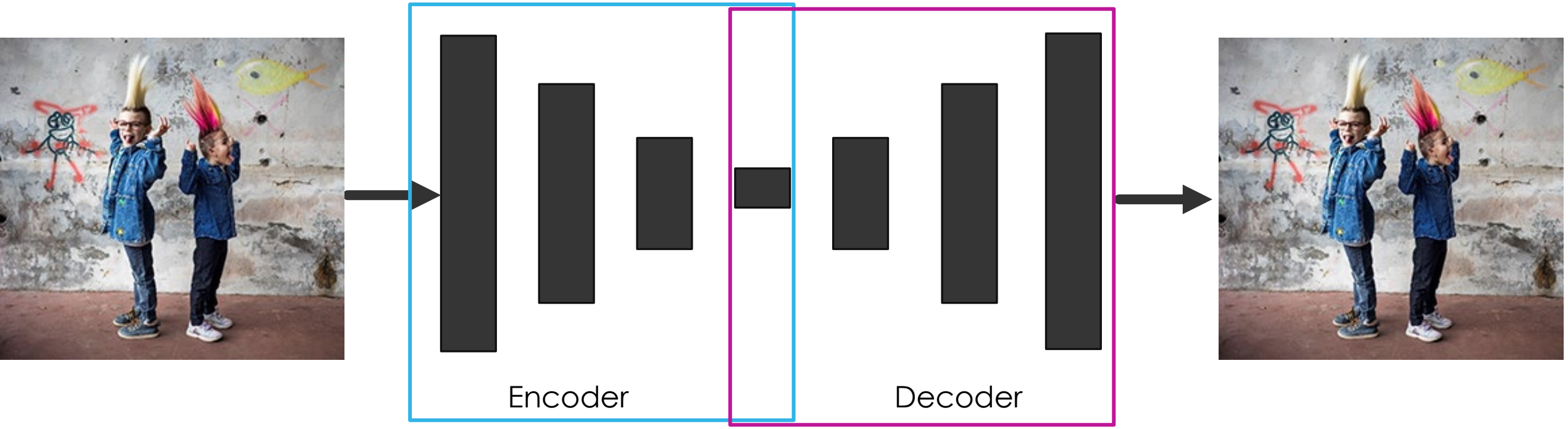
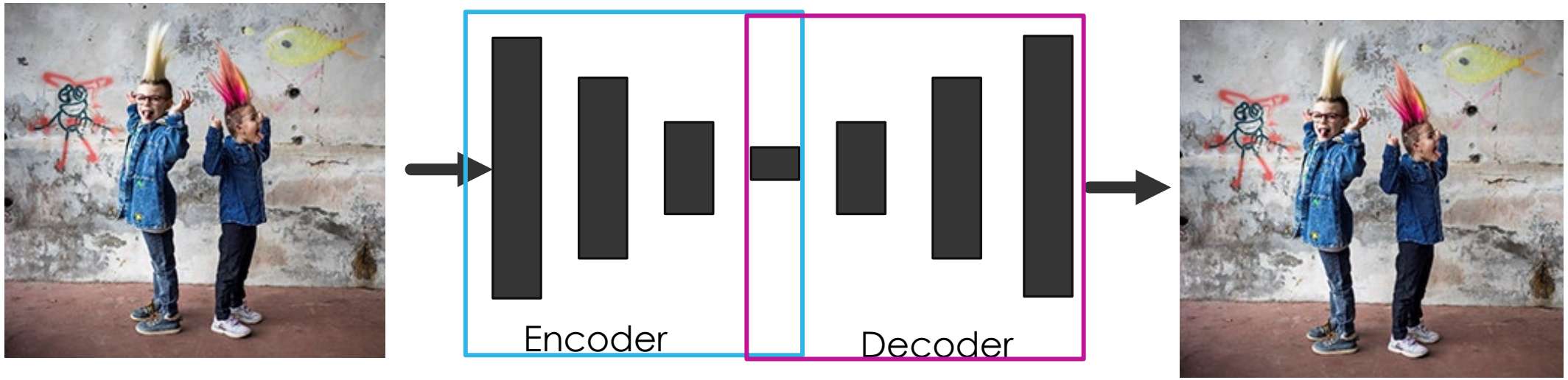
Diffusion models

Auto-encoders



Bank of 3x3 convolutional filters + nonlinearity + downsampling / upsampling

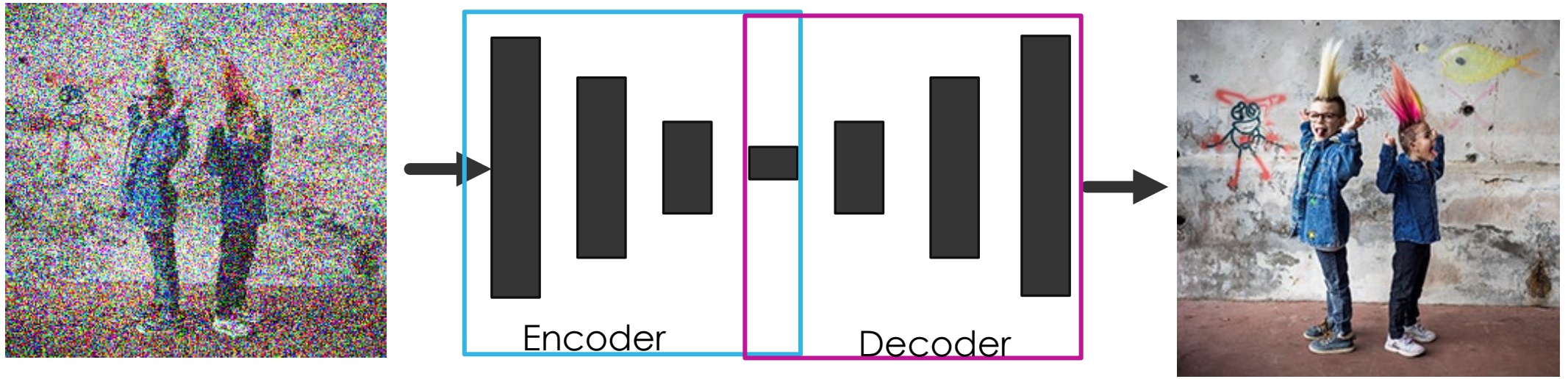
Auto-encoders



Training:

- For a given image, find convolution weights that minimizes differences between input and output images
- Assumes the use of many (many) input images : consider the sum of individual losses
- Find derivatives with automatic differentiation
- Stochastic gradient descent (for example), using small batches of images, i.e., a small random subset of input database

Restoration via Auto-encoders

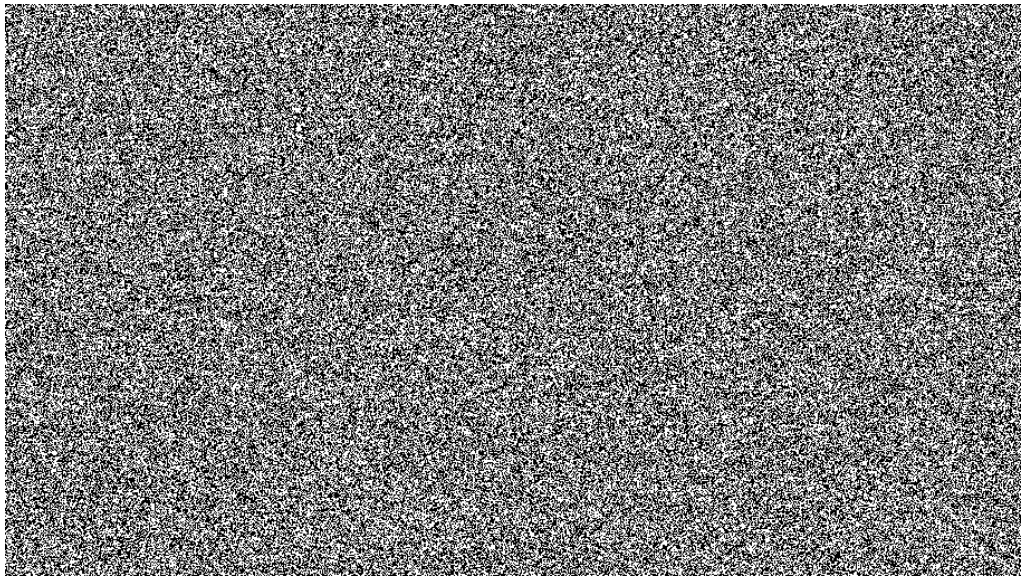


- Given trained network, encoder allows to compress input images into a small feature vector (“code”)
- Decoder recovers the image based on feature vector
- Useful for restoration ; Decoder could be used (with modest success) for generation

Diffusion models for image generation

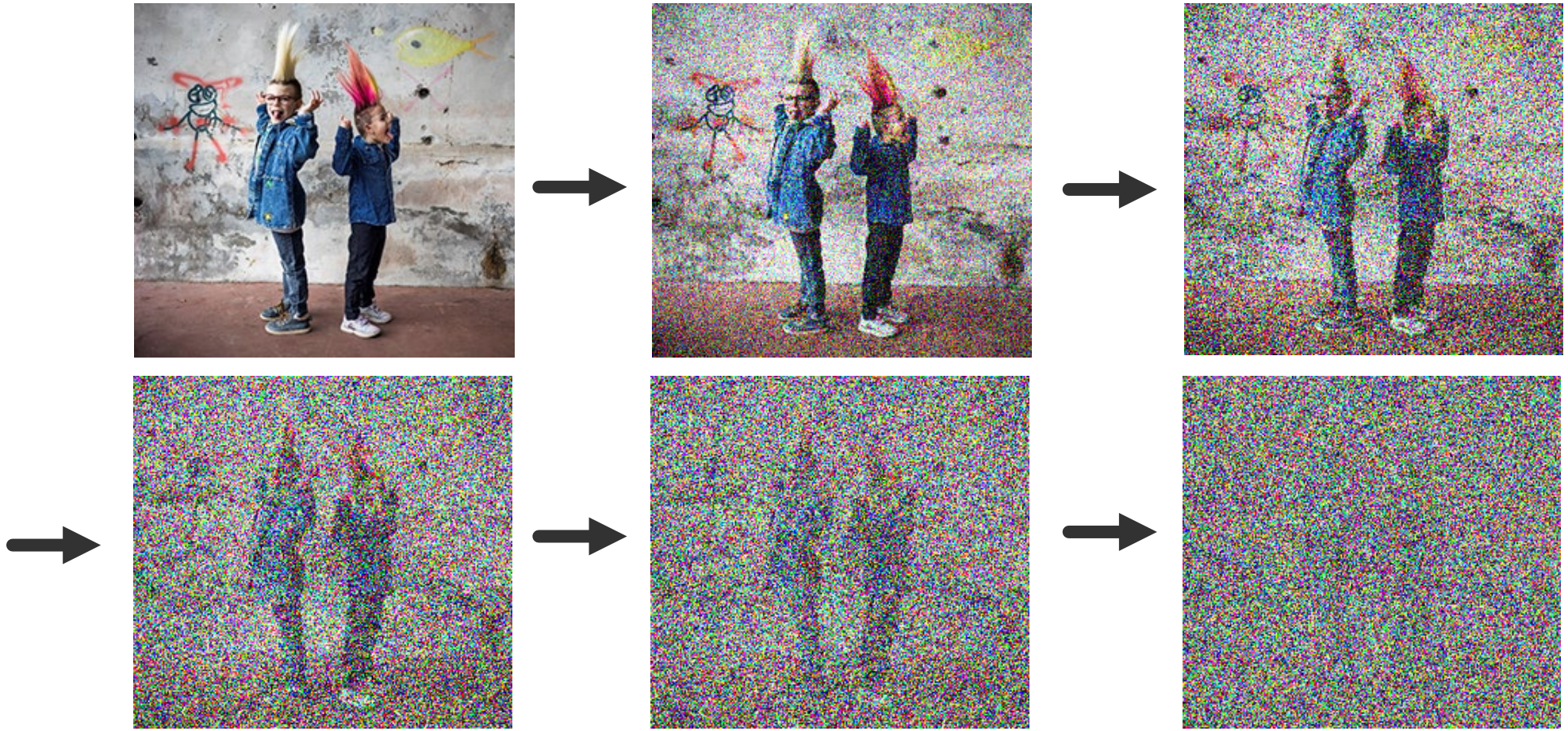
Add noise at training time

Denoise at synthesis time



Diffusion models : training

- Idea: at training time, progressively add noise



Diffusion models: training

Image 1 trajectory

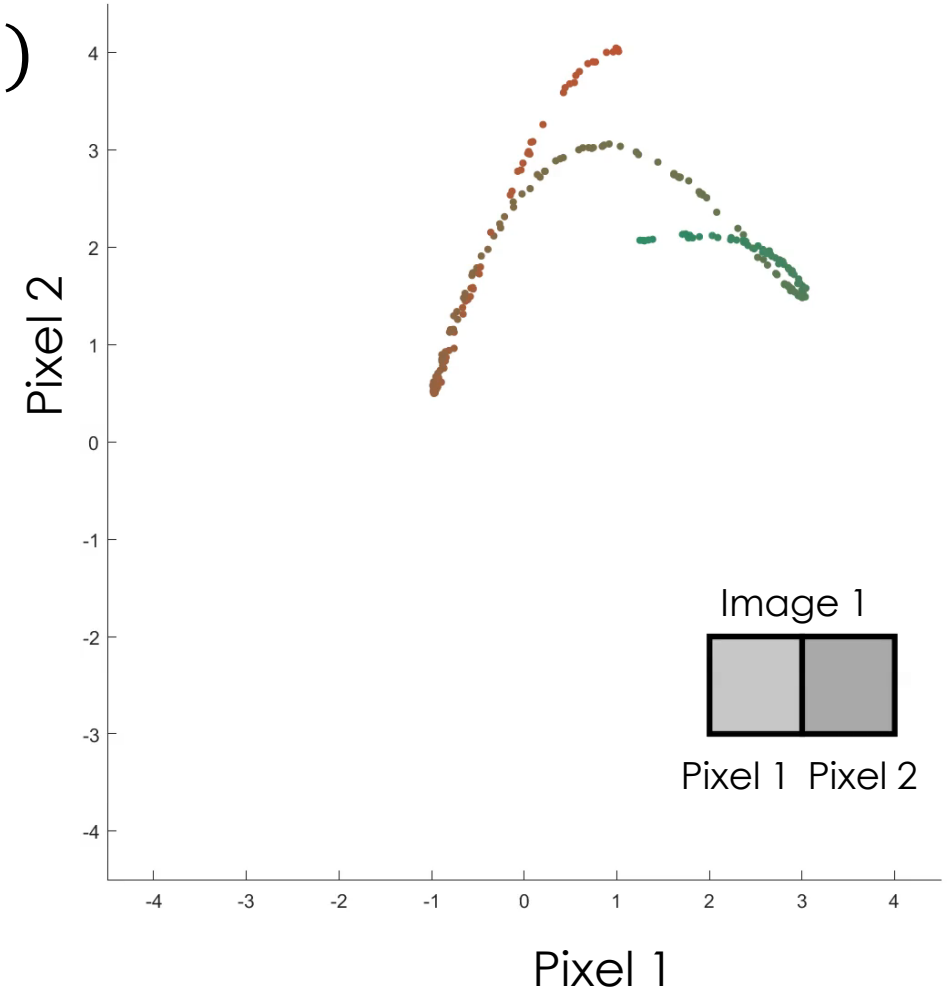
- From distribution of images to $\mathcal{N}(0,1)$
  - Remove (almost) all information
  - Repeat:

$$I^{t+1} = \sqrt{\alpha_t}\, I^t + \sqrt{1-\alpha_t}\, \varepsilon_{t\to t+1}$$

with $\varepsilon_{t\to t+1} \sim \mathcal{N}(0,1)$

α_t some parameter (could be constant)

- Converges to $I^{\infty} \sim \mathcal{N}(0,1)$



Diffusion models: training

- Known distribution at time t
  - If you know I^0:

    $I^t \sim \mathcal{N}(\mu_t, \sigma_t^2)$

    with $\mu_t = \sqrt{\overline{\alpha_t}}\, I^0$, $\sigma_t^2 = (1 - \overline{\alpha_t})\,\mathrm{Id}$

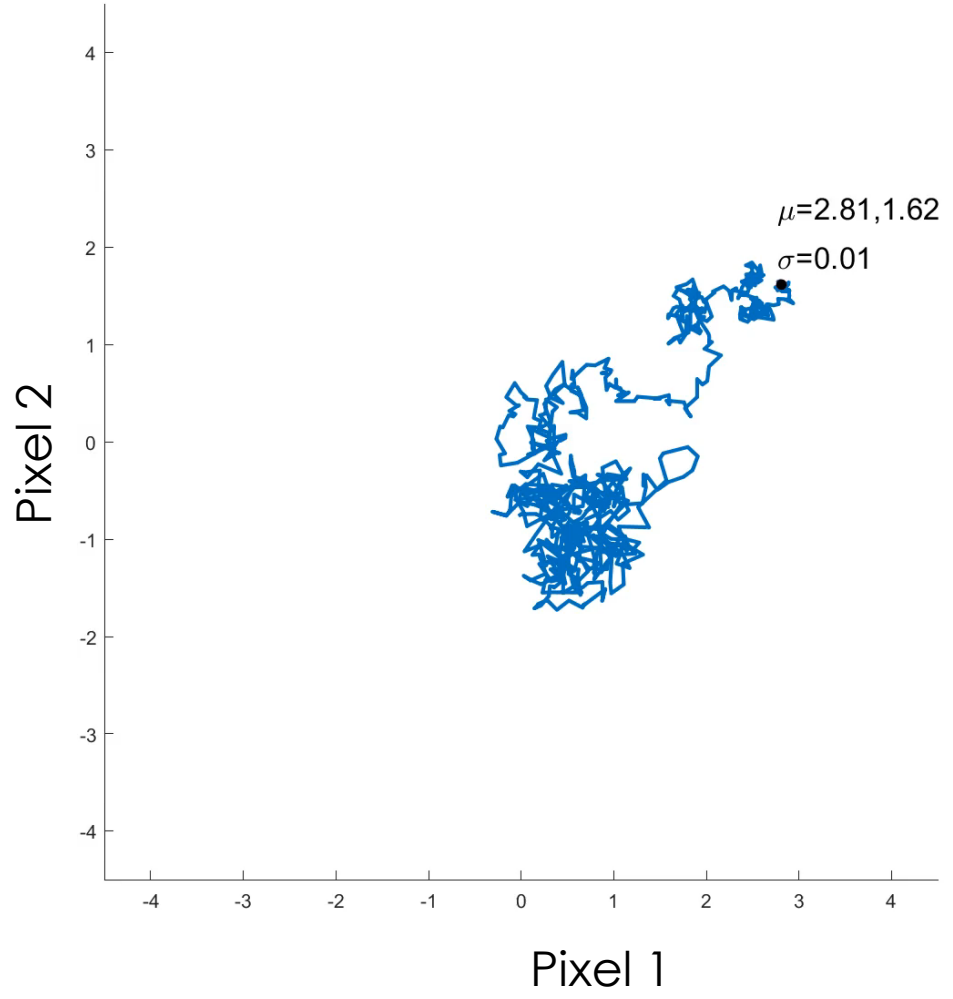
    and $\overline{\alpha_t} = \prod_{i=1}^{t} \alpha_i$

    $$I^t = \sqrt{\overline{\alpha_t}}\ I^0 + \sqrt{1 - \overline{\alpha_t}}\, \varepsilon_{0\to t}$$

    with $\varepsilon_{0\to t} \sim \mathcal{N}(0,1)$

  - You know I^0, the noise-free image, at training time (only)
  - Indeed converges to $\mathrm{I}^\infty \sim \mathcal{N}(0,1)$

  - Allows to get I^t at any time t "for free"

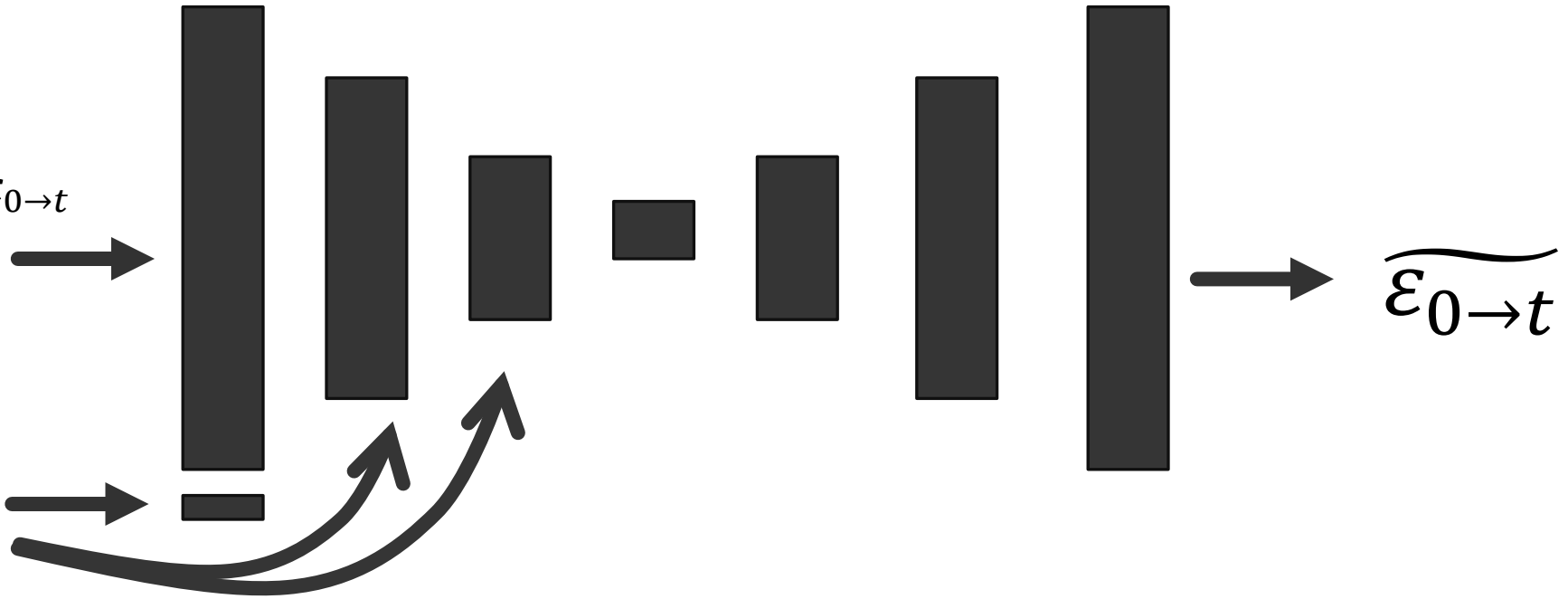


Diffusion models: training

- Train a network to predict noise:

Loss: $\|\widetilde{\varepsilon_{0\to t}} - \varepsilon_{0\to t}\|^2$

$I^t = \sqrt{\overline{\alpha_t}}\, I^0 + \sqrt{1-\overline{\alpha_t}}\, \varepsilon_{0\to t}$



$\widetilde{\varepsilon_{0\to t}}$

- Feed the network with many noisy images at random times $t \in [0, t_{max}]$

Diffusion models: inference

- Start with a random noise image at time $t = t_{max}$:

$$I^{t_{max}} = \underbrace{\sqrt{\overline{\alpha}_{t_{max}}}\, I^0}_{\approx 0} + \underbrace{\sqrt{1-\overline{\alpha}_t}}_{\approx 1}\, \varepsilon_{0 \to t_{max}}$$

$$\varepsilon_{t_{max}} \sim \mathcal{N}(0,1)$$

- Predict noise level

$$I^{t_{max}} = \sqrt{\overline{\alpha}_{t_{max}}}\, I^0 + \sqrt{1-\overline{\alpha}_t}\, \varepsilon_{0 \to t_{max}}$$

Diffusion models: inference

- Remove the noise

$$I^{t_{max}} = \sqrt{\overline{\alpha}_{t_{max}}}\, I^0 + \sqrt{1-\overline{\alpha}_{t_{max}}}\, \varepsilon_{0 \to t_{max}} \quad \sim \mathcal{N}(0,1)$$

$$\Rightarrow I^0 \approx \frac{I^{t_{max}} - \sqrt{1-\overline{\alpha}_{t_{max}}}\, \widetilde{\varepsilon_{0 \to t_{max}}}}{\sqrt{\overline{\alpha}_{t_{max}}}}$$

In practice, task is too difficult: do it progressively

Diffusion models: inference

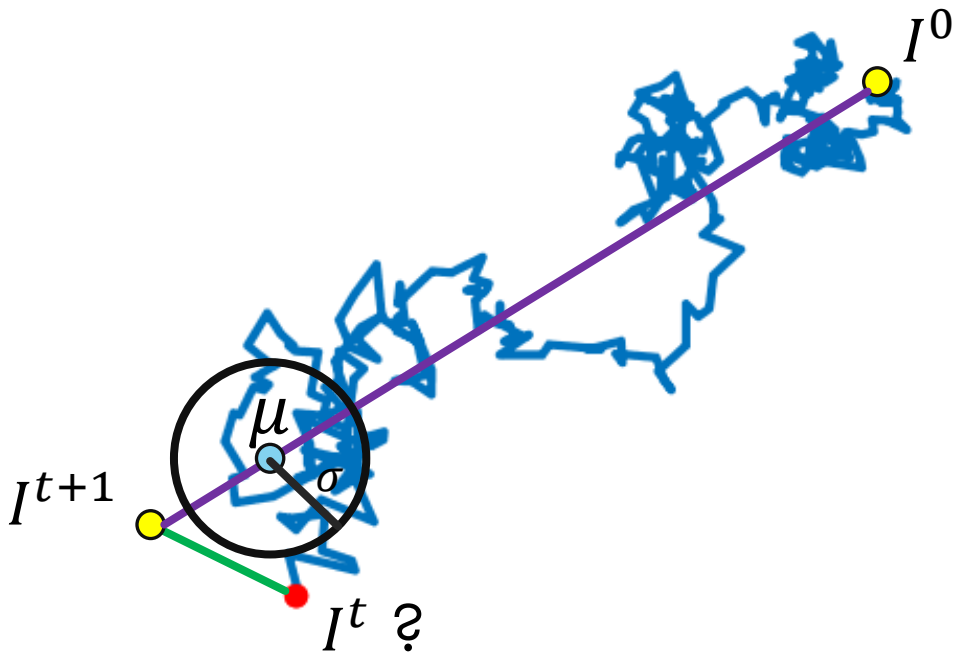
- Remove the noise progressively
  - Given I^0 and I^{t+1}, $I^t \sim \mathcal{N}(\mu, \sigma^2)$ with

$$\mu = \frac{\sqrt{\alpha_{t+1}}\,(1-\overline{\alpha_t})}{1-\overline{\alpha_{t+1}}}\,I^{t+1} + \frac{\sqrt{\overline{\alpha_t}}\,(1-\alpha_{t+1})}{1-\overline{\alpha_{t+1}}}\,I^0$$

$$\sigma^2 = \frac{1-\overline{\alpha_{t-1}}}{1-\overline{\alpha_t}}(1-\alpha_t)$$

  - Don't know I^0
    - Take our previous (poor) estimate:

$$I^0 \approx \frac{I^{t+1} - \sqrt{1-\overline{\alpha_{t+1}}}\;\widetilde{\varepsilon_{0\to t+1}}}{\sqrt{\overline{\alpha_{t+1}}}}$$



Diffusion models: inference

- Remove the noise progressively
  - We get

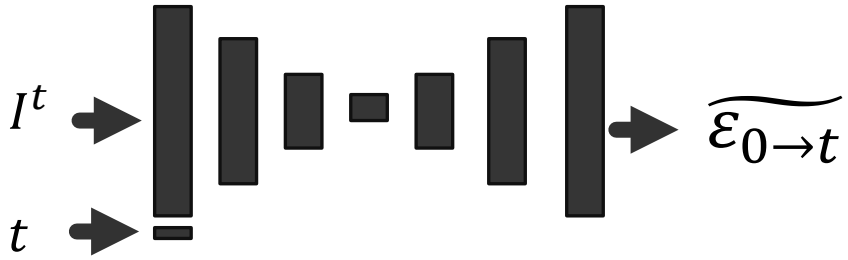
$$I^t = \frac{I^{t+1} - \frac{1-\alpha_t}{\sqrt{1-\overline{\alpha_t}}}\widetilde{\varepsilon_{0\to t+1}}}{\sqrt{\alpha_t}} + \mathcal{N}(0, \sigma^2)$$

Diffusion models: scheduling

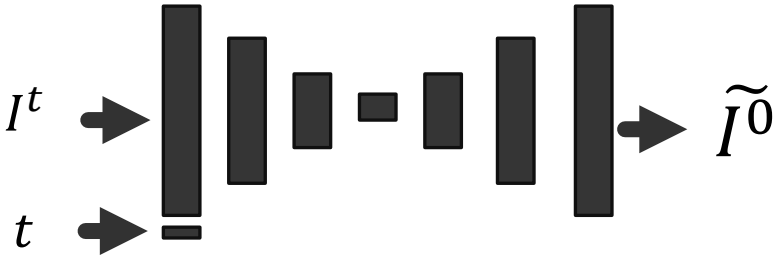
- Constant $\alpha_t \Rightarrow$ never used
- Linear $\alpha_t = 1 - t$ sometimes, but not great.
  - $\beta_t = 1 - \alpha_t = 10^{-4}\,(1 - t) + 0.02t$ in the original paper
  - Idea: near the solution $(t = 0)$, you need more accuracy
- Cosine: $\alpha_t = \cos^2(\pi t)$ better
  - More accuracy near and far from the solution

Diffusion models: prediction

- Originally: predicts noise
  - Claimed that it worked better

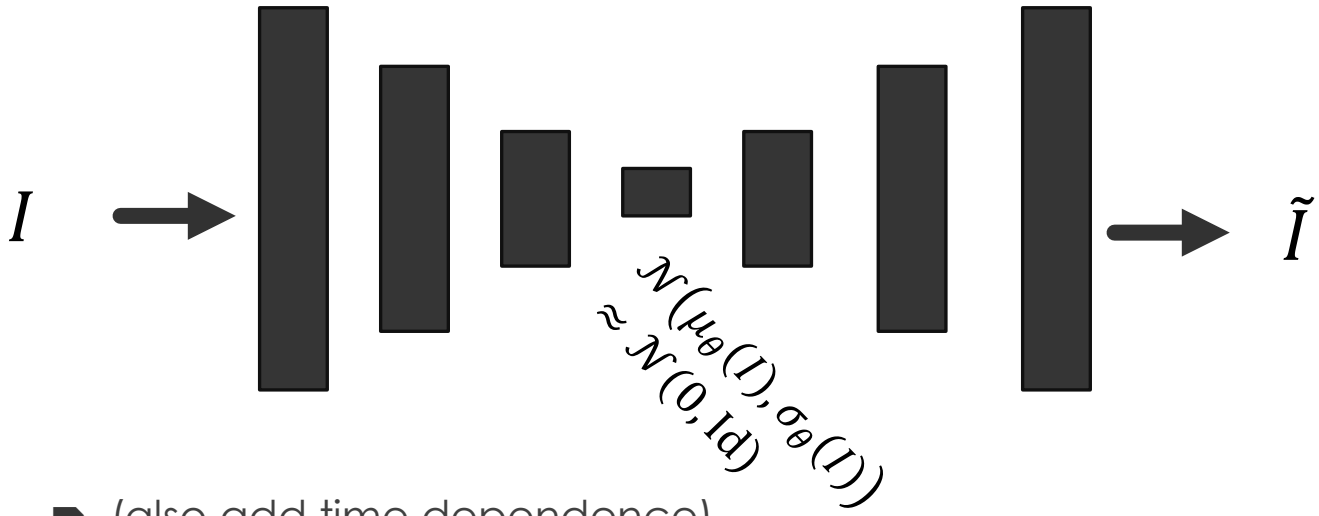


- Recently: predicts image is better
  - +Transformer
  - Directly in pixel space
  - "Back to Basics: Let Denoising Generative Models Denoise" [Li and He, Nov. 2025]



Latent diffusion models

- Same thing as diffusion… but in latent space (i.e. lower dimensional)
- Uses a Variational Auto-Encoder (VAE) to reduce dimensionality
  - Typically, from 512x512x3 ⇒ 64x64x4



  - (also add time dependence)

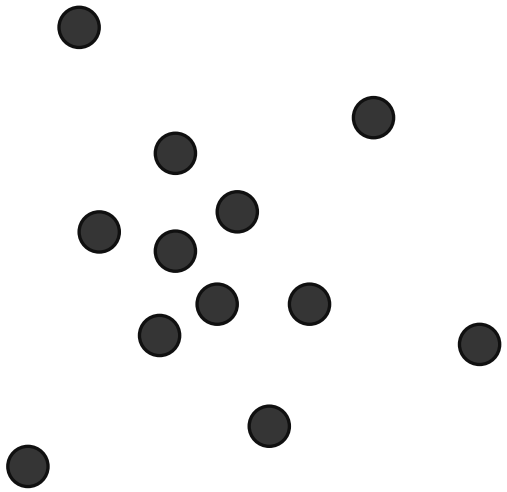
Diffusion models

- Now used by most image generative models
  - Often with conditioning by text
  - Often variants, such as “Latent diffusion models”/Stable diffusion
- Has superseded most other approaches (GANs, VAE..)

Flow matching

- Training
  - Sample from a standard Gaussian distribution
  - Random matchings between Gaussian samples and the target images
  - Learn a velocity field $v(x, t)$ to fit this assignment
- Inference (i.e., image generation)
  - Sample for a standard Gaussian distribution
  - Integrate velocity field (i.e., Euler, RK4…)
- Equivalent to diffusion models (random flow/diffusion from Gaussian to images)

Flow matching: training



$\mathcal{N}(0,1)$
Batch size = 12

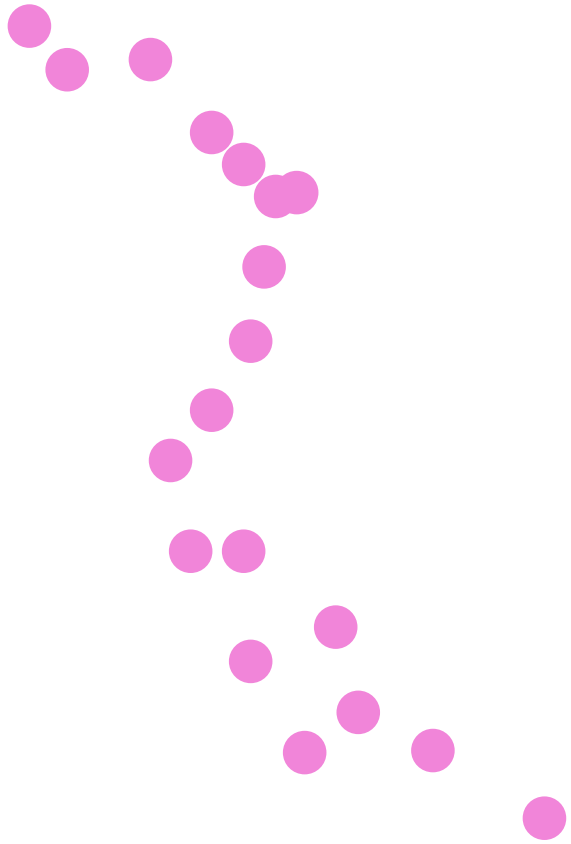
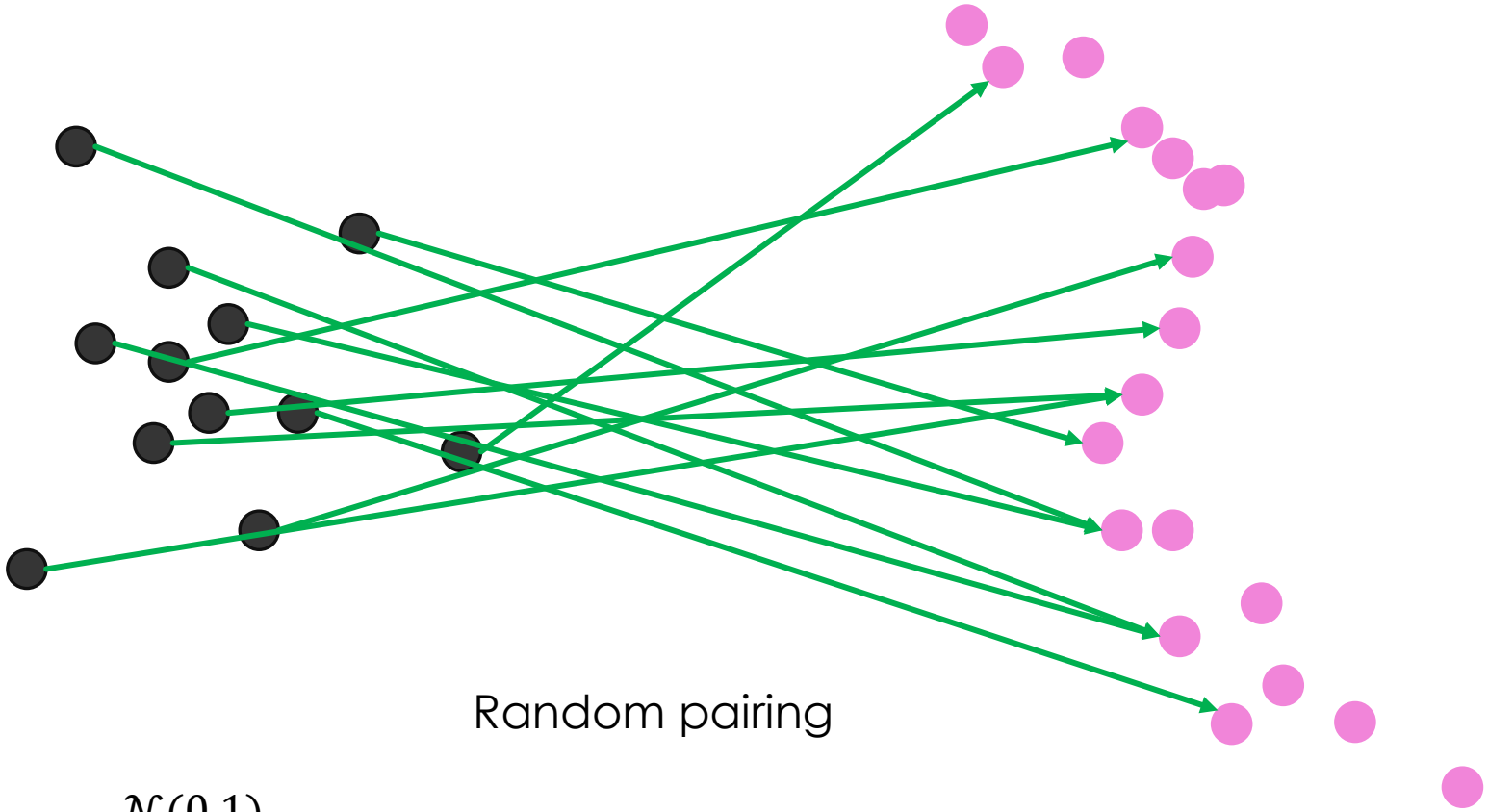
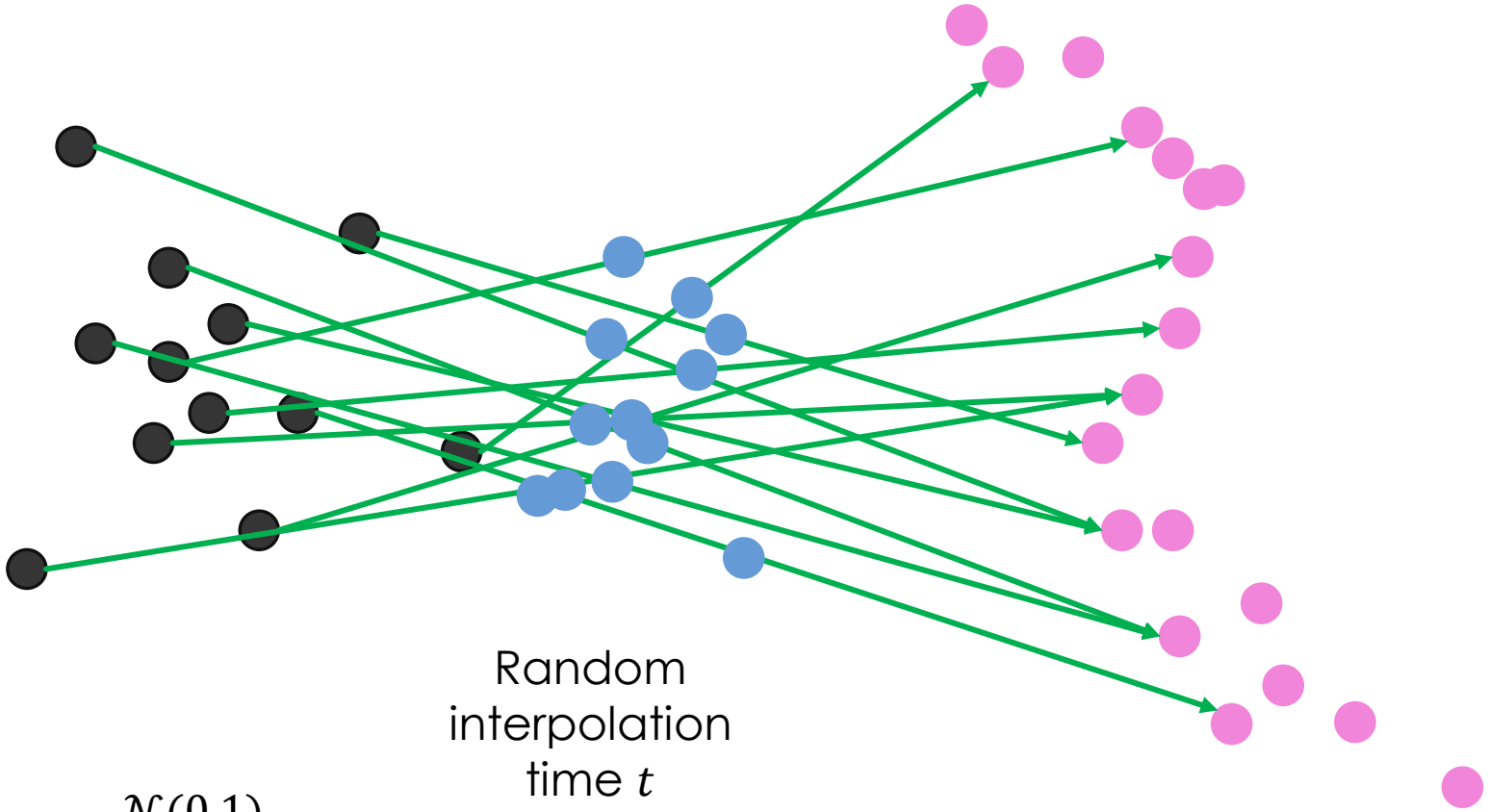


Image dataset

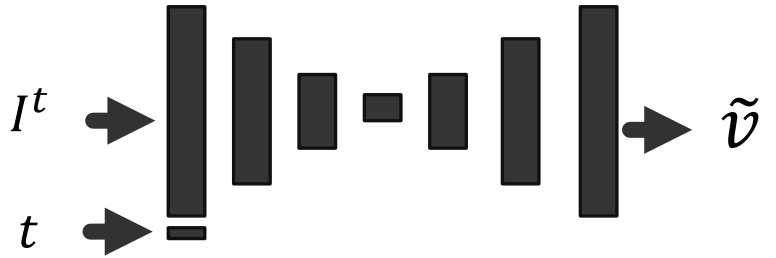
Flow matching: training



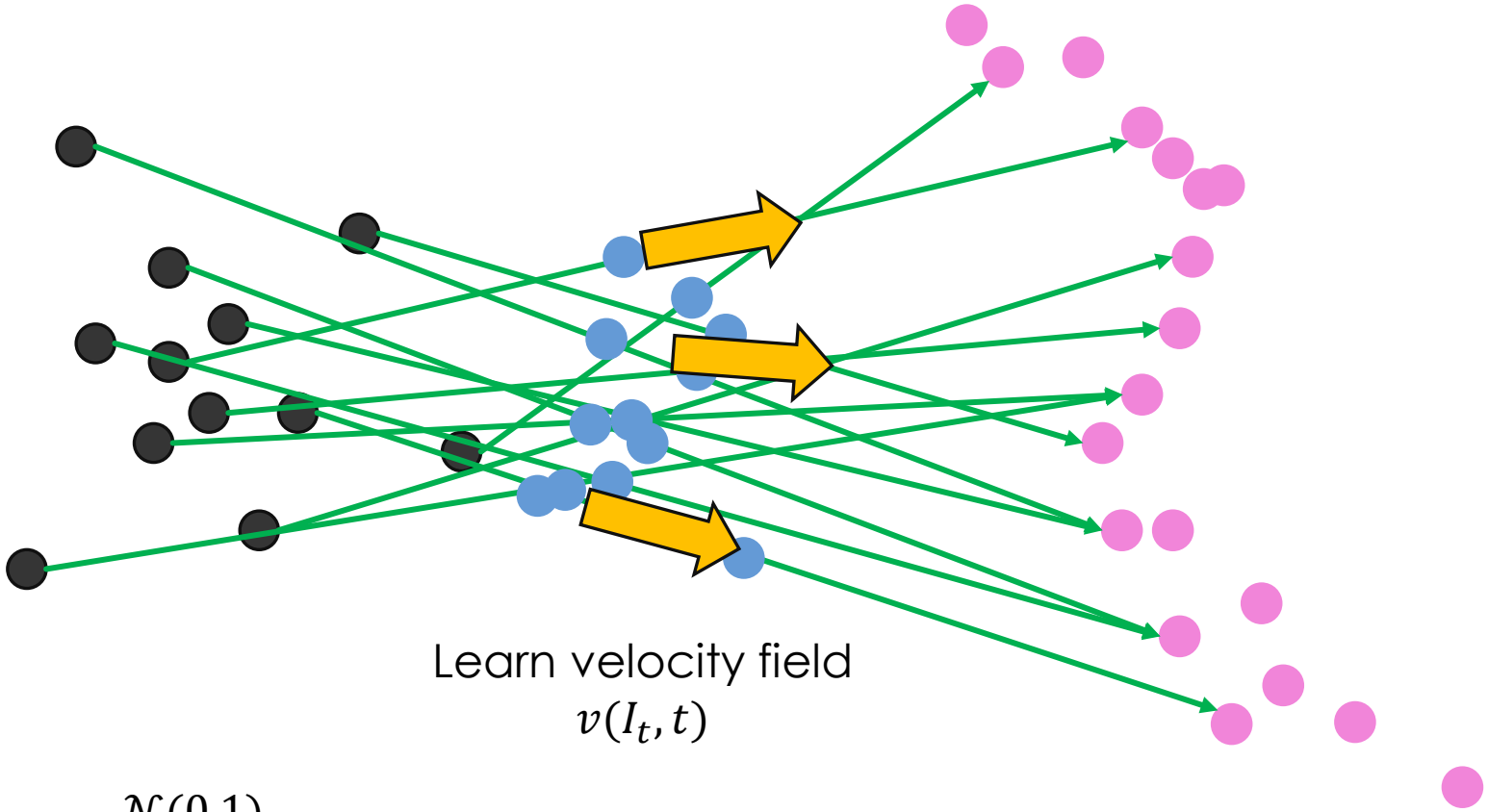
Flow matching: training



Flow matching: training



$\|v(I_t, t) - (Y - X)\|^2$

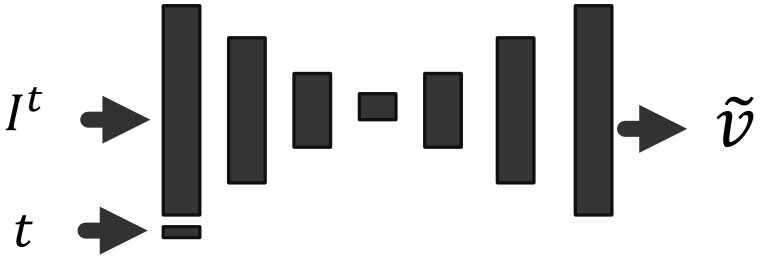


Learn velocity field
$v(I_t, t)$

$\mathcal{N}(0,1)$
Batch size = 12

Image dataset

Flow matching: inference



Predicted velocity field
$v(I_0, 0)$

$\mathcal{N}(0,1)$

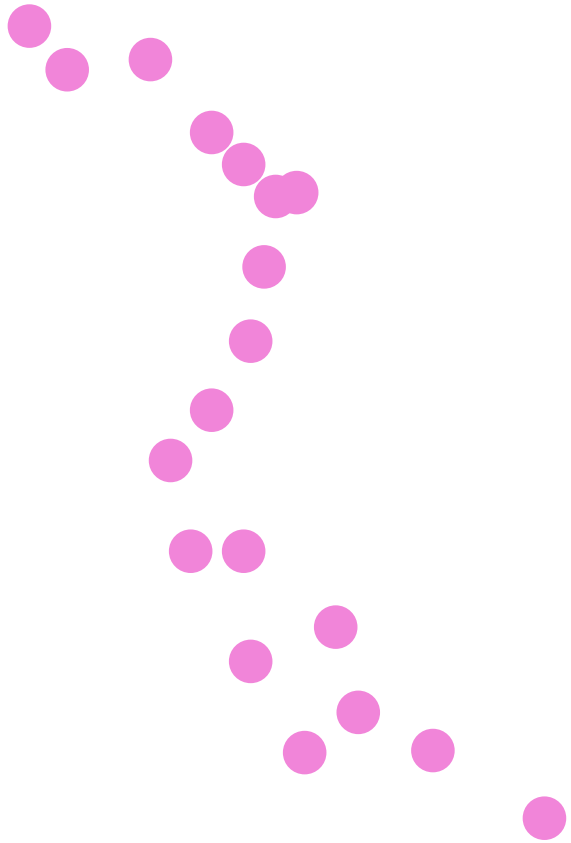
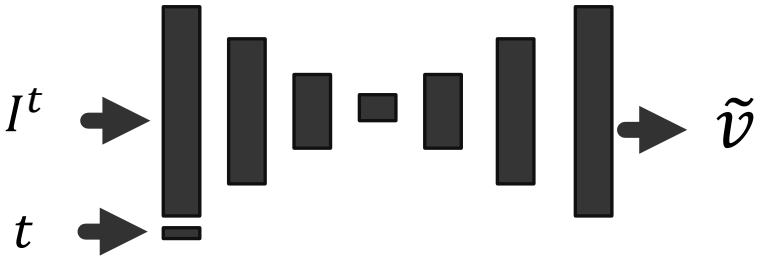


Image dataset

(here I consider t=0 is the “noisy” image…)

Flow matching: inference



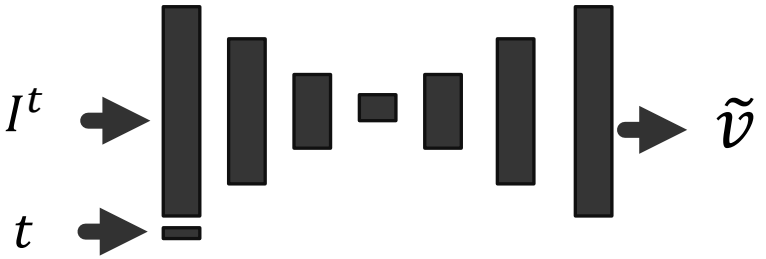
Predicted velocity field

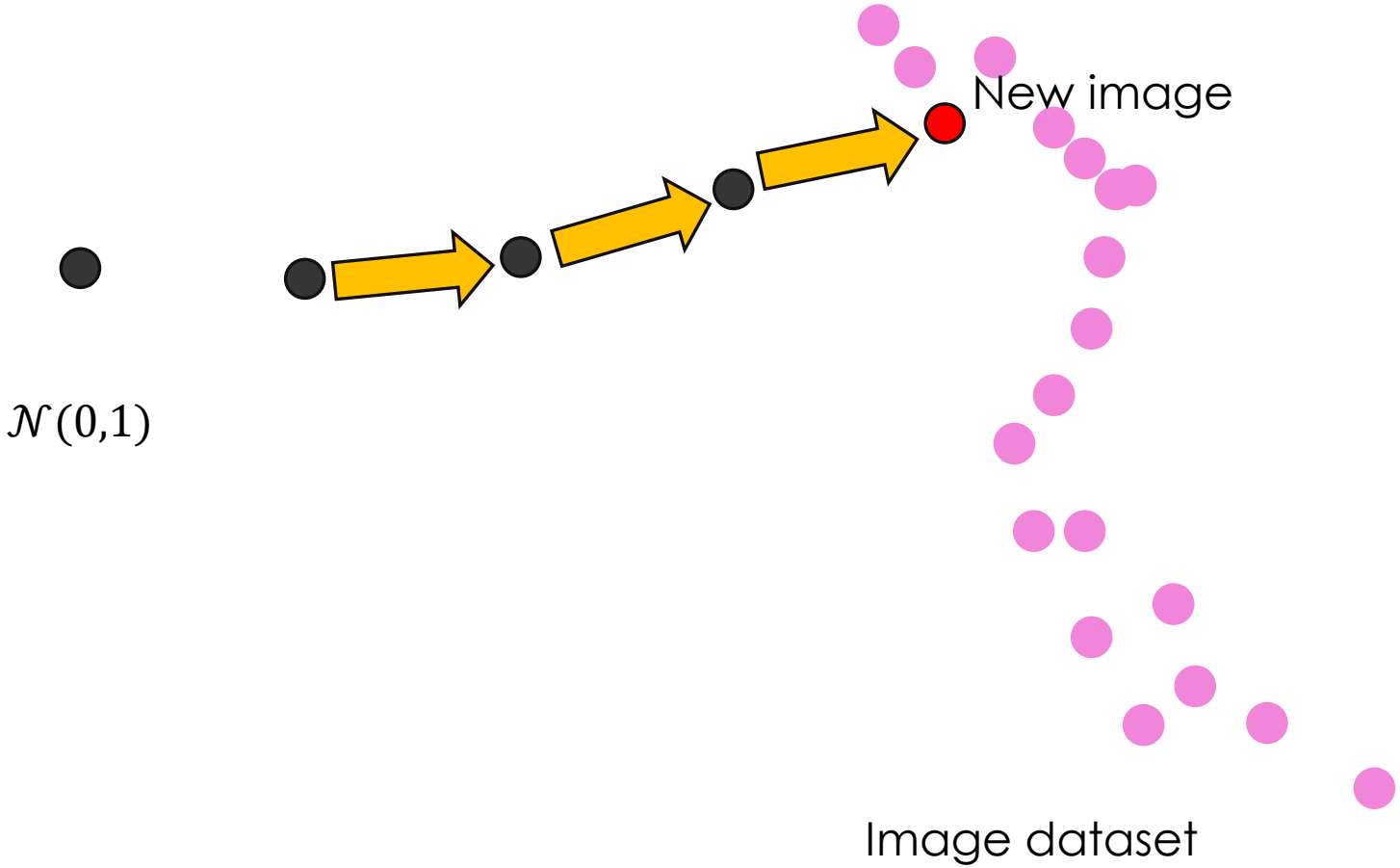
$v(I_{\delta t}, \delta t)$

$\mathcal{N}(0,1)$

Image dataset

Flow matching: inference





Flow matching

- Better matchings than random
  - Optimal transport !
  - With small batches, exact transport remains fast
  - Flow is more straight ⇒ less Euler time steps for inference
- Better than Euler (order 1 : error in $O\left(\frac{1}{\delta t}\right)$)
  - Dopri5 (4th and 5th order with time step control), Runge Kutta 4 (4th order), Dopri8…

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v1 (Février 2022)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v2 (Avril 2022)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v3 (Juillet 2022)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v4 (Novembre 2022)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v5 (Mars 2023)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v5.1 (Mai 2023)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v5.2 (Juin 2023)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v6 (Décembre 2023)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

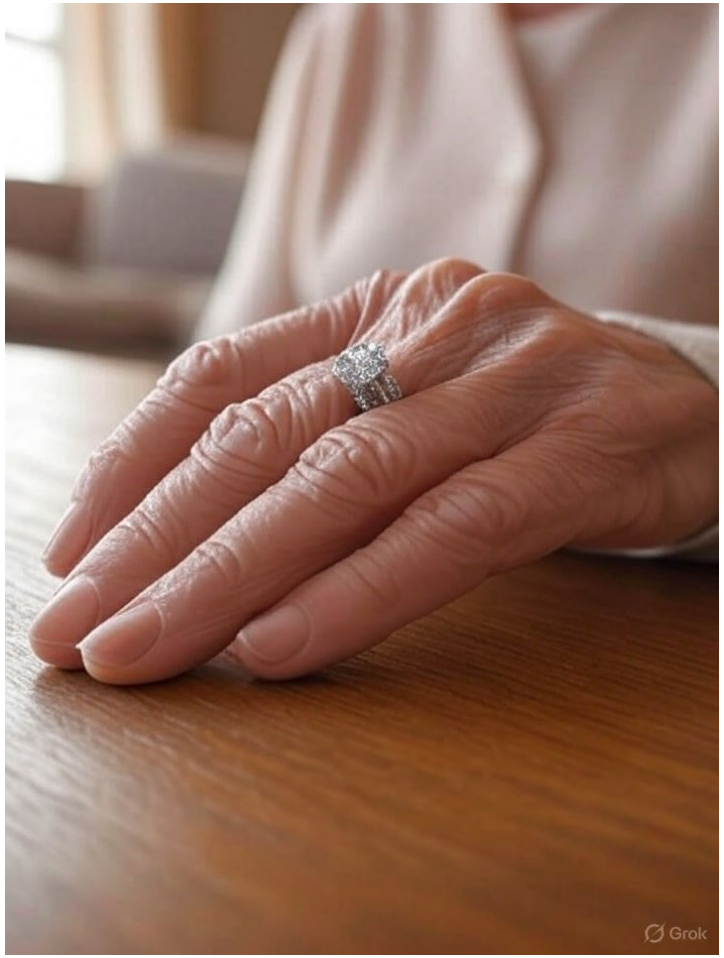
MidJourney v6.1 (Juillet 2024)

Évolution

“Realistic photograph of an elderly woman's hand featuring an expensive wedding ring with a huge diamond” (controle seed, meilleur de 4)

MidJourney v7 (avril 2025)

Comparaison (au 18/06/2025)

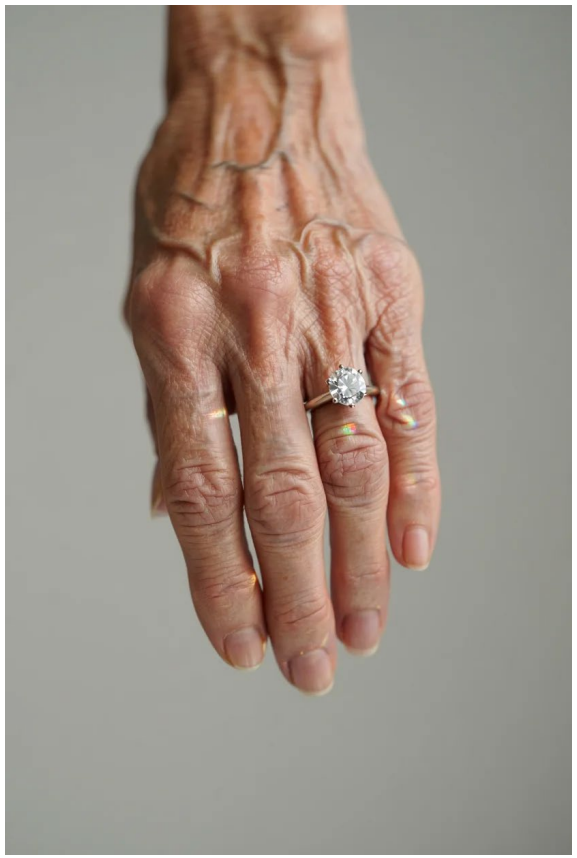


Grok

Sora/ChatGPT

MidJourney v7

Comparaison (au 18/06/2025)



Reve

Adobe Firefly 4
(pas Ultra)

StableDiffusion

Comparaison (au 18/06/2025)

Google Gemini Pro

Mistral Le Chat

Biais

- Tests
  - “Realistic photograph of a CEO in a big company behind a desk”
  - “Realistic photograph of a flying superhero saving someone falling a skyscraper”
  - “Realistic photograph of a top model walking on a stage with beautiful clothes”
  - “Realistic photograph of a secretary typing on a computer”
  - “Realistic photograph of a couple shaking hands in an office”

  - (pas de soucis de genre en lange anglaise)
  - MidJourney, ChatGPT/Sora, Adobe FireFly, Grok

Repérer les images gé

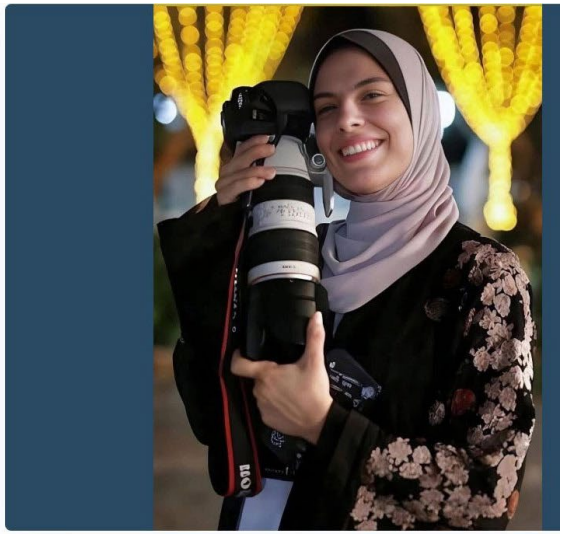
- Sites de détection moyennement fiables
- Gros problèmes
  - Nombre de doigts,

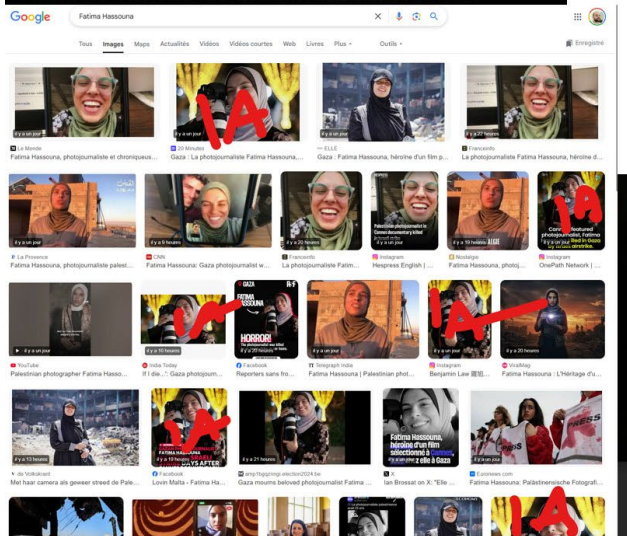
Repérer les images générées

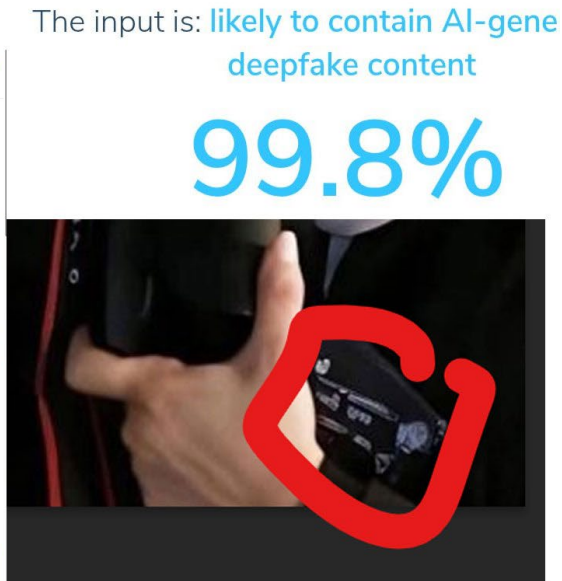
Vincent Flibustier
(NordPresse)

- Sites de détection moyennement fiables ⇒
  ⇒
- Gros problèmes
  - Nombre de doigts,
  - Texte illisible

= Upsampling par IA





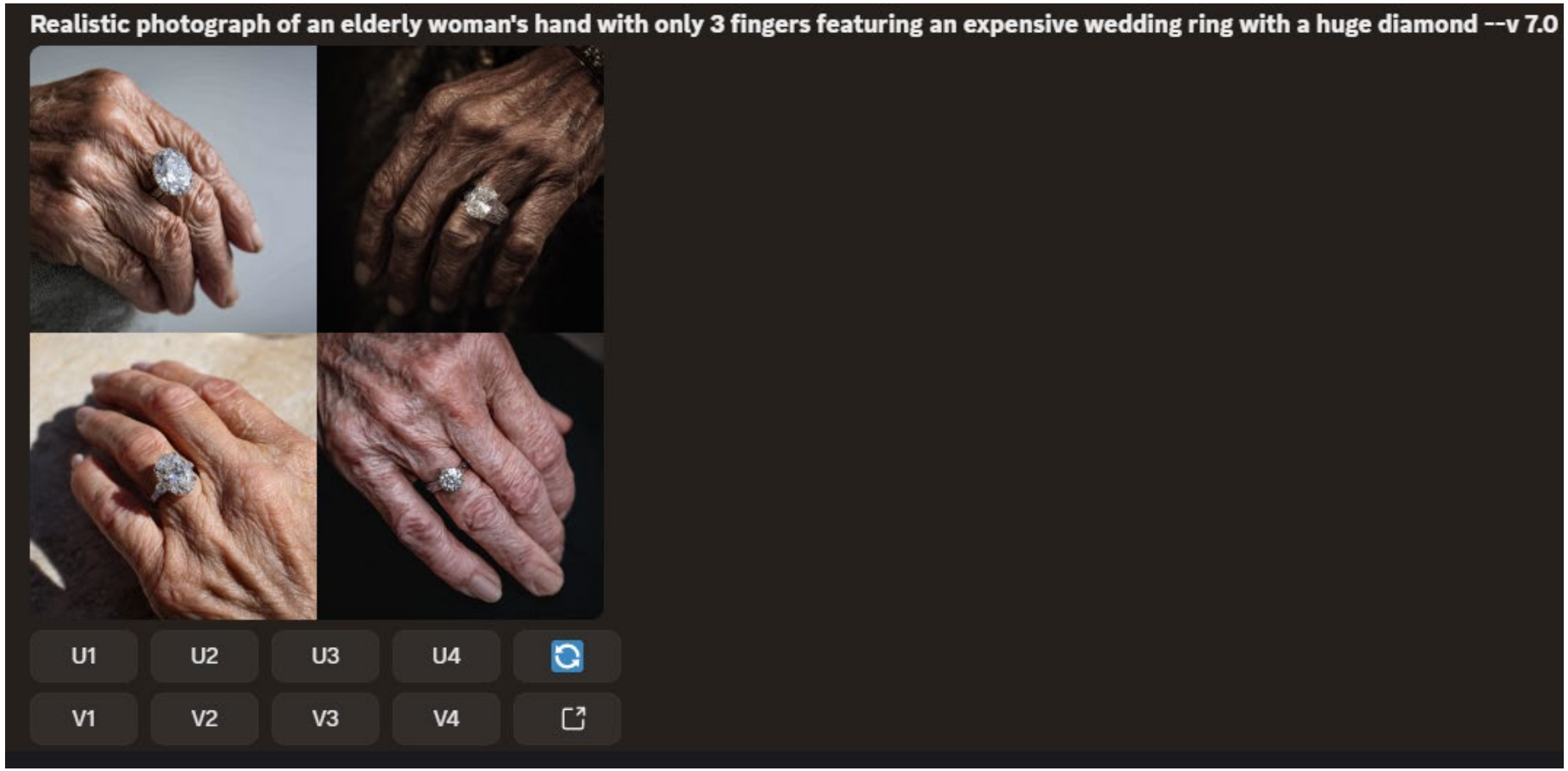


Repérer les images généré

- Sites de détection moyennement fiables
- Gros problèmes
  - Nombre de doigts,
  - Texte illisible
  - Proportions

Repérer les images générées

- Gros problèmes se “règlent” tout doucement



Repérer les images générées

- Détails plus subtils
  - Hautes fréquences, cheveux / Peaux trop lisses
  - Incohérences : manque de symétries (ou trop), anachronies, si plusieurs images : personnages pas consistants dans le temps
- Esthétique semblable
  - Sourirs
  - Flou en arrière plan (bokeh/profondeur de champs)
- Contexte clickbait pour boomer
  - « Je suis à l'hopital et personne ne me souhaite mon anniversaire »
  - « J'ai fait une tour Eiffel en allumettes et personne ne la partage »
- Sinon, Reverse image search ⇒ source
  - Tineye, google reverse image search, yandex