Université de Montréal - MILA - Optina Diagnostics Inc.

Generating Hyperspectral Retinal Images with Latent Diffusion Models

by

Arthur Boschet

Computer Science Department
Faculty of Arts and Sciences

Internship report for
Master's of Science (M.Sc.)
in Computer Science

Specialization: Machine Learning

January 15, 2024

Student and Internship Details

Student Name	Arthur Boschet
Student Number	20241359
Student Email	arthur.boschet@umontreal.ca
Project Title	Generating Hyperspectral Retinal Images with Latent Diffusion Models
Company Name	Optina Diagnostics Inc.
Company Website	https://optinadx.com/
Internship Supervisor	Jean-Sébastien Grondin
Internship Supervisor Email	jsgrondin@optinadx.com

Abstract

This report discusses an internship project at Optina Diagnostics focused on the generation of high-resolution, synthetic hyperspectral retinal images. The primary aim is to enhance self-supervised learning in retinal ophthalmology, addressing the challenge of labeled data scarcity. By augmenting Optina's dataset of both labeled and unlabeled hyperspectral retinal images with these synthetic images, we aim to improve the efficacy of the Momentum Contrast (MoCo) pretraining pipeline. This approach is anticipated to enrich the self-supervised learning process, leveraging the full potential of the dataset. The ultimate objective is the improved classification of amyloid-β in retinal images, a biomarker linked to Alzheimer's disease, thus advancing Optina's overarching goals in medical imaging and disease detection.

The project used latent diffusion models (LDMs), inspired by 'Medical Diffusion' by Khader et al. It began with lower-resolution images (224 x 224 x 16) due to computational limits, later progressing to higher resolutions (672 x 672 x 16). The study meticulously evaluated the synthetic images' quality, diversity, and authenticity. Spatial quality was measured using a modified hyperspectral Fréchet Inception Distance (FID) score, and spectral quality through Kullback-Leibler (KL) divergence, focusing on key ocular structures. Image diversity was assessed with the Multi-Scale Structural Similarity Index (MS-SSIM), and authenticity by comparing synthetic images to their nearest neighbors in the training set.

A key part of the research was comparing VQ-GAN and VQ-VAE autoencoders within the LDMs, with VQ-GAN showing better performance. This led to a detailed hyperparameter tuning of the VQ-GAN-based LDM. The best configuration had a compression factor of 8, an 8-channel latent space, and a 1024-vector codebook. The use of larger diffusion U-Nets also improved performance. For larger image formats, the model transitioned to instance normalization over batch normalization due to computational constraints. The high performance of the optimized model was evident from a hyperspectral FID score of 6.086 and an MS-SSIM score of 0.468.

This project marks a leap in self-supervised learning at Optina Diagnostics, demonstrating the potential of synthetic images in enhancing amyloid-β classification algorithms and inspiring future advancements in medical imaging.

Contents

List of tables

List of figures

List of acronyms

CI/CD	*Continuous Integration and Continuous Delivery.*
ELBO	*Evidence Lower Bound*
FID	*Fréchet Inception Distance.*
GAN	*Generative Adversarial Network*
HIPAA	*Health Insurance Portability and Accountability Act*
KL	*Kullback-Leibler.*
LDM	*Latent Diffusion Model.*
MoCo	*Momentum Contrast.*
MS-SSIM	*Multi-Scale Structural Similarity Index Measure.*
ONH	*Optical Nerve Head*
SVM	*Support Vector Machine*
VAE	*Variational Autoencoder*
VQ-GAN	*Vector Quantized Generative Adversarial Network.*

VQ-VAE *Vector Quantized Variational Autoencoder.*

Acknowledgements

I am profoundly grateful for the support and guidance I received during my internship, which was instrumental in the successful completion of this project.

First and foremost, I extend my deepest appreciation to Jean-Sébastien Grondin, the AI Research Scientist at Optina Diagnostics and my industry supervisor. His exceptional leadership and mentorship were crucial in guiding me through the complexities of this project. Jean-Sébastien's unwavering enthusiasm and extensive expertise in the field of deep learning have not only been a constant source of inspiration but have also significantly enriched my learning experience.

Special thanks are extended to Adam Ibrahim, my technical supervisor at MILA. His insightful feedback and deep understanding of the subject matter have been invaluable. Adam's guidance was crucial in steering this project towards its successful outcome, and his support has been a cornerstone of my learning journey.

I would also like to express my gratitude to my colleagues at Optina Diagnostics – Adrian Tousignant, Anudeep Konda, Wei You and Claudia Chevrefils. Working alongside them in the AI team was not only a professionally rewarding experience but also immensely enjoyable. Their camaraderie and the collaborative atmosphere they fostered contributed significantly to a positive and productive internship environment.

This journey has been an amalgamation of learning, growth, and collaboration, and it would not have been possible without the collective effort and support of everyone mentioned. I am sincerely thankful to each one of them for their contributions to my professional development and for making this internship a memorable and valuable experience.

Chapter 1

Introduction

In this section, we provide the background needed to understand the research internship on the generation of hyperspectral images taking place in the summer of 2023. We describe Optina Diagnostics, the company hosting the internship, including its role in the field and the techniques it currently uses for analyzing retinal images. We also give an overview of the internship, explaining its goals and how it fits into Optina's AI research ecosystem.

1.1. Company Description

Founded in March 2012, Optina Diagnostics is a Montreal-based medical technology firm that specializes in early-stage diagnosis of dementia and other neurological disorders. Central to the company's innovative diagnostic methodology is its Retinal Deep Phenotyping platform, empowered by its proprietary Optina-4C^{TM} hyperspectral camera. The camera facilitates a rapid, non-invasive, and highly detailed examination of retinal physiology through capturing images at 91 distinct wavelengths. The company has received FDA Breakthrough Device designation for its platform, acquired 510(k) clearance for its hyperspectral camera, and successfully raised $25.2 million USD in funding while expanding its team to over 40 members. In December 2021, a pivotal study was initiated to specifically assess the effectiveness of the Retinal Deep Phenotyping platform in diagnosing Alzheimer's Disease.

1.2. Current Methodologies and Challenges

Within the context of Optina Diagnostics, contemporary methods for predicting Alzheimer's from retinal images have primarily been based on feature extraction and ensemble classifiers such as random forests and XGBoost. These methods have been developed and evaluated in-house to address the unique challenges posed by Optina's specialized datasets. Concurrently, deep learning techniques are also under exploration within the company. It is hypothesized that these deep learning techniques may eventually outperform the traditional methods currently in use, although this is not guaranteed and remains an area of active investigation. An impending influx of unlabeled images is set to arrive soon, presenting a significant opportunity for deep learning algorithms. Specifically, algorithms that employ self-supervised learning techniques stand to benefit by extracting robust and meaningful features from this new data. As such, Optina's Deep Learning division embarked on a self-supervised learning project that uses Momentum Contrast (MoCo) [**23**]. To enhance the self-supervised learning project, Optina's AI research team initiated an internship focused on exploring generative methods for advanced data augmentation using synthetic images.

1.3. Internship Deliverables and Requirements

The primary objective of this internship is to develop a robust pipeline for generating and evaluating synthetic hyperspectral retinal images, with the aim of enhancing the company's self-supervised learning process. To align seamlessly with Optina Diagnostics' research and development goals, the internship features a well-defined set of deliverables and requirements, as outlined below:

- Develop and implement a robust pipeline for training and evaluation of a hyperspectral retinal image generation model. Assessing the quality of these synthetic images is both challenging and essential for the project's success.
- Produce clean, maintainable code and make use of CI/CD tools to ensure the developed pipeline is scalable and integrates smoothly into Optina's existing infrastructure.
- Clearly communicate analytical findings and technical solutions to internal and external stakeholders, thereby informing future research avenues.
- Engage in active collaboration with the AI research team, sharing knowledge and technologies to promote the project's comprehensive success.
- Conduct systematic analytical experiments and thoroughly document the findings to facilitate Optina's future research.

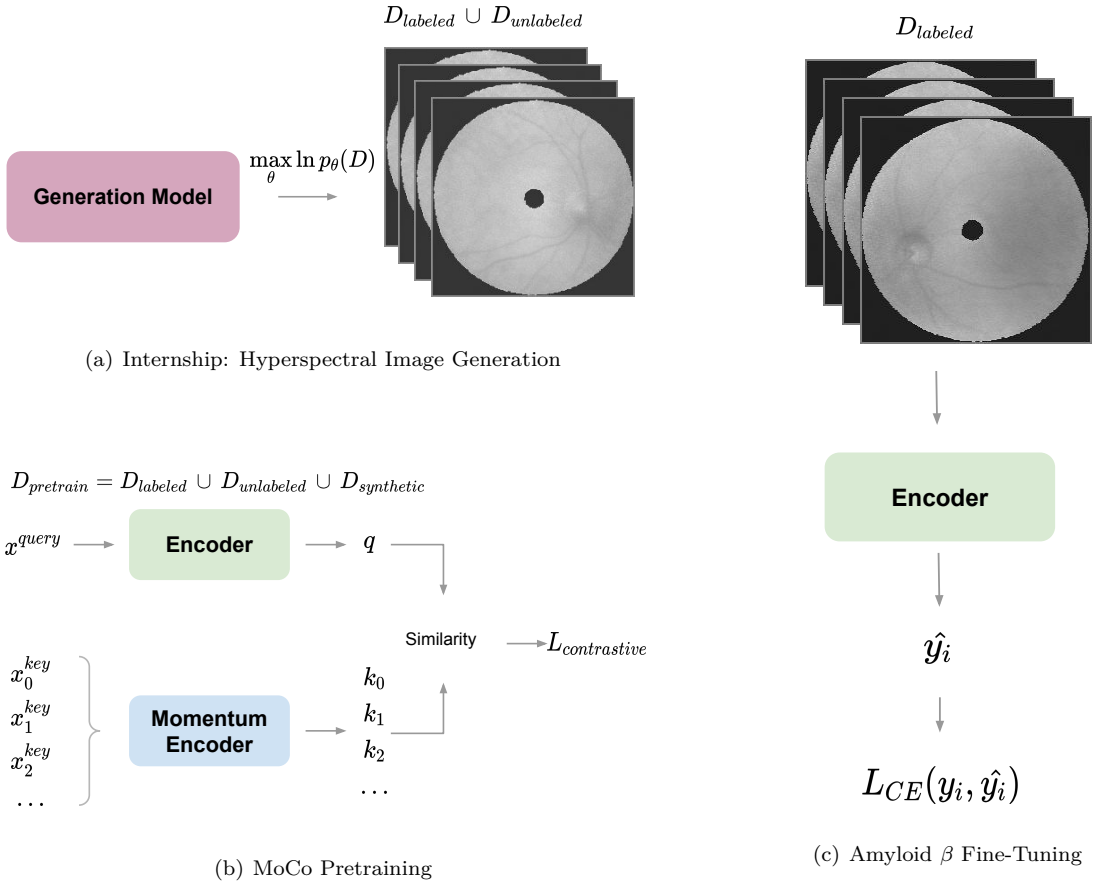
In summary, these requirements are designed to ensure not only the successful completion of the internship project but also the reproducibility and reusability of the work by Optina's research team, facilitating a seamless transfer of knowledge upon the project's completion.

1.4. Internship's Impact on Optina's Self-Supervised Learning Project

In line with Optina Diagnostics' overarching research objectives, the internship occupies a pivotal role, with its primary focus on the generation of synthetic hyperspectral retinal images. These synthetic images are a crucial component of a more extensive research project that utilizes the Momentum Contrast (MoCo) self-supervised learning algorithm to achieve effective feature extraction from unlabeled retinal images [**23**]. Optina Diagnostics has accumulated an extensive database of labeled hyperspectral retinal images dedicated to research purposes. Moreover, there is an anticipated influx of unlabeled retinal images, emphasizing the urgency and opportunity in developing strategies that maximize the utility of both labeled and unlabeled datasets.

The central aim of the internship is to facilitate the development of a pipeline for generating synthetic hyperspectral images, which are intended to significantly bolster the MoCo pretraining phase. These artificially generated images, while unlabeled, are designed to serve as advanced data augmentations. By incorporating these synthetic images into the self-supervised learning framework, the objective is to enhance the model's ability for effective feature capture during the MoCo pretraining stage. This serves as an essential preparatory step before initiating the fine-tuning phase of the project.

Once the MoCo pretraining phase has laid a robust foundation, the model will transition to the fine-tuning stage. During this stage, the pretrained model undergoes further optimization, focusing exclusively on the labeled dataset to develop a specialized classifier. The ultimate goal of this classifier is to provide a predictive measure that is correlated with the aggregation of amyloid β, a well-established biomarker for Alzheimer's Disease. The comprehensive research initiative, which integrates retinal image synthesis, Momentum Contrast (MoCo) pretraining, and model fine-tuning, is depicted in Figure 1.1.



(a) Internship: Hyperspectral Image Generation

(b) MoCo Pretraining

(c) Amyloid β Fine-Tuning

Fig. 1.1. Overview of the research project aimed at leveraging unlabeled hyperspectral retinal images within the deep learning pipeline. (a) Focus of the internship: this first step involves generating synthetic hyperspectral images using both unlabeled and labeled retinal images. (b) In the second step, Momentum Contrast (MoCo) pretraining is performed on a neural network encoder, utilizing the unlabeled images along with the labeled ones and the synthetic hyperspectral images as advanced augmentations. (c) The final step consists of fine-tuning the pretrained neural network encoder specifically for the task of Amyloid β biomarker detection

Chapter 2

Data

The retina serves as an invaluable data source due to its role as a component of the central nervous system (CNS). Optina's hyperspectral camera captures a 3D image from each scan, represented by a high-resolution 1288x1288 matrix covering 91 wavelengths, ranging from 450 nm to 900 nm. The resulting hyperspectral datasets offer the potential to reveal subtle phenotypic changes associated with neurological and systemic diseases.

Upon the initiation of the internship, Optina had assembled a development dataset consisting of 1427 volumes from 532 patients. Following quality control by Optina's data science team, the dataset was reduced to 1374 images. This dataset was partitioned into four folds: three for training and validation and one for testing. Additionally, a dataset acquired from the ongoing pivotal study is earmarked as the final test set but was not used during the internship due to its designation for FDA approval procedures. Furthermore, an auxiliary dataset comprised of unlabeled images was anticipated to be acquired at the end of the internship period, underscoring the imperative for developing robust techniques for unsupervised or self-supervised feature extraction.

During the internship, given the constraints of a small dataset, the substantial data requirements of generative algorithms, and the high computational costs involved, along with the distinct nature of evaluation metrics for generative models compared to those for classification, a decision was made to utilize all three training subsets for model training. Consequently, this approach allocated 1035 images for training and 339 images for validation. To ensure a testing dataset devoid of data leakage, it was resolved that newly acquired images by Optina in the future would be used as a test dataset.

Figure 2.1 showcases six images from Optina's annotated dataset. In compliance with the Health Insurance Portability and Accountability Act of 1996 (HIPAA) of the United States, which restricts the dissemination of sensitive health information, both patient identifiers and labels have been obfuscated and the images have been resized from an original resolution of 1288×1288 pixels to 672×672 pixels. Notwithstanding these limitations, the images clearly illustrate a wealth of data. For instance, shorter wavelengths, particularly around 555 nm, reveal a pronounced contrast between vascular structures and the surrounding tissue. In the spectral interval spanning 640 nm to 730 nm, specific images, most notably the forth image from the left, manifest a corrugated morphological pattern attributable to the underlying choroidal tissue. This visibility is particularly pronounced in patients with lighter eye coloration, a phenomenon ascribed to reduced melanin concentration within the choroidal layer [**34**]. Given these intricate hyperspectral characteristics, it is postulated that these images may serve as efficacious biomarkers for Alzheimer's disease and other neurological disorders.

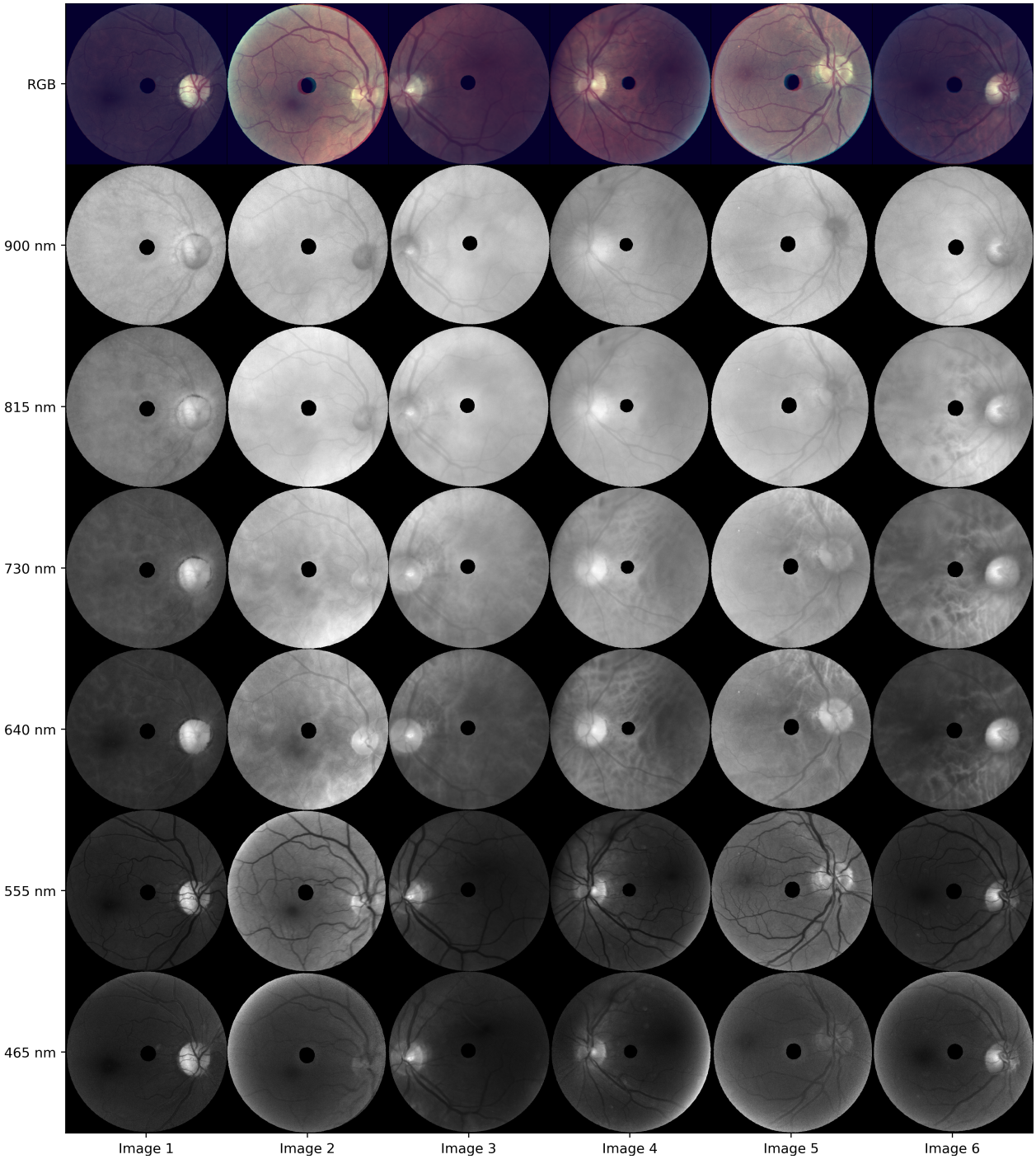


Fig. 2.1. Displayed from left to right are six illustrative images from Optina's dataset, each resized to a spatial resolution of 672×672 pixels. The images are organized to highlight the extraction of spectral data across eight specific wavelengths, enumerated from top to bottom as follows: 900 nm, 815 nm, 730 nm, 640 nm, 555 nm, 465 nm.

Chapter 3

Literature Review

This literature review comprehensively explores the foundational concepts and progressive advancements that have led to the development of latent diffusion models (LDMs), the focal model of this internship as detailed in Chapter 4. The genesis and evolution of LDMs are intrinsically linked to broader developments in the field of deep generative models, necessitating a thorough examination of their theoretical underpinnings and practical applications.

Moreover, this review extends to critically analyze various methodologies employed in the evaluation of generative models. Evaluating the performance of such models, particularly in the context of unique datasets like the one provided by Optina, remains a challenge in contemporary research. It is an area marked by ongoing debate and innovation, making it a pivotal aspect of this internship. This section, therefore, not only delves into the technical aspects and capabilities of LDMs but also places significant emphasis on the assessment criteria and metrics that are crucial for optimizing these models effectively.

3.1. Foundations of Latent Diffusion Models for Image Synthesis

Generative modeling is a significant subfield within unsupervised machine learning and statistics. Its main goal is to understand the inherent distribution of observed data and then use this knowledge to produce new, unique data points. Mathematically, this can be expressed the maximization of the marginal likelihood as:

$$\theta^* = \arg\max_{\theta} \mathbb{E}_{x \sim P_{real}} \left[\log P_\theta(x) \right] \tag{3.1.1}$$

where θ represents the model parameters. However, applying this to real-world data, especially images, brings notable challenges. The complexities of high-dimensional data are well-explained by the manifold hypothesis [**4**]. For example, retinal images might exist in a space of millions of dimensions, but key features—such as background textures or blood vessels—are often captured in just a few dimensions.

The manifold hypothesis highlights both opportunities and challenges. On one side, it suggests that the important information in an image isn't directly linked to its high dimensionality, allowing us to potentially create efficient models that grasp this essential data. On the flip side, this compact data representation can make traditional statistical modeling methods ineffective [**45**].

Fortunately, the latest advancements in generative deep learning, boosted by increasing data availability and computational power, are showing great promise in handling these intricate data distributions. This has led to better-quality generated samples over time. As such, we delve into the basic principles behind deep generative modeling techniques that serve as the foundations of LDMs.

3.1.1. Variational Autoencoders

The manifold hypothesis posits that real-world data, such as images, inhabit a lower-dimensional subspace within an expansive high-dimensional space. A demonstration of this is the unlikelihood of producing a coherent image when drawing samples from a standard normal distribution; instead, one is more likely to obtain Gaussian noise.

Given this, a compelling approach is to generate images from a basic prior $p(z)$ using a neural network parameterized as $p_\theta(z|x)$. For a deeper understanding, we explore the decomposition of $\log p_\theta(x)$. The comprehensive derivation is detailed by Kingma et al. [**39**]:

$$\log p_\theta(x) = \underbrace{\underbrace{\mathbb{E}_{\underbrace{z \sim q_\phi(z|x)}_{\text{Approx. Posterior}}} \left[\log \underbrace{p_\theta(x|z)}_{\text{Likelihood}}\right]}_{\text{Reconstruction Term}} - \underbrace{D_{KL}\left(\underbrace{q_\phi(z|x)}_{\text{Approx. Posterior}} \parallel \underbrace{p(z)}_{\text{Prior}}\right)}_{\text{Regularization Term}}}_{\text{ELBO}} + \underbrace{D_{KL}\left(\underbrace{q_\phi(z|x)}_{\text{Approx. Posterior}} \parallel \underbrace{p_\theta(z|x)}_{\text{Intractable}}\right)}_{\geq 0} \tag{3.1.2}$$

Here, $q_\phi(z|x)$ is the approximate posterior aiming to represent $p_\theta(z|x)$. The Kullback–Leibler (KL) divergence, denoted as D_{KL}, quantifies the dissimilarity between two distributions. Specifically, it measures the divergence of a data distribution $p(z)$ from a reference distribution $q(z)$:

$$D_{KL}(p(z) \parallel q(z)) = \int_{z \in Z} p(z) \log\left(\frac{p(z)}{q(z)}\right) dz \tag{3.1.3}$$

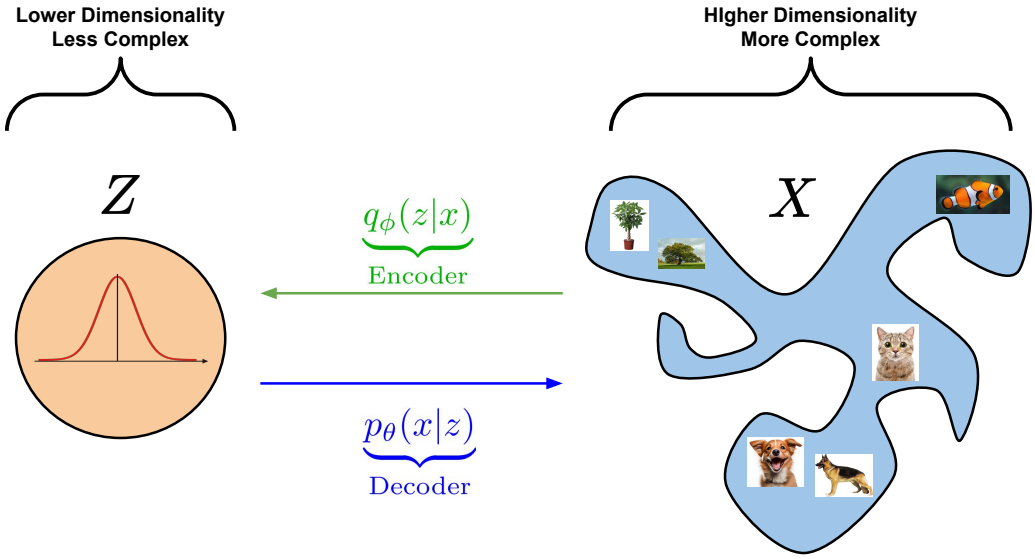


Fig. 3.1. A Variational Autoencoder (VAE) is designed to establish stochastic mappings between an observed data space, represented by x, and a latent space, denoted by z. While the empirical distribution of the observed space, $p(x)$, is often intricate, the latent space's distribution can be comparatively simpler, such as a gaussian distribution. The generative aspect of the VAE learns a joint distribution, $p_\theta(x, z)$. Typically, though not exclusively, this joint distribution is factorized as $p_\theta(x, z) = p(z)p_\theta(x|z)$, comprising a prior distribution over the latent space, $p(z)$, and a conditional decoder distribution, $p_\theta(x|z)$. Concurrently, the stochastic encoder, $q_\phi(z|x)$, approximates the true but computationally intractable posterior $p_\theta(z|x)$ of the generative model.

In an ideal scenario, the decomposition of $\log p_\theta(x)$ would involve setting the approximate posterior $q_\phi(z|x)$ to be the same as $p_\theta(z|x)$. This would facilitate the direct computation of the log-likelihood $\log p_\theta(x)$:

$$\log p_\theta(x) = \underbrace{\mathbb{E}_{\underbrace{p_\theta(z|x)}_{\text{Intractable}}}\left[\log \underbrace{p_\theta(x|z)}_{\text{Likelihood}}\right]}_{\text{Reconstruction Term}} - \underbrace{D_{KL}\left(\underbrace{p_\theta(z|x)}_{\text{Intractable}} \parallel \underbrace{p(z)}_{\text{Prior}}\right)}_{\text{Regularization Term}} \tag{3.1.4}$$

However, directly computing the true posterior $p_\theta(z|x) = \frac{p_\theta(x|z)p(z)}{p(x)}$ is computationally infeasible, mainly because the genuine data distribution $p(x)$ is inaccessible. This challenge prompted the introduction of variational inference, as illustrated in equation 3.1.2. By introducing a separate model, $q_\phi(z|x)$, it's possible to approximate the true posterior. Given the inherent positivity and intractability of the KL-divergence between the approximated and true posteriors, there's a reliance on an alternative objective to maximize the likelihood: the Evidence Lower Bound (ELBO) [**32**]:

$$\mathcal{L}_{ELBO} = \underbrace{\mathbb{E}_{\underbrace{z \sim q_\phi(z|x)}_{\text{Approx. Posterior}}}\left[\log \underbrace{p_\theta(x|z)}_{\text{Likelihood}}\right]}_{\text{Reconstruction Term}} \underbrace{- D_{KL}\left(\underbrace{q_\phi(z|x)}_{\text{Approx. Posterior}} \parallel \underbrace{p(z)}_{\text{Prior}}\right)}_{\text{Regularization Term}} \leq \log p_\theta(x) \tag{3.1.5}$$

The ELBO consists of two distinct terms. The first, the reconstruction term, can be interpreted as the mean squared difference, under the assumption that the output of $p_\theta(x|z)$ represents the mean of a normal distribution with an identity covariance. Conversely, if the reconstruction term represents the mean absolute difference, this suggests that the output of $p_\theta(x|z)$ corresponds to the mean of a Laplacian distribution. The second term, the KL-divergence, serves as a regularization term. It encourages the latent distribution to approximate the prior distribution. By designating $q_\phi(z|x)$as the encoder and $p_\theta(x|z)$ as the decoder, we delineate the architecture of the Variational Autoencoder (VAE). This structure was introduced by Kingma and Welling in 2013 [**38**], as illustrated in figure 3.1.

In the formulation of the ELBO, a challenge presents itself: both the KL-divergence and the reconstruction terms necessitate an expectation over values drawn from the encoder. This process is inherently non-differentiable, posing an obstacle to training this type of autoencoder structure. To address this limitation, the VAE paper introduced the reparameterization trick. Central to this technique is the generation of a variable ϵ sampled from a standard normal distribution, $\mathcal{N}(0,I)$ and independent of ϕ. Consequently, the expression

$$z = \mu_\phi(z|x) + \sigma_\phi(z|x) \circ \epsilon \tag{3.1.6}$$

becomes differentiable, facilitating effective training via backpropagation.

A notable concern with the ELBO formulation is its propensity to induce what is termed "component collapse", attributed primarily to the KL-divergence term. In a standard VAE, this term encourages the approximate posterior distribution, $q_\phi(z|x)$, to align closely with the prior distribution $p(z)$, typically a normal distribution. If the KL-divergence approaches zero, it suggests that $q_\phi(z|x)$ mirrors $p(z)$ - a distribution independent of x. Such convergence nullifies the encoder's capacity to retain information about x. An overt manifestation of component collapse is the dormancy of certain dimensions within the latent space, culminating in diminished model capacity, interpretability, and potential degradation in performance. To avert the pitfalls of a near-zero KL-divergence, which could indirectly compromise reconstruction quality, several enhancements to the regularization term have been proposed. One such approach is the "free-bits"

method as presented by Kingma et al. (2016) [**40**]. This method reformulates the regularization term as $\max\{\lambda, D_{KL}(q_\phi(z|x) \parallel p(z))\}$, imposing a penalty when the divergence from the target prior distribution exceeds a predefined threshold λ. An alternative solution is the deterministic warm-up for the regularization term, shown by Sønderby et al. (2016) [**61**], to ameliorate the effects of component collapse.

A pivotal aspect of VAEs is the "variational gap," representing the discrepancy between the true log-likelihood and the Evidence Lower Bound (ELBO). As elucidated by equation 3.1.2, this can be expressed as:

$$\begin{aligned}
\log p_\theta(x) - \mathcal{L}(q_\phi(z|x)) &= D_{KL}(q_\phi(z|x) \parallel p_\theta(z|x)) \\
&= \log p_\theta(x) - \mathcal{L}(q^*) + \mathcal{L}(q^*) - \mathcal{L}(q_\phi(z|x)) \\
&= \underbrace{D_{KL}(q^* \parallel p_\theta(z|x))}_{\text{Approximation Gap}} + \underbrace{\mathcal{L}(q^*) - \mathcal{L}(q_\phi(z|x))}_{\text{Amortization Gap}}
\end{aligned}$$

Here, $q^* = \arg\min_{q \in Q} \log p_\theta(x) - \mathcal{L}(q)$ designates the optimal posterior approximation within the chosen distribution family. The variational gap encompasses both the approximation gap and the amortization gap [**9**]. Specifically, the approximation gap quantifies the KL-divergence between the optimal posterior approximation and the true posterior, given a certain distribution family. Addressing this gap often entails adopting intricate distributions beyond the conventional diagonal Gaussian, thereby enhancing the encoder's output. Furthermore, by infusing the prior distribution with sophisticated parameterizations, such as a diffusion process, we enhance its expressive capability. This refinement allows the approximate posterior distribution to more accurately mirror the true posterior, leading to a reduction in the approximation gap and a more precise latent portrayal of intricate data patterns. A prominent example from literature is Vahdat et al.'s approach, where they employed score-matching diffusion on the latent space and reformulated the ELBO as presented in their study [**68**].

Mitigating the amortization gap, compared to the approximation gap, can be more straightforward. This often involves bolstering the encoder's capacity—by employing a more robust architecture—and optimizing training dynamics. Such strategies and their impact on variational inference have been highlighted in works like Cremer et al. [**9**].

In addition to the aforementioned challenges, traditional VAEs often generate images that appear blurrier compared to state-of-the-art techniques like generative adversarial networks (GANs) or diffusion models. This blurriness arises from the inherent asymmetry in the ELBO formulation, denoted as $\mathcal{L} = \mathbb{E}_{z \sim q_\phi(z|x)}\left[\log \frac{p_\theta(x,z)}{q_\phi(z|x)}\right] = -D_{KL}(q_\phi(z|x) \parallel p_\theta(x,z))$. The ELBO formulation disproportionately penalizes samples that are probable under the approximate posterior $q_\phi(z|x)$ but improbable in the generative model $p_\theta(x,z)$. Conversely, samples highly probable in $p_\theta(x,z)$ but improbable under $q_\phi(z|x)$ do not receive an equivalent penalty. This imbalance leads to increased variance during the generation phase, manifesting as blur in the resultant images [**5**].

3.1.2. Generative Adversarial Networks

Generative Adversarial Networks (GANs) were pioneered by Goodfellow et al. [**16**]. The core insight of this novel training paradigm is to approximate the real image distribution $P_r(x)$ through an adversarial game played between two neural networks: a generator G and a discriminator D. The generator G endeavors to replicate the real distribution, while the discriminator D aims to classify images as either real or synthetic.

The fundamental principle of GANs is illustrated in Figure 3.2. The adversarial game is encapsulated by the following minimax objective:

$$G^* = \arg\min_G \max_D \mathbb{E}_{x\sim p_r(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))] \tag{3.1.7}$$

where $p_z(z)$ denotes the probability distribution of the input noise, which is typically assumed to be normally distributed. This input noise is crucial as the network architecture is deterministic and would otherwise produce identical images. Recalling our discussion regarding VAEs and the generation of intricate images from a simplistic distribution, this can also be perceived as learning this mapping without necessitating the approximation of the posterior distribution via an encoder.

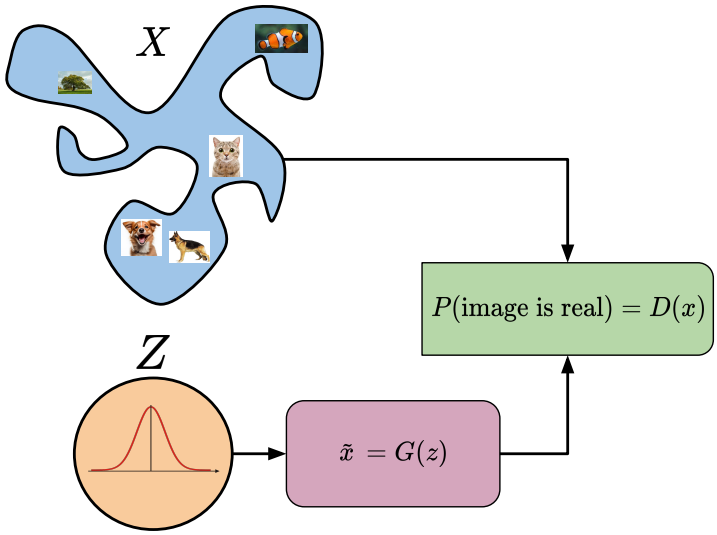


Fig. 3.2. Depiction of the fundamental operational mechanism of GANs. The objective is to train a generator network G to produce realistic images using Gaussian noise z as input. This generator is concurrently trained with a discriminator D in an adversarial framework, wherein G endeavors to deceive D while D strives to distinguish between real and synthetic images.

It can be demonstrated that under the given objective, the optimal discriminator outputs the following probability indicating the realness of an image:

$$D^*(x) = \frac{p_r(x)}{p_r(x) + p_g(x)} \tag{3.1.8}$$

Here, $p_r(x)$ represents the probability density function of real images, while $p_g(x)$ denotes the probability density function of synthetic images. This expression elucidates that, in the case of perfect generation, the ideal discriminator can output a probability no better than 0.5. This serves as an indicator that high discriminator accuracy, signifying ease in distinguishing real and synthetic images, may reflect issues in the generator's learning dynamics possibly due to an inadequate learning signal from the over-performing discriminator [**15**]. Additionally, it can be shown that with an ideal discriminator, the minimization objective aligns with minimizing the Jensen-Shannon divergence (JSD) defined as:

$$JSD(P_r \parallel P_g) = \frac{1}{2}D_{KL}\left(P_r \parallel \frac{1}{2}(P_r + P_g)\right) + \frac{1}{2}D_{KL}\left(P_g \parallel \frac{1}{2}(P_r + P_g)\right) \tag{3.1.9}$$

This demonstrates the theoretical soundness of the minimax objective. However, a challenge arises as the discriminator improves in differentiating images: the gradients of the loss with respect to the generator begin to vanish, obstructing a beneficial learning signal. The literature presents various learning heuristics to address this, starting with a simple modification by Goodfellow et al. in 2014 [**16**]:

$$\max_D \mathbb{E}_{x\sim p_{\mathrm{r}}(x)}[\log D(x)] + \mathbb{E}_{z\sim p_z(z)}[\log(1 - D(G(z)))]$$
$$\max_G \mathbb{E}_{z\sim p_z(z)}[\log(D(G(z)))]$$

This objective, however, may still falter when the discriminator saturates due to the sigmoid activation.

Another technique is Feature Matching, introduced by Salimans et al., which is employed to stabilize the training of GANs [**57**]. Instead of directly optimizing the discriminator's output, it focuses on ensuring that the generator produces data with similar intermediate representations as seen in the real data. The objective function for Feature Matching in GANs can be expressed as:

$$\mathcal{L}_{\mathrm{FM}}(G) = \left\|\mathbb{E}_{x\sim p_{\mathrm{r}}(x)}[f(x)] - \mathbb{E}_{z\sim p_z(z)}[f(G(z))]\right\|_2^2 \tag{3.1.10}$$

where $f(x)$ denotes an intermediate layer representation of data x in the discriminator, $p_{\mathrm{r}}(x)$ is the distribution of real data, $p_z(z)$ is the distribution of noise input, and $G(z)$ is the generated data. This objective aims to minimize the Euclidean distance between the expected features of real and generated data, promoting a generator capable of creating data that aligns with the feature statistics of real data.

A further modification introduced the hinge loss in Geometric GAN [**42**], aiming to enhance training stability and robustness. The modified adversarial objective is defined as:

$$\min_D \mathbb{E}[\max(0, 1 - D(x))] + \mathbb{E}[\max(0, 1 + D(G(z)))]$$
$$\min_G -\mathbb{E}[D(G(z))]$$

Employing a margin-based loss ensures a separation margin between the scores of real and fake samples, reminiscent of the margin maximization principle in Support Vector Machines (SVMs). This margin amplifies the discriminator's robustness towards overlapping regions between real and fake data distributions, providing a more stable gradient for the generator during training. Furthermore, the hinge loss facilitates a clearer separation between real and fake samples, aiding in convergence and the generation of higher-quality samples.

A significant drawback with the JSD minimization objective in the original minimax objective, assuming an ideal discriminator, is its limited informative value for the generator's improvement when real and synthetic distributions scarcely overlap—a plausible scenario in the high-dimensional image space. To address this, Arjovsky et al. introduced the Wasserstein GAN (WGAN) based on the earth-mover distance, or the Wasserstein-1 distance [**3**]. This distance, intuitively, represents the minimal work required to transform one distribution, say synthetic images, into another, say real images. Utilizing the Kantorovich-Rubinstein duality, this distance is expressed as:

$$\begin{aligned} W_1(P, Q) &= \inf_{\gamma\in\Pi(p_r, p_g)} \mathbb{E}_{(x,y)\sim\gamma}\left[\|x - y\|\right] \\ &= \sup_{\|f\|_L\leq 1} \mathbb{E}_{x\sim p_r}[f(x)] - \mathbb{E}_{x\sim p_g}[f(x)] \end{aligned}$$

where $\Pi(p_r, p_g)$ is the set of all distributions over the joint set of images $X \times X$ with marginal distributions $p_r(x)$ and $p_g(x)$, and $\|f\|_L \leq 1$ stipulates that the function must be 1-Lipschitz [**67**]. The definition of a K-Lipschitz function is given by:

$$\frac{\|f(x) - f(y)\|_1}{\|x - y\|_1} \leq K$$

Consequently, the minimax objective of the Wasserstein Generative Adversarial Network (WGAN) can be articulated as:

$$\arg\min_G \max_{\|D\|_L \leq 1} \mathbb{E}_{x \sim p_r}[D(x)] - \mathbb{E}_{z \sim p(z)}[D(G(z))]$$

Herein, the discriminator $D : \mathbb{R}^{\times} \to \mathbb{R}$ is mandated to be a 1-Lipschitz function. To uphold this Lipschitz condition, several strategies have been put forth. These include gradient clipping [**3**], gradient penalty [**20**], and notably, spectral normalization via the power method heuristic [**47**].

GANs exhibit certain drawbacks, among which mode collapse is particularly notable. Mode collapse occurs when the generator produces limited varieties of output, often generating identical or highly similar images regardless of varied input. This phenomenon leads to a decreased diversity in generated data, undermining the generative capability of the network and making GANs less effective compared to other generative methods in capturing the breadth of data distributions [**63**].

Additionally, although advancements such as Feature Matching and various modifications in loss functions have shown promise in enhancing GAN performance, training GANs remains a non-trivial task. The adversarial training process, wherein the generator and discriminator are trained simultaneously, often leads to unstable dynamics. The balance between the two networks is delicate, and achieving convergence requires careful tuning of parameters and model architectures, making the mastery of GANs a demanding endeavor in the field of machine learning [**15**].

GANs have garnered significant interest in medical imaging, aiding in image classification, segmentation, and cross-modal translation [**21, 14, 19, 51**]. Key studies have explored GANs, especially for data augmentation, enhancing medical image generation quality. Han et al. utilized WGAN and DCGAN for generating 2D MRI sequences, finding WGAN superior in visual assessments by physicians. Furthermore, integrating PGGAN with traditional data augmentation slightly improved classifier performance in brain tumor detection [**21, 22**]. Frid-Adar et al. applied ACGAN to augment liver lesions in CT images, which significantly increased the sensitivity and specificity for classifying liver lesions [**14**]. Guibas et al. employed DCGAN and CGAN to create synthetic fundus photographs, which excelled in a blood vessel segmentation task [**19**]. Platscher et al. explored multiple GAN variants to create MRI images with ischemic stroke lesion masks, with Pix2Pix showing the best visual quality and aiding in improving the Dice score when combined with U-Net training [**51**]. Despite challenges like training difficulties and mode collapse hindering image diversity, GANs continue to be invaluable in medical imaging for their ability to generate highly realistic images, as showcased by Saad et al., who addressed intra-class mode collapse, enhancing the diversity of synthetic X-ray images generated with DCGAN [**55**].

3.1.3. Diffusion Models

Since their introduction in 2015 by Sohl-Dickstein et al. through the work "Deep Unsupervised Learning using Nonequilibrium Thermodynamics"[**59**], diffusion models garnered increased attention in 2020, thanks

to Ho et al.'s seminal work on Denoising Diffusion Probabilistic Models (DDPM), which achieved a state-of-the-art FID score on CIFAR10 [**25**]. Fundamentally, diffusion models operate by incrementally adding Gaussian noise to an input image at time t based on the noisy state at $t-1$, and training a model to reverse this process, reconstructing the image at $t-1$ from the noisier state at t. This method is termed a "diffusion process" as the iterative addition of noise gradually obfuscates the original information in the image until, at time T, the image conforms to a normal distribution. The procedure of incrementally introducing noise to the image is termed the forward diffusion process, while the sequence of reconstructive steps from the noisy input is referred to as the reverse diffusion process.

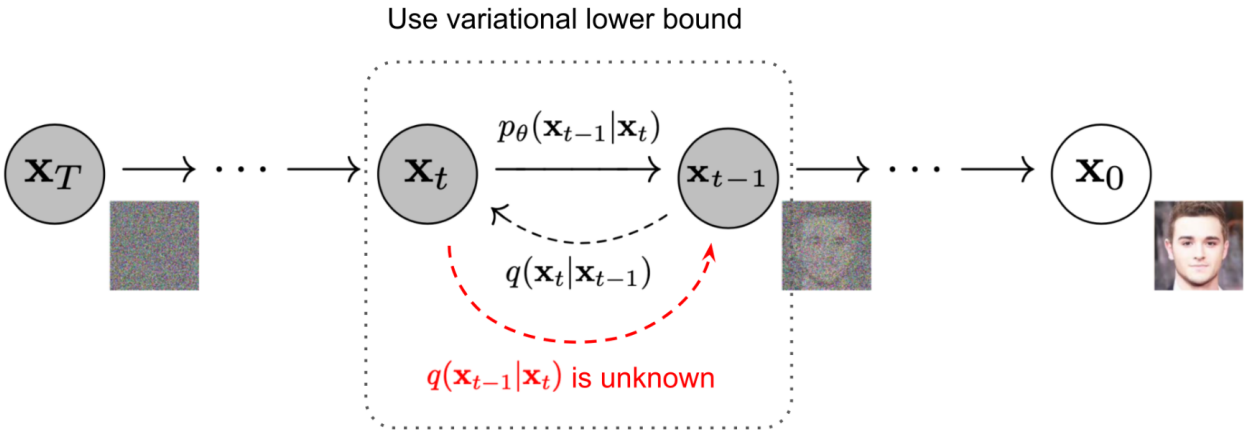


Fig. 3.3. Illustration of the Denoising Diffusion Probabilistic Model (DDPM) adapted from Lilian Weng's blog [**74**], which in turn, introduced minor modifications to the original figure by Ho et al [**25**] from the seminal 2020 DDPM paper. As depicted, given an initial image x_0, Gaussian noise is incrementally introduced through a predetermined normal distribution $q(x_t|x_{t-1})$ as delineated in equation 3.1.11. A parameterized model $p_\theta(x_{t-1}|x_t)$ is trained to systematically reduce noise at each temporal step. Due to the intractability of $q(x_{t-1}|x_t)$, reliance on the variational lower bound is imperative for training the model p_θ.

From a theoretical standpoint, diffusion models can be construed as hierarchical VAEs possessing a total of T sequential latent spaces. Nevertheless, several distinctions are apparent: firstly, the dimensionality of these latent spaces mirrors that of the original image; secondly, the latent encoder is not subject to learning as it is represented by a predefined linear Gaussian model. This implies that it is a Gaussian distribution with its mean centered around the preceding step [**43, 59**].

Formally, the forward process at each time step t is articulated by employing a variance schedule $\{\beta_t \in (0,1)\}_{t=1}^{T}$, and is defined as follows:

$$\begin{aligned} q(x_{1:T}|x_0) &= \prod_{t=1}^{T} q(x_t|x_{t-1}), \\ q(x_t|x_{t-1}) &= \mathcal{N}\left(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t \mathrm{I}\right), \end{aligned} \tag{3.1.11}$$

where $x_{1:T}$ denotes the series of latent variables from time 1 to T, and $\mathcal{N}$ represents a Gaussian distribution.

Moreover, it is feasible to draw samples from this forward process at any given time t with the initial image at time $t=0$ as the reference. By leveraging the reparameterization trick alongside the properties of normal distributions, the following relation can be easily derived [**43, 59**]:

$$q(x_t|x_0) = \mathcal{N}\left(x_t; \sqrt{\bar{\alpha}_t}x_0, (1-\bar{\alpha})\,\mathrm{I}\right), \tag{3.1.12}$$

where $\alpha_t = 1 - \beta$ and $\bar{\alpha} = \prod_{i=1}^{t} \alpha_i$.

However, a computational hurdle arises when attempting to derive the posterior of the forward diffusion process $q\left(x_{t-1}|x_t\right)$, as it necessitates the entire data distribution for its computation, rendering it intractable. Nevertheless, an alternative expression for $q\left(x_{t-1}|x_t\right)$ can be obtained by conditioning it on the input data x_0 via Bayes' Theorem, which simplifies the computation as follows:

$$\begin{aligned} q\left(x_{t-1}|x_t,x_0\right) &= \frac{q\left(x_t|x_{t-1},x_0\right) q\left(x_{t-1}|x_0\right)}{q\left(x_t|x_0\right)}, \\ &= \mathcal{N}\left(x_{t-1}; \tilde{\mu}(x_t,t), \tilde{\beta}_t \mathrm{I}\right) \end{aligned} \tag{3.1.13}$$

where the expressions for $\tilde{\mu}(x_t,t)$ and $\tilde{\beta}_t$ are given by:

$$\tilde{\mu}(x_t,t) = \frac{1}{\sqrt{\alpha_t}}\left(\mathrm{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_t\right) \tag{3.1.14}$$

$$\tilde{\beta}_t = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}_t} \cdot \beta_t. \tag{3.1.15}$$

Utilizing the expression for $q\left(x_{t-1}|x_t,x_0\right)$, Sohl-Dickstein et al. delineated the Evidence Lower Bound (ELBO) loss, which is essentially the negation of ELBO, as follows [**59**]:

$$\begin{aligned} L_{\mathrm{ELBO}} &= \mathbb{E}_{q(x_{0:T})}\left[\log \frac{q\left(x_{1:T} \mid x_0\right)}{p_\theta\left(x_{0:T}\right)}\right] \\ &= \mathbb{E}_{q(x_{0:T})}\left[\underbrace{D_{\mathrm{KL}}\left(q\left(x_T \mid x_0\right) \| p_\theta\left(x_T\right)\right)}_{\text{Prior Matching Term}} + \underbrace{\sum_{t=2}^{T} D_{\mathrm{KL}}\left(q\left(x_{t-1} \mid x_t, x_0\right) \| p_\theta\left(x_{t-1} \mid x_t\right)\right)}_{\text{Denoising Matching Term}} - \underbrace{\log p_\theta\left(x_0 \mid x_1\right)}_{\text{Reconstruction Term}}\right] \end{aligned} \tag{3.1.16}$$

Here, the Prior Matching Term doesn't encompass any trainable parameters, as it is postulated that at time step T, the image is distributed according to a standard normal distribution, thus it can be disregarded. The Denoising Matching Term reflects the adequacy of the decoder p_θ in approximating the exact reverse diffusion process. Lastly, akin to the Variational Autoencoder (VAE), the Reconstruction Term is present. Ho et al. proposed employing a U-Net architecture inspired by PixelCNN++ [**25, 58**] to model $p_\theta(x_{t-1}|x_t)$ parameterized as follows:

$$\begin{aligned} p_\theta(x_{t-1}|x_t) &= \mathcal{N}\left(x_{t-1}; \mu_\theta(x_t,t), \Sigma_\theta(x_t,t)\right) \\ \mu_\theta(x_t,t) &= \frac{1}{\sqrt{\alpha_t}}\left(\mathrm{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\epsilon_\theta(x_t,t)\right) \end{aligned} \tag{3.1.17}$$

Through this parameterization for p_θ and the expression for $\tilde{\mu}(x_t,t)$ shown in equation 3.1.14, it's demonstrable that the Denoising Matching Term in the ELBO loss is expressed as:

$$
\begin{aligned}
L_{\text{Denoising}} &= D_{\text{KL}}\left(q\left(x_{t-1} \mid x_t, x_0\right) \| p_\theta\left(x_{t-1} \mid x_t\right)\right) \\
&= \mathbb{E}_{x_0, \boldsymbol{\epsilon}}\left[\frac{1}{2\left\|\boldsymbol{\Sigma}_\theta\left(x_t, t\right)\right\|_2^2}\left\|\tilde{\boldsymbol{\mu}}_t\left(x_t, x_0\right)-\boldsymbol{\mu}_\theta\left(x_t, t\right)\right\|^2\right] \\
&= \mathbb{E}_{x_0, \boldsymbol{\epsilon}}\left[\frac{\left(1-\alpha_t\right)^2}{2\alpha_t\left(1-\bar{\alpha}_t\right)\left\|\boldsymbol{\Sigma}_\theta\right\|_2^2}\left\|\boldsymbol{\epsilon}_t-\boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t} x_0+\sqrt{1-\bar{\alpha}_t} \boldsymbol{\epsilon}_t, t\right)\right\|^2\right]
\end{aligned}
\tag{3.1.18}
$$

In their work, Ho et el further assumed that covariance matrix was a constant diagonal matrix $\Sigma_\theta(x_t,t) = \sigma_t \mathrm{I}$ where $\sigma_t^2 = \beta_t$ or $\sigma_t^2 = \frac{1-\bar{\alpha}_{t-1}}{1-\bar{\alpha}}\beta_t$. Using this simplifying assumption and a few other assumptions, they showed that the Reconstruction Term had a very similar form to the Denoising Matching Term with:

$$
L_{\text{Reconstruction}} = \mathbb{E}_{\boldsymbol{\epsilon} \sim \mathcal{N}(0, I)}\left[\frac{\beta_1}{2\alpha_1 \sigma_1^2}\left\|\epsilon-\epsilon_\theta\left(x_0 \sqrt{\alpha_1}+\sqrt{1-\alpha_1} \epsilon, 1\right)\right\|_2^2\right] \tag{3.1.19}
$$

And further found that ignoring the multiplying constants of both the reconstruction loss and the denoising matching loss to create a single unified and simplified loss led to better empirical results.

$$
\begin{aligned}
L_{\text{Simple}} &= \mathbb{E}_{t \sim[1, T], x_0, \epsilon_t}\left[\left\|\boldsymbol{\epsilon}_t-\boldsymbol{\epsilon}_\theta\left(x_t, t\right)\right\|^2\right] \\
&= \mathbb{E}_{t \sim[1, T], x_0, \epsilon_t}\left[\left\|\boldsymbol{\epsilon}_t-\boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t} x_0+\sqrt{1-\bar{\alpha}_t} \boldsymbol{\epsilon}_t, t\right)\right\|^2\right]
\end{aligned}
\tag{3.1.20}
$$

The loss mechanism manifests prominently in the DDPM sampling delineated in Algorithm 1. In each iteration, an image from the training set, a timestep t from a uniform distribution, and a noise factor $\epsilon \sim \mathcal{N}(0, \mathrm{I})$ from a normal distribution are sampled. Subsequently, a gradient descent step concerning the parameters θ of the U-Net, which models the noise engendered at each juncture of the forward diffusion process, is undertaken. Noteworthily, the authors discerned a significant resemblance between the outlined sampling procedure in Algorithm 2 and Langevin dynamics shown in equation 3.1.21.

$$
x_t \leftarrow x_{t-1}+\frac{\delta}{2} \nabla_x \log p_\theta(x)+\sqrt{\delta} \epsilon_t \tag{3.1.21}
$$

Langevin dynamics, a stochastic process employed in statistical physics and computational biology, simulates systems' behavior in a thermal bath by melding deterministic Newtonian dynamics with stochastic Brownian motion. Through discrete sampling from this process, one can navigate the energy landscape of a system, furnishing a framework for sampling from intricate probability distributions—a useful paradigm in probing the configuration space of molecular systems or optimizing in machine learning landscapes [**73**].

Algorithm 1 Training	**Algorithm 2** Sampling

repeat
  $x_0 \sim q\left(x_0\right)$
  $t \sim \text{Uniform}(\{1, \ldots, T\})$
  $\boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
  Take gradient descent step on
  $\nabla_\theta\left\|\boldsymbol{\epsilon}-\boldsymbol{\epsilon}_\theta\left(\sqrt{\bar{\alpha}_t} x_0+\sqrt{1-\bar{\alpha}_t} \boldsymbol{\epsilon}, t\right)\right\|^2$
until converged
	$x_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
	for $t = T$ to 1 **do**
	  $\mathbf{z} \leftarrow$ **if** $t > 1$ **then** $\mathcal{N}(\mathbf{0}, \mathbf{I})$ **else 0**
	  $x_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(x_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}t}}\boldsymbol{\epsilon}_\theta\left(x_t, t\right)\right) + \sigma_t \mathbf{z}$
	end for
	return x_0

Fig. 3.4. Algorithms adapted from the DDPM paper by Ho et al. [**25**], illustrating the utilization of a simplified loss for training the DDPM (left) and showcasing how the sampling algorithm bears resemblance to Langevin dynamics (right).

Up until now, our discussion has primarily centered around the diffusion model, conceptualized as a hierarchical VAE with predefined Gaussian encoders. The interconnection with Langevin dynamics underlines a profound link between diffusion models and energy-based models. We can articulate the forward diffusion process as the subsequent Stochastic Differential Equation (SDE) by employing the reparameterization trick:

$$\begin{aligned}
x_t &= \sqrt{1-\beta_t}x_{t-1} + \sqrt{\beta_t}\epsilon_t && \epsilon_t \sim \mathcal{N}(\prime,\mathrm{I}) \\
&= \sqrt{1-\beta(t)\Delta t}x_{t-1} + \sqrt{\beta(t)\Delta t}\epsilon_t && \text{Let } \beta_t = \beta(t)\Delta t \\
&\simeq x_{t-1} - \frac{\beta(t)\Delta t}{2}x_{t-1} + \sqrt{\beta(t)\Delta t}\epsilon_t && \text{Taylor expansion} \\
&\simeq x_{t-1} - \frac{\beta(t)\Delta t}{2}x_{t-1} + \sqrt{\beta(t)}\Delta w && \text{Wiener process } \Delta w = \sqrt{\Delta t}\epsilon_t \\
\Rightarrow \mathrm{d}x_t &= -\frac{\beta(t)}{2}x_t \mathrm{d}t + \sqrt{\beta(t)}dw && (3.1.22)
\end{aligned}$$

This SDE is renowned for its reverse SDE diffusion process, which serves as a generative mechanism, as elucidated by Brian D.O. Anderson in 1982 [**2**]. It is expressed as:

$$\mathrm{d}x_t = \overbrace{\left[-\frac{1}{2}\beta(t)x_t - \beta(t)\underbrace{\nabla_{x_t}\log q_t\left(x_t\right)}_{\text{Score Function}}\right]}^{\text{Drift Term}}\mathrm{d}t + \overbrace{\sqrt{\beta(t)}\mathrm{d}w}^{\text{Diffusion Term}} \tag{3.1.23}$$

The formulation of both forward diffusion and reverse generation processes through these SDEs unveils captivating concepts. From a physical standpoint, it illustrates how diffusion models can be represented using a drift term, embodying the deterministic facet of the process, while the diffusion term encapsulates the Brownian motion. Moreover, the introduction of the Score Function $\nabla_{x_t}\log q_t\left(x_t\right)$ paves the way for a natural training objective, enabling the modeling of the diffusion process with SDEs [**62**]. This entails approximating the score function with a neural network, typically a U-Net $s_\theta(x_t, t)$, through the ensuing objective:

$$\mathbb{E}_{x_0\sim P_{\text{train}}(x),\ t\sim\mathcal{U}(0,1),\ x_t\sim q(x_t|x_0)}\left[\left\|s_\theta(x_t,t) - \nabla_{x_t}\log q_t\left(x_t|\, x_0\right)\right\|_2^2\right] \tag{3.1.24}$$

Upon training the score model $s_\theta(x_t, t)$, this model can be utilized alongside SDE solvers for image generation, by replacing the score function in equation 3.1.23.

Furthermore, it's notable that the score function is the negation of the gradient of the energy concerning the image.

$$\nabla_x \log p\left(x\right) = \nabla_x \log \frac{e^{-E(x)}}{Z} = \nabla_x\left(-E(x) - \log Z\right) = -\nabla_x E(x) \tag{3.1.25}$$

This implies that the gradient actively propels the image towards a lower energy realm, which is more probable under the Boltzmann distribution. This is manifestly depicted in Langevin dynamics, as shown in equation 3.1.21, where the image is driven towards a lower energy within the image energy landscape. Conversely, within Anderson's reverse diffusion process (equation 3.1.23), the presence of the negative score function concerning q_t propels the distribution away from the normal distribution, reverting it back to the image energy landscape.

The use of diffusion models, both score-based and variational based has gained a lot of traction recently in the medical imaging literature as shown by an exponential growth in the number of articles published

about the topic [**35**] since 2021. Notable accomplishments include state-of-art result in image-to-image translation tasks. For example, Lyu and Wang have shown that diffusion models were more effective than GANs to convert from the MRI to CT imaging modality [**44**]. Interestingly, diffusion models also had major successes in the generation of synthetic data for segmentation tasks with BrainSPADE in 2022 by Fernandez et al. [**13**].

3.1.4. Vector Quantized Models

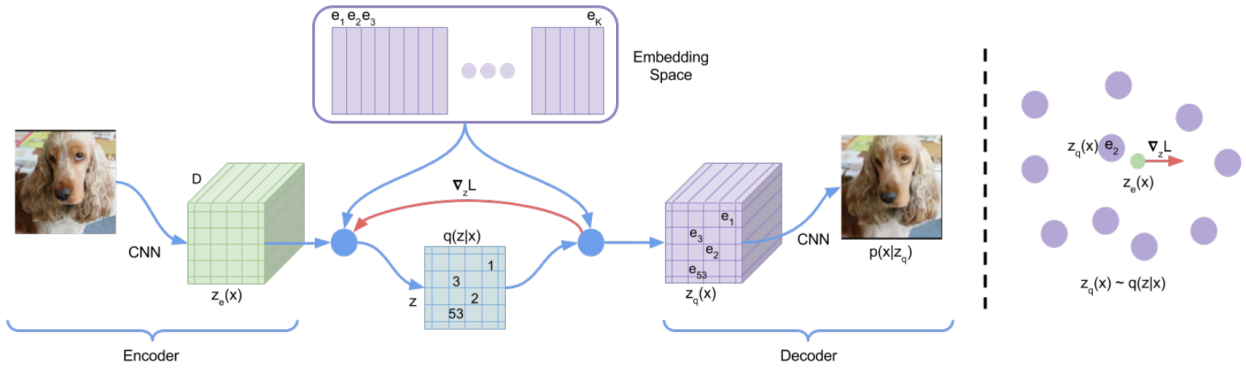


Fig. 3.5. Illustration of the VQ-VAE model, directly taken from the seminal work of Van Den Oord et al. [**69**]. This visualization highlights the straight-through gradient estimator, which allows for backpropagation through non-differentiable operations. The figure underscores the nearest neighbor assignment of encoder outputs, labeled as $z_e(x)$, to designated entries in the codebook, such as e_1, e_2, and so forth. The depicted gradient, $\nabla_z L$, guides refinements in the encoder, possibly leading to varied nearest neighbor allocations in subsequent iterations.

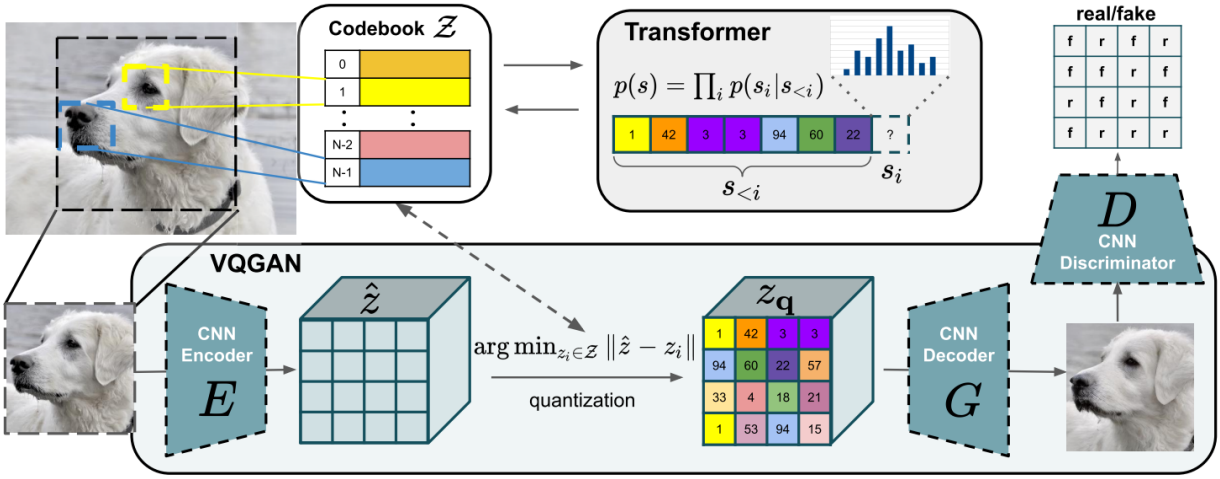


Fig. 3.6. Illustration of the VQ-GAN model, directly taken from the paper by Esser et al. [**12**]. As with VQ-VAE, a convolutional VQGAN is used to establish a codebook of context-rich visual components. By integrating a patch-based discriminator, the model ensures efficient compression while maintaining perceptual quality. Whereas the VQ-VAE model employs PixelCNN to learn the exact prior distribution of the codebook vectors during its second stage of training, the VQ-GAN utilizes a transformer architecture for the same purpose.

The Vector Quantized Variational Autoencoder (VQ-VAE) was introduced by Van Den Oord et al. [**69**] in 2017 to counteract the blurriness and component collapse observed in traditional VAEs. In this model,

the latent space of the VAE is discretized by associating each encoder output with the nearest vector in a predefined set known as a codebook. Figure 3.5 provides a visualization of this encoder-to-codebook mapping in the VQ-VAE model. For simplification, let's consider the scenario where the encoder produces a singular vector, resulting in a single random variable. The posterior approximator distribution is described by:

$$q_\phi(z|x) = \begin{cases} 1, & \text{if } z = e_{\arg\min_i \|z_e(x) - e_i\|_2} \\ 0, & \text{otherwise} \end{cases} \tag{3.1.26}$$

Here, e_i denotes the i-th vector within a codebook comprised of K vectors, and $z_e(x)$ is the output of the encoder. This distribution, $q_\phi(z|x)$, is deterministic, leading to the consistent quantization step given by:

$$z_q = e_{\arg\min_i \|z_e(x) - e_i\|_2} \tag{3.1.27}$$

However, this quantization process is not differentiable, obstructing the flow of gradients through the encoder. To address this challenge, Van Den Oord et al. applied the straight-through gradient estimator. This approach aligns the gradient concerning the encoder's output with the gradient concerning the quantized representation, i.e., $\nabla_{z_e(x)} L \leftarrow \nabla_{z_q(x)} L$. This method operates on the assumption that the matched codebook vectors closely resemble their respective inputs. Therefore, in order to learn the embeddings, the authors integrated the Vector Quantization (VQ) algorithm. This algorithm employs a straightforward l_2 objective, pulling the codebook vectors closer to the encoder's inputs. This is represented by the VQ Loss in equation 3.1.28. Moreover, to avert large disparities between the encoder vectors and codebook vectors, especially when the encoder vectors evolve more swiftly than the codebook vectors, a commitment loss is added.

The comprehensive Evidence Lower Bound (ELBO) loss function for VQ-VAE is articulated as:

$$L_{\text{VQ}}(x) = \underbrace{-\log p_\theta(x|z_q(x))}_{\text{Reconstruction Term}} + \underbrace{\|sg[z_e(x)] - e\|_2^2}_{\text{VQ Objective}} + \underbrace{\|z_e(x) - sg[e]\|_2^2}_{\text{Commitment Loss}} \tag{3.1.28}$$

In this context, sg denotes the stop-gradient operation. This operation inhibits backpropagation through its corresponding term, effectively rendering it a static target. Interestingly, the authors later suggested the substitution of the VQ objective in the loss with exponentially moving averages, a method they employed in their subsequent research on VQ-VAE-2 [**53**].

Notably absent from this loss function (3.1.28) is the KL-divergence term. This omission stems from the assumption of a uniform distribution over the codebook vectors. Consequently, the KL-divergence is a constant value of $\log K$ where K is the number of codebook vectors which does not hinge on the encoder's parameters:

$$\begin{aligned} D_{KL}(q_\phi(z|x)||p(z)) &= \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{q_\phi(z|x)}{p(z)}\right] \\ &= -\mathbb{E}_{q_\phi(z|x)}\left[-\log q_\phi(z|x)\right] - \mathbb{E}_{q_\phi(z|x)}\left[\log p(z)\right] \\ &= -\underbrace{\mathbb{H}\left[q_\phi(z|x)\right]}_{\text{Entropy is 0}} - \mathbb{E}_{q_\phi(z|x)}\left[\log \frac{1}{K}\right] \\ &= \log K \end{aligned}$$

This approach offers a solution to the component collapse challenge observed in conventional VAEs. With the absence of the KL-divergence term, there's no longer a suppressive force on the amount of information

in the approximate posterior conditioned on x. While this assumption of uniformity holds during the initial phase of VQ-VAE training, where the codebook vectors and autoencoder parameters are learned, the subsequent phase introduces an adjustment. Here, an autoregressive PixelCNN is employed to estimate a more precise empirical prior, which then guides image generation.

In February 2021, Ramesh et al. unveiled DALL-E [**52**], a new Vector Quantized model characterized by a two-stage training process. This approach introduced two pivotal innovations. The foremost innovation was the incorporation of the Gumbel-Softmax relaxation [**29**]. By approximating the sampling from a discrete distribution with a continuous and differentiable function, this method allowed for gradient propagation through the quantization step. As a result, the traditional VQ objective, commitment loss, and straight-through gradient became redundant, allowing the model to only utilize the KL divergence component of the VAE loss. The second major change was using transformers instead of PixelCNN to determine the prior distribution. The model specifically used a version of GPT-3, trained to predict image parts one after the other. This approach highlighted the effectiveness and scalability of transformer models when working with large datasets.

In 2021, Esser et al. introduced the VQ-GAN model, which presents key advancements over the traditional VQ-VAE framework [**12**]. A significant modification involves substituting the l_2 reconstruction objective with a perceptual loss. Instead of directly comparing the image with its reconstruction, the perceptual loss evaluates the intermediate feature maps extracted from a pretrained VGG network on ImageNet. Such a loss has demonstrated enhanced perceptual image quality [**31, 77**]. Additionally, the VQ-GAN incorporates adversarial training by integrating a patch-based discriminator D. The comprehensive training objective can be expressed as:

$$G^* = \arg\min_G \max_D \mathbb{E}_{x\sim P(x)} \left[L_{\text{VQ}}(x) + \log D(x) + \log(1 - D(\hat{x}))\right] \tag{3.1.29}$$

Within this equation, the generator parameters are denoted by $G = \{\theta,\phi, E\}$, with $E = \left\{\bigcup_{i=1}^{K} e_i\right\}$ representing the codebook. Analogous to DALL-E, the VQ-GAN leverages an auto-regressive transformer during the second training phase to learn the prior.

VQ-VAEs and VQ-GANs have emerged as effective paradigms for anomaly detection in medical imaging. Pinaya et al. (2022) proposed a sophisticated methodology for anomaly segmentation employing a VQ-VAE. Specifically, the autoencoder encodes the input image into a discrete latent representation, which is then used to train an autoregressive Transformer. This Transformer learns to capture the statistical distribution of the codebook vectors within the training set [**50**]. Consequently, when applied to unseen images, the Transformer can discern codebook vectors that deviate from the expected distribution, effectively identifying anomalous regions. These regions can then be re-sampled using the trained Transformer, and the disparity between the reconstructed image and the original serves as a segmentation map, isolating potential anomalies.

In a parallel development, Graham et al. (2022) introduced a comparable methodology utilizing a VQ-GAN for the encoding process. An autoregressive Transformer was subsequently trained to model the distribution of the codebook, facilitating the identification of outlier images within the sample space [**17**].

3.1.5. Latent Diffusion Models

In 2022, Gu et al. introduced VQ-Diffusion, an evolution of the VQ-VAE architecture, by incorporating a diffusion mechanism in lieu of the traditionally employed auto-regressive models [**18**]. This novel approach achieved lower FID scores, as elaborated in Section 3.2.1, compared to VQ-GAN for class-conditional image generation on ImageNet. Furthermore, VQ-Diffusion outperformed DALL-E in text-to-image synthesis tasks.

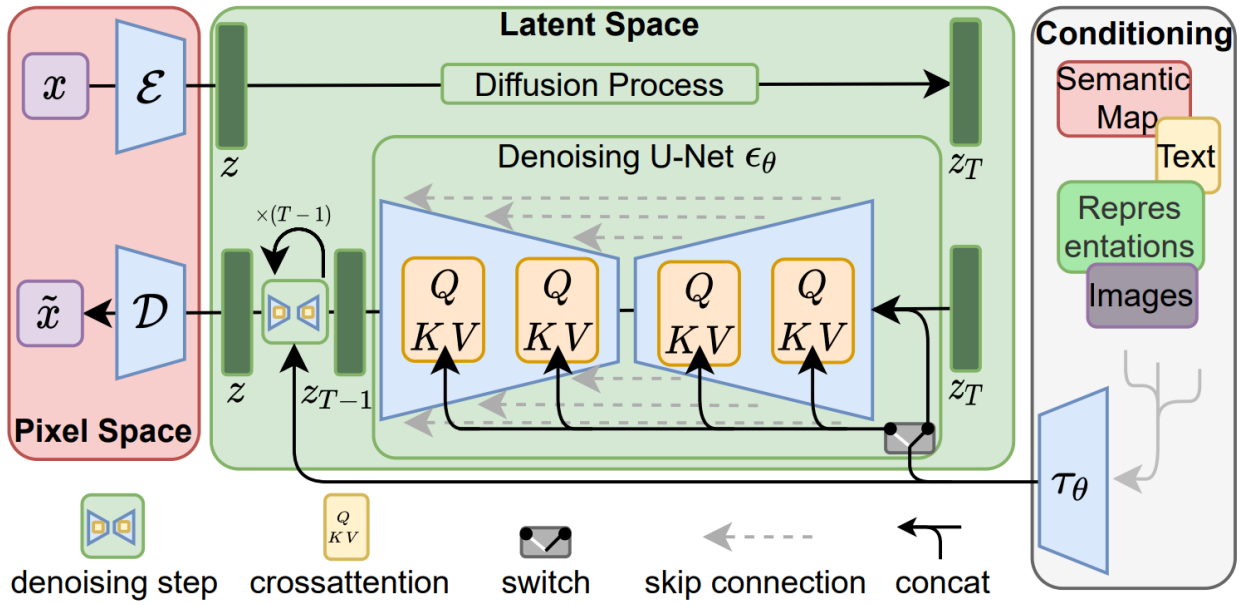


Fig. 3.7. Adapted schematic representation from Rombach et al. [**54**]. In this depiction, the autoencoder utilized is a VQ-GAN, where the decoder D incorporates both the quantization step and the decoding process, as detailed in Figure 3.6. The model accommodates conditioning inputs through two mechanisms: either by concatenating them with the Gaussian input supplied to the diffusion module or via cross-attention mechanisms.

One of the primary advantages of VQ-Diffusion, which is a type of Latent Diffusion Model (LDM), is its computational efficiency. Unlike traditional diffusion models that operate in pixel space, LDMs like VQ-Diffusion function in a more compact latent space, thereby significantly reducing computational demands during both training and inference.

Concurrently, Rombach et al. (2022) extended this principle of computational efficiency in LDMs by training them in the latent space of a pretrained VQ-GAN autoencoder. Transitioning to latent space not only decreases computational complexity but also retains the capability for high-fidelity image synthesis. The architecture was further enhanced with cross-attention layers, equipping these LDMs to manage diverse conditioning inputs such as text and bounding boxes, as illustrated in Figure 3.7. As a result, LDMs establish new performance benchmarks in multiple tasks, including image inpainting, class-conditional image synthesis, text-to-image synthesis, and super-resolution, all the while requiring significantly fewer computational resources.

In the domain of medical imaging, Pinaya et al. (2022) demonstrated the effectiveness of LDMs in generating realistic and diverse medical images. Utilizing the Fréchet Inception Distance (FID) for image quality and the Multi-Scale Structural Similarity Index (MS-SSIM) for diversity, the authors found that LDMs outperform competing models like LSGAN and VAE-GAN in both metrics [**49**].

By May 2023, a study conducted by Khader et al. revealed the advantages of integrating a VQ-GAN autoencoder with a diffusion-based probabilistic model to form an LDM, specifically for generating 3D medical images [**37**]. Their research emphasized the model's prowess in creating high-quality MRI and CT scans. To validate the realism, anatomical accuracy, and inter-slice coherence of these synthetic images, evaluations were sought from two radiologists. Additionally, the study accentuated the utility of such synthetic images in self-supervised pre-training tasks. These tasks employed masked volume inpainting, contrastive learning, and

rotation prediction as proxy tasks for learning context-aware image representations [**66**]. When included in training datasets, synthetic images elevated the performance of breast segmentation models, particularly in data-limited scenarios. Notably, this methodology led to an improvement in Dice scores, a widely recognized segmentation performance metric, from 0.91 to 0.95.

3.2. Evaluation of Deep Generative Models for Image Synthesis

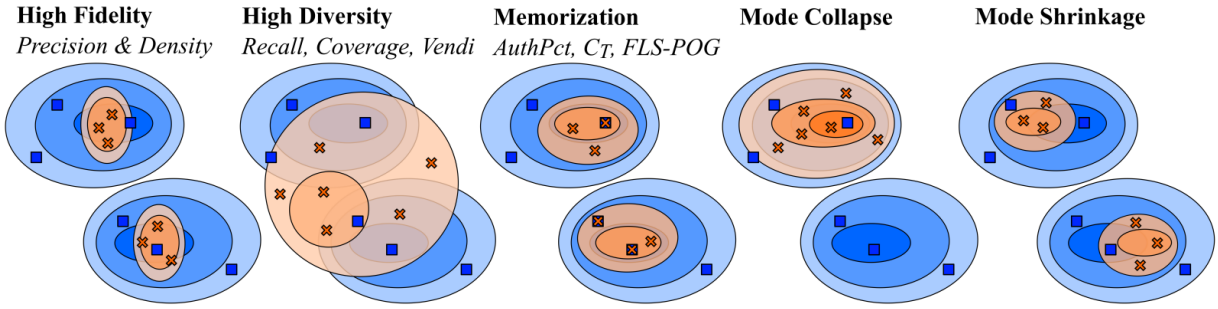


Fig. 3.8. Schematic depiction from the work of Stein et al. [**64**], illustrating the various success and failure modes of a generative model alongside an illustration of learned distributions and samples (denoted by orange crosses) exhibiting different properties compared to the true distribution and training set (denoted by blue squares). Italicized text delineates metrics purported to detect these properties. High fidelity refers to the congruence between synthetic images generated and the real distribution, implying that the synthetic images could be plausible samples from the real distribution. High diversity showcases the generative model's ability to produce varied samples, with the desired outcome being a match in diversity between the generative model and the true distribution. The failure mode of memorization occurs when the generative model regurgitates training data during generation. Mode collapse is another failure mode where the model predominantly generates samples from a specific mode of the distribution, neglecting other modes. Lastly, mode shrinkage represents a failure mode in which the diversity within each mode generated by the model is diminished compared to the true distribution.

As elaborated in the preceding section on generative modeling, the primary objective of these models is to maximize the marginal likelihood of the training data. A straightforward extension of this objective would be to evaluate the marginal likelihood with respect to a test dataset, as depicted in the following equation:

$$L_{\text{test}}(\theta) = \mathbb{E}_{x \sim P_{\text{test}}} \left[\log P_\theta(x) \right] \tag{3.2.1}$$

Nevertheless, the computation of marginal likelihood poses distinct challenges across different types of generative models. While autoregressive models are adept at directly computing the marginal likelihood, Variational Autoencoders (VAEs) and diffusion models, grounded on variational inference, can only approximate the marginal likelihood [**38**]. Conversely, Generative Adversarial Networks (GANs) lack a mechanism to compute the marginal likelihood altogether, as they are structured around an adversarial game. Owing to these computational constraints, alternative methods have been necessitated to evaluate the quality of images generated by these models. However, the quest for a robust evaluation metric for image generative models remains an open challenge. Existing metrics, while useful, fall short of encapsulating the nuanced distinction between high-quality and subpar generative models comprehensively.

In the paper "Exposing Flaws of Generative Model Evaluation Metrics and Their Unfair Treatment of Diffusion Models" by Stein et al. [**64**] various success and failure modes encountered by generative models were presented as shown in figure 3.8. Among the success indicators, *high fidelity* and *high diversity* are

paramount. High fidelity denotes the extent to which the synthetic images produced resemble real-world data. For instance, if a generative model is trained to create images of dogs and cats, a high fidelity outcome would entail the synthetic images being virtually indistinguishable from real images of dogs and cats. On the other hand, high diversity reflects the model's capability to generate a wide variety of samples. In the aforementioned example, a diverse output would encompass generating images of different breeds of dogs and cats in various poses and settings, mirroring the diversity present in the real distribution.

Nonetheless, generative models are susceptible to several failure modes. *Memorization* occurs when the model merely reproduces training data during generation, hence failing to generalize and create novel images. *Mode collapse* is a scenario where the model becomes biased towards generating samples from a specific mode of the distribution while neglecting others. For instance, the model might exclusively generate images of cats, disregarding the dog mode entirely. *Mode shrinkage* is yet another failure mode, where the model does generate samples from all relevant modes (e.g., both dogs and cats), but with reduced variety compared to the real distribution. For example, the model might generate images of only a few breeds of dogs and cats, or might produce them in a limited set of poses or settings, thus failing to capture the true diversity of the underlying distribution.

The spectrum of metrics employed for evaluating generative models within the literature is broad, reflecting the diverse challenges and objectives associated with these models. To maintain a coherent alignment with the methodologies adopted during the course of the internship, the ensuing discussion in this section will be chiefly centered around the Fréchet Inception Distance (FID) for assessing sample quality, and the Multi-Scale Structural Similarity Index Measure (MS-SSIM) for gauging diversity. Additionally, methods to scrutinize the risk of memorization will be explored, presenting a comprehensive overview of the evaluative landscape pertaining to generative models.

3.2.1. Fréchet Inception Distance

The Fréchet Inception Distance (FID) was introduced in 2018 by Heusel et al. as a means to evaluate the quality of images generated by Generative Adversarial Networks (GANs) [**24**]. The FID metric quantifies the dissimilarity between the distribution of real images and the distribution of generated images, offering an arguably more robust and perceptually meaningful assessment compared to the Inception Score [**57**], which lacks an explicit comparison between real and synthetic images. The computation of FID involves extracting feature representations from both real and synthetic image datasets using a pre-trained neural network, typically the Inception V3 network for RGB images. These feature representations are assumed to follow a Gaussian distribution. The FID is then calculated by computing the Fréchet distance, also referred to as the Wasserstein-2 distance, between these Gaussian distributions. The Fréchet distance is expressed as:

$$\mathrm{FID}(D_1, D_2) = \|\mu_1 - \mu_2\|_2^2 + \mathrm{tr}(\Sigma_1 + \Sigma_2 - 2 \cdot \sqrt{\Sigma_1 \Sigma_2}) \tag{3.2.2}$$

where:

$$\mu_D = \frac{1}{|D|} \sum_{x \in D} \phi(x) \quad \text{and} \quad \Sigma_D = \frac{1}{|D| - 1} \sum_{x \in D} (\phi(x) - \mu_D)(\phi(x) - \mu_D)^T$$

Here, D denotes the dataset, $\phi(x)$ represents the feature vector of image x, and μ_D and Σ_D are the mean and covariance of the feature vectors in dataset D, respectively.

While the Inception V3 network has been extensively utilized for evaluating 2D images, the evaluation of 3D medical images using the FID score necessitated an alternative approach due to the Inception network's requirement for a 3-channel input. Accordingly, several methods have been devised to adapt the FID score for 3D inputs. One approach entails applying the standard FID score and the Inception network to individual slices of the images. For example, Hong et al. calculated the FID scores on the central slices along the axial, coronal, and sagittal planes of the generated 3D images [**27**]. Alternatively, some researchers have advocated for the utilization of pretrained 3D networks to extract features, followed by computation of the Fréchet distance, thereby substituting the Inception V3 network [**65, 49**].

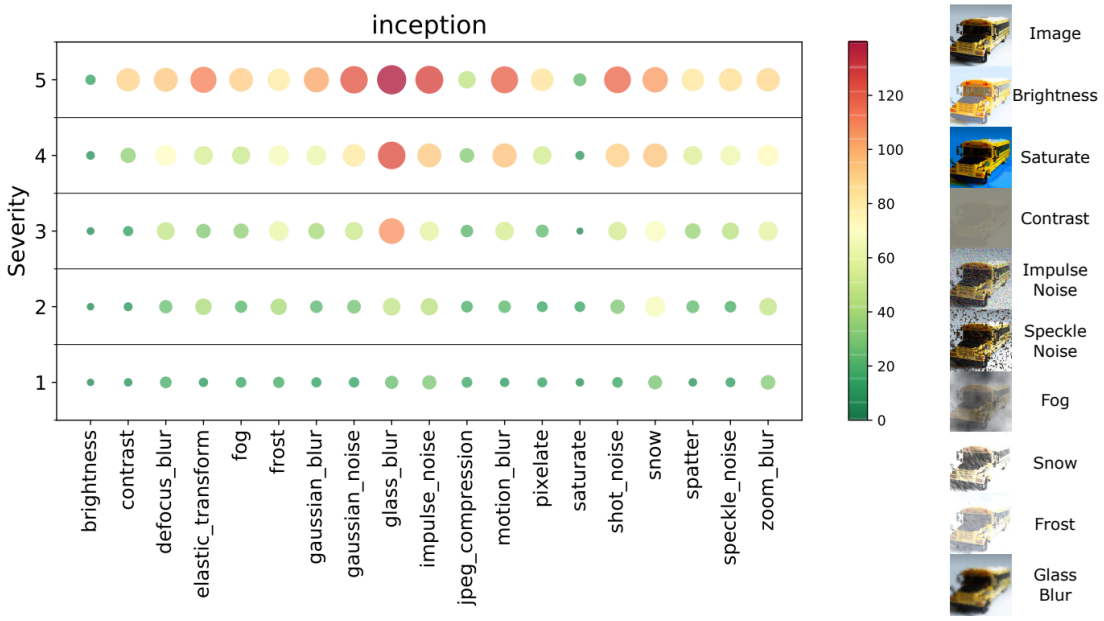


Fig. 3.9. This figure has been directly extracted from the work of Jung and Keuper. (Left) Color-coded representation of Fréchet Inception Distances (FIDs) computed between 19 different corruptions applied to ImageNet validation images at 5 severity levels and their original counterparts. Both the color intensity and circle size reflect the magnitude of the observed FID across all corruptions and severity levels. (Right) Illustrative examples showcasing various corruptions at severity level 5.

Despite its popularity as an evaluation metric for generative models, particularly Generative Adversarial Networks (GANs), the FID score has notable limitations. A primary concern identified early on is the significant bias of the FID score based on the sample size of the datasets under evaluation. Intriguingly, Chong and Forsyth demonstrated that both the FID and Inception scores are biased with respect to the number of samples and, additionally, the type of model being employed [**8**]. Through experimentation and a mathematical proof', they revealed a linear relationship between the expected FID score and $\frac{1}{N}$, where N denotes the number of images used for evaluation:

$$\mathbb{E}\left[\mathrm{FID}_N\right] = \mathrm{FID}_\infty + \frac{\alpha}{N} \tag{3.2.3}$$

Here, α is a positive constant dependent on the generative model. Consequently, obtaining an unbiased estimate of the FID score from a finite number of images is mathematically infeasible. A prevalent workaround to this issue is to employ a substantially large number of images such that the term $\frac{\alpha}{N}$ becomes negligible, a scenario often realized in large datasets commonly used in research literature. However, the authors proposed an effectively unbiased metric by fitting a linear relationship as per Equation 3.2.3, thereby

estimating the FID score for an infinite number of images. Although this metric has not gained widespread adoption in the literature, this study underscores the critical importance of understanding the effectiveness of the FID score, particularly for small datasets.

A distinct analysis conducted by Jung and Keuper elucidated that employing the Inception V3 feature extractor when computing the FID score results in an inherently flawed metric [**33**]. They substantiated this assertion through two primary observations. Initially, they demonstrated that the FID score exhibits a bias towards high-frequency features. This bias was especially pronounced when high-frequency information was diminished through the application of Gaussian blurring, significantly impacting the resultant FID score. Conversely, lower frequency features such as intensity and color exerted a lesser effect on the FID score. This phenomenon is consonant with the augmentation pipeline employed during the training of Inception V3, which introduces color distortions while preserving high-frequency information (contrary to, for instance, augmentation with Gaussian blur). Hence, the FID score inherits this bias, as illustrated in Figure 3.9.

Subsequently, they observed that incorporating the FID score as an optimization metric alongside the conventional objectives of GANs did not enhance the visual appeal of generated images to human evaluators. Despite a decrease in the FID score during model optimization, the resultant images were not deemed convincing to the human eye. This observation underscores the limitation of the Inception V3 FID as a comprehensive metric for evaluating all facets of image quality.

The assertions by Jung and Keuper find resonance in a later study by Stein et al., which highlighted the discriminatory nature of the FID score against diffusion models when juxtaposed with GANs. Despite yielding less visually appealing images, GANs exhibited lower FID scores [**64**]. Stein et al. further posited that advanced networks such as DINOv2-ViT-L, trained via self-supervised learning, harbor richer feature representations that facilitate more accurate evaluations of generative models [**48**].

3.2.2. Multi-Scale Structural Similarity Index Measure

The Multi-Scale Structural Similarity Index (MS-SSIM), initially introduced by Wang et al. is an extension of the Structural Similarity Index (SSIM) designed to provide a more comprehensive and accurate measure of the perceptual similarity between two images [**72, 71**]. Unlike SSIM, which evaluates similarity at a single scale, MS-SSIM considers image information at multiple scales, offering a more nuanced assessment of image similarity. MS-SSIM operates by applying a multi-scale decomposition to both images being compared. This process typically involves downsampling the images through several levels, where at each level, specific aspects of structural information are extracted and compared. The comparison metrics include luminance, contrast, and structure, similar to the original SSIM. However, in MS-SSIM, these comparisons are made at each scale, capturing both fine and coarse details. The overall MS-SSIM score is computed as a weighted product of the SSIM scores at each scale. This approach ensures that discrepancies at any scale significantly impact the final MS-SSIM value, making it sensitive to variations in both global structures and finer details. The weights can be adjusted to emphasize different scales, depending on the specific application or characteristics of the images under consideration.

Mathematically, the MS-SSIM index is computed as follows:

$$\text{MS-SSIM}(x, y) = [l_M(x, y)]^{\alpha_M} \cdot \prod_{j=1}^{M} [c_j(x, y)]^{\beta_j} \cdot [s_j(x, y)]^{\gamma_j} \tag{3.2.4}$$

where x and y are the two images being compared, and M is the number of scales used. The terms $l_M(x,y)$, $c_j(x,y)$, and $s_j(x,y)$ represent the luminance, contrast, and structure comparisons at each scale, respectively. The exponents α_M, β_j, and γ_j are the weights for the respective components.

The luminance comparison at the M^{th} scale is given by:

$$l_M(x,y) = \frac{2\mu_{xM}\mu_{yM} + C_1}{\mu_{xM}^2 + \mu_{yM}^2 + C_1} \tag{3.2.5}$$

where μ_{xM} and μ_{yM} are the average luminance values of x and y at the M^{th} scale, and C_1 is a small constant to avoid division by zero.

The contrast comparison at each scale j is defined as:

$$c_j(x,y) = \frac{2\sigma_{xj}\sigma_{yj} + C_2}{\sigma_{xj}^2 + \sigma_{yj}^2 + C_2} \tag{3.2.6}$$

where σ_{xj} and σ_{yj} are the standard deviations of x and y at scale j, and C_2 is a small constant.

Finally, the structure comparison at each scale j is:

$$s_j(x,y) = \frac{\sigma_{xyj} + C_3}{\sigma_{xj}\sigma_{yj} + C_3} \tag{3.2.7}$$

where σ_{xyj} is the covariance of x and y at scale j, and C_3 is a small constant.

While the primary application of the MS-SSIM has been in assessing the similarity between pairs of images, its utility extends to evaluating the diversity within synthetic datasets. This application of MS-SSIM is particularly relevant in contexts where dataset variability is crucial, such as in the development and validation of deep learning models for medical imaging tasks. The technique involves computing the expected MS-SSIM score between two images randomly selected from the dataset. A higher average MS-SSIM score across numerous pairs indicates lower diversity, as it suggests that the images within the dataset are more similar to each other. Conversely, a lower average MS-SSIM score implies greater diversity, reflecting a wider range of structural and textural features within the dataset. This approach has been effectively employed in biomedical settings, including the "Medical Diffusion" study [**56, 36**]. In these studies, the MS-SSIM index was employed as a quantitative metric to evaluate the heterogeneity, or diversity, within synthetic medical image datasets. This assessment is crucial for determining how comprehensively generative models are able to replicate the full range of variability inherent in the underlying data distribution.

In summary, the application of MS-SSIM for dataset diversity assessment offers a valuable tool for researchers and practitioners in the field of medical imaging and deep learning. By providing a quantitative measure of dataset variability, MS-SSIM helps in evaluating and improving the quality of synthetic datasets.

3.2.3. Investigating Memorization in Deep Generative Models for Image Synthesis

Recent advancements have highlighted a significant challenge in deep generative models, particularly diffusion models, known as memorization. This phenomenon is especially pronounced in scenarios where training datasets are limited in size and contain redundant data. Memorization manifests in two distinct forms: exact replication at the pixel level, and semantic duplication, where the essence or meaning of the image is replicated despite variations in pixel arrangement. Somepalli et al. demonstrated that diffusion models, when applied to digital art creation, can generate outputs that are semantically analogous to the training images, raising substantial copyright concerns for artistic applications of these models [**60**]. They termed this form of semantic replication as 'reconstructive memory'. Furthermore, Carlini et al. revealed the capability of diffusion models to regenerate near-duplicates of training data [**7**]. Additionally, Stein

et al. observed the inclination of diffusion models towards pixel-level memorization in smaller datasets like CIFAR10, and also identified instances of semantic memorization in more intricate datasets such as ImageNet, FFHQ, or LSUN-Bedroom [**64**].

To detect memorization, several studies have advocated the use of the l_2 pixel-wise distance metric, which is effective in identifying duplicates within the training set [**64**]. However, this approach is notably sensitive to standard data augmentation techniques employed during training, such as rotations and projections, and is also computationally intensive [**70**]. Consequently, many researchers have shifted towards employing image embeddings for identifying closely resembling images. Notable methodologies include the use of CLIP [**7**] and models such as SSCD, DINOv2, or Swin transformer [**7**]. This technique has also been previously applied in works like BigGAN by Brock et al., where a KNN search on various embeddings like VGG-16 and ResNet-50 was used to qualitatively evaluate the extent of memorization [**6**].

In terms of quantitative assessment of memorization, various metrics have been proposed, including AuthPct [**1**], the C_T score [**46**], and FLS-POG [**30**]. However, Stein et al., in their study titled "Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models", concluded that these metrics do not reliably measure memorization. Specifically, they found that the FLS-POG metric had a poor correlation with their estimates of memorized sample percentages, while the C_T score was more indicative of mode shrinking and image fidelity rather than actual memorization [**64**].

Chapter 4

Methodology

This section details the methodologies employed in adapting and implementing a Latent Diffusion Model (LDM) for medical imaging applications at Optina. The choice to explore diffusion models, particularly LDMs, was motivated by their proven stability in training, a significant advantage over Generative Adversarial Networks (GANs), as discussed by Dhariwal et al. [**11**] and Rombach et al. [**54**].

Optina's focus on classifying Amyloid-β in hyperspectral retinal images presented a unique challenge. Given the similarity of these images to CT and MRI scans, the "Medical Diffusion" model, developed by Khader et al. [**36**], was selected as our baseline model. This model has shown efficacy in breast scan segmentations and employs a novel training approach involving self-supervised learning on synthetic images followed by fine-tuning on real data, as detailed in Section 3.1.5 of our literature review.

In the subsequent subsections, we detail the specific modifications and experimental methodologies applied to adapt the Medical Diffusion model for Optina's goals. This involved creating a comprehensive evaluation pipeline to identify the most effective model configuration, ultimately enabling the generation of high-quality, diverse synthetic hyperspectral retinal images with high resolution.

4.1. Medical Diffusion: a Latent Diffusion Model

Diffusion models have been gaining traction in both research and industry because they tend to be more stable during training compared to generative adversarial networks [**11**]. This made diffusion models a fitting choice for Optina's needs. However, because training these models can be resource-intensive, we opted for a variant called a latent diffusion model [**54**]. Our initial choice was the "Medical Diffusion" model developed by Khader et al. [**36**]. As detailed in our literature review section 3.1.5, this model showed improved results in breast scan segmentations when it was pretrained via self-supervised learning on synthetic images before being fine-tuned on real data. Although our task at Optina is different — classifying Amyloid-β - the closest types of images to Optina's hyperspectral retinal shots are CT and MRI scans. Plus, Medical Diffusion's pretraining method includes a contrastive element, aligning with our goals. Given these points, we believed this model was a good fit for our aim of enhancing a Momentum Contrast (MoCo) self-supervised learning process. For a clearer picture of our objectives, refer to section 1.4.

4.1.1. VQ-GAN as the Autoencoder of Choice

Medical Diffusion utilizes Vector-Quantized Generative Adversarial Networks (VQ-GAN) as its autoencoder framework. As delineated in Section 3.1.4 on Vector-Quantized models, VQ-GAN encodes images into a discretized latent space using vector quantization within an autoencoder structure, thereby averting the

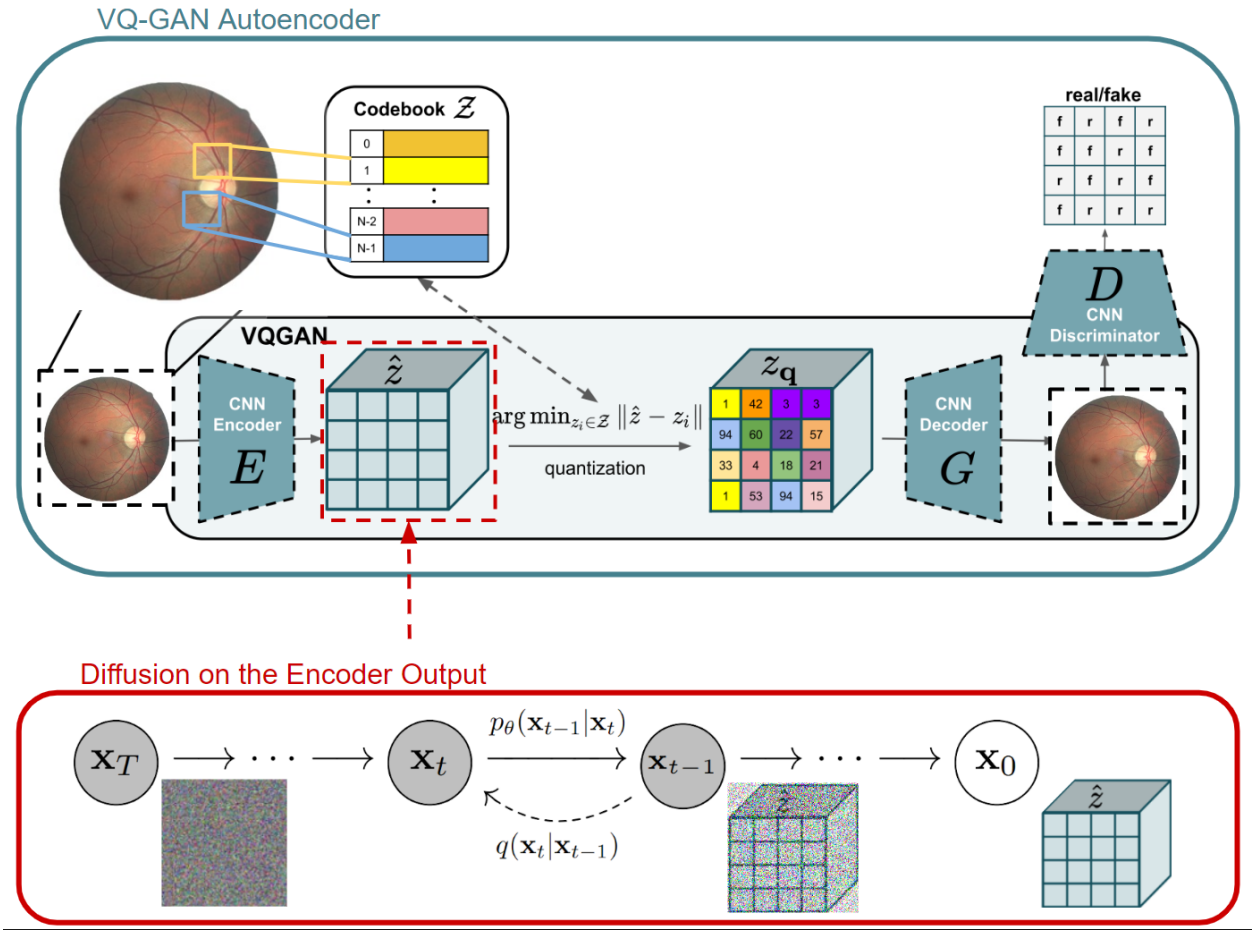


Fig. 4.1. Schematic representation adapted from the works of Esser et al. and Ho et al. of the Medical Diffusion model for the synthesis of hyperspectral retinal images [**12, 25**]. The framework employs a two-stage training approach: initially, a VQ-GAN autoencoder compresses images into compact latent representations; subsequently, a diffusion model is trained to generate new latent spaces from these representations.

blurred outcomes commonly seen with variational autoencoders. Specifically, the encoder converts the input image into a latent code with dimensions corresponding to height (H), width (W), depth (D) and a set (k) of feature maps, reduced by a compression factor (c). During the vector quantization phase, each latent vector is replaced by its nearest equivalent from a learned codebook Z, after which the decoder reconstitutes the image from these quantized vectors. The training prioritizes the reduction of the reconstruction loss L_{rec}—a combination of L1 loss and perceptual loss (LPIPS), alongside a commitment loss L_{commit}, which is expounded upon in the literature section 3.1.4 on Vector-Quantized models. The codebook itself is learned by computing an exponential moving average of all the latent vectors mapped to it. A patch-based, slice-wise discriminator with hinge feature-matching losses for GAN training stabilization is also employed to elevate the quality of reconstruction. To adapt to hyperspectral images, where adjacent wavelengths are closely interrelated, 2D convolutions have been supplanted by 3D convolutions. Differing from the original Medical Diffusion model that processed single-channel slices, the improved method extracts RGB slices from three randomly chosen channels, in decreasing order of wavelength, optimizing the discriminator's efficacy and utilizing the associated wavelength correlations in hyperspectral images.

To enhance the stability of VQ-GAN training and optimize codebook utilization, we introduced a K-means based reassignment of codebook vectors at the start of each training epoch. This reassignment leverages a rolling queue of training batches. A comprehensive justification for integrating this module is delineated in Annex A.4.

4.1.2. Denoising Diffusion Probabilistic Model

Diffusion models, as reviewed in Section 3.1.3, are a class of generative models that generate a sequence of images through a Markov chain by incrementally adding Gaussian noise over a set of time steps (T). Typically, a U-Net-based neural network is trained to reverse this noise addition, which enables the synthesis of new images. The U-Net used by Medical Diffusion follows the framework proposed by Ho et al. [**26**], which extends the denoising diffusion probabilistic model (DDPM) for use with video data. For the adaptation to hyperspectral imaging—interpreted as video sequences with each wavelength representing a frame—the model integrates 3D convolutions and employs attention mechanisms that operate on both spatial and spectral dimensions. These mechanisms are vital for the accurate reconstruction of hyperspectral images, ensuring the preservation of the data's spatial and spectral integrity.

4.1.3. Integration of VQ-GAN with DDPM for Latent Space Diffusion

The integration of the VQ-GAN with DDPMs involves an initial phase where VQ-GAN is trained to project the images into meaningful latent representations, detailed in the methods section 4.1.1. To prepare these representations for the diffusion process, they are normalized to the range of -1 to 1. This normalization is achieved by scaling the latent vectors based on the maximum and minimum values within the learned codebook, a method supported by the tendency of the encoder, under the influence of the commitment loss, to generate embeddings that align closely with the codebook vectors. The normalized latent space is then utilized to train the DDPM, as delineated in section 4.1.2, which is tasked with creating new latent representations through the inversion of the diffusion process. These latent representations are subsequently quantized with the VQ-GAN's codebook and decoded to reconstruct images. The training thus unfolds in two primary stages: the VQ-GAN first learns to compress the data into a latent form, and then the DDPM is trained to generate latent representations that can be converted back into new synthetic images.

4.2. Evaluation of Synthetic Hyperspectral Images

To effectively evaluate the performance of diverse LDM configurations, we implemented a comprehensive evaluation pipeline. This pipeline considers multiple factors: spatial and spectral quality, diversity, and the risk of memorization.

4.2.1. Adapting FID for Spatial Quality Assessment of Hyperspectral Datasets

As discussed in Section 3.2.1 of the literature review, the Fréchet Inception Distance (FID) score, computed using the Inception V3 network, is a prevalent metric for evaluating the quality of generated images in the RGB domain. However, applying this metric directly to hyperspectral images poses a challenge due to the disparity in channel dimensions between RGB and hyperspectral images. To circumvent this, we propose a pipeline to create pseudo RGB datasets from the hyperspectral dataset, thereby enabling the adaptation of the Inception V3 FID score to the hyperspectral domain.

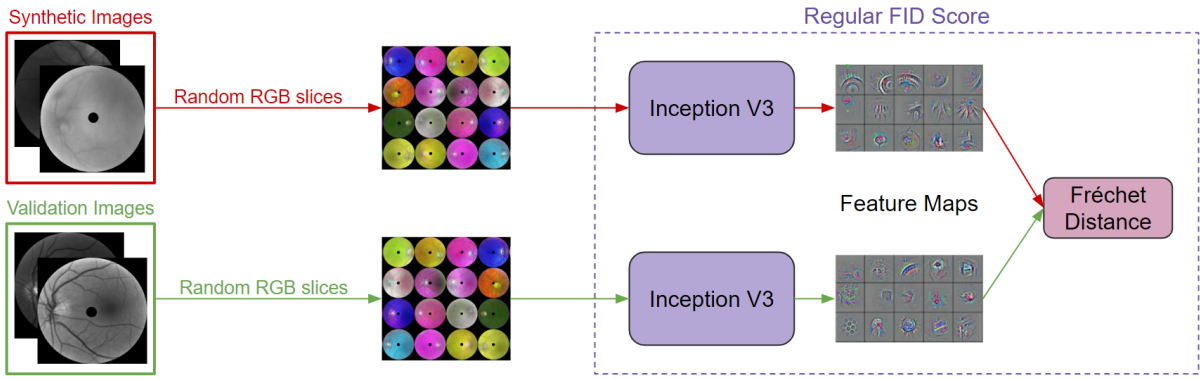


Fig. 4.2. Spatial Quality Evaluation Pipeline: For each image in the validation and synthetic hyperspectral datasets, N random RGB slices are extracted to construct a pseudo-RGB dataset. This procedure not only adapts the FID score for hyperspectral data but also mitigates bias from a limited dataset size by effectively enlarging the sample pool. The FID score is then computed by obtaining feature representations from an Inception V3 network and calculating the Fréchet distance between real and synthetic features to evaluate image quality.

Given a hyperspectral dataset D, where each image $x \in D$ is a 3D tensor of dimensions $C \times H \times W$, we define a function f to extract a random RGB sub-image from a hyperspectral image by selecting three distinct channels:

$$f(x) = x_{\text{rgb}} = x[[i,j,k], :, :]$$

where i, j, k are drawn from a uniform random distribution over the channel indices $\{1, 2, \ldots, C\}$ such that $i \neq j \neq k \neq i$, and $x[[i,j,k], :, :]$ denotes the selection of channels i, j, k from the hyperspectral image x to form the RGB sub-image x_{rgb}.

Applying function f N times to each hyperspectral image $x \in D$ yields N RGB sub-images per hyperspectral image x, forming the pseudo RGB datasets $D_{\text{rgb}}^{\text{val}}$ and $D_{\text{rgb}}^{\text{synth}}$ for the validation and synthetic datasets respectively:

$$D_{\text{rgb}}^{\text{val}} = \bigcup_{x \in D_{\text{val}}} \{f(x)_n\}_{n=1}^{N}, \quad D_{\text{rgb}}^{\text{synth}} = \bigcup_{x \in D_{\text{synth}}} \{f(x)_n\}_{n=1}^{N} \tag{4.2.1}$$

Finally, the hyperspectral FID is defined as the FID score computed between the pseudo RGB datasets of the validation and synthetic datasets:

$$\text{FID}_{\text{hyperspectral}}(D_{\text{val}}, D_{\text{synth}}) = \text{FID}(D_{\text{rgb}}^{\text{val}}, D_{\text{rgb}}^{\text{synth}}) \tag{4.2.2}$$

An experiment presented in Appendix A.1 determined that $N = 100$ is ideal for Optina's datasets. This proposed pipeline provides a mechanism for assessing the quality of synthetic hyperspectral images by leveraging the established FID score metric, thereby offering an avenue to evaluate the resemblance between the validation and synthetic hyperspectral datasets analogously to the evaluation of RGB images.

4.2.2. Custom Pipeline for Spectral Quality Evaluation of Optina's Datasets

The spectral quality of synthetic images within Optina's hyperspectral datasets was evaluated using a custom pipeline. A W-Net architecture [**75**] was adapted for segmenting veins, arteries, and the Optic Nerve Head (ONH) in the hyperspectral images. This segmentation utilized ground truth masks from Optina's

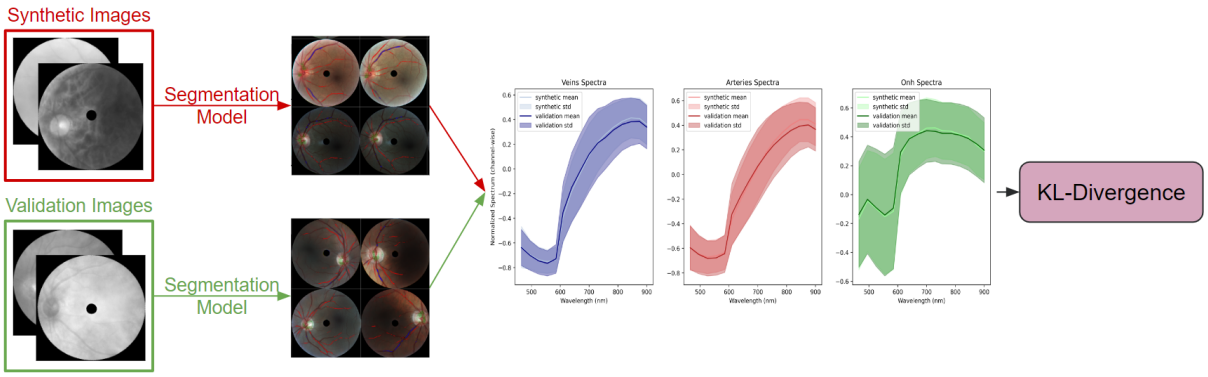


Fig. 4.3. Schematic of the Spectral Quality Evaluation Pipeline: The process involves segmenting images from the synthetic and validation datasets to identify regions corresponding to arteries, veins, and the ONH. The spectral distributions of these regions, parameterized as multivariate Gaussian distributions, are compared. The KL divergence between the distributions from the validation and synthetic datasets serves as the metric for the spectral quality of the synthetic images, with separate KL-divergence values calculated for the veins, arteries, and ONH.

database. Given the model's original design for RGB images, three hyperspectral channels—555, 585, and 610 nm—were selected for their enhanced contrast between veins and arteries.

Post-training, the W-Net underwent additional post-processing to enhance the precision of the segmentation. This step was crucial for the spectral quality assessment, aiming to accurately classify pixels corresponding to veins, arteries, or ONH. Detailed information about this post-processing is available in Annex A.2.

For the spectral quality assessment, the Kullback-Leibler (KL) divergence was calculated between the validation and synthetic datasets, with both distributions parameterized as multivariate Gaussian distributions. The KL divergence, considering the validation distribution as the reference, is defined as:

$$D_{KL}(P||Q) = \sum_{x \in X} P(x) \log \left(\frac{P(x)}{Q(x)} \right) \tag{4.2.3}$$

In this equation, P represents the probability distribution of the validation dataset, and Q represents the probability distribution of the synthetic dataset, with the summation over the set X of all possible events. For multivariate Gaussian distributions, this expression simplifies to:

$$D_{KL}(P||Q) = \frac{1}{2} \left(\text{tr}(\Sigma_Q^{-1}\Sigma_P) + (\mu_Q - \mu_P)^T \Sigma_Q^{-1}(\mu_Q - \mu_P) - k + \ln \left(\frac{\det \Sigma_Q}{\det \Sigma_P} \right) \right) \tag{4.2.4}$$

4.2.3. Diversity Evaluation Pipeline

To quantify the diversity of hyperspectral images, our approach is based on the MS-SSIM (Multi-Scale Structural Similarity Index) methodology, as implemented in the repository by Khader et al. [**36**]. This implementation is crucial for our work on medical image generation and follows the guidelines set forth in the original MS-SSIM framework by Wang et al. [**72**].

The MS-SSIM methodology is instrumental in assessing image diversity, evaluating structural similarities at multiple resolutions. The process involves systematically down-sampling images and calculating structural similarity at each level. The adopted implementation utilizes five resolution scales, following the specifications of Wang et al., with weighted coefficients assigned to each scale. These coefficients, in order of descending resolution, are: 0.0448, 0.2856, 0.3001, 0.2363, and 0.1333.

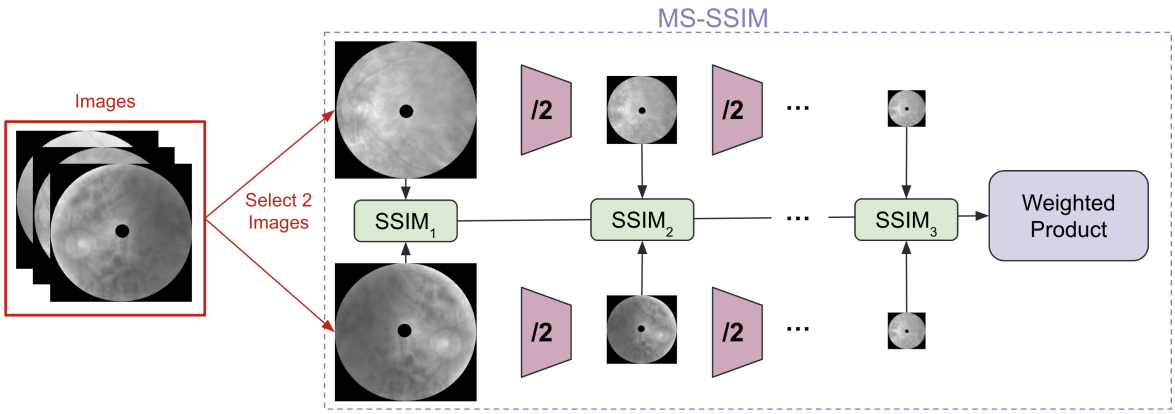


Fig. 4.4. Diversity Evaluation Pipeline: This figure illustrates the MS-SSIM implementation used for assessing image diversity. The process entails down-sampling the images by a factor of 2 at each successive scale, with the computation of structural similarity at each of the five resolutions. The weights for these resolutions, as established in the original MS-SSIM model by Wang et al. [**72**], are assigned in a descending order corresponding to the resolutions, ranging from 0.0448 to 0.1333.

Higher MS-SSIM scores denote greater similarity and, consequently, lower diversity among images, whereas lower MS-SSIM scores suggest increased diversity. Ideally, the goal is to achieve a close alignment between the MS-SSIM scores of the validation and synthetic datasets, indicating comparable levels of diversity.

4.2.4. Assessment of Memorization Risk

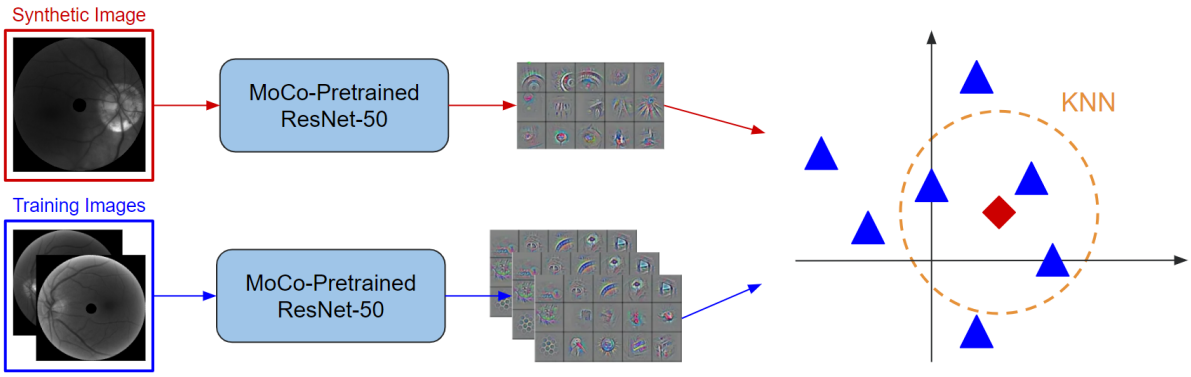


Fig. 4.5. Memorization Risk Assessment Pipeline: Synthetic images and the entire training set are processed through the MoCo-pretrained ResNet-50 model. Feature maps are extracted, followed by nearest neighbor retrieval using the Spotify Annoy algorithm [**41**], facilitating an efficient memorization risk evaluation.

The assessment of memorization risk in synthetic images is conducted using feature maps extracted by a ResNet-50 model, pretrained on the EyePacs dataset [**10**] employing the MoCo algorithm. The rationale for selecting this specific model for feature extraction is detailed in Annex A.3.

For efficient retrieval of nearest neighbors, we utilize Spotify's Annoy, an approximate K-NN algorithm [**41**]. This method allows for the rapid and effective identification of the 10 closest neighbors.

The memorization assessment involves a qualitative analysis of the 10-nearest neighbors recovered for each synthetic image. This analysis aims to discern the extent of similarities between the synthetic images and those in the training set, thus evaluating the risk of memorization.

4.3. Experiments

This section delineates the experimental procedures employed to rigorously assess the performance of various models using the evaluation pipeline described earlier. The evaluation criteria included quality, diversity, and the risk of memorization.

4.3.1. Initial Configuration on Low-Resolution Images

Initially, experiments focused on identifying the optimal LDM configuration for low-resolution (224 x 224 x 16) images. This resolution was selected to balance the visual recognition of key structures (such as veins, arteries, optic nerve head (ONH), and background texture) against the computational demands on Optina's GPUs.

Each experiment targeted specific hyperparameters of the VQ-GAN and examined the performance of the DDPM operating in the VQ-GAN latent space. We used U-Nets of varying sizes, characterized by a base number of channels. Each U-Net comprised three levels, with the number of feature maps doubling at each level. The U-Net sizes tested were 128, 256, and 512 as the base number of channels.

The main experiments conducted were:

- Comparing the VQ-GAN autoencoder against the VQ-VAE autoencoder by omitting the discriminator. Fixed parameters: 8 channels in the latent image, a compression factor of 8, and a codebook size of 1024.
- Determining the ideal codebook size, with tests conducted on codebooks of sizes 102, 1024, and 10240. Fixed parameters: 32 channels in the latent image, a compression factor of 8, and using a VQ-GAN autoencoder.
- Identifying the optimal size of the latent space, a function of the compression factor from the original image and the number of channels in the latent image. In this experiment, a VQ-GAN autoencoder with a codebook size of 1024 is used.

4.3.2. Fine-Tuning on High-Resolution Images

Upon establishing the best LDM configuration for 224 x 224 x 16 images, the chosen model was fine-tuned on higher-resolution images (672 x 672 x 16). While models trained on smaller images accommodated batch sizes of 10, those trained on larger images necessitated the use of stochastic gradient descent, leading to a preference for instance normalization over batch normalization to improve stability.

4.4. Computational Resources

For our experimental setup, we utilized two machines provided by Optina, each equipped with an Nvidia A40 GPU boasting 46 GiB of GPU memory. These machines, available at the onset of the internship, served as the primary computational resource for our deep learning experiments. The Nvidia A40 GPUs, while not as advanced as top-tier options like the Nvidia A100, were sufficiently capable for our needs. They supported the processing of large datasets and computational demands of our models, including tasks involving high-resolution image processing and complex model computations. This setup, though not the cutting edge in GPU technology, effectively facilitated our deep generative modeling tasks within the constraints of the available resources.

Chapter 5

Results and Discussion

In this section, we present a thorough examination of our experimental results with latent diffusion models (LDMs). The discussion initially centers on the generation of images at a resolution of 224 x 224 x 16 pixels. This phase is essential as it facilitates faster iteration across experiments due to the significantly reduced computational time and resources required when working with lower resolutions, compared to higher resolution images. This strategic approach allows for a more efficient exploration of the model's fundamental capabilities and limitations in a controlled experimental setup. Following this, we advance to a more detailed analysis, both quantitative and qualitative, focused on the generation of higher resolution images at 672 x 672 x 16 pixels. This progression is not only critical for assessing the scalability of our models but also offers valuable insights into their adaptability and effectiveness in generating diverse, high-resolution hyperspectral retinal images suitable for Optina's self-supervised learning projects.

5.1. Performance Evaluation on 224 x 224 x 16 Images

Our initial experiments focused on 224 x 224 x 16 resolution images, representing a balanced compromise between computational efficiency on Optina's GPUs and maintaining sufficient image quality. This resolution enabled the clear visualization of critical image details such as veins, arteries, the optic nerve head (ONH), and background texture. The choice of lower resolution facilitated the evaluation of a wide array of model configurations, providing deeper insights into the critical hyperparameters of the LDM. Key findings from these experiments include:

- The VQ-GAN consistently surpassed the VQ-VAE in all evaluated metrics, irrespective of the DDPM size. This superiority was particularly evident in spectral quality, with the synthetic spectra in areas of interest appearing significantly more realistic using VQ-GAN, attributable to the inclusion of the discriminator.
- The codebook size exhibited a minimal influence on LDM performance, with similar scores across most metrics. No single configuration consistently outperformed others across a majority of metrics, indicating no clear advantage for any specific codebook size.
- A notable trade-off was observed between image quality and sampling efficiency, with a visible Pareto frontier. A compression factor of 8 notably enhanced image diversity compared to factors of 4 and 16. While the number of channels in the latent space had a less pronounced effect than the compression factor, fewer channels were found to reduce the LDM's sensitivity to U-Net size. This characteristic is advantageous, particularly when considering subsequent model upscaling to larger 672 x 672 x 16 images.

Considering these results, the optimal model configuration for further upscaling was determined to be a VQ-GAN with a 1024-sized codebook, a compression factor of 8, 8 channels in the latent image, and a DDPM with a base of 512 channels.

5.1.1. VQ-GAN Outperforms VQ-VAE Across All Metrics

Table 5.1. This table presents the outcomes of preliminary experiments designed to assess the performance of a VQ-GAN autoencoder versus a VQ-VAE, with the primary distinction being the presence or absence of a discriminator. The evaluation hinges on three critical metrics: Spatial Quality (Hyperspectral FID), Diversity (MS-SSIM), and Sample Efficiency (Images/sec). Each autoencoder functions with a compression factor of 8, condensing image dimensions from 224x224x16 to 28x28x2 and employing 8 latent channels alongside 1024 codebook vectors for quantization. Moreover, each autoencoder utilizes K-means clustering to initialize the codebook at the start of every epoch. Additionally, the study investigates the effect of varying the base number of channels (128, 256, 512) in the DDPM to understand the influence of model size.

		Hyp. FID			**MS-SSIM-3D**				**Efficiency**
Model Type	**Base Ch.**	**Train**	**Recon**	**Synth**	**Val**	**Train**	**Recon**	**Synth**	**img/s**
gan	128	1.692	**4.682**	25.674	0.452	0.471	**0.474**	0.491	0.611
vae	128		7.005	26.035			0.493	0.506	0.597
gan	256		**4.682**	25.423			**0.474**	0.489	0.528
vae	256		7.005	26.297			0.493	0.503	0.527
gan	512		**4.682**	**24.358**			**0.474**	**0.486**	0.274
vae	512		7.005	27.333			0.493	0.494	0.273

Table 5.2. This table extends the analysis of the previous table 5.1 by presenting results on the spectral KL-divergence for veins, arteries, and the optic nerve head (ONH).

		KL-Div. Veins			**KL-Div. Arteries**			**KL-Div. ONH**		
Model Type	**Base Ch.**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**
gan	128	0.475	**0.814**	1.243	0.345	**0.537**	1.306	0.301	**0.9**	1.758
vae	128		2.89	4.588		2.781	5.048		3.207	4.505
gan	256		**0.814**	1.253		**0.537**	**1.304**		**0.9**	**1.659**
vae	256		2.89	4.394		2.781	4.882		3.207	4.338
gan	512		**0.814**	**1.234**		**0.537**	1.379		**0.9**	1.694
vae	512		2.89	4.404		2.781	4.763		3.207	4.162

The challenge of training GANs, notably due to their dual objective leading to potential instability, is well documented in the literature [**25, 57**]. In this context, our investigation aimed to assess if the GAN component in the VQ-GAN significantly enhances model performance compared to the simpler VQ-VAE. The rationale was straightforward: if the adversarial component could be omitted without performance degradation, it would streamline the training process, particularly beneficial for Optina in adapting models to new datasets efficiently.

Contrary to this hypothesis, our results, as detailed in Tables 5.1 and 5.2, unequivocally demonstrate the superior performance of VQ-GAN across all evaluated metrics. This superiority is evident in both spatial and spectral quality assessments, encompassing reconstructions and synthetic image generation, and is consistent across various sizes of the DDPM. Notably, the spectral quality analysis in Table 5.2 reveals a marked difference in performance, particularly in spectral quality. The Kullback-Leibler (KL) divergence for veins, arteries, and the ONH is substantially lower for VQ-GAN than VQ-VAE, with differences ranging from 2 to 3 fold in favor of VQ-GAN for both reconstructions and synthetic images.

Therefore, considering these findings, the inclusion of the discriminator in VQ-GAN is validated. Despite the operational simplicity of the likelihood-based VQ-VAE, the discriminator's role in significantly enhancing the realism of spectral signatures and overall model performance is evident and cannot be overlooked.

5.1.2. Minimal Impact of Codebook Size on Model Performance

Table 5.3. This table details the results of initial experiments aimed at determining the optimal number of codebook vectors (102, 1024, 10240) for a VQ-GAN autoencoder. The evaluation is anchored on three essential metrics: Spatial Quality (Hyperspectral FID), Diversity (MS-SSIM), and Sample Efficiency (Images/sec). In these experiments, each VQ-GAN autoencoder operates with a compression factor of 8, reducing image dimensions from 224x224x16 to 28x28x2, and uses 32 latent channels. The study explores the performance impact of varying the number of codebook vectors for quantization. Additionally, each autoencoder employs K-means clustering to initialize the codebook at the beginning of every epoch. The effect of different base channel numbers (128, 256, 512) in the Diffusion Denoising Probabilistic Model (DDPM) is also examined to assess the impact of model size.

		Hyp. FID			**MS-SSIM-3D**				**Efficiency**
Codebook Size	**Base Ch.**	**Train**	**Recon**	**Synth**	**Val**	**Train**	**Recon**	**Synth**	**img/s**
102	128	1.692	4.813	36.882	0.452	0.471	0.474	**0.471**	0.602
1024	128		**4.45**	26.779			**0.472**	0.487	0.604
10240	128		4.626	29.332			0.478	0.5	0.603
102	256		4.813	29.415			0.474	0.477	0.53
1024	256		**4.45**	**25.764**			**0.472**	0.497	0.527
10240	256		4.626	29.209			0.478	0.509	0.528
102	512		4.813	29.639			0.474	0.496	0.276
1024	512		**4.45**	27.348			**0.472**	0.48	0.274
10240	512		4.626	29.677			0.478	0.5	0.28

Table 5.4. This table extends the analysis of the previous table 5.3 by presenting results on the spectral KL-divergence for veins, arteries, and the optic nerve head (ONH).

		KL-Div. Veins			**KL-Div. Arteries**			**KL-Div. ONH**		
Codebook Size	**Base Ch.**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**
102	128	0.475	**0.803**	**1.244**	0.345	0.55	1.339	0.301	0.713	1.077
1024	128		0.884	1.296		0.634	1.311		0.972	1.141
10240	128		0.836	1.259		**0.518**	**1.292**		**0.656**	**0.856**
102	256		**0.803**	1.313		0.55	1.305		0.713	1.158
1024	256		0.884	1.453		0.634	1.55		0.972	1.377
10240	256		0.836	1.436		**0.518**	1.445		**0.656**	0.951
102	512		**0.803**	1.532		0.55	1.54		0.713	1.407
1024	512		0.884	1.261		0.634	1.337		0.972	1.202
10240	512		0.836	1.346		**0.518**	1.324		**0.656**	0.867

In vector-quantized models, the codebook size is a crucial hyperparameter, as it directly influences the information capacity of the latent space. The codebook, denoted as $\mathcal{Z}$, plays a central role in defining the number of potential latent spaces. The total number of these latent spaces is mathematically represented as:

$$|\mathcal{I}_{\mathcal{L}}| = |\mathcal{Z}|^{H_L \times W_L \times D_L} \tag{5.1.1}$$

Here, $\mathcal{I}_{\mathcal{L}}$ represents the set of all possible latent spaces, and H_L, W_L, and D_L correspond to the height, width, and depth of the latent space, respectively.

Despite the theoretical importance of the codebook size, our empirical investigation showed that variations in codebook size, ranging from 102 to 1024 and then to 10240, did not significantly impact the model's

performance. This is evident from Tables 5.3 and 5.4, where no clear performance differences are observed across different configurations. The metrics are closely matched for all three codebook sizes, in both reconstruction and synthetic image generation tasks. This finding is particularly noteworthy when juxtaposed with the significant improvements in all metrics, including a 2 to 3 fold increase in spectral quality, achieved by VQ-GAN over VQ-VAE (refer to Section 5.1.1).

Therefore, we deduce that the size of the codebook, $\mathcal{Z}$, while theoretically significant, did not demonstrate a substantial impact on performance in our experimental context. Hence, we maintained the default codebook size of 1024.

5.1.3. Identifying Optimal Latent Space Dimensions

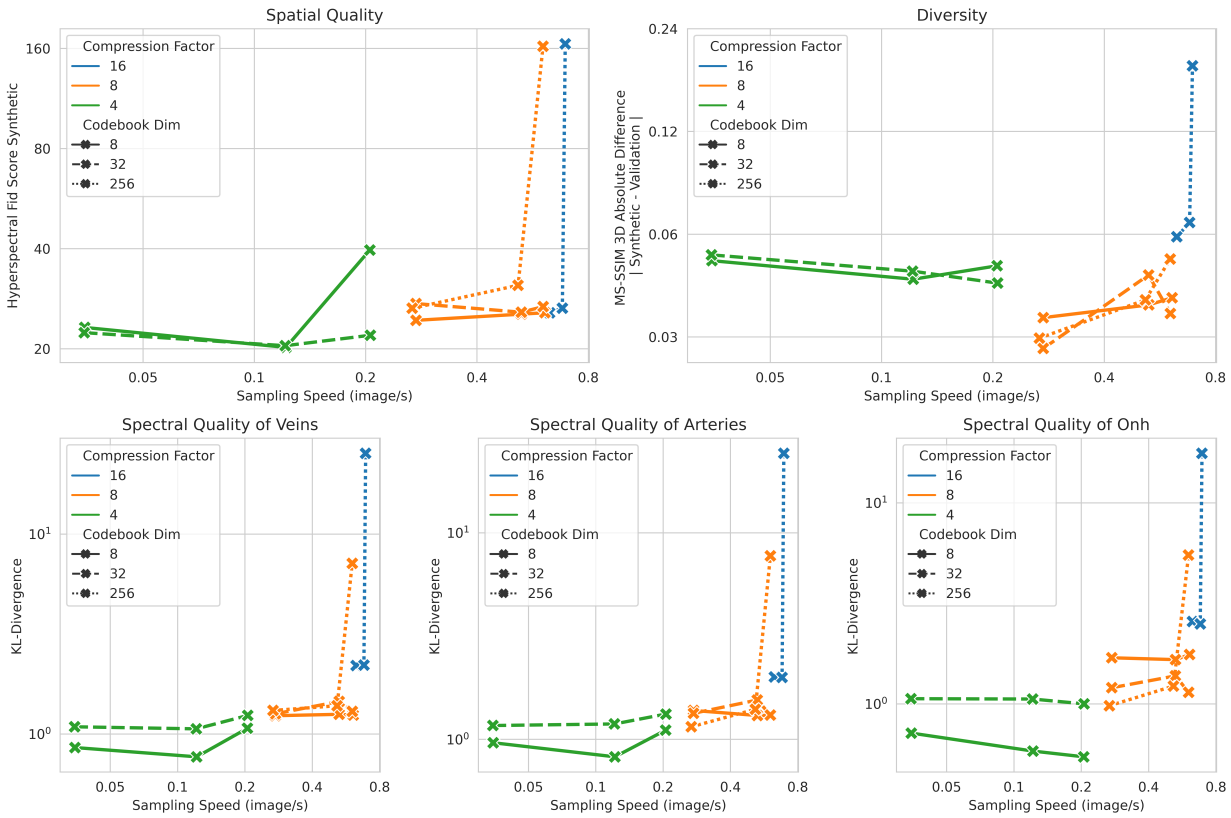


Fig. 5.1. Visualization of the synthetic results from the latent space size experiments detailed in Tables 5.5 and 5.6. This figure, focusing exclusively on synthetic images, omits reconstruction results and presents five plots. These plots illustrate key metrics: Hyperspectral FID score, absolute difference between validation and synthetic MS-SSIM, and Spectral KL-Divergence for veins, arteries, and the Optical Nerve Head (ONH), plotted against the model's sampling speed (images per second) on an NVIDIA A40 GPU. The architecture employs a VQ-GAN with a 1024-vector codebook and a three-level Diffusion U-Net, incorporating K-Means clustering in the VQ-GAN training phase. The experiments varied latent space sizes (compression factors of 16, 8, 4), the number of input channels in the latent image (8, 32, 256), and the base number of channels (128, 256, 512) in the DDPM. A Pareto front is evident across all metrics, indicating a trade-off between sampling efficiency and both spectral and spatial quality. Notably, quality improvements beyond models using a U-Net with 512 base channels and a compression factor of 8 are minimal, despite a marked decrease in sampling efficiency. Furthermore, models with these specific configurations exhibit superior diversity compared to other models.

Table 5.5. The primary objective of this experiment, as depicted in this table, is to ascertain the optimal latent space size for models operating on $16 \times 224 \times 224$ images. The evaluation is based on three critical metrics: Spatial Quality (Hyperspectral FID), Diversity (MS-SSIM), and Sample Efficiency (Images/sec). The underlying architecture includes a VQ-GAN with a 1024-vector codebook and a Diffusion U-Net with three levels. K-Means clustering is incorporated during the VQ-GAN training phase. The experiment varies the latent space size (compression factors of 16, 8, 4), the number of input channels in the latent image (8, 32, 256), and the base number of channels (128, 256, 512) in the Denoising Diffusion Probabilistic Model (DDPM) U-Net to determine the configuration that yields optimal performance.

		Hyp. FID			**MS-SSIM-3D**				**Efficiency**
Latent	**Base Ch.**	**Train**	**Recon**	**Synth**	**Val**	**Train**	**Recon**	**Synth**	**img/s**
(14, 14, 1, 256)	128	1.692	10.725	165.159	0.452	0.471	0.482	0.639	0.693
(14, 14, 1, 256)	256		10.725	26.482			0.482	0.517	0.681
(14, 14, 1, 256)	512		10.725	25.658			0.482	0.511	0.628
(28, 28, 2, 8)	128		4.682	25.674			0.474	0.491	0.611
(28, 28, 2, 8)	256		4.682	25.423			0.474	0.489	0.528
(28, 28, 2, 8)	512		4.682	24.358			0.474	0.486	0.274
(28, 28, 2, 32)	128		4.45	26.779			**0.472**	0.487	0.604
(28, 28, 2, 32)	256		4.45	25.764			**0.472**	0.497	0.527
(28, 28, 2, 32)	512		4.45	27.348			**0.472**	**0.48**	0.274
(28, 28, 2, 256)	128		5.155	162.377			0.473	0.401	0.604
(28, 28, 2, 256)	256		5.155	31.041			0.473	0.49	0.517
(28, 28, 2, 256)	512		5.155	26.478			0.473	0.482	0.267
(56, 56, 4, 8)	128		1.841	39.636			**0.472**	0.5	0.205
(56, 56, 4, 8)	256		1.841	**20.197**			**0.472**	0.496	0.122
(56, 56, 4, 8)	512		1.841	23.187			**0.472**	0.502	0.035
(56, 56, 4, 32)	128		**1.834**	21.966			0.477	0.495	0.206
(56, 56, 4, 32)	256		**1.834**	20.432			0.477	0.499	0.121
(56, 56, 4, 32)	512		**1.834**	22.364			0.477	0.504	0.035

Table 5.6. This table extends the analysis of the previous table 5.5 by presenting results on the spectral KL-divergence for veins, arteries, and the optic nerve head (ONH).

		KL-Div. Veins			**KL-Div. Arteries**			**KL-Div. ONH**		
Latent	**Base Ch.**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**
(14, 14, 1, 256)	128	0.475	1.661	25.346	0.345	1.374	24.245	0.301	1.819	17.636
(14, 14, 1, 256)	256		1.661	2.208		1.374	1.993		1.819	2.499
(14, 14, 1, 256)	512		1.661	2.199		1.374	2.005		1.819	2.566
(28, 28, 2, 8)	128		0.814	1.243		0.537	1.306		0.9	1.758
(28, 28, 2, 8)	256		0.814	1.253		0.537	1.304		0.9	1.659
(28, 28, 2, 8)	512		0.814	1.234		0.537	1.379		0.9	1.694
(28, 28, 2, 32)	128		0.884	1.296		0.634	1.311		0.972	1.141
(28, 28, 2, 32)	256		0.884	1.453		0.634	1.55		0.972	1.377
(28, 28, 2, 32)	512		0.884	1.261		0.634	1.337		0.972	1.202
(28, 28, 2, 256)	128		0.837	7.133		0.564	7.736		0.659	5.499
(28, 28, 2, 256)	256		0.837	1.383		0.564	1.397		0.659	1.225
(28, 28, 2, 256)	512		0.837	1.31		0.564	1.149		0.659	0.978
(56, 56, 4, 8)	128		0.608	1.068		**0.404**	1.107		**0.345**	**0.545**
(56, 56, 4, 8)	256		0.608	**0.768**		**0.404**	**0.823**		**0.345**	0.582
(56, 56, 4, 8)	512		0.608	0.855		**0.404**	0.963		**0.345**	0.715
(56, 56, 4, 32)	128		**0.585**	1.237		0.42	1.326		0.494	1.0
(56, 56, 4, 32)	256		**0.585**	1.06		0.42	1.187		0.494	1.056
(56, 56, 4, 32)	512		**0.585**	1.086		0.42	1.166		0.494	1.06

This subsection explores the optimal dimensions of the latent space in 224 x 224 x 16 resolution images by examining the effects of two critical hyperparameters: the compression factor and the number of channels in the latent space. We assessed three distinct compression factors—4, 8, and 16—and experimented with latent channel configurations of 8, 32, and 256.

The analysis of the compression factor reveals some intriguing trends. According to Tables 5.5 and 5.6, a compression factor of 4 generally enhances spatial and spectral quality compared to factors of 8 and 16. However, in terms of diversifying the image content, a factor of 8 emerges as the more effective choice. This observation suggests a trade-off between the model's fidelity to detail and its capacity for generating diverse images. A particularly noteworthy point is that a compression factor of 16 markedly reduces performance across all metrics, likely due to the loss of too much information during the VQ-GAN compression phase.

Delving deeper, we explored how this trade-off between quality and diversity manifests in terms of sampling efficiency. Illustrated in Figure 5.1, this exploration indicates that while there is a balance to be struck, the benefits of lower compression factors (such as 4) diminish in comparison to a factor of 8. Especially notable is the performance of larger DDPMs with 512 base channels in combination with a compression factor of 8, which results in the most favorable diversity outcomes.

Another aspect of our investigation was the impact of the number of channels in the latent space. While this factor did not have as pronounced an impact as the compression factor, the experiments showed that configurations with fewer channels are more adaptable to reductions in U-Net DDPM size. This finding is particularly relevant in the context of future scaling to higher-resolution images, where computational constraints might necessitate smaller neural networks.

In conclusion, after considering the various trade-offs, including sampling efficiency and the marginal quality gains beyond a compression factor of 8, quality and diversity, we identified that a model configuration with a compression factor of 8 and 8 latent channels is optimally suited for upscaling to larger image resolutions. This configuration aligns well with our objectives of balancing computational efficiency with the performance requirements of high-resolution image analysis.

5.2. Upscaling to 672 x 672 x 16 for High-Resolution Image Generation

Building upon the insights gained from our experiments with 224 x 224 x 16 resolution images, as discussed in Section 5.1, we progressed to upscale our model for high-resolution image generation. The optimal model configuration identified in these experiments—a VQ-GAN with a compression factor of 8, 8 channels in the latent space, a 1024-vector codebook, and a U-Net with 512 base channels—formed the foundation for this upscaling effort.

Our primary focus was on fine-tuning the VQ-GAN to adapt to higher resolution images, specifically 672 x 672 x 16. This endeavor proved to be successful, as the model generated images of high quality and diversity at this increased resolution. A critical adjustment in this process was the replacement of batch normalization with instance normalization. This change was necessitated by the switch to stochastic gradient descent, a decision driven by the larger image dimensions, which represented the maximum capacity manageable on Optina's GPUs.

The transition to higher resolutions and the consequent modifications in the model underline the scalability and adaptability of the chosen configuration. They demonstrate the model's ability to maintain performance and quality, even when confronted with the increased computational demands of larger image sizes.

5.2.1. Quantitative Assessment of Upscaled High-Resolution Models

Table 5.7. The goal of this experiment, as presented in this table, is to evaluate the optimal configuration for upscaling models to generate high-resolution $672 \times 672 \times 16$ images. The assessment utilizes three key metrics: Spatial Quality (Hyperspectral FID), Diversity (MS-SSIM), and Sample Efficiency (Images/sec). The architecture is consistent across models, featuring a VQ-GAN with a 1024-vector codebook and a Diffusion U-Net with three levels and a base number of 512 channels. This study introduces a novel asymmetric compression factor of (8,8,4), with the first figure representing spatial compression and the latter spectral, in addition to the standard factor of 8. The number of input channels in the latent space is always 8. The table compares models with varying normalization techniques: the first employs batch normalization, whereas the subsequent models use instance normalization and benefit from extended training durations as a result of increased VQ-GAN training stability.

			Hyp. FID			**MS-SSIM-3D**				**Efficiency**
Model	**Normalization**	**Latent**	**Train**	**Recon**	**Synth**	**Val**	**Train**	**Recon**	**Synth**	**img/s**
gan	BatchNorm	(84, 84, 2, 8)	1.692	3.542	**5.788**	0.452	0.471	0.498	0.521	0.032
gan*	InstanceNorm	(84, 84, 2, 8)		**2.06**	6.086			**0.475**	**0.468**	0.032
gan	InstanceNorm	(84, 84, 4, 8)		5.478	6.02			0.486	0.492	0.019

Table 5.8. This table extends the analysis of the previous table 5.7 by presenting results on the spectral KL-divergence for veins, arteries, and the optic nerve head (ONH).

			Veins			**Arteries**			**ONH**		
Model	**Normalization**	**Latent**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**
gan	BatchNorm	(84, 84, 2, 8)	0.475	2.905	3.031	0.346	2.657	2.996	0.301	3.219	3.486
gan*	InstanceNorm	(84, 84, 2, 8)		**0.769**	**1.869**		**0.529**	**1.104**		**0.379**	**1.081**
gan	InstanceNorm	(84, 84, 4, 8)		4.445	5.453		3.737	4.746		3.612	4.356

Our evaluation of the models upscaled to 672 x 672 x 16 resolutions involved a quantitative analysis using established metrics for spatial and spectral quality, along with diversity. The outcomes of this upscaling process are detailed in Tables 5.7 and 5.8.

Initially, when employing batch normalization in the VQ-GAN's discriminator for high-resolution models, the results appeared promising, indicated by a hyperspectral FID score of 5.788. However, a notable decrease in synthetic diversity and spectral quality was observed compared to the lower resolution models. This led to the hypothesis that batch normalization might not be well-suited for stochastic gradient descent, thereby hindering the VQ-GAN's optimization efficiency at higher resolutions. To address this issue, we switched to instance normalization, which has demonstrated effectiveness in style transfer tasks as per Huang et al. [**28**]. This modification resulted in improved performance across most metrics, with the exception of the hyperspectral FID score, which remained comparable to the previous score at 0.608. It is important to note, as highlighted in Section 3.2.1 and Stein et al. [**64**], that minor variations in FID scores may not always reliably indicate changes in image quality.

In an effort to further enhance model performance, we conducted an experiment with an asymmetric compression factor—8 for spatial dimensions and 4 for the spectral dimension. This approach was intended to infuse additional information into the spectral dimension, potentially enhancing spectral quality. However, this modification empirically led to instability in the VQ-GAN training, culminating in overall diminished model performance as evidenced in the corresponding entries in Tables 5.7 and 5.8. This outcome underscores the complexity of achieving a balance in model parameters, particularly in the context of higher-resolution image generation.

5.2.2. Qualitative Evaluation of the Optimal Upscaled Model

In addition to quantitative assessments, the qualitative evaluation of the upscaling process offers critical insights into the model's performance. This section discusses the synthetic images generated by the optimal model post-upscaling to 672 x 672 x 16 resolution.

Figure 5.2 presents a series of synthetic hyperspectral images at selected wavelengths. The remarkable similarity of these images to real hyperspectral images (as shown in figure 2.1) underscores the model's ability to replicate realistic spectral data. The nuances captured at various wavelengths, from 900 nm to 465 nm, attest to the model's fidelity in reproducing complex spectral characteristics.

The diversity in the synthetic dataset is further illustrated in Figure 5.5, where 36 hyperspectral images, converted to RGB format, are displayed. The variation in shades and features, such as the presence or absence of a visible choroid, highlights the model's capacity to avoid mode collapse, a common issue in generative models. This diversity not only demonstrates the model's robustness but also suggests its potential utility in augmenting self-supervised learning datasets, thereby enriching their variety and depth.

Figure 5.5 also includes a comparison of synthetic images with their nearest neighbors from the training set, as identified in the feature space of a MoCo-pretrained ResNet-50. The resemblance between synthetic and training set images confirms that the model effectively learned from the training data, yet without any overt signs of direct memorization. This balance between learning and creating novel representations is crucial for the effective use of synthetic data in self-supervised medical contexts. It should be emphasized that, for practical purposes, the 10 nearest neighbors were employed to evaluate the authenticity of images, with a more extensive set undergoing qualitative analysis. Nevertheless, to avoid excessive disclosure of Optina's confidential dataset, only 4 nearest neighbors for 5 representative images are presented in this report.

Lastly, Figure 5.5 offers a comparative analysis of spectral signatures, highlighting key eye structures such as veins, arteries, and the optical nerve head (ONH). The detailed visualization of these spectral signatures in synthetic versus validation datasets provides a compelling illustration of the model's capability in capturing and replicating intricate spectral features.

Together, these qualitative evaluations paint a comprehensive picture of the model's performance, demonstrating its effectiveness in generating high-quality, diverse, and realistic synthetic hyperspectral images. The balance achieved between quality, diversity, and realism underscores the model's potential for future applications in medical imaging and research.

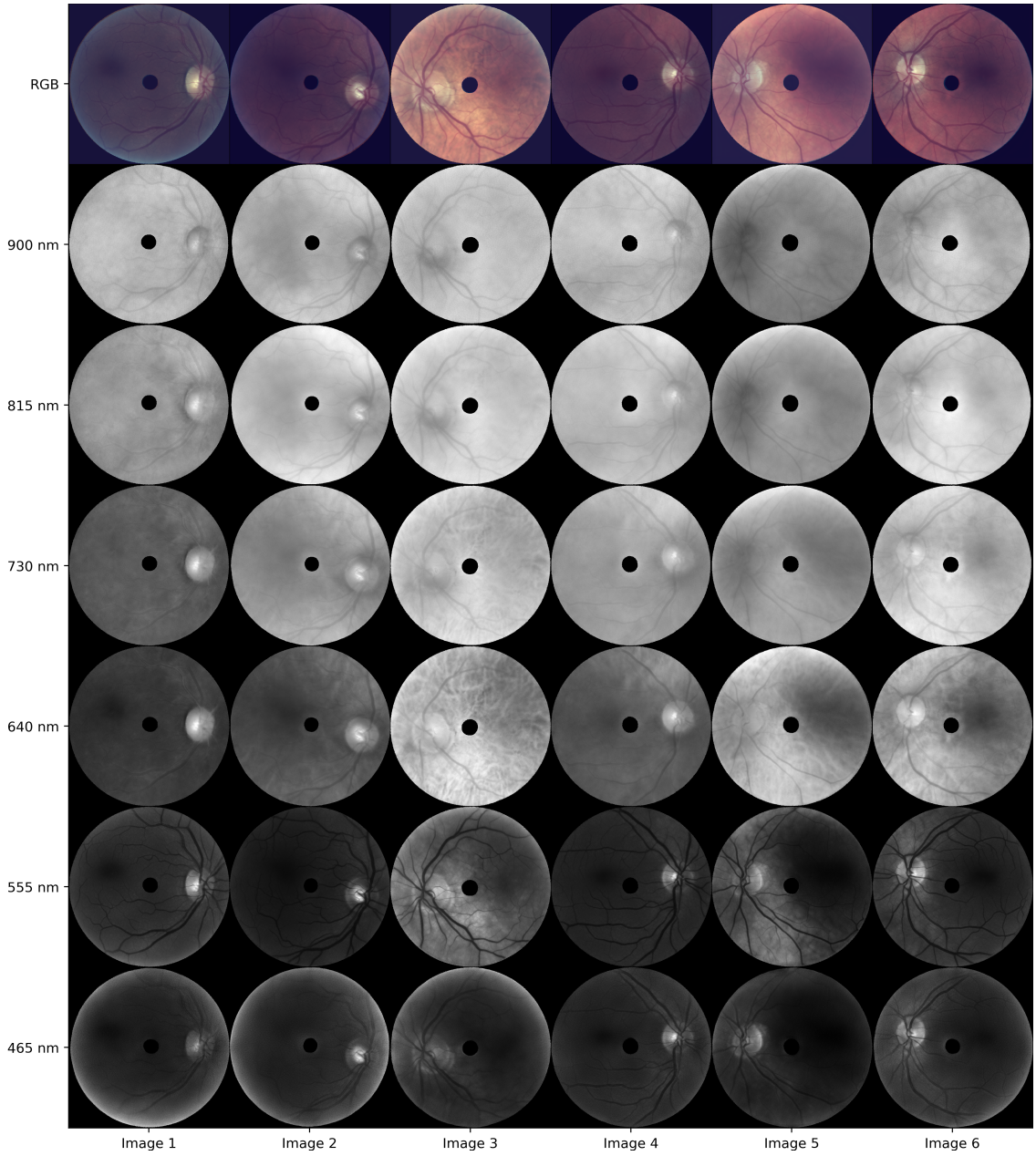


Fig. 5.2. This figure displays six synthetic images generated by the optimal model with a resolution of 672×672x16 pixels. Arranged from left to right, these images showcase the extraction of spectral data across eight specific wavelengths, enumerated from top to bottom as follows: 900 nm, 815 nm, 730 nm, 640 nm, 555 nm, and 465 nm. A comparison with the real images shown in figure 2.1 reveals that the synthetic images are almost indistinguishable from the real ones to the human eye, demonstrating the model's high level of realism.

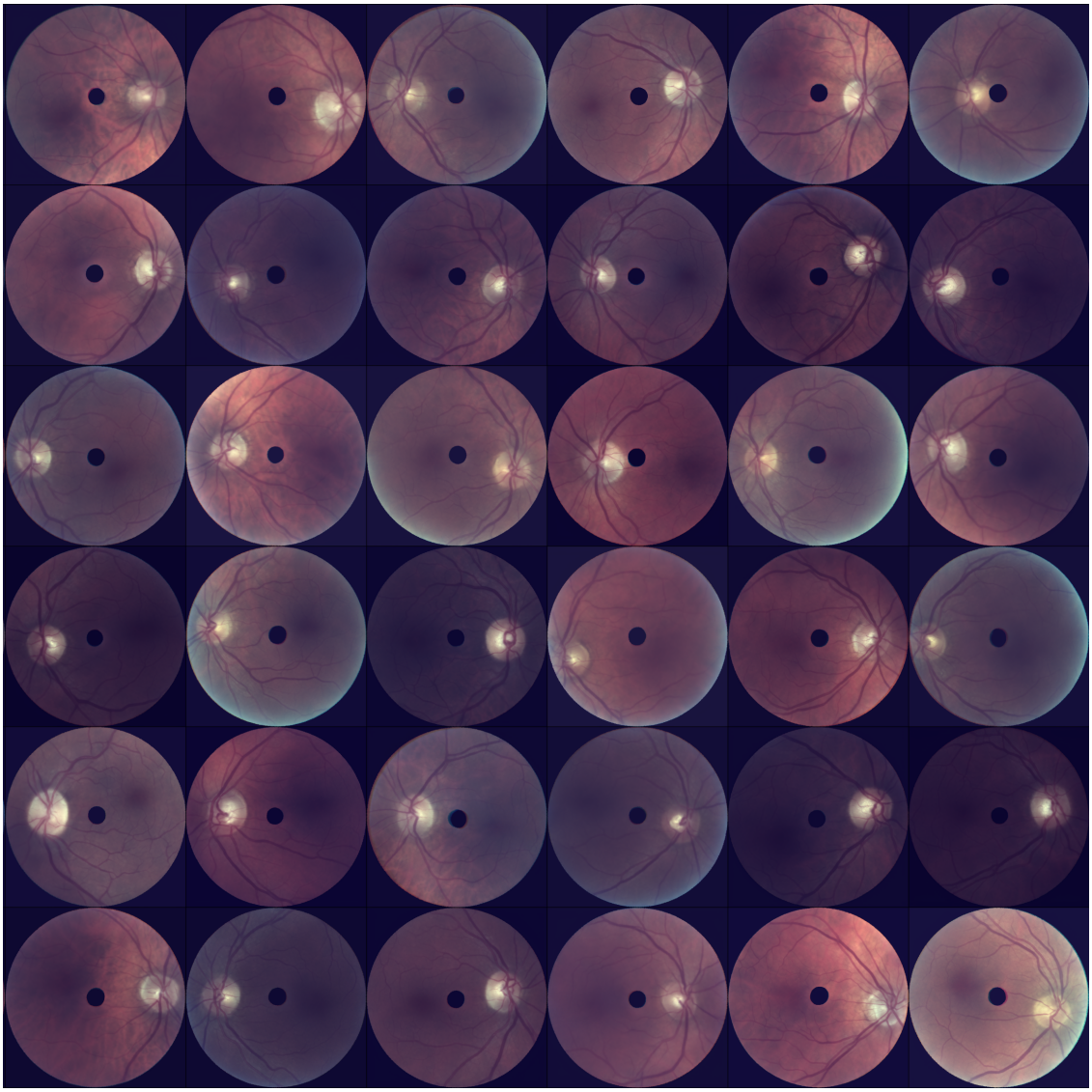


Fig. 5.3. Presented here are 36 hyperspectral images generated by the optimal model, subsequently projected to RGB for ease of visualization. The diversity observed in these images, including varied shades and the presence or absence of a visible choroid, suggests a low likelihood of mode collapse in the model. This variety indicates the potential utility of this synthetic dataset for augmenting self-supervised learning datasets, enhancing their richness and diversity.

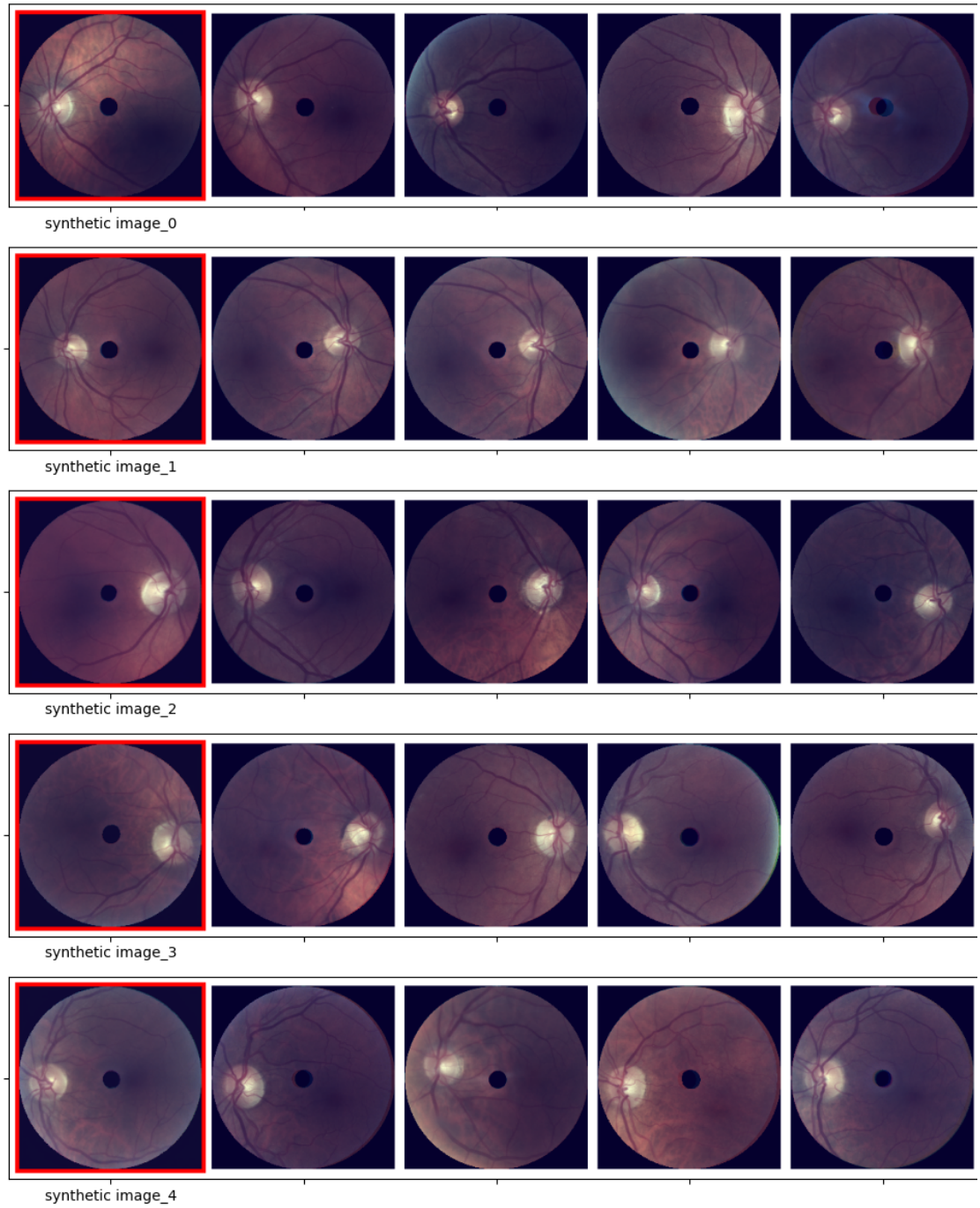


Fig. 5.4. This figure displays the four nearest neighbors from the training set for five synthetic images, as determined in the feature space of a MoCo-pretrained ResNet-50. While similarities between the synthetic and training set images are observable, indicating inspiration from the training data, there is no clear evidence of direct memorization. For confidentiality and compliance with HIPAA regulations, patient identifiers in the training set images have been obscured in this report.

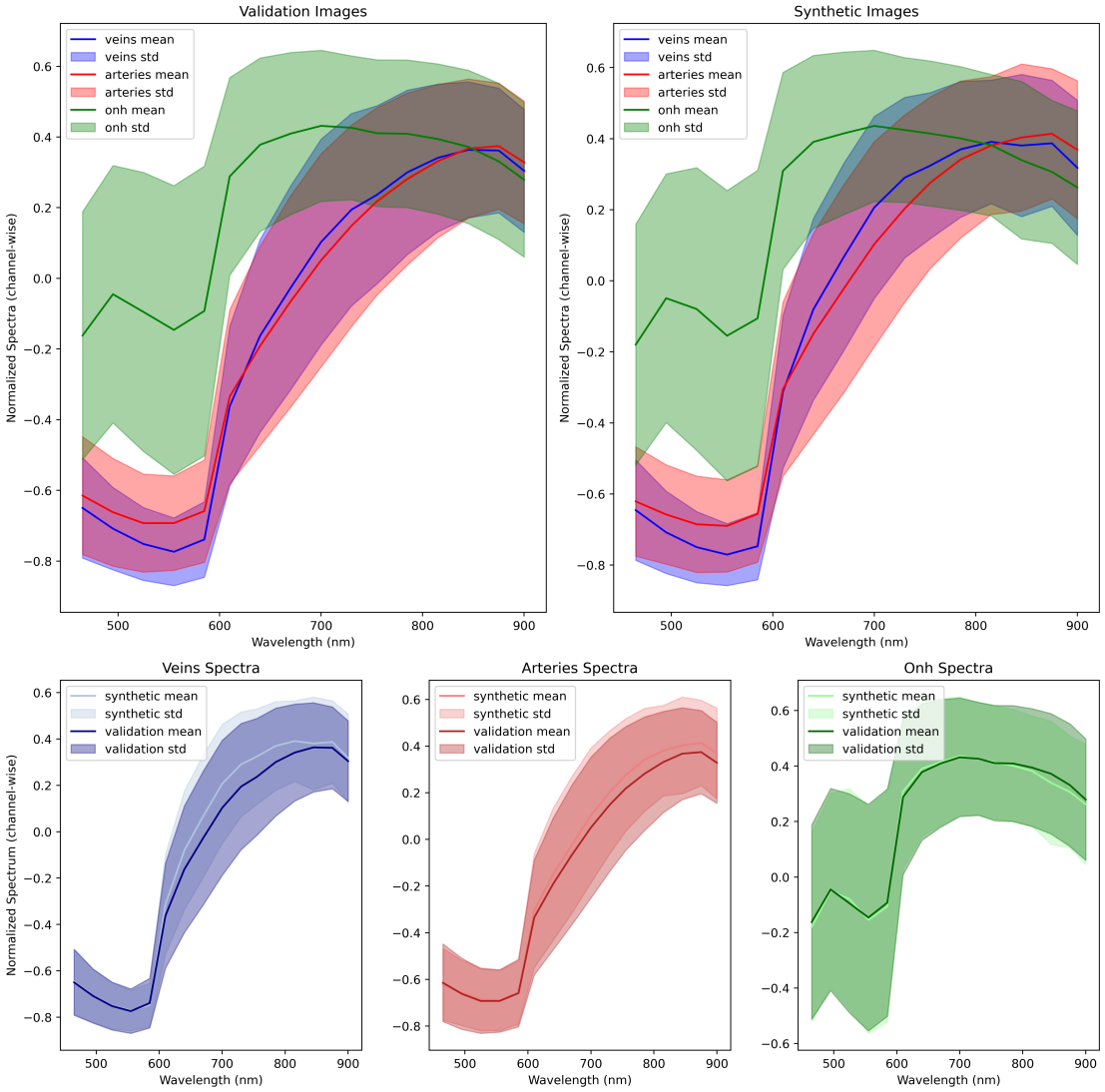


Fig. 5.5. Illustration of Spectral Signatures: Comparative Analysis of Synthetic and Validation Image Datasets. This figure presents a detailed comparison of spectral signatures, highlighting the eye structures of interest including veins, arteries, and the optical nerve head (ONH).

Chapter 6

Future Directions and Development

The successful conclusion of the internship, marked by the promising results achieved with the Latent Diffusion Model (LDM), has established a solid foundation for various future initiatives. Moving forward, our focus will be on harnessing these recent advancements and applying them to novel datasets and methodologies. The key areas targeted for future development include:

- **Training with Newly Arrived Unlabeled Images:** The recent acquisition of new unlabeled images presents an exciting opportunity. The forthcoming step entails training the LDM with this newly arrived batch of data. Such an endeavor will not only refine the model's performance but also broaden its application in the realm of self-supervised learning.
- **Enhancing MoCo with Synthetic Images:** A significant direction involves bolstering the MoCo self-supervised learning framework by incorporating synthetic images with unlabeled data. This amalgamation is anticipated to enrich the learning process, offering a more robust and diverse dataset for the MoCo algorithm to exploit.
- **Versatility in Self-Supervised Learning:** The application of synthetic images is not limited to MoCo. Plans are underway to utilize these images in various self-supervised learning algorithms, including Masked AutoEncoders and Pretext Tasks. The inherent diversity of synthetic images renders them ideal for experimenting with and enhancing these methodologies.
- **Conditioned Diffusion Model Development:** In future endeavors, we foresee the potential in developing a conditioned diffusion model capable of generating images based on specific Amyloid-β information. This approach could lead to a form of supervised pretraining, potentially yielding more accurate Amyloid-β classifications.
- **Anomaly Detection Pipeline Creation:** Leveraging the custom spectral quality assessment pipeline developed during the internship, we aim to construct an anomaly detection pipeline. This new pipeline will be crucial in identifying and discarding inferior synthetic images, thus ensuring the utmost quality and reliability of the generated data.

These prospective initiatives are not only expected to augment the capabilities of the current model but also to pave the way for groundbreaking applications of generative models in advancing Optina's objectives. The integration of sophisticated techniques, such as conditioned diffusion models and anomaly detection pipelines, promises to make significant contributions to the field, propelling advancements in both technological innovation and healthcare outcomes.

Chapter 7

Conclusion

During my internship at Optina Diagnostics, the primary goal was to develop high-resolution, high-quality synthetic hyperspectral retinal images to bolster a self-supervised learning pipeline, particularly a Momentum Contrast (MoCo) pretraining framework. The underlying hypothesis was anchored in the potential of image generative models, especially latent diffusion models (LDMs) as exemplified in "Medical Diffusion" by Khader et al. [**36**], to provide effective data augmentation in medical imaging contexts where data scarcity is a significant challenge.

Given the computational constraints and the necessity to converge rapidly on an effective solution, the initial phase of the project focused on lower resolution images (224 x 224 x 16). This approach allowed for determining an efficient architecture that captured essential details from the training images. Subsequently, the most promising configuration was fine-tuned on larger resolution images (672 x 672 x 16), the maximum size manageable within the available computational limits.

The evaluation of LDMs constituted a critical component of the methodology. Assessing the utility of the generated images required consideration of several key aspects: quality, diversity, and the risk of memorizing the training set. For Optina's specific use case, this involved evaluating both the spatial and spectral qualities. Spatial quality was assessed by adapting the Fréchet Inception Distance (FID) score to hyperspectral images. Spectral quality was measured for crucial eye structures—veins, arteries, and the optic nerve head (ONH)—by comparing the similarity between the spectral signatures of real and synthetic images using the Kullback-Leibler (KL) divergence. Diversity was quantified using the Multi-Scale Structural Similarity Index (MS-SSIM), and the risk of memorization was qualitatively ascertained by comparing synthetic images with their nearest neighbors in the training set.

Using the aforementioned methodology, a significant finding emerged from the implementation of the VQ-GAN model. This approach, which integrates an adversarial objective, was contrasted with the simpler VQ-VAE autoencoder framework. Remarkably, the VQ-GAN model demonstrated superior performance across all quantitative metrics, notably enhancing both spatial and spectral quality, as well as the diversity of the generated images. Conversely, the number of codebook vectors appeared to have a negligible effect on the model's performance.

Further optimization revealed that a compression factor of 8 across all dimensions (height, width, and spectral) presented an optimal balance between image quality, diversity, and sampling efficiency. A notable conclusion was the identification of an 8-channel dimension in the latent space as optimal. This configuration notably augmented the model's resilience in U-Net architectures with fewer parameters, a desirable characteristic when upscaling to larger image dimensions. Consequently, the final LDM configuration was

established as the largest possible diffusion U-Net along with a VQ-GAN autoencoder with a compression factor of 8, featuring 8 channels in the latent space, and a codebook of 1024 vectors.

Modifications to the model were strategically implemented during the transition to upscaling the images to 672 x 672 x 16 dimensions. This included substituting batch normalization with instance normalization in the upscaling process. This critical adjustment stemmed from computational constraints and the necessity for optimization using stochastic gradient descent. The efficacy of these modifications, particularly in the context of larger image dimensions, was substantiated by the model's performance metrics. Specifically, the model achieved a hyperspectral Fréchet Inception Distance (FID) score of 6.086 and a Multi-Scale Structural Similarity Index (MS-SSIM) score of 0.468. Furthermore, a qualitative analysis reaffirmed the high quality and diversity of the upscaled images and verified minimal memorization from the training dataset, underlining the success of these modifications in the context of larger image dimensions.

This research, focusing on high-resolution image generation via LDMs, constitutes a significant advancement for Optina Diagnostics' self-supervised learning initiatives. By generating a high-quality and diverse image dataset, this project lays the groundwork for its subsequent integration with labeled and unlabeled images in the MoCo pretraining phase. Ensuring the high quality of these images is pivotal, as their utility should not be hampered by quality constraints. The forthcoming phase of this project involves employing these synthetic images within the MoCo pretraining pipeline, with the anticipation that they will enhance the performance of the amyloid-β classification algorithm.

References

[1] Ahmed Alaa, Boris Van Breugel, Evgeny S Saveliev et Mihaela van der Schaar : How faithful is your synthetic data? sample-level metrics for evaluating and auditing generative models. *In International Conference on Machine Learning*, pages 290–306. PMLR, 2022.

[2] Brian DO Anderson : Reverse-time diffusion equation models. *Stochastic Processes and their Applications*, 12(3):313–326, 1982.

[3] Martin Arjovsky, Soumith Chintala et Léon Bottou : Wasserstein generative adversarial networks. *In International conference on machine learning*, pages 214–223. PMLR, 2017.

[4] Pratik Prabhanjan Brahma, Dapeng Wu et Yiyuan She : Why deep learning works: A manifold disentanglement perspective. *IEEE transactions on neural networks and learning systems*, 27(10):1997–2008, 2015.

[5] Gustav Bredell, Kyriakos Flouris, Krishna Chaitanya, Ertunc Erdil et Ender Konukoglu : Explicitly minimizing the blur error of variational autoencoders. *arXiv preprint arXiv:2304.05939*, 2023.

[6] Andrew Brock, Jeff Donahue et Karen Simonyan : Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018.

[7] Nicolas Carlini, Jamie Hayes, Milad Nasr, Matthew Jagielski, Vikash Sehwag, Florian Tramer, Borja Balle, Daphne Ippolito et Eric Wallace : Extracting training data from diffusion models. *In 32nd USENIX Security Symposium (USENIX Security 23)*, pages 5253–5270, 2023.

[8] Min Jin Chong et David Forsyth : Effectively unbiased fid and inception score and where to find them. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6070–6079, 2020.

[9] Chris Cremer, Xuechen Li et David Duvenaud : Inference suboptimality in variational autoencoders. *In International Conference on Machine Learning*, pages 1078–1086. PMLR, 2018.

[10] Jorge Cuadros et George Bresnick : Eyepacs: an adaptable telemedicine system for diabetic retinopathy screening. *Journal of diabetes science and technology*, 3(3):509–516, 2009.

[11] Prafulla Dhariwal et Alexander Nichol : Diffusion models beat gans on image synthesis. *Advances in neural information processing systems*, 34:8780–8794, 2021.

[12] Patrick Esser, Robin Rombach et Bjorn Ommer : Taming transformers for high-resolution image synthesis. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 12873–12883, 2021.

[13] Virginia Fernandez, Walter Hugo Lopez Pinaya, Pedro Borges, Petru-Daniel Tudosiu, Mark S Graham, Tom Vercauteren et M Jorge Cardoso : Can segmentation models be trained with fully synthetically generated data? *In International Workshop on Simulation and Synthesis in Medical Imaging*, pages 79–90. Springer, 2022.

[14] Maayan Frid-Adar, Idit Diamant, Eyal Klang, Michal Amitai, Jacob Goldberger et Hayit Greenspan : Gan-based synthetic medical image augmentation for increased cnn performance in liver lesion classification. *Neurocomputing*, 321:321–331, 2018.

[15] Ian Goodfellow : Nips 2016 tutorial: Generative adversarial networks. *arXiv preprint arXiv:1701.00160*, 2016.

[16] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville et Yoshua Bengio : Generative adversarial nets. *In Advances in neural information processing systems*, pages 2672–2680, 2014.

[17] Mark S Graham, Petru-Daniel Tudosiu, Paul Wright, Walter Hugo Lopez Pinaya, U Jean-Marie, Yee H Mah, James T Teo, Rolf Jager, David Werring, Parashkev Nachev *et al.* : Transformer-based out-of-distribution detection for clinically safe segmentation. *In International Conference on Medical Imaging with Deep Learning*, pages 457–476. PMLR, 2022.

[18] Shuyang Gu, Dong Chen, Jianmin Bao, Fang Wen, Bo Zhang, Dongdong Chen, Lu Yuan et Baining Guo : Vector quantized diffusion model for text-to-image synthesis. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10696–10706, 2022.

[19] John T Guibas, Tejpal S Virdi et Peter S Li : Synthetic medical images from dual generative adversarial networks. *arXiv preprint arXiv:1709.01872*, 2017.

[20] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin et Aaron C Courville : Improved training of wasserstein gans. *Advances in neural information processing systems*, 30, 2017.

[21] Changhee Han, Hideaki Hayashi, Leonardo Rundo, Ryosuke Araki, Wataru Shimoda, Shinichi Muramatsu, Yujiro Furukawa, Giancarlo Mauri et Hideki Nakayama : Gan-based synthetic brain mr image generation. *In 2018 IEEE 15th international symposium on biomedical imaging (ISBI 2018)*, pages 734–738. IEEE, 2018.

[22] Changhee Han, Leonardo Rundo, Ryosuke Araki, Yujiro Furukawa, Giancarlo Mauri, Hideki Nakayama et Hideaki Hayashi : Infinite brain mr images: Pggan-based data augmentation for tumor detection. *In Neural approaches to dynamics of signal exchanges*, pages 291–303. Springer, 2019.

[23] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie et Ross Girshick : Momentum contrast for unsupervised visual representation learning. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[24] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler et Sepp Hochreiter : Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems*, 30, 2017.

[25] Jonathan Ho, Ajay Jain et Pieter Abbeel : Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020.

[26] Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi et David J. Fleet : Video diffusion models, 2022.

[27] Sungmin Hong, Razvan Marinescu, Adrian V Dalca, Anna K Bonkhoff, Martin Bretzner, Natalia S Rost et Polina Golland : 3d-stylegan: A style-based generative adversarial network for generative modeling of three-dimensional medical images. *In Deep Generative Models, and Data Augmentation, Labelling, and Imperfections: First Workshop, DGM4MICCAI 2021, and First Workshop, DALI 2021, Held in Conjunction with MICCAI 2021, Strasbourg, France, October 1, 2021, Proceedings 1*, pages 24–34. Springer, 2021.

[28] Xun Huang et Serge Belongie : Arbitrary style transfer in real-time with adaptive instance normalization. *In Proceedings of the IEEE international conference on computer vision*, pages 1501–1510, 2017.

[29] Eric Jang, Shixiang Gu et Ben Poole : Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

[30] Marco Jiralerspong, Avishek Joey Bose et Gauthier Gidel : Feature likelihood score: Evaluating generalization of generative models using samples. *arXiv preprint arXiv:2302.04440*, 2023.

[31] Justin Johnson, Alexandre Alahi et Li Fei-Fei : Perceptual losses for real-time style transfer and super-resolution. *In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 694–711. Springer, 2016.

[32] Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola et Lawrence K Saul : An introduction to variational methods for graphical models. *Machine learning*, 37:183–233, 1999.

[33] Steffen Jung et Margret Keuper : Internalized biases in fréchet inception distance. *In NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.

[34] Rajendra S Kadam, Narayan PS Cheruvu, Henry F Edelhauser et Uday B Kompella : Sclera-choroid-rpe transport of eight β-blockers in human, bovine, porcine, rabbit, and rat models. *Investigative ophthalmology & visual science*, 52(8):5387–5399, 2011.

[35] Amirhossein Kazerouni, Ehsan Khodapanah Aghdam, Moein Heidari, Reza Azad, Mohsen Fayyaz, Ilker Hacihaliloglu et Dorit Merhof : Diffusion models for medical image analysis: A comprehensive survey. *arXiv preprint arXiv:2211.07804*, 2022.

[36] Firas Khader, Gustav Mueller-Franzes, Soroosh Tayebi Arasteh, Tianyu Han, Christoph Haarburger, Maximilian Schulze-Hagen, Philipp Schad, Sandy Engelhardt, Bettina Baessler, Sebastian Foersch *et al.* : Medical diffusion–denoising diffusion probabilistic models for 3d medical image generation. *arXiv preprint arXiv:2211.03364*, 2022.

[37] Firas Khader, Gustav Müller-Franzes, Soroosh Tayebi Arasteh, Tianyu Han, Christoph Haarburger, Maximilian Schulze-Hagen, Philipp Schad, Sandy Engelhardt, Bettina Baessler, Sebastian Foersch *et al.* : Denoising diffusion probabilistic models for 3d medical image generation. *Scientific Reports*, 13(1):7303, 2023.

[38] Diederik P Kingma et Max Welling : Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[39] Diederik P Kingma, Max Welling *et al.* : An introduction to variational autoencoders. *Foundations and Trends® in Machine Learning*, 12(4):307–392, 2019.

[40] Durk P Kingma, Tim Salimans, Rafal Jozefowicz, Xi Chen, Ilya Sutskever et Max Welling : Improved variational inference with inverse autoregressive flow. *Advances in neural information processing systems*, 29, 2016.

[41] Wen Li, Ying Zhang, Yifang Sun, Wei Wang, Mingjie Li, Wenjie Zhang et Xuemin Lin : Approximate nearest neighbor search on high dimensional data—experiments, analyses, and improvement. *IEEE Transactions on Knowledge and Data Engineering*, 32(8):1475–1488, 2019.

[42] Jae Hyun Lim et Jong Chul Ye : Geometric gan. *arXiv preprint arXiv:1705.02894*, 2017.

[43] Calvin Luo : Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*, 2022.

[44] Qing Lyu et Ge Wang : Conversion between ct and mri images using diffusion and score-matching models. *arXiv preprint arXiv:2209.12104*, 2022.

[45] Siwei Lyu et Eero Simoncelli : Statistical modeling of images with fields of gaussian scale mixtures. *Advances in Neural Information Processing Systems*, 19, 2006.

[46] Casey Meehan, Kamalika Chaudhuri et Sanjoy Dasgupta : A non-parametric test to detect data-copying in generative models. *In International Conference on Artificial Intelligence and Statistics*, 2020.

[47] Takeru Miyato, Toshiki Kataoka, Masanori Koyama et Yuichi Yoshida : Spectral normalization for generative adversarial networks. *arXiv preprint arXiv:1802.05957*, 2018.

[48] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby *et al.* : Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[49] Walter HL Pinaya, Petru-Daniel Tudosiu, Jessica Dafflon, Pedro F Da Costa, Virginia Fernandez, Parashkev Nachev, Sebastien Ourselin et M Jorge Cardoso : Brain imaging generation with latent diffusion models. *In MICCAI Workshop on Deep Generative Models*, pages 117–126. Springer, 2022.

[50] Walter HL Pinaya, Petru-Daniel Tudosiu, Robert Gray, Geraint Rees, Parashkev Nachev, Sebastien Ourselin et M Jorge Cardoso : Unsupervised brain imaging 3d anomaly detection and segmentation with transformers. *Medical Image Analysis*, 79:102475, 2022.

[51] Moritz Platscher, Jonathan Zopes et Christian Federau : Image translation for medical image generation: Ischemic stroke lesion segmentation. *Biomedical Signal Processing and Control*, 72:103283, 2022.

[52] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen et Ilya Sutskever : Zero-shot text-to-image generation. *In International Conference on Machine Learning*, pages 8821–8831. PMLR, 2021.

[53] Ali Razavi, Aaron Van den Oord et Oriol Vinyals : Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

[54] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser et Björn Ommer : High-resolution image synthesis with latent diffusion models. *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10684–10695, 2022.

[55] Muhammad Muneeb Saad, Mubashir Husain Rehmani et Ruairi O'Reilly : Addressing the intra-class mode collapse problem using adaptive input image normalization in gan-based x-ray images. *In 2022 44th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pages 2049–2052. IEEE, 2022.

[56] Muhammad Muneeb Saad, Mubashir Husain Rehmani et Ruairi O'Reilly : Assessing intra-class diversity and quality of synthetically generated images in a biomedical and non-biomedical setting. *arXiv preprint arXiv:2308.02505*, 2023.

[57] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford et Xi Chen : Improved techniques for training gans. *Advances in neural information processing systems*, 29, 2016.

[58] Tim Salimans, Andrej Karpathy, Xi Chen et Diederik P Kingma : Pixelcnn++: Improving the pixelcnn with discretized logistic mixture likelihood and other modifications. *arXiv preprint arXiv:1701.05517*, 2017.

[59] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan et Surya Ganguli : Deep unsupervised learning using nonequilibrium thermodynamics. *In International conference on machine learning*, pages 2256–2265. PMLR, 2015.

[60] Gowthami Somepalli, Vasu Singla, Micah Goldblum, Jonas Geiping et Tom Goldstein : Diffusion art or digital forgery? investigating data replication in diffusion models. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6048–6058, 2023.

[61] Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby et Ole Winther : Ladder variational autoencoders. *Advances in neural information processing systems*, 29, 2016.

[62] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon et Ben Poole : Score-based generative modeling through stochastic differential equations. *arXiv preprint arXiv:2011.13456*, 2020.

[63] Akash Srivastava, Lazar Valkov, Chris Russell, Michael U Gutmann et Charles Sutton : Veegan: Reducing mode collapse in gans using implicit variational learning. *Advances in neural information processing systems*, 30, 2017.

[64] George Stein, Jesse C Cresswell, Rasa Hosseinzadeh, Yi Sui, Brendan Leigh Ross, Valentin Villecroze, Zhaoyan Liu, Anthony L Caterini, J Eric T Taylor et Gabriel Loaiza-Ganem : Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. *arXiv preprint arXiv:2306.04675*, 2023.

[65] Li Sun, Junxiang Chen, Yanwu Xu, Mingming Gong, Ke Yu et Kayhan Batmanghelich : Hierarchical amortized gan for 3d high resolution medical image synthesis. *IEEE journal of biomedical and health informatics*, 26(8):3966–3975, 2022.

[66] Yucheng Tang, Dong Yang, Wenqi Li, Holger R Roth, Bennett Landman, Daguang Xu, Vishwesh Nath et Ali Hatamizadeh : Self-supervised pre-training of swin transformers for 3d medical image analysis. *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 20730–20740, 2022.

[67] John Thickstun : Kantorovich-rubinstein duality, 2019.

[68] Arash Vahdat, Karsten Kreis et Jan Kautz : Score-based generative modeling in latent space. *Advances in Neural Information Processing Systems*, 34:11287–11302, 2021.

[69] Aaron Van Den Oord, Oriol Vinyals *et al.* : Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

[70] Liwei Wang, Yan Zhang et Jufu Feng : On the euclidean distance of images. *IEEE transactions on pattern analysis and machine intelligence*, 27(8):1334–1339, 2005.

[71] Zhou Wang, Alan C Bovik, Hamid R Sheikh et Eero P Simoncelli : Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[72] Zhou Wang, Eero P Simoncelli et Alan C Bovik : Multiscale structural similarity for image quality assessment. *In The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003*, volume 2, pages 1398–1402. Ieee, 2003.

[73] Max Welling et Yee W Teh : Bayesian learning via stochastic gradient langevin dynamics. *In Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688, 2011.

[74] Lilian Weng : What are diffusion models?, 2021.

[75] Xide Xia et Brian Kulis : W-net: A deep model for fully unsupervised image segmentation. *arXiv preprint arXiv:1711.08506*, 2017.

[76] Neil Zeghidour, Alejandro Luebs, Ahmed Omran, Jan Skoglund et Marco Tagliasacchi : Soundstream: An end-to-end neural audio codec. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 30:495–507, 2021.

[77] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman et Oliver Wang : The unreasonable effectiveness of deep features as a perceptual metric. *In Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018.

Appendix A

Supplementary Experiments for Methodology Justification

A.1. Optimal RGB Extractions per Hyperspectral Image for FID Assessment

To ascertain the optimal number of RGB extractions (N) per hyperspectral image for hyperspectral FID score computation, we drew inspiration from the work of Chong and Forsyth. As elaborated in Section 3.2.1 of the literature review, their work demonstrated a linear relationship between the FID score and the reciprocal of the number of images, underscoring the necessity for a sufficiently large image dataset to achieve a relatively unbiased FID score [**8**]. To identify the ideal N, the FID score was computed using $N = 1, 5, 10, 25, 50, 100, 200$, and a linear regression was performed to fit a line between the hyperspectral FID and $\frac{1}{N}$. The data points exhibited the anticipated linear trend. Notably, negligible differences were observed between $N = 100$ and $N = 200$, indicating that $N = 100$ provides a judicious balance between computational efficiency and a relatively unbiased score.

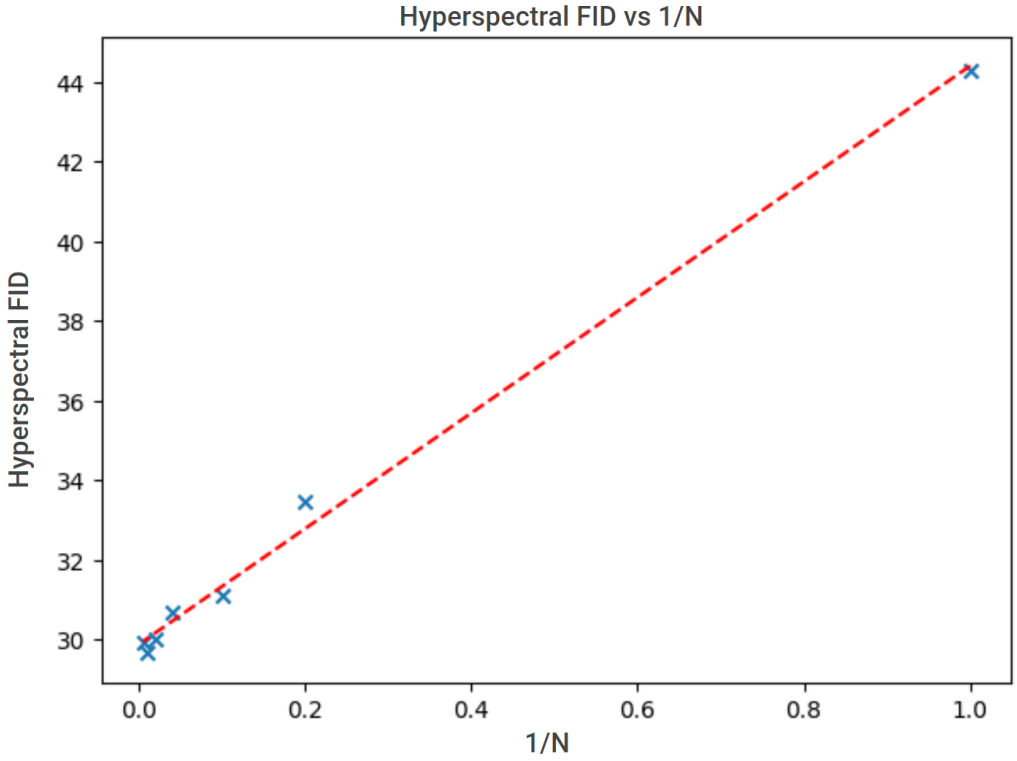


Fig. A.1. Linear relationship between the hyperspectral FID and $\frac{1}{N}$, illustrated using data points for $N = 1, 5, 10, 25, 50, 100, 200$. The similarity in results between $N = 100$ and $N = 200$ suggests that $N = 100$ is optimal for balancing reduced bias with reasonable computational time.

A.2. Post-Processing of Segmentation to Maximize Precision

In the context of evaluating the spectral quality using the W-Net architecture, precision is prioritized over sensitivity. This approach is due to the specific requirement of accurately sampling pixels from veins, arteries, and the Optic Nerve Head (ONH). Consequently, the primary focus is on ensuring the correctness of pixel classification, rather than exhaustive pixel inclusion.

The post-processing strategy incorporated the following methods:

- Implementation of a probability threshold set at 0.95. This threshold was empirically determined to maximize precision, while also maintaining a substantial sample size.

- Application of aggressive Region of Interest (ROI) cropping. This step involved excluding predictions near the center and periphery of the circular retinal images, areas identified as prone to less reliable predictions.

These post-processing measures led to a substantial increase in overall precision, as evidenced in the validation dataset. Specifically, the precision improved from 0.72 ± 0.08 to 0.87 ± 0.07. A breakdown by segmentation categories reveals that vein segmentation achieved a precision of 0.97 ± 0.07, while the precision for arteries and ONH observed a slightly lower, yet significantly improved, precision of 0.87 ± 0.1 and 0.81 ± 0.11, respectively.

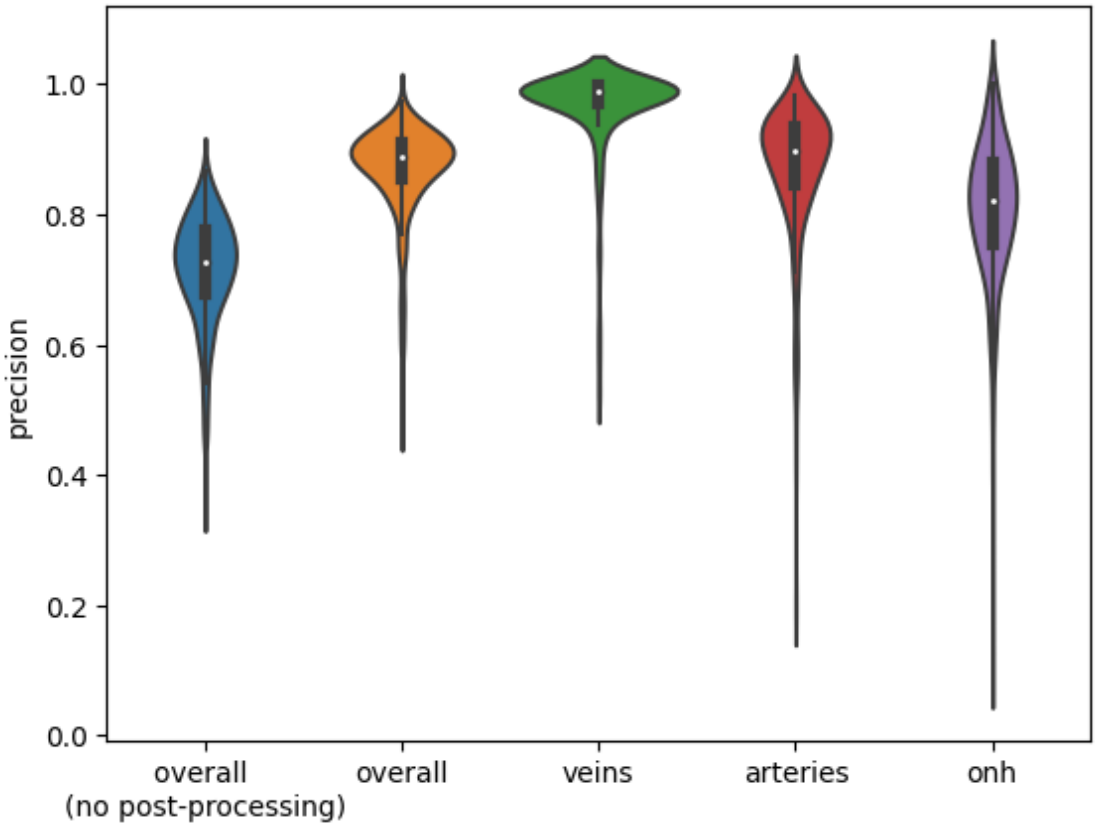


Fig. A.2. Illustration of the impact of post-processing on W-Net prediction precision. The raw segmentation yields an overall precision of 0.72 ± 0.08, which is notably enhanced to 0.87 ± 0.07 through post-processing. In particular, vein segmentation precision escalates to 0.97 ± 0.07, while artery and ONH segmentation precision also exhibit marked improvements, achieving 0.87 ± 0.1 and 0.81 ± 0.11, respectively. These results underscore the effectiveness of post-processing in enhancing segmentation precision.

A.3. MoCo-ResNet-50 Leads in Memorization Assessment

This analysis quantifies the ability of various models to assess the risk of memorization in synthetic images. It focuses on utilizing k-nearest neighbors (k-NN) search on the feature maps proposed by these models to evaluate memorization. The configurations tested include pixel space, random weights ResNet-18, random weights ResNet-50, ImageNet-pretrained ResNet-18, ImageNet-pretrained ResNet-50, and a MoCo-pretrained on EyePacs ResNet-50.

The ResNet-50 pretrained on EyePacs using MoCo (Momentum Contrast) demonstrated the highest proficiency in recovering original images from transformed versions. The recovery rate, a key metric in this context, is computed by identifying the transformed images and then assessing the model's ability to recover the nearest neighbors accurately. For each transformed image, typically corresponding to 2 to 3 images in the dataset, the model retrieves the 10 nearest neighbors, with the recovery rate calculated based on the proportion of correctly identified original images.

The MoCo-pretrained ResNet-50, particularly when combined with the Manhattan distance metric, exhibited a superior recovery rate of 65%. It's noteworthy that only ResNet-50 was tested with MoCo, as it was the only available pretraining checkpoint. However, considering the observed performance where ResNet-18 outperformed ResNet-50 in other training schemes (likely due to its smaller feature vector offering resistance to the curse of dimensionality), exploring a MoCo-pretrained ResNet-18 could potentially yield improved results in future studies focusing on memorization assessment in synthetic image generation.

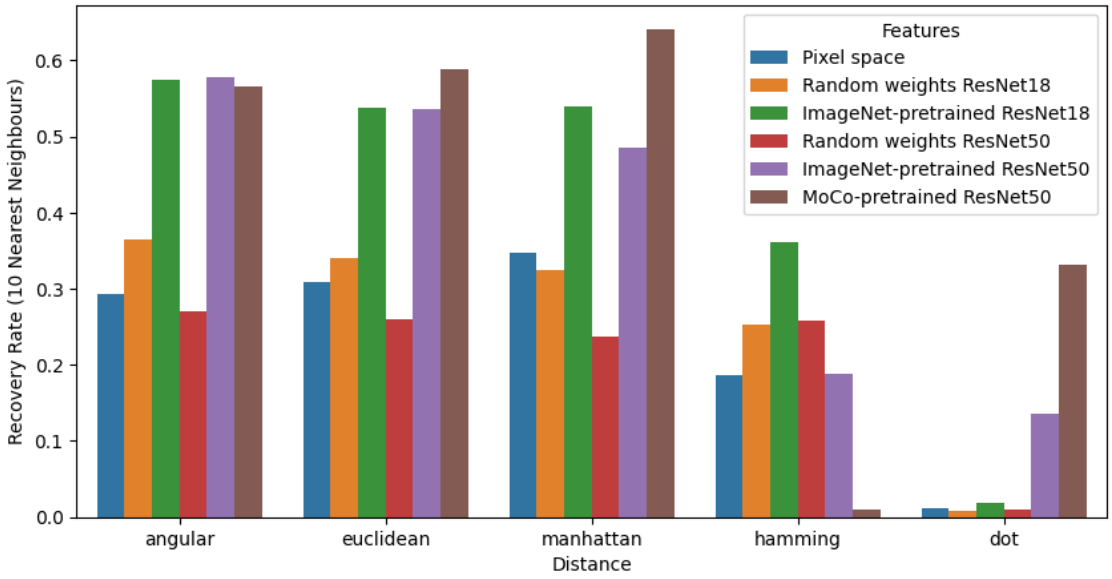


Fig. A.3. Recovery rate comparison across various model configurations. This figure highlights the effectiveness of the MoCo-pretrained ResNet-50, especially when using the Manhattan distance, in achieving a 65% recovery rate. The potential for enhanced performance with a MoCo-pretrained ResNet-18 in future memorization assessments is also suggested.

A.4. Epoch-wise K-means Codebook Re-Initialization in VQ-Models

Implementing K-means codebook initialization at the onset of each training epoch was implemented to stabilizie the training of the VQ-GAN. This approach, drawing inspiration from Zeghidour et al.'s SoundStream [**76**], distinctively applies the K-means algorithm at each epoch's start using a queue of training batches, diverging from traditional methods.

Figure A.4 indicates that the perplexity is initially much higher when K-means codebook initialization is applied, which corresponds to expectations of increased codebook diversity. In contrast, when K-means is not used, perplexity is not only lower at the start but also concludes higher, lacking a plateau even after 2500 epochs. The K-means-initialized model, on the other hand, demonstrates a steady-state perplexity, finding equilibrium around 1000 epochs. This suggests a balancing effect of K-means on the training dynamics.

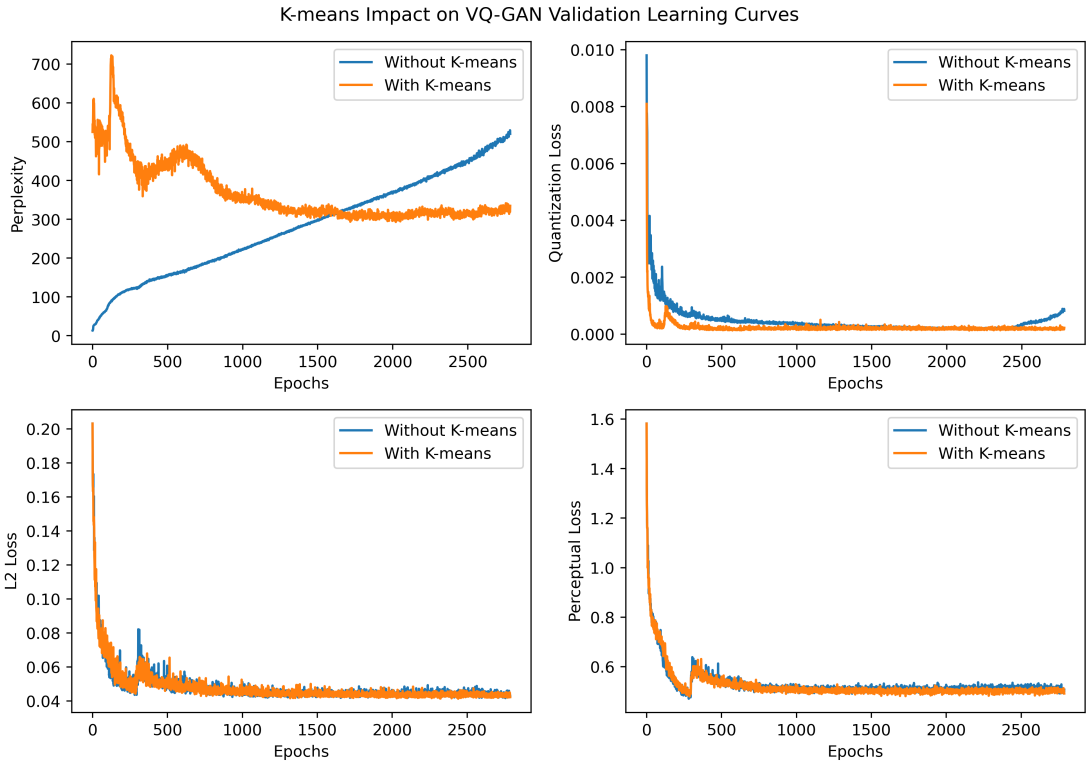


Fig. A.4. Impact of K-means Codebook Initialization on VQ-GAN Validation Learning Curves: A Comparative Analysis of Perplexity, Quantization Loss, L2 Loss, and Perceptual Loss Across Training Epochs.

Moreover, the model incorporating K-means consistently exhibits lower quantization loss, implying improved training stability. This is particularly evident when considering the uptick in quantization loss observed in the non-K-means model towards the end of the training period, as depicted in Figure A.4. However, the relatively minor differences in L2 and perceptual losses between the models do not decisively point to the superiority of the K-means approach in those aspects.

The impact of K-means codebook initialization was further examined through three differently sized diffusion models, each utilizing a three-level Diffusion U-Net with progressively doubling channel numbers at each level. The base channel numbers for these models were set at 128, 256, and 512, respectively, to evaluate the effect of model size. The outcomes, documented in tables A.1 and A.2, present a nuanced picture. While K-means initialization seemingly improves reconstruction quality (lower hyperspectral FID), it does not consistently translate into superior FID scores for synthetic images. Divergent trends are noted in diversity metrics (MS-SSIM), with reconstructions appearing more diverse without K-means, yet synthetic image diversity remains comparable between the two setups. The spectral quality results, measured through KL-divergence across different ocular structures (veins, arteries, and optical nerve heads), also exhibit marginal differences, underscoring that K-means initialization does not significantly alter performance upon proper model convergence.

However, given the observed enhancement in training stability with K-means, as evidenced by the learning curve behaviors, this initialization approach was adopted for subsequent experiments.

Table A.1. The primary goal of this experiment, as illustrated in this table, is to assess the impact of K-means codebook initialization in a VQ-GAN on images of size $16 \times 224 \times 224$. The evaluation focuses on three key metrics: Spatial Quality (Hyperspectral FID), Diversity (MS-SSIM), and Sample Efficiency (Images/sec). The experimental setup includes a VQ-GAN with a fixed 1024-vector codebook and a three-level Diffusion U-Net. The U-Net's configuration varies in terms of the base number of channels (128, 256, 512). This study compares the performance of the VQ-GAN with and without K-means codebook initialization at the beginning of each training epoch, maintaining a constant compression factor of 8 and 32 latent input channels, to determine the influence of K-means initialization on the model's effectiveness.

		Hyp. FID			**MS-SSIM-3D**				**Efficiency**
Base Ch.	**K-means**	**Train**	**Recon**	**Synth**	**Val**	**Train**	**Recon**	**Synth**	**img/s**
128	No	1.692	5.294	37.424	0.452	0.471	**0.467**	**0.481**	0.871
128	Yes		**4.012**	37.152			0.487	0.484	0.869
256	No		5.294	**27.413**			**0.467**	**0.481**	0.699
256	Yes		**4.012**	27.827			0.487	0.486	0.701
512	No		5.294	28.611			**0.467**	0.484	0.302
512	Yes		**4.012**	28.248			0.487	0.482	0.301

Table A.2. This table extends the analysis of the previous table A.1 by presenting results on the spectral KL-divergence for veins, arteries, and the optic nerve head (ONH).

		Veins			**Arteries**			**ONH**		
Base Ch.	**K-means**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**
128	No	0.475	**0.709**	1.491	0.346	**0.487**	1.569	0.301	0.525	0.86
128	Yes		0.849	1.528		0.557	1.688		**0.453**	0.867
256	No		**0.709**	1.259		**0.487**	1.33		0.525	**0.755**
256	Yes		0.849	**1.253**		0.557	**1.222**		**0.453**	0.84
512	No		**0.709**	1.275		**0.487**	1.367		0.525	0.841
512	Yes		0.849	1.315		0.557	1.249		**0.453**	0.828

Appendix B

Supplementary Results

B.1. Detailed Results Tables

Table B.1. This table encapsulates the entirety of the experiments conducted during the internship, primarily focusing on models with $16 \times 224 \times 224$ images for efficient training and model configuration identification. The most promising models were subsequently fine-tuned on larger $16 \times 672 \times 672$ images. The experiments included comparisons between VQ-VAE and VQ-GAN, analysis of the latent space compression factor, variations in the number of channels in the latent image, the quantity of codebook vectors in the VQ model, the base number of channels in the DDPM U-Net, the levels in the U-Net, and the incorporation of K-Means for codebook vector reinitialization in VQ-GAN training. The table presents results based on three critical metrics: Spatial Quality (Hyperspectral FID), Diversity (MS-SSIM), and Sample Efficiency (Images/sec). The architecture employed includes a VQ-GAN with a 1024-vector codebook and a Diffusion U-Net. The primary aim was to ascertain the optimal configuration for various parameters, including the latent space size, number of input feature maps in the latent image, and the base number of channels in the DDPM U-Net.

							Hyp. FID			MS-SSIM-3D				Efficiency
Model	**Res.**	**Latent**	**Codes**	**Base Ch.**	**Ch. Mult.**	**K-means**	**Train**	**Recon**	**Synth**	**Val**	**Train**	**Recon**	**Synth**	**img/s**
gan	(224, 224, 16)	(14, 14, 1, 256)	1024	128	1, 2, 4	yes	1.692	10.725	165.159	0.452	0.471	0.482	0.639	0.693
gan	(224, 224, 16)	(14, 14, 1, 256)	1024	256	1, 2, 4	yes		10.725	26.482			0.482	0.517	0.681
gan	(224, 224, 16)	(14, 14, 1, 256)	1024	512	1, 2, 4	yes		10.725	25.658			0.482	0.511	0.628
gan	(224, 224, 16)	(28, 28, 2, 8)	1024	128	1, 2, 4	yes		4.682	25.674			0.474	0.491	0.611
vae	(224, 224, 16)	(28, 28, 2, 8)	1024	128	1, 2, 4	yes		7.005	26.035			0.493	0.506	0.597
gan	(224, 224, 16)	(28, 28, 2, 8)	1024	256	1, 2, 4	yes		4.682	25.423			0.474	0.489	0.528
vae	(224, 224, 16)	(28, 28, 2, 8)	1024	256	1, 2, 4	yes		7.005	26.297			0.493	0.503	0.527
gan	(224, 224, 16)	(28, 28, 2, 8)	1024	512	1, 2, 4	yes		4.682	24.358			0.474	0.486	0.274
vae	(224, 224, 16)	(28, 28, 2, 8)	1024	512	1, 2, 4	yes		7.005	27.333			0.493	0.494	0.273
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	128	1, 2, 4	no		5.294	37.424			**0.467**	0.481	0.871
gan	(224, 224, 16)	(28, 28, 2, 32)	102	128	1, 2, 4	yes		4.813	36.882			0.474	0.471	0.602
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	128	1, 2, 4	yes		4.45	26.779			0.472	0.487	0.604
gan	(224, 224, 16)	(28, 28, 2, 32)	10240	128	1, 2, 4	yes		4.626	29.332			0.478	0.5	0.603
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	256	1, 2, 4	no		5.294	27.413			**0.467**	0.481	0.699
gan	(224, 224, 16)	(28, 28, 2, 32)	102	256	1, 2, 4	yes		4.813	29.415			0.474	0.477	0.53
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	256	1, 2, 4	yes		4.45	25.764			0.472	0.497	0.527
gan	(224, 224, 16)	(28, 28, 2, 32)	10240	256	1, 2, 4	yes		4.626	29.209			0.478	0.509	0.528
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	512	1, 2, 4	no		5.294	28.611			**0.467**	0.484	0.302
gan	(224, 224, 16)	(28, 28, 2, 32)	102	512	1, 2, 4	yes		4.813	29.639			0.474	0.496	0.276
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	512	1, 2, 4	yes		4.45	27.348			0.472	0.48	0.274
gan	(224, 224, 16)	(28, 28, 2, 32)	10240	512	1, 2, 4	yes		4.626	29.677			0.478	0.5	0.28
gan	(224, 224, 16)	(28, 28, 2, 256)	1024	128	1, 2, 4	yes		5.155	162.377			0.473	0.401	0.604
gan	(224, 224, 16)	(28, 28, 2, 256)	1024	256	1, 2, 4	yes		5.155	31.041			0.473	0.49	0.517
gan	(224, 224, 16)	(28, 28, 2, 256)	1024	512	1, 2, 4	yes		5.155	26.478			0.473	0.482	0.267
gan	(224, 224, 16)	(56, 56, 4, 8)	1024	128	1, 2, 4	yes		1.841	39.636			0.472	0.5	0.205
gan	(224, 224, 16)	(56, 56, 4, 8)	1024	256	1, 2, 4	yes		1.841	20.197			0.472	0.496	0.122
gan	(224, 224, 16)	(56, 56, 4, 8)	1024	512	1, 2, 4	yes		1.841	23.187			0.472	0.502	0.035
gan	(224, 224, 16)	(56, 56, 4, 32)	1024	128	1, 2, 4	yes		**1.834**	21.966			0.477	0.495	0.206
gan	(224, 224, 16)	(56, 56, 4, 32)	1024	256	1, 2, 4	yes		**1.834**	20.432			0.477	0.499	0.121
gan	(224, 224, 16)	(56, 56, 4, 32)	1024	256	1, 2, 4, 8	yes		**1.834**	26.717			0.477	0.503	0.094
gan	(224, 224, 16)	(56, 56, 4, 32)	1024	512	1, 2, 4	yes		**1.834**	22.364			0.477	0.504	0.035
gan	(672, 672, 16)	(84, 84, 2, 8)	1024	512	1, 2, 4	yes		3.542	**5.788**			0.498	0.521	0.032
gan*	(672, 672, 16)	(84, 84, 2, 8)	1024	512	1, 2, 4	yes		2.06	6.086			0.475	**0.468**	0.032
gan	(672, 672, 16)	(84, 84, 4, 8)	1024	512	1, 2, 4	yes		5.478	6.02			0.486	0.492	0.019

Table B.2. This table extends the analysis of the previous table B.1 by presenting results of all the experiments conducted over the course of the internship on the spectral KL-divergence for veins, arteries, and the optic nerve head (ONH).

							Veins			Arteries			ONH		
Model	**Res.**	**Latent**	**Codes**	**Base Ch.**	**Ch. Mult.**	**K-means**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**	**Train**	**Recon**	**Synth**
gan	(224, 224, 16)	(14, 14, 1, 256)	1024	128	1, 2, 4	yes	0.475	1.661	25.346	0.345	1.374	24.245	0.301	1.819	17.636
gan	(224, 224, 16)	(14, 14, 1, 256)	1024	256	1, 2, 4	yes		1.661	2.208		1.374	1.993		1.819	2.499
gan	(224, 224, 16)	(14, 14, 1, 256)	1024	512	1, 2, 4	yes		1.661	2.199		1.374	2.005		1.819	2.566
gan	(224, 224, 16)	(28, 28, 2, 8)	1024	128	1, 2, 4	yes		0.814	1.243		0.537	1.306		0.9	1.758
vae	(224, 224, 16)	(28, 28, 2, 8)	1024	128	1, 2, 4	yes		2.89	4.588		2.781	5.048		3.207	4.505
gan	(224, 224, 16)	(28, 28, 2, 8)	1024	256	1, 2, 4	yes		0.814	1.253		0.537	1.304		0.9	1.659
vae	(224, 224, 16)	(28, 28, 2, 8)	1024	256	1, 2, 4	yes		2.89	4.394		2.781	4.882		3.207	4.338
gan	(224, 224, 16)	(28, 28, 2, 8)	1024	512	1, 2, 4	yes		0.814	1.234		0.537	1.379		0.9	1.694
vae	(224, 224, 16)	(28, 28, 2, 8)	1024	512	1, 2, 4	yes		2.89	4.404		2.781	4.763		3.207	4.162
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	128	1, 2, 4	no		0.709	1.491		0.487	1.569		0.525	0.86
gan	(224, 224, 16)	(28, 28, 2, 32)	102	128	1, 2, 4	yes		0.803	1.244		0.55	1.339		0.713	1.077
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	128	1, 2, 4	yes		0.884	1.296		0.634	1.311		0.972	1.141
gan	(224, 224, 16)	(28, 28, 2, 32)	10240	128	1, 2, 4	yes		0.836	1.259		0.518	1.292		0.656	0.856
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	256	1, 2, 4	no		0.709	1.259		0.487	1.33		0.525	0.755
gan	(224, 224, 16)	(28, 28, 2, 32)	102	256	1, 2, 4	yes		0.803	1.313		0.55	1.305		0.713	1.158
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	256	1, 2, 4	yes		0.884	1.453		0.634	1.55		0.972	1.377
gan	(224, 224, 16)	(28, 28, 2, 32)	10240	256	1, 2, 4	yes		0.836	1.436		0.518	1.445		0.656	0.951
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	512	1, 2, 4	no		0.709	1.275		0.487	1.367		0.525	0.841
gan	(224, 224, 16)	(28, 28, 2, 32)	102	512	1, 2, 4	yes		0.803	1.532		0.55	1.54		0.713	1.407
gan	(224, 224, 16)	(28, 28, 2, 32)	1024	512	1, 2, 4	yes		0.884	1.261		0.634	1.337		0.972	1.202
gan	(224, 224, 16)	(28, 28, 2, 32)	10240	512	1, 2, 4	yes		0.836	1.346		0.518	1.324		0.656	0.867
gan	(224, 224, 16)	(28, 28, 2, 256)	1024	128	1, 2, 4	yes		0.837	7.133		0.564	7.736		0.659	5.499
gan	(224, 224, 16)	(28, 28, 2, 256)	1024	256	1, 2, 4	yes		0.837	1.383		0.564	1.397		0.659	1.225
gan	(224, 224, 16)	(28, 28, 2, 256)	1024	512	1, 2, 4	yes		0.837	1.31		0.564	1.149		0.659	0.978
gan	(224, 224, 16)	(56, 56, 4, 8)	1024	128	1, 2, 4	yes		0.608	1.068		**0.404**	1.107		**0.345**	**0.545**
gan	(224, 224, 16)	(56, 56, 4, 8)	1024	256	1, 2, 4	yes		0.608	**0.768**		**0.404**	**0.823**		**0.345**	0.582
gan	(224, 224, 16)	(56, 56, 4, 8)	1024	512	1, 2, 4	yes		0.608	0.855		**0.404**	0.963		**0.345**	0.715
gan	(224, 224, 16)	(56, 56, 4, 32)	1024	128	1, 2, 4	yes		**0.585**	1.237		0.42	1.326		0.494	1.0
gan	(224, 224, 16)	(56, 56, 4, 32)	1024	256	1, 2, 4	yes		**0.585**	1.06		0.42	1.187		0.494	1.056
gan	(224, 224, 16)	(56, 56, 4, 32)	1024	256	1, 2, 4, 8	yes		**0.585**	0.976		0.42	1.103		0.494	1.045
gan	(224, 224, 16)	(56, 56, 4, 32)	1024	512	1, 2, 4	yes		**0.585**	1.086		0.42	1.166		0.494	1.06
gan	(672, 672, 16)	(84, 84, 2, 8)	1024	512	1, 2, 4	yes		2.905	3.031		2.657	2.996		3.219	3.486
gan*	(672, 672, 16)	(84, 84, 2, 8)	1024	512	1, 2, 4	yes		0.769	1.869		0.529	1.104		0.379	1.081
gan	(672, 672, 16)	(84, 84, 4, 8)	1024	512	1, 2, 4	yes		4.445	5.453		3.737	4.746		3.612	4.356