t}}(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon) \tag{6}$$

Hence we train the model $\epsilon_\theta$ to estimate $\tilde{\mu}_t(x_t, x_0)$, since only $\epsilon_\theta$ is unknown.

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)\right) \tag{7}$$

# 3 Method

## 3.1 Conditioning on the known Region

In our task, we predict missing pixels of an image using a *mask region* as a condition. Denote the ground truth image as $x$, the unknown part as $m \odot x$ and the known part as $(1-m) \odot x$.

- We can sample the intermediate image $x_t$ at any point in time using (4)
- Using (5) for the unknown region and (4) for the known regions
- For one reverse step, we have

$$x_{t-1}^{\text{known}} \sim \mathcal{N}(\sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha}_t)\mathbf{I}) \tag{8}$$
$$x_{t-1}^{\text{unknown}} \sim \mathcal{N}(\mu_\theta(x_t, t), \Sigma_\theta(x_t, t)) \tag{9}$$
$$x_{t-1} = m \odot x_{t-1}^{\text{known}} + (1-m) \odot x_{t-1}^{\text{unknown}} \tag{10}$$

Where $x_{t-1}^{\text{known}}$ is sampled from the known pixels and $x_{t-1}^{\text{umknown}}$ is sampled from the reverse model. These are combined to the new sample $x_{t-1}$ using the mask.

## 3.2 Resampling

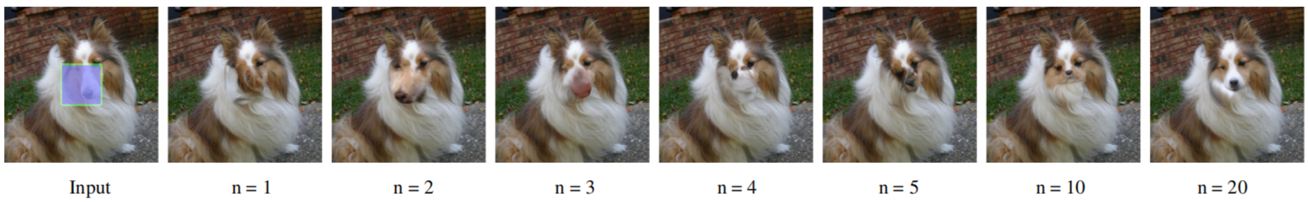


Figure 2: The effect of applying n sampling steps.

The DDPM is leveraging on the context of the known region, yet it is not harmonizing it well with the rest of the image.

The reason is that

- Using (8), it doesn't consider the generated parts of the image.
- Due to the variance schedule of $\beta_t$, the maximum change to an image declines.

As a result, the model needs more time to harmonize the conditional information $x_{t-1}^{\text{known}}$ with generated information $x_{t-1}^{\text{unknown}}$.

### 3.2.1 Algorithm

**Algorithm 1** Inpainting using our RePaint approach.

$x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
**for** $t = T, \ldots, 1$ **do**
  **for** $u = 1, \ldots, U$ **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 $\mathbf{z} = \mathbf{0}$
    $x_{t-1}^{\text{unknown}} = \frac{1}{\sqrt{\alpha_t}} \left( x_t - \frac{\beta_t}{\sqrt{1-\bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(x_t, t) \right) + \sigma_t z$
    $x_{t-1} = m \odot x_{t-1}^{\text{known}} + (1 - m) \odot x_{t-1}^{\text{unknown}}$
    **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** $x_0$

It implies that in each reverse step, we use this DDPM property to harmonize the input of the model, that is, we diffuse the output $x_{t-1}$ back to $x_t$ by sampling from (1) as $\mathcal{N}(x_t; \sqrt{1 - \beta_t} x_{t-1}, \beta_t \mathbf{I})$.

# 4 Experiments

## 4.1 Comparison with State-of-the-Art

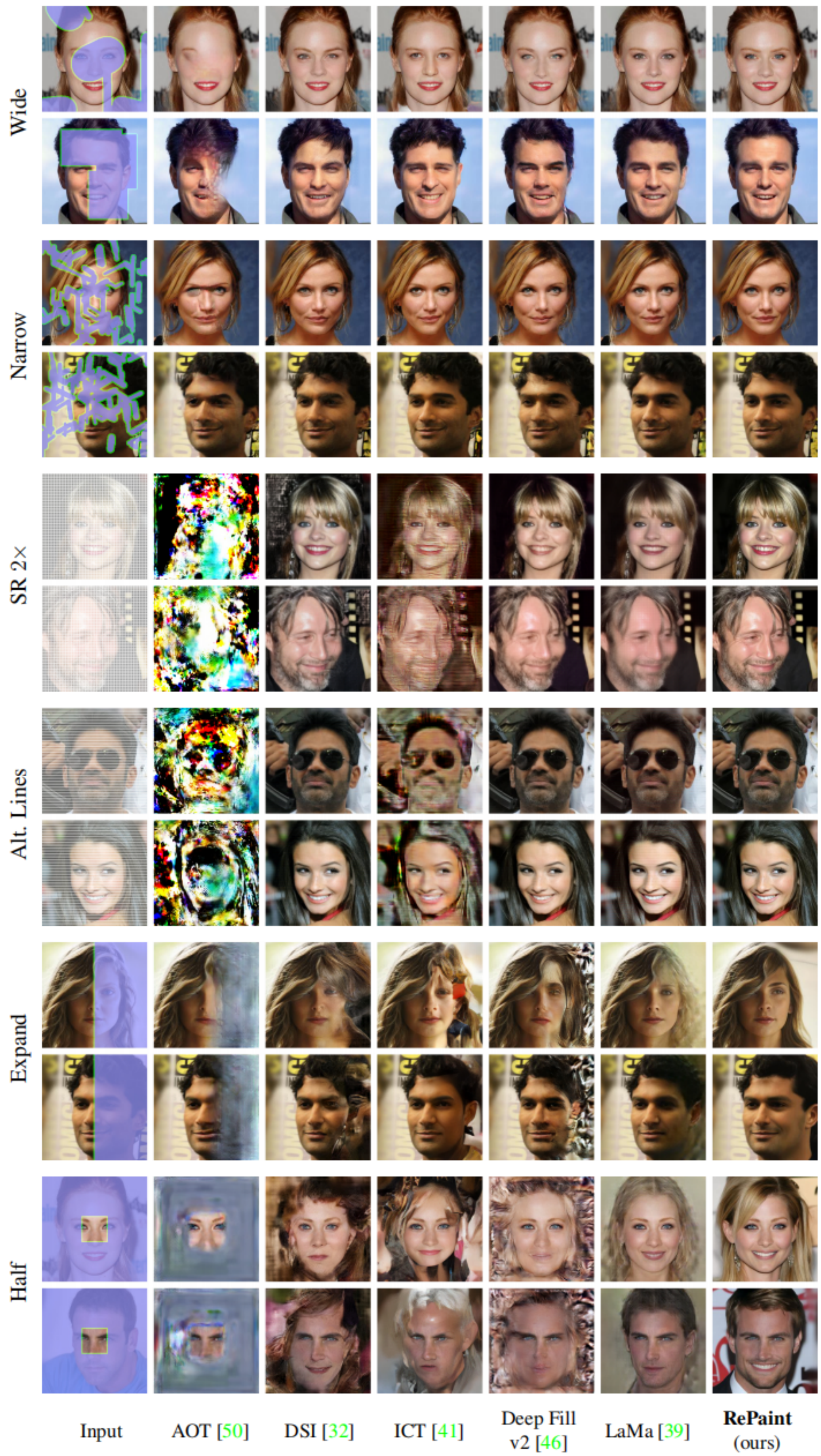


Figure 3: Comparison with State-of-the-Art

| **CelebA-HQ** | Wide | | Narrow | | Super-Resolve 2× | | Altern. Lines | | Half | | Expand | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Methods | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] |
| AOT [50] | 0.104 | $11.6 \pm 2.0$ | 0.047 | $12.8 \pm 2.1$ | 0.714 | $1.1 \pm 0.6$ | 0.667 | $2.4 \pm 1.0$ | 0.287 | $9.0 \pm 1.8$ | 0.604 | $8.3 \pm 1.7$ |
| DSI [32] | 0.067 | $16.0 \pm 2.3$ | 0.038 | $22.3 \pm 2.6$ | 0.128 | $5.5 \pm 1.4$ | 0.049 | $5.1 \pm 1.4$ | 0.211 | $4.5 \pm 1.3$ | 0.487 | $4.7 \pm 1.3$ |
| ICT [41] | 0.063 | $27.6 \pm 2.8$ | 0.036 | $30.9 \pm 2.9$ | 0.483 | $4.2 \pm 1.2$ | 0.353 | $0.7 \pm 0.5$ | 0.166 | $12.7 \pm 2.1$ | 0.432 | $8.8 \pm 1.8$ |
| DeepFillv2 [46] | 0.066 | $23.9 \pm 2.6$ | 0.049 | $21.0 \pm 2.5$ | 0.119 | $9.8 \pm 1.8$ | 0.049 | $10.6 \pm 1.9$ | 0.209 | $4.1 \pm 1.2$ | 0.467 | $13.1 \pm 2.1$ |
| LaMa [39] | 0.045 | $41.8 \pm 3.1$ | 0.028 | $33.8 \pm 3.0$ | 0.177 | $5.5 \pm 1.4$ | 0.083 | $20.6 \pm 2.5$ | 0.138 | $35.6 \pm 3.0$ | 0.342 | $24.7 \pm 2.7$ |
| **RePaint** | 0.059 | *Reference* | 0.028 | *Reference* | 0.029 | *Reference* | 0.009 | *Reference* | 0.165 | *Reference* | 0.435 | *Reference* |

| **ImageNet** | Wide | | Narrow | | Super-Resolve 2× | | Altern. Lines | | Half | | Expand | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Methods | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] | LPIPS↓ | Votes [%] |
| DSI [32] | 0.117 | $31.7 \pm 2.9$ | 0.072 | $28.6 \pm 2.8$ | 0.153 | $26.9 \pm 2.8$ | 0.069 | $23.6 \pm 2.6$ | 0.283 | $31.4 \pm 2.9$ | 0.583 | $9.2 \pm 1.8$ |
| ICT [41] | 0.107 | $42.9 \pm 3.1$ | 0.073 | $33.0 \pm 2.9$ | 0.708 | $1.1 \pm 0.6$ | 0.620 | $6.6 \pm 1.5$ | 0.255 | $51.5 \pm 3.1$ | 0.544 | $25.6 \pm 2.7$ |
| LaMa [39] | 0.105 | $42.4 \pm 3.1$ | 0.061 | $33.6 \pm 2.9$ | 0.272 | $13.0 \pm 2.1$ | 0.121 | $9.6 \pm 1.8$ | 0.254 | $41.1 \pm 3.1$ | 0.534 | $20.3 \pm 2.5$ |
| **RePaint** | 0.134 | *Reference* | 0.064 | *Reference* | 0.183 | *Reference* | 0.089 | *Reference* | 0.304 | *Reference* | 0.629 | *Reference* |

Table 1. **CelebA-HQ (*top*) and ImageNet (*bottom*) Quantitative Results.** Comparison against the state-of-the-art methods. We compute the LPIPS (lower is better) and *Votes* for six different mask settings. *Votes* refers to the ratio of votes with respect to ours.

## 4.2 Ablation Study

### 4.2.1 Comparison Slowing down and Resampling

| | 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 |

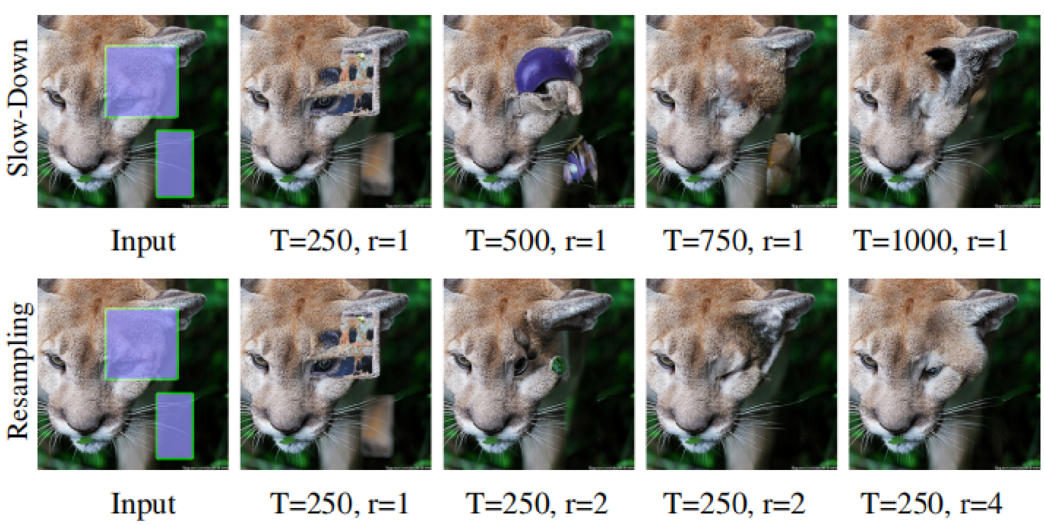


Figure 4: Comparison Slowing down and Resampling

#### 4.2.2 Comparison to alternative sampling strategy

| Dataset | Method | Wide | Narrow | Super-Res. | Alt. Lin. | Half | Expand |
|---|---|---|---|---|---|---|---|
| ImageNet | SDEdit [24] | 0.1532 | 0.0952 | 0.3902 | 0.1852 | 0.3272 | **0.6281** |
| | RePaint (Ours) | **0.1341** | **0.0641** | **0.1831** | **0.0891** | **0.3041** | 0.6292 |
| Places2 | SDEdit [24] | 0.1302 | 0.0622 | 0.2712 | 0.1302 | 0.3042 | 0.6202 |
| | RePaint (Ours) | **0.1051** | **0.0441** | **0.0991** | **0.0511** | **0.2861** | **0.6151** |
| CelebA-HQ | SDEdit [24] | 0.0762 | 0.0462 | 0.1132 | 0.0302 | 0.1892 | 0.4492 |
| | RePaint (Ours) | **0.0591** | **0.0281** | **0.0291** | **0.0091** | **0.1651** | **0.4351** |

Table 4. Comparison with the resampling schedule proposed in [24] in terms of LPIPS. The resampling method proposed in our RePaint (Sec. 4.2) achieves substantially better results, in particular for the Super-Resolution masks.