Geração de Regiões de Interesse Mamográficas utilizando um Modelo de Difusão Latente

Franklin Anthony Ramos Coêlho[1], **Thaís Gaudencio do Rêgo**[1]

[1]Centro de Informática – Universidade Federal da Paraíba (UFPB)
João Pessoa – PB – Brasil

franklinanthony@eng.ci.ufpb.br, gaudenciothais@gmail.com

Abstract. *Breast cancer is the most common and leading cause of cancer deaths among women worldwide. In 2020, approximately 2.3 million new cases were registered in women, representing 11.7% of all cancer cases. Early diagnosis plays a crucial role in reducing breast cancer mortality. This study aims to generate Regions of Interest (ROIs) using a Latent Diffusion Model (LDM). The LDM is a probabilistic model that allows for the representation of complex latent information in images, which can be useful in identifying subtle patterns and efficiently generating ROIs. This work contributes to the advancement of synthetic medical image generation, offering a promising approach for generating diverse mammography ROIs. The effectiveness of the proposed method is evaluated based on the Mean Absolute Error, Mean Squared Error, and Fréchet Distance on a dataset of mammograms, from which the target regions for this application are extracted.*

Resumo. *O câncer de mama é o mais comum e a principal causa de mortes por câncer entre mulheres em todo o mundo. Em 2020, aproximadamente 2,3 milhões de novos casos foram registrados em mulheres, representando 11,7% de todos os casos de câncer. O diagnóstico precoce desempenha um papel crucial na redução da mortalidade por câncer de mama. Este estudo visa gerar Regiões de Interesse (ROIs) usando um Modelo de Difusão Latente (LDM). O LDM é um modelo probabilístico que permite a representação de informações latentes complexas em imagens, o que pode ser útil na identificação de padrões sutis e na geração eficiente de ROIs. Este trabalho contribui para o avanço da geração sintética de imagens médicas, oferecendo uma abordagem promissora para a geração de ROIs de mamografias diversificadas. A eficácia do método proposto é avaliada a partir do Erro Médio Absoluto, do Erro Médio Quadrático e da Distância de Fréchet em um conjunto de dados de mamografias, das quais as regiões-alvo desta aplicação são extraídas.*

Catalogação na publicação
Seção de Catalogação e Classificação

C672g Coêlho, Franklin Anthony Ramos.
Geração de regiões de interesse mamográficas utilizando um modelo de difusão latente / Franklin Anthony Ramos Coêlho. - João Pessoa, 2024.
25 f. : il.

Orientação: Thaís Gaudencio do Rêgo.
TCC (Graduação) - UFPB/CI.

1. Autoencoder. 2. Mamografia. 3. Modelo de difusão latente. 4. Região de interesse. I. Rêgo, Thaís Gaudencio do. II. Título.

UFPB/CI CDU 004.92

Elaborado por Michelle de Kássia Fonseca Barbosa - CRB-738

1. Introdução

O câncer de mama é o mais comum e a principal causa de mortes provenientes de câncer entre as mulheres em todo o mundo [Bray et al. 2018]. No ano de 2020, por exemplo, ocorreram aproximadamente 2,3 milhões de novos casos em mulheres, representando 11,7% de todos os casos de câncer. Isso equivale a um risco estimado de 47,80 casos a cada 100 mil mulheres. As taxas mais altas de incidência foram observadas na América do Norte, Oceania e países da Europa Ocidental [Ferlay et al. 2021, Sung et al. 2021].

O diagnóstico precoce desempenha um papel fundamental na redução da mortalidade causada pelo câncer de mama. Dentre os métodos existentes, a mamografia é amplamente utilizada na detecção de lesões na mama. Entretanto, esta técnica pode não identificar tumores em estágios iniciais, ocasionando avaliações falsas negativas [Khalid et al. 2023].

A detecção automatizada de regiões de interesse (*Regions of Interest* - ROIs, do inglês), como tumores malignos ou lesões anormais, tem o potencial de melhorar a eficiência e a precisão do diagnóstico [Huynh et al. 2023]. Nesse âmbito, a aplicação de técnicas de processamento de imagens e aprendizado de máquina tem sido amplamente explorada para auxiliar os profissionais na análise de imagens médicas, como na detecção automatizada de ROIs existentes em mamografias [Xue et al. 2022]. Contudo, muitos modelos existentes ainda apresentam insuficiências em termos de precisão quando comparados à análise humana, ocasionadas pela limitação na quantidade e diversidade de dados [Houssami et al. 2019].

Este estudo tem como objetivo a geração de ROIs a partir de um Modelo de Difusão Latente (*Latent Diffusion Model* - LDM, do inglês). O LDM é um modelo probabilístico que permite a representação de informações latentes complexas nas imagens [Hwang e Woo 2023], o que pode ser útil na identificação de padrões sutis e na geração de ROIs de forma eficiente e precisa.

Este trabalho contribui para o avanço da geração sintética de imagens médicas, oferecendo uma abordagem promissora para a geração de ROIs de mamografias diversificadas. A eficácia do método proposto é avaliada a partir do Erro Médio Absoluto, do Erro Médio Quadrático e da Distância de Fréchet em um conjunto de dados de mamografias, das quais as regiões-alvo desta aplicação são extraídas.

2. Trabalhos relacionados

Para esta pesquisa, foram analisados 4 artigos que utilizam, em algum grau de semelhança, modelos equivalentes àqueles que são objeto deste estudo. O objetivo é analisar, de forma condensada, o processo de desenvolvimento, os algoritmos utilizados e os resultados obtidos pelos autores.

O trabalho de [Rombach et al. 2022] teve o propósito de desenvolver LDMs capazes de obter um desempenho de última geração em várias tarefas de síntese de imagem, incluindo pintura em imagem, geração incondicional, síntese de texto em imagem e super-resolução.

Como banco de dados, foram utilizados o CelebA-HQ, conjunto de imagens de alta resolução de faces de famosos; o DIV2K, um conjunto de imagens para fins de estudos envolvendo super-resolução; o FFHQ, que, assim como o CelebA-HQ, contém imagens em alta resolução de faces; o ImageNet [Deng et al. 2009], que dispõe de milhões de imagens subdivididas em milhares de classes; o LSUN-Beds, composto por imagens de camas e quartos; o LSUN-Churches, que contém imagens de igrejas e catedrais; e o MS-COCO, conhecido por sua diversidade de cenas e objetos.

No que se refere aos algoritmos, foram utilizados o ADM, ADM-G, BigGAN-deep, CogView, DALL-E, DC-VAE, DDPM, Modelos de Regressão, ImageBART, Lafite, LDM, LSGM, PGGAN, ProjectedGAN, SR3, StyleGAN, StyleGAN-2, U-Net GAN, UDM e VQGAN + T.

Do ponto de vista da síntese de imagens, por exemplo, o modelo com os melhores resultados foi o LDM-4-G – desenvolvido pelos autores –, o qual obteve, para a Distância de Fréchet (*Fréchet Inception Distance* - FID, do inglês), *Inception Score* (IS), Precisão e Revocação, respectivamente, 3,60, 247,67±5,59, 0,87 e 0,48.

O estudo de [Pinaya et al. 2022] explorou o uso de LDMs para gerar imagens sintéticas de varreduras cerebrais 3D de alta resolução, a partir de imagens de ressonância magnética (*Magnetic Resonance Imaging - MRIs, do inglês*. Os dados foram provenientes do UK Biobank, formado por informações sobre 500.000 participantes, incluindo algumas ressonâncias magnéticas do cérebro; foi utilizado um subconjunto de 31.740 MRIs cerebrais.

Em relação à qualidade das imagens geradas, o LDM, proposto pelos autores, obteve os melhores resultados em um cenário composto pelos algoritmos LSGAN e VAE-GAN. Do ponto de vista da FID, da Medida Multiescalar do Índice de Similaridade Estrutural (MS-SSIM) e 4-G-R-SSIM, o modelo obteve os seguintes resultados: 0,0076, 0,6555 e 0,3883, respectivamente.

Na aplicação de [Müller-Franzes et al. 2022], foi avaliado o desempenho do *Medfusion*, um Modelo Probabilístico de Difusão de Eliminação de Ruído (DDPM) latente condicional, em comparação com modelos baseados em Redes Adversárias Generativas (*Generative Adversarial Networks* - GANs, do inglês) na geração de imagens médicas com e sem condições específicas, como glaucoma e cardiomegalia.

Os bancos de imagens utilizados foram o AIROGS, composto por imagens médicas oftalmológicas; o CheXpert, que dispõe de 200.000 imagens de radiografias do tórax; e o CRCDX, um conjunto de imagens histológicas de câncer colorretal.

Em comparação com os algoritmos StyleGAN3, cGAN e ProGAN, o *Medfusion* demonstrou o melhor desempenho em termos de métricas. No contexto do AIROGS, obteve uma FID de 11,63, uma Precisão de 0,70 e uma Revocação de 0,40. Para o CheXpert, alcançou 17,28, 0,68 e 0,32, respectivamente. No cenário envolvendo o CRCDX, os autores conseguiram, respectivamente, 30,03, 0,66 e 0,41.

O estudo de [Sagers et al. 2023], por sua vez, avaliou a utilidade de modelos generativos, especificamente LDMs, na geração de imagens sintéticas de doenças de pele e propôs desenvolver um novo conjunto de imagens sintéticas, usando diferentes estratégias de geração. Como fonte de imagens, foram utilizados o Fitzpatrick 17k, que contém imagens de rostos classificados de acordo com o tom de pele; e o Stanford DDI, formado por imagens de alta resolução da pele com diferentes condições dermatológicas.

Os autores conseguiram, dentre os resultados esperados, gerar 458.920 imagens sintéticas, a partir dos dois bancos de imagens citados anteriormente. Em relação ao impacto do aumento de dados a partir de imagens sintéticas, observou-se uma melhoria na precisão dos classificadores de até 13,2%.

Fica claro, neste contexto, que a síntese de imagens é um campo com vastas aplicações possíveis. Ao analisar os estudos em questão, identificam-se modelos, bancos de dados e métricas utilizados tanto nos trabalhos citados quanto neste estudo, como detalhado no Quadro 1.

Este trabalho se difere dos mencionados anteriormente, dentre outros fatores, pela proposta de geração sintética de ROIs de mamas a partir de um Autoencoder acrescido da Divergência de Kullback-Leibler (*Kullback–Leibler Divergence* - KL, do inglês), o que pode aprimorar substancialmente a capacidade de generalização de modelos de geração de imagens médicas.

Quadro 1. Resumo sobre os trabalhos relacionados analisados.

Autor	Objetivo	Algoritmo	Dataset	Métrica
[Rombach et al. 2022]	Desenvolver modelos de difusão latente (LDMs) que alcancem desempenho de última geração em várias tarefas de síntese de imagem, incluindo pintura em imagem, geração incondicional, síntese de texto em imagem e super-resolução.	ADM, ADM-G, BigGAN-deep, CogView, DALL-E, DC-VAE, DDPM, Modelos de Regressão, ImageBART, Lafite, LDM, LSGM, PGGAN, ProjectedGAN, SR3, StyleGAN, StyleGAN-2, U-Net GAN, UDM e VQGAN + T	CelebA-HQ, DIV2K, FFHQ, ImageNet, LSUN-Beds, LSUN-Churches e MS-COCO	Precisão, Revocação, Distância de Fréchet (FID), *Inception Score* (IS), Relação Sinal-Ruído de Pico (PSNR), Medida do Índice de Similaridade Estrutural (SSIM) e Similaridade de Imagem Perceptual Aprendida (LPIPS)
[Pinaya et al. 2022]	Explorar o uso de modelos de difusão latente (LDMs) para gerar imagens sintéticas de varreduras cerebrais 3D de alta resolução a partir de imagens de ressonância magnética.	LDM, LSGAN e VAEGAN	UK Biobank	L1 Loss, Perceptual Loss, Distância de Fréchet (FID), Medida Multiescalar do Índice de Similaridade Estrutural (MS-SSIM) e 4-G-R-SSIM
[Müller-Franzes et al. 2022]	Avaliar o desempenho do *Medfusion*, um Modelo Probabilístico de Difusão de Eliminação de Ruído (DDPM) latente condicional, contra modelos baseados em GANs na geração de imagens médicas com e sem condições específicas, como glaucoma e cardiomegalia.	cGAN, DDIM, ProGAN, Stable Diffusion, StyleGAN3 e U-Net	AIROGS, CheXpert e CRCDX	Distância de Fréchet (FID), Erro Médio Quadrático (MSE), Medida Multiescalar do Índice de Similaridade Estrutural (MS-SSIM), Precisão e Revocação
[Sagers et al. 2023]	Avaliar a utilidade de modelos generativos, especificamente modelos de difusão latente, na geração de imagens sintéticas de doenças de pele, e desenvolver um novo conjunto de imagens sintéticas usando diferentes estratégias de geração.	EfficientNetV2	Fitzpatrick 17k e Stanford DDI	Acurácia média, p-valor e Procedimento de Benjamini-Hockberg
Este estudo	Gerar ROIs mamográficas a partir de um Modelo de Difusão Latente	AutoencoderKL e LDM	EMBED	MAE, MSE e FID

3. Metodologia

Nesta seção, é descrita a metodologia utilizada ao longo deste estudo, o que inclui a descrição da base de dados, as técnicas de pré-processamento realizadas, o ambiente de desenvolvimento utilizado, as métricas de avaliação e os modelos implementados.

3.1. Base de dados

As imagens utilizadas neste trabalho foram extraídas do *Emory Breast Imaging Dataset* (EMBED) [Jeong et al. 2023]. O EMBED é composto por mais de 3,3 milhões de mamografias de 116.902 mulheres, com dados extraídos entre 2013 e 2020. As imagens são agrupadas de acordo com o Sistema de Laudos e Dados para a Imagem da Mama (*Breast Imaging Reporting and Data System* - BI-RADS, do inglês), em 7 classes [Xing et al. 2021]: 0 - Exame inconclusivo; 1 - Normal; 2 - Achado benigno; 3 - Provavelmente benigno; 4 - Suspeito; 5 - Altamente suspeito; e 6 - Maligno. Alguns exemplos do EMBED podem ser vistos na Figura 1. Da primeira à última coluna, podem ser observadas instâncias normais, altamente suspeitas e inconclusivas.

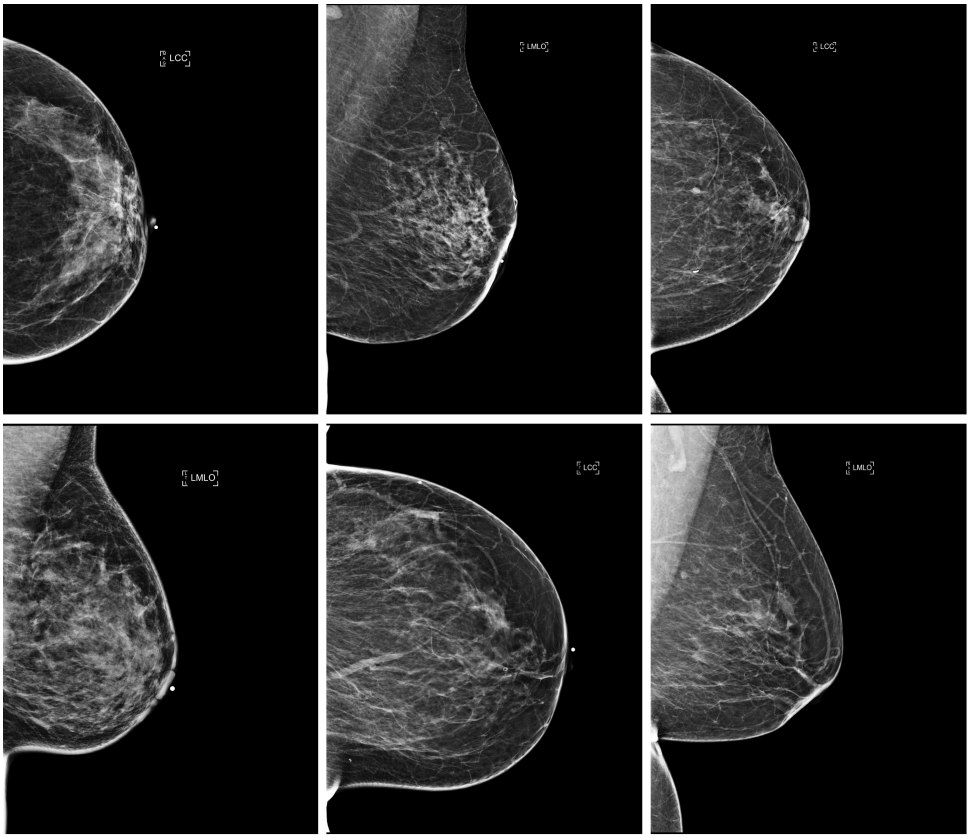

Figura 1. Exemplos de imagens que compõem o EMBED.

Em relação aos equipamentos utilizados na geração das imagens, o banco conta com três tipos: *Fujifilm*, *GE Health Care* e *Hologic*. Em relação à modalidade de reconstrução das imagens radiológicas, o EMBED contém instâncias formadas pela visualização em duas dimensões (2D), em três dimensões (3D) e através de *C-View*, técnica utilizada para visualização 2D de forma sintética a partir de projeções e reconstruções das imagens 3D [Greer 2014].

3.1.1. Pré-processamento

3.1.1.1. Filtragem das imagens

As instâncias utilizadas neste estudo corresponderam a 1.464 imagens pertencentes à seguinte categoria encontrada no EMBED: Classe 0, indicativo de diagnóstico inconclusivo; visualização 2D; e geradas através do Hologic. A escolha da Classe 0 se deu pela presença de estruturas diversas e complexas presentes nas imagens, fator importante para que a aplicação consiga generalizar bem; maiores detalhes serão mencionados ao longo deste trabalho. Também foram utilizadas 225 imagens pertencentes à Classe β, que representa a junção de subamostras das Classes 1, 4, 5 e 6, seguindo as demais condições impostas para a Classe 0.

3.1.1.2. Extração das ROIs

Uma vez concluída a filtragem das imagens, o processo de extração das ROIs foi conduzido da seguinte forma:

- A identificação das regiões foi realizada através do metadado `ROI_coords` fornecido pelo EMBED;
- Com base no conjunto de pontos $(Y_{\text{min}}, X_{\text{min}}, Y_{\text{max}}, X_{\text{max}})$ associado a cada imagem, os pontos iniciais da ROI, X_{min} e Y_{min}, e os finais, X_{max} e Y_{max}, foram projetados sobre a mamografia;
- A ROI foi, então, extraída e salva, como mostra a Figura 2.

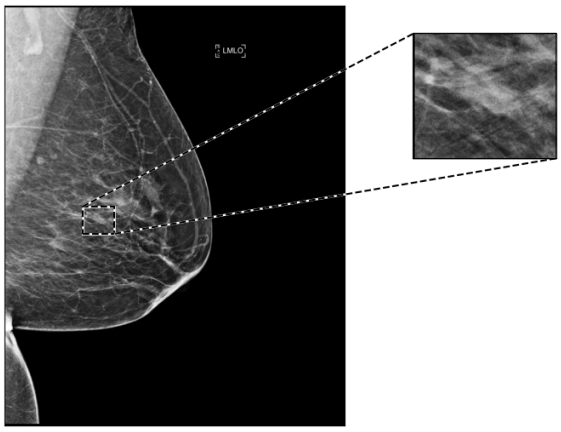

Figura 2. Ilustração do processo de extração da ROI (direita), a partir da mamografia (esquerda). Fonte: Autor.

3.1.1.3. Redimensionamento

Concluída a etapa de extração das ROIs, o próximo passo no pré-processamento foi o de redimensionamento. Neste sentido, foi escolhida a resolução de 512x512 pixels (65,77% do total de imagens estava dentro deste limite) para a padronização das imagens, garantindo assim que todas as ROIs tivessem o mesmo tamanho, o que facilitou a etapa de treinamento do modelo. Esse redimensionamento foi importante para garantir, também, a qualidade no processo de síntese das imagens.

3.1.1.4. Normalização

A normalização é uma técnica que ajusta os valores de determinado conjunto de dados para um intervalo específico, como $[0, 1]$ ou $[-1, 1]$ [Ali e Faraj 2014]. Neste estudo, a normalização foi realizada para o intervalo $[0, 1]$, a partir da captura do maior pixel no conjunto de treinamento, aplicando este valor para os conjuntos de teste e validação, mencionados no decorrer deste trabalho.

3.2. Ambiente de desenvolvimento

Para o desenvolvimento deste estudo, foi utilizada a linguagem de programação Python em sua versão 3.9, acompanhada das seguintes bibliotecas:

- `Matplotlib` - versão 3.8.3: Biblioteca *open-source* para plotagem de dados;
- `MONAI` - versão 1.2.0: Biblioteca *open-source* para aplicações envolvendo imagens médicas;
- `Numpy` - versão 1.26.4: Biblioteca *open-source* para processamento de dados multi-dimensionais e matrizes;
- `Pandas` - versão 2.2.0: Biblioteca *open-source* para manipulação e análise de dados
- `Scikit-learn` - versão 1.4.0: Biblioteca *open-source* para desenvolvimento de aplicações em *machine learning*;
- `Torch` - versão 2.2.0+cu121: Biblioteca *open-source* para desenvolvimento de modelos e processamento via GPU.

O hardware utilizado para o treinamento dos modelos e geração das amostras sintéticas foi o seguinte:

- Processador: Intel® Core™ i9-10900X CPU @ 3.70 GHz
- Memória RAM: 32 GB, 3200 MHz
- Placa de vídeo: NVIDIA RTX A4000, 16 GB de VRAM

3.3. Métricas de avaliação

Neste estudo, a avaliação dos resultados é de suma importância para comprovar a capacidade de generalização dos modelos e a semelhança entre as imagens originais e as geradas. As métricas utilizadas durante a execução deste trabalho estão descritas a seguir.

3.3.1. Erro Médio Absoluto

Também conhecido como *L1 Loss*, o Erro Médio Absoluto (*Mean Absolute Error* - MAE, do inglês), como mostra a Equação (1), é usado para calcular a diferença absoluta entre a imagem real ($y_{i_{\text{real}}}$) e a imagem gerada ($y_{i_{\text{gen}}}$), onde n é o número de observações [Hodson 2022].

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |y_{i_{\text{real}}} - y_{i_{\text{gen}}}|. \tag{1}$$

3.3.2. Erro Médio Quadrático

Como pode ser observado na Equação (2), o Erro Médio Quadrático (*Mean Square Error* - MSE, do inglês), chamado também de *L2 Loss*, semelhante ao MAE, mede a diferença quadrática entre a imagem real ($y_{i_{\text{real}}}$) e a imagem gerada ($y_{i_{\text{gen}}}$).

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(y_{i_{\text{real}}} - y_{i_{\text{gen}}})^2. \tag{2}$$

3.3.3. Perda do Discriminador

Presente nas GANs e também utilizada como métrica neste estudo, a Perda do Discriminador mede o quão bem o modelo de *Autoencoder*, neste caso, consegue identificar imagens reais e geradas [Pan et al. 2020].

$$\text{PD} = \frac{1}{m}\sum_{i=1}^{m}\left[\log D\left(x^{(i)}\right) + \log\left(1 - D\left(G\left(z^{(i)}\right)\right)\right)\right]. \tag{3}$$

A Perda do Discriminador é dada pela Equação (3), onde:

- $D\left(x^{(i)}\right)$ é a probabilidade do discriminador D classificar corretamente uma imagem real como real;
- $D\left(G\left(z^{(i)}\right)\right)$ é a probabilidade de D considerar que uma imagem criada pelo gerador G é real
- $\left(1 - D\left(G\left(z^{(i)}\right)\right)\right)$ é o que D pretende maximizar, visto que trata-se da correta identificação das imagens geradas, atribuindo probabilidades baixas para elas

3.3.4. Perda do Gerador

A Perda do Gerador, descrita pela Equação (4), estima, neste contexto, a qualidade com que o *Autoencoder* aprendeu a distribuição dos dados de entrada [Pan et al. 2020].

$$\text{PG} = \frac{1}{m}\sum_{i=1}^{m}\log\left(1 - D\left(G\left(z^{(i)}\right)\right)\right). \tag{4}$$

3.3.5. Distância de Fréchet

A FID é uma métrica que analisa o quão parecidas as imagens geradas são das originais, através da comparação entre a média e o desvio padrão das mesmas [Heusel et al. 2018], com o auxílio da ResNet50 [Mascarenhas e Agarwal 2021], uma rede treinada com a ImageNet [Deng et al. 2009], onde o cálculo é realizado na penúltima camada da rede.

$$\text{FID}(x, g) = ||\mu_x - \mu_g||^2 + \text{Tr}\left(\sum x + \sum g - 2\sqrt{\left(\sum x \sum g\right)}\right). \tag{5}$$

A FID é calculada de acordo com a Equação (5), na qual:

- x e g são os vetores de características das imagens reais e geradas, respectivamente;
- μ_x e μ_g são, respectivamente, as magnitudes dos vetores x e g;
- Tr é uma operação conhecida como traço, que calcula a soma dos elementos na diagonal principal de uma matriz quadrada;
- $\sum x$ e $\sum g$ são as matrizes de covariância dos vetores x e g, respectivamente.

3.4. Modelos

Acerca dos modelos presentes neste estudo, foram utilizados o *AutoencoderKL* e um LDM.

3.4.1. *Autoencoder*

O modelo de *Autoencoder* escolhido foi o *AutoencoderKL*, uma adaptação do *Autoencoder* Variacional (*Variational Autoencoder* - VAE, do inglês) acrescido da função de perda KL, que atua como um regularizador do espaço latente, ocasionando uma melhoria na capacidade de generalização do modelo [Asperti e Trentin 2020, Kingma e Welling 2022]. A Figura 3 ilustra a arquitetura do VAE.

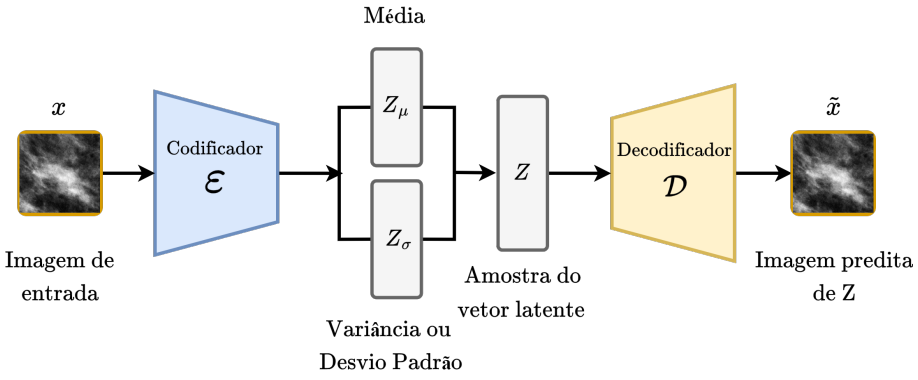

Figura 3. Arquitetura de um VAE. Fonte: Adaptado de [Rautela et al. 2024].

Em linhas gerais, a imagem x é mapeada, a partir do codificador $\mathcal{E}$, para uma distribuição sobre o espaço latente, da qual é gerada uma amostra $Z = \mathcal{E}(x)$. Posteriormente, a amostra passa pelo decodificador $\mathcal{D}$, o que resulta em uma distribuição de Z, $\tilde{x} = \mathcal{D}(Z) = \mathcal{D}(\mathcal{E}(x))$, cujos maiores valores prováveis devem corresponder à x [Cinelli et al. 2021].

3.4.2. Modelo de Difusão Latente

Um LDM é um modelo probabilístico feito para aprender uma distribuição de dados, eliminando gradualmente o ruído gerado durante este processo [Rombach et al. 2022], como demonstrado na Figura 4.

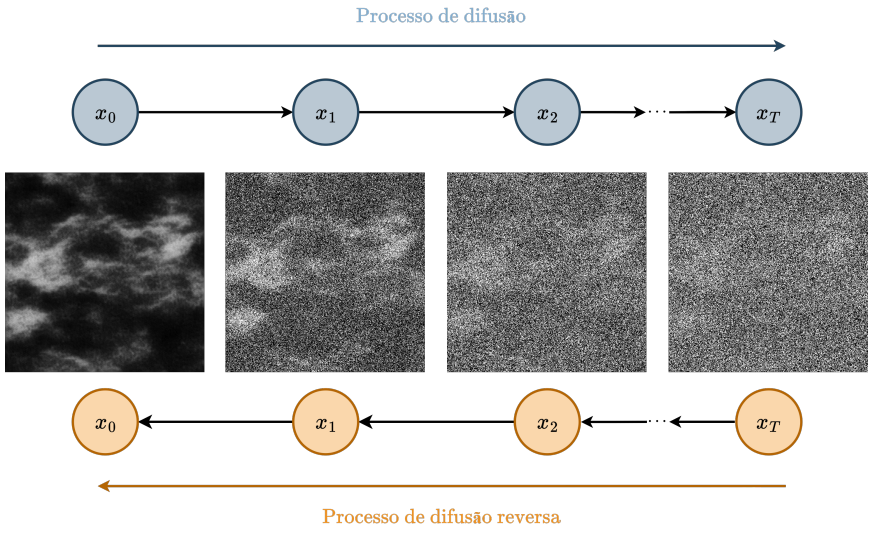

Figura 4. Processos de difusão e difusão reversa. Fonte: Adaptado de [Wang et al. 2023].

Durante o processo de difusão, os dados de entrada, como imagens, são gradualmente perturbados a cada passo, sendo T o número máximo de iterações, até se transformarem totalmente em ruído, como em x_T. Um processo de difusão reversa, por sua vez, é aprendido para reduzir iterativamente esse ruído e gerar novos dados semelhantes aos de entrada.

Para a construção do modelo, foi projetada uma *pipeline* de difusão, ilustrada na Figura 5. O conjunto de dados x passa pelo codificador $\mathcal{E}$ do *AutoencoderKL*, que transforma as imagens para o espaço latente. O ruído gaussiano é, então, adicionado. Em seguida, o resultado é passado para o treinamento do estimador de ruído, papel desempenhado pela rede U-Net, responsável pelo processo de redução de ruído, o que gera, ao final, um dado semelhante ao original [Rombach et al. 2022].

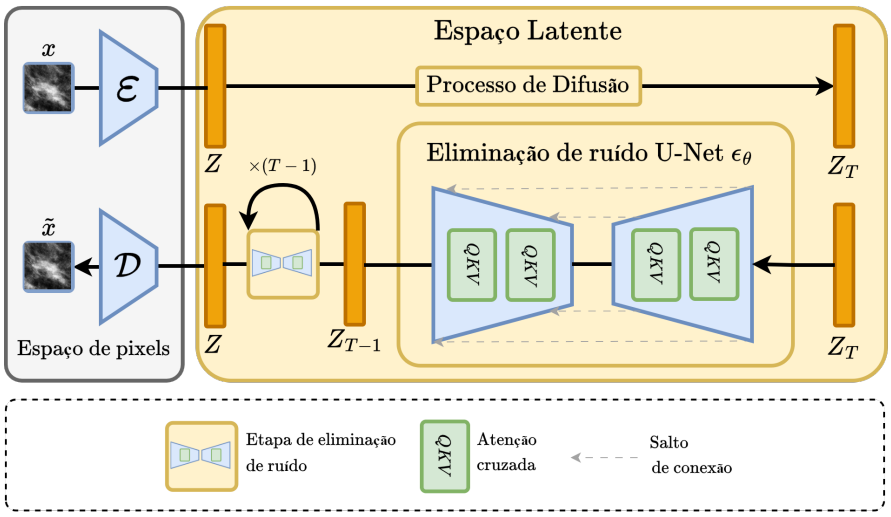

Figura 5. *Pipeline* do modelo de difusão latente. Fonte: Adaptado de [Rombach et al. 2022].

3.5. Etapa de Treinamento

A seguir, a etapa de treinamento é descrita para os Experimentos 1 e 2.

3.5.1. Experimento 1: *Fine-tuning* entre as Classes 0 e β

No Experimento 1, as bases de dados compostas pela Classe 0 e pela Classe β foram divididas em subconjuntos de treinamento, teste e validação para o *AutoencoderKL*. Isso foi feito usando o método `train_test_split` da biblioteca `scikit-learn` da seguinte forma: 80% das amostras foram destinadas ao treinamento, o que correspondeu a 1.171 imagens para a Classe 0 e 180 para a Classe β; 10% foram reservadas para o teste, totalizando 147 para a Classe 0 e 23 para a Classe β; e os 10% restantes foram utilizados para a validação, resultando em 146 para a Classe 0 e 22 para a Classe β.

O *AutoencoderKL* foi instanciado através da classe `AutoencoderKL` do `MONAI`[1] e contou com os seguintes parâmetros:

- `spatial_dims = 2`: Refere-se à dimensão 2D das imagens de entrada.
- `in_channels = 1`: Um único canal de entrada, visto que as imagens são em escala de cinza.
- `out_channels = 1`: Indica, também, apenas um único canal de saída.
- `num_channels = (128, 128, 256)`: Número de canais em cada camada do *Autoencoder*.
- `latent_channels = 3`: Número de canais na representação latente.
- `num_res_blocks = 2`: Número de blocos residuais em cada camada convolucional.
- `attention_levels = (False, False, False)`: Parâmetro desabilitado para maior eficiência computacional.
- `with_encoder_nonlocal_attn = False`: Parâmetro desabilitado para maior eficiência computacional.
- `with_decoder_nonlocal_attn = False`: Parâmetro desabilitado para maior eficiência computacional.

Foi escolhida a Perda Perceptual, com um peso de 0,001, fundamentada na arquitetura *AlexNet*, com o objetivo de calcular a diferença entre duas imagens, com base em características perceptuais. Também foi utilizada a Perda Adversária por *Patch*, com peso 0,01. Esta, por sua vez, foi responsável por calcular a diferença entre partes das imagens.

O otimizador *Adam* foi utilizado para o *AutoencoderKL*, com taxa de aprendizado ajustada para 0,0001. Foi aplicado o `GradScaler` no otimizador em questão.

Para um processamento mais eficiente das imagens, foi utilizado o `DataLoader` do `MONAI`, extensão do `DataLoader` do `PyTorch`, que permite a divisão das instâncias em lotes.

O treinamento começou com a base de treinamento e validação da Classe 0, sendo condicionadas a 10 épocas e 4 imagens por lote. Foram utilizadas as funções de perda

[1] Documentações e aplicações disponíveis em: monai.io

MAE, Perdas do Gerador e do Discriminador. Ao término deste processo, um *checkpoint* do modelo foi salvo. Em seguida, uma nova instância do `AutoencoderKL` foi criada com base no *checkpoint* gerado anteriormente, e um novo treinamento foi realizado exclusivamente com as instâncias de treinamento e validação da Classe β, submetidas a 20 épocas e 2 imagens por lote.

De modo geral, a abordagem proposta visou inicialmente treinar o modelo com instâncias da Classe 0, uma vez que essas amostras são mais genéricas e representam estruturas mais complexas das imagens. Em seguida, foi realizado um ajuste fino (*fine-tuning*, do inglês), uma abordagem na qual os pesos de um modelo pré-treinado (neste contexto, com a Classe 0) são treinados em novos dados (Classe β, neste cenário, que possui ROIs mais específicas e homogêneas) [Singla et al. 2023]. Este procedimento teve como objetivo aprimorar a capacidade do modelo de lidar com poucas instâncias e gerar resultados representativos para estas regiões.

Concluída a etapa de treinamento do *AutoencoderKL*, a *pipeline* seguiu para o treinamento do LDM. O modelo foi instanciado através da classe `DiffusionModelUNet` do `MONAI` e contou com os seguintes parâmetros:

- `spatial_dims = 2`: Refere-se à dimensão 2D das imagens de entrada.
- `in_channels = 3`: Número de camadas latentes advindas do *AutoencoderKL*.
- `out_channels = 3`: Indica o número de camadas latentes geradas pelo LDM.
- `num_res_blocks = 2`: Indica a quantidade de `ResNetBlocks`.
- `num_channels = (128, 256, 512)`: Define a quantidade de blocos do tipo `DownBlock` e `UpBlock`.
- `attention_levels = (False, True, True)`: Define a quantidade e se haverá `AttentionBlocks`.
- `num_head_channels = (0, 256, 512)`: Número de canais em cada *attention head*.

O otimizador *Adam* foi utilizado com uma taxa de aprendizado ajustada para 0,0004. Foi aplicado o `GradScaler` para escalonamento durante o treinamento. Para um processamento mais eficiente das imagens, foi utilizado, assim como no *AutoencoderKL*, o `DataLoader` do `MONAI`.

Com base na divisão das imagens mencionada na etapa de treinamento do *AutoencoderKL*, o treinamento do LDM se deu com as bases de treinamento e validação da Classe 0 unificadas (1.317 instâncias no total), sendo condicionadas a 10 épocas e 4 imagens por lote. Foi utilizada a função de perda MSE. Ao término deste processo, um *checkpoint* do modelo foi salvo, como ocorreu com o *AutoencoderKL*. Em seguida, a partir da proposta de *fine-tuning*, uma nova instância do `DiffusionModelUNet` foi criada com base no *checkpoint* gerado anteriormente, e um novo treinamento foi realizado exclusivamente com a unificação das instâncias de treinamento e validação da Classe β (202 imagens no total), sujeitadas a 40 épocas e 2 imagens por lote.

3.5.2. Experimento 2: Abordagem com a Classe β

O Experimento 2, por sua vez, contou apenas com instâncias da Classe β. Em relação ao *AutoencoderKL*, 80% das imagens foram destinadas ao treinamento, o que correspondeu a 180 instâncias; 10% foram para o teste, totalizando 23 imagens; e os 10% restantes foram utilizados para a validação, resultando em 22 amostras. O treinamento foi submetido a 20 épocas e 4 imagens por lote. O processo de normalização, a instanciação do modelo e as funções de perda seguiram os mesmos procedimentos adotados no Experimento 1.

Em relação ao LDM, 202 imagens (junção das bases de imagens de treino e validação) foram utilizadas no treinamento. O treinamento foi submetido a 20 épocas e 4 imagens por lote. O processo de normalização, a instanciação do modelo e as funções de perda seguiram os mesmos procedimentos adotados no Experimento 1.

3.6. Etapa de Avaliação

Em paralelo à etapa de treinamento, ocorreu a fase de avaliação, conforme descrita nesta seção.

3.6.1. Experimento 1: *Fine-tuning* entre as Classes 0 e β

No contexto do *AutoencoderKL*, o Experimento 1 contou com etapas de validação a cada 2 épocas para a Classe 0 e a cada 5 épocas para a Classe β. A métrica utilizada foi a MAE. No contexto do LDM, o Experimento 1 contou com etapas de validação a cada 2 épocas para a Classe 0 e a cada 5 épocas para a Classe β. A métrica considerada foi a MSE. Houve uma diferença nos intervalos de validação, uma vez que, para a Classe 0, o treinamento contou com poucas épocas, devido à quantidade elevada de instâncias e à rápida convergência do modelo; para a Classe β, porém, o treinamento ocorreu por por mais épocas, devido à quantidade menor de imagens.

Ao término da execução da pipeline, imagens foram geradas com números de iterações distintos: 1000, 2000 e 3000. Esta abordagem permitiu analisar o impacto do número de iterações no desempenho do modelo e na qualidade das imagens sintetizadas, o que serviu de base para o cálculo da FID, medindo a semelhança entre as instâncias geradas e as que formaram os bancos de teste do experimento.

Vale ressaltar que foram geradas imagens tanto para a Classe 0 (antes do processo de *fine-tuning*), quanto para a Classe β (após o processo de *fine-tuning*), comparando, assim, as métricas de treinamento e as de validação, inclusive a FID.

3.6.2. Experimento 2: Abordagem com a Classe β

O Experimento 2 contou com etapas de validação a cada 4 épocas, para o *AutoencoderKL*. A métrica considerada foi a MAE. Em relação ao LDM, a validação ocorreu a cada 5 épocas. A métrica utilizada foi a MSE. Assim como foi descrito na Seção 3.6.1, também houve uma diferença nos intervalos de validação, justificada pelas mesmas condições elucidadas. Também houve a geração de imagens com números de iterações distintos.

4. Resultados e discussões

Nesta seção, são apresentados os resultados e as discussões dos Experimentos 1 e 2.

4.1. Experimento 1: *Fine-tuning* entre as Classes 0 e β

4.1.1. *AutoencoderKL*

Ao término do treinamento do *AutoencoderKL* com as instâncias da Classe 0, as métricas foram geradas e podem ser observadas na Figura 6. Uma vez instanciado um novo modelo, a partir do *checkpoint* gerado com as instâncias da Classe 0, o treinamento com a Classe β foi realizado e os resultados estão dispostos na Figura 7. É importante mencionar que as curvas de validação não começaram na época 0 devido ao fato de terem ocorrido a cada 2 épocas para a Classe 0 e a cada 5 épocas para a Classe β. Já as Perdas do Gerador e do Discriminador começaram a ser computadas a partir da 4ª época.

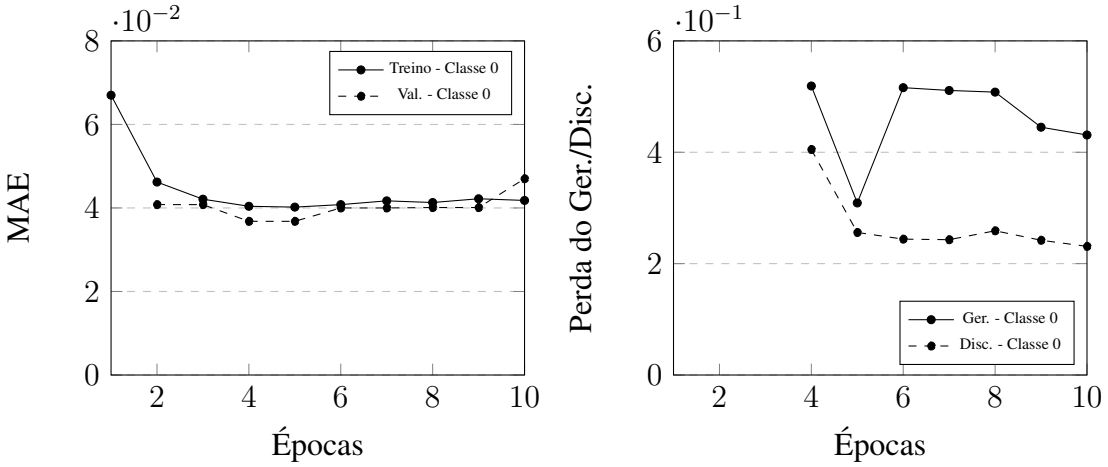

Figura 6. MAE (esquerda) e Perdas do Gerador e do Discriminador (direita) da Classe 0, durante o treinamento do *AutoencoderKL*.

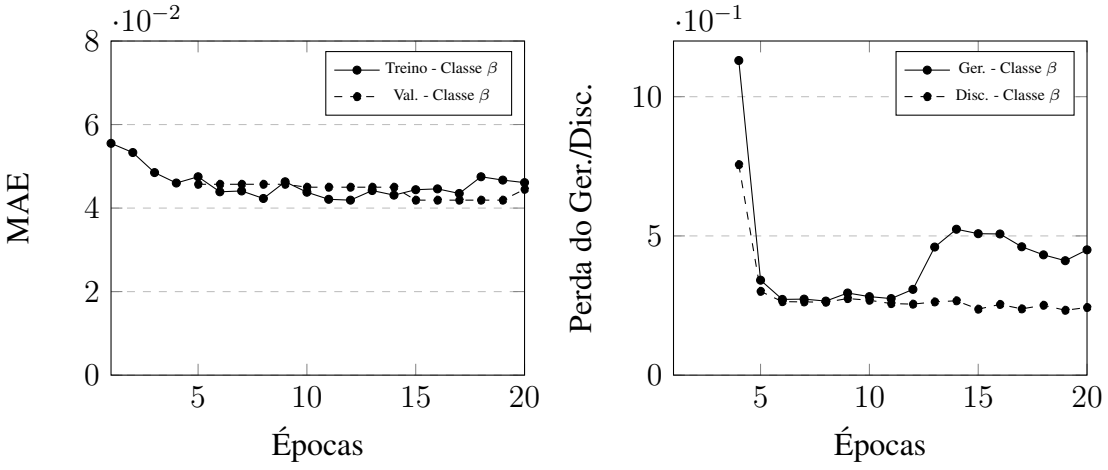

Figura 7. MAE (esquerda) e Perdas do Gerador e do Discriminador (direita) da Classe β, durante o treinamento do *AutoencoderKL*.

Nota-se, com o uso do *fine-tuning*, que os resultados das métricas da Classe β foram equivalentes aos obtidos na Classe 0, o que pode indicar que o modelo conseguiu generalizar bem para a nova classe. Durante o treinamento do *AutoencoderKL* com as Classes 0 e β, algumas imagens foram geradas durante a etapa de validação, como podem ser vistas na Figura 8.

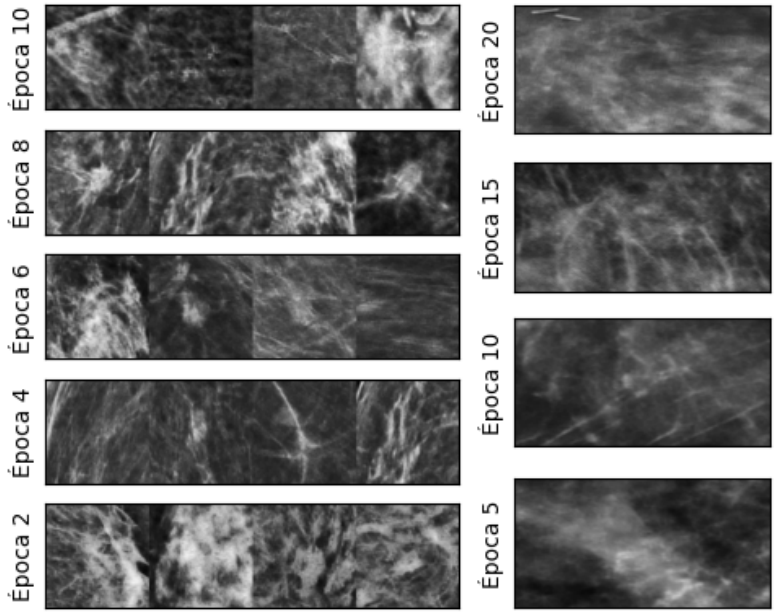

Figura 8. Amostras reconstruídas das Classes 0 (esquerda) e β (direita), ao final do treinamento do *AutoencoderKL*.

A reconstrução das imagens da Classe 0 pelo *AutoencoderKL* foi satisfatória, do ponto de vista da fidelidade de reconstrução. É possível observar a presença de estruturas complexas presentes nas ROIs originais, tais como regiões mais e menos densas. Em relação à Classe β, o processo de reconstrução não gerou imagens tão nítidas, apesar da existência de estruturas características, também presentes na Classe 0.

4.1.2. Modelo de Difusão Latente

As métricas geradas ao final da execução do LDM podem ser visualizadas na Tabela 1. Quatro amostras foram selecionadas para cada número de iterações, conforme a Figura 9.

Tabela 1. Comparativo entre a média do MSE e a FID para as Classes 0 e β com diferentes números de iterações.

Classes	Iterações	$\overline{\text{MSE}}\downarrow$	FID $\downarrow$
Classe 0	1000	0,2236	169,7993
	2000	0,1516	178,0812
	3000	0,1162	186,0836
Classe β	1000	0,2179	310,4529
	2000	0,1573	306,1325
	3000	0,1049	291,5312

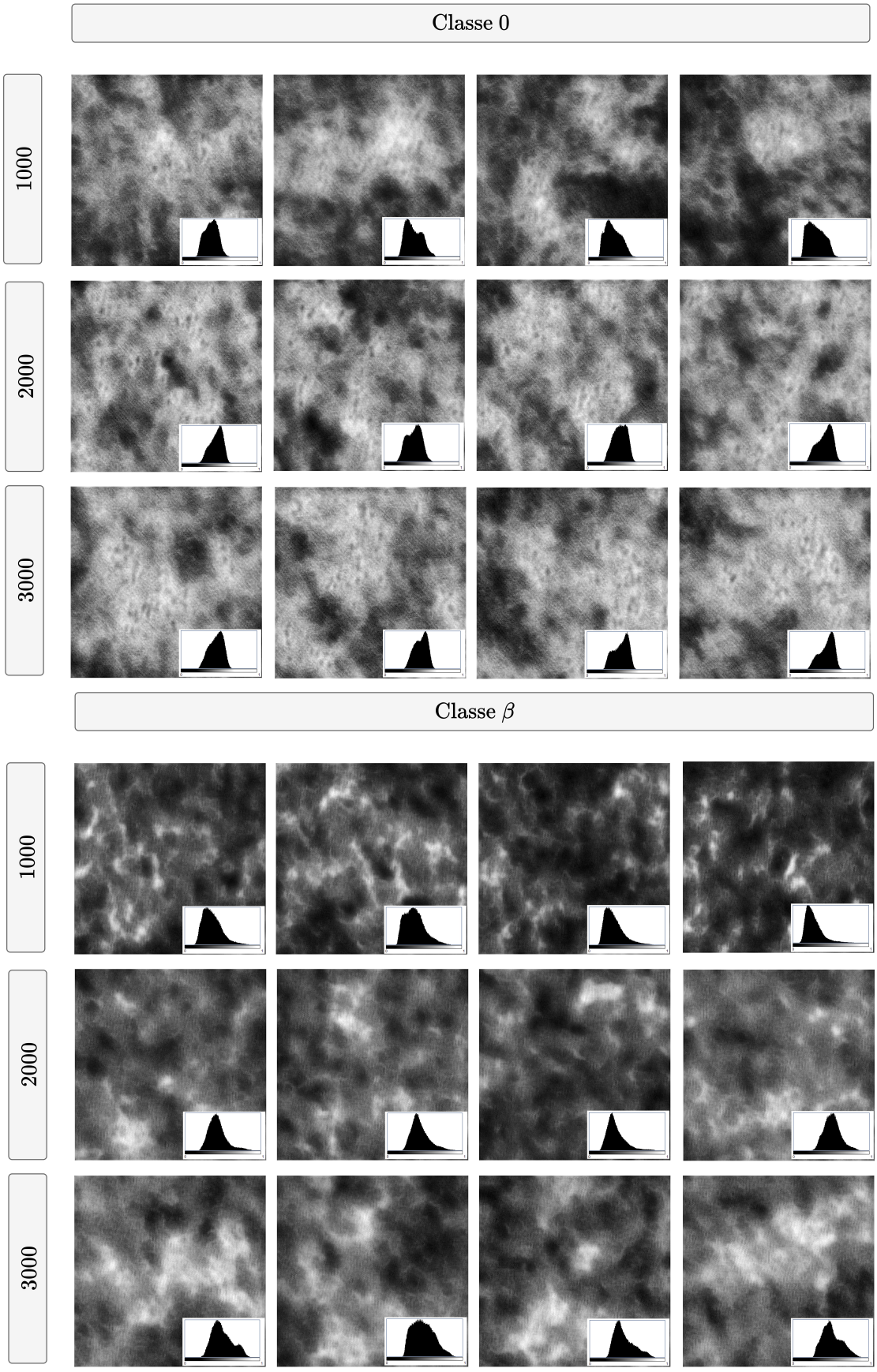

Figura 9. Amostras de ROIs geradas das Classes 0 (acima) e β (abaixo) com 1000, 2000 e 3000 iterações.

Os histogramas presentes nas imagens representam a distribuição da densidade dos pixels: regiões mais densas, como calcificações, são próximas ao branco, enquanto que regiões menos densas, como o tecido mamário, são localizadas próximas ao preto.

Sobre o MSE, foi possível notar um declínio ao passo em que o número de iterações aumentou. Para a Classe 0, apesar disso, a FID não seguiu a mesma tendência; para a Classe β, entretanto, houve uma pequena diminuição.

Apesar da heterogeneidade presente na Classe 0, a quantidade de instâncias pode ter levado o modelo a conseguir capturar estruturas mais complexas presentes nas mesmas, como mostra a FID na Tabela 1. Entretanto, embora mais homogênea, a Classe β contou com pouco mais de 15,3% do total de imagens da Classe 0. Mesmo com o uso do *fine-tuning*, a FID, neste cenário, foi superior.

Quanto à composição das imagens geradas (exemplos vistos na Figura 9), notou-se a presença de um artefato de textura do tipo "tabuleiro de xadrez". A presença do artefato "tabuleiro de xadrez"é comum em Redes Neurais Convolucionais (*Convolutional Neural Networks* - CNNs, do inglês) [Sugawara et al. 2019], como a U-Net, utilizada neste estudo. É válido mencionar, também, a existência de artefatos nas instâncias geradas da Classe 0, equivalentes a pontos escuros. Entretanto, como pode ser observado na Figura 10, é possível notar regiões semelhantes nas imagens utilizadas no treinamento do modelo, o que pode justificar seu surgimento.

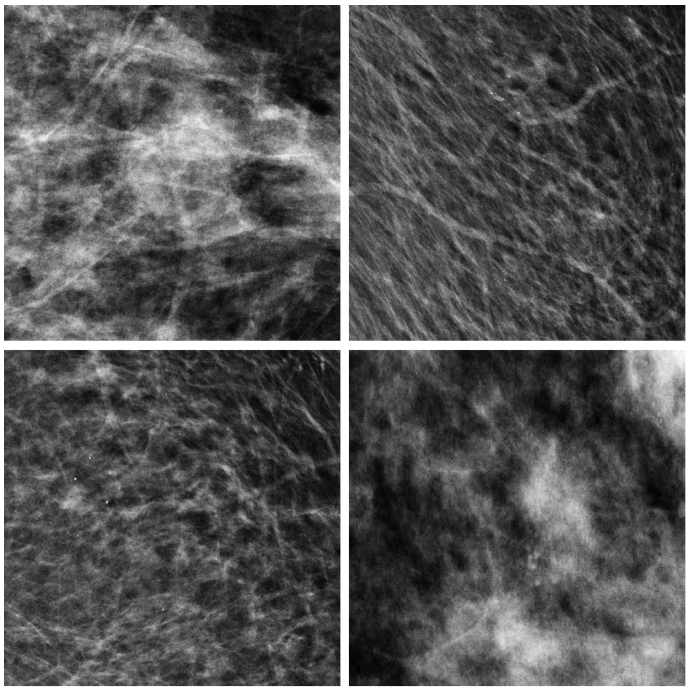

Figura 10. Amostras de ROIs de treinamento da Classe 0, nas quais é possível observar artefatos também presentes nas imagens geradas.

Por fim, é importante analisar o impacto do número de iterações do modelo na geração das imagens. Foi possível constatar, como demonstram os histogramas na Figura 9, que o aumento das iterações implicou na conversão da distribuição gaussiana em uma distribuição bimodal, comportamento esperado, evidenciado nos resultados e que representa regiões mais e menos densas das ROIs.

4.2. Experimento 2: Abordagem com a Classe β

4.2.1. *AutoencoderKL*

A Figura 11 demonstra os resultados do *AutoencoderKL*, exclusivamente com as imagens da Classe β. Assim como observado no Experimento 1, o MAE não sofreu muitas variações, equiparando-se aos valores obtidos anteriormente; a mesma situação pode ser observada para as Perdas do Gerador e do Discriminador, onde ambas permaneceram equivalentes ao longo do treinamento. Algumas imagens foram geradas durante a validação do *AutoencoderKL*, como observado na Figura 12.

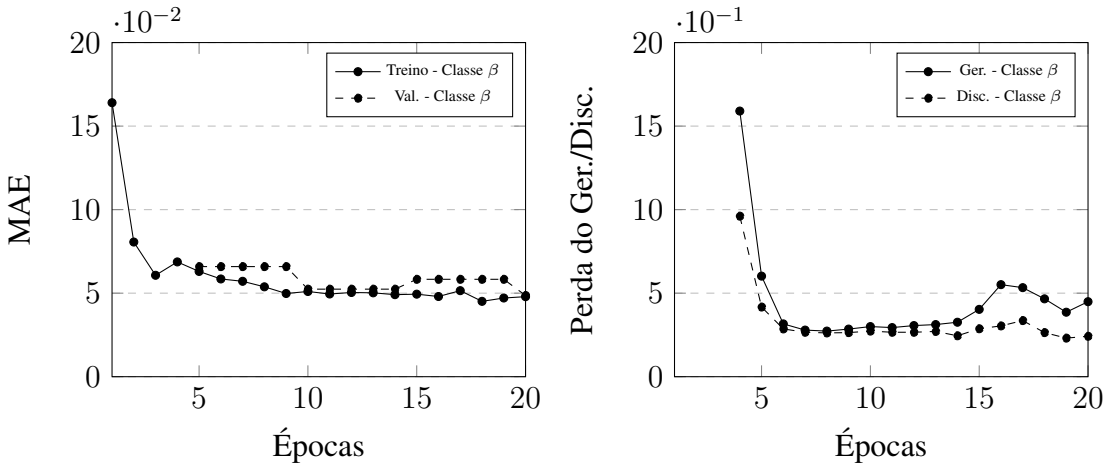

Figura 11. MAE (esquerda) e Perdas do Gerador e do Discriminador (direita) da Classe β, durante o treinamento do *AutoencoderKL*.

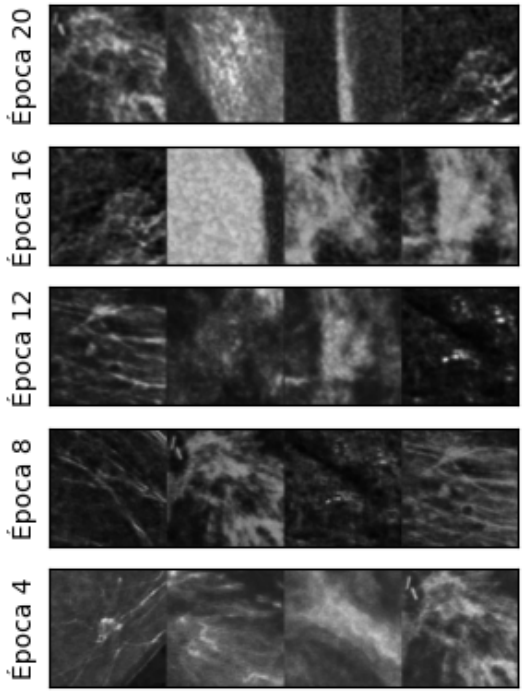

Figura 12. Amostras reconstruídas da Classe β, ao final do treinamento do *AutoencoderKL*.

As observações sobre as curvas de validação e das Perdas do Gerador e do Discriminador seguem os mesmos pontos mencionados na Seção 4.1.1.

A reconstrução das imagens da Classe β pelo *AutoencoderKL*, assim como no Experimento 1, foi satisfatória, apesar da existência de regiões como a segunda imagem da época 16, onde o aspecto resultante não condiz com o esperado. É possível notar, além disso, ROIs muito escuras.

4.2.2. Modelo de Difusão Latente

A média do MSE e a FID do Experimento 2 para cada número de iterações podem ser observadas na Tabela 2. O comportamento do MSE foi semelhante ao obtido anteriormente. Entretanto, a FID aumentou significativamente neste caso. Com isso, fica claro que a aplicação do *fine-tuning* produziu resultados significativamente melhores para a Classe β, como demonstrado na Tabela 1. Assim como no Experimento 1, 4 amostras foram selecionadas para cada número de iterações, como apresentadas na Figura 13.

Tabela 2. Comparativo entre a média do MSE e a FID para a Classe β com diferentes números de iterações.

Iterações	$\overline{\text{MSE}} \downarrow$	FID $\downarrow$
1000	0,2096	358,8904
2000	0,1739	350,2567
3000	0,1181	399,4162

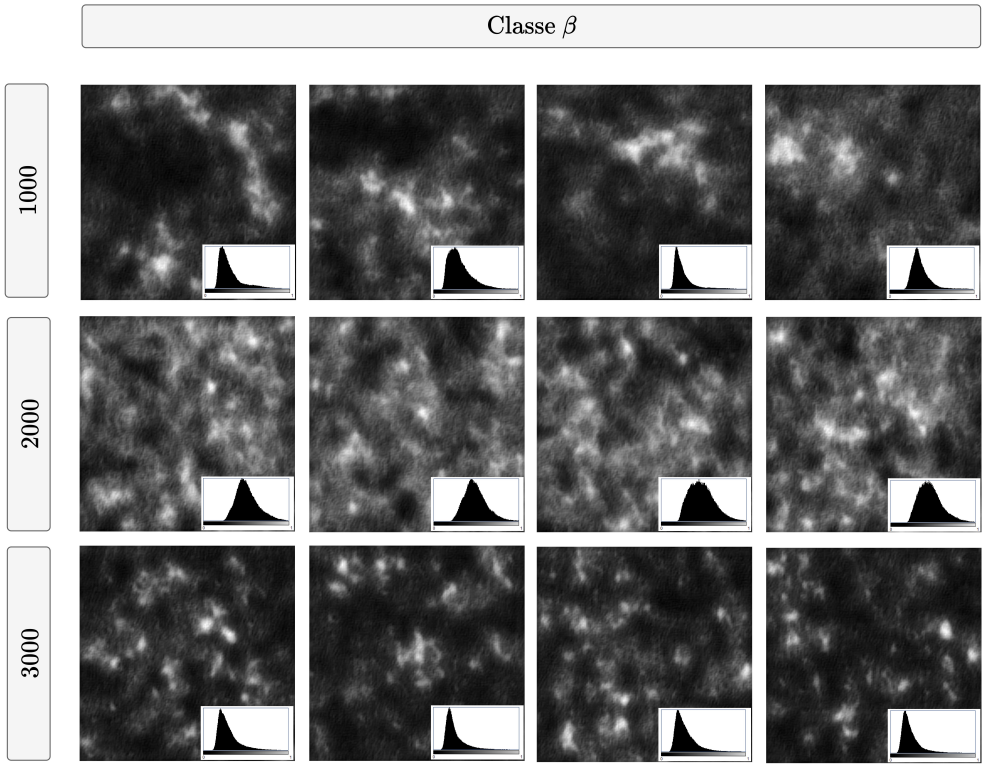

Figura 13. Ruídos gerados da Classe β com 1000, 2000 e 3000 iterações.

Quanto ao contexto dos artefatos, houve apenas a incidência do ”tabuleiro de xadrez”, também presente no Experimento 1. Em suma, percebeu-se a permanência da distribuição gaussiana em todos os cenários, o que implicou em resultados não representativos para as ROIs da Classe β.

5. Conclusão

Este trabalho propôs a criação de um modelo generativo para síntese de ROIs a partir de mamografias. O objetivo geral foi a geração de ROIs representativas a partir de 2 Classes distintas: instâncias com diagnóstico inconclusivo, representadas pela Classe 0; e a Classe β, composta por imagens com ROIs normais, suspeitas, altamente suspeitas e malignas.

Este objetivo foi alcançado em parte, com resultados satisfatórios apenas com a aplicação de *fine-tuning* entre as Classes 0 e β. Neste cenário, foram obtidas as seguintes métricas, com base na menor FID: para a Classe 0, com 1000 iterações, FID = 169,7993 e $\overline{\text{MSE}}$ = 0,2236; para a Classe β, com 3000 iterações, FID = 291,5312 e $\overline{\text{MSE}}$ = 0,1049. Apesar das FIDs elevadas, constatou-se um resultado satisfatório, a partir da análise dos histogramas das imagens geradas. Do ponto de vista da aplicação envolvendo apenas a Classe β, foi obtido o seguinte resultado, com base na menor FID: com 2000 iterações, FID = 358,8904 e $\overline{\text{MSE}}$ = 0,1739. Estes valores não foram relevantes, visto que as imagens geradas não representaram os dados de entrada, a partir, também, da distribuição presente nas instâncias.

Ademais, este trabalho também contribui para a comunidade com uma nova *pipeline* de treinamento e avaliação para a tarefa de geração sintética de ROIs mamográficas, permitindo o teste de diferentes tamanhos para a representação latente, além do ajuste de outros parâmetros nos modelos.

5.1. Trabalhos futuros

Apesar do resultado satisfatório – em parte – do modelo generativo, é possível aprofundar o estudo e melhorar a eficácia da *pipeline* para este conjunto de dados, incluindo, por exemplo:

- A troca de funções de perda no treinamento do *AutoencoderKL*
- Uma abordagem com outros modelos de *Autoencoder*, como o VQGAN e o VQ-VAE
- A síntese das imagens com um maior número de iterações

Referências

[Ali e Faraj 2014] Ali, P. J. M. e Faraj, R. H. (2014). Data normalization and standardization: A technical report. *Machine Learning Technical Reports*, 1(1):1–6.

[Asperti e Trentin 2020] Asperti, A. e Trentin, M. (2020). Balancing reconstruction error and kullback-leibler divergence in variational autoencoders. *IEEE Access*, 8:199440–199448.

[Bray et al. 2018] Bray, F., Ferlay, J., Soerjomataram, I., Siegel, R. L., Torre, L. A., e Jemal, A. (2018). Global cancer statistics 2018: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 68(6):394–424.

[Cinelli et al. 2021] Cinelli, L. P., Marins, M. A., Silva, E. A. B. d., e Netto, S. L. (2021). *Variational Methods for Machine Learning with Applications to Deep Networks*. Springer, 1ª edição.

[Deng et al. 2009] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., e Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255.

[Ferlay et al. 2021] Ferlay, J., Colombet, M., Soerjomataram, I., Parkin, D., Piñeros, M., Znaor, A., e Bray, F. (2021). Cancer statistics for the year 2020: An overview. *International Journal of Cancer*, 149(4):778–789.

[Greer 2014] Greer, L. R. N. (2014). The benefits of using synthesized 2d (c-view™) images in breast tomosynthesis exams. *Applied Radiology*, 43(11).

[Heusel et al. 2018] Heusel, M., Ramsauer, H., Unterthiner, T., e Nessler, B. (2018). Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv.1706.08500*.

[Hodson 2022] Hodson, T. O. (2022). Root-mean-square error (rmse) or mean absolute error (mae): when to use them or not. *Geoscientific Model Development*, 15(14):5481–5487.

[Houssami et al. 2019] Houssami, N., Kirkpatrick-Jones, G., Noguchi, N., e I. Lee, C. (2019). Artificial intelligence (ai) for the early detection of breast cancer: a scoping review to assess ai's potential in breast screening practice. *Expert Review of Medical Devices*, 16(5):351–362.

[Huynh et al. 2023] Huynh, H. N., Tran, A. T., e Tran, T. N. (2023). Region-of-interest optimization for deep-learning-based breast cancer detection in mammograms. *Applied Sciences*, 13(12).

[Hwang e Woo 2023] Hwang, I. e Woo, M. (2023). Image compression and decompression framework based on latent diffusion model for breast mammography. *arXiv:2310.05299*.

[Jeong et al. 2023] Jeong, J. J., Vey, B. L., Bhimireddy, A., Kim, T., Santos, T., Correa, R., Dutt, R., Mosunjac, M., Oprea-Ilies, G., Smith, G., Woo, M., McAdams, C. R., Newell, M. S., Banerjee, I., Gichoya, J., e Trivedi, H. (2023). The emory breast imaging dataset (embed): A racially diverse, granular dataset of 3.4 million screening and diagnostic mammographic images. *Radiology: Artificial Intelligence*, 5(1).

[Khalid et al. 2023] Khalid, A., Mehmood, A., Alabrah, A., Alkhamees, B. F., Amin, F., AlSalman, H., e Choi, G. S. (2023). Breast cancer detection and prevention using machine learning. *Diagnostics*, 13(19):3113.

[Kingma e Welling 2022] Kingma, D. P. e Welling, M. (2022). Auto-encoding variational bayes. *arXiv:1312.6114*.

[Mascarenhas e Agarwal 2021] Mascarenhas, S. e Agarwal, M. (2021). A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification. *International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)*, 1:96–99.

[Müller-Franzes et al. 2022] Müller-Franzes, G., Niehues, J., Khader, F., Arasteh, S., Haarburger, C., Kuhl, C., Wang, T., Han, T., Nebelung, S., Kather, J., e Truhn, D. (2022). Diffusion probabilistic models beat gans on medical images. *arXiv:2212.07501*.

[Pan et al. 2020] Pan, Z., Yu, W., Wang, B., Xie, H., Sheng, V. S., Lei, J., e Kwong, S. (2020). Loss functions of generative adversarial networks (gans): Opportunities and challenges. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 4(4):500–522.

[Pinaya et al. 2022] Pinaya, W. H. L., Tudosiu, P.-D., Dafflon, J., Costa, P., Fernandez, V., Nachev, P., Ourselin, S., e Cardoso, M. J. (2022). Brain imaging generation with latent diffusion models. *arXiv:2209.07162*.

[Rautela et al. 2024] Rautela, M., Senthilnath, J., Huber, A., e Gopalakrishnan, S. (2024). Toward deep generation of guided wave representations for composite materials. *IEEE Transactions on Artificial Intelligence*, 5(3):1102–1109.

[Rombach et al. 2022] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., e Ommer, B. (2022). High-resolution image synthesis with latent diffusion models. *arXiv:2112.10752*.

[Sagers et al. 2023] Sagers, L. W., Diao, J. A., Melas-Kyriazi, L., Groh, M., Rajpurkar, P., Adamson, A. S., Rotemberg, V., Daneshjou, R., e Manrai, A. K. (2023). Augmenting medical image classifiers with synthetic data from latent diffusion models. *arXiv:2308.12453*.

[Singla et al. 2023] Singla, C., Kaur, R., Singh, J., Nisha, N., Singh, A. K., e Singh, T. (2023). Fine tuned pre-trained deep neural network for automatic detection of diabetic retinopathy using fundus images. *International Journal of Intelligent Systems and Applications in Engineering*, 11(9s):735–742.

[Sugawara et al. 2019] Sugawara, Y., Shiota, S., e Kiya, H. (2019). Checkerboard artifacts free convolutional neural networks. *APSIPA Transactions on Signal and Information Processing*, 8(1):e9.

[Sung et al. 2021] Sung, H., Ferlay, J., Siegel, R. L., Laversanne, M., Soerjomataram, I., Jemal, A., e Bray, F. (2021). Global cancer statistics 2020: Globocan estimates of incidence and mortality worldwide for 36 cancers in 185 countries. *CA: A Cancer Journal for Clinicians*, 71(3):209–249.

[Wang et al. 2023] Wang, D., Ma, C., e Sun, S. (2023). Novel paintings from the latent diffusion model through transfer learning. *Applied Sciences*, 13(18).

[Xing et al. 2021] Xing, J., Chen, C., Lu, Q., Cai, X., Yu, A., Xu, Y., Xia, X., Sun, Y., Xiao, J., e Huang, L. (2021). Using bi-rads stratifications as auxiliary information for breast masses classification in ultrasound images. *IEEE Journal of Biomedical and Health Informatics*, 25(6):2058–2070.

[Xue et al. 2022] Xue, P., Wang, J., Qin, D., Yan, H., Qu, Y., Seery, S., Jiang, Y., e Qiao, Y. (2022). Deep learning in image-based breast and cervical cancer detection: a systematic review and meta-analysis. *npj Digital Medicine*, 5(19).