Diffusion models for Image Restoration

Daniela Ivanova

Computer Vision and Autonomous Systems (CVAS)

What is image restoration?

Colourisation

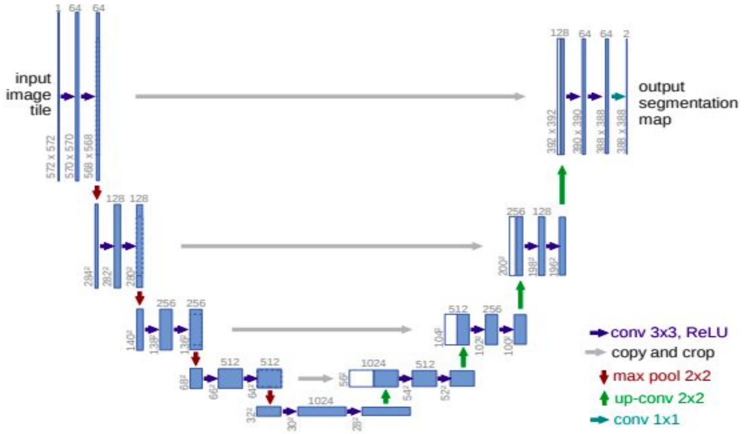
Super-resolution

Inpainting (regular shape)

Inpainting (arbitrary artefacts)

Diffusion architecture - recap

- Since the *input, output and latents have the same dimensionality*, for each step the denoising model is a U-Net with some modifications
  - Group normalisation
  - Global self-attention
  - Sinusoidal positional **time embeddings** concatenated to the input of each block
- Instead of predicting the denoised image, the network predicts the noise that was added to it



Credit: Ronneberger et al, "Convolutional Networks for Biomedical Image Segmentation"

Diffusion process - recap

Distribution of noised images

$$q(x_1, \dots, x_T | x_0) := \prod_{t=1}^{T} q(x_t|x_{t-1}) \tag{1}$$

scaling

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

diagonal covariance matrix

- To produce each latent, we can add noise iteratively (slow)

Diffusion process - recap

noised latents directly conditioned on the input x_0. With $\alpha_t := 1 - \beta_t$ and $\bar{\alpha}_t := \prod_{s=0}^{t} \alpha_s$, we can write the marginal

$$q(x_t|x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t} x_0, (1 - \bar{\alpha}_t)\mathbf{I}) \tag{8}$$

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon \tag{9}$$

where $\epsilon \sim \mathcal{N}(0, \mathbf{I})$. Here, $1 - \bar{\alpha}_t$ tells us the variance of the noise for an arbitrary timestep, and we could equivalently use this to define the noise schedule instead of β_t.

- …or we can derive the variance scale for an arbitrary step - accumulate the noise from the first step to the step we need

- We can condition on class labels - embed class label v_i along with time embedding e_t - (Nichol & Dhariwal, 2021)

Credit: Nichol & Dhariwal, "Improved denoising diffusion probabilistic models."

University of Glasgow

Diffusion conditioning - recap

- Classifier guidance (Dhariwal & Nichol, 2021)
  - y comes from the downsampling half of the UNet, which is used as a classifier

Figure 6: Samples from BigGAN-deep with truncation 1.0 (FID 6.95, left) vs samples from our diffusion model with guidance (FID 4.59, middle) and samples from the training set (right).

Credit: Dhariwal & Nichol, "Diffusion Models Beat GANs on Image Synthesis"

Diffusion conditioning - recap

- Classifier guidance (Dhariwal & Nichol, 2021)

$$p_{\theta,\phi}(x_t|x_{t+1},y) = Zp_\theta(x_t|x_{t+1})p_\phi(y|x_t)$$

$$\begin{aligned}\nabla_{x_t}\log(p_\theta(x_t)p_\phi(y|x_t)) &= \nabla_{x_t}\log p_\theta(x_t) + \nabla_{x_t}\log p_\phi(y|x_t)\\ &= -\frac{1}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t) + \nabla_{x_t}\log p_\phi(y|x_t)\end{aligned}$$

$$\hat{\epsilon}(x_t) := \epsilon_\theta(x_t) - \sqrt{1-\bar{\alpha}_t}\,\nabla_{x_t}\log p_\phi(y|x_t)$$

Diffusion conditioning - recap

"a hedgehog using a calculator" | "a corgi wearing a red bowtie and a purple party hat" | "robots meditating in a vipassana retreat" | "a fall landscape with a small cottage next to a lake"

"a surrealist dream-like oil painting by salvador dalí of a cat playing checkers" | "a professional photo of a sunset behind the grand canyon" | "a high-quality oil painting of a psychedelic hamster dragon" | "an illustration of albert einstein wearing a superhero costume"

- We can condition on text descriptions
  - Each attention layer is attending to each token for the text embedding
  - Doesn't work very well still

Credit: Dhariwal & Nichol, "GLIDE: Towards Photorealistic Image Generation and Editing with Text-Guided Diffusion Models"

Diffusion conditioning - recap

- CLIP guided diffusion:
  - At inference time, use CLIP guidance
  - CLIP outputs a similarity score between image and text for each pixel
  - Use that gradient at each time step to push the image in the direction which would give it higher score/smaller CLIP loss

(c) GLIDE (CLIP guidance, scale 2.0)

Diffusion conditioning - recap

- Classifier-free guidance:
  - Train with and without text embeddings
  - Predict an image without the text prompt and with the text prompt
  - Find the difference between the two
  - Use that gradient to go in the direction of the image with text using a scaling factor for the vector

(d) GLIDE (Classifier-free guidance, scale 3.0)

Diffusion models for image restoration

End-to-end training with conditioning

- SR3
- Palette

Using pre-trained models, conditioning only during inference

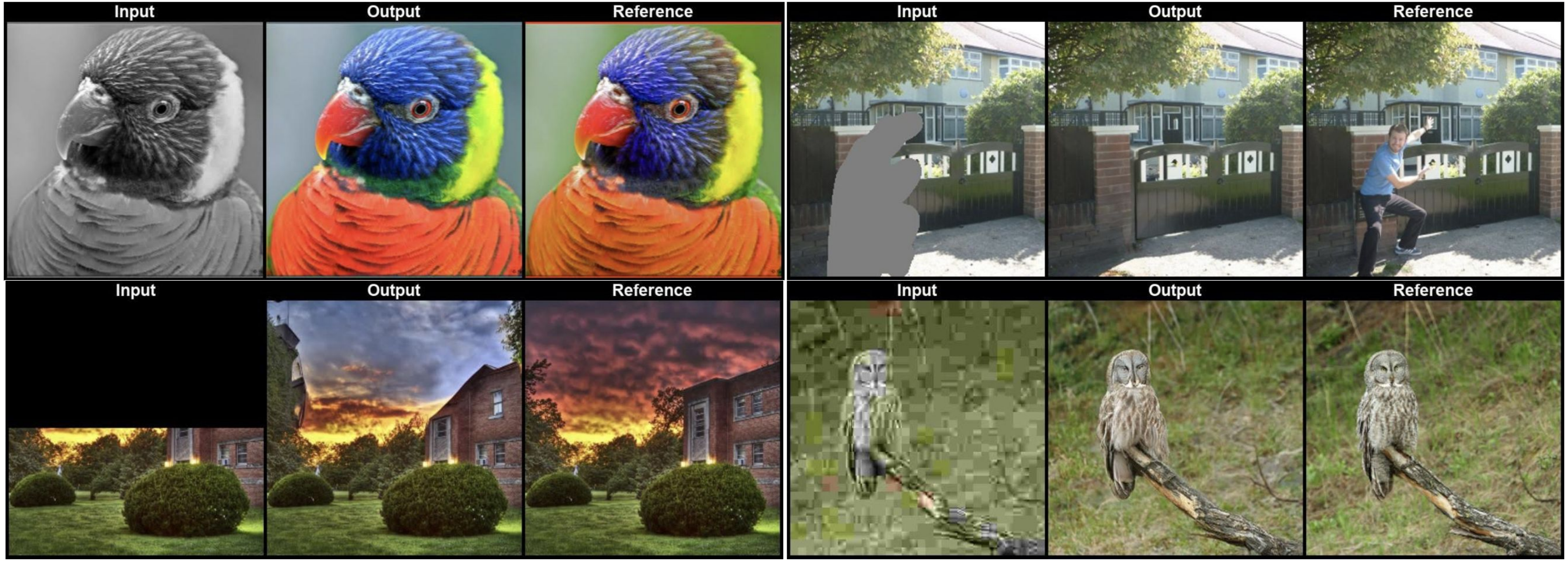
- DDRM
- RePaint
- Stable Diffusion
- DiffEdit (bonus!)

SR3 (Saharia et al.)

- We can condition on another image
  - low res image for superresolution
  - grayscale image for colourisation
  - images with missing patches for inpainting
- Concatenate noise vector with the conditioning image
- Slow, diffusing the entire image

Credit: Saharia et al, “Image Super-Resolution via Iterative Refinement”

Palette: Image-to-Image Diffusion Models (Saharia et al.)



Credit: Saharia et al, "Palette: Image-to-Image Diffusion Models"

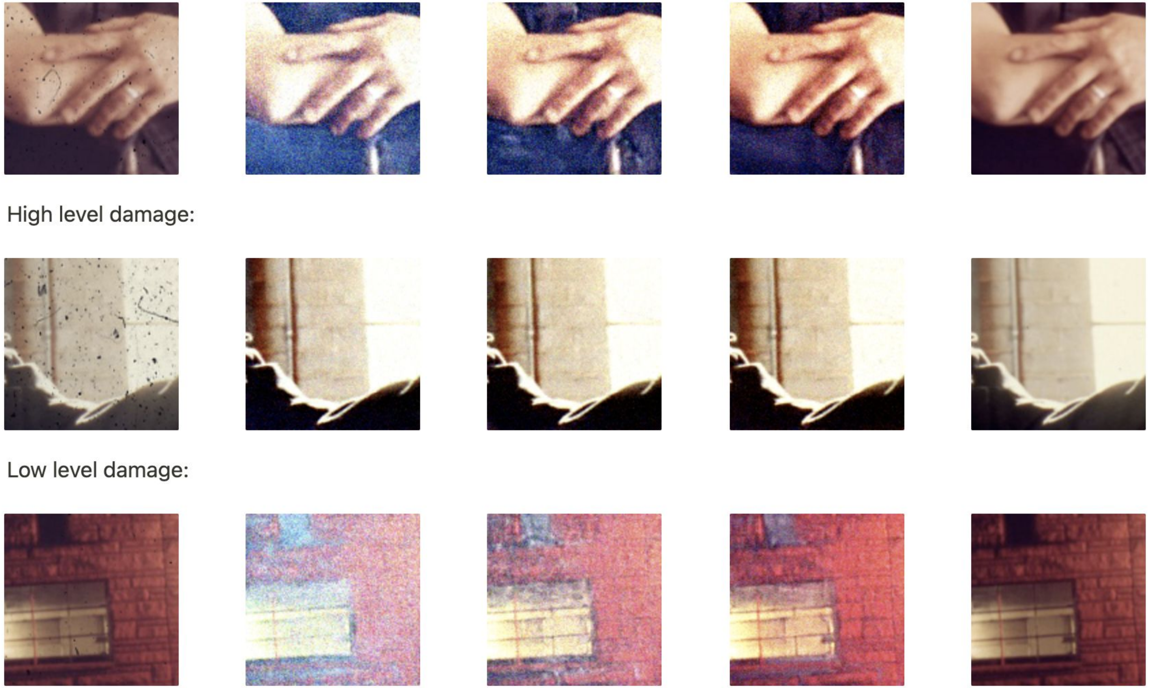
Diffusion model for film artifact removal

input | reconstruction 3 weeks ago | reconstruction 2 weeks ago | reconstruction now | GT

Medium level damage:

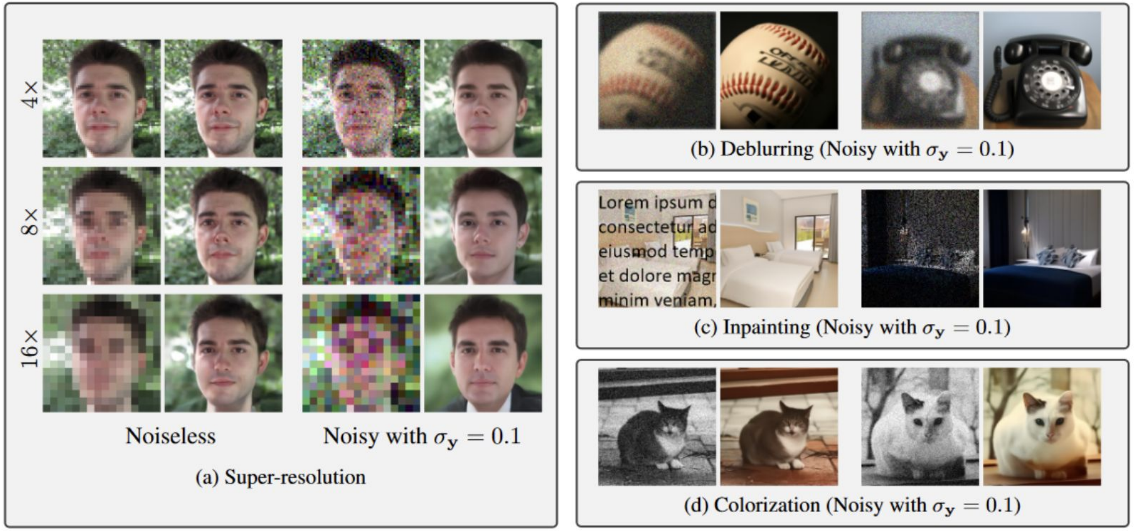
High level damage:

Low level damage:



- 97.8 M params
- bs=8
- 1M iterations in SR3 paper
- ***very*** slow if you don't have a TPU
- task-specific - need lots of data

DDRM (Kawar et al., 2022)

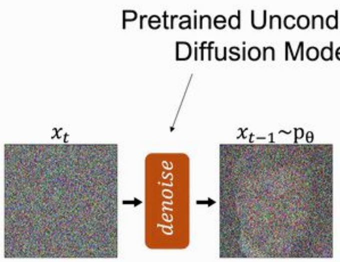


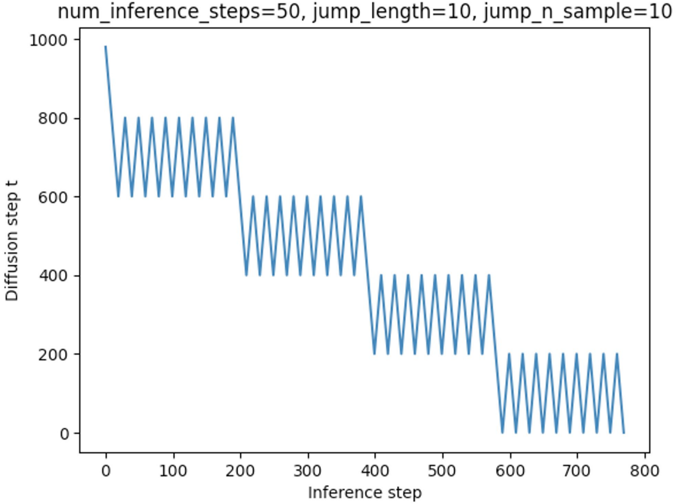
- use pretrained unconditional DDPM
- decompose degradation operator ***H*** using SVD
- perform diffusion in spectral space

Credit: Kawar et al, "Denoising Diffusion Restoration Models"

$$\mathbf{y} = \boldsymbol{H}\mathbf{x} + \mathbf{z}, \qquad q(\mathbf{x}_{1:T}|\mathbf{x}_0, \mathbf{y}) = q^{(T)}(\mathbf{x}_T|\mathbf{x}_0, \mathbf{y}) \prod_{t=0}^{T-1} q^{(t)}(\mathbf{x}_t|\mathbf{x}_{t+1}, \mathbf{x}_0, \mathbf{y}),$$

RePaint (Lugmayr et al., 2022)



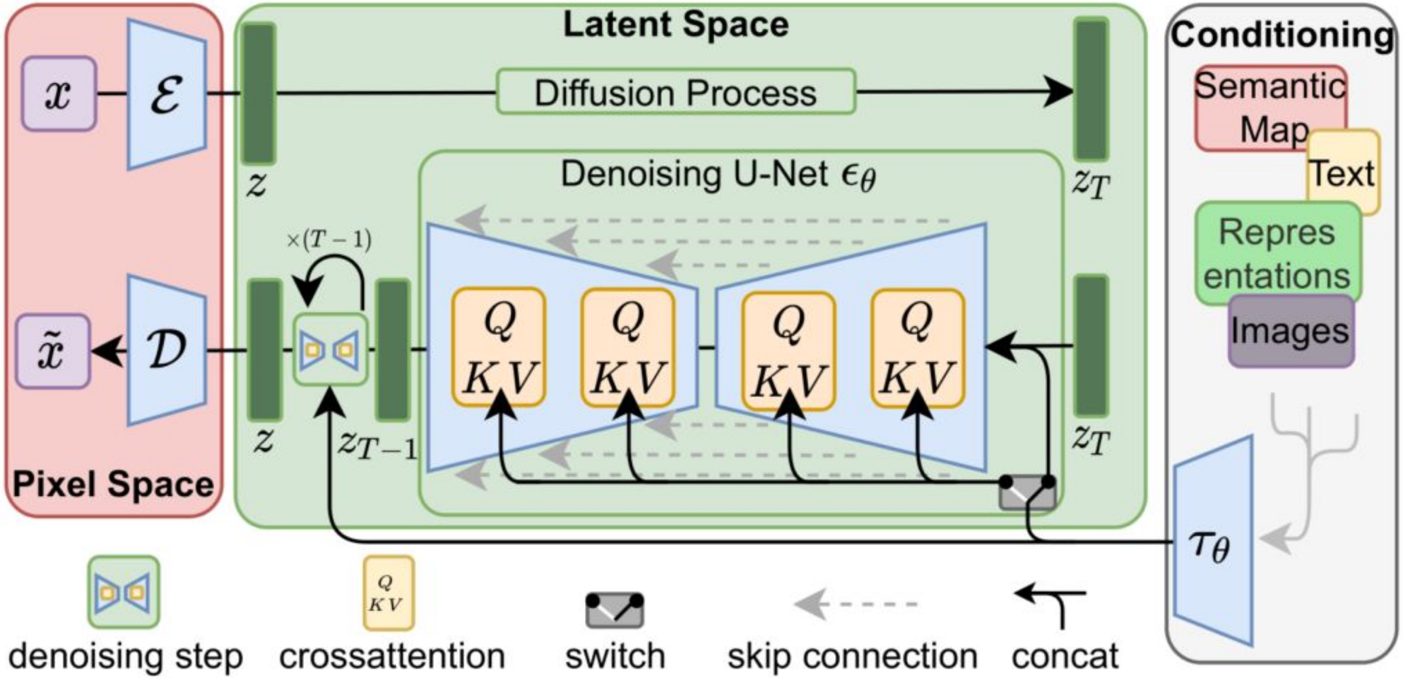


Input n = 1 n = 2 n = 3 n = 4 n = 5 n = 10 n = 20

Figure 3. **The effect of applying n sampling steps.** The first example with $n = 1$ is the DDPM baseline, the second with $n = 2$ is with one resample step. More resampling steps lead to more harmonized images. The benefit saturates at about $n = 10$ resamplings.

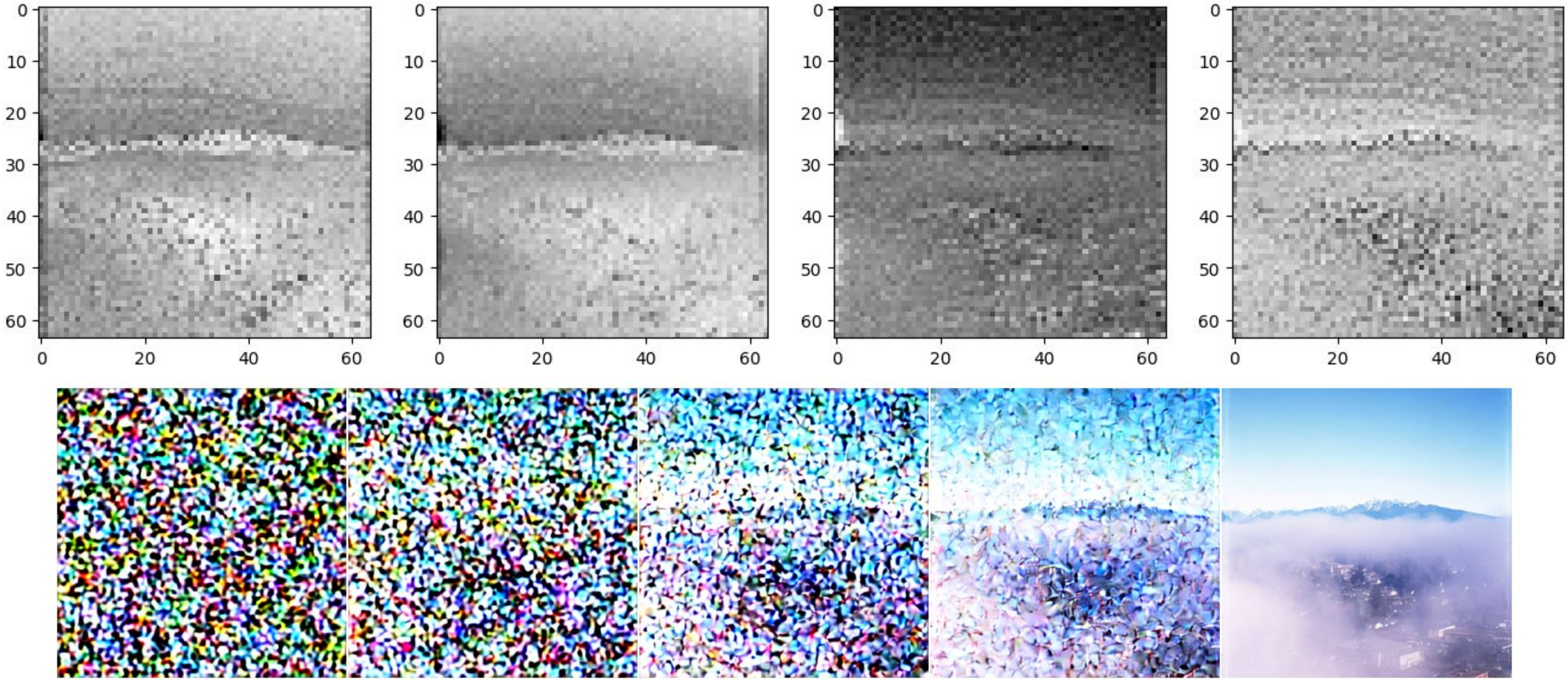
Credit: Lugmayr et al, "RePaint: Inpainting using Denoising Diffusion Probabilistic Models"

Latent diffusion (Romach et al.)



Credit: Romach et al., “High-Resolution Image Synthesis with Latent Diffusion Models”

Latent diffusion (Romach et al.)



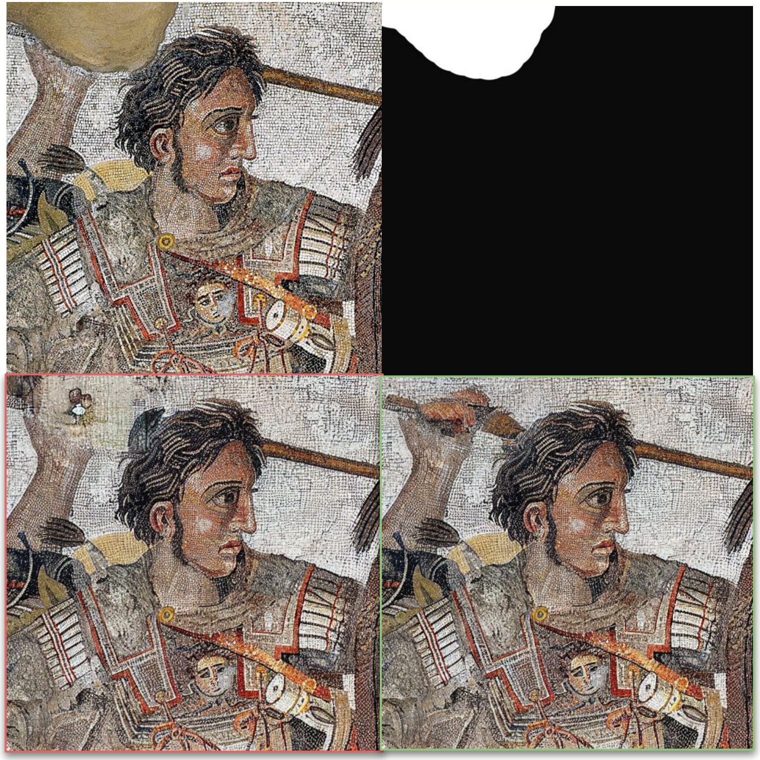
Stable Diffusion Inpainting

“cat sitting on a bench”

Stable Diffusion Inpainting

Unconditional (legacy)

- use pre-trained SD
- make prediction from noise
- mask out the latents
- make next prediction
- etc

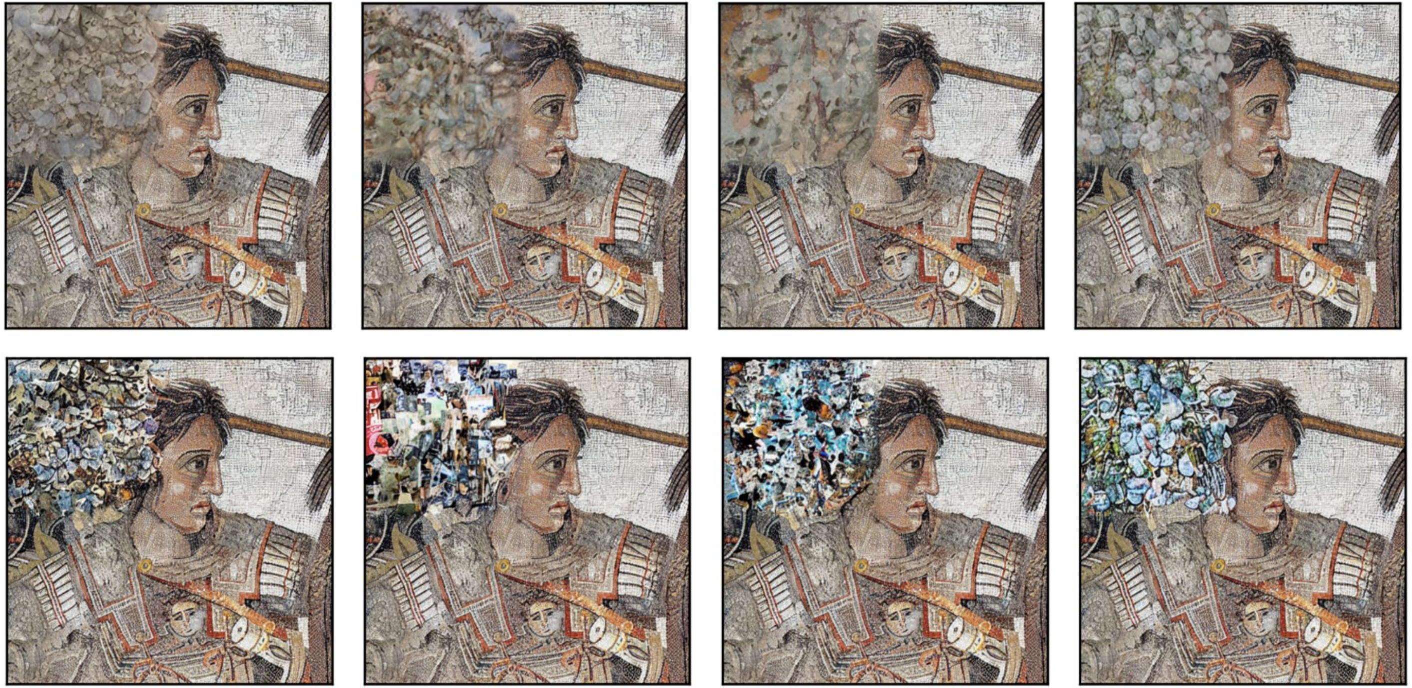


Conditional

- fine-tune pre-trained SD model on inpainting
- pass the mask, masked latents, original latents as a 9-channel input to the U-Net

RePaint (Lugmayr et al., 2022)

What if we tried RePaint it in latent space?

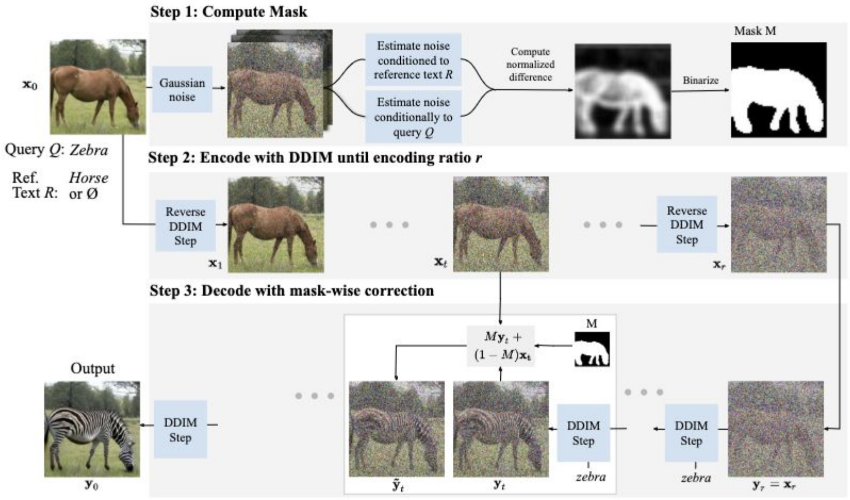


RePaint (Lugmayr et al., 2022)

What if we tried it in latent space?

- Only works with the sampler used to train SD

DiffEdit (Couairon et al., 2022)



Credit: Couairon et al.,"DiffEdit: Diffusion-based semantic image editing with mask guidance“

- denoise once using reference text
- denoise again using query text
- the difference in noise estimates => locations that are predicted to change the most between conditioning on the original and new texts

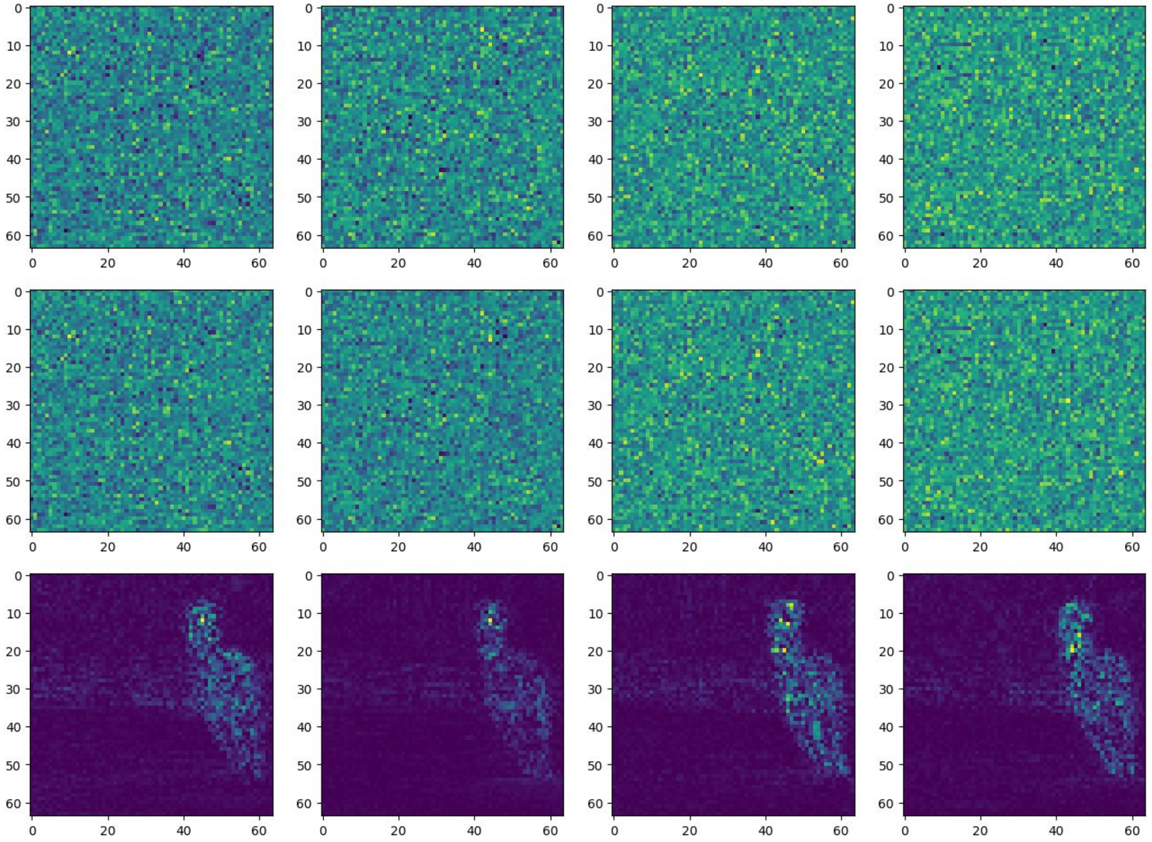
DiffEdit (Couairon et al., 2022)

reference: “horse”
query: “zebra”

- noise-denoise 10 times with each prompt
- accumulate predicted noises
- find difference

Check out “[DiffEdit paper implementation” by Kevin Bird]

DiffEdit (Couairon et al., 2022)

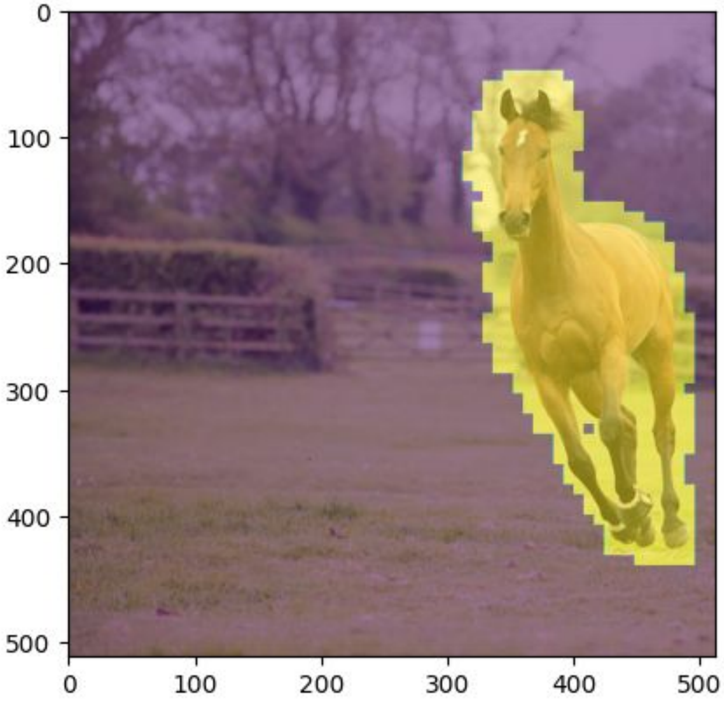


“horse” noise

“zebra” noise

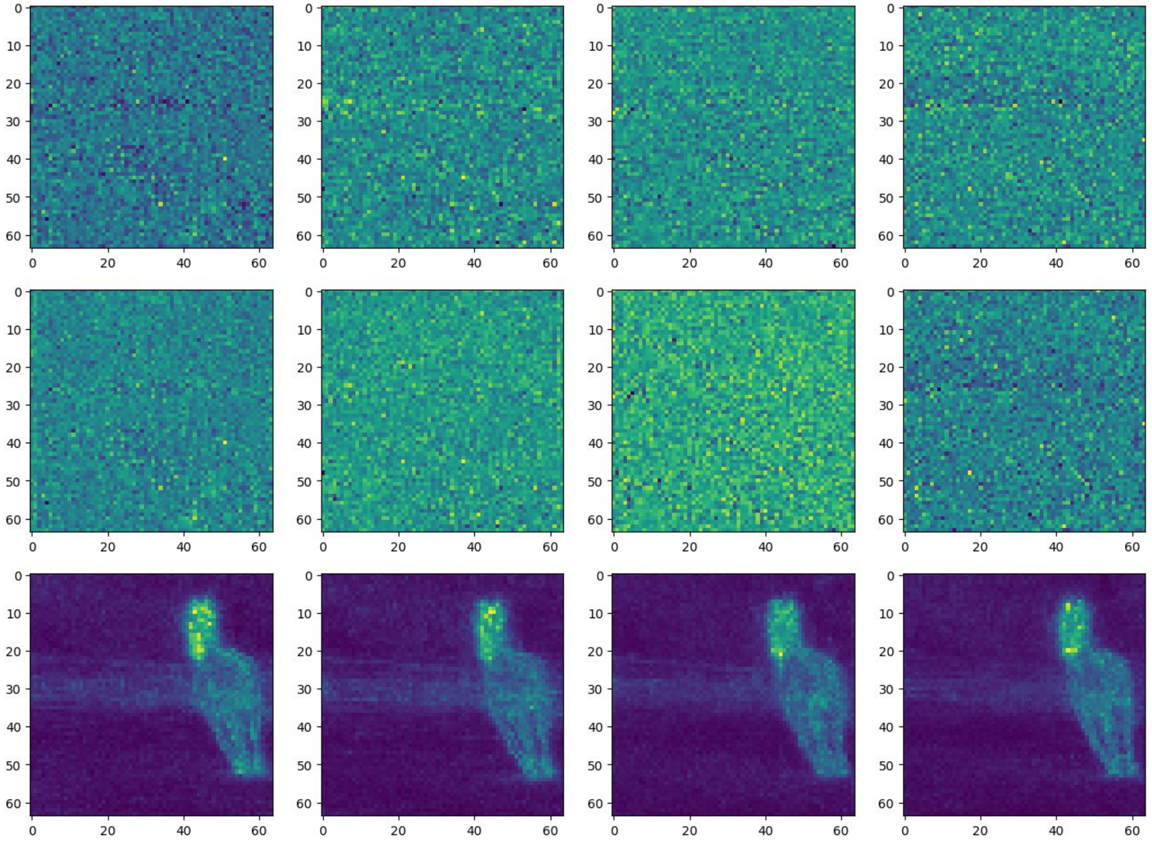
difference

DiffEdit (Couairon et al., 2022)



horse - zebra

DiffEdit (Couairon et al., 2022) - new idea

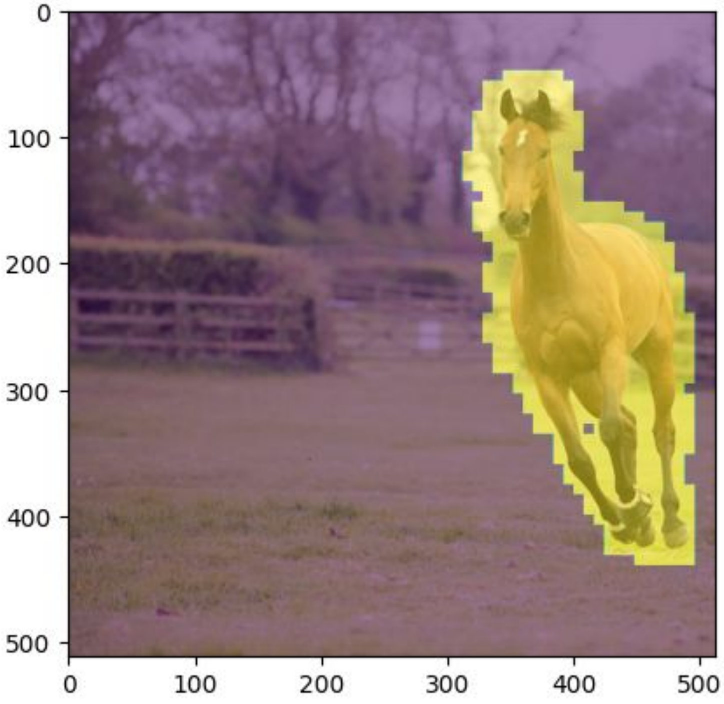


"horse" noise

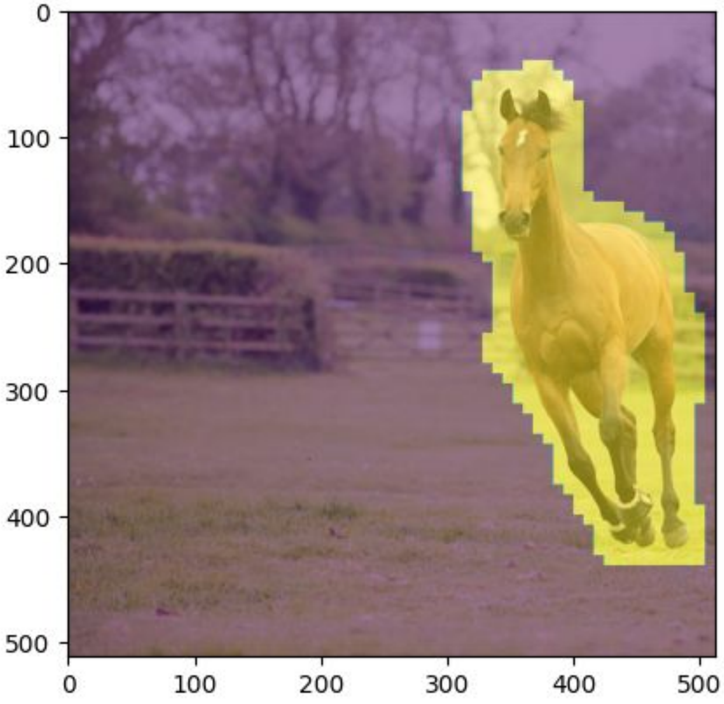
"- horse" noise

difference

DiffEdit (Couairon et al., 2022)



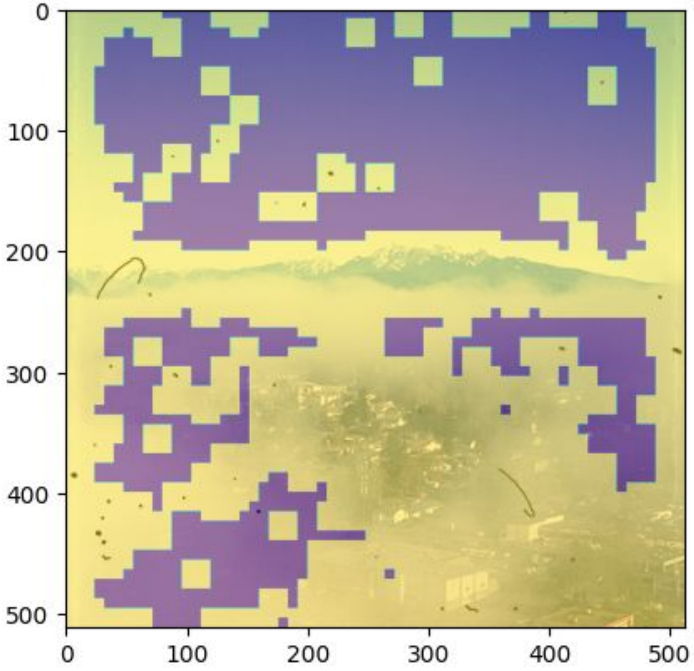
horse - zebra



horse - (-horse)

DiffEdit (Couairon et al., 2022)

damaged film photo



damaged film photo - (-damaged film photo)

Thank you!

References

Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*. Springer International Publishing, 2015.

Ho, Jonathan, Ajay Jain, and Pieter Abbeel. "Denoising diffusion probabilistic models." *Advances in Neural Information Processing Systems* 33 (2020): 6840-6851.

Song, Jiaming, Chenlin Meng, and Stefano Ermon. "Denoising diffusion implicit models." *arXiv preprint arXiv:2010.02502* (2020).

Nichol, Alexander Quinn, and Prafulla Dhariwal. "Improved denoising diffusion probabilistic models." *International Conference on Machine Learning*. PMLR, 2021.

Dhariwal, Prafulla, and Alexander Nichol. "Diffusion models beat gans on image synthesis." *Advances in Neural Information Processing Systems* 34 (2021): 8780-8794.

Nichol, Alex, et al. "Glide: Towards photorealistic image generation and editing with text-guided diffusion models." *arXiv preprint arXiv:2112.10741* (2021).

Saharia, Chitwan, et al. "Image super-resolution via iterative refinement." *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2022).

Saharia, Chitwan, et al. "Palette: Image-to-image diffusion models." *ACM SIGGRAPH 2022 Conference Proceedings*. 2022.

Kawar, Bahjat, et al. "Denoising diffusion restoration models." *arXiv preprint arXiv:2201.11793* (2022).

Lugmayr, Andreas, et al. "Repaint: Inpainting using denoising diffusion probabilistic models." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

Rombach, Robin, et al. "High-resolution image synthesis with latent diffusion models." *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2022.

Couairon, Guillaume, et al. "Diffedit: Diffusion-based semantic image editing with mask guidance." *arXiv preprint arXiv:2210.11427* (2022).