ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DEPARTMENT OF COMPUTER SCIENCE
AND ENGINEERING

ARTIFICIAL INTELLIGENCE

MASTER THESIS

in

Machine Learning for Computer Vision

REDUCING MEMORIZATION IN LATENT DIFFUSION MODELS FOR 3D MEDICAL IMAGES GENERATION

CANDIDATE
Giacomo Melacini

SUPERVISOR
Samuele Salti

CO-SUPERVISORS
Giuseppe Lisanti
Stefano Mazzocchetti

Academic year 2022-2023

Abstract

In the medical domain, the use of machine learning techniques for diagnosis, treatment planning, and medical imaging interpretation is becoming increasingly important. However, these approaches require a large amount of data, which is challenging to access due to its sensitive nature and related privacy concerns. Synthetic data generation, enabled by advances in generative techniques, provides a solution to create large anonymized datasets for training models without compromising patient privacy. Nonetheless, the presence of memorization in such datasets, meaning the exact replication of training images, has been assessed by many studies. This dissertation explores the use of Latent Diffusion Models (LDMs) for generating medical data, focusing on head CT scans, and investigates the phenomenon of memorization in synthetic datasets together with methodologies to detect and mitigate it. The study proposes an adaptation of the Lowe's ratio test to detect potential copies and evaluates two approaches, Privacy Distillation and Latent Filtering, for their effectiveness in addressing memorization issues. The findings contribute to understanding the potential of LDMs in generating realistic medical data while reducing concerns regarding their sharing. Results validate the Lowe's ratio test as a metric for assessing memorization and demonstrate the efficacy of the investigated memorization-countering techniques.

Contents

1 Introduction **1**

2 Background **4**

2.1 Denoising Diffusion Probabilistic Models 4

2.2 Latent Diffusion Models 5

2.3 Autoencoders 6

2.4 Related work 7

3 Data and Methods **9**

3.1 Data 9

3.1.1 Dataset 9

3.1.2 Data Preprocessing 10

3.2 Architecture 12

3.2.1 Autoencoder 12

3.2.2 Latent Diffusion Model 13

3.2.3 Training Workflow 14

3.3 Quality assessment techniques 14

3.4 Memorization assessment techniques 16

3.5 Memorization countering techniques 17

4 Experiments and Results **20**

4.1 LDM training and dataset generation 20

4.2 Memorization assessment 24

4.3 Memorization countering 26

5 Conclusions **30**

A Code Availability **32**

B Hyperparameters **33**

Bibliography **34**

List of Figures

2.1 The diffusion process in both forward and reverse fashion. . . 5
2.2 The architecture of Latent Diffusion Models. 6

3.1 Examples of meshes with different quality scores. 9
3.2 Distribution of the quality scores in train and test sets. 10
3.3 Example of middle slices of a preprocessed input image. . . . 11
3.4 An original mesh together with its reconstructed version after the preprocessing operations. 11
3.5 The architecture of the 3D VQ-VAE. 12

4.1 Plots of the training loss of the VQ-VAE and of the DDPM. . . 20
4.2 Axial (on the left) and saggital (on the right) view of a skull. . 21
4.3 Original, preprocessed and VQ-VAE reconstructed meshes. . . 21
4.4 Examples of skulls generated by the trained LDM. 22
4.5 Distribution of the highest correlation coefficient and of the Lowe's ratio in the test set and in the generated dataset. 25
4.6 Middle slice of the saggital plane of generated QS5 skulls with the corresponding copied training slice. 25
4.7 Distribution of the highest correlation coefficient and of the Lowe's ratio in the test set and all the synthetic datasets. . . . 27
4.8 Examples of generated skulls after applying different memorization countering techniques. 29

List of Tables

4.1 Quantitative evaluation of the quality of synthetic images via MS-SSIM and FID scores. 23
4.2 FID scores of the datasets grouped by QS. 24
4.3 MS-SSIMs of the datasets grouped by QS. 24
4.4 Quantitative evaluation of the memorization in the synthetic datasets. 28

B.1 Hyperparameters used for training the Latent Diffusion Model 33

1. Introduction

In the medical domain, data plays a crucial role in diagnosis, treatment planning, and research. Medical data can include patient records, clinical trial data, imaging data and more, yet this data is often highly sensitive, as it contains personal health information that is protected under privacy laws and regulations. The sensitive nature of medical data presents unique challenges in its handling and usage since unauthorized access or disclosure can lead to serious privacy violations, with potential legal and ethical implications. Therefore, stringent measures are required to ensure the confidentiality, integrity, and availability of this data. Despite these challenges, the use of medical data is essential for advancing healthcare, and machine learning techniques are demonstrating to be more and more important in this domain. For instance, ML models can be trained on medical data to predict disease outcomes, personalize treatment plans, or assist in medical imaging interpretation. However, these models often require large volumes of data, which can be difficult to obtain due to privacy concerns. One solution to this problem is the use of synthetic data, which can be generated from existing data and used for training models without compromising patient privacy. Thanks to the advances in the field of generative models, this approach is becoming increasingly popular in the medical domain, as it enables to create public anonymized datasets to be freely used for various medical tasks [1, 2]. For instance, generative models have been successfully used to improve results in tumor segmentation tasks [3] and Retinopathy of Prematurity detection [4]. Despite the high potential of generative techniques, these come also with challenges: the first one lies in the three

dimensional nature of medical images, such as MRIs and CT scans, which require a large amount of computational resources to be processed; while another challenge is inherent to the functioning of generative models like Latent Diffusion Models, which can suffer from memorization. Memorization happens when the generative model starts to generate samples that are almost identical to the training examples, which happens when the model assigns very high likelihood values to the training data points [5]. This is particularly frequent in the medical domain because the datasets used to train the models are often small, therefore being problematic not only because of the uselessness of replicated data, but also because it nullifies the efforts made to overcome the aforementioned privacy issues.

The goal of this thesis is threefold. The first objective is to successfully train a generative model, more specifically a Latent Diffusion Model, to generate novel medical data. The training dataset used for this task comprises head CT scans wherein only the skeletal structures have been retained. The trained model is then used to create an analogous synthetic dataset which is subsequently analyzed to assess its quality and realisticity. The second objective is to perform an in-depth analysis of the impact of memorization in the generated dataset: this includes ways to qualitatively and quantitatively measure the effects of memorization as well as a method to detect possible copies of the training set. Finally, the work focuses on addressing and overcoming memorization using two different methodologies. The first is an adaptation of Privacy Distillation, as initially proposed by Fernandex et al. [6]. It aims at developing a Latent Diffusion Model without exposing it to the original training dataset, while concurrently producing a synthetic dataset that closely resembles real-world data. The second methodology, Latent Filtering, modifies the generation process so that only images that are distant to the training ones are produced.

The dissertation follows a structured approach, beginning with a Background chapter which introduces the most relevant techniques used in this work, providing necessary context for the subsequent discussion. Following this, the Data and Methods chapter focuses on the dataset description, the preprocessing steps, and explains in depth the architecture of the used models and the training workflow. Moreover, it introduces the metrics used to measure the quality of the generations, as well as ways to detect memorization and counter it. An Experiments and Results chapter follows, presenting findings from the study, including the results of the generation process and the effectiveness of the strategies used to mitigate memorization. Finally, the Conclusions summarize the key insights derived from the research.

2. Background

This chapter provides an in-depth exploration of the generative techniques exploited in this work starting with their core concepts, advantages, and limitations. It concludes with a review of the related work in the field, highlighting the current challenges and potential solutions in the generation of 3D medical images using these models, as well as techniques to investigate and overcome memorization.

2.1 Denoising Diffusion Probabilistic Models

Denoising Diffusion Probabilistic Models (DDPMs) [7] represent a popular approach in the field of generative modeling, particularly in the context of image and signal processing. These belong to the family of generative models that aim at learning the probability distribution of complex data, allowing the synthesis of new high quality samples. The core concept behind DDPMs is the diffusion process, an iterative mechanism that transforms the input data into noise. It is defined as a Markov Chain of diffusion steps in which, at each step, Gaussian noise is added to the data. The model learns the reverse process, i.e. to remove the noise from the data at a specific step of the chain. In this way, starting from random noise, the model can generate a new sample by performing multiple reverse diffusion steps, each contributing to a progressive refinement of the image. The incorporation of a denoising mechanism enables DDPMs to handle a variety of data types with remarkable success, including images and signals, and makes them valuable in different domains, such as the medical one. The iterative refinement through multiple diffusion

steps ensures that the generated samples exhibit realistic textures, structures, and features, making DDPMs particularly valuable in applications like image synthesis, where faithful representation of complex visual content is crucial.

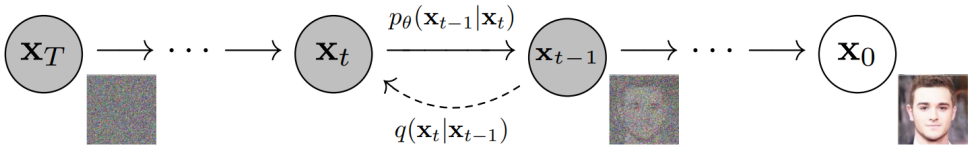


Figure 2.1: The diffusion process in both forward and reverse fashion as shown in Ho et al. [7].

A major drawback of Denoising Diffusion Probabilistic Models is that the probability distribution is learned in the pixel space, i.e. the same as the input data, which happens because the Markov Chain of noising and denoising is applied directly to the pixels of the input image. This is an issue because processing data in such an high dimensional space requires huge computational resources. As a result, the only way to use DDPMs in the pixel space consists in processing images at a very low resolution, which is not desirable. In the next section it will be showed how this problem can be faced in order to generate higher quality images.

2.2 Latent Diffusion Models

Latent Diffusion Models (LDMs), introduced by Rombach et al. [8], can be seen as an evolution of DDPMs as they overcome the computational constraints derived from the size of the data used in the diffusion process. The fundamental innovation of LDMs is the introduction of a compressed latent space, which captures the hidden patterns and features within the data. LDMs take advantage of the latents by training a DDPM in that space, so that the reverse diffusion process is learned not in the pixel space, but in the compressed latent space, thus making both training and inference faster. To create such compressed space, LDMs make use of an autoencoder model, which is trained to compress and reconstruct the input data.

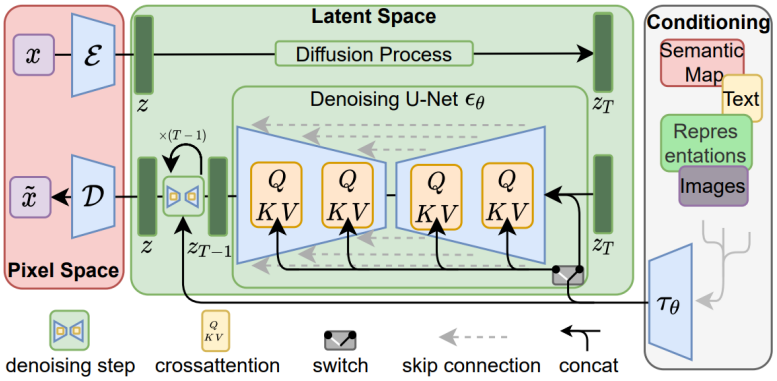


Figure 2.2: The architecture of Latent Diffusion Models as shown in Rombach et al. [8].

Latent Diffusion Models can also be used for conditional generation, i.e. they can generate new samples conditioned on class labels or descriptive information. The latent space can be designed not only to represent the input data, but also to incorporate such conditions by embedding and fusing them to the input image representation via cross-attention, as shown in figure 2.2. Putting everything together, the training of an LDM begins by training an autoencoder network to efficiently encode and reconstruct images, establishing a latent space. Subsequently, a diffusion model is trained to generate novel samples within this latent space, potentially conditioned on labels or textual information. Finally, the generated latent representations undergo reconstruction through the autoencoder, yielding realistic high quality images.

2.3 Autoencoders

As shown in the previous section, autoencoders are a key part of LDMs as they are trained to shape the latent space. An autoencoder model can be seen as a compression algorithm and consists of two main components: an encoder, which compresses the input data; and a decoder, which reconstructs the original data from the compressed representation. The encoder and decoder are trained together, with the goal of minimizing the difference between the original input data and the reconstructed output. The autoencoder network used in

this work is a Vector Quantized - Variational Autoencoder (VQ-VAE) [9, 10]. The main feature of this model consists in creating a discrete latent space by using a vector quantization layer after the encoder. The use of a discrete latent space has shown to produce higher quality reconstructions with respect to traditional VAEs, being able to model more complex data distributions. Moreover, the discrete nature of the latents makes the model easier to interpret, with each discrete latent variable potentially representing a specific feature or characteristic of the data.

2.4 Related work

The generation of 3D medical images presents many challenges related to the complexity of the data and the sensitive nature of the information involved, and not many studies have explored the potential of LDMs in this context. Pinaya et al. [11] leveraged LDMs to generate synthetic data from high-resolution 3D brain images modelled from a public dataset of 31740 images and conditioned on features such as sex and age. Khader et al. [12] reproposed the same approach using four different datasets with about 1000 elements each, obtaining good results in generating novel images. Their work showed how LDMs can be successfully used also when the training dataset is very small, while also proving how generated data can be used in self-supervised pre-training to improve subsequent tasks.

However, neither of these works analyzed the impact of memorization in the synthetic datasets, but recent studies have shown the significance of this problem. Different works have analyzed the causes of memorization and which ones mostly contribute to it [13, 14]. The most relevant factor is the size of the dataset, which can be particularly problematic in the medical domain because the availability of data is scarce. Akbar et al. [15] investigated the impact of memorization in LDMs when trained on 2D medical images: they used correlation between images to measure memorization and showed that many

generated images are highly correlated with the training ones, especially when the available datasets are small. Dar et al. [16] extended the investigation to LDMs trained on 3D medical images: they used a more sophisticated metric to measure memorization, i.e. a self-supervised model based on contrastive learning. Their study confirmed the fact that latent diffusion models indeed memorize training images.

Despite the relevance of the problem there are not many studies proposing solutions to memorization, which is problematic because this problem affects the primary goal of generating medical data, i.e. overcoming the privacy related issues. Fernandez et al. [6] introduced Privacy Distillation, a framework to train a generative model without exposing it to any identifiable data. Their solution, tested on 2D medical images, consists in the following steps: training a first diffusion model on real data; generating a synthetic dataset using this model; filtering it to exclude images with a re-identifiability risk; and finally training a second diffusion model on the filtered synthetic data only.

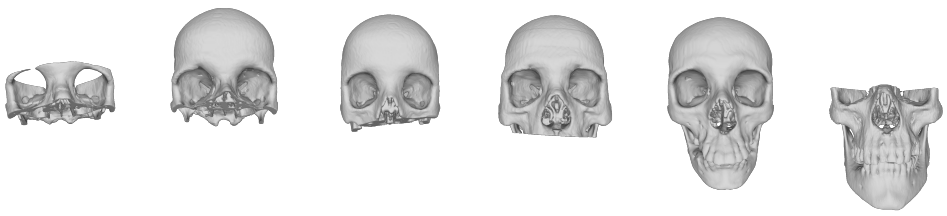
3. Data and Methods

This chapter provides a comprehensive view of the data, methods and evaluation techniques employed in the subsequent experiments, setting the stage for the detailed analysis and results presented in the following chapter.

3.1 Data

3.1.1 Dataset

The data used in this work is the union of two different datasets of anonymized head CAT scans: CQ500, a publicly available dataset of 355 scans introduced by Chilamkurthy et al. [17]; and a private dataset of 591 scans from Bologna's Sant'Orsola hospital. Each scan had already been segmented to retrieve the meshes depicting the skeletal part of the head. Since the CTs have been acquired for different purposes, most of the scans do not depict the entirety of the skull. Therefore, a quality score (QS) was added to describe the extension and completeness of each shape: QS 1 images represent the least complete skulls; QS 5 images contain almost complete skulls; QS 6 scans contain the mandible but miss the upper part of the skull.



(a) QS 1 (b) QS 2 (c) QS 3 (d) QS 4 (e) QS 5 (f) QS 6

Figure 3.1: Examples of meshes with different quality scores.

Meshes with a quality score of 1 are not informative enough to be kept for training the models and, similarly, other scans had been manually labelled as qualitatively bad. Thus, in our study, entries belonging to these categories have been removed. The final dataset is made of 909 scans, 341 from the public dataset and 568 from Sant'Orsola's dataset, and is split in a stratified fashion to obtain a train (727) and a test set (182). The distribution of quality scores among the splits is depicted in Figure 3.2. As it is possible to notice, the dataset is not balanced: while QS 2 to QS 5 skulls have a similar cardinality, there are only few QS 6 images. This could influence the training process, particularly when training the DDPM conditioned on the QS, and the subsequent evaluation metrics.

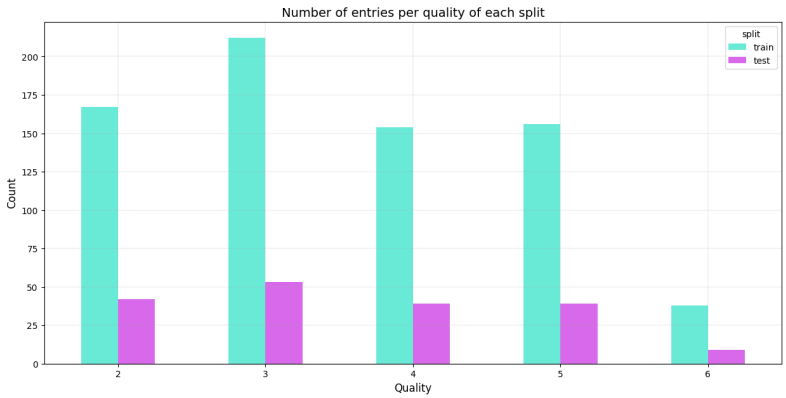


Figure 3.2: Distribution of the quality scores in train and test sets.

3.1.2 Data Preprocessing

Since the models require voxels as input data, the first data preprocessing step consists in the voxelization of the meshes. A global bound is calculated from the whole dataset to delimit a volume containing all the meshes. In this way we expect the anatomycal parts to be of the same size irrespectively of the quality score of the scan. A uniformly distributed set of query points with shape 512x512x512 is extracted from the volume and, for each mesh, the occupancy is calculated at each point. This leads to a spacing of 0.463mm, 0.335mm and 0.499mm in the saggital, trasversal and longitudinal axes respectively. The

images are min-max normalized to the range between -1 and 1 and, because of memory constraints, they are scaled to 128x128x128 pixels. Moreover, the quality score of each mesh is one-hot encoded to be fed to the Diffusion Model for quality score conditioned generation.

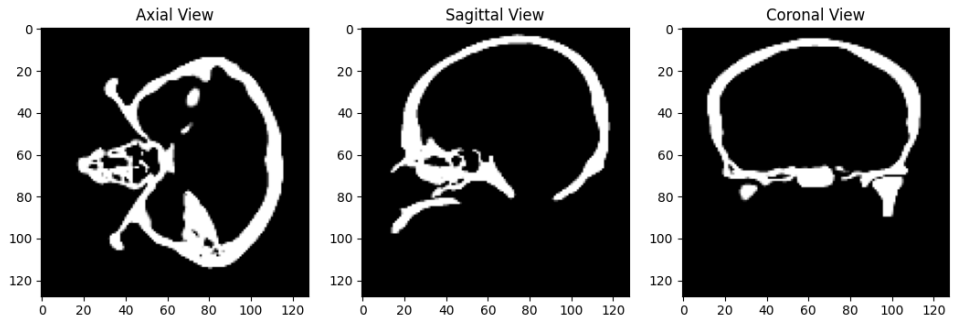
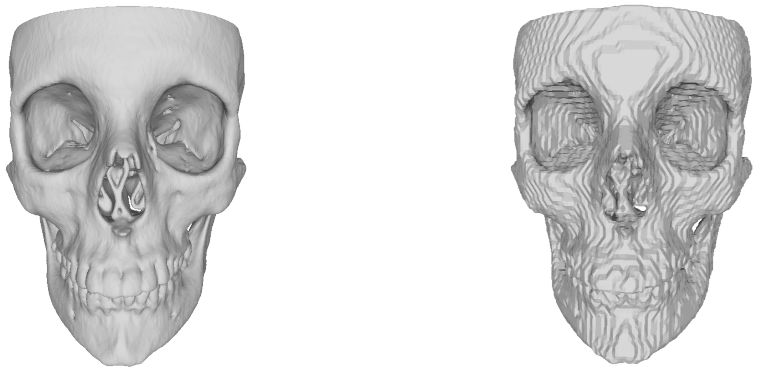


Figure 3.3: Example of middle slices of a preprocessed input image.

Figure 3.4 shows the original mesh of a skull compared to a reconstructed one. The latter is computed as the result of the marching cube algorithm [18] applied to the preprocessed image. Areas which are richer in detail, such as the dental region and the paranasal sinuses, suffer the major loss of information. This is a result of both the voxelization and the rescaling, which contribute to a sensible decrease in the quality of the input data.



(a) Original mesh (b) Re-built mesh

Figure 3.4: An original mesh together with its reconstructed version after the preprocessing operations.

3.2 Architecture

3.2.1 Autoencoder

The neural network used to encode (and decode) the images into a compressed latent representation is a Vector Quantized Autoencoder (VQ-VAE) [9, 10]. In order to encode and decode 3D images, we followed the approach of Khader et al. [12], i.e. 2D convolutions are replaced by 3D convolutions. Similarly to a traditional autoencoder, in a VQ-VAE the input data $x \in \mathbb{R}^{c \times D \times W \times H}$ is compressed into a dense representation by an encoder network to obtain $z_e \in \mathbb{R}^{k \times (D/s) \times (W/s) \times (H/s)}$, where k is the embedding dimension, s is a compression factor and D, W and H are respectively the depth, width and height of the input image. The dense representation is then passed through a vector quantization layer, which consists of a codebook of n vectors $Z \in \mathbb{R}^{n \times k}$, where each element of the dense representation is replaced by the nearest code vector (in Euclidean distance), resulting in a discrete latent representation of the input data. Finally, the quantized representation is passed through a decoder network, which attempts to reconstruct the original image.

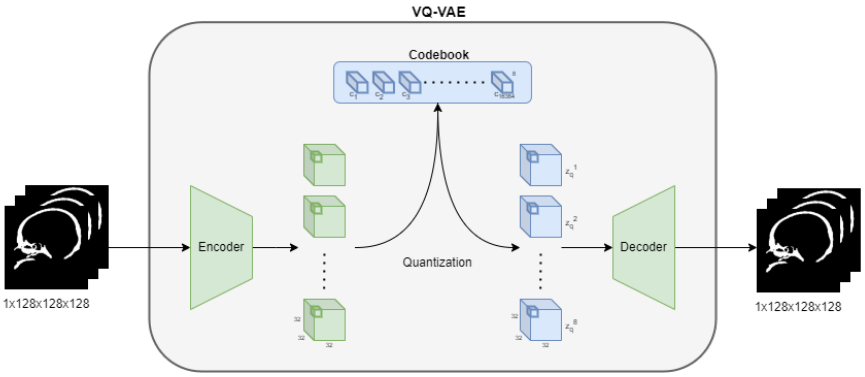


Figure 3.5: The architecture of the 3D VQ-VAE.

The VQ-VAE is optimized with three loss components: a reconstruction loss L_{rec}, which is computed as the mean absolute error between the input image and the reconstructed one; a perceptual loss $L_{perceptual}$, which measures the perceptual difference between the input and reconstructed images; and a

commitment loss L_{commit}, which is defined as the mean squared difference between the encoder's output and the selected code vector. Each vector of the codebook is optimized by maintaining an exponential moving average of all the dense vectors that get mapped to it.

3.2.2 Latent Diffusion Model

The technique employed for synthesizing new head CT scans leverages the power of a Latent Diffusion Model (LDM) [8]. This model operates on the unquantized representations produced by the VQ-VAE described in the previous section, using them as inputs to a Diffusion Model [7]. LDMs are built on top of a fixed Markov chain over the latent variables, which is used to modify the input image by adding Gaussian noise with increasing variance over a series of timesteps T. A neural network, conditioned on the noised version of the image at a given timestep t and the timestep itself, is trained to learn the reverse process, i.e. it learns the noise distribution used to modify the image. This allows the data distribution at $t-1$ to be inferred. The loss used to train the network is calculated as the L1 difference between the noise that was added at the specific timestep and the output of the network. As T grows large, the distribution of the final timestep can be approximated by a standard normal distribution, and by sampling from it and traversing the Markov chain in reverse, a new realistic image can be yielded. Following the approach of Khader et al. [12], the architecture used to denoise the images is a UNet3D, which is a UNet [19] modified to support 3D input data by substituting 2D convolutions with 3D convolutions. The models are also conditioned on the quality scores, which are fed as one-hot encoded vectors and subsequently embedded through two fully connected layers. The quality score embedding is then concatenated to the timestep embedding and fed as input to the cross-attention layers of the diffusion model.

3.2.3 Training Workflow

The training of each model is performed in two stages: the first stage consists in training the VQ-VAE to encode and decode 3D images into compressed latent representations; the second stage is the training of the Diffusion Model. The input to this model has to be normalized to the range between -1 and 1 [7], hence the output of the encoder of the VQ-VAE must also be in that range. Following the work of Khader et al. [12], this is done by min-max normalizing the dense representations by taking as min and max values the minimum and maximum values that can be found in the learned codebook. The output of the diffusion model is then quantized and decoded to build 128x128x128 images. The hyperparameters used to train the networks are shown in appendix B.

3.3 Quality assessment techniques

The trained models are then exploited to create a synthetic dataset analogous to the training one. Two metrics are used to evaluate the performance of the generative models and the quality of the generated datasets: FID score and MS-SSIM.

The Fréchet Inception Distance (FID) score [20] is a popular metric used to evaluate how realistic the synthetic datasets are: it measures the similarity between two sets of images, typically the generated images and the real images they are supposed to mimic. The FID score does this by comparing the distributions of Inception network features extracted from the two sets of images. A lower FID score indicates that the two sets of images are more similar, and thus that the generative model is performing better. The FID score is calculated in two steps: first, a pre-trained Inception network is used to extract a feature vector from each image in both sets; the feature vectors are then used to compute a multivariate Gaussian distribution (characterized by a mean and covariance matrix) for each set. The FID score is then the Fréchet distance between these two Gaussians, which is a measure of similarity between the

two distributions. When it comes to 3D data, such as voxelized meshes, the FID score can be computed using a network that extracts features from 3D images, such as Med3D [21]. Med3D is a pre-trained 3D convolutional neural network that has been trained on an aggregation of datasets coming from different medical tasks. Following the work of Pinaya et al. [11], Med3D is used to extract the features from the 3D images, which can be used to compute the FID score in the same way as for 2D images, thus allowing for a meaningful comparison of the realisticity of 3D generative models.

The Multi-Scale Structural Similarity Index Measure (MS-SSIM) [22] is an extension of the Structural Similarity Index Measure (SSIM), which compares local patterns of pixel intensities that have been normalized for luminance and contrast. While SSIM is computed on a single scale, MS-SSIM considers similarity at multiple scales, which makes it more robust and better aligned with human visual perception. MS-SSIM is calculated by sliding a window over the image and comparing the structure, luminance, and contrast of the two images in each window. The final MS-SSIM score is the product of these comparisons at multiple scales with a score of 1 indicating perfect similarity, and a score of 0 indicating no similarity. In the case of MS-SSIM computed on 3D images, the only difference lies in the fact that the slid window is not two dimensional, but three dimensional. While this metric is typically used to measure similarity between images, it can also be used to compute the diversity of a dataset. The idea is to compute the MS-SSIM for n random pair of images in a dataset, and then use the average of these scores to quantify the overall diversity of the dataset itself. If the dataset is diverse, then the images in it should be quite different from each other, leading to a low MS-SSIM, conversely, if the dataset is not diverse, then the images in it will be similar to each other, leading to a high MS-SSIM. This is useful to compare the diversity of the generated dataset with respect to the one of the training and test sets. In this case, the optimal result is not to obtain the lowest MS-SSIM possible, but

to obtain similar results in the real and synthetic datasets, i.e. obtaining the same degree of diversity.

3.4 Memorization assessment techniques

Metrics like FID and MS-SSIM are useful to measure the realisticity and diversity of the generated dataset, but they are not significant to highlight the impact of memorization. Optimal results on those metrics could potentially hide a high number of copies in the synthetic dataset. Therefore, other metrics must be used to assess the presence of memorization. For instance, following the approach of Akbar et al. [15], image correlation can be used. The idea is to measure the correlation between each generated image and all the images in the training set. Similarly, this process can be repeated correlating the test set to the training set. By comparing the distribution of the highest correlation coefficients in both scenarios, one can gain insights into the extent of memorization. If the model is generalizing well, then the distribution of the highest correlation coefficients for the generated images should be similar to the distribution for the test set. This would indicate that the model is creating novel images that are similar in structure to the training set, without directly copying or memorizing it. On the other hand, if the model is memorizing the training data, the distribution of the highest correlation coefficients for the generated images with the training set would be significantly higher than the results obtained for the test set. This would suggest that the model is producing images that are overly similar to the training ones, indicating memorization.

Since the objective of memorization assessment is to verify the degree of copying in the synthetic dataset, correlation may not be the most accurate metric because an high maximum correlation does not always imply that an image is a copy. For instance, a generated image could be highly correlated with two different real images, therefore being a copy of neither. A different approach to memorization assessment is to consider the task of identifying copies in the

synthetic dataset as a matching problem: finding a match would correspond to detecting a copy. To do so, it comes handy a modified version of the Lowe's ratio test. This is a method originally proposed in the context of SIFT (Scale-Invariant Feature Transform) for matching key points between images. In the context of finding copies with correlation, the test would be done using the ratio between the second highest correlation coefficient and the highest correlation coefficient. The idea is that if a synthetic image is a direct copy of a training image, it should have a very high correlation with that training image and significantly lower correlation with all other training images. This would result in a low Lowe's ratio, indicating a potential copy. On the other hand, if a synthetic image is not a direct copy, it should have similar correlation with multiple training images, resulting in a higher Lowe's ratio. This is because the highest and second highest correlation would be close in value, making their ratio closer to 1. Also in this case the Lowe's ratio can be computed both for the generated dataset and for the test set.

The Jensen-Shannon (JS) divergence can be used to quantitatively measure the degree of memorization. It is a measure of similarity between two distributions that can be computed for both the two highest correlation coefficient distributions and the two Lowe's ratio ones. Taking two probability distributions P and Q, the JS divergence is defined as the average of the Kullback-Leibniz (KL) divergence of P from the average distribution M, and the KL divergence of Q from the average distribution M, where M is the average of P and Q. This metric is always between 0 and 1 being 0 when the two distributions are the same, and 1 when they are completely different. In this case we aim for the JS divergence to be as small as possible.

3.5 Memorization countering techniques

As explained in the first chapter, an objective of this thesis is to address and overcome the problem of memorization. A possible solution, following the

work of Fernandez et al. [6], is to perform Privacy Distillation, i.e. training a Diffusion Model without exposing it to any re-identifiable data. This process consists in the following steps:

1. Train an LDM $(\theta_{real}^{vq-vae}, \theta_{real}^{ddpm})$ with real data D_{real};
2. Use $(\theta_{real}^{vq-vae}, \theta_{real}^{ddpm})$ to generate a synthetic dataset D_{gen};
3. Filter the generated dataset D_{gen} removing re-identifiable data, i.e. copies of the training set, to obtain D_{filt};
4. Train a new Diffusion Model θ_{filt}^{ddpm} with the filtered synthetic dataset D_{filt};
5. Use $(\theta_{real}^{vq-vae}, \theta_{filt}^{ddpm})$ to generate a synthetic dataset D_{final} which is blind to the original dataset D_{real}.

The challenge lies in the filtering step, where copies of the training set must be identified and removed from the generated data. Since the values of the correlation of non copies, i.e. the values of the test set, are known, a simple solution is to filter out images whose maximum correlation is higher with respect to those values. The actual threshold can be chosen from a quantile of the maximum correlation of the test set. This leads to a fairly conservative filtering: even if it is possible to assume that the images with maximum correlation lower than the test set threshold are non copies, it is not safe to imply that the opposite is always true. An alternative is to take into consideration the Lowe's ratio: it is possible to identify copies by thresholding it. In this case we consider copies the images whose Lowe's ratio is lower than a certain value. As previously done with the maximum correlation filter, also in this case the value of the threshold can be chosen from a quantile of the Lowe's ratios of the test set. Moreover, it is also possible to rank the generated images on descending Lowe's ratio and keep the top-k results. Since the Lowe's ratio represents a degree of copying of an image, this ranking can be interpreted as an ordering from the least copied to the most copied image.

The second approach used to counter memorization acts directly on the generation process. It is based on the idea that copies of the training images are generated from synthetic latents that are close to the training ones, and that by getting far from those latents memorization could be avoided. The process is the following:

1. Create, for each quality score, an index of training latents;
2. Generate through reverse diffusion M latents conditioned on the quality score;
3. Keep only the latent z_e^m that is farthest (in euclidean distance) from the training latents of the same QS, i.e. the one that has farthest nearest neighbour;
4. Finally, quantize and decode z_e^m to yield the image.

On one hand, this method reduces the likelihood of reproducing exact copies of the training images and does not require another training of the DDPM. On the other hand, generating multiple latents is computationally expensive, thus making the generation process slower.

The next chapter will present the experiments and the results of the generation, as well as the results of the methodologies for countering memorization proposed in this section.

4. Experiments and Results

This chapter offers a comprehensive exploration of the experiments done for training an LDM, generating a novel synthetic dataset and evaluating its quality. It also displays the different strategies adopted to analyze and address memorization. Through empirical analyses and quantitative assessments, we gain valuable insights into the capabilities and limitations of the proposed techniques.

4.1 LDM training and dataset generation

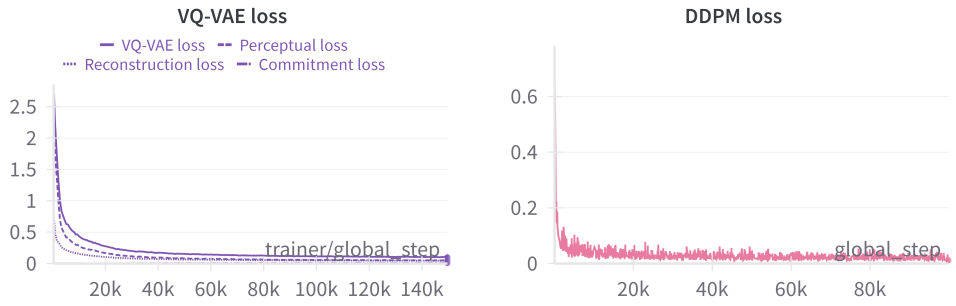


Figure 4.1: Plots of the training loss of the VQ-VAE and of the DDPM. Both models converged successfully. The VQ-VAE plot shows the different components of the loss: the reconstruction, perceptual and commitment losses.

As explained in section 3.2.2, the LDM has been trained in two stages: the first one is the training of the VQ-VAE while the second one is the training of the DDPM. In both cases, as shown in figure 4.1, the model converged successfully and no fine-tuning of the hyperparameters was required.

In order to measure the ability of the autoencoder, the trained VQ-VAE has

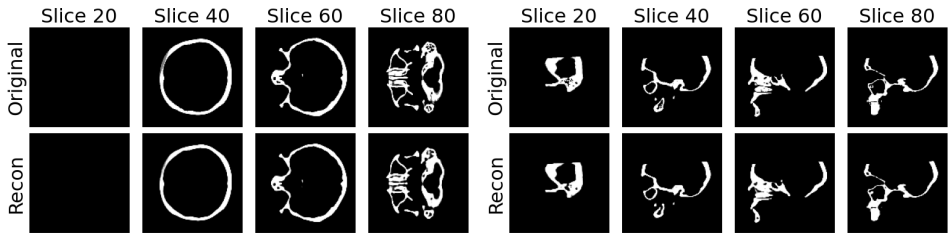


Figure 4.2: Axial (on the left) and saggital (on the right) view of a skull. The first row shows the original image, while the second row shows the reconstructed one.

been used to reconstruct the images of the test set. Even if the model converged well, by inspecting figure 4.2 it is possible to notice that the test set reconstructions present some defects. For instance, smaller holes and details which are present in the original images are missing in the reconstructions. This is problematic because we can expect just the same level of detail coming out of the DDPM.

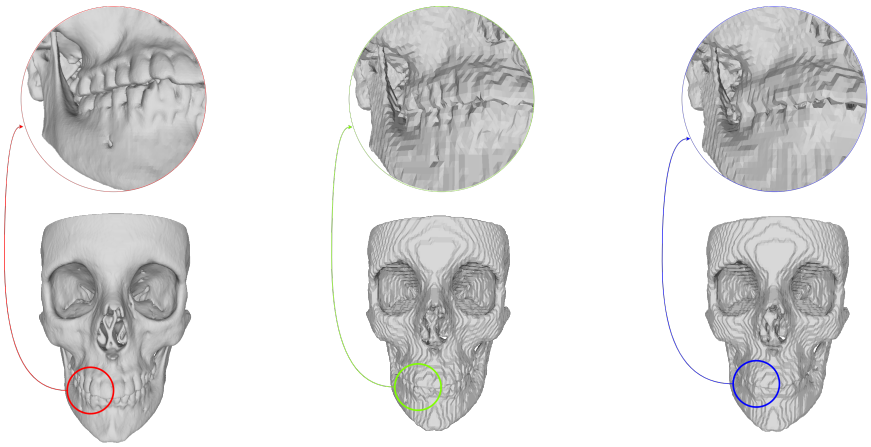


Figure 4.3: Original, preprocessed and VQ-VAE reconstructed meshes. It is possible to observe how the quality diminishes during the process. Still, most of the detail is lost in the preprocessing.

In order to qualitatively analyze the skulls, the meshes of both the preprocessed test voxel grids and the ones reconstructed by the VQ-VAE have been re-built using the marching cubes algorithm. Similarly to what was observed in the voxel space, also by looking at the meshes it is possible to recognize

the missing details. Still, the quality bottleneck is in the preprocessing phase, when a lot of information is lost during the voxelization of the mesh and the rescaling of the high quality voxel grid.

The training of the DDPM worked well too, and the fit model was used to generate a synthetic dataset of 3000 images with the same QS distribution of the training set. The generated images show a good degree of realisticity and, at the same time, present the peculiarities of the QS they were conditioned with. Still, it is possible to find in the generated images the same defects that were visible in the skulls reconstructed by the VQ-VAE. For instance, the level of detail visible in the dental area of the synthetic images is worse than the one of the real and preprocessed skulls. Overall, there are also some wrongly generated images, showing only partial anatomical structures. This happens more frequently for QS 6 skulls, probably due to their low cardinality in the training set.

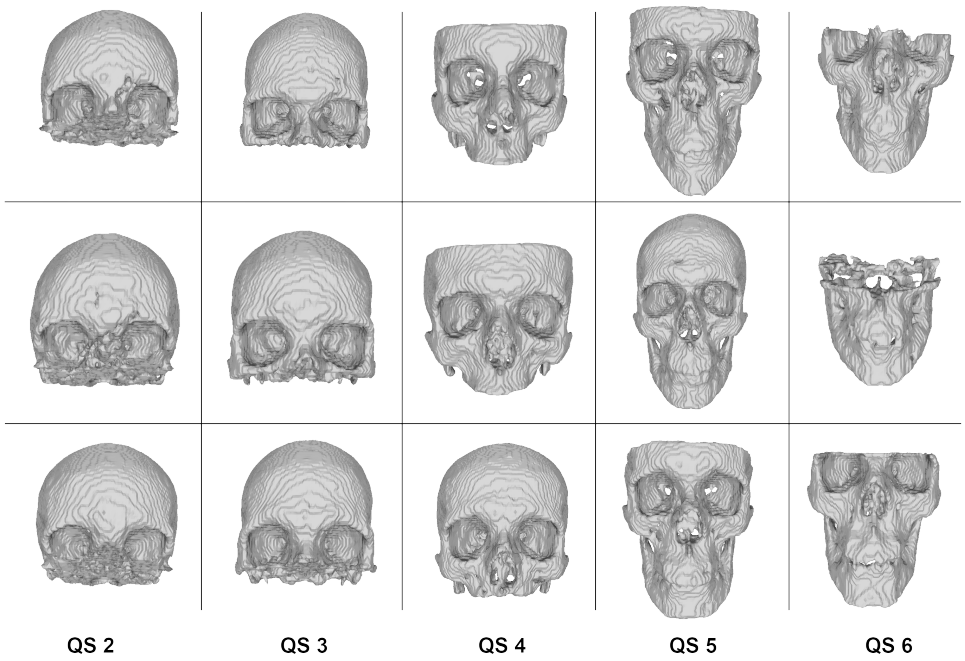


Figure 4.4: Examples of skulls generated by the trained LDM.

MS-SSIM and FID scores have been computed to quantitatively measure the

quality of the generations, as well as to analyze the impact of both the VQ-VAE and the DDPM in the result. By looking at table 4.1, which presents the metrics for the various datasets, it is possible to notice how the reconstructed test set FID score is much higher than the one of the original test set. This confirms the fact that the autoencoder contributes severely to the quality loss. On the other hand, the generated dataset obtained a very similar FID score to the reconstructed one, meaning that the DDPM is working well. As expected, the VQ-VAE does not contribute to a change in the diversity of the datasets, as proved by the MS-SSIM of the reconstructed test set being close to the one of real data. Conversely, the generated dataset losed diversity, as shown by its higher MS-SSIM. These results have been compared to the ones obtained in similar works on LDMs trained on 3D medical images. Pinaya et al. [11] achieved better scores in both metrics: the MS-SSIMs of real and synthetic data are close and the FID score is very low. However, this could be due to the difference in the cardinality of the datasets used to train the models (909 vs 31740). Differently, the MS-SSIM obtained in this work is comparable to the one of Kahder et al. [12], with which we share the cardinality of the used dataset.

	MS-SSIM	FID
Training set	0.5232	0
Test set	0.5236	0.0003
Reconstructed test set	0.5244	0.0158
Generated dataset	0.5435	0.0195
Distilled top-k	0.6073	0.0111
Distilled corr threshold	0.5448	0.0323
Distilled LR threshold	0.6045	0.0132
Filtered Latents	0.5738	0.0101
Pinaya et al. real	0.6536	0.0005
Pinaya et al. generated	0.6555	0.0076
Khader et al. real	0.8095	-
Khader et al. generated	0.8557	-

Table 4.1: Quantitative evaluation of the quality of synthetic images via MS-SSIM and FID scores. The results are compared to the ones of similar studies.

In order to better understand the influence of the distribution of the quality scores in the results, both metrics were also calculated separately for each QS. The results in table 4.2 show that, in fact, QS 6 skulls have suffered from their small cardinality, obtaining a much higher FID score with respect to the other quality scores. This is true for the reconstructed test set, but particularly relevant in the generated dataset. This also confirms the presence of wrongly generated images of that QS. Also the grouped MS-SSIMs shown in table 4.3 highlight the same behaviour: all quality scores register a comparable increase in the value, while QS 6 images, influenced by the wrong generations, obtain a lower value than real data.

	FID	QS2	QS3	QS4	QS5	QS6
Test set	0.0003	0.0003	0.0016	0.0007	0.0014	0.0021
Recon test set	0.0158	0.0145	0.0125	0.0186	0.0201	0.0331
Generated ds	0.0195	0.0121	0.0136	0.0234	0.0168	0.1221
Distilled top-k	0.0111	0.0107	0.0185	0.0098	0.0186	0.0636
Distilled corr	0.0323	0.0158	0.0207	0.0456	0.0334	0.219
Distilled LR	0.0132	0.0118	0.0134	0.0172	0.0195	0.0665
Filtered latents	0.0101	0.0139	0.0147	0.0091	0.0101	0.0276

Table 4.2: FID scores of the datasets grouped by QS.

	MS-SSIM	QS2	QS3	QS4	QS5	QS6
Test set	0.5236	0.6517	0.6261	0.5722	0.615	0.594
Recon test set	0.5244	0.6453	0.6312	0.5703	0.6177	0.611
Generated ds	0.5435	0.7354	0.6614	0.6314	0.6796	0.5602
Distilled top-k	0.6073	0.7921	0.8028	0.6705	0.7627	0.6418
Distilled corr	0.5448	0.7456	0.6388	0.7323	0.7854	0.6951
Distilled LR	0.6045	0.8107	0.7906	0.6696	0.7582	0.6
Filtered latents	0.5738	0.7041	0.7359	0.6394	0.6632	0.6264

Table 4.3: MS-SSIMs of the datasets grouped by QS.

4.2 Memorization assessment

While the previous section was useful to analyze the ability of the model to generate realistic images, it does not provide any insight about the degree of

memorization in the synthetic dataset. In order to do so, the correlation between each generated image and the training set has been computed, and the same has been done for each image in the test set. Also the Lowe's ratio, defined as the ratio between the second highest correlation coefficient and the highest correlation coefficient, has been computed in both scenarios.

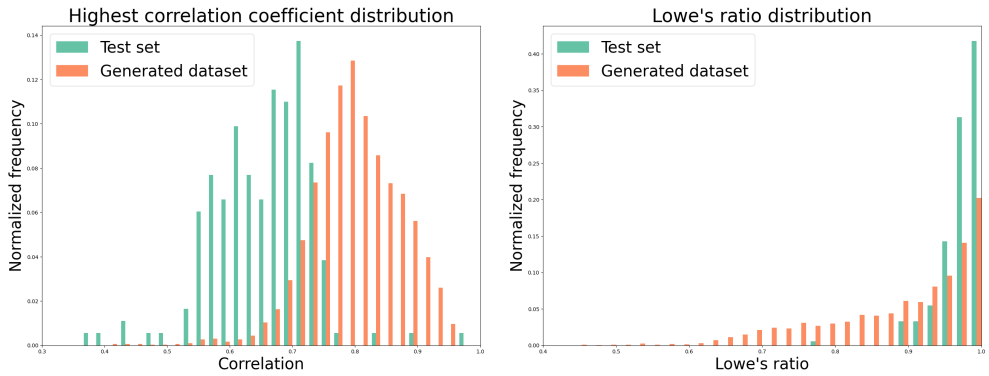


Figure 4.5: Distribution of the highest correlation coefficient and of the Lowe's ratio in the test set and in the generated dataset.

The distribution of the maximum correlation coefficient highlights the fact that the synthetic dataset is much more correlated to the training set with respect to the test set. This means that the model is somewhat memorizing the training data. The same conclusion can be drawn by looking at the distributions of the Lowe's ratio, which show that the synthetic dataset obtains lower values, thus indicating the presence of copies.

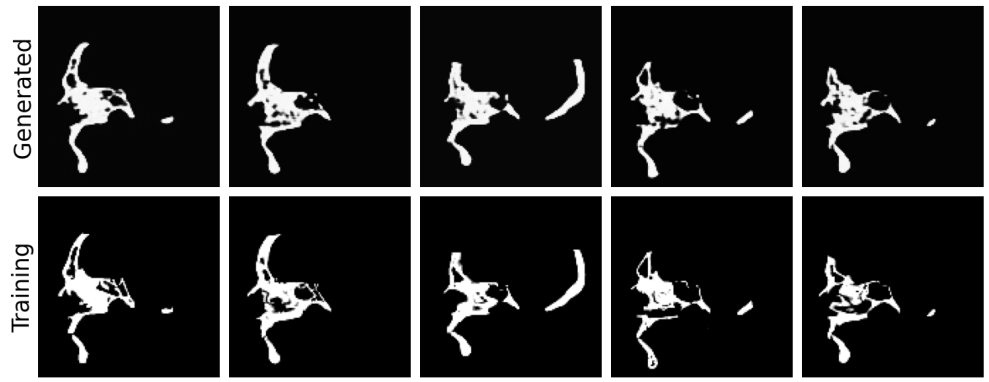


Figure 4.6: Middle slice of the saggital plane of generated QS5 skulls with the corresponding copied training slice.

Image 4.6 shows the QS 5 generated skulls with lowest Lowe's ratio matched with the corresponding copied training images. It is possible to notice how the structure of the synthetic images is perfectly modelled on the real ones. On the other hand, as a consequence of the reconstruction error, the details are not perfectly matched. We can assume that an improvement of the ability of the VQ-VAE would lead to the presence of even more accurate copies.

The degree of memorization has also been quantitatively measured using the Jensen-Shannon divergence (Table 4.4). This has been computed between the distributions of the memorization metrics of the generated dataset and of the test set, setting up a baseline for the evaluation of the effectiveness of the memorization countering techniques.

4.3 Memorization countering

The first experimented way to counter memorization in the generated dataset is Privacy Distillation, meaning training a new DDPM without exposing it to any re-identifiable data. As explained in section 3.5, the key part of this method lies in the strategy used for filtering the generated dataset. In this work, three different techniques have been adopted and all of them aimed at building a new dataset with the same cardinality and QS distribution of the training set. The first one is to rank the synthetic dataset based on descending Lowe's ratios and to take the top-k images. This corresponds to building a filtered dataset containing only the images which are least copied from the training set. The second and third strategies rely on the information provided by the test set and consist in filtering the generated dataset based on a threshold over the highest correlation coefficient and Lowe's ratio respectively. The thresholds have been selected from the quantiles of the values in the test set to remove the outliers of the distributions: the 0.983 quantile of the highest correlation coefficients (0.7640) and the 0.005 quantile of the Lowe's ratios (0.8719). In this case the synthetic dataset has not been sorted and, for each quality score,

the first k images that got through the filtering were kept. Three different DDPMs have been trained with the dataset created from the different filtering strategies and they were subsequently used to generate three distilled datasets with same cardinality and QS distribution of the training set.

Memorization was also countered using Latent Filtering, whose only hyperparameter is M, the number of generated latents among which to select the one that will be decoded, i.e. the farthest from the training latents. In this work this hyperparameter was set to 5 and, as before, this technique was used to generate a new synthetic dataset with the same characteristics of the training set.

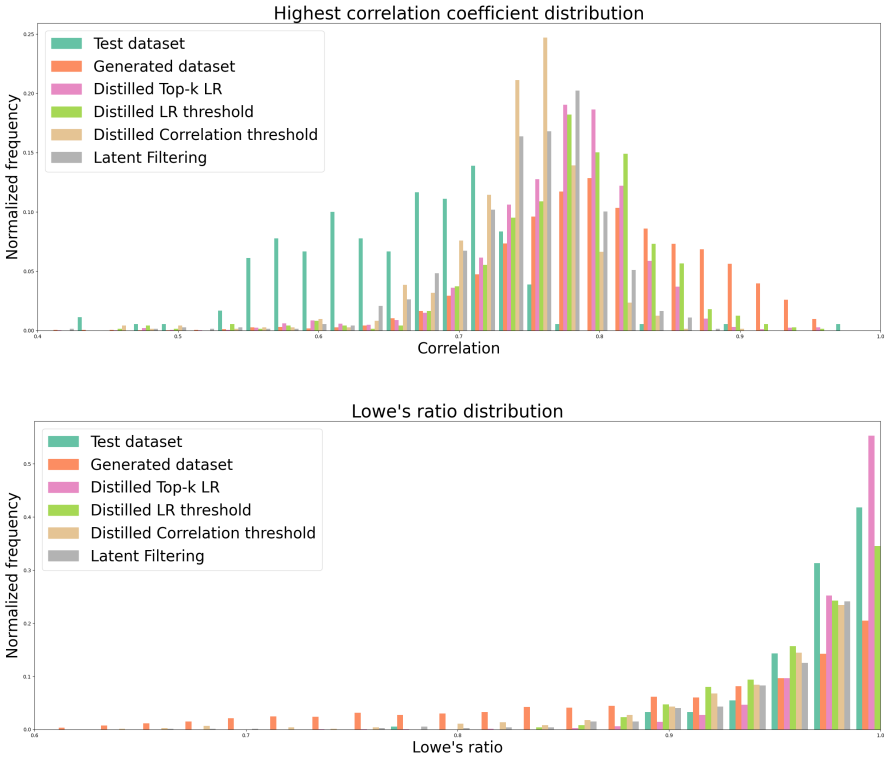


Figure 4.7: Distribution of the highest correlation coefficient and of the Lowe's ratio in the test set and all the synthetic datasets.

Figure 4.7 shows the effect of the different memorization countering techniques in the distribution of the Lowe's ratio and of the highest correlation coefficient. It is possible to appreciate how, independently from the method,

the frequency of higher correlations has dropped. Similarly, the distribution of the Lowe's ratio has become much more alike the one of the test set. Table 4.4 shows how the different methods influenced the degree of memorization. Firstly, it is possible to appreciate how each technique was able to bring the distributions closer to the test set ones. This is particularly visible in the Jensen-Shannon divergence computed on the Lowe's ratio distributions. It is relevant to note how the filtering done directly on the latent space, in term of memorization, was as effective as the distillation techniques. In particular, the results were very similar to the ones obtained by thresholding the highest correlation coefficient.

	JS - HCC	JS - LR
Generated dataset	0.6365	0.4037
Distilled top-k	0.6044	**0.1251**
Distilled corr threshold	**0.5055**	0.2159
Distilled LR threshold	0.6183	0.1631
Filtered Latents	0.5198	0.1759

Table 4.4: Quantitative evaluation of the memorization in the synthetic datasets. It is computed as the Jensen-Shannon divergence between the highest correlation coefficient (or the Lowe's ratio) distributions of a synthetic dataset and of the test set.

The effects in term of generated images realisticity can be analyzed in table 4.1. The distilled datasets gave different results depending on the filtering technique used. When based on Lowe's Ratio values, the effect was a good realisticity at the expense of a lower diversity. Astonishingly, the FID values resulted better than the ones obtained with the originally generated dataset and with the reconstructed test set. This is primarily caused by the better values obtained with QS 4 and QS 6 skulls, as observable in table 4.2. The dataset directly built from the filtered latents obtained very similar results, displaying high realisticity but lower diversity. On the other hand, the distilled dataset based on the correlation filter showed an opposite behaviour: the diversity is higher but the realisticity of the skulls is considerably lower.

Overall, Privacy Distillation (using Lowe's ratio) and Latent Filtering have

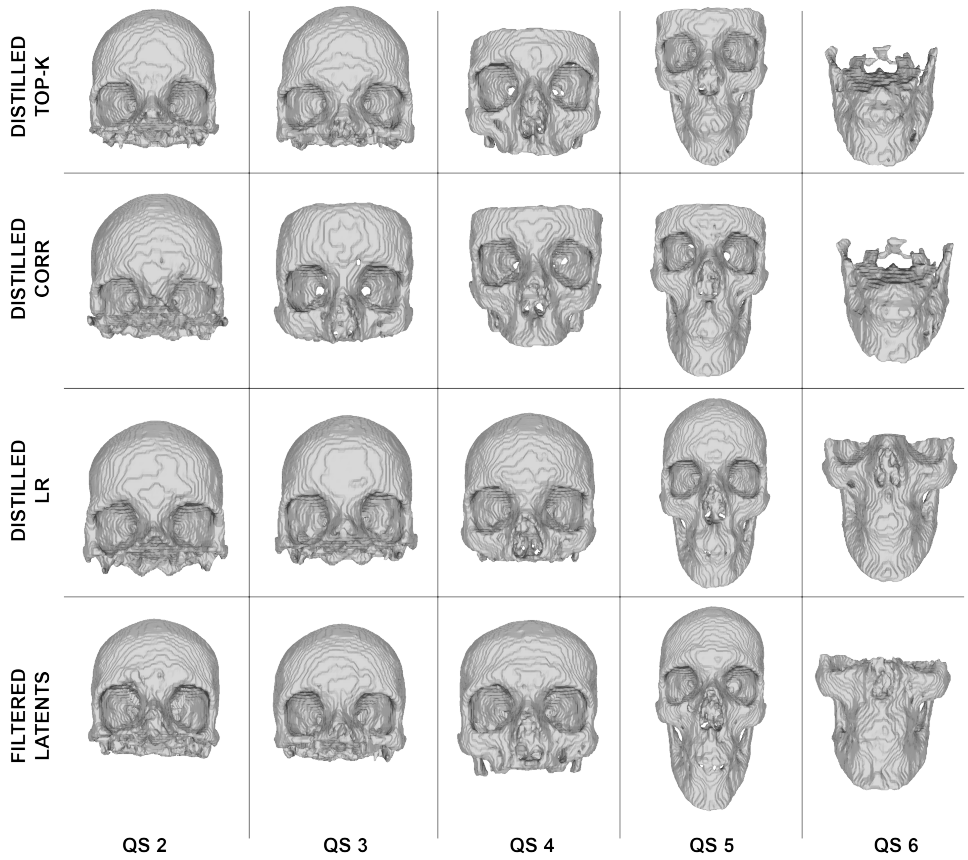


Figure 4.8: Examples of generated skulls after applying different memorization countering techniques.

shown to be equivalently effective both in term of quality and in term of memorization countering. The advantage of using one or the other mainly depends on the subsequent tasks. If the aim is to publish a realistic but privacy preserving dataset, then Latent Filtering is probably the best solution because it can be seamlessly integrated in the generation process. On the other hand, if the goal is to publish a trained model, then Privacy Distillation proves to be more useful because the privacy is enforced at the model weights level.

5. Conclusions

Given the expanded capabilities of generative models, there is a corresponding increase in the interest in using them in the medical domain. This is primarily due to their potential in overcoming privacy related issues, enhancing data availability, and improving the outcomes of subsequent tasks.

This work has confirmed the ability of LDMs to generate realistic 3D medical images which adhere to the anatomical structures of the skeletal parts of the head. The synthetic skulls have displayed a good degree of diversity and the generation, conditioned on the quality score, has proved to be effective. The quantitative analysis has highlighted the different impact of the autoencoder and of the DDPM in the effectiveness of the generation, showing that the first is the main responsible for the quality loss. A possible continuation of this work could involve a fine hyperparameter tuning of the VQ-VAE to improve the scores, or a comparison of different architectural choices for the autoencoder. However, the results have also shown that the primary quality bottleneck is the preprocessing phase and this is due to hardware limitations. The problem can be addressed by using GPUs with larger memory or, more interestingly, by training another model to upsample the images yielded by the LDM via super-resolution [23]. This could be done because the raw data have a very high quality, which is sacrificed because of memory constraints. The size of the training set (727 images) had only a limited influence on the evaluation metrics, which is valuable information in the context of medical tasks, where large datasets are rare. Still, the generation of images of the least represented quality score (QS 6, 38 training examples) was indubitably affected

by the small cardinality of its training set, leading to some wrongly generated images.

This dissertation confirms the presence of memorization and shows how LDMs tend to generate images which are exact copies of training examples. Nonetheless, the experiments done to counter this behaviour were effective. Lowe's ratio as proved to be an effective metric to measure the degree of memorization as well as to decide whether an image is a copy or not. Also when used on 3D images, Privacy Distillation has proved to work well but, at the same time, the experiments have shown the influence of the filtering strategy in the distilled datasets evaluation metrics. Overall, filters based on Lowe's ratios were more effective in term of quality than the one based on highest correlation coefficients. Considering memorization metrics, all the filtering strategies have shown to be able to reduce memorization. The dataset obtained via Latent Filtering achieved comparable results both in term of diversity and in term of realisticity, and it proved to be a valid alternative for countering memorization. The preference of one technique over the other depends on each work's goals: Privacy Distillation is better suited for publishing a model; Latent Filtering is an easier solution for sharing a dataset.

In summary, this work demonstrated the effectiveness of LDMs in generating novel medical data while simultaneously pointing out their flaws. Moreover, in this dissertation the issue of memorization was investigated in depth: it was proposed the use of Lowe's ratio (computed on correlation coefficients) as a novel memorization assessment technique; and two different countering techniques were successfully implemented, being effective in reducing the amount of memorization while, at the same time, not leading to image quality loss.

A. Code Availability

This work is a fork of the repository of Khader et al. [12].
It is available on GitHub at the following link:
`https://github.com/Chavelanda/medicaldiffusion`

B. Hyperparameters

All the models have been trained on a NVIDIA GeForce RTX 3090 Ti with 24GB GPU RAM.

	Value
No. images	909
Image size	128x128x128
VQ-VAE	
Batch size	2
Training steps	150000
Learning rate	3e-4
Embedding dimension k	8
Codebook size n	16384
Compression factor s	4
Training time	20h
DDPM	
Batch size	10
Training steps	100000
Learning rate	1e-4
Timesteps T	300
Training time	20h
Generation time*	80m

Table B.1: Hyperparameters used for training the Latent Diffusion Model

* The time required to generate a dataset of same cardinality as the training set, i.e. to generate 727 images.

Bibliography

[1] T. Han, S. Nebelung, C. Haarburger, N. Horst, S. Reinartz, D. Merhof, F. Kiessling, V. Schulz, and D. Truhn. Breaking medical data sharing boundaries by using synthesized radiographs. *Science Advances*, 6(49):eabb7973, December 2020. ISSN: 2375-2548. DOI: 10.1126/sciadv.abb7973.

[2] A. Kazerouni, E. K. Aghdam, M. Heidari, R. Azad, M. Fayyaz, I. Hacihaliloglu, and D. Merhof. Diffusion models in medical imaging: a comprehensive survey. *Medical Image Analysis*, 88:102846, August 1, 2023. ISSN: 1361-8415. DOI: 10.1016/j.media.2023.102846.

[3] H.-C. Shin, N. A. Tenenholtz, J. K. Rogers, C. G. Schwarz, M. L. Senjem, J. L. Gunter, K. Andriole, and M. Michalski. Medical image synthesis for data augmentation and anonymization using generative adversarial networks, September 13, 2018. DOI: 10.48550/arXiv.1807.10225. arXiv: 1807.10225[cs,stat].

[4] A. S. Coyner, J. S. Chen, K. Chang, P. Singh, S. Ostmo, R. V. P. Chan, M. F. Chiang, J. Kalpathy-Cramer, and J. P. Campbell. Synthetic medical images for robust, privacy-preserving training of artificial intelligence: application to retinopathy of prematurity diagnosis. *Ophthalmology Science*, 2(2):100126, June 1, 2022. ISSN: 2666-9145. DOI: 10.1016/j.xops.2022.100126.

[5] S. U. H. Dar, M. Seyfarth, J. Kahmann, I. Ayx, T. Papavassiliu, S. O. Schoenberg, and S. Engelhardt. Unconditional latent diffusion models

memorize patient imaging data, February 1, 2024. DOI: 10.48550/arXiv.2402.01054. arXiv: 2402.01054[cs,eess].

[6] V. Fernandez, P. Sanchez, W. H. L. Pinaya, G. Jacenków, S. A. Tsaftaris, and J. Cardoso. Privacy distillation: reducing re-identification risk of multimodal diffusion models, June 2, 2023. DOI: 10.48550/arXiv.2306.01322. arXiv: 2306.01322[cs].

[7] J. Ho, A. Jain, and P. Abbeel. Denoising diffusion probabilistic models, December 16, 2020. DOI: 10.48550/arXiv.2006.11239. arXiv: 2006.11239[cs,stat].

[8] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-resolution image synthesis with latent diffusion models, April 13, 2022. DOI: 10.48550/arXiv.2112.10752. arXiv: 2112.10752[cs].

[9] A. v. d. Oord, O. Vinyals, and K. Kavukcuoglu. Neural discrete representation learning, May 30, 2018. DOI: 10.48550/arXiv.1711.00937. arXiv: 1711.00937[cs].

[10] A. Razavi, A. v. d. Oord, and O. Vinyals. Generating diverse high-fidelity images with VQ-VAE-2, June 2, 2019. DOI: 10.48550/arXiv.1906.00446. arXiv: 1906.00446[cs,stat].

[11] W. H. L. Pinaya, P.-D. Tudosiu, J. Dafflon, P. F. da Costa, V. Fernandez, P. Nachev, S. Ourselin, and M. J. Cardoso. Brain imaging generation with latent diffusion models, September 15, 2022. DOI: 10.48550/arXiv.2209.07162. arXiv: 2209.07162[cs,eess,q-bio].

[12] F. Khader, G. Müller-Franzes, S. Tayebi Arasteh, T. Han, C. Haarburger, M. Schulze-Hagen, P. Schad, S. Engelhardt, B. Baeßler, S. Foersch, J. Stegmaier, C. Kuhl, S. Nebelung, J. N. Kather, and D. Truhn. Denoising diffusion probabilistic models for 3d medical image generation. *Scientific Reports*, 13(1):7303, May 5, 2023. ISSN: 2045-2322. DOI: 10.1038/s41598-023-34341-2.

[13] G. Somepalli, V. Singla, M. Goldblum, J. Geiping, and T. Goldstein. Understanding and mitigating copying in diffusion models, May 31, 2023. DOI: 10.48550/arXiv.2305.20086. arXiv: 2305.20086[cs].

[14] X. Gu, C. Du, T. Pang, C. Li, M. Lin, and Y. Wang. On memorization in diffusion models, October 4, 2023. arXiv: 2310.02664[cs].

[15] M. U. Akbar, W. Wang, and A. Eklund. Beware of diffusion models for synthesizing medical images – a comparison with GANs in terms of memorizing brain tumor images, May 12, 2023. DOI: 10.48550/arXiv.2305.07644. arXiv: 2305.07644[cs,eess].

[16] S. U. H. Dar, A. Ghanaat, J. Kahmann, I. Ayx, T. Papavassiliu, S. O. Schoenberg, and S. Engelhardt. Investigating data memorization in 3d latent diffusion models for medical image synthesis, July 6, 2023. DOI: 10.48550/arXiv.2307.01148. arXiv: 2307.01148[cs,eess].

[17] S. Chilamkurthy, R. Ghosh, S. Tanamala, M. Biviji, N. G. Campeau, V. K. Venugopal, V. Mahajan, P. Rao, and P. Warier. Development and validation of deep learning algorithms for detection of critical findings in head CT scans, April 12, 2018. DOI: 10.48550/arXiv.1803.05854. arXiv: 1803.05854[cs].

[18] W. E. Lorensen and H. E. Cline. Marching cubes: a high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.*, 21(4):163–169, August 1987. ISSN: 0097-8930. DOI: 10.1145/37402.37422.

[19] O. Ronneberger, P. Fischer, and T. Brox. U-net: convolutional networks for biomedical image segmentation, May 18, 2015. DOI: 10.48550/arXiv.1505.04597. arXiv: 1505.04597[cs].

[20] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter. GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[21] S. Chen, K. Ma, and Y. Zheng. Med3d: transfer learning for 3d medical image analysis, July 17, 2019. DOI: 10.48550/arXiv.1904.00625. arXiv: 1904.00625[cs].

[22] Z. Wang, E. Simoncelli, and A. Bovik. Multiscale structural similarity for image quality assessment. In *The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*. The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003, volume 2, 1398–1402 Vol.2, November 2003. DOI: 10.1109/ACSSC.2003.1292216.

[23] H. Yang, Z. Wang, X. Liu, C. Li, J. Xin, and Z. Wang. Deep learning in medical image super resolution: a review. *Applied Intelligence*, 53(18):20891–20916, September 1, 2023. ISSN: 1573-7497. DOI: 10.1007/s10489-023-04566-9.