

---

# DENOISING DIFFUSION PROBABILISTIC MODEL - APPLICATION TO SATELLITE AND HANDWRITTEN DIGITS IMAGE GENERATION

---

FYS5429 - PROJECT 2

 **Romain Corseri**  
University of Oslo  
romain.corseri@gmail.com

June 5, 2024

## ABSTRACT

This project investigates the potential of denoising diffusion probabilistic models for image generation, with a focus on optical satellite images. We adapt a PyTorch implementation of a diffusion model for image resolution enhancement to generate realistic satellite images and use the MNIST handwritten digit dataset to benchmark our model architecture. Our experiments consist in testing the amount and type of training images to demonstrate the effectiveness of diffusion models in generating high-quality images. The results highlight the striking potential of diffusion model for satellite imagery. Despite the computational challenges and tractability, we emphasize the increasing value of pre-trained "foundations" models for democratizing access to advanced generative capabilities and suggest further research into practical applications in remote sensing and Earth observation.

## 1 Introduction

In recent years, deep generative models have gained considerable attention with striking image generation capability and the advent of large language models (Jonathan Ho [2020], Kingma; and Welling [2019], Nichol [2021], Goodfellow [2014]). Many researchers believe that these generative models have the potential to profoundly impact research methodologies, potentially triggering a new scientific revolution (Kuhn [1962]). In this work, we explore the potential of denoising diffusion probabilistic models for image generation, with a particular focus on satellite imagery. The availability of vast amounts of optical satellite images online presents a significant corpus of training data for diffusion models. There are countless applications in the field of change detection from satellite image analysis. Detection change using synthesized images from diffusion model (Patel [2022]) could contribute from disaster mitigation, climate-change monitoring to military surveillance. In this work, we adapt a PyTorch implementation of diffusion model for image resolution enhancement (Chitwan et al. [2021]) to satellite image generation. We also use MNIST handwritten digits images to benchmark our model architecture requiring less training images and GPU-cycles. Finally, we discuss diffusion model performance, tractability and the potential of open-access large pre-trained "foundations" diffusion model for scientific purposes.

## 2 Denoising diffusion probabilistic model

Denoising diffusion probabilistic models, commonly referred to as diffusion models, are part of the broader class of probabilistic generative models. This category also includes generative adversarial networks (GAN), variational autoencoders (VAE), energy-based models, and others.(Goodfellow [2014], Kingma; and Welling [2019], Luo [2022]). More specifically, diffusion models are hierarchical variational auto-encoders that fulfill three conditions:

- The latent spaces dimension is equal to the input data dimension
- The encoder is predefined as a linear Gaussian model
- The parameters of the Gaussian model of the encoder vary over time steps so the image at the final time step  $t = T$  is completely contaminated by Gaussian noise (Fig. 1a).

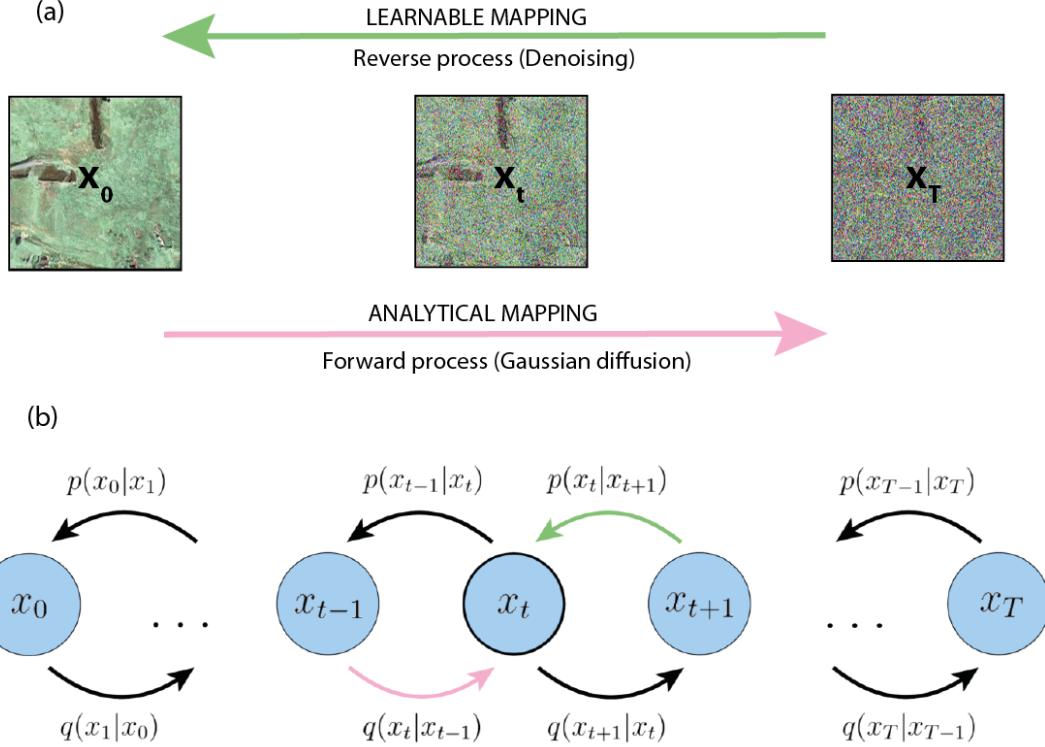


Figure 1: a) Conceptual depiction of denoising diffusion model b) Probability distributions for both forward and reverse Markov chains in a typical diffusion model for unconditional image generation (figure modified from Luo [2022])

In the two following subsections, we go through the mathematical basics of diffusion models. We will spell out the most important equations, starting from the forward gaussian diffusion modelling and finally the description of the reverse learnable denoising process (Fig. 1a). The section is adapted from Luo [2022], Jonathan Ho [2020] where the mathematical proofs are fully developed.

## 2.1 Gaussian diffusion process

An important property of diffusion models is that the forward process (also called "encoder" in VAE terminology) can be analytically expressed as the sampling on a Markov chain that gradually adds Gaussian noise to the initial images following a variance scheduler  $\beta_1, \dots, \beta_t, \dots, \beta_T$  until the image becomes purely Gaussian noise. In our case,  $\beta_t$  and  $t$  the number of time steps are hyperparameters. Then, at an arbitrary time steps  $t$ , the sampling process of an image  $\mathbf{x}_t$  (Fig. 1) is expressed as:

$$q(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_t; \sqrt{\alpha_t}\mathbf{x}_0, (1 - \alpha_t)\mathbf{I}) \quad (1)$$

$$\text{with } \alpha_t = 1 - \beta_t \text{ and } \overline{\alpha_t} = \prod_{i=1}^t \alpha_i \quad (2)$$

## 2.2 Reversing diffusion by optimizing the denoising model

The denoising model takes as input the noisy image  $\mathbf{x}_t$  and output a denoised image  $\mathbf{x}_{t-1}$  (Fig. 1b). It can proven that, under some assumption, the denoising step from time  $t$  to  $t - 1$  can be expressed as:

$$q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t\mathbf{I}) \quad (3)$$

$$\text{where } \tilde{\mu}_t(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\alpha_{t-1}}\beta_t}{1 - \overline{\alpha_t}}\mathbf{x}_0 + \frac{\sqrt{\alpha_t}(1 - \overline{\alpha_{t-1}})}{1 - \overline{\alpha_t}}\mathbf{x}_t \text{ and } \tilde{\beta}_t = \frac{1 - \overline{\alpha_{t-1}}}{1 - \overline{\alpha_t}}\beta_t. \quad (4)$$

Given that the ultimate objective of the diffusion model is to learn a Markov chain  $p_\theta(\mathbf{x}_0)$  that aims to reproduce the original data distribution (i.e. the original image  $p(\mathbf{x}_0)$ ) by maximizing the Evidence Lower Bound (ELBO). The Kullback–Leibler (KL) divergence  $\mathcal{D}_{KL}$  measures the dissimilarity between two probability distributions. Here, the ELBO maximisation problem is equivalent to minimizing  $\mathcal{D}_{KL}$  between the learned probability distribution  $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)$  and the ground-truth denoising transition step  $q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0)$ :

$$\arg \min_{\theta} \mathcal{D}_{KL}(q(\mathbf{x}_{t-1}|\mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t)) \quad (5)$$

Which, in turn, simplifies to (see Jonathan Ho [2020] for the mathematical proof):

$$\arg \min_{\theta} \frac{\overline{\alpha_{t-1}}}{2(1 - \overline{\alpha_{t-1}})} \frac{(1 - \alpha_t)}{(1 - \overline{\alpha_t})} \| \hat{\mathbf{x}}_\theta(\mathbf{x}_t, t) - \mathbf{x}_0 \|_2^2 \quad (6)$$

In this framework, optimizing a diffusion model boils down to training a convolutional neural network  $\hat{x}_\theta$  to predict the original ground-truth image  $\mathbf{x}_0$  from an arbitrarily noisified version of it  $\mathbf{x}_t$  (Luo [2022]).

### 2.3 The Super-Resolution diffusion model (SR3) and its PyTorch implementation

The diffusion model architecture and Pytorch implementation used in this work are based on the work of Chitwan et al. [2021], Patel [2022]. The implementation of Chitwan et al. [2021] is a conditional denoising diffusion model, denoted "SR3", that enhances the resolution of the input images, for example from 16 x 16 to 128 x 128 pixels. In contrast, Patel [2022] adapted the latter implementation to unconditional satellite image synthesis and trained a diffusion model on approx. 500000 unlabeled Google Earth Engine screendumps. The PyTorch implementation of Patel [2022] is provided in the github repository DDPM-CD: Denoising Diffusion Probabilistic Models as Feature Extractors for Change Detection. This implementation served as a basis for the training and sampling experiments reported in the result section.

The "SR3" model is based on U-Net architecture including a number of residual blocks, up-sampling, drop-out and channel multipliers at different resolutions (Fig. 2). The details of the architecture of the three diffusion models used in this work are summarized in Table. 1. To grasp the power of satellite image synthesis with generative diffusion models, we also experiment sampling on a pre-trained diffusion model made of 500M parameters approx. trained on 500000 unlabeled Google Earth Engine screendumps (Chitwan et al. [2021]), which is untractable for GPU-poor ML practitioners. The weights and biases of this pre-trained diffusion model can be downloaded here.

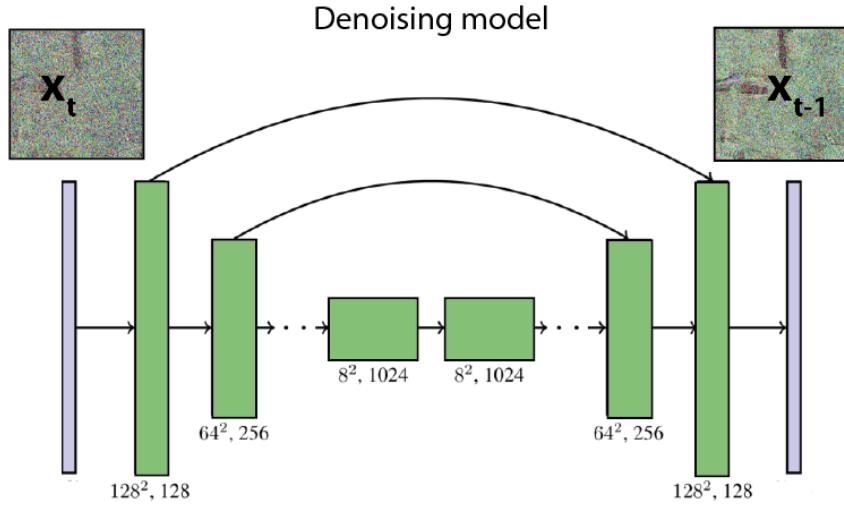


Figure 2: "SR3" super-resolution diffusion model based on U-net architecture from Chitwan et al. [2021]

### 3 Image datasets

#### 3.1 Optical satellite images

The experiments described in this report were conducted using LEVIR-CD256 change detection satellite image dataset from Patel [2022], available for download at <https://www.dropbox.com/scl/fi/r28vh4c6soxk7q9l2hg1a/LEVIR-CD256.zip>. LEVIR-CD256 is of approx. 10000 optical satellite images, made of 3-channels and a resolution of 256 by 256 pixels. They were all collected from Google Earth Engine.

#### 3.2 Handwritten digits

The MNIST dataset <http://yann.lecun.com/exdb/mnist/> is a large collection of handwritten digits commonly used for training and testing in computer vision problems. It contains 60,000 training images and 10,000 test images of digits from 0 to 9, with a resolution of 128 by 128 pixels in gray scale. The dataset serves as a benchmark for evaluating image processing systems and has been pivotal in the development of various neural network and deep learning techniques. In this work, we use a subset of MNIST data (1000 images) to train a denoising diffusion model (section 4.1). We downloaded the MNIST image dataset in png format from a Kaggle repository: <https://www.kaggle.com/datasets/alexanderyyy/mnist-png>.

Table 1: "sr3-type" diffusion model architecture experimented in the framework of this project

Task	Channel Dim	Depth Multipliers	# ResNet Blocks	# Parameters	# input images
Training DDPM	64	[1, 2, 8]	2	45M	500 MNIST images
Training DDPM	64	[1, 2, 4, 8]	2	55M	1000 Satellite images
Sampling DDPM	128	[1, 2, 4, 8, 8]	2	391M	Pre-trained

### 4 Results

#### 4.1 MNIST image generation

We trained a "sr3-type" diffusion model (see first line in Table. 1) to generate handwritten digits using 1000 MNIST inputs images with 28 by 28 pixels resolution. The training is performed on educloud research machines (Nvidia RTX3090 24GB 24 CPU cores, 64GB RAM). The model training is evaluated on pixel-by-pixel loss metrics based on the generated image and the ground-truth image. We achieved a loss of 4.4841e-03 in approx. 6 hours, with an initial loss of 1e-01. The number of times steps is set to 2000 and the noise (variance) follows a cosine schedule from 1e-06 to 0.01.

On Fig. 3, we show a typical example of the sampling process of the learned diffusion model and showing how the denoising is performed along the reverse diffusion chains at 11 time steps. Qualitatively, we find the digit 7 is adequately reproduced and easily recognizable. On Fig. 4, nine sampling results are displayed, showing a varying degree of quality. We find that out of nine generated images, seven handwritten digits were adequately generated.

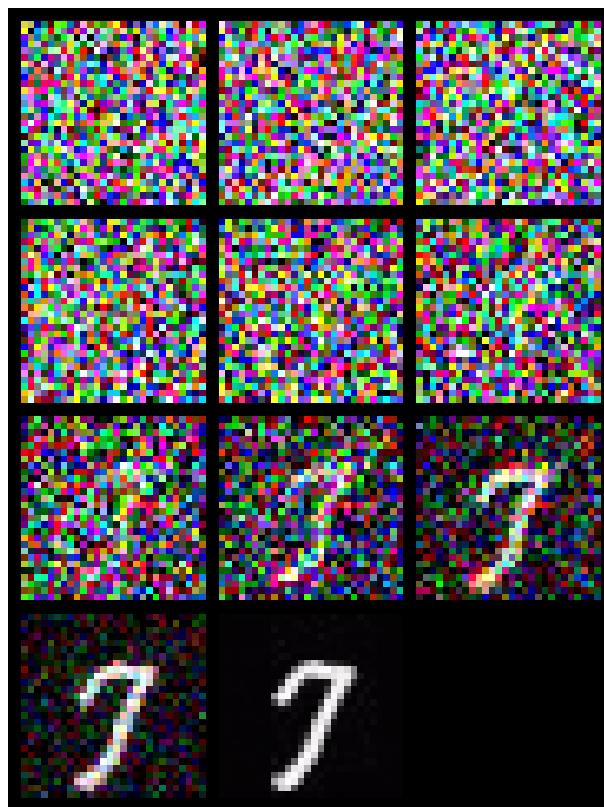


Figure 3: Denoising process from the learned "sr3-type" diffusion model trained on 1000 MNIST images

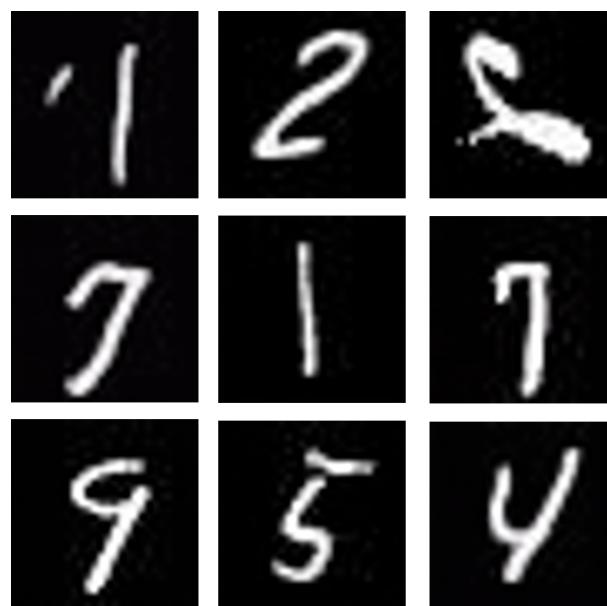


Figure 4: Sampling results showing generated handwritten digits from the trained "sr3-type" diffusion model trained on 1000 MNIST images

## 4.2 Satellite image generation

### 4.2.1 Training a diffusion model in 1000 satellite input images

We trained a "sr3-type" diffusion model (see second line in Table. 1) to generate satellite images using 1000 LEVIR-CD inputs images with 256 by 256 pixels resolution. The training is performed on educloud research machines (Nvidia RTX3090 24GB 24 CPU cores, 64GB RAM). For the experiment, we chose to limit the number of inputs images to reduce the number of GPU-cycles and make the project computationally manageable in the framework of a semester project. We do expect poor generalization capability but we find that the results allow to grasp the power of diffusion model for remote-sensing and Earth observation. The training results are evaluated on pixel-by-pixel loss metrics based on the generated image and the ground-truth image. We achieved a loss of 2.0e-03 in approx. 7 hours, with an initial loss of approx. 1. The number of times steps is set to 2000 and the noise (variance) follows a linear schedule from 1e-06 to 0.01.

On Fig. 5, we show a typical example of the sampling process of the trained diffusion model, illustrating the performance of the denoising process along the reverse diffusion chains at 11 time steps. Qualitatively, we find the synthesized image does not seem to represent realistic landscape features. Some lookalike of dirt roads can be observed but they do not form a realistic network as they are abruptly terminated. In the next subsection, we compare the sampling results from the trained diffusion model to 10 times larger diffusion model trained on a 100 time larger training dataset.

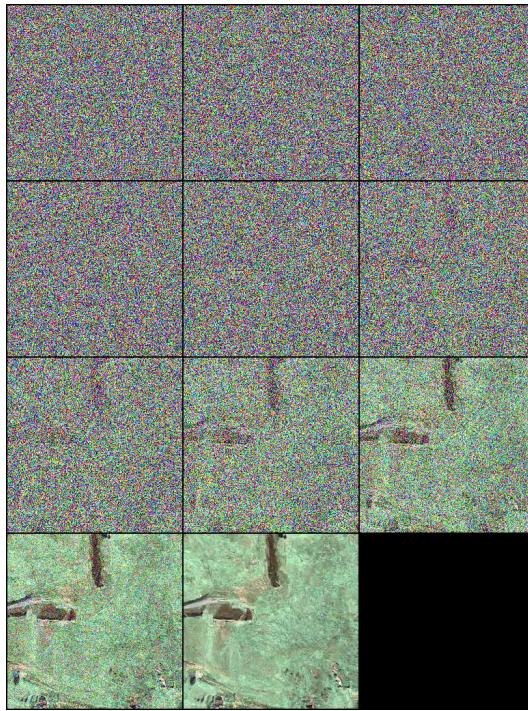


Figure 5: Sampling results from diffusion model trained on 1000 satellite images

### 4.2.2 Sampling a pre-trained model from Patel [2022]

Two synthesized satellite images from the pre-trained diffusion model containing 391M weights and biases (See third line in Table. 1) are displayed in Fig. 6 along with the learned denoising chain of images. Comparing with images generated from the diffusion model trained in section 4.2.1, the generated satellite images are stunning in both level of realistic details (trees, shadows, buildings) and their variability: urban area with buildings and road infrastructures to forest or field crop areas including seasonal changes and partial snow cover). On the Educloud research nodes (Nvidia RTX3090 24GB 24 CPU cores, 64GB RAM), the sampling process for a single image takes approx. 45 minutes.

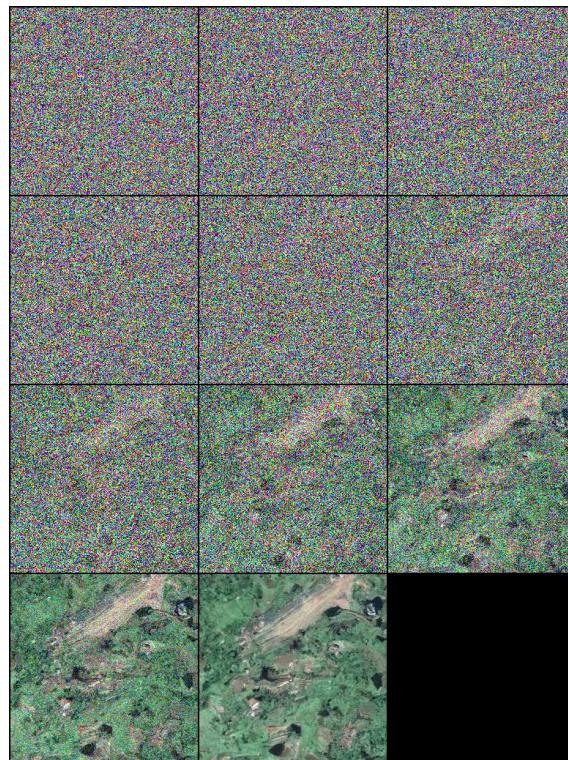
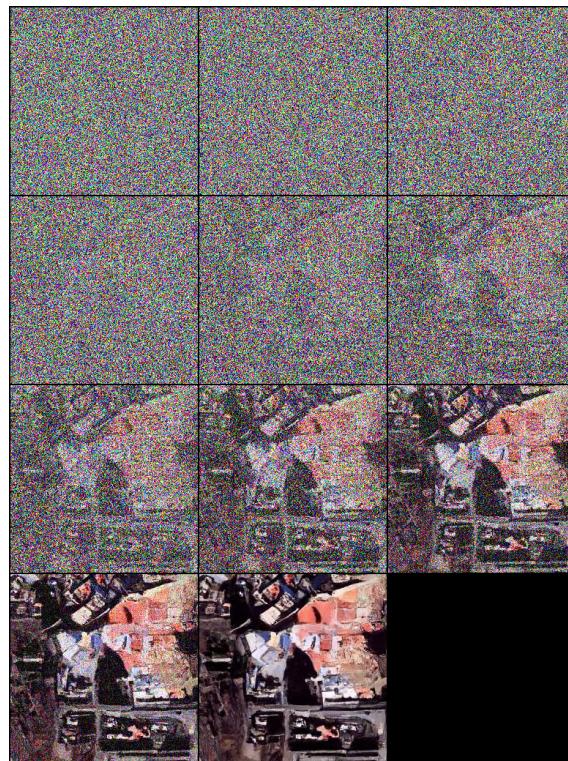


Figure 6: Sampling results from a pre-trained diffusion model from Patel [2022], based on a dataset of 500000 satellite images.

## 5 Discussion and conclusion

### 5.1 Tractability of diffusion models

The results presented in this report confirm the potential of diffusion model for satellite image generation. Sophisticated diffusion models like "sr3" are "convenient" to implement, modify and train in PyTorch. We have adapted the codes to handle handwritten digits and satellite images from the implementation of Patel [2022], Jonathan Ho [2020]. However, we note that the size of diffusion models are consistently very large (ranging from 10M to 100M parameters) and therefore requires a lot of computing resources for training. For application in remote-sensing and Earth observation, we realize that generalization capability of generative models largely depend on the amount of training images. Gathering 500000 satellite images is not a straightforward process and training a large diffusion model made of 390M parameters would take weeks on the computer used for diffusion model training for this piece of work (i.e Nvidia RTX3090 24GB 24 CPU cores, 64GB RAM).

### 5.2 The rise of pre-trained "foundation models"

This work shows that it is practical and relatively easy to download and use pre-trained diffusion model available open source online. Indeed, such "foundation" models are trained on a very large corpus of unlabeled training data and have a great potential for fine-tuning lighter diffusion models, or any other downstream tasks on less GPU-intensive learning problems (Patel [2022]). Such foundations model could help democratize generative models and unleash their increasing power to a larger audience.

### 5.3 Further work

This project deals with satellite image generation with diffusion model but does not investigate the practical applications of diffusion models for Earth Observation data. The potential of diffusion models for change detection in satellite images is still unexplored and we can foresee important developments in remote-sensing applications like disaster relief (earthquakes, fires), climate-related (glacier, ice-sheet) monitoring or military surveillance. Finally, the amount of open-source satellite imagery is dramatically increasing and are readily available online with tools like Google Earth Engine. We conclude that denoising diffusion probabilistic models will likely be a tool of choice for analysis of remote-sensing and Earth Observation data. From 2021 and onward, the shift to generative models in the industry has given rise to the new "foundation model" paradigm, potential revolutionizing scientific methodologies.

## References

- Pieter Abbeel Jonathan Ho, Ajay Jain. Denoising diffusion probabilistic models. *ArXiv*, 2020. doi:<https://doi.org/10.48550/arXiv.2006.11239>.
- Diederik P. Kingma; and Max Welling. An introduction to variational autoencoders. *arXiv*, 2019. doi:<https://arxiv.org/abs/1906.02691>.
- Prafulla Dhariwal; Alex Nichol. Diffusion models beat gans on image synthesis. *ArXiv*, 2021. doi:<https://arxiv.org/abs/2105.05233>.
- Goodfellow. Generative adversarial network. 2014.
- Thomas Samuel Kuhn. *The Structure of Scientific Revolutions*. 1962.
- Wele Gedara Chaminda Bandara; Nithin Gopalakrishnan Nair; Vishal M. Patel. Ddpm-cd: Denoising diffusion probabilistic models as feature extractors for change detection. *IEEE TRANSACTIONS ON GEOSCIENCE AND REMOTE SENSING*, 2022. doi:<https://arxiv.org/abs/2206.11892>.
- Saharia Chitwan, Ho Jonathan, Chan William, Salimans Tim, Fleet David J., and Norouzi Mohammad. Image super-resolution via iterative refinement. *ArXiv*, 2021. doi:<https://arxiv.org/abs/2104.07636>.
- Calvin Luo. Understanding diffusion models: A unified perspective. *Blog note*, 2022. doi:<https://calvinyluo.com/2022/08/26/diffusion-tutorial.html>.