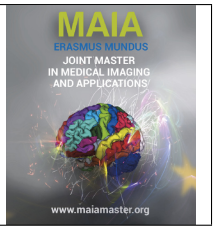
Medical Imaging and Applications

Master Thesis, June 2023




# *MAM-E*: Mammographic synthetic image generation with diffusion models

Ricardo Montoya del Ángel, Robert Martí Marly

*ViCOROB Lab, Universitat de Girona, Girona, Spain*

**Abstract**

Generative models have been used as an alternative data augmentation technique to counter the data scarcity problem faced in the medical imaging field. Diffusion models have gathered special attention due to their innovative generation approach, the high quality of the generated images and their relatively less complex training process compared with GANs. Still, the implementation of such models in the medical domain remains at early stages. In this work, we propose exploring the use of diffusion models for the generation of high quality full-field digital mammograms using state-of-the-art conditional diffusion pipelines. Additionally, we propose using stable diffusion models for the inpainting of synthetic lesions on healthy mammograms. We introduce *MAM-E*, a pipeline of generative models for high quality mammography synthesis controlled by a text prompt and capable of generating synthetic lesions on specific sections of the breast. Finally, we provide quantitative and qualitative assessment of the generated images and easy-to-use graphical user interfaces for mammography synthesis.

*Keywords:* generative models, mammography, stable diffusion

## 1. Introduction

Artificial intelligence has gained important attention in the last decade in essentially all aspects of human life. Thanks to the increasing data availability neural networks have played a key role on unveiling unsolved challenges, redefining AI research, and discovering new technological boundaries and applications.

A field that has attracted special recent attention is the generation of synthetic data, with the notable popularity of AI tools such as ChatGPT and DALL-E. Specifically in the imaging domain, generative models (GMs) started to gain notability in 2014 due to the impressive generative power of Generative Adversarial Networks (GANs). According to Yann LeCun, an important voice in the DL community, GANs were "..the most interesting idea in the last 10 years", as mentioned in his keynote at the Neural Information Processing Systems conference (NIPS) 2016 in Barcelona.

In the following years the appearance of new architectures and DL techniques, such as the rise of attention and transformers (Vaswani et al., 2017), further improved the generation capabilities and photorealism of the generated images in the natural imaging domain[1]. At the same time, researchers started to introduce these synthetic generation techniques into the medical imaging domain.

Contrary to natural images, medical images suffer from a data scarcity problem. Medical images are inherently more expensive than natural images due to their acquisition, processing and labeling procedure. Moreover, they are subject to more privacy and data protection concerns and, for some rare medical cases, images are difficult to find or suffer from underrepresentation, which leads to a subsequent data unbalance problem. All these issues dramatically reduce the volume of medical data available for the training of DL models, which limits the models performance and holds back the development of CAD systems, compared with non-medical imaging applications.

To counter this issue, GMs have been used to complement traditional data augmentation techniques and expand medical datasets, aiming to improve CAD mod-

[1] We refer as natural images to non-medical images, such as those included in large-scale datasets like ImageNet and LAION-B5. Other authors like Pinaya et al. (2022) have used this term.

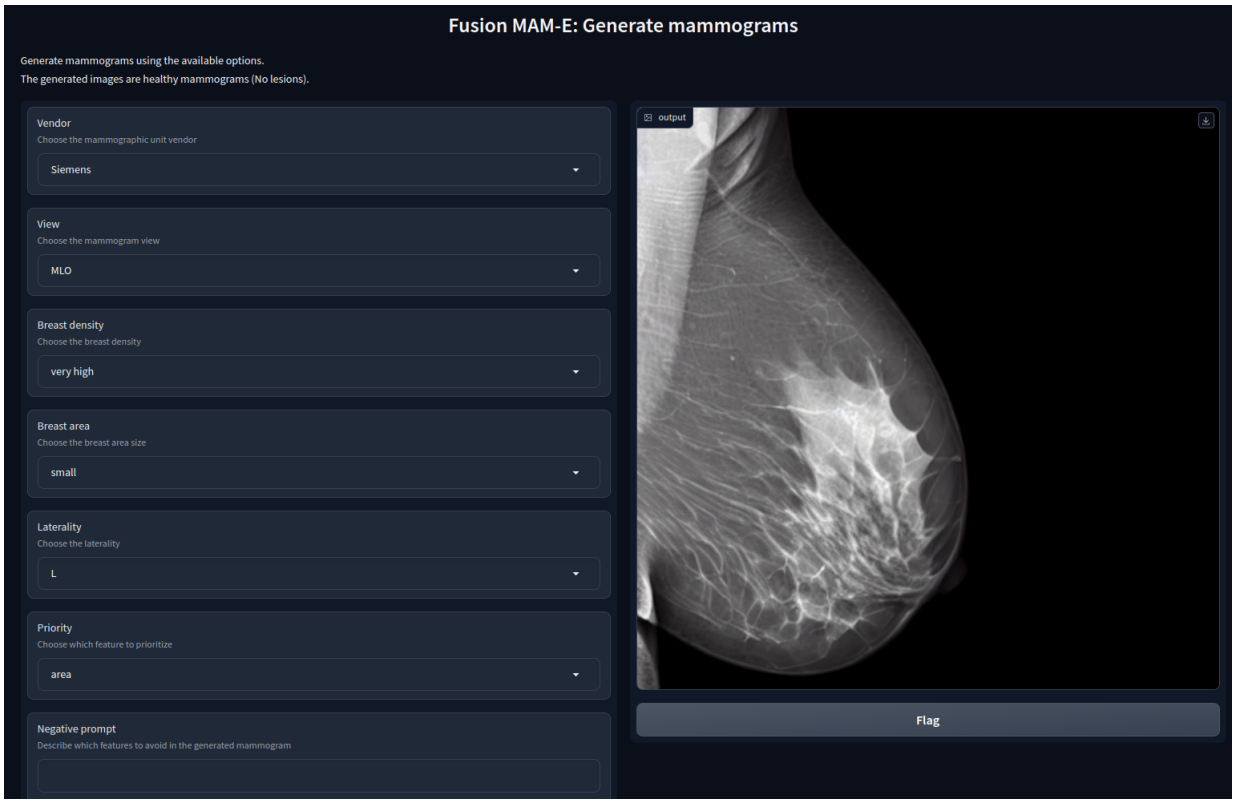



Figure 1: MAM-E: a synthetic mammogram generation tool.

els performance. Until a couple of years, GANs were the state-of-the-art (SOTA) for synthetic image generation tasks due to their high image quality and impressive photorealism. Nevertheless, some important limitations and drawbacks are inherent to these models. Due to its generator-discriminator architecture, GANs training is notoriously unstable and can be difficult to converge, as well as suffering from low diversity generation due to mode collapse issues (Kazerouni et al., 2023).

These issues make the use of GAN-like architectures challenging in some research domains. This is specially crucial for medical data, as medical diagnosis can highly depend on subtle changes in the organs appearance reflected in the images, changing the prediction of a CAD system (Müller-Franzes et al., 2022).

In 2021 diffusion models (DMs) captured the spotlight of the GMs community after the publication of OpenAI's belligerent article *Diffusion models beat GANs on Image Synthesis* by Dhariwal and Nichol (2021). Inspired by non-equilibrium thermodynamics, diffusion models rely on the idea that data distributions can be learned by iteratively destroying input information, adding certain noise, and then tasking a DL model to learn how to remove it in a denoising process, following a Markov chain.

Since the breakthrough of DMs, a great number of applications and research papers for natural images have been published to explore this new image generation principle. Results have shown promising improvements to the image generation task that continues to outperform GANs-like pipelines. Two main enhancements on traditional DM architectures are latent diffusion (LD), introducing the use of a latent space for higher image resolution, and stable diffusion models (SD) for additional input during training and inference for a more controlled generation process.

The medical image community has started to implement these improvements to generate high quality, high fidelity and realistic medical images, crucial characteristics for CAD systems development. Nevertheless, to the moment of publication of this work, the use of diffusion models in the medical imaging field continues at early stages. Even though a number of works have been published for the generation of several medical imaging modalities, such as brain MRI and chest X-ray, there is still no implementation of DM techniques for mammographic image synthesis.

### *1.1. Project description*

The objective of this master thesis project is to explore the use of diffusion models for the generation of high-resolution mammographic images and to develop a synthesis pipeline using SOTA conditional diffusion models. This pipeline was developed using stable diffusion, a diffusion model technique that uses both conditioning, to control the image generation, and a latent

space to allow high-resolution without requiring large computational resources. The generated images are *for presentation*, meaning that their appearance and pixel intensities are meant for radiologist inspection, with the limitations on resolution and pixel depth inherent to the current state of diffusion pipelines.

The pipeline can be separated in two main tasks: healthy mammogram generation and lesion inpainting. For the first task, the generation process is controlled (or guided) using text conditioning with the description of the image using common mammography characteristics such as view position, laterality, breast density and breast area. For the second task we use an stable diffusion inpainting model designed to generate synthetic lesions in desired regions of the a mammogram.

We introduce *MAM-E*, a pipeline of generative models for high quality mammographic image synthesize, capable of generating images based on a text prompt, and also capable of generating lesions on a specific section of the breast. We selected the name after DALL-E, OpenAI's famous image generation tool for natural images presented by Ramesh et al. (2021), as we aimed to create a graphical user interface (GUI) similar to DALL-E to allow user personalization of the generated image based on customizable settings.

## 2. State of the art

### *2.1. Diffusion on medical imaging*

Several relevant works have explored the implementation of diffusion models for synthetic medical images generation. Dorjsembe et al. (2022) proposed using the original pipeline of diffusion models on computer vision, introduced by Ho et al. (2020) called denoising diffusion probabilistic models (DDPM), for the generation of high-quality MRI of brain tumors, being the first attempt to investigate diffusion models for 3D medical images. This vanilla model was able to reproduce SOTA results, outperforming the baseline models based on 3D GANs.

A further improvement for synthetic brain MRI generation was presented by Pinaya et al. (2022), who used a Latent Diffusion model (LDM) to generate high-resolution 3D brain images. The use of a LDMs allowed increasing the image resolution from 64x64x64 to 160x224x160 without requiring more GPU memory usage or overall training time. More about latent diffusion will be explained in section 3.3.3. To assess the performance of the model and the quality of the synthetic images two main metrics were computed, the Fréchet Inception Distance (FID) for fidelity, and the MS-SSIM for diversity. In both cases DMs metrics ourperformed GANs results.

The first implementation of stable diffusion for medical images, the closest to our work, was introduced by (Chambon et al., 2022) who proposed a model for chest X-ray generation. Their model, named *RoentGen*, was able to create visually convincing, diverse chest X-rays, and the output could be controlled by using text prompts with radiology-specific language. Similar to the work of Pinaya et al., the FID and MS-SSIM metrics were computed although no comparison with GAN-based models was made. A key characteristic of this work is the use of pretrained weights coming from the *Hugging Face Hub*. Instead of training from scratch the network, their suggestion was to fine-tune specific parts of the network to adapt to this new domain. This DM fine-tuning approach is called *Dreambooth* and was first introduced by Ruiz et al. (2023).

The only work we found for lesion inpainting using DM was made for brain MRI by Rouzrokh et al. (2022) from Mayo Clinic. They developed a DDPM to execute several inpainting tasks, like generating lesions or healthy tissue, on slices of the 3D volumes in various sequences. Their model was capable of generate realistic tumoral lesions and tumor-free brain tissue, although the performance of the model was only assessed visually.

### *2.2. Generative models for mammography*

Despite the absence of DM-based model for mammography, other GMs, specially GANs, have been used for several tasks. Wu et al. (2018) tried to tackle the data scarcity and unbalance problem by using class-conditional GANs to synthetically augment mammogram datasets. They focused on training a model for contextual in-filling to synthetize lesions onto healthy screening mammograms. Then, they used this model to generate synthetic images to improve the AUC of a ResNet50 lesion classifier from 0.887 with traditional augmentation to 0.896 with GAN-generated data augmentation.

A full-field digital mammogram (FFDM) generation approach was performed by Korkinof et al. (2019) using GANs as well. Special attention was given to stabilize the GAN training by using stabilization methods and progressive training. The dataset consisted of around 450K images in both MLO and CC view, and they trained the network with a final image size of $1280x1024$. The final size used represented a training stability problem even after using the stabilization techniques. The generated images were only conditioned on the mammogram view. The training was done using 8 V100 GPUs of 16 GB each training for about 70 hours of total training. The trained model was able to generate highly realistic, high resolution synthetic images in appearance, although no quantitative assessment of the fidelity, diversity and radiologist opinion was conducted.

## 3. Material and methods

### *3.1. Datasets*

We decided to use two datasets for the training of the stable diffusion models (SDMs) to consider different patient populations and mammography unit vendors.

#### *3.1.1. OMI-DB*

We used a subset of the OPTIMAM Mammography Image Database (OMI-DB), consisting of around 140k images from several UK breast screening centers (Halling-Brown et al., 2021), various scanner manufacturers and with different image views. The dataset is composed of images with and without lesions (benign, malignant and interval-cancers), and expert annotations are included in the respective cases. Most of the images are available in both *raw* and in *for presentation* format, giving us a total number of $77,035$ images suitable for our generation purpose, distributed among $5,982$ patients. The images were available in DICOM files and its respective metadata was provided in a JSON file format.

Given that the dataset included images from several protocols, such as screening, biopsy and lesion magnification, an extensive image filtering was conducted. The criteria used for the selection of images were the following:

1. No special imaging protocol allowed. E.g. magnification and biopsy images. ($\sim 27k$ images removed)
2. No breast implants ($\sim 3K$ images removed)
3. Only CC and MLO view positions ($\sim 500$ images removed)
4. Only images coming from the *Hologic* mammography units. ($\sim 8K$ images removed)

From the criteria above, only criterion 1 is unavoidable, as we strictly require FFDM. Criteria 2-4 were set to keep a diverse yet uniform data distribution which would be easier for a GM to learn. The generation of minority cases as breast implant mammograms or uncommon mammographic views (such as exaggerated craniocaudal views) were considered out-of-the-scope of this project and let for future work. Because only Hologic images were used the dataset subset is called OMI-H.

The OMI-DB dataset includes metadata at the patient and image level in JSON file format. This information can be accessed using a Python library specifically designed for this dataset. The metadata includes basic DICOM information (laterality, view, pixel size, etc.) and clinical information such as patient status, lesion opinion, biopsy results, and more.

#### *3.1.2. VinDr-Mammo*

A second dataset called VinDr-Mammo composed of FFDM with breast-level assessment and extensive lesion annotation was also used. It consists of 5,000 mammography exams, each with 4 standard views (CC and MLO for both lateralities), coming from two primary hospitals from Vietnam, giving a total of 20,000 images in DICOM files (Nguyen et al., 2023). Metadata of each image consisting of both technical and clinical information were also available in a CSV file.

In this case, the only image filtering step performed was to keep images coming exclusively from SIEMENS scanners to avoid learning very different data distributions.

Table 1: Distribution of cases for both datasets.

| | OMI-H | VinDr | Combined |
|---|---|---|---|
| Healthy | 33,643 | 13,942 | 47,585 |
| With lesion | 6,908 | 1,533 | 8,441 |
| Total | 40,551 | 15,475 | 56,026 |

### *3.2. Data preparation and preprocessing*

Both datasets were subject to similar preparation and preprocessing steps. First, all images were saved to PNG format to ensure faster access and less memory usage. Secondly, given the DM architecture used (described in section 3.3.4) the images were saved in RGB format, repeating for each RGB channels the single-channel mammograms, resulting in a visually gray-level image. The original image intensities saved with *uint16* datatype were scaled to a [0, 255] range with a reduced *uint8* datatype.

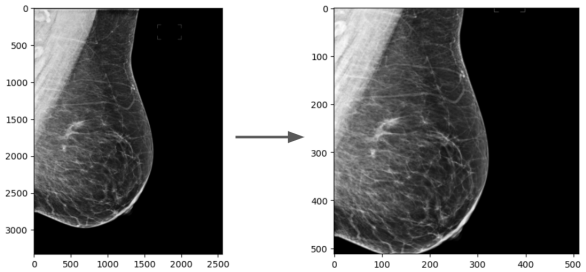
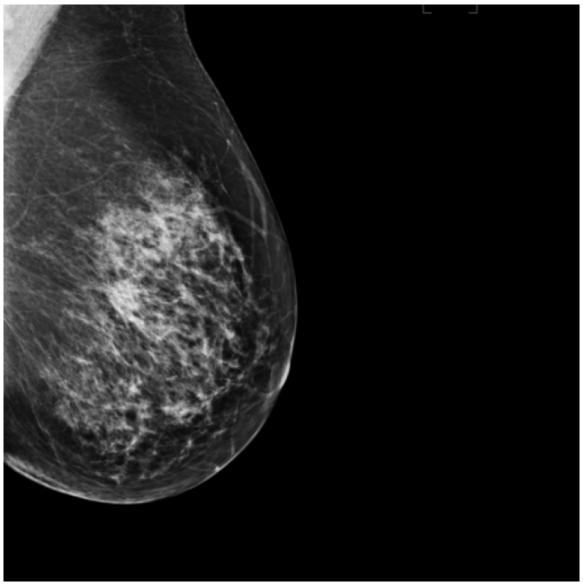
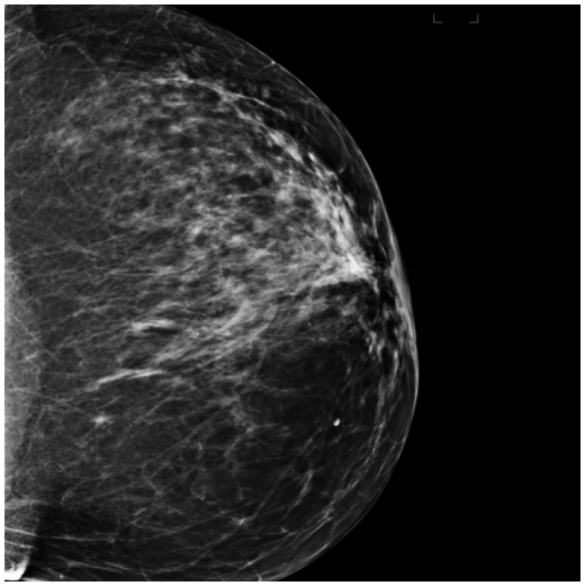



Figure 2: Resizing and cropping of an OMI-H mammogram. The same process was conducted for VinDr mammograms.

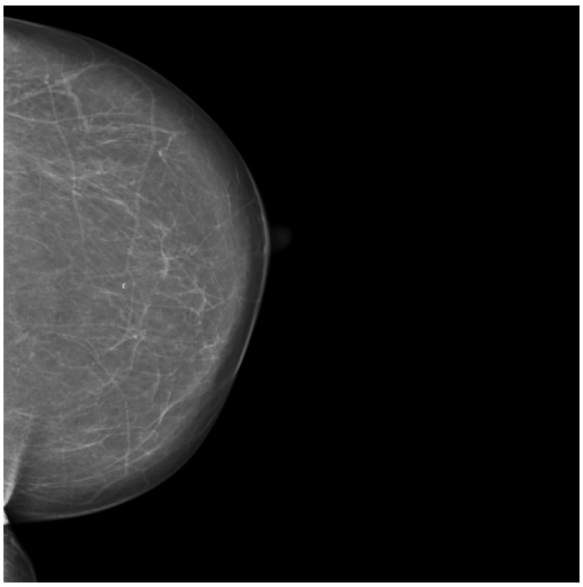
Additionally, in order to use the pretrained weights available for SD, the images were resized to a 512x512 square using bilineal interpolation and center cropping as shown in figure 2. Finally images with right laterality (R) were horizontally flipped so all images have the breast region in the same side, which can potentially facilitate the learning of the data distribution.



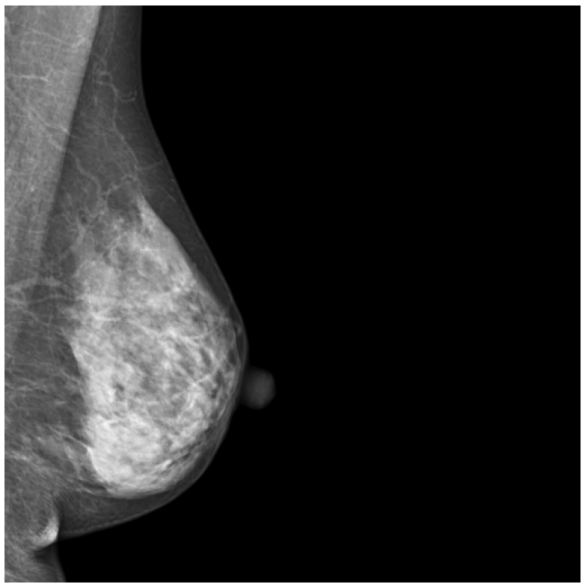
(a) "a mammogram in MLO view with small area"



(b) "a mammogram in CC view with big area"



(c) "a mammogram in CC view with very low density"



(d) "a mammogram in MLO view with very high density"

Figure 3: Examples of training mammograms (real) and their respective text prompts for OMI-H (a-b) and VinDr (c-d).

*3.2.1. Task one: healthy image generation*

For the first task the healthy images were saved in separated directories, one for each dataset. A text prompt with the description of the image was created and saved along with the image ID in a JSON file. In the case of the OMI-H dataset we created a prompt with the image view and breast area size information. Examples of prompts and their corresponding images are shown in figures 3a and 3b.

To compute the breast area size we first obtained a breast mask using the intensity information of the image and then applying a threshold to separate background and breast tissue. After getting the breast mask we computed the ratio of pixels in the mask compared with the total image. Finally, we define a criteria for three different breast area sizes which can be found in table 2a.

Table 2: Criteria for breast area size and breast density.

(a) Pixel ratio prompt assignment.

| Breast area size | |
|---|---|
| Small | ratio <0.4 |
| Medium | 0.4 <ratio <0.6 |
| Big | ratio >0.6 |

(b) BI-RADS breast density.

| Breast density | |
|---|---|
| Very low | Density A |
| Low | Density B |
| High | Density C |
| Very high | Density D |

For the VinDr dataset, we decided not to compute the breast area and, instead, included the breast density information for the prompt description. Breast density was available in BI-RADS scale so we needed to transform this information in a semantically easier text value. We classified the density BI-RADS following the criteria in table 2b. Examples of some images and their prompts can be found in figure 3c and 3d.

#### *3.2.2. Task 2: Lesion inpainting*

The second task requires mammograms with confirmed lesions only. Consequently we stored the selected mammograms in separated directories, one for each dataset. Then, using the bounding boxes coordinates available in the metadata, binary masks were generated. Naturally, due to the resizing and cropping preprocessing performed previously, the original coordinates required a proper redefinition using simple geometrical properties.

The binary mask has a pixel value of 255 inside of the bounding box and zero elsewhere. Figure 4 show an example of a mask overlapping a OMI-H mammogram. Because the SD architecture used for the inpainting task requires an input text prompt for the generation, a toy prompt with "a mammogram with a lesion" text was used for all training images.

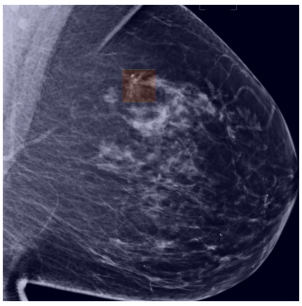


Figure 4: A OMI-H mammogram with a lesion overlapping with its corresponding bounding box mask.

### *3.3. Diffusion models*

The original diffusion model idea was presented by Sohl-Dickstein et al. (2015) and consisted on using a Markov chain[2] to gradually convert one known distribution (e.g. Gaussian distribution) into another (target distribution). Inspired by non-equilibrium statistical physics, the main idea is to systematically and iteratively destroy structure in a data distribution through a process called **forward diffusion**. Then, the **reverse diffusion process** is learned and used to restore structure in data, creating therefore a generative model that implicitly has learned the data distribution.

[2]Defined as a sequence of stochastic events whose time steps depend on the previous one.

The first practical implementation of the DM premise on images was developed by Ho et al. (2020) introducing *Denoising diffusion probabilistic models* (DDPM). In this framework, the data is destroyed by adding Gaussian noise to the image in an iterative fashion described by the Markov chain as shown in figure 5. The total number of diffusion timesteps $T$ is defined by the user but initial experiments were performed with $T = 1000$. To learn the reverse process a encoder-decoder-like neural network (such as a UNet) is used to carry on the denoising process.

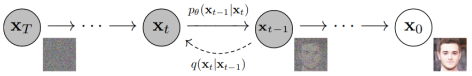



Figure 5: Markov chain of the forward and reverse diffusion process.

A vanilla DM has three main components:

1. Noise scheduler: to add noise in the forward process.
2. UNet: to denoise in the reverse process.
3. Timestep encoder: to encode the timestep $t$.

#### *3.3.1. Forward diffusion process*

Let $x_0$ be the original image and $x_t$ the noisy version of that image at time $t$. For the forward diffusion we can define the Markov chain process as

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

where $q$ is a probability distribution from which the noisy version of the image at time $t$ can be sampled, given $x_{t-1}$. The proposal of the DDPM framework is to define $q$ as a Gaussian (normal) distribution given by

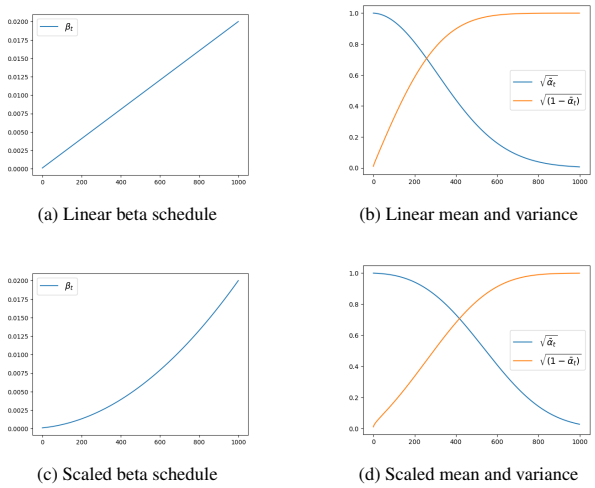
$$q(x_t|x_{t-1}) = N(x_t;\ \sqrt{1-\beta_t}x_{t-1}, \beta_t I), \tag{2}$$

where $x_t$ is the output of the distribution sampling, $\sqrt{1-\beta_t}x_{t-1}$ is the mean and $\beta$ the variance of the distribution. Therefore the sampling of the next noisy version of the image is essentially controlled by $\beta$, as its value affects both the mean and the variance of the sampling distribution. Selecting the manner in which $\beta$ changes through time is called beta scheduling and is control by the **noise scheduler**. In figures 6a and 6c two examples of beta scheduling are shown.

Thanks to the additive properties of Gaussian distributions, we can obtain a noisy image at any timestep $t$ directly by rewriting the sampling distribution 2 as

$$q(x_t|x_0) = N(x_t;\ \sqrt{\tilde{\alpha}_t}x_0, (1-\tilde{\alpha})I), \tag{3}$$

with $\tilde{\alpha}_t = \Pi_{s=1}^{t} a_s$ and $\alpha_t = 1 - \beta_t$, where $\alpha$ can be interpreted as measure of much information from the previous image is being kept during the diffusion process. The importance of $\tilde{\alpha}_t$, and therefore of $\beta_t$, can be understand by looking at figure 6. For $t$ values close to 0



(a) Linear beta schedule

(b) Linear mean and variance

(c) Scaled beta schedule

(d) Scaled mean and variance

Figure 6: Linear and scaled beta schedulers (left) and their effects on the mean (blue) and variance (orange) of the noise sampling distributions (right).

the distribution from which we sample have $\mu \approx 1$ and $\sigma \approx 0$, meaning that the sample images are every similar to the original image. On the other hand, for large $t$ values where $\mu \approx 0$ and $\sigma \approx 1$ the distribution is close to a standard normal distribution (SND) and the sampled image will be essentially pure Gaussian noise.

Finally, to be able to define the training goal in the reverse diffusion process, we express the sampling from the probability distribution in equation 3 using the *reparameterization trick* (Kingma and Welling, 2022). The reparameterization trick allows us to write the generation of a sample $X$ from a normal distribution $N(\mu, \sigma)$ as $X = \mu + \sigma Z$, where $Z \sim N(0, 1)$, i.d. $Z$ was sampled from a SND. With this, the forward diffusion sampling process can be expressed by

$$x_t = \sqrt{\tilde{\alpha}_t} x_0 + \sqrt{1 - \tilde{\alpha}_t}\epsilon, \tag{4}$$

where $\epsilon \sim N(0, 1)$. The stochastic variable epsilon ($\epsilon$) in equation 4 is crucial to understand the reverse diffusion process as it is basically the prediction target of the UNet.

### 3.3.2. *Reverse diffusion process*

The reconstruction of the data destroyed by noise can be done using a UNet that has learned to denoise the images. Formally, the reverse process is also a Markov chain that can be defined in a similar way as

$$p_\theta(x_{0:T}) = p(x_T)\Pi_{t=1}^{T} p_\theta(x_{t-1}|x_t), \tag{5}$$

where $p_\theta$ is the learned probability distribution from which the denoised images are sampled at each timestep $t$. $\theta$ indicates that the distribution is parameterized as it was learned by the UNet. This also explains why the term $p(x_T)$ has no subscript $\theta$ as it is the starting point of the reverse process, i.e. pure Gaussian noise.

Assuming that $p$ can also be modeled as a normal distribution, it can be expressed as

$$p_\theta(x_{t\_1}|x_t) = N(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)), \tag{6}$$

where $\mu_\theta$ and $\Sigma_\theta$ are the learnable mean and variance of the reverse sampling distribution. To reduce the training complexity, and because it showed to give similar results, $\Sigma_\theta = \beta I$, therefore only $\mu_\theta$ has to be learned. Sadly, due to limitations of space in this report, the complete formulation of the optimization of the usual variational bound on negative log likelihood cannot be fully described. Key considerations of this formulation are given instead.

The first consideration is that $\mu_\theta$ can be computed as

$$\mu_\theta(x_t, t) = \frac{1}{\sqrt{\alpha_T}}(x_t - \frac{\beta}{\sqrt{1 - \tilde{\alpha}_t}}\epsilon_\theta(x_t, t)), \tag{7}$$

where the key is to notice that we only need to predict $\epsilon_\theta$ to predict $\mu_\theta$.

The second consideration is that the term we need to optimize, and which consequently defines the loss function of our UNet, is

$$L = \|\epsilon - \epsilon_\theta\|^2, \tag{8}$$

where epsilon ($\epsilon$) is the same Gaussian noise we defined in equation 4, sampled from a Gaussian distribution $\epsilon \sim N(0, 1)$, and $\epsilon_\theta$ is the output of the UNet. In other words, the UNet objective is to implicitly learn the data distribution by predicting the scaled Gaussian noise $\epsilon$ added to the images at timestep $t$.

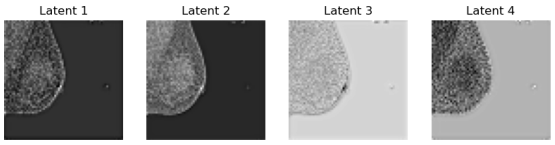
Finally, to include the timestep as an additional input to the Unet, a timestep encoder is used to embed this information and use it during training. More information will be given in section 3.3.4.

### 3.3.3. *Latent diffusion*

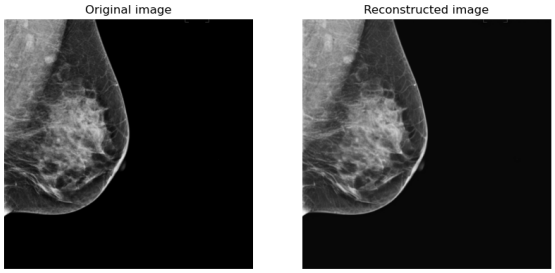
Image size is one of the main constrains when training generative models. Medical images usually require high resolution, specially in the case of mammograms where the sizes go up to 3 or 4 thousand pixels per image side. Training a DM for such sizes would require large computational resources and extensive training time.

Latent diffusion tries to solve this issue by using encoders to *compress* images from their original sizes in the image space into a smaller representation in the latent space. The motivation behind this is that images usually contain redundant information and an encoder can produce a smaller representation that can later be reconstructed back using a decoder.

An example of this is shown in figure 7, where an original image of 512x512 pixels is compressed to 4 latent representations of 64x64 using a Variational Autoencoder (VAE), reducing 16 times its original shape (Kingma and Welling, 2022).




(a) Latent representations.




(b) The original image and its reconstruction.

Figure 7: Example of the latent space representation of an image and its reconstruction.

Consequently, as introduced by Rombach et al. (2022), in latent diffusion the diffusion processed (described in the previous section) is performed on the latent representations rather than the original images. This allows using diffusion pipelines with lower memory usage, fewer layers in the UNet, and faster training and generation.

There exist different types of encoders that can be used and the selection criteria reside mainly on the type of images and their task. Chambon et al. (2022) found that a pretrained VAE on natural images can have good perform in medical images as well. For this reason we decided to use a VAE for our work, obtaining visually successful encoding for mammograms as shown in figure 7b. More information on the model architecture can be found in section 3.4.

#### *3.3.4. Stable diffusion*

Pure latent diffusion does not include conditioning at training or inference time, and synthetic images are generated from the learned distribution, depending on the starting Gaussian noise. Stable diffusion is an improvement to Rombach et al. (2022) work, in which text conditioning is added to the model for additional control on the generation process.

In stable diffusion the text conditioning is a prompt with the description of the image. To create a numeric representation of the prompt we use a pretrained transformer called CLIP (Radford et al., 2021). CLIP, which stands for Contrastive Language-Image Pre-training, maps both text and images into the same representational space, allowing comparison and similarity quantification between them (Frans et al., 2021). In other words, CLIP allows us to compare images and text.

CLIP first uses a subword-based tokenizer to convert any prompt text to a fixed 77 tokens length. Then, the CLIP encoder sends each token into a 768-dimensional vector, which lives in the image-text CLIP space. The CLIP embedded text is then used in the attention layers of the UNet through a cross-attention mechanism. More details are given in section 3.4.

#### *3.3.5. Fine-tuning SD: DreamBooth*

In 2022 Stability AI and LAION made the pretrained weights of Rombach et al. (2022) model publicly available, which allowed the GM community to train domain-specific fine-tuned SD models. Nevertheless, fine-tuning a large text-to-image model and teaching it new concepts can be challenging and one can face several difficulties such as catastrophic forgetting[3], overfitting and low image generation diversity.

Ruiz et al. (2023) presented an approach for fine-tuning SD called *DreamBooth*. They proposed using only a few images of the new subject with its respective text prompt, to train the model using a small learning rate. Additionally, if the subject semantically exists in the model domain, prior generation images can be included for the training. This allows the binding of the new subject to a new unique identifier in the text embedding space, as well as a learned representation in the pretrained data distribution.

This fine-tuning technique has been tried for chest X-ray by Chambon et al. (2022) and showed promising results on adapting the SD domain into their images to generate high-fidelity and diverse images thanks to the control given by the text prompt.

### *3.4. Task 1: Normal mammogram synthesis*

We propose putting together all the pieces presented above and adapting the Dreambooth fine-tuning technique for mammograpic images generation, using the pretrained *stable-diffusion-v1-5* model as baseline, publicly available in the *Hugging Face* model hub repository.

For each dataset we decided to train a separate model using only healthy images, as each dataset contains independent semantic information in the prompt and because the intensity ranges and image details differ between populations. This means that we trained separate models for *Siemens* and *Hologic* mammograms. Additionally, we decided to train a combined model with images of both vendors, adding in the prompt text the vendor's name.

Our SD pipeline has three independent models that could potentially be trained at the same time. Given the good performance of the VAE encoder on mammograms, we decided to keep it frozen an train only the CLIP text encoder and the UNet weights. The three models are summarized as follows:

[3]This means the model forgetting previous information and concepts.

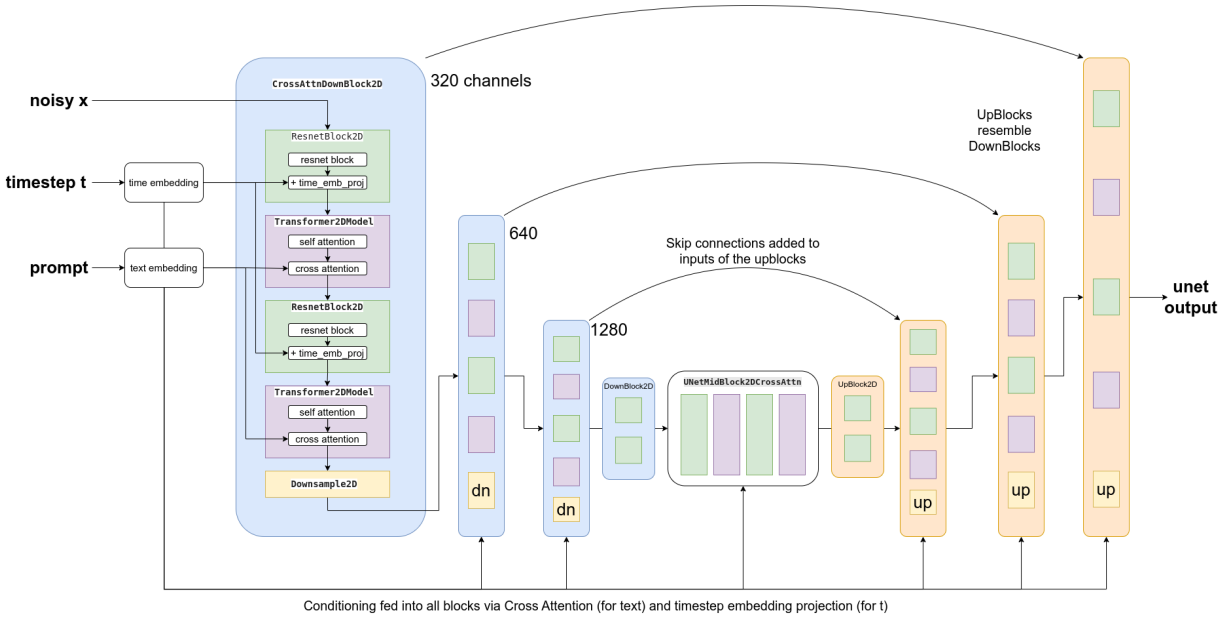



Figure 8: Training pipeline for full mammogram generation. UNet architecture from the original stable diffusion paper (Rombach et al., 2022).

1. VAE as image encoder (frozen): Pretrained with LAION-b5
2. CLIP text encoder: *CLIP ViT large* using a ViT-L/14 Transformer architecture from Open AI.
3. UNet: Pretrained with LAION-b5.

The pretrained VAE model inputs RGB 512x512 and outputs latent representations of 64x64x4, just as shown in figure 7.

### *3.4.1. Training*

The UNet architecture is the original SD UNet proposed by Rombach et al. (2022) and its presented in figure 8. The network has 4 2D down- and upsampling blocks. Except for the last downsampling block (and its corresponding upsampling block) all blocks are composed of two ResNet blocks and two Transformer blocks, one after the other. The timestep embedding is added to the ResNet blocks whereas the text embedding is added through cross attention into the Transformer blocks. For the last downblock (and first upblock) only the timestep information is fed.

One training step can be summarized as follows:

1. Sample a batch of images $x_0 \sim q(x_0)$
2. Encode $x_0$ into the latent space
3. Sample a random timestep from a uniform distribution $t \sim U(1, ..., T)$
4. Sample random Gaussian noise from a normal distribution $\epsilon \sim N(0, I)$
5. Create $x_t$ by adding noise to the batch images $x_0$ using the noise $\epsilon$ and timestep $t$.
6. Take a optimization step in the direction of the gradient $\nabla_\theta \|\epsilon - \epsilon_\theta(x_t, t)\|$
7. Repeat

We notice that, contrary to what is popularly believed, the DM training process does not consist on denoising the same image in a sequential order. This confusion comes from the way the diffusion process is presented as a Markov chain in figure 5. Instead, the training encompasses three stochastic processes by randomly sampling the main components of the diffusion process: the original image $x_0$, the Gaussian noise $\epsilon$ and the timestep $t$. By doing so we avoid the overfitting of the network on the sequential way the images are given and focuses on the denosing process per se.

The CLIP text encoder and the UNet parameters are updated simultaneously at each training step using an AdamW optimizer, with the MSE as loss function described in equation 8.

The main training hyperparameters (HP) are the following:

- Batch size: We ranged from 8, 16, 32, 64, 128 and 256.
- Training steps: Experiments ranged from 1k up to 16k.
- Learning rate: We explore three main values $1e^{-6}$, $1e^{-5}$, $1e^{-4}$.

Other HP that were not changed include: constant lr schedule, Adam weight decay and epsilon, gradient clipping and dropping the last incomplete batch per epoch. All detailed HP information can be found in the configuration file of each experiment in the GitHub repository.

We generated 4 sample images every 100 or 200 training steps to track the performance of the models, as well as the training loss. This was loaded to the cloud using *Weights and Biases* (WandB) logging tool. Additionally all models were uploaded to the authors personal Hugging Face repository and are publicly available.

### *3.4.2. Inference: image generation*

With the UNet prediction we can denoise pure Gaussian noise and generate new mammograms. The procedure is as follow:

1. Sample random Gaussian noise $\epsilon \sim N(0, I)$
2. for $t = T, ...1$ do:
3. $z \sim N(0, I)$ if $t > 1$ else $z = 0$
4. $x_{t-1} = \frac{1}{\sqrt{\alpha_T}}(x_t - \frac{\beta}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t, t)) + \sigma_t z$
5. end for
6. Decode image using VAE

First, random Gaussian noise is sampled as starting point. Then the denoising process is repeated for $T$ steps. The loop consists on using the predicted noise $\epsilon_\theta$ to compute the distribution mean using equation 7. By adding $\sigma_t z$ to this mean term we are essentially sampling from the learned data distribution of the reverse diffusion process. After the denoising process is finished the image is send back to the image space using the VAE decoder.

The inference process has two main HP to consider: number of timsteps $T$ and the guidance scale. First, the number of timesteps $T$ will depend on the type of sampling method that we use for denosing. The traditional DDPM sampling requires around 100 steps to generate good quality images, which is time consuming and represent a bottleneck in the image generation. The best alternative we found is to use the DPM-solver proposed by Lu et al. (2022), which allows fast diffusion sampling with only 20 steps for good quality image generation. In the result section we show how the change of $T$ affects the image quality.

The second HP is called the guidance scale. Even though the SD architecture uses cross attention in several parts of the network, so the generation process focuses on the text prompt, in reality this is still not enough and the model tends to ignore the text prompt at inference time. To solve this issue Ho and Salimans (2022) proposed a technique called classifier-free guidance.

In essence, classifier-free guidance consists on generating two noise predictions $\epsilon$ at each step, one using the prompt ($\epsilon_{text}$) and one without it ($\epsilon_{free}$). Then, the difference between the prompt-generated noise and the free-generated noise is computed. This difference can be considered as a vector in the image distribution space, which points in the direction of the image with text. As such, we can scale this vector and sum it to the free-generated noise to force it to go more in the direction of the prompt text. This geometrical trick is illustrated in figure 9.

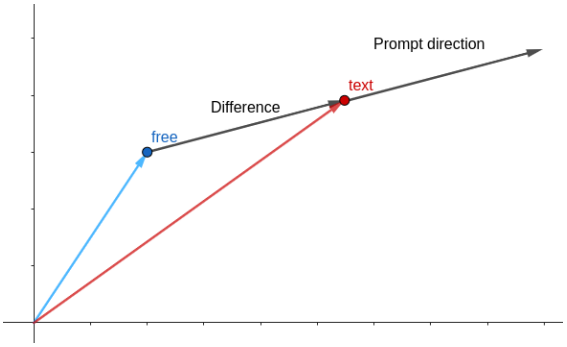



Figure 9: Classifier-free guidance geometrical interpretation. As the guidance scale increases, the image is pushed further in the prompt direction.

Formally, the scaling factor is called guidance scale and the formulation can be summarized as follows:

$$\epsilon_\theta = \epsilon_{free} + guidance * (\epsilon_{text} - \epsilon_{free}). \quad (9)$$

### *3.5. Task 2: mammographic lesion inpainting*

The SD pipeline described for task 1 can be modified in some key aspects to be able to perform the inpainting task. We propose using the modified DreamBooth fine-tuning pipeline to inpaint lesion in a designated region of the breast. To the knowledge of the authors, this is the first work to use SD fine-tuning for lesion inpainting in medical images.

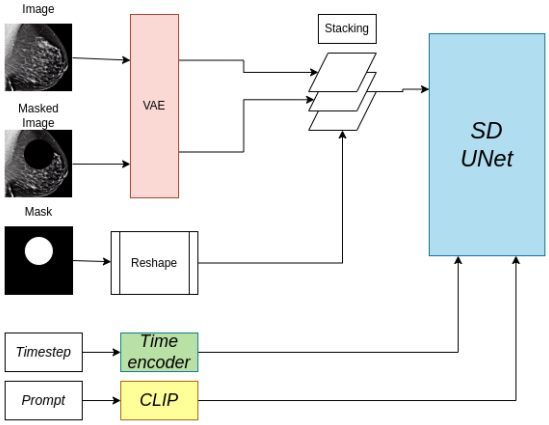



Figure 10: Inpainting training pipeline. The same UNet as in the SD pipeline in figure 8 is used.

At the dataloader level, for each batch two new elements are added per example: the mask and a masked version of the original image. The masked version means that the pixel values inside of the bounding box are set to zero.

At training time, first both the image and the masked image are encoded using the latent space. Also the mask

must be reshaped to the latent representation size. The rest of the diffusion process remains the same, only one crucial difference is made: instead of feeding the latent representation only to the UNet, the latent representation, the mask, and the masked latent representation are stacked into one tensor. This new input is then fed to the UNet, as well as the encoded timestep and prompt text as in a traditional SD. This process is described in figure 10.

This small change in the training process allows the network to pay attention only to the pixels inside the mask, as the pixel outside of it are always provided. The rest of the pipeline follows the same principles as the ones we described for task one.

### *3.6. Complete MAM-E pipeline*

The models used in task 1 and task 2 can be put together in a sequential order so that a full synthetic mammogram with lesion can be generated. Figure 11

### *3.7. Resources management*

Having three large models loaded at the same time, and enabling the gradient tracking for two of them for training, can represent a dramatic increase of GPU and processor resources. Thankfully there exist a handful of techniques and frameworks to reduce this demand and fit the training in a GPU memory of circa 20 GB with an efficient batch size of 256.

First, we used mixed precision using the *fp16* arithmetic, and the revision model (model version) specifically for that precision. When training in the (Ampere) A30 or A40 GPU we activate the bf16 precision, with no improvement in the time or apparent quality of the training.

We also used lighter version of the AdamW optimizer, the 8-bit AdamW optimizer by *Bitsnadbytes*. Additionally, because our three models use attention layers, we made us of the *Xformers* efficient memory usage for transformers which speeds the training time and decreases the GPU usage.

To achieve the 256 batch size in one single GPU we used gradient accumulation, a technique that consists on computing the gradient for a mini-batch without updating the model variables, for a set number of times, summing the gradients. By doing so, the general batch size is accumulated an essentially the batch size increases. In our case, using a mini-batch size of 16, and 16 gradient accumulation steps the accumulated batch size is 256. This technique, although clever comes with an increase in training time.

Gradient checkpointing is another technique to use the CPU processors power to help release some GPU memory at the expenses of a slower training. Gradient checkpointing saves strategically selected activations throughout the computational graph of the model tensors so only a fraction of the activations need to be re-computed for the gradients.

Finally one can simply set the optimizer gradients to None instead of zero after the weights update have been completed. This will in general have lower memory footprint, and can modestly improve performance.

Most of these techniques can be implemented directly using *Hugging Face Accelerate* library and framework for distributed training and resources management.

### *3.8. Assessment*

The assessment of generative models depends on the application of the synthetic images and it may not be straightforward as in other DL models. While it is possible to observe the changes in the loss values during training, the loss curve rapidly converges to a specific region and no further difference is notice. It is specially difficult to see any substantial loss differences when performing DreamBooth fine-tuning.

Regardless of the apparent plateaued loss function, the semantics learned by the model are continuously changing during training. One possible way to keep track of the model training performance is to log examples of synthetic images every several steps to see this semantic changes.

At inference time there exists other types of assessment and they can be categorized as follows:

- Qualitative assessment: Focusing on the visual appearance of the mammograms.
- Quantitative assessment: Computing metrics to attest the diversity and fidelity of the generated images, as well as generation time.
- Quantitative CAD assessment: Exploring the potential benefits of synthetic images on CAD systems performance.

#### *3.8.1. Qualitative assessment*

We performed two types of visual assessment. First, an overall simple visualization of the images to see any clear inconsistency (noise remnants, anatomical irregularities).

Then, we assessed the quality of the images by asking a radiologist with 30 years of experience to rate 53 mammograms, with a 50/50 real-synthetic ratio, in a scale from 0 to 4, using the following criteria:

- 0: Definitely real
- 1: Probably real
- 2: Not sure
- 3: Probably synthetic
- 4: Definitely synthetic.

These results were then converted to probabilities to be able to obtain a ROC curve and its respective area. The main objective is to attest the radiologist's ability to differentiate synthetic mammograms apart from real ones.

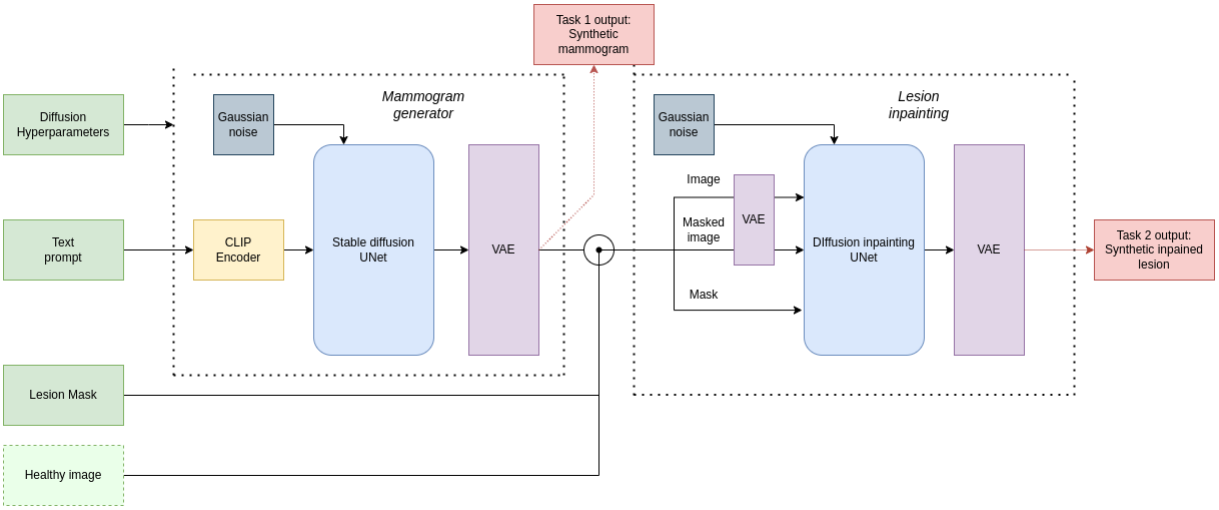



Figure 11: The complete MAM-E pipeline combining task 1 and task 2 pipelines. In dark green, the inputs needed for a full synthetic mammogram generation with lesion. In light green the optional input for lesion inpaining on real images, overriding task 1. In red, the outputs of each task.

#### *3.8.2. Quantitative assessment*

Generative models are expected to have two main characteristics: generation diversity and fidelity to the original dataset. There exists metrics to quantitatively assess them.

Generation diversity can be computed using the pairwise Multi-scale structural similarity index metric (MS-SSIM). If a pair of synthetic images are sampled, a low MS-SSIM value would mean that the compared images are not structurally similar and, therefore, implies diversity.

Fidelity can be calculated using the Fréchet Inception Distance (FID), a metric first proposed for GAN-generated image quality assessment by Heusel et al. (2018). FID captures the similarity of generated images to real ones by comparing the statistics of a collection of synthetic images and a collection of real images.

Formally, the activation vector from the last pooling layer of a ImageNet-pretrained Inception V3 is computed for a set of real and synthetic images. This 2048-vector is called the feature vector and contains computer-vision-specific information of the images. The FID consists, then, on calculating the Fréchet distance between the Multivariate Gaussian distribution of both population, synthetic and real. This is done by sampling $N$ examples from each distribution. $N$ is recommended to be $10,000$ for the best approximation, although some works suggest using a lower $N$ number can also be representative of the distribution.

Notwithstanding, the use of the FID metric is controversial and even discoraged. Chong and Forsyth (2020) found that the FID is biased towards the generative model and should not be used. Moreover, given that no other work has explored the FID for FFDM generation we decided not to compute the FID and assess the image fidelity based on the radiological expert visual assessment of the previous section.

Finally, generation time has to be assessed as the denoising time of DM is one of its main drawbacks. Generation time is closely tied to other inference HP like the guidance scale or the prompt length and order.

#### *3.8.3. Quantitative CAD assessment*

The performance of the generative models and the utility of the images for training CAD models can be assessed using CAD pipelines and referring to the effect of adding synthetic images during training.

We decided to collaborate with a master thesis defendant, Sam-Millan (2023) from the ViCOROB lab and whose works are included in the proceedings of this year.

Sam-Millan et al. worked on explainability AI (EAI) for patch classification and full-field mammogram lesion classification. Specifically for FF mammogram classification problem, the EAI system explores which regions of the image are more relevant for the classification task. A heatmap of these regions importance is generated for several EAI methods. It is expected that the lesion region captures the main attention.

We asked the authors to create heatmaps of a healthy real mammogram with a synthetic inpainted lesion to assess if the classifier is focusing on the synthetic lesion region.

For more details on any of these methods refer to the corresponding report.

## 4. Results

### *4.1. Training unconditional model*

The first experiments were conducted on unconditional diffusion models, meaning that no text prompt guidance was used as input. This first steps were vital

for the building of the conditional models later on. Visual assessment of the generated images during training time is presented in this section.

#### *4.1.1. Image space: vanilla DM*

The first trial consisted on generating 64x64 mammograms to observe the evolution and behavior of DM while trained from scratch using mammograms.

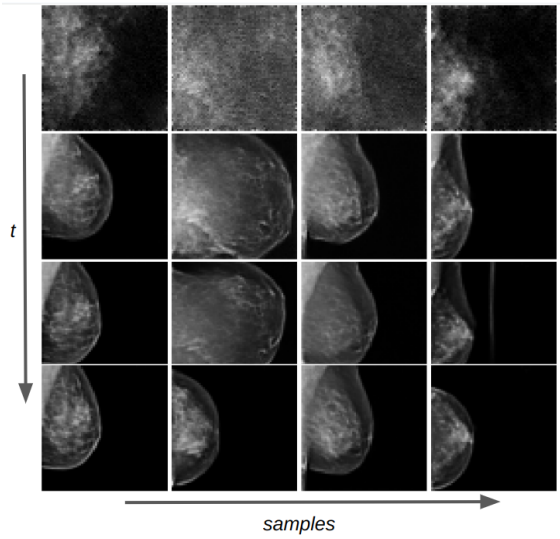



Figure 12: Training evolution of the vanilla image space diffusion model at 1k, 2k, 3k and 4k timesteps. This corresponds to epoch 1, 16, 3 and 50.

The corresponding loss and log-loss function of this vanilla DM are presented in figure 13.

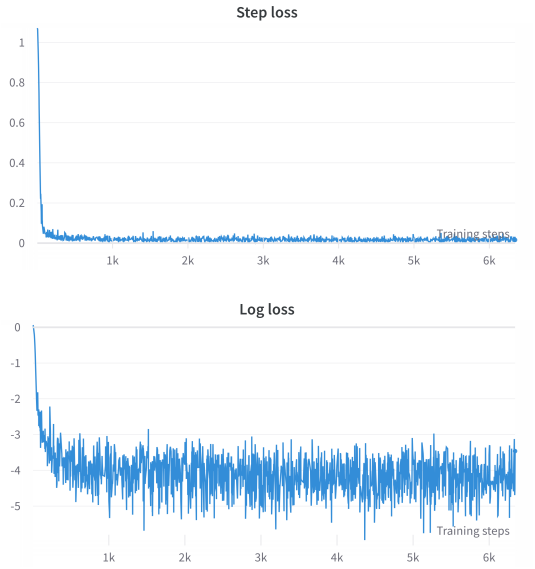



Figure 13: On the top: Loss of the vanilla DM training for 6k steps. On the bottom: the corresponding log loss.

This vanilla DM training helped us to make two initial main considerations. First, it allowed us to assess the training of a diffusion model from scratch and how the denoising process can effectively generate meaningful images after the first 2k training steps, as figure 12 shows.

Secondly, we observe that the loss function rapidly reaches a plateau, which show how fast the loss objective can be minimized in diffusion. Nevertheless, as seen in the log-loss, the function fluctuates in a specific range. This is a common behavior of DM losses, which tend to reach a stability region where the loss varies and then slowly starts to decrease as the denoising process is learned.

#### *4.1.2. Latent space diffusion*

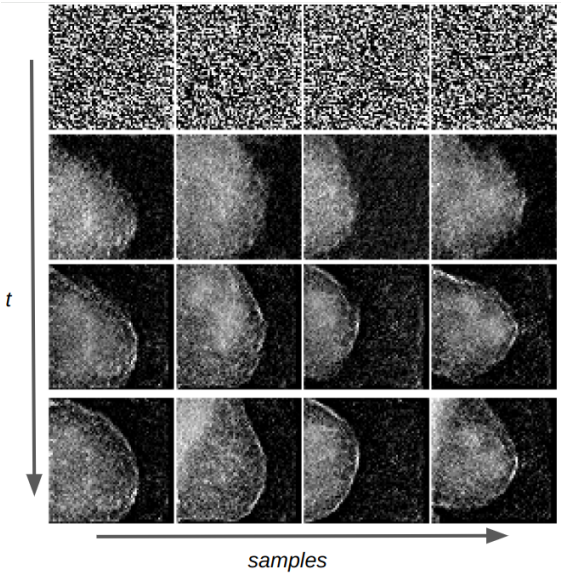



Figure 14: Training evolution of only one latent representation at epoch 1, 16, 36 and 50.

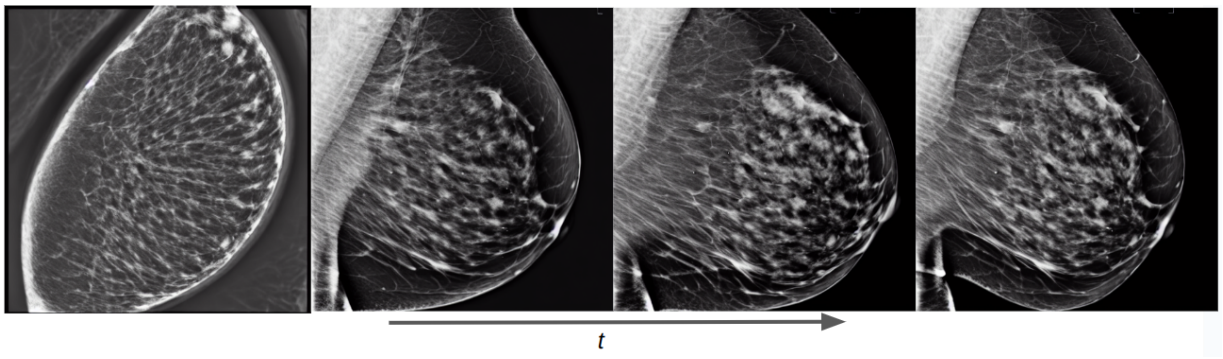
The second group of experiments consisted on sending the images to the latent space and implement the diffusion process on the latent representations. Figure 14 shows the latent denoising process for one channel (there are 4) of the latent representation.

It can be seen that the denoising process is more difficult and slower in the latent space. After 50 epochs the latent image still presents remnants of the original Gaussian noise. In contrast, in the image space the image present almost no signs of Gaussian noise after the same number of training epochs are completed.

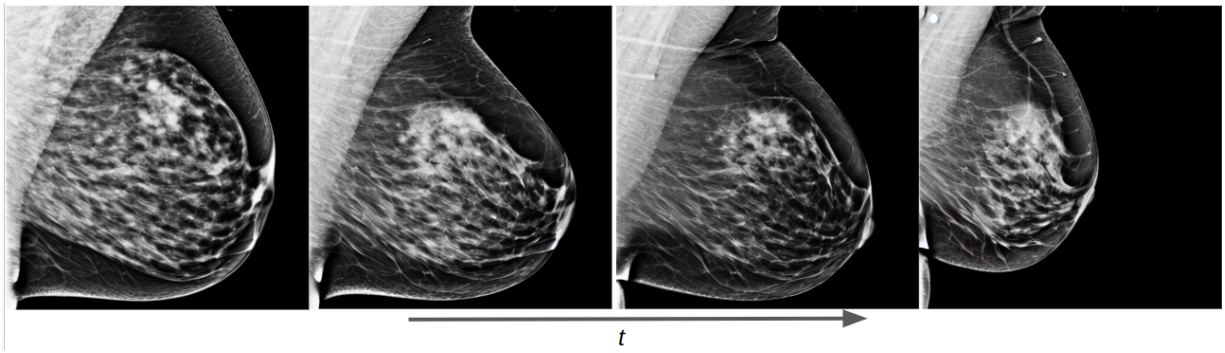
This behavior is expected as the prediction objective of the UNet in the image space consists of predicting a 64x64 one-channel $\epsilon$ matrix. On the other hand, the prediction objective in the latent spaces is a 54x54 four-channel $\epsilon$ matrix, 4 times bigger representation, which represents a more challenging objective to train.

#### *4.1.3. Unconditional pretrained models*

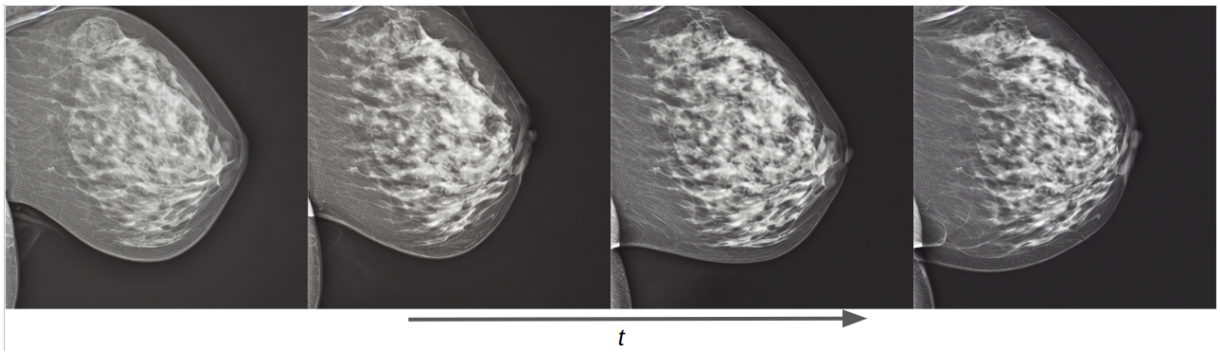
We solved the issue of denoising the mammograms latent representation using a pretrained unconditional



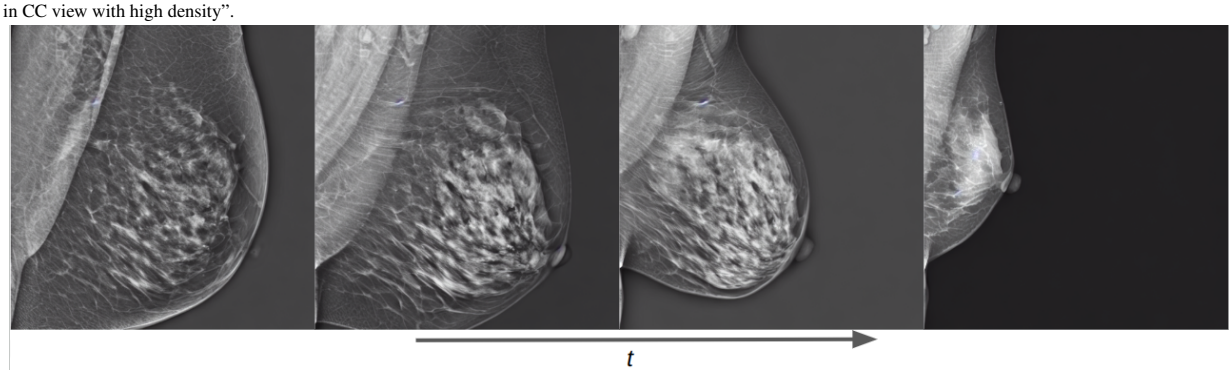

(a) Training evolution of the diffusion process on an unconditional pretrained model at epoch 1, 3, 6 and 10.




(b) Training evolution of the diffusion process on a conditional pretrained model trained with Hologic images at epoch 1, 3, 6 and 10. The prompt is: "a mammogram in MLO view with small area".




(c) Training evolution of the diffusion process on a conditional pretrained model trained with Siemens images at epoch 1, 3, 6 and 10. The prompt is: "a mammogram in CC view with high density".




(d) Training evolution of the diffusion process on a conditional pretrained model trained with both Siemens and Hologic images at epoch 1, 3, 7 and 40. The prompt is: "a siemens mammogram in MLO view with high density and small area".

Figure 15: Training evolution for several diffusion processes.

latent diffusion model and fine-tuning it. Figure 15a shows the evolution of the diffusion process as the training steps progress. It can be seen that from the first epoch the generated image has essentially no signs of residual Gaussian noise, although the synthetic image does not resemble a mammogram. This implies that

the pretrained model has already learned how to denoise images and that the new task is to learn a new concept (a mammogram) and find its representation in the data distribution of the model.

We can also notice that in only 3 epochs the model has already learned the fundamental characteristics of a mammogram and can generate realistic images. In the following epochs the model focuses on improving smaller details on the image, like the edges of the breast and the details of the breast parenchyma.

### *4.2. Fine-tuning conditional models: DreamBooth*

The training of the conditional model using prompt text can be shown in figure 15b for the Hologic dataset and 15c for the Siemens dataset.

First, we observe that the conditional model, besides learning the anatomical structure and form of a mammogram, pushes the generated image in the direction of the text prompt semantics as the training process increases. In the case of the Hologic training, in figure 15b we can see that the mammogram reduces its shape in accordance to the area described in the prompt text.

In the case of the Siemens example in figure 15c, the image view starts in a tiled position similar to MLO but it is slowly corrected to match with the prompt description, that is a CC view. Similarly, the apparent breast density is kept high, in accordance to the input prompt.

Therefore we can acknowledge that a conditional model, thanks to the combined training of the CLIP text encoder and the UNet, learns to modify the generated image to better match the generated pair image-prompt similarly to how they are paired in the training set.

### *4.3. Fusion MAM-E: combined datasets*

The combination of both datasets allowed us to train a model we called **Fusion MAM-E**. Besides allowing us to also select the vendor type of the generated mammogram, this model allowed us to extrapolate the characteristics of one dataset to the other. This means that, e.g. the breast density of the Hologic mammograms could be controlled, even though this information was not available in the Hologic dataset.

Figure 15d show some training samples generated at different epochs. During the first epoch, the model generates a synthetic images with in the correct view but with large breast area and low breast density. We observe how, as the training process advances, the breast area shrinks and the breast density augments, in accordance with the prompt. Also, the image intensities and overall texture starts to be more similar to a Siemens mammogram.

It is also valuable to remark the gray background present during the first training epochs of the fusion model. The complete removal of this feature, shifting it into a black background, required 40 training epochs, a considerable difference with the other conditional models. This is expected as the fusion model incorporates more semantic concepts to the CLIP text encoder that have to be learned, as well as a larger image dataset.

### *4.4. Quantitative assessment*

All synthetic examples above, even though logged during training time, naturally involves a inference pipeline. As explained in section 3.4.2 there are two main HP that have to be tuned during inference.

First, the denosing steps $T$ must be set. In our case, because we used the DPM-solver of Lu et al. (2022) we only needed, in general, 24 timesteps for denoising. This usually means an average time of 2 seconds for the denoising of one sample. In some cases, due to the increase of the guidance scale, the number of $T$ steps must be increased to completely remove the noise. The longest generation samples that we run used $T = 50$, needing maximum 4 seconds to denoise.

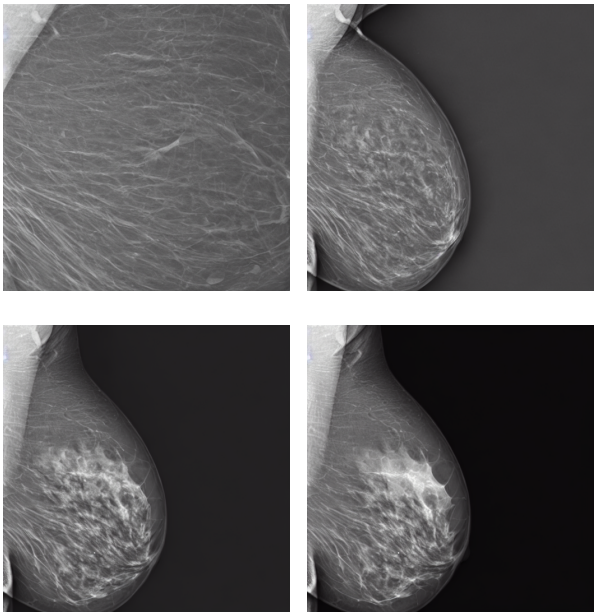


Figure 16: Guidance effect on the generation output. From upper-left to lower-bottom the guidance varies in a range from 1 to 4. Prompt: "A siemens mammogram in MLO view with small area and very high density".

The guidance scale, on the other hand, played a more crucial role in the quality and diversity of the generated images. Figure 16 shows the effect of the guidance scale on the image generation.

First, we observe that a guidance scale of 1 does not suffice for a meaningful generation. This is a common behavior for stable diffusion pipelines, as the image must be pushed further in the prompt direction (figure 9). It can be seen that the increase in the guidance value not only generates a more meaningful image, but also adjusts the characteristics of the mammogram to better match the text prompt. For example, at guidance 2, the mammogram still presents low breast density. In

the following 3 and 4 guidance values the breast density increases, as well as the overall quality of the image.

Nevertheless, there exists a trade-off between prompt fidelity and generation diversity. If the guidance scale is high, the generated images may all look similar, creating some kind of "mode collapse" for DM.

To quantitatively assess this phenomenon, we computed the MS-SSIM metric for different guidance scale values. The mean and standard deviation of the MS-SSIM value among 20 images of the same prompt and guidance value were computed and are shown in table 3. The experiment was repeated for the two vendors and the fusion model. The prompt was randomly selected for each model.

Table 3: Guidance scale effect on the MS-SSIM of the three SD models. The lower the MS-SSIM the higher the image diversity.

| | Hologic | | Siemens | | Fusion | |
|---|---|---|---|---|---|---|
| Guidance | Mean↓ | STD | Mean↓ | STD | Mean↓ | STD |
| 4 | 0.29 | 0.16 | 0.38 | 0.19 | 0.37 | 0.14 |
| 5 | 0.34 | 0.16 | 0.36 | 0.17 | 0.44 | 0.16 |
| 6 | 0.38 | 0.12 | 0.41 | 0.17 | 0.51 | 0.15 |
| 7 | 0.38 | 0.1 | 0.34 | 0.17 | 0.49 | 0.19 |
| 8 | 0.43 | 0.11 | 0.42 | 0.2 | 0.53 | 0.14 |
| 9 | 0.42 | 0.13 | 0.43 | 0.16 | 0.44 | 0.17 |
| 10 | 0.49 | 0.12 | 0.41 | 0.13 | 0.6 | 0.11 |
| 11 | 0.5 | 0.12 | 0.47 | 0.17 | 0.51 | 0.14 |
| 12 | 0.52 | 0.11 | 0.46 | 0.16 | 0.47 | 0.12 |
| 13 | 0.48 | 0.1 | 0.42 | 0.16 | 0.51 | 0.17 |
| 14 | 0.5 | 0.11 | 0.4 | 0.18 | 0.47 | 0.14 |

From these results, it can be seen that, overall, the higher the guidance value the lower the generation diversity, as the MS-SSIM value decreases. This suggests that the value of the guidance scale must be carefully selected as a very low value will generate low quality images but with high diversity. Conversely, a high guidance value (above 6) will generate a mammogram more faithful to the prompt description but with low diversity.

Also, we attest that the optimum guidance scale will depend on the model, so empirical experiments using the MS-SSIM metric are encouraged.

### *4.5. Qualitative visual assessment*

A more formal visual assessment was performed with the radiological evaluation of 53 synthetic image by a radiologists. The results of the test are summarized as a ROC curve in figure 17. The shape of the ROC curve bears resemblance to the random guess curve, suggesting that the radiologists cannot easily identify the difference between real and synthetic images. Moreover, the AUROC value obtained by the radiologist for this synthetic classification task was 0.49.

### *4.6. Quantitative CAD assessment*

The heatmaps of six Explainability AI methods, computed by Sam et al., were obtained for a healthy mammogram with an inpainted synthetic lesion. All six

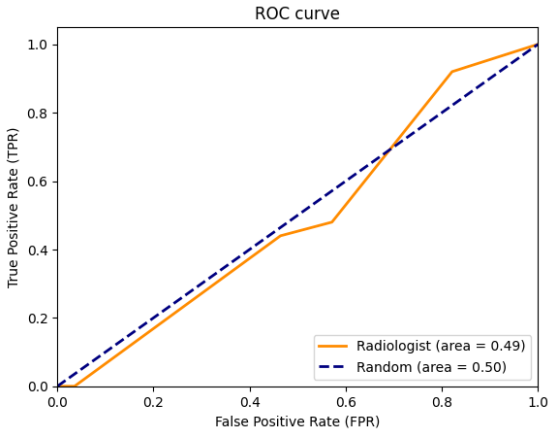
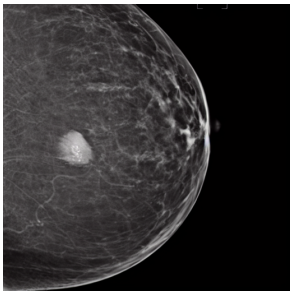
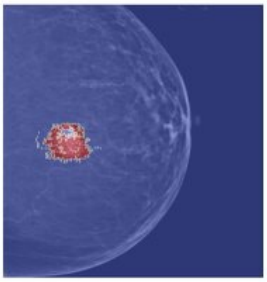



Figure 17: ROC curve of radiological assessment.

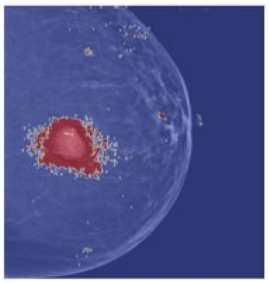
methods can be found in Sam-Millan's thesis report. here we present only three methods: gradcam, saliency and occlusion, with their respective heatmaps in figure 18.



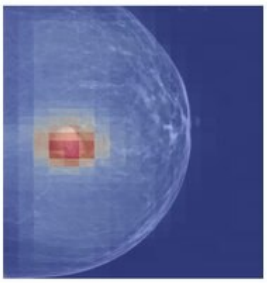
(a) Inpainted synthetic lesion.



(b) Gradcam heatmap.



(c) Saliency heatmap.



(d) Occlusion heatmap.

Figure 18: Explainability AI methods heatmaps of synthetic lesion over real healthy mammogram.

All three maps show that the classification method used by Sam et al. focuses on the synthetic lesion to make the prediction. This means that this CAD system is sensible to the presence of the lesion, which suggests that it may contain a pixel distribution similar to those present in real images.

### *4.7. MAM-E Graphical user interfaces*

We decided to build GUIs to make the pipelines of both tasks available and easy to use to the public. Our

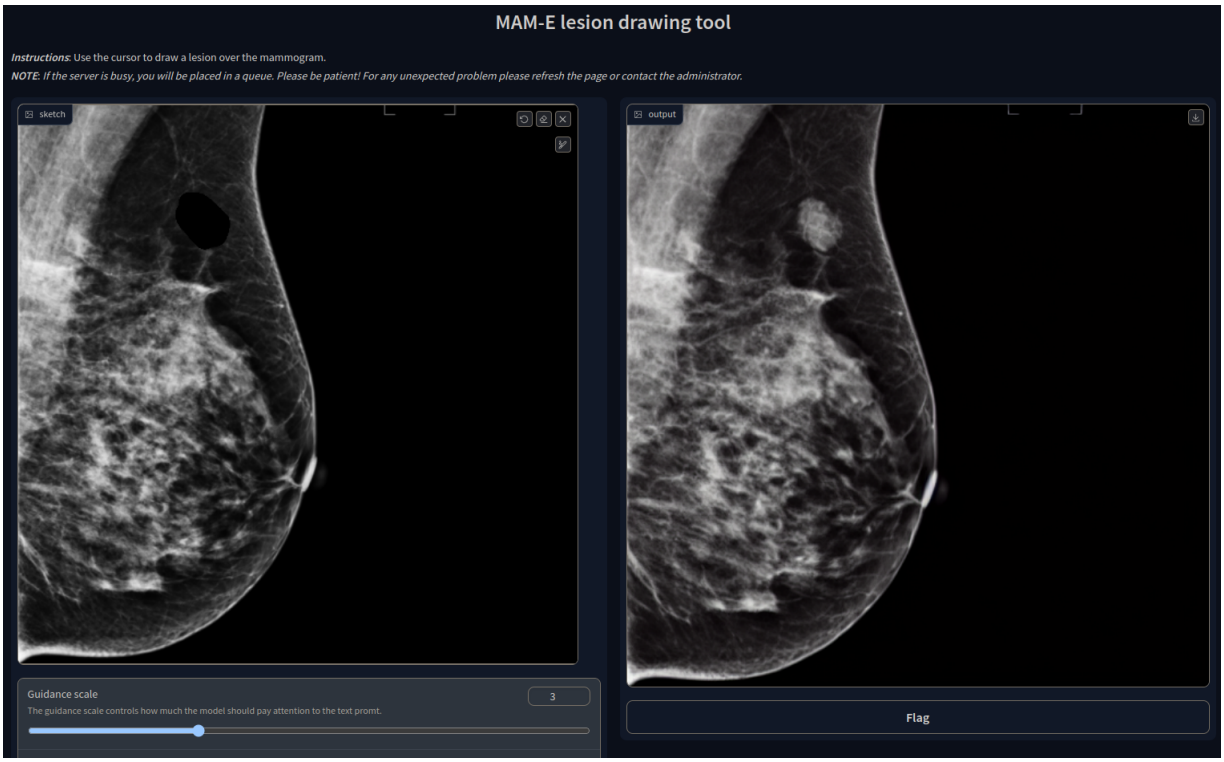



Figure 19: MAM-E for lesion drawing.

GUIs can run in remote servers and be accessible on the web thanks to the *GradIO*, an open-source Python package for rapid generation of visual interface of ML models, by Abid et al. (2019).

We developed five GUIs, one for each of our main diffusion pipelines. Two were designed for the conditional generation of mammograms of the original Siemens and Hologic datasets separately, with their own prompt characteristics. Similarly, one pipeline was created for the fusion of both datasets and it is presented as example in figure 1. In these three cases, the personalization options are set fixed and the user can only pick from the available options. Nevertheless, we added the option of a negative prompt, which allows the user to further personalize the generation.

The idea of the negative prompt is to specify some features that would like to avoid. For instance, in the cases when a synthetic image presents a gray or white background, a negative prompt of "white background" or "no black background" has shown to make the backgorund black.

In the case of the inpainting task, the GUIs has the option to upload the image that will be inpainted, although a default image is available. An interactive drawing brush is then activated, with which a lesion can be inpainted in any part of the mammogram, as shown in figure 19.

Given that the pretrained weights are available in the *Hugigng Face* personal repository of the first author, and that the code to run the GUI interface is publicly available in the GitHub repository of the same authorship, all five GUIs can be run with graphic cards of around 4 GB of GPU memory capacity.

## 5. Discussion and conclusions

We can encompass the results of this master thesis in three main blocks. The first block consists on exploring the implementation of diffusion models for digital mammography synthesis. The results of the vanilla and pure latent diffusion pipelines show that DM can be adapted for synthetic mammography generation, and that the data distribution of such images can be learned from scratch, although it would require a large dataset and long training time. Indeed, it follows that even though training a DM with one-channel 64x64 images is possible, training a similar model but with four-channels 64x64 latent representations requires more images, GPU resources and time.

Secondly, we found that fine-tuning a SD model pretrained on natural images with mammographic images is feasible and that the objective of the training process reduces to shift the learned data distribution from a non-medical one into the correspondent to our mammography datasets, individually for each mammography unit vendor or combined. This means that we can profit from the essential diffusion properties learned by pretrained natural models which, after trained with huge datasets and for long periods of time, have mastered to denoise images well.

Moreover, we found that stable diffusion text conditioning is a suitable generative model implementation to synthesize mammograms with specific characteristics and properties, giving the possibility to control several aspects of it, such as vendor, view, breast density and breast area. Stable diffusion also opened the possibility of extrapolating characteristics of one dataset into another, thanks to the control given by the CLIP text encoder through attention layers in the UNet, and at inference time by applying classifier-free guidance. We also found that SD can be modified for inpainting of synthetic lesions over healthy mammograms. The developed pipeline essentially only requires the modification of the input latent representation to include a mask to focus the generation process only in that region. All these models inference pipelines were made accessible and ready-to-use through GU interfaces, and the weights and code was made available through personal repositories.

Thirdly, we found initial evidence that the synthetic images coming from our implementation of SD could potentially be used for CAD systems in need of specific image characteristics or with the presence of lesions. A radiological assessment showed that the initial image quality can be compared with real mammograms and the use of explainability AI models helped to explore the behavior of a classification model when tested with our synthetic images with the help of heatmaps.

### *5.1. Limitations*

The first clear limitation of this work is the resolution and pixel depth of the synthetic mammograms. Although at the moment of publication of this work there are some improvements on the SD model for using pretrained weights for 768$x$768 resolution images, we decided to develop a pipeline first on 512x512 images. This limited resolution reduces the use of our synthetic images on CAD system that require higher resolution, such as micro-calcification detection. The pixel depth was also reduced from its original 16 bits to 8 bits to match the pretrained model requirements. This reduction losses some information in the images and reduces the overall contrast.

Despite the extensive visual assessments perform on the synthetic images, quantitative assessments remained limited in this work. Even though widely used fidelity and diversity metrics, such as the FID score, are being discouraged due apparent model bias, they can still be helpful during training to complement the training performance monitoring. This way, e.g. the model can be stopped or the learning rate can be modified if the generation diversity decreases.

Furthermore, even though some of the SD hyperparameters, such as learning rate and batch size, were changed to explore their effect on the training performance, this work did not prioritized HP tuning and limited itself to HP used in other medical and non-medical DreamBooth implementations.

An important limitation of this work is the lack of deep exploration of the effect of our synthetic images on CAD systems. Even though the assessment of EAI models can give some insights, it is required to train complete CAD pipelines with and without synthetic images to analyze performance changes.

### *5.2. Future work*

Acknowledging the limitations cited above, we plan to explore the use of quantitative metrics during training time, as well as an extensive and organized grid search of the optimal HP for our tasks.

After the release of the pretrained weights for 768x768 resolution images, we expect to perform minimal changes in our current pipeline to allow higher resolution mammography generation.

Additionally, even though the task 1 and task 2 pipelines can be combined manually by directly loading the synthetic or real images into the *MAM-E* drawing tool, we plan to combine this pipeline to fully automatize the generation process in one single GUI.

To assess the training performance of a CAD system using synthetic mammograms, we have started talks with the authors of another ViCOROB lab thesis defendant (Mekonnen et al.) to train a Fast RCNN architecture for lesion detection and bounding box prediction.

## Acknowledgments

I would like to thank my master thesis and future PhD supervisor, Robert Martí, for the guidance, patience, flexibility and the opportunity to develop such an interesting research topic in a pleasant work environment. I would also like to thank the ViCOROB community, specially the MAIA fellows, for the mutual camaraderie and the daily motivation to work. A special acknowledgment to the *Hugging Face* organization, for their grandiose labor to make deep learning resources, documentation, courses and source code available for everybody.

*Rangiferā tarandā.*

## Source code

The source code can be found in: https://github.com/Likalto4/diffusion-models_master.
The pretrained weights can be found in: https://huggingface.co/Likalto4.

## References


Abid, A., Abdalla, A., Abid, A., Khan, D., Alfozan, A., Zou, J., 2019. Gradio: Hassle-Free Sharing and Testing of ML Models in the Wild. URL: http://arxiv.org/abs/1906.02569. arXiv:1906.02569 [cs, stat].

Chambon, P., Bluethgen, C., Delbrouck, J.B., Van der Sluijs, R., Połacin, M., Chaves, J.M.Z., Abraham, T.M., Purohit, S., Langlotz, C.P., Chaudhari, A., 2022. RoentGen: Vision-Language Foundation Model for Chest X-ray Generation. URL: http://arxiv.org/abs/2211.12737. arXiv:2211.12737 [cs].

Chong, M.J., Forsyth, D., 2020. Effectively Unbiased FID and Inception Score and Where to Find Them, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, Seattle, WA, USA. pp. 6069–6078. URL: https://ieeexplore.ieee.org/document/9156949/, doi:10.1109/CVPR42600.2020.00611.

Dhariwal, P., Nichol, A., 2021. Diffusion Models Beat GANs on Image Synthesis. URL: http://arxiv.org/abs/2105.05233. arXiv:2105.05233 [cs, stat].

Dorjsembe, Z., Odonchimed, S., Xiao, F., 2022. Three-Dimensional Medical Image Synthesis with Denoising Diffusion Probabilistic Models.

Frans, K., Soros, L.B., Witkowski, O., 2021. CLIPDraw: Exploring Text-to-Drawing Synthesis through Language-Image Encoders. URL: http://arxiv.org/abs/2106.14843. arXiv:2106.14843 [cs].

Halling-Brown, M.D., Warren, L.M., Ward, D., Lewis, E., Mackenzie, A., Wallis, M.G., Wilkinson, L.S., Given-Wilson, R.M., McAvinchey, R., Young, K.C., 2021. OPTIMAM Mammography Image Database: A Large-Scale Resource of Mammography Images and Clinical Data. Radiology: Artificial Intelligence 3, e200103. URL: http://pubs.rsna.org/doi/10.1148/ryai.2020200103, doi:10.1148/ryai.2020200103.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S., 2018. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium. URL: http://arxiv.org/abs/1706.08500. arXiv:1706.08500 [cs, stat].

Ho, J., Jain, A., Abbeel, P., 2020. Denoising Diffusion Probabilistic Models. URL: http://arxiv.org/abs/2006.11239. arXiv:2006.11239 [cs, stat].

Ho, J., Salimans, T., 2022. Classifier-Free Diffusion Guidance. URL: http://arxiv.org/abs/2207.12598. arXiv:2207.12598 [cs].

Kazerouni, A., Aghdam, E.K., Heidari, M., Azad, R., Fayyaz, M., Hacihaliloglu, I., Merhof, D., 2023. Diffusion Models for Medical Image Analysis: A Comprehensive Survey. URL: http://arxiv.org/abs/2211.07804. arXiv:2211.07804 [cs, eess].

Kingma, D.P., Welling, M., 2022. Auto-Encoding Variational Bayes. URL: http://arxiv.org/abs/1312.6114. arXiv:1312.6114 [cs, stat].

Korkinof, D., Rijken, T., O'Neill, M., Yearsley, J., Harvey, H., Glocker, B., 2019. High-Resolution Mammogram Synthesis using Progressive Generative Adversarial Networks. URL: http://arxiv.org/abs/1807.03401. arXiv:1807.03401 [cs].

Lu, C., Zhou, Y., Bao, F., Chen, J., Li, C., Zhu, J., 2022. DPM-Solver: A Fast ODE Solver for Diffusion Probabilistic Model Sampling in Around 10 Steps. URL: http://arxiv.org/abs/2206.00927. arXiv:2206.00927 [cs, stat].

Müller-Franzes, G., Niehues, J.M., Khader, F., Arasteh, S.T., Haarburger, C., Kuhl, C., Wang, T., Han, T., Nebelung, S., Kather, J.N., Truhn, D., 2022. Diffusion Probabilistic Models beat GANs on Medical Images. URL: http://arxiv.org/abs/2212.07501. arXiv:2212.07501 [cs, eess].

Nguyen, H.T., Nguyen, H.Q., Pham, H.H., Lam, K., Le, L.T., Dao, M., Vu, V., 2023. VinDr-Mammo: A large-scale benchmark dataset for computer-aided diagnosis in full-field digital mammography. URL: http://arxiv.org/abs/2203.11205. arXiv:2203.11205 [cs, eess].

Pinaya, W.H.L., Tudosiu, P.D., Dafflon, J., da Costa, P.F., Fernandez, V., Nachev, P., Ourselin, S., Cardoso, M.J., 2022. Brain Imaging Generation with Latent Diffusion Models. URL: http://arxiv.org/abs/2209.07162. arXiv:2209.07162 [cs, eess, q-bio].

Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I., 2021. Learning Transferable Visual Models From Natural Language Supervision. URL: http://arxiv.org/abs/2103.00020. arXiv:2103.00020 [cs].

Ramesh, A., Pavlov, M., Goh, G., Gray, S., Voss, C., Radford, A., Chen, M., Sutskever, I., 2021. Zero-Shot Text-to-Image Generation. URL: http://arxiv.org/abs/2102.12092. arXiv:2102.12092 [cs].

Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B., 2022. High-Resolution Image Synthesis with Latent Diffusion Models. URL: http://arxiv.org/abs/2112.10752. arXiv:2112.10752 [cs].

Rouzrokh, P., Khosravi, B., Faghani, S., Moassefi, M., Vahdati, S., Erickson, B.J., 2022. MULTITASK BRAIN TUMOR INPAINTING WITH DIFFUSION MODELS: A METHODOLOGICAL REPORT.

Ruiz, N., Li, Y., Jampani, V., Pritch, Y., Rubinstein, M., Aberman, K., 2023. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation. URL: http://arxiv.org/abs/2208.12242. arXiv:2208.12242 [cs].

Sohl-Dickstein, J., Weiss, E.A., Maheswaranathan, N., Ganguli, S., 2015. Deep Unsupervised Learning using Nonequilibrium Thermodynamics. URL: http://arxiv.org/abs/1503.03585. arXiv:1503.03585 [cond-mat, q-bio, stat].

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I., 2017. Attention Is All You Need. URL: http://arxiv.org/abs/1706.03762. arXiv:1706.03762 [cs].

Wu, E., Wu, K., Cox, D., Lotter, W., 2018. Conditional Infilling GANs for Data Augmentation in Mammogram Classification, in: Stoyanov, D., Taylor, Z., Kainz, B., Maicas, G., Beichel, R.R., Martel, A., Maier-Hein, L., Bhatia, K., Vercauteren, T., Oktay, O., Carneiro, G., Bradley, A.P., Nascimento, J., Min, H., Brown, M.S., Jacobs, C., Lassen-Schmidt, B., Mori, K., Petersen, J., San José Estépar, R., Schmidt-Richberg, A., Veiga, C. (Eds.), Image Analysis for Moving Organ, Breast, and Thoracic Images. Springer International Publishing, Cham. volume 11040, pp. 98–106. Series Title: Lecture Notes in Computer Science.