Modelos de Difusão Latente na Super-Resolução de Imagens Mamográficas

Dayvison Gomes de Oliveira

CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

João Pessoa, PB

2024

Dayvison Gomes de Oliveira

Modelos de Difusão Latente na Super-Resolução de Imagens Mamográficas

Monografia apresentada ao curso
Ciência de dados e Inteligência Artificial
do Centro de Informática, da Universidade Federal da Paraíba,
como requisito para a obtenção do grau de bacharel

Orientador: Yuri de Almeida Malheiros Barbosa

Maio de 2024

Catalogação na publicação
Seção de Catalogação e Classificação

D276m Oliveira, Dayvison Gomes de.
Modelos de difusão latente na super-resolução de imagens mamográficas / Dayvison Gomes de Oliveira. - João Pessoa, 2024.
48 f. : il.

Orientação: Yuri de Almeida Malheiros Barbosa.
TCC (Graduação) - UFPB/CI.

1. Modelos generativos. 2. Inteligência artificial. 3. Modelos de difusão latente. 4. AutoencoderKL. 5. Super-resolução. I. Barbosa, Yuri de Almeida Malheiros. II. Título.

UFPB/CI CDU 004.8

Elaborado por Michelle de Kássia Fonseca Barbosa - CRB-738

CENTRO DE INFORMÁTICA
UNIVERSIDADE FEDERAL DA PARAÍBA

Trabalho de Conclusão de Curso de Ciência de Dados e Inteligência Artificial intitulado ***Modelos de Difusão Latente na Super-Resolução de Imagens Mamográficas*** de autoria de Dayvison Gomes de Oliveira, aprovada pela banca examinadora constituída pelos seguintes professores:

Prof. Dr. Yuri de Almeida Malheiros Barbosa
Universidade Federal da Paraíba (UFPB)

Prof. Dr. Thaís Gaudencio do Rêgo
Universidade Federal da Paraíba (UFPB)

Prof. Dr. Bruno Barufaldi
Universidade da Pensilvânia (UPenn)

Prof. Dr. Telmo De Menezes E Silva Filho
Universidade Federal da Paraíba (UFPB)

João Pessoa, 7 de maio de 2024

Centro de Informática, Universidade Federal da Paraíba
Rua dos Escoteiros, Mangabeira VII, João Pessoa, Paraíba, Brasil CEP: 58058-600
Fone: +55 (83) 3216 7093 / Fax: +55 (83) 3216 7117

"A vida não é sobre esperar a tempestade passar, mas aprender a dançar na chuva."

AGRADECIMENTOS

Gostaria de expressar minha profunda gratidão aos meus amigos, familiares e professores que estiveram ao meu lado ao longo deste curso. Suas palavras de incentivo, apoio incondicional e orientação foram fundamentais para minha jornada acadêmica. Sem o encorajamento de vocês, não teria sido possível alcançar este marco significativo em minha vida. Agradeço de coração a cada um de vocês por seu apoio constante e por tornarem esta jornada tão especial.

RESUMO

Nos anos recentes os modelos generativos ganharam popularidade para diversos tipos de tarefas, tanto como geração de texto e como também para geração de imagens. Este estudo tem como objetivo principal desenvolver um modelo generativo para realizar a tarefa de super-resolução em imagens, mais especificamente, em regiões de interesse de mamografias, utilizando *AutoencodersKL* e modelos de difusão latente. A tarefa de super-resolução consiste em aumentar a resolução de uma imagem, isso é feito a partir de uma imagem de baixa resolução, muitas vezes com perda de qualidade, e o objetivo é gerar uma versão de alta resolução que melhore a qualidade visual. É especialmente útil em áreas como processamento de imagens médicas, onde a precisão é crucial para diagnósticos precisos. No contexto de mamografias, a super-resolução pode ajudar a revelar detalhes sutis que podem passar despercebidos em imagens de baixa resolução, contribuindo assim para uma detecção mais precisa de anomalias e para o desenvolvimento de tratamentos mais eficazes. Os dados consistem em duas bases de imagens não rotuladas, uma de alta e outra de baixa resolução, ambas com 625 imagens em escala de cinzas. Os resultados foram avaliados com FID, SSIM e PSNR, alcançando valores de 0,00006, 0,80 e 26, respectivamente na tarefa de super-resolução que são satisfatórios e comparáveis à literatura. No entanto, devido à heterogeneidade e à natureza simulada dos dados, o modelo *AutoencoderKL* apresentou imagens com degradação de qualidade. Sugere-se aprofundar o estudo para melhorar a acurácia do modelo, explorando diferentes estratégias de treinamento, diferentes normalizações e outros *Autoencoders* para a geração da representação latente. Além disso, o trabalho contribui para a comunidade ao propor uma nova pipeline de treinamento e avaliação para a tarefa de super-resolução, com o código containerizado para reprodução escalável em diferentes sistemas operacionais e conjuntos de dados.

Palavras-chave: <Modelos Generativos>, <Inteligência Artificial>, <Modelos de Difusão Latente>, <AutoencoderKL>, <Super-resolução>.

ABSTRACT

In recent years, generative models have gained popularity for various tasks, including text and image generation. In this thesis, I will develop a generative model to improve image resolution of mammography simulations using super-resolution techniques based on latent diffusion with *AutoencoderKL* methods. Super-resolution task involves increasing the resolution of an image, typically from a low-resolution image, often with loss of quality, aiming to generate a high-resolution version that enhances visual quality. It is particularly useful in fields like medical image processing, where precision is crucial for accurate diagnoses. In the context of mammography, super-resolution can help reveal subtle details that may go unnoticed in low-resolution images, thereby contributing to more precise anomaly detection and the development of more effective treatments. The data consists of two sets of unlabeled images, one in high resolution and the other in low resolution, both containing 625 grayscale images. The results were evaluated using FID, SSIM, and PSNR, achieving values of 0.00006, 0.80, and 26, respectively in the super-resolution task, which are satisfactory and comparable to the literature. However, due to the heterogeneity and simulated nature of the data, the *AutoencoderKL* model produced images with slightly degraded images. Further study is suggested to enhance the model's effectiveness by exploring different training strategies, normalization techniques, and alternative *Autoencoders* for latent representation generation. Additionally, this work contributes to the community by proposing a new pipeline for training and evaluation in the super-resolution task, with containerized code for scalable reproducibility across different operating systems and datasets.

Keywords: <Generative Models>, <Artificial Intelligence>, <Latent Diffusion Models>, <AutoencoderKL>, <Super-resolution>.

LISTA DE FIGURAS

LISTA DE TABELAS

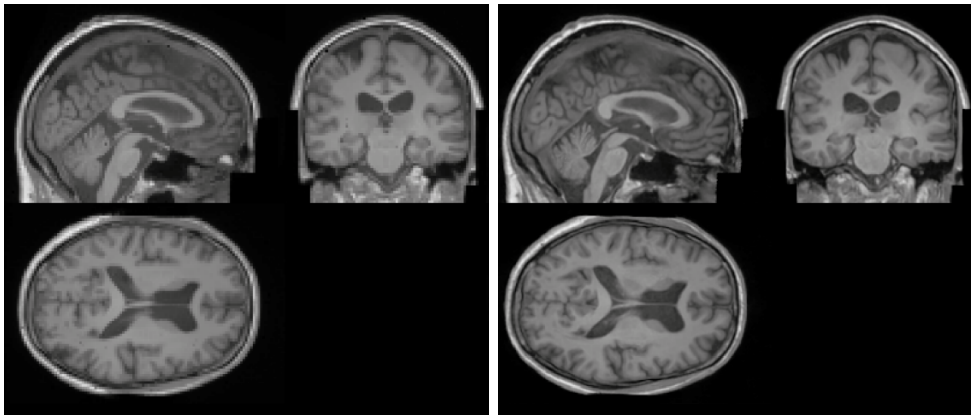
LISTA DE ABREVIATURAS

- FID - Distância de Interseção Frechet
- LDM - Modelo de Difusão Latente
- MAE - Erro Absoluto médio
- MSE - Erro Quadrático médio
- PSNR - Relação Sinal-Ruído de Pico
- SSIM - Índice de Similaridade Estrutura
- VAEs - Autoencoders Variacionais

SUMÁRIO

1 INTRODUÇÃO

A busca por melhorias na qualidade das imagens médicas tornou-se crucial na pesquisa científica. Laudos imprecisos ou mal interpretados de exames de imagem representam desafios significativos para médicos solicitantes, dificultando diagnósticos precisos e podendo resultar em tratamentos inadequados ou atrasos no atendimento médico. Essa situação não só compromete a saúde do paciente, mas também pode levar a procedimentos invasivos desnecessários. Estratégias avançadas, como a aplicação de *AutoencodersKL* e modelos de difusão latente, têm se destacado como promissoras para otimizar a super-resolução, que visa aprimorar diagnósticos médicos ao aumentar a resolução de imagens de baixa qualidade para revelar detalhes sutis. Esse aprimoramento é especialmente crucial em áreas como mamografias, onde a detecção precisa de anomalias é fundamental para tratamentos eficazes. Um exemplo do aumento da qualidade da imagem pode ser observado na figura 1.



(a) Imagem de baixa resolução **(b) Imagens de alta resolução**

Figura 1: Imagem de baixa resolução vs Imagens de alta resolução. Fonte: Pinaya et al. (2023)

Os *AutoencodersKL* são uma variante refinada dos *Autoencoders* convencionais, em que é inserido a divergência de Kullback-Leibler (KL) (KINGMA; WELLING, 2022) na função de custo. Essa adaptação torna-se particularmente relevante ao lidar com uma complexidade e variabilidade de imagens que demandam uma representação latente mais robusta. Ao incorporar a divergência KL, adquirimos a capacidade de aprender representações mais eficientes, capturando de maneira mais precisa a distribuição dos dados subjacentes.

No âmbito da super-resolução em mamografias, a aplicação dos *AutoencodersKL* possibilita a aprendizagem de representações latentes significativas das imagens de alta resolução, viabilizando uma compreensão mais profunda das características essenciais presentes neste tipo de exame.

Adicionalmente, a inclusão de modelos de difusão latente se apresenta como uma estratégia complementar para aprimorar a qualidade da super-resolução. Estes modelos, ao abordarem complexidades na distribuição dos dados, contribuem para uma reconstrução mais fiel das características intrínsecas das imagens. A integração destes modelos em conjunto com *AutoencodersKL* potencializa a capacidade do sistema em fornecer imagens de alta resolução, preservando detalhes relevantes para a análise clínica.

1.1 Objetivo geral

Este trabalho tem como objetivo desenvolver um modelo generativo para realizar super-resolução em regiões de interesse de mamografias. Para alcançar esse objetivo, utilizaremos dois modelos, um *Autoencoderkl* e um modelo de difusão latente com base no trabalho de Pinaya et al. (2023).

1.2 Objetivos específicos

Abaixo serão listados os objetivos específicos para resolução do trabalho:

- Treinar um modelo *autoencoderkl*.
- Treinar um modelo de difusão latente.
- Comparar os modelos *autoencoderkl* e de difusão latente.

Adicionalmente, este trabalho também contribui com:

- Uma nova *pipeline* de treinamento e avaliação para realizar super-resolução.
- Conteinerização de todo o código baseado no trabalho de Pinaya et al. (2023).

1.3 Estrutura da monografia

A estrutura deste trabalho foi dividida em cinco capítulos: introdução, conceitos gerais e revisão de literatura, metodologia, apresentação e análise dos resultados, conclusões e trabalhos futuros. Na introdução, apresentamos a definição do problema, o objetivo geral do estudo e os objetivos específicos. Em conceitos gerais e revisão de literatura, abordamos os conceitos importantes para o entendimento da pesquisa. Na metodologia, descrevemos os passos a serem seguidos para reproduzir os experimentos. Em seguida, na capítulo de apresentação e análise dos resultados, mostramos os resultados obtidos descritos na metodologia. Por fim, na capítulo de conclusões e trabalhos futuros, apresentamos as observações a partir dos resultados alcançados, juntamente com sugestões para trabalhos futuros.

2 CONCEITOS GERAIS E REVISÃO DA LITERATURA

Este capítulo é estruturado em 7 seções principais e cada uma delas aborda um aspecto que proporciona uma visão detalhada das teorias, modelos e métricas utilizadas, fornecendo assim o necessário para o entendimento e compreensão do trabalho. As seções tratam de Visão Computacional, Redes Neurais, Autoencoder, AutoencoderKL, Difusão Latente, Métricas de Avaliação e Trabalhos relacionados.

2.1 Visão Computacional

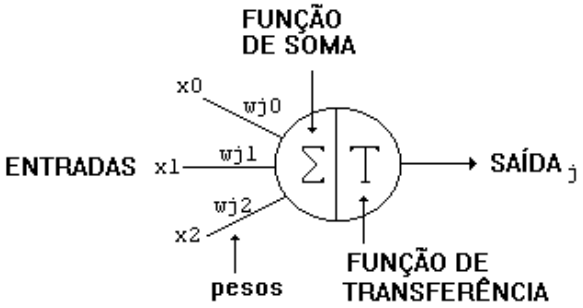
Ballard (1982), em seu livro clássico, aborda a temática da visão computacional, definindo-a como a ciência responsável pela forma como um computador "enxerga" o meio à sua volta, extraindo informações significativas, a partir de imagens capturadas por câmeras de vídeo, sensores, *scanners*, entre outros dispositivos. Nessa abordagem, é possível reconhecer e manipular os objetos que compõem uma imagem. Essa definição ressoa com a ideia contemporânea de visão computacional, que é considerada um campo interdisciplinar na junção da ciência da computação e da Inteligência Artificial. Seu propósito fundamental é analisar, interpretar e extrair informações relevantes de imagens e vídeos, permitindo a tomada de decisões ou a geração de dados de interesse para futuras aplicações.

Essencialmente, a visão computacional busca emular a capacidade visual humana, como destacado por pesquisadores do Massachusetts Institute of Technology - MIT, ao definirem o objetivo de "ensinar computadores a enxergarem como humano" (GREENE-MEIRR, 2008). Assim como o olho humano consegue perceber e interpretar objetos em uma imagem de forma rápida, isso acontece no córtex visual do cérebro, uma das partes mais complexas no sistema de processamento cerebral. Alguns cientistas concentram seus estudos na compreensão do funcionamento dessa região cerebral para trazer tais ideias para a visão computacional como visto em Greenemeirr (2008). Dessa forma, ao fornecer ao computador informações precisas a partir de imagens e vídeos, a visão computacional capacita-o a realizar tarefas inteligentes, aproximando-se da inteligência humana, que utiliza de redes neurais, e permitindo a execução de diversas tarefas, como reconhecimento de objetos, avaliação de distâncias e compreensão do contexto de um ambiente.

2.2 Redes Neurais

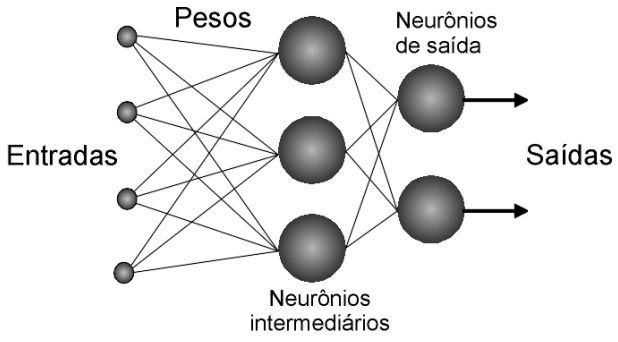
A construção de redes neurais artificiais é inspirada na forma como o cérebro humano realiza o seu processamento como visto na Seção 2.1, que é consideravelmente diferente da forma como os computadores o fazem. O cérebro tem um processamento altamente complexo, não-linear e paralelo. De forma geral, uma rede neural artificial é uma máquina projetada com o objetivo de simular a maneira como o cérebro opera. Os componentes deste modelo podem ser eletrônicos ou algoritmos de programação, e seguem a mesma estrutura básica de uma rede neural orgânica, ou seja, empregam interligações maciças de células computacionais denominadas "neurônios". A definição formal adotada por Haykin (2000) é: "uma rede neural é um processador maciço e paralelamente distribuído, constituído de unidades de processamento simples, que têm a propensão natural para armazenar conhecimento experimental e torná-lo disponível para o uso". Um exemplo de neurônio artificial é visto na Figura 2.

Figura 2: Modelo neurônio artificial de Tafner (1998)



De acordo com Tafner (1998), podemos combinar diversos neurônios artificiais para formar o que é chamada de rede neural artificial. As entradas, simulando uma área de captação de estímulos, podem ser conectadas em muitos neurônios, resultando, em uma série de saídas, onde cada neurônio é representado como na Figura 3. As diferentes possibilidades de conexões entre as camadas de neurônios podem gerar n números de estruturas diferentes, como a que veremos na Seção 2.3.

Figura 3: Exemplo de rede neural artificial de Tafner (1998)

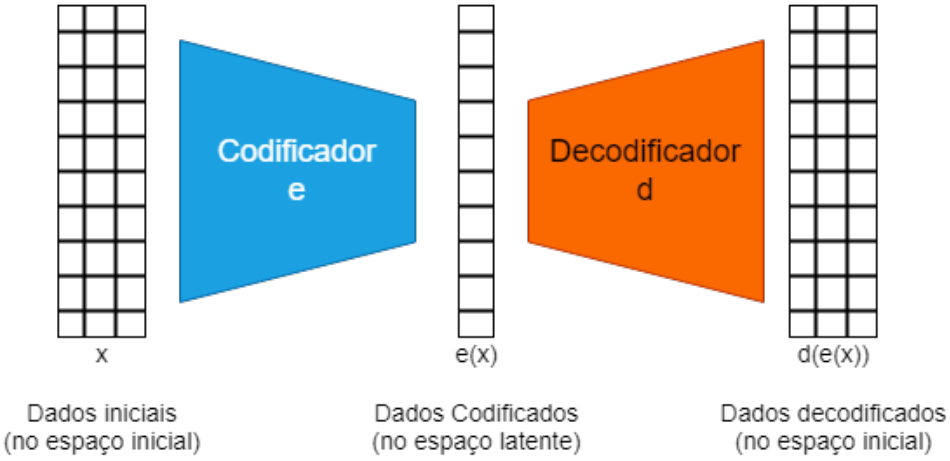


2.3 Autoencoder

Conforme destacado pela *Data Science Academy* (ACADEMY, 2022), os *autoencoders* representam uma classe fundamental de algoritmos de aprendizado profundo não supervisionado, cuja finalidade é a captura de representações compactas de dados. Esses modelos, divididos em um codificador e um decodificador, desempenham o papel crucial de transformar os dados de entrada em uma representação de baixa dimensionalidade, conhecida como espaço latente, e reconstruir os dados originais a partir dessa representação. Além disso, os *autoencoders* são reconhecidos como modelos generativos (GOODFELLOW et al., 2014), permitindo a síntese de dados similares aos dados de treinamento.

De acordo com a visão da DSA Academy (2022), a redução da dimensionalidade pelos *autoencoders* é alcançada pela minimização da diferença entre os dados originais e suas reconstruções. Durante o processo de treinamento, o objetivo é que a saída do *autoencoder* seja próxima à entrada, e a discrepância entre os dados reconstruídos e os originais é minimizada por meio de uma função de perda. A Figura 4 demonstra a estrutura genérica de um *autoencoder*.

Figura 4: Estrutura de um Autoencoder. Fonte: Academy (2022) e adaptado de Teixeira (2023)



Dentro do universo das redes adversárias generativas (GANs, do inglês *Generative Adversarial Networks*) (GOODFELLOW et al., 2014), existe outro modelo que merece nossa atenção: o *Autoencoder* Variacional (VAE, do inglês *Variational Autoencoder*) (KINGMA; WELLING, 2022). O VAE é uma variação do *autoencoder* clássico, que adiciona um componente probabilístico à sua estrutura. Enquanto o *autoencoder* convencional é eficaz na reconstrução direta dos dados de entrada, o VAE é projetado para aprender uma distribuição probabilística sobre o espaço latente das representações dos dados, assim, permitindo ao VAE gerar novas amostras de dados, introduzindo um elemento de aleatoriedade durante o processo de geração (KINGMA; WELLING, 2022).

Na próxima seção, detalharemos mais profundamente no VAE e exploraremos suas características e aplicações no próximo tópico.

2.4 AutoencoderKL

O VAE com função de custo *Kullback-Leibler* (KL) foi introduzido no trabalho de Kingma e Welling (2022). Como dito na seção anterior 2.3, é uma versão probabilística do *autoencoder* determinístico. Enquanto o *autoencoder* projeta a entrada para uma incorporação específica no espaço latente, o VAE projeta os dados de entrada para distribuições de probabilidade. Portanto, a vantagem do VAE é que ele pode gerar novos dados ou imagens, por meio de amostragem aleatória (KINGMA; WELLING, 2022).

A divergência KL (PAIN, 2023) é calculada comparando a distribuição aprendida pelo *AutoencoderKL* e uma distribuição normal. A função objetivo otimiza a incorporação do espaço latente do VAE. Com isso, temos que uma distribuição normal ou distribuição gaussiana é um tipo de distribuição de probabilidade contínua para uma variável aleatória com valores reais que possui a seguinte função de densidade:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{-1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}, \tag{1}$$

onde μ é a média e σ^2 é a variância.

De acordo com Ou (2022), existem duas funções de custo no treinamento de um VAE: Erro Quadrático Médio (MSE) definida na equação 2, para calcular o erro entre a imagem de entrada e a imagem reconstruída, e a Divergência KL, para calcular a distribuição codificada e a distribuição normal com média 0 e variância 1.

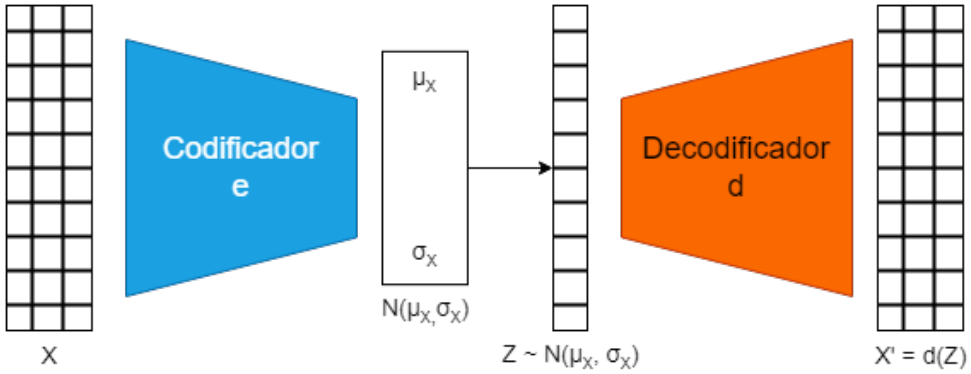
$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(y_i - f_i\right)^2. \tag{2}$$

Podemos calcular a divergência KL definida na equação 3 comparando duas Gaussianas com a seguinte fórmula:

$$KL = \left(N\left(x|\mu_1, \sigma_1\right)||N\left(x|\mu_2, \sigma_2\right)\right) = \log\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{\sigma_1^2 + \left(\mu_1 - \mu_2\right)^2}{2 \cdot \sigma_2^2} - \frac{1}{2}. \tag{3}$$

Na Figura 5, observamos a estrutura e o fluxo dos dados de entrada, que é passado para o codificador, até alcançar a representação latente e, a partir dela, gerar uma amostra de uma distribuição normal padrão, como mencionado por Ou (2022). Essa amostra probabilística é então encaminhada para o decodificador, que reconstrói a imagem original.

Figura 5: Estrutura do *AutoencoderKL*. Fonte: Academy (2022) e adaptado de Teixeira (2023)



Conforme discutido pela DSA Academy (2022), essa abordagem probabilística não apenas aprimora a representação dos dados em um espaço latente, mas também estabelece uma estrutura regularizada nesse espaço. Os VAEs têm a capacidade de capturar a essência de um conjunto de dados e representá-lo em uma dimensão menor do que a original, possibilitando a representação uniformizada de diferentes tipos de dados nesse espaço latente. Dentro do contexto de modelos generativos, essa habilidade pode ser utilizada para treinar outros modelos no espaço latente, em vez do espaço original do conjunto de dados (ACADEMY, 2022), como será abordado no próximo tópico .

2.5 Difusão Latente

De acordo com Ho, Jain e Abbeel (2020), os modelos de difusão são definidos como: "Um modelo de difusão ou modelo de difusão probabilística é uma cadeia de Markov parametrizada treinada usando inferência variacional para produzir amostras correspondentes aos dados após um tempo finito". Em termos simples, os modelos de difusão têm a capacidade de produzir dados que se parecem com aqueles com os quais foram treinados. Por exemplo, se o modelo for treinado com imagens de gatos, ele será capaz de gerar novas imagens realistas de gatos.

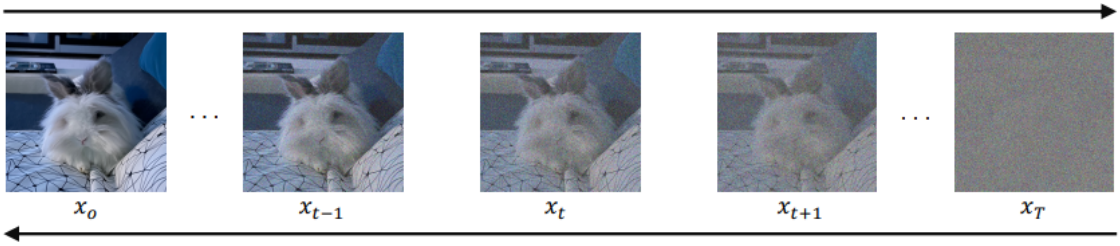
Os modelos de difusão são fundamentados no princípio e nos fundamentos matemáticos de um modelo probabilístico, que tem a capacidade de analisar e prever o comportamento de um sistema dinâmico ao longo do tempo. A definição técnica especifica que esses modelos são cadeias de *Markov* parametrizadas e treinadas com inferência variacional (SALIMANS; KINGMA; WELLING, 2015). Parafraseando Wang (2022): Uma cadeia de *Markov* é um processo estocástico que passa de um estado para outro em um conjunto de estados finitos ou infinitos, onde a probabilidade de transição depende apenas do estado atual e não da sequência de eventos que levou a esse estado.

Durante o treinamento do modelo por meio de inferência variacional como dito em

(SALIMANS; KINGMA; WELLING, 2015), são realizados cálculos complexos envolvendo distribuições de probabilidade. O objetivo desse processo é identificar os parâmetros precisos da cadeia de *Markov* que correspondam aos dados observados, sejam eles conhecidos ou reais, em um determinado ponto no tempo. Esse procedimento visa minimizar a função de perda do modelo, que é a discrepância entre o estado previsto (desconhecido) e o estado observado (conhecido) . Após o treinamento, o modelo torna-se capaz de gerar amostras que refletem os dados observados. Essas amostras representam possíveis trajetórias ou estados que o sistema poderia assumir ao longo do tempo, sendo que cada trajetória possui uma probabilidade distinta de ocorrência. Consequentemente, o modelo é capaz de prever o comportamento futuro do sistema, gerando um conjunto de amostras e determinando as probabilidades correspondentes à ocorrência desses eventos (HO; JAIN; ABBEEL, 2020).

De acodo com Sajid (2023), os modelos de difusão são modelos generativos profundos que operam inserindo ruído (geralmente ruído gaussiano) nos dados de treinamento disponíveis, em um processo conhecido como difusão direta. Posteriormente, eles revertem esse processo, removendo o ruído em um procedimento chamado redução de ruído ou difusão reversa introduzido em Ho, Jain e Abbeel (2020), a fim de recuperar os dados originais. Ao longo do treinamento, o modelo aprende gradualmente a eliminar o ruído (CROITORU et al., 2023), resultando na geração de novas imagens de alta qualidade a partir de sementes aleatórias, que são essencialmente imagens com ruído aleatório. Essa capacidade de reduzir o ruído de maneira aprendida é ilustrada na Figura 6.

Figura 6: Um modelo de difusão adiciona e remove ruído de uma imagem. Fonte: Croitoru et al. (2023)



O estudo realizado por Rombach et al. (2021) apresenta uma nova abordagem na síntese de imagens e super-resolução, utilizando modelos de difusão no espaço latente de *autoencoders* pré-treinados, ou seja, os modelos operam em um espaço latente comprimido. Essa técnica possibilita o treinamento de modelos de difusão de forma mais eficiente em termos de recursos computacionais, sem comprometer a qualidade ou a flexibilidade, melhorando a eficiência e amostragem de modelos de difusão de remoção de ruído.

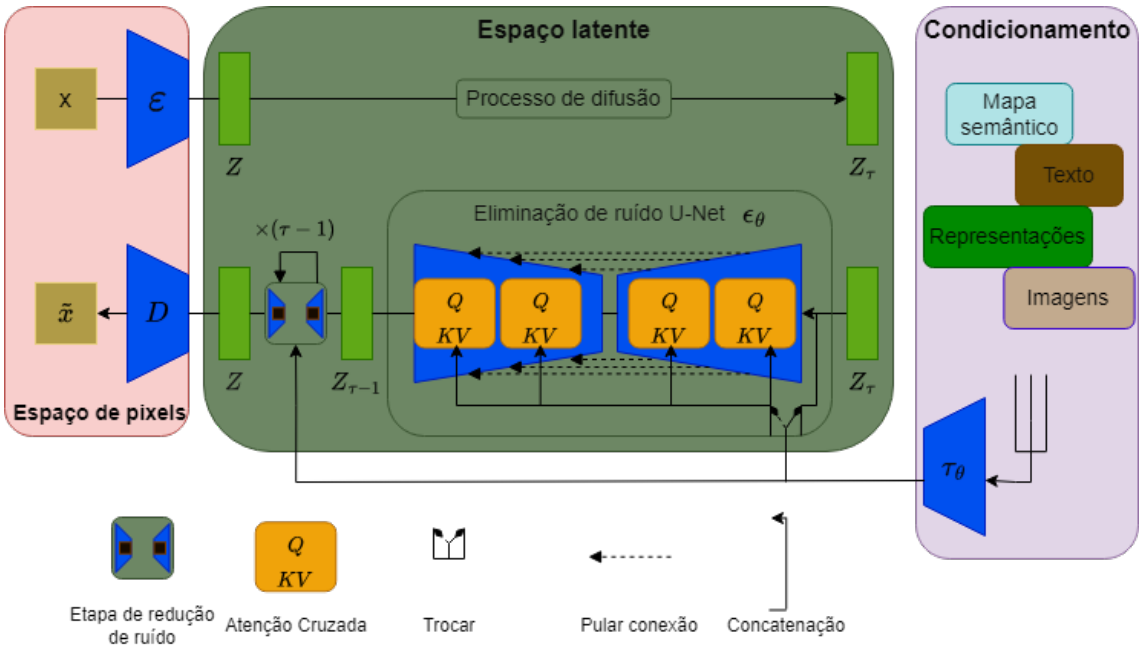
Enquanto os modelos de difusão tradicionais lidam diretamente com os *pixels* das imagens (ROMBACH et al., 2021), o que pode demandar considerável tempo e recursos computacionais para treinamento e inferência, a estratégia proposta por Rombach et al. (2021) resulta em um equilíbrio otimizado entre simplificação da complexidade e preser-

vação dos detalhes, resultando em uma notável melhoria na fidelidade visual das imagens geradas.

De acordo com Rombach et al. (2021), os modelos de difusão latente (LDMs) demonstram um desempenho competitivo em diversas tarefas de síntese de imagem e no processo de super-resolução, superando métodos de ponta em termos de pontuações Distância de Interseção Frechet (FID) e eficiência computacional.

Na Figura 7, é ilustrado todo o processo, desde o início até a obtenção da imagem final. Inicialmente, o codificador do *autoencoder* é utilizado para obter o espaço latente z, seguido pela aplicação do processo de difusão direta para adição de ruído e obtenção de z_t. Em seguida, um texto ou imagem é passado para o *transformer* T_θ e concatenado ao espaço latente. Posteriormente, é realizado o processo de difusão reversa que utiliza camadas de atenção cruzada (do inglês, *cross-attention*) QKV que significa (*Query, Key, Value*) na arquitetura do modelo de difusão, o tornando flexíveis para diferentes tipos de entradas condicionais, como texto, para recuperar o espaço latente original, utilizando um *Denoising Diffusion Probabilistic Model* (HO; JAIN; ABBEEL, 2020). Após essa etapa, é feita a passagem para o decodificador do *autoencoder*, resultando na geração da imagem final.

Figura 7: Pipeline do modelo de difusão latente adaptado de Rombach et al. (2021)



Os modelos desenvolvidos neste estudo requerem avaliação para determinar sua capacidade de realizar super-resolução, ou seja, o quão bem a imagem gerada se parece com a original. No contexto de super-resolução, diversas métricas são empregadas para esse propósito, incluindo a Distância Fréchet de Inception (FID), Relação Sinal-Ruído de

Pico (PSNR) e Índice de Similaridade Estrutural (SSIM), que serão descritas no próximo tópico.

2.6 Métricas de Avaliação

Este estudo faz uso de várias métricas para avaliar a qualidade das imagens geradas pelo modelo de difusão proposto. A seguir, descreveremos detalhadamente três métricas essenciais que são fundamentais para nossa análise, que são: Distância de Interseção Frechet (FID) (Subseção 2.6.1), Relação Sinal-Ruído de Pico (PSNR) (Subseção 2.6.2) e Índice de Similaridade Estrutural (SSIM) (Subseção 2.6.3).

2.6.1 Distância de Interseção Frechet

Distância de Interseção Frechet (do inglês, *Frechet Inception Distance* - (FID), proposta por Heusel et al. (2018), é uma métrica para avaliar a qualidade das imagens geradas e desenvolvida especificamente para avaliar o desempenho de GANs.

O objetivo no desenvolvimento da métrica FID é avaliar imagens sintéticas com base nas estatísticas de uma coleção de imagens sintéticas, em comparação com as estatísticas de uma coleção de imagens reais do domínio alvo. A métrica FID usa o modelo *Inception v3*(SZEGEDY et al., 2015). É na última camada de *pooling*, antes da classificação de saída de imagens, seguindo com os cálculos da média e a covariância das imagens e, é feito o cálculo da distância entre essas duas distribuições, usando a FID, também chamada de distância *Wasserstein-2* (KANTOROVICH, 1960). A métrica é calculada da seguinte maneira:

$$d^2 = \|\mu_1 - \mu_2\|^2 + T_r\left(\sigma_1 + \sigma_2 - 2 \cdot \sqrt{(\sigma_1 \cdot \sigma_2)}\right). \tag{4}$$

Onde d^2 é a distância ao quadrado, μ_1 e μ_2 refere-se à média de características das imagens reais e geradas. Para o modelo *Inception v3* e *Resnet50*, temos vetores de 2.048 elementos pois é a quantidade de neurônios da última camada antes da classificação de saída, onde cada elemento representa a média da característica observada nas imagens, σ_1 e σ_2 são as matrizes de covariância para os vetores de características reais e gerados.

O $\|\mu_1 - \mu_2\|^2$ refere-se à soma dos quadrados das diferenças entre os dois vetores com as médias. *Tr* é a operação de traço na álgebra linear, ou seja, a soma dos elementos ao longo da diagonal principal de uma matriz quadrada.

2.6.2 Relação Sinal-Ruído de Pico

A Relação Sinal-Ruído de Pico (do inglês, *Peak Signal-to-Noise Ratio* - PSNR), visto no trabalho de Wang et al. (2004) e em Horé e Ziou (2010), é um termo da engenharia para a relação entre a potência máxima possível de um sinal e a potência do ruído corruptor, que afeta a fidelidade de sua representação. Como muitos sinais têm uma faixa dinâmica muito ampla, o PSNR é geralmente expresso como uma quantidade logarítmica usando a escala de decibéis.

É comumente usado para quantificar a qualidade de reconstrução de imagens e vídeos e é definido através do MSE, utilizado para medir a qualidade da reconstrução de *codecs* de compactação com perdas (por exemplo, para compactação de imagem). O sinal, neste caso, são os dados originais e o ruído é o erro introduzido pela compressão. Ao comparar *codecs* de compressão, o PSNR é uma aproximação da percepção humana da qualidade da reconstrução (KELEş et al., 2021) e pode ser utilizado também para avaliar as imagens geradas da super-resolução. É calculada da seguinte maneira:

$$PSNR = 10 \log_{10} \left(\frac{R^2}{MSE} \right). \tag{5}$$

Onde o MSE é a diferença de pixels da imagem ao quadrado, R^2 é a máxima flutuação da imagem de entrada. Por exemplo, se a imagem de entrada tem um tipo de dado de ponto flutuante de dupla precisão, então R é 1. Se ela possui um tipo de dado de inteiro sem sinal de 8 bits, R é 255, etc.

2.6.3 Índice de Similaridade Estrutural

O Índice de Similaridade Estrutural (do inglês, *Structural Similarity Index Measure* - SSIM), visto no trabalho de Horé e Ziou (2010), é um método utilizado para prever a qualidade percebida de imagens e vídeos digitais. O SSIM é usado para medir a similaridade entre duas imagens e é classificado como uma métrica de referência completa, o que significa que a medição, ou previsão da qualidade da imagem, é baseada em uma imagem inicial não comprimida ou livre de distorção, usada como referência.

O SSIM é um modelo baseado na percepção humana de acordo com Wang, Simoncelli e Bovik (2003), que considera a degradação da imagem como uma mudança percebida na informação estrutural, ao mesmo tempo em que incorpora importantes fenômenos perceptivos, incluindo termos de mascaramento de luminância e mascaramento de contraste. A diferença em relação a outras técnicas, como o MSE ou o PSNR, é que essas abordagens estimam erros absolutos. A informação estrutural se baseia na ideia de que os pixels têm fortes interdependências, especialmente quando estão espacialmente próximos. Essas de-

pendências contêm informações importantes sobre a estrutura dos objetos na cena visual (WANG; SIMONCELLI; BOVIK, 2003).

É calculada da seguinte forma:

$$SSIM(x, y) = \frac{(2 \cdot \mu_x \mu_y + C_1)(2 \cdot \sigma_{xy} + C_2)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)}. \tag{6}$$

Onde:

- μ_x, média de amostra de pixel de x;
- μ_y, média de amostra de pixel de y;
- σ_x^2 variância de x;
- σ_y^2 variância de y;
- σ_{xy} covariância de x e y;
- $c1 = (k1L)^2$ e $c2 = (k2L)^2$ variáveis para estabilizar a divisão por denominador pequeno;
- L, valor máximo da faixa dinâmica dos valores dos pixels. $(2^{\#bitsporpixel}-1)$;
- $k_1 = 0,01$, $k_2 = 0,03$ por padrão;

Embora existam outras métricas disponíveis para avaliar a qualidade das imagens na tarefa de super-resolução, escolhemos estas devido à sua eficácia comprovada e relevância para nossa análise. A métrica FID é reconhecida por avaliar a qualidade das imagens geradas por modelos generativos como visto sendo utilizado em Rombach et al. (2021) e Pinaya et al. (2023), oferecendo uma medida objetiva da discrepância estatística entre as imagens geradas e as reais, o que nos permite analisar a distância entre suas distribuições Heusel et al. (2018). O PSNR é amplamente usado na avaliação de reconstruções de imagem, especialmente em contextos de compressão com perdas, permitindo quantificar o MSE entre as imagens originais e as reconstruídas (WANG et al., 2004), crucial para compreender a fidelidade das imagens geradas. O SSIM leva em consideração a percepção humana da qualidade de imagem, incorporando aspectos de textura, contraste e estrutura, sendo particularmente relevante para a tarefa de super-resolução, onde nosso objetivo é nos aproximar da imagem de referência, garantindo melhor qualidade perceptual (WANG; SIMONCELLI; BOVIK, 2003).

Na próxima seção, abordaremos os trabalhos relacionados a este estudo, explorando suas propostas, metodologias e as métricas adotadas para avaliação.

2.7 Trabalhos Relacionados

Nesta seção, apresentaremos uma revisão dos trabalhos relacionados ao tema abordado neste estudo, super-resolução de imagens em contextos médicos e o uso de modelos de difusão. Quatro artigos que exploraram modelos de difusão, em contexto de síntese de imagens e super-resolução, são apresentados a seguir.

No trabalho de Rombach et al. (2021) foram introduzidos os LDMs, como uma abordagem inovadora para a síntese de imagens de alta resolução. A metodologia consiste em treinar um LDM no espaço latente de um *autoencoder* pré-treinado. Isso permite reduzir a complexidade computacional e melhorar a eficiência do processo de geração de imagens de alta resolução. O LDM é composto por camadas de *Cross-Attention*, que permitem a entrada de condicionantes, como texto ou caixas delimitadoras. Essas camadas *Cross-Attention* tornam o LDM um gerador poderoso e flexível para diferentes tipos de entradas condicionais, como por exemplo, textos e imagens.

Os resultados obtidos com o modelo de difusão para super-resolução foram altamente competitivos em várias tarefas, incluindo síntese de imagens incondicionais, síntese de imagens condicionadas por texto e super-resolução. Além disso, esses resultados foram determinados com base em comparações com métodos anteriores e métricas de avaliação estabelecidas. O trabalho demonstrou que o modelo de difusão para super-resolução superou abordagens anteriores em termos de qualidade das amostras, como avaliado por métricas como FID, introduzido em Heusel et al. (2018), *Precision* e *Recall*. Além disso, foram realizadas análises qualitativas, onde as imagens geradas pelo modelo foram comparadas visualmente com as imagens originais de alta resolução. Os autores utilizaram métricas de avaliação de qualidade de imagem, além das já citadas, SSIM, e PSNR. Os experimentos foram realizados em diferentes conjuntos de dados, incluindo o ImageNet. Além disso, foram realizadas análises comparativas com outros métodos, como o SR3, e também foram conduzidos estudos de usuário para avaliar a preferência das imagens geradas pelo LDM, em comparação com uma linha de base de pixels.

As técnicas dos LDMs podem ser estendidas para melhorar a qualidade das imagens médicas de baixa resolução, tornando-as mais úteis para diagnóstico. Os LDMs combinam a compressão perceptual de imagens com modelos de difusão, uma estratégia relevante para aprimorar a interpretação de imagens médicas, por meio da incorporação de informações contextuais.

No modelo proposto por Li et al. (2021), denominado *SRDiff: Single Image Super-Resolution with Diffusion Probabilistic Models*, temos um modelo probabilístico de difusão de super-resolução de imagem única. O SRDiff foi desenvolvido para abordar questões comuns em métodos de super-resolução, incluindo suavização excessiva, tornando-se uma abordagem valiosa, especialmente em imagens médicas, onde a preservação de detalhes é

crucial para diagnósticos precisos. A utilização de um processo de difusão, para transformar gradualmente o ruído gaussiano em uma imagem de super-resolução, condicionada a uma imagem de baixa resolução, demonstra a relevância dessas métricas na melhoria da qualidade das imagens de baixa resolução.

O modelo foi treinado nos conjuntos de dados DIV2K e Flickr2K e avaliado no conjunto de validação DIV2K. O SRDiff utiliza um preditor de ruído condicional e um codificador de baixa resolução, com vários blocos densos residuais em cascata (RRDB). O método alcança resultados de ponta em tarefas de super-resolução facial e geral, produzindo imagens de super-resolução de alta qualidade, com detalhes ricos e menos artefatos. Os resultados experimentais mostram que o SRDiff supera os métodos existentes em termos de métricas de avaliação como PSNR, SSIM, LPIPS, e LR-PSNR, além de possuir um tamanho considerado pequeno, com 12 milhões de parâmetros.

O trabalho de Wang et al. (2023) propõe uma técnica não supervisionada usando um LDM. A relevância direta dessa abordagem reside na sua aplicação no campo das imagens médicas, concentrando-se especificamente na super-resolução de ressonância magnética (MRI, do inglês *Magnetic Resonance Image*). Essa importância decorre da necessidade de adaptar-se a diversas configurações de problemas de super-resolução de MRI, uma vez que os protocolos de aquisição variam amplamente em aplicações médicas. Os resultados experimentais, avaliados por meio das métricas SSIM e PSNR, demonstram que o método proposto supera os modelos de referência não supervisionados, tanto em termos de qualidade de imagem, quanto na preservação de detalhes.

O LDM, como modelo generativo, é capaz de capturar a distribuição anterior das imagens de MRI ponderadas em T1 (refere-se a um tipo específico de sequência, que utiliza o tempo de relaxamento longitudinal T1 como um contraste). Nesse tipo de sequência, os tecidos com diferentes tempos de relaxamento T1 aparecem com intensidades de sinal distintas, permitindo a diferenciação e visualização de diferentes estruturas e características no cérebro. O método consiste em duas estratégias distintas: InverseSR (LDM) e InverseSR (*Decoder*). A primeira, InverseSR (LDM), é aplicada a casos em que a MRI apresenta baixa esparsidade, enquanto a segunda, InverseSR (*Decoder*), é empregada em situações de alta esparsidade na imagem. Ambas as estratégias exploram diferentes espaços latentes no modelo LDM, com o objetivo de encontrar o código latente ideal para mapear a MRI de baixa resolução, para a MRI de alta resolução.

Para validar o método, mais de 100 MRIs cerebrais *T1-weighted* do conjunto de dados IXI foram utilizadas. Os resultados experimentais evidenciaram um desempenho significativamente superior, em relação aos modelos de referência. No entanto, é fundamental ressaltar que a eficácia do método é limitada pela exigência de recursos computacionais consideráveis, bem como pela heterogeneidade na saída do gerador LDM.

O trabalho de Yue, Wang e Loy (2023) apresenta um modelo de difusão inovador e eficiente para a super-resolução de imagens, reduzindo significativamente o número de etapas de difusão e melhorando a velocidade do processo de inferência. A aplicação do modelo ResShift pode ser particularmente benéfica em cenários médicos, onde a eficiência do processamento é crucial. O controle flexível da velocidade de deslocamento e da intensidade do ruído é essencial para ajustar o processo de super-resolução, de acordo com as necessidades específicas de cada imagem médica. Os resultados experimentais, avaliados por meio das métricas SSIM e PSNR, demonstram a eficácia do ResShift na super-resolução de imagens.

O modelo apresentado neste trabalho (YUE; WANG; LOY, 2023) utiliza um processo de difusão para realizar a super-resolução de imagens, e esse processo é realizado em várias etapas. Durante cada etapa de difusão, o modelo desloca o resíduo entre as imagens de alta e baixa resolução, gradualmente reduzindo a diferença entre elas e, como resultado, produzindo uma imagem de alta resolução mais precisa. O controle do deslocamento do resíduo é feito por meio de um cronograma de ruído, que influencia a velocidade e a intensidade desse deslocamento. O propósito dessas etapas de difusão é melhorar a qualidade das imagens de baixa resolução, aproximando-as da imagem de alta resolução original. Esse processo gradativo permite que o modelo obtenha resultados mais precisos e realistas.

A metodologia deste trabalho consiste em propor um modelo de difusão eficiente e inovador para a super-resolução de imagens. Esse modelo faz uso de uma cadeia de Markov para transferir informações entre as imagens de alta e baixa resolução, deslocando o resíduo entre elas. Um cronograma de ruído é desenvolvido para controlar a velocidade de deslocamento e a intensidade do ruído durante o processo de difusão. Vale ressaltar que o treinamento do modelo ocorre no espaço latente do VQGAN, o que contribui para a redução da sobrecarga computacional.

Os resultados quantitativos reforçam a superioridade do ResShift em relação a outros métodos, conforme evidenciado pelas métricas LPIPS, que significa *Learned Perceptual Image Patch Similarity*, e calcula a similaridade perceptual entre duas imagens, em diferentes conjuntos de dados, como o ImageNet-Test. Os resultados qualitativos também demonstram a capacidade do modelo em produzir imagens de alta resolução, com maior fidelidade e realismo em comparação a outros métodos.

O trabalho de Pinaya et al. (2023) utiliza uma estratégia em cascata, em que se resume a treinar dois modelos para a realização da tarefa de super-resolução: combina o *AutoencoderKL* e um modelo de difusão latente. O modelo de difusão latente é treinado no espaço latente do *autoencoder*. Além disso, foi utilizado um *framework* chamado "MONAI", proveniente do trabalho de Cardoso et al. (2022), que possui os dois modelos, as transformações que podemos utilizar nas imagens e as principais métricas para este tipo

de problema. Os experimentos realizados envolveram a avaliação destes modelos para diferentes tarefas, além da super-resolução. A metodologia incluiu a avaliação da capacidade de adaptação dos modelos a diferentes tipos de imagens médicas, como radiografias de tórax, mamografias e ressonâncias magnéticas. As métricas utilizadas para avaliar a qualidade dos modelos incluíram FID e MS-SSIM (*Multi-Scale Structural Similarity Index Measure*). Os resultados demonstraram um desempenho excepcional na geração de imagens de alta qualidade em diferentes cenários avaliados.

Portanto, esses trabalhos fornecem abordagens e modelos de difusão que podem ser adaptados e aplicados ao problema de super-resolução em imagens médicas, visando melhorar a qualidade das imagens e, assim, contribuir para diagnósticos mais precisos e eficazes na área médica. A seguir, será apresentada a Tabela 1 para destacar as principais diferenças e semelhanças entre esses quatro trabalhos e o apresentado aqui.

Trabalho	Modelo	Dados	Métricas de Avaliação
High-Resolution Image Synthesis with Latent Diffusion Models Rombach et al. (2021)	VAE + LDM	ImageNet	FID, *Precision*, *Recall*, SSIM, PSNR
SRDiff: Single Image Super-Resolution with Diffusion Probabilistic Models Li et al. (2021)	SRDiff	DIV2K e Flickr2K	SSIM, PSNR, LPIPS, LR-PSNR
InverseSR: 3D Brain MRI Super-Resolution Using a Latent Diffusion Model Wang et al. (2023)	InverseSR	IXI	SSIM, PSNR
ResShift: Efficient Diffusion Model for Image Super-resolution by Residual Shifting Yue, Wang e Loy (2023)	ResShift + VQGAN	ImageNet-Test	SSIM, PSNR, LPIPS,CLIPIQA
Generative AI for Medical Imaging: extending the MONAI Framework Pinaya et al. (2023)	LDM + AutoencoderKL	radiografias de tórax, mamografias e ressonâncias magnéticas	MS-SSIM, FID, PSNR
Este Trabalho	LDM + AutoencoderKL	Mamografias	SSIM, FID, PSNR

Tabela 1: Trabalhos Relacionados.

No geral, as diferenças entre este trabalho e os outros relacionados concentra-se,

principalmente, na utilização de estratégias e modelos ligeiramente distintos, seguindo o trabalho de Pinaya et al. (2023) que obtêm excelentes resultados. Por exemplo, enquanto este estudo emprega um *autoencoder* variacional com a perda KL, o trabalho de Rombach et al. (2021) utiliza um *autoencoder* variacional sem essa perda. Além disso, houve variação nas bases de dados utilizadas, como observado nos estudos de Yue, Wang e Loy (2023) e Li et al. (2021), que recorreram aos conjuntos *ImageNet* e *DIV2K*, respectivamente, para avaliação dos resultados.

O trabalho mais próximo em termos de abordagem é o de Pinaya et al. (2023), com a principal diferença sendo a escolha das bases de dados. Enquanto o estudo de Pinaya et al. (2023) emprega radiografias da cabeça para experimentos de super-resolução, este trabalho utiliza recortes de mamografias simuladas. Inspirado no trabalho de Pinaya et al. (2023), este estudo foi conduzido visando aplicar a tarefa de super-resolução, especificamente, a mamografias. Para tal, o *framework* MONAI foi adotado para a construção dos modelos, implementação das transformações necessárias nas imagens e cálculo das métricas pertinentes.

Ademais, o código foi reestruturado e adaptado para se adequar ao problema em questão, além de ter sido conteinerizado. Esse processo visa facilitar a configuração do experimento para diferentes conjuntos de dados, permitindo que a comunidade científica explore o potencial dos modelos de difusão latente em diversas imagens médicas.

3 METODOLOGIA

Neste capítulo, descreveremos em detalhes a metodologia utilizada para conduzir os experimentos desenvolvidos nesse trabalho, explicitando ferramentas, conjuntos de dados e abordagens desenvolvidas.

Este estudo busca desenvolver um modelo generativo para a tarefa de super-resolução. Com base nos trabalhos e conceitos apresentados no capítulo anterior, propomos uma solução que utiliza um modelo de difusão latente, seguindo a abordagem proposta por Pinaya et al. (2023). Para implementar a solução do modelo cascata, usaremos apenas a combinação responsável pela super-resolução, dito isto, o treinamento foi estruturado em duas fases.

A primeira fase refere-se ao treinamento de um modelo de *autoencoderkl*, cuja função é atuar como um conversor entre uma imagem e seu respectivo espaço latente, e vice-versa. Após a conclusão deste treinamento, foi realizada uma avaliação para investigar a qualidade da reconstrução da imagem.

Na segunda fase, foi feito o treinamento de um modelo de difusão latente, que aprende a gerar vetores no espaço latente a partir de ruído gaussiano. O codificador do *autoencoder*, cujos pesos foram congelados nesta segunda etapa, foi usado para converter as imagens de treinamento para o espaço latente. Estas imagens convertidas foram então transformadas em ruído pelo nosso modelo de difusão. Subsequentemente, utilizamos uma rede *DiffusionModelUNet* para o processo de redução de ruído no espaço latente. Em seguida, um modelo de difusão implícita, ou DDPM, foi empregado para gerar as amostras. Por fim, o conjunto de imagens gerado foi avaliado, a fim de verificar a capacidade do modelo de realizar super-resolução.

O código-fonte está disponibilizado no repositório do *GitHub* [1].

3.1 Ambiente e *framework*

Para o desenvolvimento deste trabalho foi utilizada a linguagem de programação Python na versão 3.8, com a imagem nvcr.io/nvidia/pytorch:23.04-py3 [2] do Docker que vem com Torch-cuda e com os seguintes pacotes:

- numpy==1.22.2
- scikit-learn==1.3.2
- pandas==2.1.1

[1] <https://github.com/DayvisonGomes/mamografIA_upscaler/tree/main>
[2] <https://docs.nvidia.com/deeplearning/frameworks/pytorch-release-notes/rel-23-04.html>

- monai[nibabel, matplotlib, pandas, mlflow, tqdm, pydicom]==1.2.0
- omegaconf==2.3.0
- protobuf==3.20.3
- pynvml==11.5.0
- wandb==0.16.6
- tifffile==2023.9.26
- monai-generative

O hardware utilizado para o treinamento e geração das imagens do conjunto de validação foi:

- Memória RAM: 256 GB, 3000 MHz.
- Processador: Intel (R) Core (TM) i9-10900X @ 3.70 GHz
- Placa de vídeo: Nvidia RTX A6000, 48 GB VRAM dedicada

A escolha do *framework* MONAI[3] proporcionou benefícios significativos, permitindo a aplicação de transformações nas imagens para pré-processamento, a utilização dos modelos necessários baseados em PyTorch e a incorporação das métricas pertinentes ao escopo deste trabalho para avaliação dos resultados. A necessidade de treinamento em cascata foi crucial para alcançar os objetivos propostos.

Nas seções subsequentes, será detalhada a base de dados e o procedimento adotado durante cada fase de treinamento, proporcionando uma compreensão abrangente do processo e dos elementos essenciais envolvidos.

3.2 Base de dados

Os dados de entrada foram obtidos através de uma simulação de equipamentos reais utilizados em exames de mamografia. Com isto, foram geradas duas bases de dados não rotuladas de 16 bits de precisão de regiões de interesse (do inglês, *Region of interest* ou ROI), a primeira composta por imagens de alta resolução, com dimensões 732x732 pixels, e a segunda composta por imagens de baixa resolução, com dimensões 512x512 pixels. Ambas são simulações com 70µm (70 micrômetros), para as de alta, e 100µm (100

[3] <https://github.com/Project-MONAI>

micrômetros), para as de baixa, que se referem aos valores de espessura de cada pixel. Além disso, os pixels estão em escala de cinza.

As duas bases consistem em 625 imagens de alta e 625 imagens de baixa resolução, em que as imagens são dividas em 3 classes: Classe A sendo composta por imagens menos densa, onde os pixels estão concentrados próximos ao zero. Classe B sendo composta por imagens consideradas no meio-termo entre densa e menos densa, os pixels estão concentrados entre o zero e o valor máximo 2^{14}. Classe C sendo composta por imagens mais densas, onde os pixels estão concentrados próximos do valor máximo 2^{14}.

As duas bases de dados foram divididas da mesma maneira, em conjuntos de treinamento e validação seguindo o padrão 90% e 10%, com 563 e 62 imagens, respectivamente, de maneira aleatória utilizando o método *train_test_split* com *random_state* de 42 da biblioteca *scikit-learn*, com a seguinte distribuição no treinamento, Classe A:196, Classe B:147 e Classe C:220, e na validação temos Classe A:13, Classe B:20, Classe C 29. A subdivisão em questão foi escolhida pela pouca quantidade de dados disponíveis. Na Figura 8 podemos ver alguns exemplos de imagens dos das duas bases. Vale destacar que as imagens possuem calcificações, que são pequenas "partes brancas", consistindo de pixels com um alto valor nessa região.

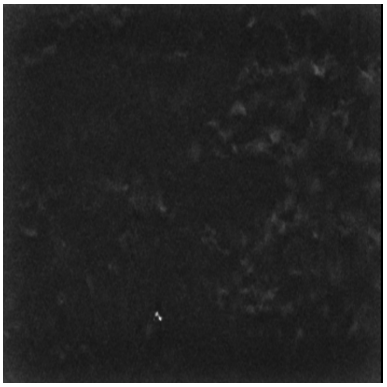


Imagem de alta resolução menos densa

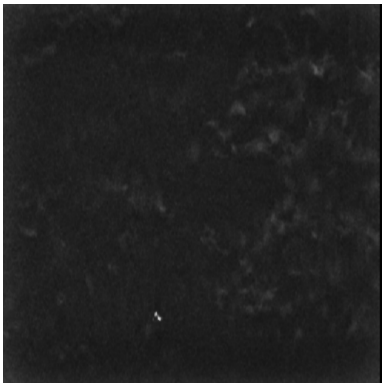


Imagem de baixa resolução menos densa

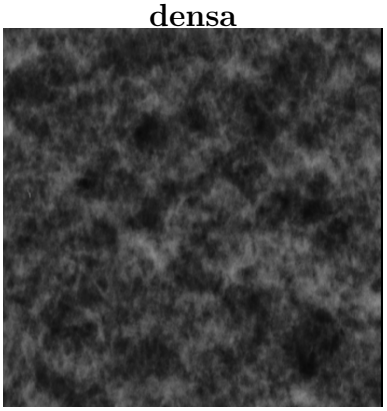


Imagem de alta resolução densa

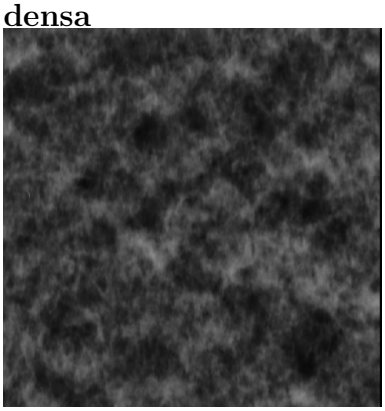


Imagem de baixa resolução densa

Figura 8: Exemplos de imagens de ambas as bases de dados

Com o MONAI conseguimos representar as imagens como tensores, e assim podemos utilizar transformações como *flip* horizontal ou vertical e rotações em 90 graus. Foi realizado uma normalização entre [0,1] utilizando o *ScaleIntensityd* do MONAI para cada imagem. Utilizamos o *DataLoader* do MONAI, que é uma extensão da classe *DataLoader* do *PyTorch*, dividindo os dados em *batche* de 1, sendo o par de imagem de alta e de baixa que nos permite um consumo eficiente de *VRAM*.

3.3 Modelo AutoencoderKL

Nessa subseção serão explicadas as fases de treinamento e de avaliação do modelo *autoencoderkl* utilizado nesse trabalho, para compressão e descompressão do espaço das imagens para o espaço latente. O modelo foi escolhido devido ao seu baixo custo computacional e alto desempenho em outros conjuntos de dados de imagens médicas, demonstrado por Pinaya et al. (2023). A Figura 9 e 10 mostram a arquitetura do modelo.

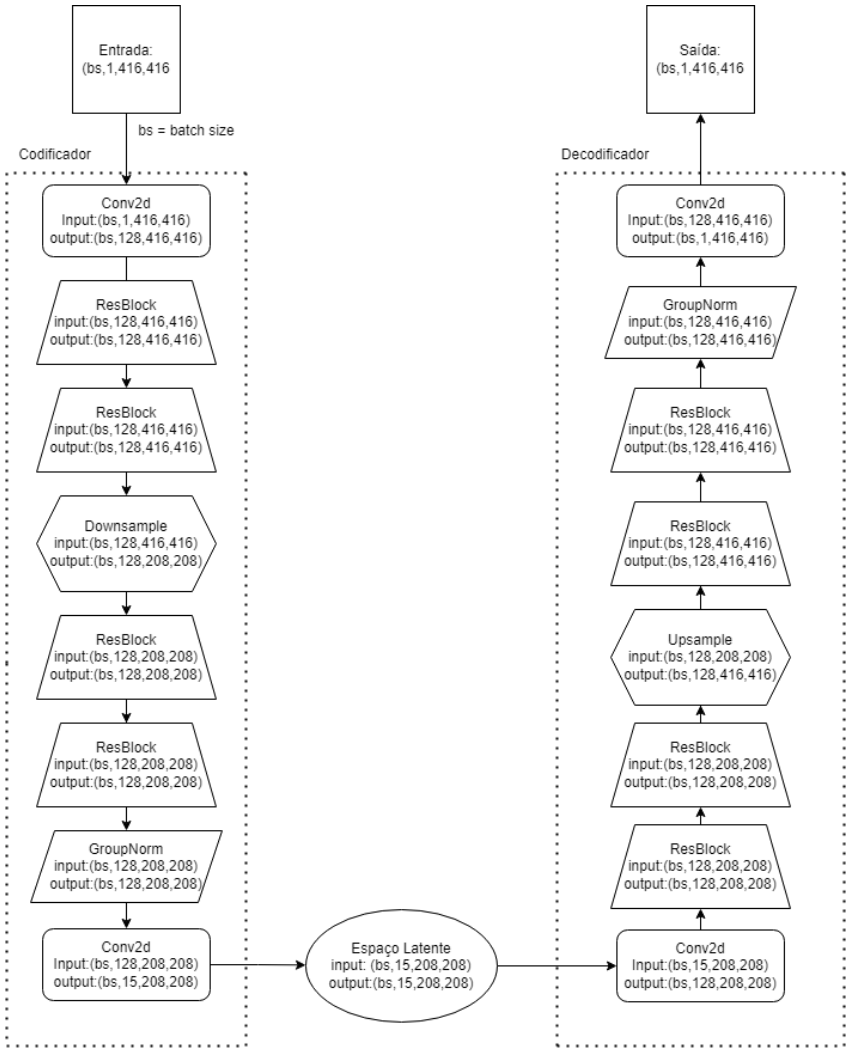


Figura 9: Arquitetura do Autoencoder. Fonte: Autor

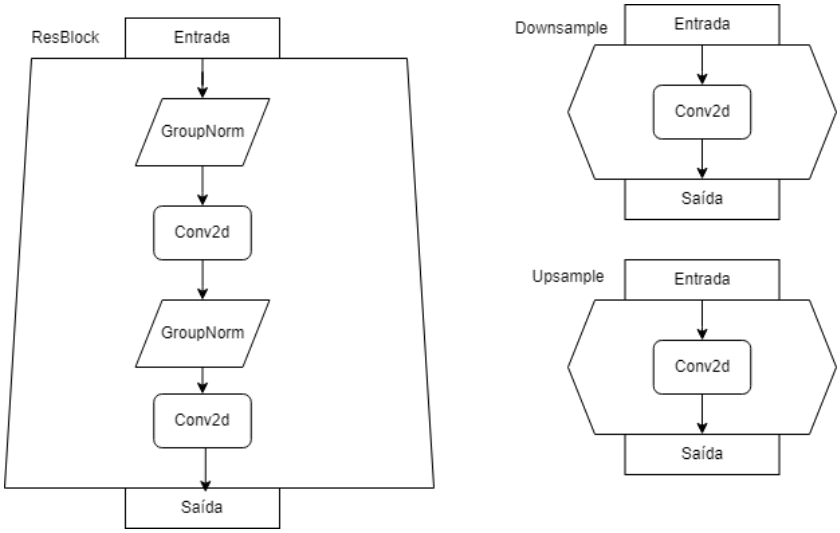
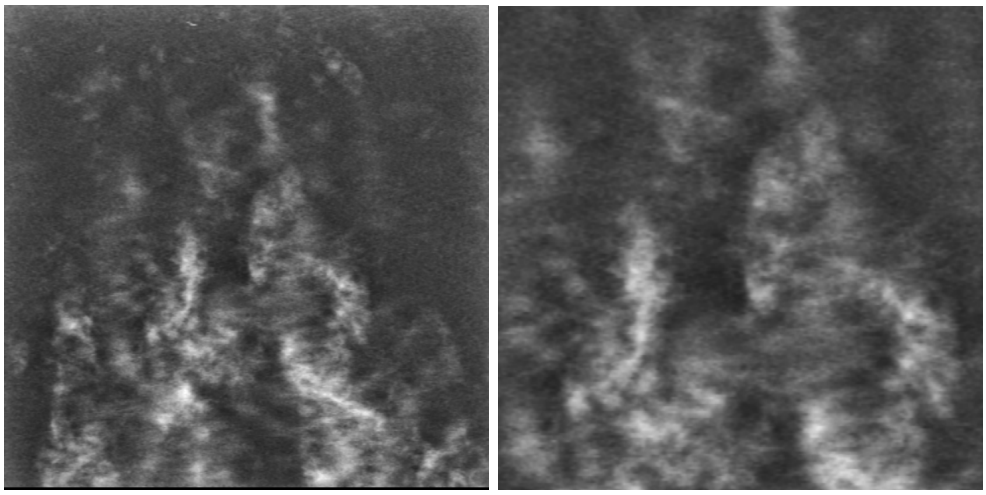


Figura 10: Blocos do Autoencoder. Fonte: Autor

3.3.1 Fase de treinamento

Antes do treinamento, é necessário realizar dois *center cropping*, sendo um para cada tipo de resolução das imagens, para as de alta, o *crop* foi de 416x416 pixels como na figura 11 e para as de baixa, 291x291.



(a) Imagem Original **(b) Crop 416x416**

Figura 11: Imagem Original vs Crop 416x416.

A escolha desses valores dependem do primeiro *cropping* e de como as imagens foram simuladas (70um e 100um). Originalmente, as imagens de alta possuem dimensões 732x732 e as de baixa 512x512, diminuindo a imagem de alta para 416x416 diminuiu-se a de baixa na mesma proporção, então o valor do segundo *cropping* seria de:

$$value = \frac{416 \cdot 0,07}{0,01} = \lfloor 291,2 \rfloor. \tag{7}$$

Seguido disto, é feita uma normalização das imagens de alta e baixa resolução, como dito em 3.2 e no fim, é feito um redimensionamento de 291x291 para 208x208 nas imagens de baixa, por causa do parâmetro utilizado no *Autoencoderkl*. Colocamos as duas imagens em tensores com *batch* 1 por causa da limitação de VRAM, com 48. A função de perda utilizada para treinar o modelo foi a *l1 loss* ou o Erro Absoluto Médio (do inglês, Mean Absolut Error (MAE)). Essa escolha foi feita de acordo com o trabalho de Pinaya et al. (2023). Posteriormente, incorporamos os dados ao *DataLoader* do MONAI.

Ao instanciar o *autoencoder*, foi parametrizado da seguinte maneira

- $spatial\_dims = 2$, nossa imagem é bidimensional.
- $in\_channels = 1$, apenas um canal de cores.
- $out\_channels = 1$, a saída em escala de cinza.
- $num\_channels = (128, 128)$, que define a quantidade de blocos *Downsample* e *Upsample*, no caso temos apenas um de cada.
- $latent\_channels = 15$, isso significa que temos 15 canais latentes, os quais representam de forma compacta e informativa os dados de entrada.
- $num\_res\_blocks = 2$, dois *ResBlocks* antes e depois do *Downsample* e *Upsample*.
- $attention\_levels = (\text{False, False})$, define tanto a quantidade como se haverá ou não *AttentionBlocks*.

Para configurar o discriminador, adotamos uma arquitetura composta por três camadas, cada uma com 64 canais e dimensões espaciais bidimensionais. A fim de guiar o treinamento da rede, optamos por utilizar duas funções de perda distintas: a *PerceptualLoss*, inspirada na *AlexNet*, com um peso de 0,002, e a *PatchAdversarialLoss*, com peso 0,005, aplicadas tanto ao gerador quanto ao discriminador.Estes pesos têm como finalidade ponderar as contribuições dos diferentes componentes na obtenção das perdas. Para o gerador, a função de perda é composta pela:

$$loss\_generator = l1\_loss + (kl\_weight \cdot kl\_loss) + (perceptual\_weight \cdot p\_loss), \quad (8)$$

e a do discriminador:

$$discriminator\_loss = (loss\_d\_fake + loss\_d\_real) \cdot 0.5, \quad (9)$$

$$loss\_d = adv\_weight \cdot discriminator\_loss, \quad (10)$$

Em que temos a *l1_loss*, a regularização kl e a *PerceptualLoss* no gerador na equação 8, enquanto para o discriminador, a função de perda é composta pelas perdas *loss_d_fake* e *loss_d_real* nas equações 9 e 10, calculadas ao submeter respectivamente a imagem gerada e a imagem real à *PatchAdversarialLoss*, conforme descrito em Pinaya et al. (2023).

É importante notar que a contribuição da perda adversarial *loss_adv_g*:

$$loss\_adv\_g = adv\_weight \cdot adversarial\_loss, \tag{11}$$

A perda total do gerador da equação 11 é adicionada apenas após um período inicial de aquecimento (do inglês, warmup) de 10 épocas. Esse aquecimento é necessário para estabilizar o treinamento da rede. Esta perda adversarial *loss_adv_g* é calculada ao passar a imagem gerada pela *PatchAdversarialLoss*, instruindo-a a ser classificada como uma imagem real como em Pinaya et al. (2023).

Utilizamos otimizadores Adam para o *autoencoder* e discriminador, ajustando as taxas de aprendizado para 0,00005 e 0,0001, respectivamente. O treinamento estendeu-se por 100 épocas, com validação a cada 10 épocas. Os parâmetros do *GradScaler* foram empregados nos otimizadores do *autoencoder* e do discriminador. Implementamos um aquecimento de 10 épocas para o *autoencoder*, antes de introduzir a *adversarial loss*.

Ao longo do treinamento, monitoramos as perdas de reconstrução, generativas e discriminativas. Além disso, utilizamos uma classe chamada *Early Stopping*, para uma eventual parada de treinamento, com base na função de perda da validação. Caso exista uma melhora, salvamos um *best model*.

3.3.2 Fase de Avaliação

Na fase de avaliação, três métricas foram escolhidas para avaliar o desempenho do modelo *autoencoderkl*: SSIM, FID e MAE. As imagens originais do conjunto de validação foram carregadas utilizando a classe *DataLoader* do MONAI, em *batches* de 1 imagem. Cada *batch* foi passado como entrada para o modelo, a fim de adquirir a imagem reconstruída, e então passamos as duas para o cálculo do MAE e SSIM, No final, foram calculadas as médias do MAE e do SSIM. Por fim, calculamos o FID utilizando a rede *Resnet50* (MASCARENHAS; AGARWAL, 2021) como feito em Pinaya et al. (2023) ao invés da rede *Inception V3* (SZEGEDY et al., 2015) para a obtenção das *features* necessárias, então utilizamos o conjunto de dados original e o conjunto de dados reconstruídos para a obtenção da métrica FID. Todos os resultados foram salvos em um arquivo de texto.

3.4 Modelo de difusão latente

Nessa subseção serão explicadas as fases de treinamento e de avaliação do modelo de difusão latente utilizado nesse trabalho, para geração de novas representações latentes, baseadas no conjunto de dados originais.

Para a construção do modelo, foi utilizada uma *pipeline* de difusão, ilustrada na Figura 7, em que o conjunto de dados x passa pelo codificador do *Autoencoderkl*, obtendo assim o espaço latente. O ruído gaussiano é adicionado neste espaço no processo de difusão direta. Em seguida, para o caso de super-resolução, concatenamos a imagem de baixa resolução com ruído ao espaço latente (PINAYA et al., 2023). O resultado é passado para o treinamento do estimador de ruído *UNet*, responsável pelo processo de redução de ruído.

No caso de amostragem ou inferência, inicialmente é gerado um conjunto gaussiano que representa o espaço latente e então concatenamos com a imagem de baixa resolução também com ruído, que são passados para o estimador de ruído, por 1000 *steps*, que retorna um conjunto $\tilde{x}$ no espaço latente. Subsequentemente, $\tilde{x}$ passa pelo decodificador do *Autoencoderkl*, que transforma as representações nas imagens.

Ao instanciar o *DiffusionModelUnet*, este foi parametrizado da seguinte maneira:

- *spatial_dims* = 2, nossa imagem é bidimensional.
- *in_channels* = 16, 15 do espaço latente mais a imagem de baixa resolução.
- *out_channels* = 15, quantidade de camadas latentes.
- *num_channels* = (256, 256), que define a quantidade de blocos *DownBlock* e *UpBlock*, no caso temos apenas um de cada.
- *latent_channels* = 15, temos 15 canais latentes.
- *num_res_blocks* = 2, dois *ResNetBlocks*.
- *attention_levels* = (False, True), define tanto a quantidade como se haverá ou não *AttentionBlocks*.
- *num_head_channels* = (0,64), número de canais em cada *attention head*.

3.4.1 Fase de treinamento

Inicialmente, as imagens originais foram carregadas utilizando o *DataLoader* do MONAI, sendo agrupadas em *batches* de uma imagem (de alta e de baixa no mesmo *batch*).

A função de perda utilizada no treinamento do modelo *UNet*, estimador de ruído, foi o MSE e a escolha dessa função foi feita de acordo com a abordagem de Pinaya et al. (2023).

O treinamento é conduzido ao longo de 120 épocas e com validação a cada 10 épocas, salvando o *best model*, a cada melhora do MSE na validação, utilizando a classe *Early Stopping*. A escolha da quantidade de épocas se dá por causa da lenta melhora do modelo, de acordo com a MSE.

Durante o treinamento, o modelo *UNet* é ajustado, enquanto utilizamos o *Autoencoderkl*, sem o uso de gradientes para gerar a representação latente. O otimizador Adam é aplicado ao modelo *UNet*, com uma taxa de aprendizado de 0,00005, e um *GradScaler* é utilizado para escalonamento durante o treinamento.

A cada iteração de treinamento, a representação latente é obtida a partir da imagem de alta resolução, multiplicada pelo fator de escala, que é previamente calculado como o inverso do desvio padrão dos valores da representação latente nos dados de treinamento de acordo com Rombach et al. (2021). Um ruído gaussiano é aplicado, tanto à representação latente, quanto às imagens de baixa resolução. A entrada do modelo *UNet* é formada pela concatenação da representação latente ruidosa e da imagem de baixa resolução ruidosa.

3.4.2 Fase de Avaliação

Após o final do treinamento, realizamos a super-resolução das imagens de baixa resolução e utilizamos as métricas propostas neste estudo: FID, SSIM e PSNR. Como treinamos com imagens normalizadas, estamos comparando dois conjuntos, originais e gerados na mesma escala de pixels, dito isto as métricas são calculada tal como no *Autoencoderkl*.

4 APRESENTAÇÃO E ANÁLISE DOS RESULTADOS

O objetivo dessa seção é apresentar e discutir os resultados dos experimentos desenvolvidos nesse trabalho, que foram descritos no capítulo anterior. Esta seção foi dividida em duas partes, sendo a primeira relacionada aos resultados do modelo *Autoencoderkl* e a segunda, ao modelo de difusão latente.

4.1 Modelo *AutoencoderKL*

Essa seção foi dividida em duas partes. Na primeira, observaremos os valores das métricas neste modelo, utilizando os dados da validação. Na segunda parte, serão expostas amostras de reconstrução.

4.1.1 Análise de métricas

Como dito na fase de avaliação do modelo *Autoencoderkl*, foram calculadas as métricas após o treinamento em um conjunto de validação que consistia de 62 imagens.

A Tabela 2 mostra os resultados das métricas.

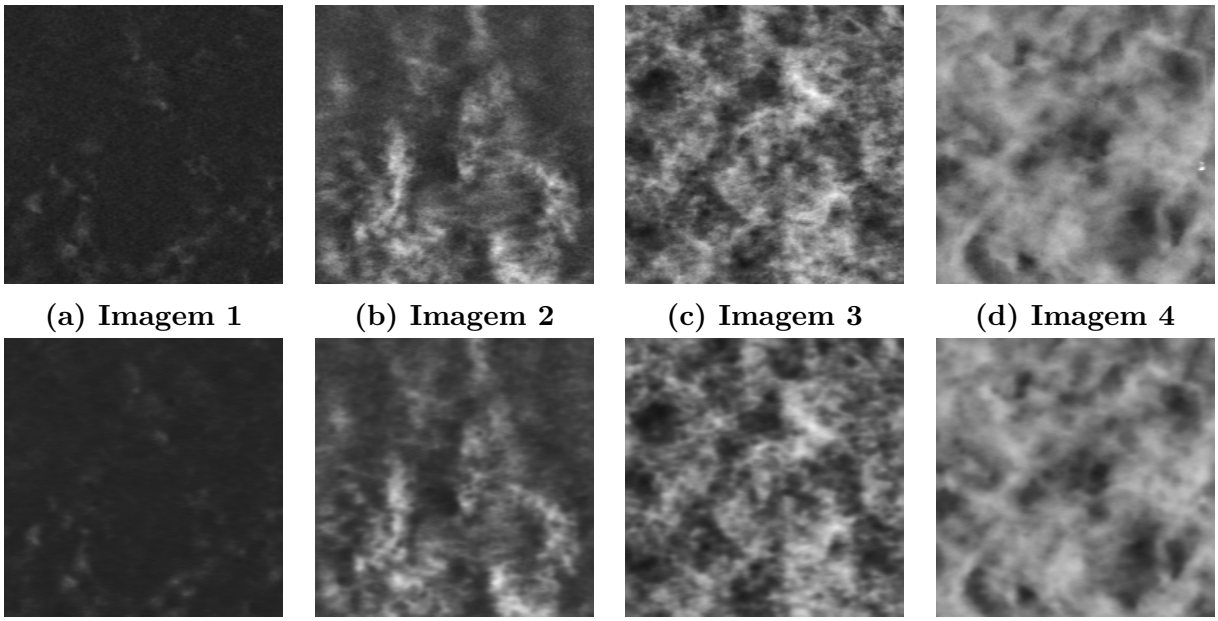
# Camadas	FID	SSIM	PSNR
15	0,00005	0,87	33

Tabela 2: Resultados das métricas.

Em comparação com Pinaya et al. (2023), que obteve FID=0,0009, conseguimos reproduzir uma distribuição próxima da original. Ambos SSIM e PSNR, foram pouco maiores em, comparação com Rombach et al. (2021), que ficam na faixa de 0,7 e 0,8 para o SSIM e 26 e 27 para o PSNR. Isso pode ter ocorrido devido à natureza dos nossos dados e por estarmos realizando a reconstrução das imagens, indicando sua boa qualidade.

4.1.2 Amostras de reconstrução

A Figura 12 mostra quatro exemplos com o *AutoencoderKL* com 15 camadas na representação latente. Na parte superior, são mostradas imagens de entrada e na parte inferior, imagens reconstruídas.



(a) Imagem 1 **(b) Imagem 2** **(c) Imagem 3** **(d) Imagem 4**

(e) Reconstrução 1 **(f) Reconstrução 2** **(g) Reconstrução 3** **(h) Reconstrução 4**

Figura 12: Amostra do *autoencoderKL* com 15 camadas no vetor latente.

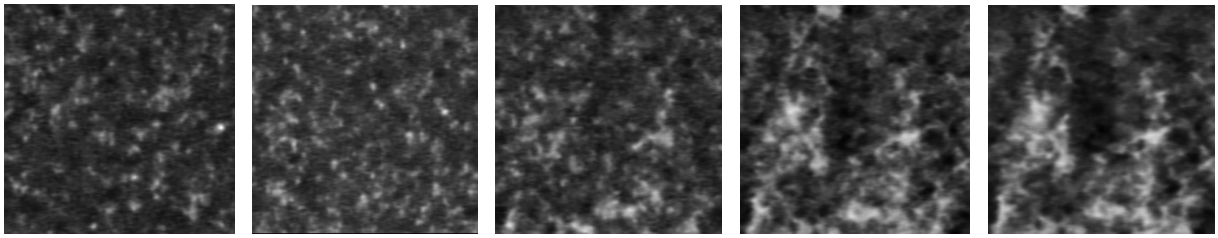
Percebemos que as imagens geradas estão levemente borradas, devido, provavelmente, a normalização das imagens, que Pinaya et al. (2023) realiza também em seus experimentos. Isso acontece por causa da presença de *outliers*, que seriam as calcificações, onde os valores dos pixels presentes nestas regiões são discrepantes em relação ao resto da imagem.

4.2 Modelo de difusão latente

Nessa seção é feita a análise do modelo de difusão latente. Para isso, mostraremos o processo para a geração da imagem final e a análise de métricas e da qualidade visual das imagens originais e geradas, escolhidas do conjunto de validação.

A Figura **??** mostra o processo de diminuição de ruído do DDPM e ao passar no decodificador do *Autoencoderkl*, ou seja, a imagem final é o *upscaling* da imagem de baixa resolução. Esta diminuição funciona como o esperado, pois no final da figura, temos o nosso resultado, que se aproxima da imagem original. Podemos ver outro exemplo no final do código feito pelos criadores do MONAI.[4]

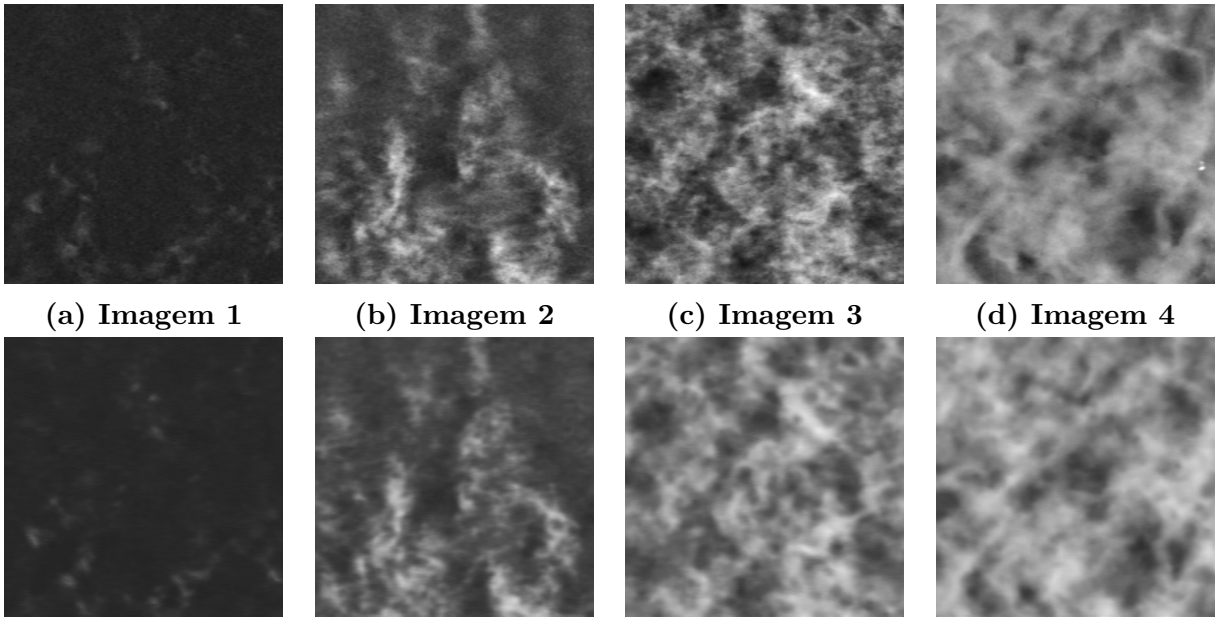
[4]<https://github.com/Project-MONAI/GenerativeModels/blob/main/tutorials/generative/2d_ldm/2d_ldm_tutorial.ipynb>



(a) 0 *steps* **(b) 400 *steps*** **(c) 600 *steps*** **(d) 800 *steps*** **(e) 1000 *steps***

Figura 13: Processo de diminuição de ruído.

Na Figura 14 temos as mesmas 4 imagens vistas na Figura 12, que são correspondentes às geradas pelo modelo de difusão.



(a) Imagem 1 **(b) Imagem 2** **(c) Imagem 3** **(d) Imagem 4**

(e) *Upscaling* 1 **(f) *Upscaling* 2** **(g) *Upscaling* 3** **(h) *Upscaling* 4**

Figura 14: Amostra do modelo de difusão.

Ao olho humano, percebemos um nível de borramento semelhante ao que tivemos no *Autoencoderkl*. É seguro presumir que se conseguirmos treinar um *Autoencoderkl*, normalizando de maneira efetiva considerando os *outliers* citados, as imagens geradas pelo modelo de difusão seriam melhores. Podemos observar isto na Tabela 3, em que as métricas se aproximam das vistas na Tabela 2.

# Camadas	FID	SSIM	PSNR
15	0,00006	0,80	26

Tabela 3: Resultados das métricas no modelo de difusão.

Para o caso de super-resolução, podemos comparar nossos valores com o trabalho de Pinaya et al. (2023), que obteve um FID=0,0024, maior do que observamos (0,00006), indicando um bom resultado. O PSNR=29,8042 $\pm$ 0.4173 foi maior que o nosso (26),

podendo indicar o surgimento do borramento. O valor do SSIM entre 0,7 e 0,8 visto, em Rombach et al. (2021), indica que a similaridade de estruturas estão boas, como vemos na Figura 14.

5 CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho propõe a criação de um modelo generativo de super-resolução em regiões de interesse de imagens de mamas. O objetivo é a geração de conjuntos de imagens que sejam próximos do conjunto original.

Este objetivo foi alcançado parcialmente, com resultados satisfatórios nas métricas propostas: FID=0,00006 , SSIM=0,80 e PSNR=26, que se equiparam aos resultados obtidos nos trabalhos de Rombach et al. (2021) e Pinaya et al. (2023), que obtiveram os valores: FID=0,0024, PSNR=29,8042 ± 0,4173, SSIM entre 0,7 e 0,8, em que são utilizados outros dados, com outra natureza. Os dados testados no atual trabalho são heterogêneos e, com isto, o *AutoencoderKL* acabou performando como o esperado, com imagens levemente borradas. Por mais que a regularização KL obrigue o modelo a melhorar a distribuição aprendida, a quantidade de dados e a sua natureza simulada influenciaram os resultados obtidos, além de uma normalização efetiva.

Apesar do resultado esperado do modelo generativo, ainda é possível aprofundar o estudo e melhorar a eficácia do modelo para esse conjunto de dados, com a troca de *losses* no treinamento do *AutoencoderKL* e outros tipos de normalização, e a utilização de outros modelos para a geração da representação latente, que podem fazer surgir melhorias significativas. Adicionalmente, este trabalho também contribui para a comunidade com uma nova *pipeline* de treinamento e avaliação para a tarefa de super-resolução, permitindo o teste de diferentes tamanhos para a representação latente, além de ajuste dos outros parâmetros dos dois modelos e uma versão conteinerizada do projeto, que permite o treinamento de forma escalável para diferentes sistemas operacionais e para diferentes conjuntos de dados.

5.1 Trabalhos Futuros

Este trabalho pode ser usado como ponto inicial para diversos projetos e estudos futuros, dentre eles, destacam-se:

- ***Autoencoders* determinísticos** para o aprendizado da representação latente, como por exemplo, o VQGAN visto em Yu et al. (2022), que utiliza *Vision Transformer*.
- **Minimização do *Blur***, proposto por Bredell et al. (2023), que atribui um peso para o termo de reconstrução, afim de diminuir o impacto do borramento.
- ***Data augmentation***, atualmente, os *flips* e rotações são aplicados apenas durante o carregamento dos dados, dentro do *DataLoader*, modificando as imagens em tempo

de execução. Uma possível alternativa seria realizar essas transformações de forma definitiva, fora do processo de carregamento. Isso permitiria a geração efetiva de novas imagens e a criação dos arquivos correspondentes. Essa abordagem envolveria a criação de novos conjuntos de dados com imagens transformadas, antes do início do treinamento, proporcionando ao modelo uma maior variedade de exemplos para aprender.

REFERÊNCIAS

ACADEMY, D. S. **Deep Learning Book. Cap 60**. 2022. <https://www.deeplearningbook.com.br/variational-autoencoders-vaes-definicao-reducao-de-dimensionalidade-espaco-latente-e-regularizacac >.

BALLARD, D. H. **Computer Vision**. 1982.

BREDELL, G.; FLOURIS, K.; CHAITANYA, K.; ERDIL, E.; KONUKOGLU, E. **Explicitly Minimizing the Blur Error of Variational Autoencoders**. 2023.

CARDOSO, M. J.; LI, W.; BROWN, R.; MA, N.; KERFOOT, E.; WANG, Y.; MURREY, B.; MYRONENKO, A.; ZHAO, C.; YANG, D.; NATH, V.; HE, Y.; XU, Z.; HATAMIZADEH, A.; MYRONENKO, A.; ZHU, W.; LIU, Y.; ZHENG, M.; TANG, Y.; YANG, I.; ZEPHYR, M.; HASHEMIAN, B.; ALLE, S.; DARESTANI, M. Z.; BUDD, C.; MODAT, M.; VERCAUTEREN, T.; WANG, G.; LI, Y.; HU, Y.; FU, Y.; GORMAN, B.; JOHNSON, H.; GENEREAUX, B.; ERDAL, B. S.; GUPTA, V.; DIAZ-PINTO, A.; DOURSON, A.; MAIER-HEIN, L.; JAEGER, P. F.; BAUMGARTNER, M.; KALPATHY-CRAMER, J.; FLORES, M.; KIRBY, J.; COOPER, L. A. D.; ROTH, H. R.; XU, D.; BERICAT, D.; FLOCA, R.; ZHOU, S. K.; SHUAIB, H.; FARAHANI, K.; MAIER-HEIN, K. H.; AYLWARD, S.; DOGRA, P.; OURSELIN, S.; FENG, A. **MONAI: An open-source framework for deep learning in healthcare**. 2022.

CROITORU, F.-A.; HONDRU, V.; IONESCU, R. T.; SHAH, M. Diffusion models in vision: A survey. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, Institute of Electrical and Electronics Engineers (IEEE), v. 45, n. 9, p. 10850–10869, set. 2023. ISSN 1939-3539. Disponível em: <http://dx.doi.org/10.1109/TPAMI.2023.3261988>.

GOODFELLOW, I. J.; POUGET-ABADIE, J.; MIRZA, M.; XU, B.; WARDE-FARLEY, D.; OZAIR, S.; COURVILLE, A.; BENGIO, Y. **Generative Adversarial Networks**. 2014.

GREENEMEIRR, L. **Visionary Research: Teaching Computers to See Like a Human**. 2008.

HAYKIN, S. **Redes Neurais: Princípios e Prática**. 2000.

HEUSEL, M.; RAMSAUER, H.; UNTERTHINER, T.; NESSLER, B.; HOCHREITER, S. **GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium**. 2018.

HO, J.; JAIN, A.; ABBEEL, P. **Denoising Diffusion Probabilistic Models**. 2020.

HORé, A.; ZIOU, D. Image quality metrics: Psnr vs. ssim. In: . [S.l.: s.n.], 2010. p. 2366–2369.

KANTOROVICH, L. V. Mathematical methods of organizing and planning production. **Image Processing, IEEE Transactions on**, v. 6, p. 366 – 422, 07 1960.

KELEş, O.; YıLMAZ, M. A.; TEKALP, A. M.; KORKMAZ, C.; DOGAN, Z. **On the Computation of PSNR for a Set of Images or Video**. 2021.

KINGMA, D. P.; WELLING, M. **Auto-Encoding Variational Bayes**. 2022.

LI, H.; YANG, Y.; CHANG, M.; FENG, H.; XU, Z.; LI, Q.; CHEN, Y. **SRDiff: Single Image Super-Resolution with Diffusion Probabilistic Models**. 2021.

MASCARENHAS, S.; AGARWAL, M. A comparison between vgg16, vgg19 and resnet50 architecture frameworks for image classification. In: **2021 International Conference on Disruptive Technologies for Multi-Disciplinary Research and Applications (CENTCON)**. [S.l.: s.n.], 2021. v. 1, p. 96–99.

OU, T. **Variational AutoEncoder, and a bit KL Divergence, with PyTorch**. 2022. <https://medium.com/@outerrencedl/variational-autoencoder-and-a-bit-kl-divergence-with-pytorch-ce04fd55d0d7>.

PAIN, J.-C. **Kullback-Leibler divergence for the Fréchet extreme-value distribution**. 2023.

PINAYA, W. H. L.; GRAHAM, M. S.; KERFOOT, E.; TUDOSIU, P.-D.; DAFFLON, J.; FERNANDEZ, V.; SANCHEZ, P.; WOLLEB, J.; COSTA, P. F. da; PATEL, A.; CHUNG, H.; ZHAO, C.; PENG, W.; LIU, Z.; MEI, X.; LUCENA, O.; YE, J. C.; TSAFTARIS, S. A.; DOGRA, P.; FENG, A.; MODAT, M.; NACHEV, P.; OURSELIN, S.; CARDOSO, M. J. **Generative AI for Medical Imaging: extending the MONAI Framework**. 2023.

ROMBACH, R.; BLATTMANN, A.; LORENZ, D.; ESSER, P.; OMMER, B. **High-Resolution Image Synthesis with Latent Diffusion Models**. 2021.

SAJID, H. **Modelos de difusão em IA – tudo o que você precisa saber**. 2023. <https://www.unite.ai/pt/modelos-de-difusão-em-ai-tudo-o-que-você-precisa-saber/>.

SALIMANS, T.; KINGMA, D. P.; WELLING, M. **Markov Chain Monte Carlo and Variational Inference: Bridging the Gap**. 2015.

SZEGEDY, C.; VANHOUCKE, V.; IOFFE, S.; SHLENS, J.; WOJNA, Z. **Rethinking the Inception Architecture for Computer Vision**. 2015.

TAFNER, M. A. **O Que São as Redes Neurais Artificiais**. 1998. <https://cerebromente.org.br/n05/tecnologia/rna.htm>.

TEIXEIRA, J. P. V. **Síntese de Imagens de Mamas Sintéticas usando um Modelo de Difusão Latente**. 2023.

WANG, J.; LEVMAN, J.; PINAYA, W. H. L.; TUDOSIU, P.-D.; CARDOSO, M. J.; MARINESCU, R. **InverseSR: 3D Brain MRI Super-Resolution Using a Latent Diffusion Model**. 2023.

WANG, W. **An effective introduction to the Markov Chain Monte Carlo method**. 2022.

WANG, Z.; BOVIK, A.; SHEIKH, H.; SIMONCELLI, E. Image quality assessment: From error visibility to structural similarity. **Image Processing, IEEE Transactions on**, v. 13, p. 600 – 612, 05 2004.

WANG, Z.; SIMONCELLI, E.; BOVIK, A. Multiscale structural similarity for image quality assessment. In: **The Thrity-Seventh Asilomar Conference on Signals, Systems Computers, 2003**. [S.l.: s.n.], 2003. v. 2, p. 1398–1402 Vol.2.

YU, J.; LI, X.; KOH, J. Y.; ZHANG, H.; PANG, R.; QIN, J.; KU, A.; XU, Y.; BALDRIDGE, J.; WU, Y. **Vector-quantized Image Modeling with Improved VQGAN**. 2022.

YUE, Z.; WANG, J.; LOY, C. C. **ResShift: Efficient Diffusion Model for Image Super-resolution by Residual Shifting**. 2023.