
UNIVERSIDADE ESTADUAL DE MARINGÁ
DEPARTAMENTO DE FÍSICA

LUIZ FERNANDO MENEZES PANTALEÃO

ESTUDO DE MODELOS PROBABILÍSTICOS
DE DIFUSÃO COM IA APLICADOS À
GERAÇÃO DE IMAGENS DE CRISTAIS
LÍQUIDOS

Maringá, Dia 17 de Julho de 2025.

UNIVERSIDADE ESTADUAL DE MARINGÁ
DEPARTAMENTO DE FÍSICA

LUIZ FERNANDO MENEZES PANTALEÃO

ESTUDO DE MODELOS PROBABILÍSTICOS
DE DIFUSÃO COM IA APLICADOS À
GERAÇÃO DE IMAGENS DE CRISTAIS
LÍQUIDOS

*Dissertação de mestrado apresentada ao Programa
de Pós-Graduação em Física da Universidade
Estadual de Maringá.*

Orientador: Prof. Dr. Rafael Soares Zola

Maringá, Dia 17 de Julho de 2025.

Dados Internacionais de Catalogação-na-Publicação (CIP)
(Biblioteca Central - UEM, Maringá - PR, Brasil)

P197e

Pantaleão, Luiz Fernando Menezes

Estudo de modelos probabilísticos de difusão com IA aplicados à geração de imagens de cristais líquidos / Luiz Fernando Menezes Pantaleão. -- Maringá, PR, 2025.
62 f. : il. color., figs.

Orientador: Prof. Dr. Rafael Soares Zola.

Dissertação (mestrado) - Universidade Estadual de Maringá, Centro de Ciências Exatas, Departamento de Física, Programa de Pós-Graduação em Física, 2025.

1. Inteligência artificial. 2. Cristais líquidos - Textura. 3. Modelos de difusão. I. Zola, Rafael Soares, orient. II. Universidade Estadual de Maringá. Centro de Ciências Exatas. Departamento de Física. Programa de Pós-Graduação em Física. III. Título.

CDD 23.ed. 530.429

LUIZ FERNANDO MENEZES PANTALEÃO

**ESTUDO DE MODELOS PROBABILÍSTICOS DE DIFUSÃO COM IA APLICADOS
À GERAÇÃO DE IMAGENS DE CRISTAIS LÍQUIDOS**

Dissertação apresentada à Universidade Estadual de Maringá, como requisito parcial para a obtenção do título de mestre.

Aprovado em: Maringá, 30 de julho de 2025.

BANCA EXAMINADORA

Prof. Dr. Rafael Soares Zola – UTFPR/Apucarana e PFI/UEM
Orientador – Presidente

Prof^a. Dr^a. Roberta Rarumy Ribeiro de Almeida – UTFPR/Apucarana
Membro Externo

Prof. Dr. Rodolfo Teixeira de Souza – UTFPR/Apucarana e PFI/UEM
Membro Interno (Docente PFI/UEM)

Agradecimentos

Este trabalho é uma composição coletiva. Homenageio, aqui, a Física, que me acompanhou em tantos momentos de solidude e aumentou o poder de minha lucidez quanto às coisas do mundo material, ou seja, do tudo. Agradeço a todos meus companheiros e companheiras que alimentaram minha consciência com palavras positivas, de tal modo que me ajudaram a contrapor todo descrédito que eu dava à minha vontade de exaltar o simples. Este trabalho será simples, e por isso altruísta, porque os motivos de minhas incursões conceptuais em busca de sentido, como num feitiço de ourobouros, me levaram de volta à minha infância, à origem do pensamento sofisticado. Agradeço Rodrigo Azzolini, um pragmático de carteirinha, especialista em desbravar minhas lógicas junto comigo. Agradeço minhas amigas e amigos, e minha *eternal wife*, Flaviana Bersan, que sempre esteve comigo quando precisei. A minha mãe e meu pai: sem vocês eu não conseguiria. Também agradeço a meu orientador por acompanhar meu conturbado processo criativo. Por fim, agradeço às poesias de Nietzsche, fundamento da real crítica à ciência.

*“Essa língua não é minha.
Qualquer um percebe.
Enquanto o sentido caminha,
a palavra permanece.”*

PAULO LEMINSKI

Resumo

Modelos de difusão tornaram-se recentemente muito populares e altamente eficazes em tarefas generativas. Baseados na termodinâmica fora do equilíbrio, os modelos de difusão são modelos profundos de variáveis latentes treinados para aprender um processo reverso sobre as latentes dos dados. Os dados são transformados em latentes por meio do processo direto, definido por uma destruição lenta e bem comportada dos dados em ruído Gaussiano isotrópico, com média zero e covariância Identidade. O processo reverso segue a mesma forma lenta e bem comportada do processo direto, com média e covariância sendo parâmetros aprendíveis. No caso da difusão com remoção de ruído, a covariância é fixada como um parâmetro não aprendível, sendo apenas a média aprendida. Usando essa estrutura, desenvolvemos nossa própria rede neural utilizando PyTorch, e a treinamos condicional e incondicionalmente em imagens de diversos cristais líquidos. Nosso objetivo com este estudo é entender melhor os modelos de difusão ao mesmo tempo em que propomos um modelo de baixo custo para geração de imagens de cristais líquidos com objetivos didáticos.

Palavras-chave: Aprendizado de Máquinas. Aprendizado Profundo. Cristais Líquidos. Ciência de Dados. Modelos Probabilísticos de Difusão por Remoção de Ruído.

Abstract

Diffusion models have recently become very popular and highly performative on generative tasks. Based on non-equilibrium thermodynamics, diffusion models are deep latent variable models trained to learn a reverse process over latents of data. Data is turned into latents over the forward process, defined by a slow and well behaved destruction of data into isotropic Gaussian noise, with mean zero and covariance Identity. The reverse process follows the same slow and well behaved form of the forward process, with mean and covariance being learnable parameters. In the case of denoising diffusion probabilistic models, the covariance is set to a fixed non-learnable parameter and only the mean is learnable. Using this framework, we developed our own neural network with PyTorch and trained it conditionally and unconditionally on various crystal liquid's images. Our goal with this study is to better understand diffusion models while proposing a low cost model for liquid crystal's images generation.

Keywords: Machine Learning. Deep Learning. Images. Data Science. Liquid Crystals. Diffusion Models.

Introdução	10
1 Modelos de Difusão por Remoção de Ruído	13
1.1 Introdução	13
1.2 DDPM	15
1.3 Processo Direto	15
1.4 Processo Reverso	17
1.5 Treino	18
1.6 Conexão com Modelos Baseados em Energia	20
1.7 Avaliação do Modelo	20
2 Cristais Líquidos	22
2.1 Introdução	22
2.2 Nemáticos	24
2.3 Colestéricos	25
2.4 Esméticos	28
2.4.1 Esmético A	28
2.4.2 Esmético C	29
2.4.3 Esmético I	29
2.4.4 Esmético F	29
2.4.5 Esmético G	30
3 Testes para Escolha da Arquitetura da Rede Neural	32
3.1 Introdução	32
3.2 Teste de Regressão	33

3.3	Teste de Classificação Categórica	36
3.4	Escolha da Arquitetura do Modelo de Difusão	39
3.5	Arquitetura do Modelo de Difusão	41
4	Resultados	43
4.1	Introdução	43
4.2	Treino Condicional	44
4.2.1	Nemático	45
4.2.2	SmA	47
4.2.3	SmC	49
4.2.4	SmF	49
4.2.5	SmG	50
4.2.6	SmI	51
4.3	Treino Condicionado por Campo Elétrico	52
4.4	Treino Incondicional	53
4.5	Escalabilidade	54
	Conclusão	55
	A Resultados de Arquiteturas Anteriores	56
	Referências bibliográficas	62

A inteligência artificial (IA) remonta à década de 1950, quando os primeiros pesquisadores começaram a explorar a ideia de criar máquinas que pudessem simular o comportamento humano [1]. Inicialmente, o foco da IA estava na resolução de problemas lógicos e de jogo, como o xadrez [2]. Com o passar dos anos, o campo evoluiu, incorporando novas abordagens, como redes neurais e aprendizado de máquina. Na década de 1980, o desenvolvimento de redes neurais profundas começou a ganhar destaque [3], mas foi com a explosão de dados e o aumento do poder computacional na década de 2010 que a IA se consolidou como uma tecnologia revolucionária [4].

Dentro desse contexto, as IAs que geram imagens emergiram como uma área fascinante e altamente inovadora. O desenvolvimento de modelos de aprendizado profundo trouxe novas possibilidades para a criação de imagens. Em 2014, a introdução das Redes Generativas Adversariais (GANs) [5] marcou um ponto de inflexão, permitindo a geração de imagens inesperadamente realistas. Desde então, a pesquisa nessa área evoluiu rapidamente, levando ao surgimento de diversas abordagens e modelos, incluindo os métodos probabilísticos, que utilizam processos baseados em ruído para criar imagens a partir de descrições textuais ou transformações de outras imagens.

Desde então, a geração de imagens por IAs tem se tornado importante em várias áreas. No campo da educação, as ferramentas de IA para geração de imagens oferecem um potencial revolucionário para aprimorar o processo de aprendizado e torná-lo mais acessível e envolvente. Isso porque podem gerar diagramas interativos, gráficos dinâmicos e ilustrações detalhadas sob demanda durante o processo de ensino [6]. Na área da medicina, a geração de imagens por IA oferece ferramentas poderosas para auxiliar no diagnóstico, planejamento cirúrgico e pesquisa clínica [7]. Além disso, elas podem ser utilizadas para gerar dados sintéticos para uso em treinamento de novos profissionais em qualquer área que utilize imagens, como medicina, astronomia, geologia e vários outros. Designers podem usar IA para gerar

conceitos de produtos, como roupas, itens de decoração e embalagens de produtos e publicidade. Artistas podem usar ferramentas de geração de imagens como colaboradores. Por fim, a geração de imagens pode ser utilizada como uma forma de inclusão, para criar recursos visuais que ajudem pessoas com deficiências visuais a interagir com o mundo ao seu redor, como gráficos em braille ou imagens adaptadas, além de gerar padrões fluídos que simbolizem diferentes sons para deficientes auditivos.

Uma área que é altamente dependente do uso de imagens, tanto para a caracterização como literalmente com a sua aplicação final mais importante, os *displays*, é a área de cristais líquidos [8]. A geração de imagens impulsionada IAs oferece um potencial transformador na pesquisa e disseminação de informações sobre cristais líquidos. Através da IA, é possível gerar modelos visuais detalhados das microestruturas complexas dos cristais líquidos, facilitando a compreensão de suas propriedades físicas e ópticas, que são essenciais para o desenvolvimento de novos materiais e dispositivos.

A capacidade de visualizar simulações e transições de fase em cristais líquidos, geradas por IA, permite que pesquisadores identifiquem padrões e comportamentos que seriam difíceis de observar experimentalmente. Essas visualizações auxiliam na otimização de parâmetros de design para aplicações específicas, como displays de alta resolução ou sensores altamente sensíveis [9]. Além disso, a IA pode gerar representações interativas e acessíveis de dados complexos de pesquisa sobre cristais líquidos, tornando o conhecimento mais acessível a estudantes, engenheiros e outros profissionais não especializados [10]. Isso facilita a disseminação de informações científicas, impulsionando a inovação e a colaboração interdisciplinar.

A geração de imagens por IA também pode ser utilizada para criar materiais de marketing e divulgação mais atraentes, aumentando o interesse público pela ciência dos cristais líquidos e suas aplicações. Ao tornar a ciência mais visual e envolvente, a IA pode ajudar a inspirar a próxima geração de cientistas e engenheiros a explorar este campo [11].

Modelos probabilísticos de difusão por remoção de ruído (DDPM¹), ou, mais popularmente, modelos de difusão são modelos de variáveis latentes [12] altamente competitivos na tarefa de síntese de imagens [13] [14]. Baseados no conceito de termodinâmica fora do equilíbrio [15], esses modelos aprendem a remover o ruído de variáveis latentes $x_{1:T}$ de acordo com um processo reverso. As latentes são obtidas pelo processo direto, via amostragem aleatória Markoviana, e podem ser computadas rapidamente [16].

O processo direto é escolhido de tal forma que x_T é, aproximadamente, ruído Gaussiano isotrópico com média $\mathbf{0}$ e covariância $\mathbf{I}$ [15, 16]. Com auxílio da teoria de autocodificadores variacionais [17] [18], é possível aprender o processo reverso usando uma rede neural que atua como um decodificador das latentes $x_{1:T}$. Como o processo reverso é bem descrito [15], tendo em mãos o modelo treinado, para gerar um dado, parte-se de ruído Gaussiano, de média $\mathbf{0}$ e covariância $\mathbf{I}$, em T e segue-se o caminho reverso até $t = 1$.

¹do inglês *Denoising Diffusion Probabilistic Models*

No primeiro capítulo, introduzimos os principais conceitos utilizados neste trabalho, baseados em DDPM, e expomos relações documentadas entre DDPM e modelos generativos baseados em energia (aqui, usamos uma tradução livre para o português de SGM², baseado em [13]).

No segundo capítulo, apresentamos os principais conceitos de cristais líquidos, com foco no banco de dados utilizado neste trabalho.

No terceiro capítulo, investigamos a escolha da arquitetura de DDPM [16], que se baseia na arquitetura U-Net [19]. Comparamos os resultados desta arquitetura e de uma arquitetura convolucional de [20] nos problemas de regressão e classificação categórica, sob banco de dados consideravelmente reduzido. Observamos que U-Nets possuem considerável vantagem em relação a redes convolucionais no que tange problemas de regressão sob banco de dados reduzido. Baseados nessas observações, contruímos nossa rede neural customizada como uma versão simplificada da arquitetura de DDPM [16].

No quarto capítulo, apresentamos os resultados da nossa rede de difusão customizada. Por fim, expomos as conclusões de nossa pesquisa, bem como intenções de pesquisas futuras.

²do inglês *Score-Based Generative Models*

Modelos de Difusão por Remoção de Ruído

1.1 Introdução

IAs que geram imagens representam um campo vasto e diversificado, abrangendo uma variedade de técnicas e modelos, cada um com suas características distintas e aplicações específicas. Esses modelos têm a capacidade de criar imagens completamente novas a partir do zero, baseando-se em padrões complexos aprendidos de extensos conjuntos de dados. Essa abordagem permite a geração de visuais originais que podem variar de paisagens hiper-realistas a obras de arte abstratas. Além disso, essas IAs podem modificar imagens já existentes, adaptando-as e transformando-as de maneiras inovadoras, como no caso da estilização neural [21], que é uma técnica que possibilita aplicar as características artísticas de uma imagem de referência a outra imagem, combinando conteúdo e estilo de forma harmoniosa. Isso abre portas para a criação de obras visuais únicas, onde o estilo de um pintor famoso pode ser transferido para uma fotografia, resultando em uma peça que evoca a estética desejada.

Um dos métodos mais renomados e influentes neste campo é o das GANs. As GANs utilizam uma arquitetura composta por duas redes neurais que competem entre si em um "jogo" estratégico. Uma rede, conhecida como geradora, tem a tarefa de criar imagens que se assemelhem o máximo possível às imagens reais presentes no conjunto de dados de treinamento. A outra rede, chamada discriminadora, atua como um avaliador crítico, tentando distinguir entre as imagens geradas e as imagens reais [5]. Esse processo competitivo leva a geradora a aprimorar continuamente sua capacidade de criar imagens realistas, enquanto a discriminadora se torna cada vez mais precisa em sua avaliação. O resultado final é a geração de imagens que muitas vezes são indistinguíveis das imagens reais, demonstrando o

poder das GANs em produzir conteúdo visual de alta qualidade.

Outra abordagem importante são os Autocodificadores Variacionais (VAEs) [22], que oferecem uma maneira diferente de gerar novas imagens. Os VAEs aprendem a representar os dados de entrada em um espaço latente comprimido, capturando as características essenciais das imagens. Ao amostrar desse espaço latente, os VAEs podem gerar novas imagens que compartilham características com o conjunto de dados original, mas que também apresentam variações e inovações. Essa capacidade de gerar imagens a partir de representações compactas torna os VAEs úteis para tarefas como a criação de animações e a geração de imagens com atributos específicos controlados.

Com o advento da arquitetura Transformer, originalmente desenvolvida para o processamento de linguagem natural, surgiram modelos inovadores como o DALL-E [23], que podem criar imagens a partir de descrições textuais [24]. Esses modelos são treinados em vastos conjuntos de dados que contêm tanto imagens quanto texto, permitindo que a IA aprenda a associar palavras a conceitos visuais. Quando um usuário fornece uma descrição textual, o modelo utiliza essa informação para gerar uma imagem que corresponda à descrição, abrindo novas possibilidades para a criação de conteúdo visual personalizado e automatizado.

Um método que vem ganhando grande destaque é o DDPM. Em comparação com outros métodos de geração de imagens, como GANs e VAEs, apresenta algumas vantagens significativas. Essas vantagens abrangem diversos aspectos, desde a qualidade das imagens geradas até a estabilidade do treinamento e o nível de controle oferecido ao usuário.

Os modelos de difusão geralmente produzem imagens de alta qualidade, com maior fidelidade visual em comparação a outros métodos como as GANs, especialmente em resoluções mais altas. Essa superioridade se deve à forma como os modelos de difusão operam, refinando gradualmente uma imagem a partir de ruído aleatório. Eles tendem a ser menos propensos a artefatos e problemas de instabilidade que frequentemente afetam as GANs, que podem gerar imagens com ruídos, borrões ou outros defeitos visuais indesejados. A capacidade de gerar imagens mais limpas e detalhadas torna os modelos de difusão ideais para aplicações onde a qualidade visual é primordial [25]. Os modelos de difusão também são capazes de gerar uma maior variedade de imagens, explorando o espaço latente de maneira mais completa. Isso resulta em imagens que são mais diversas e menos propensas a "colapsar" em um modo único, um problema que pode ocorrer com outros métodos, onde o modelo tende a gerar sempre o mesmo tipo de imagem [26]. Os modelos de difusão também oferecem maior controle sobre o processo de geração de imagens. Essa característica é particularmente evidente em modelos onde é possível influenciar a imagem final através de prompts textuais detalhados [27]. Essa capacidade permite ajustes finos e a criação de imagens que correspondam precisamente à descrição desejada. Por fim, o treinamento de modelos de difusão tende a ser mais estável, por serem menos sensíveis a hiperparâmetros [16], e são notáveis pela capacidade de gerar detalhes finos e complexos nas imagens. Eles conseguem capturar nuances sutis, como

texturas, sombras e outros elementos que tornam as imagens mais realistas [28].

1.2 DDPM

Introduzimos agora os principais conceitos teóricos por trás dos modelos probabilísticos de difusão por remoção de ruído (DDPM¹), ou, popularmente, modelos de difusão, que são utilizados neste trabalho. Modelos de difusão são modelos de variáveis latentes com promissores resultados documentados no campo de síntese de imagens [13]. Variáveis latentes são variáveis ocultas que representam de forma comprimida as características essenciais dos dados [17]. Com base nisto, a teoria dos modelos de difusão propõe um processo de destruição do dado x_0 através da geração de variáveis latentes $x_1, x_2, \dots, x_T$ deste, via processo de destruição por ruído Gaussiano conhecido, $q(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t\mathbf{I})$. Se β_t for pequeno, o processo reverso $p(x_{t-1}|x_t)$ também é Gaussiano [15]. Dado T suficientemente grande, e transições bem conhecidas, a variável latente x_T é uma Gaussiana. Treinamos um modelo $p_\theta(x_{t-1}|x_t, y)$ que aprende um processo reverso aproximado, parametrizado por suas próprias variáveis θ , e condicionado por um condicional y . Este presente capítulo aborda os principais aspectos da teoria de difusão.

1.3 Processo Direto

Começamos por um conjunto de dados com imagens $x_0 \sim q(x_0)$. O modelo de difusão tem como ponto de partida o processo de destruição por ruído aleatório Gaussiano de x_0 gerando latentes $x_1, x_2, \dots, x_T$. O elemento x_t possui distribuição condicionada em x_{t-1} por definição:

$$q(x_t|x_{t-1}) := \mathcal{N}(x_t; \sqrt{1 - \beta_t}x_{t-1}, \beta_t\mathbf{I}), \quad (1.1)$$

onde $\beta_t \in (0, 1)$ é uma variância de ruído pequeno e $\mathcal{N}(\mu, \Sigma)$ denota uma distribuição normal. Este processo descreve a distribuição do dado aleatório x_t condicionado somente pelo dado aleatório anterior x_{t-1} . Dada essa definição, amostra-se uma latente x_t de uma distribuição Gaussiana com média $\sqrt{1 - \beta_t}x_{t-1}$ e covariância $\beta_t\mathbf{I}$. A distribuição conjunta das latentes de x_0 é:

$$q(x_0, x_1, \dots, x_T) = q(x_0) \prod_{t=1}^T q(x_t|x_{t-1}), \quad (1.2)$$

Ou:

¹do inglês *Denoising Diffusion Probabilistic Models*

$$q(x_1, x_2, \dots, x_T | x_0) = \prod_{t=1}^T q(x_t | x_{t-1}). \quad (1.3)$$

Como notado em [16], mediante parametrização, a amostragem das latentes pode ser consideravelmente simplificada. Ou seja, podemos derivar uma equação que calcula x_t diretamente a partir de x_0 e ruído $\epsilon \sim \mathcal{N}(0, 1)$, sem passar pelos passos intermediários. Seja $\alpha_t := 1 - \beta_t$ e $\bar{\alpha}_t := \prod_{i=1}^t \alpha_i$, então:

$$q(x_t | x_0) = \mathcal{N}(x_t; \sqrt{\bar{\alpha}_t} x_0, (1 - \bar{\alpha}_t) \mathbf{I}), \quad (1.4)$$

em $t = T$, o dado se torna praticamente ruído Gaussiano isotrópico puro com média zero e covariância identidade. Nesse contexto, a distribuição marginal de x_T , amostrado estocasticamente, é:

$$q(x_T) \approx \mathcal{N}(x_T; \mathbf{0}, \mathbf{I}). \quad (1.5)$$

Via equação 1.4, amostramos quaisquer latentes x_t com pouco custo computacional:

$$x_t = \sqrt{\bar{\alpha}_t} x_0 + \sqrt{1 - \bar{\alpha}_t} \epsilon, \quad \epsilon \sim \mathcal{N}(0, \mathbf{I}) \quad (1.6)$$

o que significa que no tempo t , a imagem é uma combinação linear da imagem original e do ruído gaussiano. A figura 1.1 é formada por latentes x_t geradas pelo processo descrito acima. O sinal de ruído $\sqrt{1 - \bar{\alpha}_t}$ se torna aproximadamente 1 em T , ao passo que o sinal $\sqrt{\bar{\alpha}_t}$ do dado x_0 se torna aproximadamente 0. Nas figuras 1.2 e 1.3, são apresentados os sinais da média e da raiz quadrada da covariância, respectivamente, das latentes ao longo de t . Na figura 1.4, são apresentados os sinais da média e raiz quadrada da covariância, utilizados na amostragem das latentes em t .

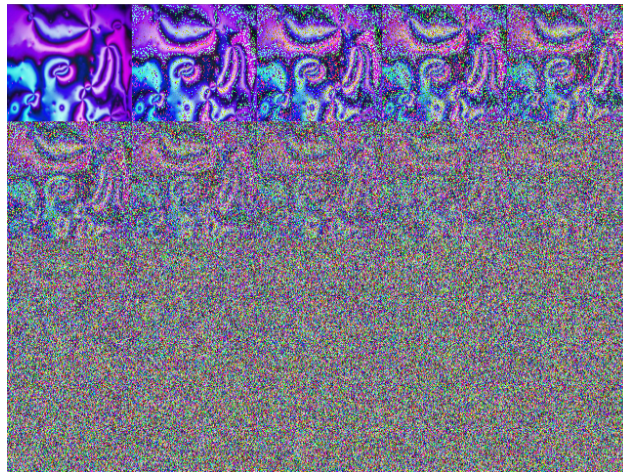


Figura 1.1: Imagem contendo 20 latentes x_t igualmente separadas, e aclopadas de forma contínua, da esquerda para direita. O processo da imagem possui 200 passos.

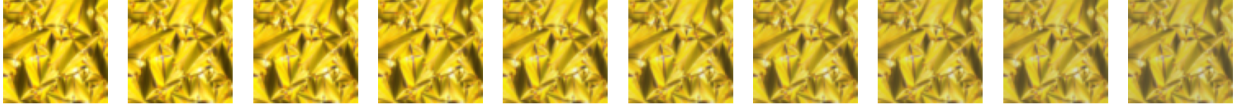


Figura 1.2: Médias amostradas de 10 latentes x_t igualmente espaçadas de uma corrente de 100 passos.

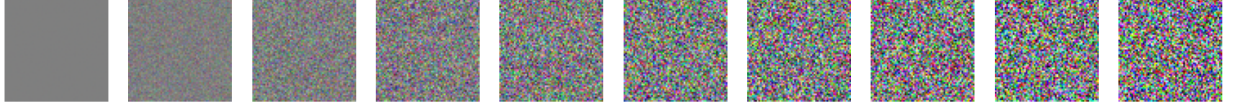


Figura 1.3: Elementos de ruído amostrados da mesma corrente. Como podemos ver, o sinal $\sqrt{1 - \bar{\alpha}_t}$ é próximo de 1 em x_T . Dados valores de β_t pequenos com transições fixas, quanto mais passos o processo tiver menor será $\bar{\alpha}_T$.

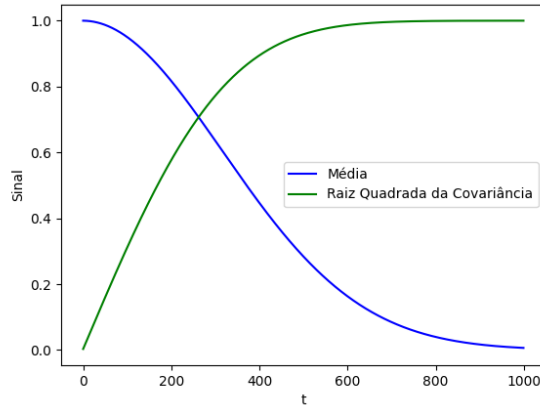


Figura 1.4: As intensidades da média e da raiz quadrada da covariância são construídas de forma que as amostras latentes aleatórias sejam paulatinamente amostras de ruído Gaussiano puro, com média zero e covariância identidade.

1.4 Processo Reverso

Como visto no anteriormente, se T é suficientemente grande, $q(x_T) \approx \mathcal{N}(x_T; \mathbf{0}, \mathbf{I})$. Seja $p(x_{t-1} | x_t)$ o processo reverso. A distribuição conjunta do processo reverso é definida como [15]:

$$p(x_0, x_1, \dots, x_T) = p(x_T) \prod_{t=1}^T p(x_{t-1} | x_t), \quad (1.7)$$

com:

$$p(x_T) = \mathcal{N}(x_T; \mathbf{0}, \mathbf{I}). \quad (1.8)$$

Seja $p_\theta(x_{t-1} \mid x_t)$ o processo reverso aprendível pela rede neural, onde θ são os parâmetros do modelo:

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \Sigma_\theta(x_t, t)). \quad (1.9)$$

A média $\mu_\theta(x_t, t)$ e a covariância $\Sigma_\theta(x_t, t)$ são parametrizadas pela rede. Em DDPM, a média $\Sigma_\theta(x_t, t)$ é definida de forma que esta seja igual ou proporcional à β_t e, portanto, não é parametrizada pela rede. O treino é definido a partir do objetivo de aproximar a distribuição conjunta do processo reverso aproximado, parametrizado por θ , à distribuição conjunta do processo direto [15, 29]:

$$p_\theta(x_0, x_1, \dots, x_T) \approx q(x_0, x_1, \dots, x_T). \quad (1.10)$$

Este objetivo é alcançado mediante análise do posterior do processo direto, via teorema de Bayes [30]:

$$q(\mathbf{x}_{t-1} \mid \mathbf{x}_t, \mathbf{x}_0) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_q(\mathbf{x}_t, \mathbf{x}_0), \Sigma_q(\mathbf{x}_t)) \quad (1.11)$$

$$\mu_q(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \mathbf{x}_0 + \frac{\sqrt{\alpha_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_t \quad (1.12)$$

$$\Sigma_q(\mathbf{x}_t) = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t \mathbf{I} \quad (1.13)$$

Ho et al. [16] notam que a minimização da divergência entre as distribuições $q(x_{t-1} \mid x_t, x_0)$ e $p_\theta(x_{t-1} \mid x_t)$ é crucial para o treino, já que a forma de $p_\theta(x_{t-1} \mid x_t)$ também é uma Gaussiana, tornando a aproximação entre as distribuições computável.

1.5 Treino

O objetivo do modelo é alcançado através da minimização da divergência Kullback-Leibler (KL) [31] entre as duas distribuições conjuntas [13]:

$$KL(q(x_0, x_1, \dots, x_T) \parallel p_\theta(x_0, x_1, \dots, x_T)) \quad (1.14)$$

$$= -\mathbb{E}_{q(x_0, x_1, \dots, x_T)} [\log p_\theta(x_0, x_1, \dots, x_T)] + C \quad (1.15)$$

$$= \underbrace{\mathbb{E}_{q(x_0, x_1, \dots, x_T)} \left[-\log p(x_T) - \sum_{t=1}^T \log \frac{p_\theta(x_{t-1} | x_t)}{q(x_t | x_{t-1})} \right]}_{:= -L_{VLB}(x_0)} + C \quad (1.16)$$

$$\geq \mathbb{E}[-\log p_\theta(x_0)] + C. \quad (1.17)$$

Em que 1.15 vem da definição de divergência KL, 1.16 da propriedade que os conjuntos de p e q são produtos de distribuições [13], e 1.17 da desigualdade de Jensen [15]. C representa constantes que não dependem de θ e, portanto, não estão envolvidas no treino. O primeiro termo da equação 1.16 é o negativo do limite inferior variacional.

A expressão 1.16 pode ser reescrita como [16]:

$$\mathbb{E} \left[\underbrace{D_{\text{KL}}(q(\mathbf{x}_T | \mathbf{x}_0) | p(\mathbf{x}_T))}_{L_T} + \sum_{t>1} \underbrace{D_{\text{KL}}(q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) | p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t))}_{L_{t-1}} - \underbrace{\log p_\theta(\mathbf{x}_0 | \mathbf{x}_1)}_{L_0} \right] \quad (1.18)$$

Pode-se notar que o termo L_{t-1} é destaque da expressão. A divergência entre $q(x_T | x_0)$ e $p(x_T)$ em L_T já é pequena pois garantimos transições pequenas para um T suficientemente grande, garantindo que $q(x_T | x_0)$ é aproximadamente uma Gaussiana isotrópica. E a probabilidade do modelo $p_\theta(x_0 | x_1)$, representada pelo termo L_0 da expressão, que pode ser calculada medindo a probabilidade de um bin estar correto dentre 256 bins para cada pixel da imagem, também é consideravelmente menos significativa que L_{t-1} , já que a transição entre $t-1$ e t sempre será suave (garantido pelo processo direto e pela forma do posterior 1.11). Assim, o termo L_{t-1} é o principal otimizador do treino.

Em DDPM, o objetivo do treino é maximizar L_{VLB} , ou, minimizar $-L_{VLB}$. Originalmente, objetivando otimizar a qualidade amostral do modelo, propôs-se a simplificação do objetivo para a minimização da função de erro quadrado médio (MSE ²) [13], baseado na simplificação do termo L_{t-1} da expressão 1.18 [16]:

$$\mathbb{E}_{t \sim \mathcal{U}([1, T]), \mathbf{x}_0 \sim q(\mathbf{x}_0), \epsilon \sim \mathcal{N}(0, \mathbf{I})} [\lambda(t) \|\epsilon - \epsilon_\theta(\mathbf{x}_t, t)\|^2] \quad (1.19)$$

Foi notado em [16] que esse objetivo lembra o de modelos baseados em energia:

$$\mathbb{E}_{t \sim \mathcal{U}[0, T], \mathbf{x}_0 \sim q(\mathbf{x}_0), \mathbf{x}_t \sim q(\mathbf{x}_t | \mathbf{x}_0)} [\lambda(t) \|\mathbf{s}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log q_{0t}(\mathbf{x}_t | \mathbf{x}_0)\|^2], \quad (1.20)$$

em que $\lambda(t)$ é a função de ponderação positiva, x_t a latente computada de x_0 , e ϵ_θ é uma rede neural com parâmetros θ que prevê o ruído ϵ dado x_t e t . Adotamos a mesma escolha de $\lambda(t)$ de DDPM [16] para este trabalho, bem como um conjunto igualmente distribuído

²do inglês *Mean Squared Error*

β_t linear.

Após treinar a rede neural, podemos amostrar x_{t-1} , lembrando que $x_{t-1} \sim p_\theta(x_{t-1}|x_t)$. Como $p_\theta(x_{t-1}|x_t)$ é uma Gaussiana, as médias e covariâncias podem ser parametrizadas pelos posteriores do processo direto [16], resultando na amostragem reversa:

$$\mathbf{x}_{t-1} = \underbrace{\frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{\beta_t}{\sqrt{1 - \bar{\alpha}_t}} \boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t) \right)}_{\tilde{\boldsymbol{\mu}}_t} + \sigma_t \mathbf{z}, \quad \text{onde } \mathbf{z} \sim \mathcal{N}(0, \mathbf{I}) \quad (1.21)$$

A média $\tilde{\boldsymbol{\mu}}_t$ é parametrizada pela equação 1.12 e pelo fato de x_0 ser parametrizado por x_t pela equação 1.6. Pode-se escolher σ_t^2 como igual à covariância do posterior do processo direto (equação 1.13), mas há uma forma simplificada, utilizada por Ho et al. [16], que também funciona: $\sigma_t^2 = \beta_t$. Escolhemos essa mesma forma, utilizando $\sigma_t = \sqrt{\beta_t}$ durante amostragem reversa.

1.6 Conexão com Modelos Baseados em Energia

Há uma relação entre modelos de difusão e modelos generativos baseados em energia (SGM³) [16]. Por consequência, o processo de amostragem em DDPM segue o processo de Langevin Monte Carlo [32] análogo ao usado em SGM. Baseado nessa relação, [13] nota-se que DDPM é uma solução da equação diferencial estocástica (EDE):

$$dx = - \underbrace{\frac{1}{2}\beta(t)x dt}_{f(x,t)} + \underbrace{\sqrt{\beta(t)} dw}_{g(t)}, \quad (1.22)$$

em que $f(x, t)$ e $g(t)$ são as funções de difusão e de *drift* da EDE, e w é movimento Browniano, com $\beta(\frac{t}{T}) = T\beta_t$ se T tende ao infinito.

Analogamente, o processo reverso é a solução da EDE:

$$dx = -\frac{1}{2}\beta(t)(x_t - \nabla_{x_t} \log q_t(x_t))dt + \sqrt{\beta(t)}d\bar{w}, \quad (1.23)$$

onde $\bar{w}$ é o movimento Browniano no sentido contrário do tempo e dt é um intervalo de tempo negativo infinitesimal. Soluções da equação 1.23 são processos de difusão que gradualmente convertem ruído em dado [13, 33].

1.7 Avaliação do Modelo

No contexto de modelos generativos por difusão, o objetivo do treinamento é maximizar a log-verossimilhança L do conjunto de dados reais sob o modelo p_θ [15]. Para isso, busca-se

³do inglês *Score-Based Generative Models*

maximizar a probabilidade marginal:

$$p_\theta(x_0) = \int p_\theta(x_0, x_1, \dots, x_T) dx_1 \dots dx_T \quad (1.24)$$

A log-verossimilhança L do modelo é expressa como:

$$L = \int dx_1 \dots dx_T q(x_0) \log p_\theta(x_0) \quad (1.25)$$

Partindo da desigualdade de Jensen, reformula-se a maximização (intratável [15]) L da log-verossimilhança como a maximização do limite inferior $L_{VLB}(x_0)$:

$$L \geq \underbrace{\int d\mathbf{x}^{(0\dots T)} q(\mathbf{x}^{(0\dots T)}) \log \left[p(\mathbf{x}^{(T)}) \prod_{t=1}^T \frac{p(\mathbf{x}^{(t-1)} | \mathbf{x}^{(t)})}{q(\mathbf{x}^{(t)} | \mathbf{x}^{(t-1)})} \right]}_{L_{VLB}(x_0)} \quad (1.26)$$

Em DDPM, L_{VLB} é maximizado via minimização do erro MSE entre as previsões do modelo e os ruídos aplicados nas amostras. Assim, a avaliação prática do modelo durante o treinamento é realizada mediante o acompanhamento dessa métrica, cuja minimização está diretamente associada ao bom desempenho do processo de geração.

Portanto, ainda que o objetivo teórico envolva a maximização da log-verossimilhança, na prática, avaliamos o modelo pela métrica MSE, a qual reflete a capacidade do modelo em prever corretamente o ruído aplicado, etapa fundamental para uma reconstrução eficiente das amostras originais.

Neste capítulo, apresentamos um breve resumo de cristais líquidos e diferentes fases líquido-cristalinas, especialmente as utilizadas neste trabalho. Este conhecimento foi usado no tratamento das imagens e escolha do banco de dados.

2.1 Introdução

O microscópio óptico de luz polarizada (POM¹) desempenha um papel crucial no estudo de cristais líquidos, funcionando como uma ferramenta essencial para revelar a beleza e a complexidade destes materiais fascinantes [34]. A capacidade de visualizar a estrutura e as propriedades ópticas dos cristais líquidos é fundamental para a sua compreensão e aplicação. Ao utilizar a microscopia de polarização, é possível observar a orientação molecular e a ordem que caracteriza as diferentes fases líquido-cristalinas. As variações na cor e no padrão de interferência da luz polarizada fornecem informações valiosas sobre a estrutura do cristal líquido, permitindo a identificação das diferentes fases [34]. A microscopia óptica, portanto, oferece uma janela direta para a compreensão das propriedades ópticas e estruturais dos cristais líquidos, facilitando o estudo de como estes materiais respondem a diferentes estímulos, como campos elétricos ou alterações de temperatura.

A análise de texturas é igualmente importante no estudo das fases líquido-cristalinas. As texturas, ou padrões característicos observados no POM, são uma impressão digital visual de cada fase. Cada fase líquido-cristalina apresenta uma textura única, resultante da maneira como as moléculas se organizam e se alinham. A observação cuidadosa das texturas permite identificar e diferenciar as fases, pois as texturas são altamente sensíveis à estrutura molecu-

¹do inglês *Polarized Optical Microscopy*

lar e às forças intermoleculares. Por exemplo, a fase nemática, caracterizada pela orientação direcional das moléculas, exibe texturas como fios ou *schlieren*, enquanto a fase esmética, com camadas moleculares ordenadas, apresenta texturas como ‘degraus’ ou ‘bastões’. A análise das texturas, portanto, é uma ferramenta poderosa para caracterizar e classificar as diferentes fases líquido-cristalinas. A análise de texturas é uma ferramenta indispensável no estudo de cristais líquidos, juntamente com a microscopia óptica, fornecendo uma compreensão abrangente da estrutura, propriedades e comportamentos destes materiais. Portanto, o desenvolvimento da pesquisa e de um pesquisador na área está diretamente associado ao entendimento e à aprendizagem da interpretação de texturas.

Os cristais líquidos representam um estado de agregação da matéria intermediário entre os líquidos convencionais e os sólidos cristalinos [35]. Nesses materiais, as moléculas mantêm um certo grau de ordenação dirigida (ordem orientacional) e, em alguns casos, apresentam uma disposição posicional em certo grau. Essa combinação ímpar de propriedades resulta em características físicas e ópticas únicas, que têm sido amplamente exploradas na fabricação de dispositivos como telas planas, monitores, sensores e sistemas de modulação óptica. A principal característica dos cristais líquidos é a anisotropia, ou seja, propriedades físicas que mudam conforme a orientação molecular referente a um certo sistema de referência. Esta anisotropia ocorre em basicamente todas as propriedades físicas, assim como se observa em cristais sólidos, porém, por terem propriedades de líquidos, apresentam propriedades anisotrópicas também relacionadas à fluidez. Desta forma, estes tipos de materiais possuem propriedades únicas, respondendo de maneira elástica a distúrbios que tendam a destituir a ordem molecular e fluir caso o distúrbio não altere a ordem molecular, ou seja, são materiais viscoelásticos. Existem basicamente dois tipos de cristais líquidos: os Liotrópicos e os Termotrópicos. Os Liotrópicos ocorrem quando moléculas anfífilas (com dupla afinidade como surfactantes com uma parte polar e outra apolar) são adicionadas a meios aquosos, por exemplo, causando autoagregação e a formação de estruturas micelares que podem formar fases líquido-cristalinas. Este trabalho foca nas fases Termotrópicas, mas o leitor pode encontrar mais informações sobre fases Liotrópicas em [36]. Termotrópicos, por sua vez, são formados por materiais com moléculas orgânicas com formato anisométrico (como bastões). A figura 2.1 (a) mostra uma molécula do material 5CB, um dos materiais mais utilizados no estudo de cristais líquidos. Nestes materiais, mudanças em temperatura determinam a existência de diferentes fases. Na figura 2.1 (b) damos um exemplo de como a mudança da temperatura resulta na perda de ordem gradual. Em baixas temperaturas, o material se encontra na fase sólido-cristalina. Conforme a temperatura aumenta, uma fase líquido-cristalina 1 surge. Esta fase 1, que ainda tem propriedades cristalinas, é chamada de esmética e será discutida melhor nas próximas seções. Em seguida, com o aumento da temperatura, uma nova fase líquido-cristalina 2, chamada de nemática (também discutida abaixo), é observada. Com o aumento da temperatura, eventualmente todas as propriedades

líquido-cristalinas são perdidas e uma fase isotrópica ordinária surge.

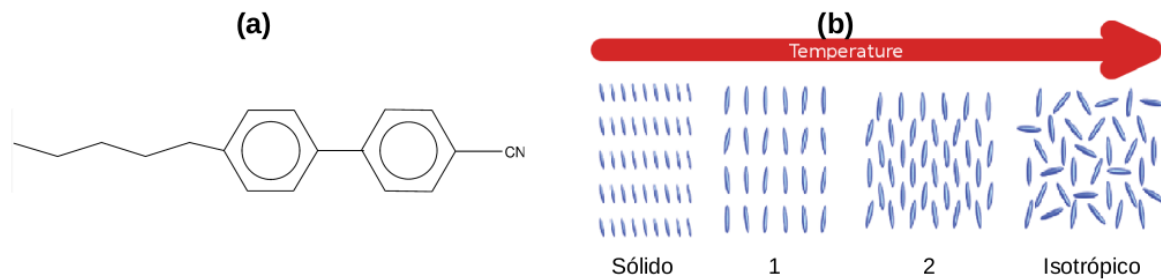


Figura 2.1: (a) Típica molécula de cristal líquido termotrópico, conhecida pelo nome 5CB. (b) Transições de fase encontradas em um material que forma fases líquido cristalinas: em baixa temperaturas, temos a fase sólido cristalina, enquanto em alta temperaturas encontra-se a fase líquido isotrópica. No exemplo da figura (b), as fases 1 e 2 representam duas fases líquido cristalinas (esméctico e nemático), porém, podem existir várias fases em temperaturas entre o sólido cristalino e o líquido isotrópico.

As principais características dos cristais líquidos incluem: 1-Anisotropia Óptica e Elástica: A presença de uma orientação preferencial das moléculas gera propriedades ópticas que podem ser moduladas por campos elétricos e magnéticos. 2-Sensibilidade a Estímulos Externos: Alterações na temperatura, pressão e campos aplicados podem induzir transições entre diferentes fases, possibilitando o controle de suas propriedades. 3- Auto-organização Molecular: A tendência das moléculas em se organizarem de forma ordenada sem a necessidade de um template externo favorece o aparecimento de estruturas complexas e diversificadas. 4- Aplicabilidade Tecnológica: Desde a sua descoberta, os cristais líquidos têm sido a base para avanços na área de displays, dispositivos optoeletrônicos e novas tecnologias de materiais inteligentes.

O estudo dos cristais líquidos não só permite entender os fenômenos de transição de fase e a relação entre a estrutura molecular e as propriedades macroscópicas, mas também abre caminhos para o desenvolvimento e aperfeiçoamento de dispositivos inovadores.

2.2 Nemáticos

Na fase isotrópica, as moléculas estão orientadas de forma aleatória. Na fase nemática, a mais simples das fases líquido-cristalinas, as moléculas exibem uma orientação preferencial definida por um eixo diretor, sem uma ordem de posição de longo alcance. A direção molecular comum, descrita por um vetor, chamado de diretor $\vec{n}$, representa a orientação média das moléculas. Na fase nemática, $\vec{n} \equiv -\vec{n}$, o que significa que a fase é não polar. A Figura 2.1 (a) exibe a famosa molécula 5CB, que foi um dos primeiros e mais investigados cristais líquidos

nemáticos. Assim como a maioria das moléculas de cristal líquido, o 5CB é constituído por duas partes: um núcleo rígido e uma cauda flexível, ambos igualmente importantes para a formação da fase. Enquanto o núcleo rígido assegura a rigidez necessária para a interação estérica (essencial, porém, não suficiente para a obtenção da fase nemática), a cauda flexível aprimora o arranjo das moléculas em relação às demais [37]. A Figura 2.1(b)-2, mostra a estrutura da fase nemática.

Fases nemáticas são observadas para basicamente todos os tipos de estruturas que formam fases líquido-cristalinas, como moléculas calamíticas (bastões), discóticas e com centro dobrado (bananas). Essa disposição molecular confere um caráter fluido ao sistema, no qual as flutuações térmicas induzem variações locais na orientação. O nome nemático vem do grego *nema* que significa fio, visto que uma das características principais da fase é a presença de defeitos chamados de disclinações, que são regiões em que a orientação do diretor muda abruptamente em um pequeno espaço, resultando em regiões onde não existe uma orientação molecular bem definida. Por isso mesmo, sob microscopia óptica com luz polarizada, as amostras nemáticas geralmente exibem texturas do tipo *Schlieren* (quando a orientação não é homogênea, por exemplo, em uma célula sem tratamento nas superfícies). Características marcantes são as linhas de disclinação, que podem ter cargas topológicas de $+1/2$ ou $-1/2$, evidenciando os defeitos orientacionais e as transições suaves no campo do diretor. Defeitos do tipo ± 1 também são observados, mas tipicamente são menos estáveis e eventualmente tornam-se do tipo $\pm 1/2$. A Figura 2.2 (a) mostra os diretores (pequenos retângulos) e o contraste entre zonas escuras e claras que se observa desta textura em um microscópio de luz polarizada. A Figura 2.2 (b) mostra uma foto típica de um nemático com a textura *Schlieren*. Se uma amostra nemática é preenchida em uma célula com superfícies tratadas para que as moléculas fiquem perpendiculares a elas (chamado de homeotrópico), em um POM não haverá propagação de luz devido a uniaxialidade da fase. Por outro lado, em uma amostra tratada para induzir orientação paralela as superfícies (chamado de planar), uma textura uniforme com cores devido a birrefringência será observada. A Figura 2.2 (c) mostra um exemplo de um nemático planar observado via POM.

2.3 Colestéricos

Se uma fase nemática se torna quiral, o que pode ser feito sintetizando nemáticos com grupos quirais ou simplesmente dopando um nemático com materiais quirais, tem-se a fase colestérica. Os colestéricos são essencialmente nemáticos, o que significa que as moléculas não têm ordem posicional e apenas ordem de orientação. Os colestéricos são organizados localmente em “planos”, cada um se comportando como nemáticos, mas girando de um “plano” para outro ao longo de uma direção perpendicular (chamada eixo helicoidal) que se repete a cada 360° , em um comprimento chamado passo, P , onde este comprimento é

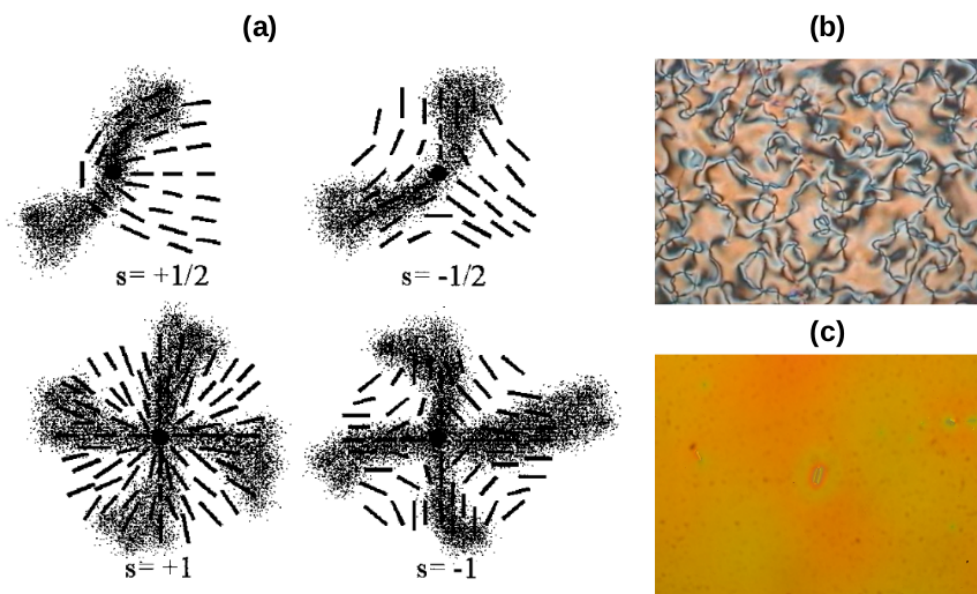


Figura 2.2: (a) Relação entre a ordem molecular e os padrões claro e escuro típicamente observados em cristais líquidos nemáticos devido aos defeitos. Em (b), uma típica textura nemática, também conhecido como textura *Schlieren*. Em (c), uma textura de POM de um nemático orientado de forma planar. Figuras (a) e (b) retiradas de [8].

muito maior que os comprimentos moleculares. A Figura 2.3 (a) representa um esboço da estrutura colestérica. Em alguns casos em que a razão de torção é alta, os colestéricos podem ser considerados uma estrutura em camadas, apresentando defeitos semelhantes aos encontrados em fases lamelares. Por ser também a fase cristal líquido quiral mais simples, os colestéricos foram amplamente utilizados antes, desde o uso em displays até aplicações em lasing.

Se no caso nemático o estado uniforme não exige distorção, no caso colestérico a hélice deve ser o estado de energia mínima. Quando o comprimento de deformação é longo ($L/P > 1$), podem aparecer defeitos topológicos relacionados à estrutura em camadas, considerados defeitos macroscópicos, como focal cônicas e *oily streaks*. Por outro lado, para casos em que o comprimento das distorções é pequeno, a característica de orientação é predominante e, portanto, encontram-se defeitos relacionados à orientação das moléculas, assim como nos nemáticos. Neste caso, os colestéricos terão defeitos como disclinações e deslocamentos.

Defeitos relacionados à estrutura em camadas no nível macroscópico são muito importantes para aplicações em displays utilizando colestéricos. Existe um tipo de estrutura comum à maioria das estruturas em camadas, como esmétricos e colestéricos, que são as focal cônicas. Estas estruturas são a forma de manter o espaçamento e a orientação das camadas constantes, pois muitas vezes é imposto pela superfície ou fatores externos, como campos elétricos. Desta forma, o material se organiza em uma estrutura polidomínio onde cada domínio se orienta de forma aleatória mas mantendo o passo uniforme. O segundo defeito desta família

são as *oily streaks*, e é o defeito mais comum encontrado em cristais líquidos colestéricos [24]. Como o nome sugere, esses defeitos são linhas que atravessam, muitas vezes, todo o campo de visão da célula. Basicamente, seu comportamento depende das condições de ancoragem e das constantes elásticas do material. Sua estrutura interna é bastante complicada, basicamente dobrando camadas que não podem “se encontrar” após passar por algum obstáculo, como um espaçador, por exemplo. Tipicamente, quando um colestérico é colocado em uma célula planar, uma textura chamada de *Grandjean* é observada no POM. Esta textura se caracteriza pela presença de linhas, que são as *oily streaks*. Uma outra textura comumente observada é quando o colestérico se encontra em uma amostra com condições de contorno homeotrópicas. Neste caso, a hélice não é compatível com as condições de contorno, e surgem estruturas chamadas de *fingerprint* (se o passo é comparado à espessura da amostra) ou focal cônica se o passo for bem menor que a espessura. Na figura 2.3 (b) a (d), é mostrada uma textura de POM (lado esquerdo) e um esboço da estrutura helicoidal em uma célula (lado direito).

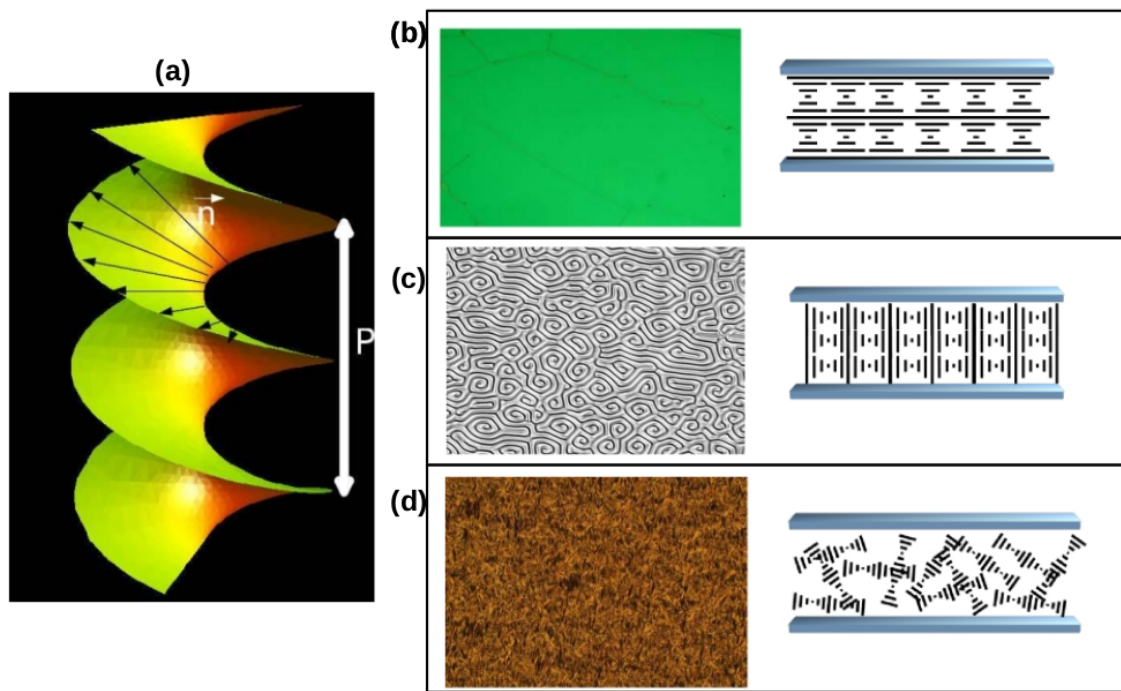


Figura 2.3: (a) Estrutura helicoidal colestérica e a distância P representando o passo. Nas figuras (b) a (d), a textura a esquerda mostra uma imagem de POM enquanto a ilustração a direita representa a estrutura helicoidal para as texturas *Grandjean* (ou planar), *fingerprint* e focal cônica, respectivamente. Imagens obtidas de [38].

2.4 Esméticos

As fases esméticas são um tipo de fase de cristal líquido mais ordenadas do que os nemáticos. As moléculas nas fases esméticas organizam-se em camadas distintas. Dentro de cada camada, as moléculas podem exibir um grau de ordem posicional, ao contrário dos nemáticos. Essa disposição em camadas proporciona um grau maior de ordem e um comportamento mais viscoso. O espaçamento entre as camadas é normalmente da ordem do comprimento molecular. Existem diferentes fases esméticas, variando na disposição dentro das camadas (por exemplo, inclinadas ou não inclinadas). Essas fases geralmente aparecem em temperaturas mais baixas em comparação com os nemáticos. As fases esméticas podem exibir propriedades como elasticidade e fluidez, mas com restrições devido à sua estrutura em camadas. Elas são comumente encontradas abaixo da fase nemática ou diretamente da fase isotrópica durante o resfriamento. Essa estrutura em camadas influencia suas propriedades ópticas e mecânicas. Elas desempenham um papel crucial em várias aplicações, incluindo displays e sensores. Dentre as várias fases esméticas, as relevantes para este trabalho são a esmética A, esmética C, esmética G, esmética I e esmética F, brevemente discutidas abaixo. Mais abaixo, na Figura 2.4 é mostrado um esboço de cada uma das fases esmética e nemática. Na Figura 2.4 (a), um esboço da orientação molecular média vista pelo lado (figura da esquerda) e vista por cima para demonstrar empacotamento molecular (figura da direita). Na Figura 2.4 (b), a ordem de temperatura que cada fase aparece, em ordem crescente de temperatura.

2.4.1 Esmético A

A fase esmética A (SmA) é uma fase cristal líquido notável por sua estrutura lamelar, onde as moléculas se auto-organizam em camadas bem definidas. Dentro destas camadas, as moléculas exibem um alinhamento direcional preferencial, mantendo-se aproximadamente perpendiculares à superfície da camada. Ao contrário das fases cristalinas, não há ordem posicional rígida dentro das camadas, permitindo que as moléculas deslizem umas sobre as outras, o que confere à fase SmA características de fluidez, o que faz com que a fase seja considerada um líquido em duas dimensões (entre as camadas). A distância interplanar é aproximadamente equivalente ao comprimento molecular das moléculas que a compõem. Tipicamente, a fase SmA manifesta-se numa janela de temperatura entre as fases nemática e cristalina sólida. Quando sujeita a campos externos, como elétricos ou magnéticos, a fase SmA apresenta respostas distintas, que podem ser utilizadas para fins de identificação. Devido às suas propriedades ópticas únicas, como a formação de texturas focais cônicas, a fase SmA encontra aplicações em displays, sensores e outros dispositivos optoeletrônicos. A Figura 2.5 mostra uma textura típica de um SmA, conhecida como *Fan-Shaped* (focal-cônicas).

2.4.2 Esmético C

A fase esmética C (SmC) é uma fase cristal líquido com uma característica estrutural distinta: as moléculas exibem um alinhamento em camadas, mas com uma inclinação em relação à normal das camadas. Essa inclinação, um elemento chave, confere à SmC propriedades piezoelétricas e ferroelétricas, dependendo se o material for quirál. Dentro de cada camada, as moléculas mantêm um grau de ordem direcional, mas com a inclinação, a estrutura se torna menos simétrica do que na fase SmA. As moléculas dentro de cada camada também podem deslizar umas sobre as outras, contribuindo para a fluidez, embora menos do que na SmA devido à inclinação. A inclinação molecular pode ser influenciada por campos elétricos, o que é crucial para aplicações em displays de cristal líquido. A SmC geralmente aparece em temperaturas mais baixas do que a SmA, e a transição entre as fases pode ser sensível a fatores como a estrutura molecular e as forças intermoleculares. As propriedades ferroelétricas da SmC, especialmente, a tornam relevante para dispositivos de comutação rápida e memória. Sua organização única e propriedades funcionais abrem portas para diversas tecnologias avançadas. A Figura 2.5 mostra uma textura típica de um SmC, conhecida como *Broken Fan-Shapped*, que ocorre quando a fase SmC transita a partir da textura de um SmA.

2.4.3 Esmético I

A fase esmética I (SmI) é uma fase cristal líquido caracterizada por uma estrutura em camadas com moléculas inclinadas em relação à normal das camadas, semelhante à fase SmC. No entanto, a fase SmI difere significativamente na organização dentro das camadas. As moléculas na fase SmI exibem uma ordem posicional hexática, formando uma estrutura mais organizada dentro de cada camada em comparação com a SmC. Esta ordem hexática intra-camada confere à fase SmI propriedades mecânicas e ópticas únicas. A inclinação das moléculas e a ordem hexática resultam em uma maior viscosidade em comparação com as fases SmA ou SmC. A orientação das moléculas e a ordem posicional são afetadas por campos elétricos e magnéticos, possibilitando o ajuste de suas propriedades. A fase SmI é geralmente observada em temperaturas ainda mais baixas do que as fases SmC e SmA. A combinação de inclinação molecular e ordem hexática intra-camada torna a fase SmI relevante para aplicações especializadas em materiais e dispositivos ópticos, onde a anisotropia precisa é essencial. A Figura 2.5 mostra uma textura típica de um SmI, conhecida como *Mosaico*.

2.4.4 Esmético F

A fase esmética F (SmF) compartilha semelhanças com a SmI, possuindo uma estrutura em camadas com moléculas inclinadas em relação à normal da camada. Assim como na SmI, a SmF também apresenta ordem posicional hexática dentro das camadas, indicando uma organização molecular significativa. A principal diferença entre SmF e SmI reside na direção

da inclinação molecular. Na SmF, a inclinação molecular ocorre em direção aos vértices da rede hexática dentro da camada, enquanto que na SmI essa orientação é na direção do ápice da rede hexática. Essa orientação da inclinação em relação à rede hexática tem implicações nas propriedades de simetria e nos comportamentos físicos da fase. A Figura 2.5 mostra uma textura típica de um SmF.

2.4.5 Esmético G

A fase esmética G (SmG) exibe um alto grau de ordem, caracterizada por uma estrutura em camadas com moléculas inclinadas em relação à normal da camada e com ordem posicional triclínica dentro das camadas. Isso significa que, dentro de cada camada, as moléculas estão dispostas em uma rede cristalina com simetria triclínica, a mais baixa das sete classes de cristal. A fase SmG tem ordem posicional de longo alcance, ao contrário das outras fases esméticas, que tem ordem posicional de curto alcance. A combinação da inclinação molecular e da ordem triclínica resulta em uma fase altamente ordenada e viscosa. Devido à sua alta ordem, a fase SmG exibe um comportamento semelhante ao de um cristal em duas dimensões. As propriedades da fase SmG são altamente anisotrópicas, o que significa que suas propriedades dependem da direção em que são medidas. A fase SmG é tipicamente observada em temperaturas mais baixas do que as fases SmA, SmC, SmI e SmF, refletindo seu alto grau de ordem. Figura 2.5 mostra uma textura típica de um SmG.

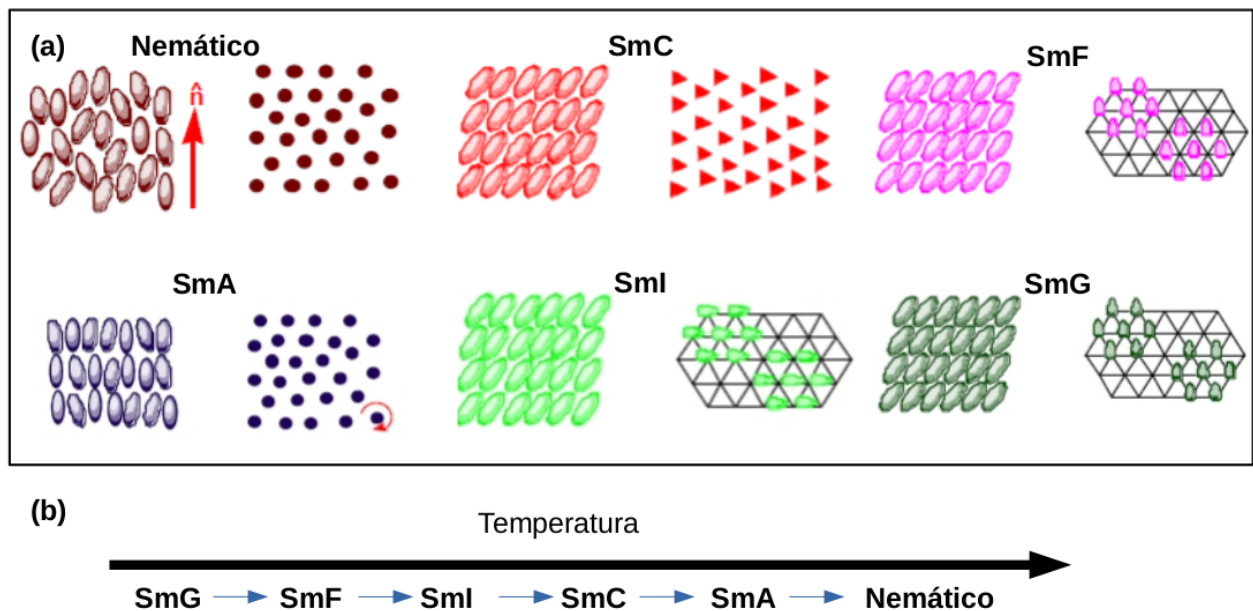


Figura 2.4: Figura (a): Esquema representando a orientação média de cada uma das fases esméticas e nemáticas visto pelo lado (figura da esquerda) e por cima (figura da direita). Na Figura (b) é mostrado a ocorrência de cada fase em ordem crescente de temperatura.

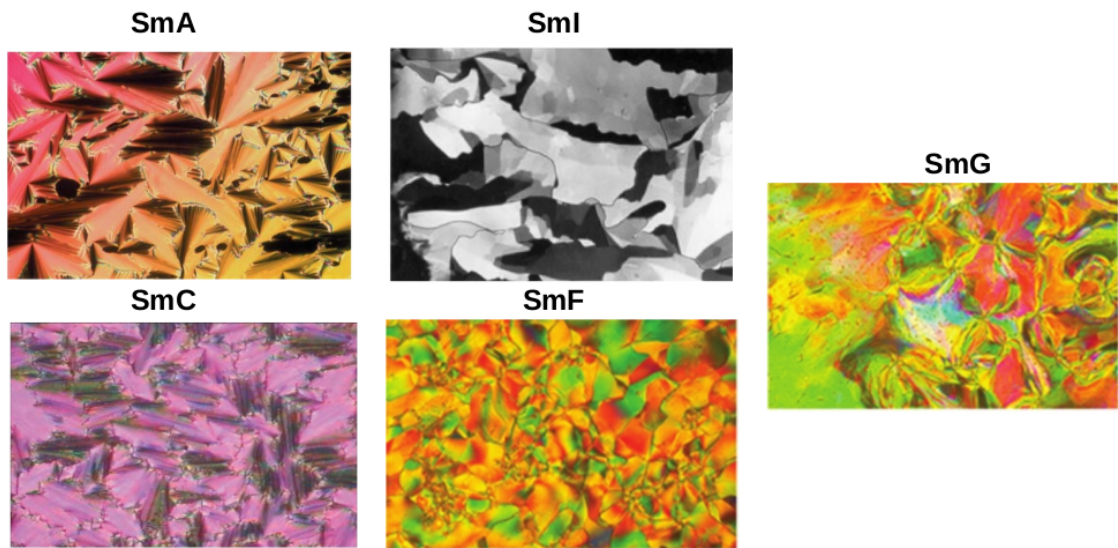


Figura 2.5: Texturas típicas das fases esmécticas estudadas neste trabalho. A textura do tipo *fan* para o SmA, a *broken fan* para o SmC, a textura mosaico do SmI e texturas do SmG e SmF. Figuras do SmA e SmC adaptadas de [8]. Textura do SmI reproduzida de [39]. Texturas do SmF e SmG reproduzidas de [40].

Testes para Escolha da Arquitetura da Rede Neural

3.1 Introdução

Recentemente, redes neurais convolutivas foram utilizadas como preditores de propriedades físicas em texturas de cristais líquidos por Sigaki et al. [20]. Entre as tarefas estudadas, estão a identificação do passo de um colestérico através de imagens geradas por simulações computacionais e a obtenção da temperatura através de imagens de POM de amostras do cristal líquido nemático E7. Tendo como base a arquitetura de Sigaki et al. [20], estudamos o desempenho desta rede e de uma U-Net, representada pelo diagrama da figura 3.1, para as mesmas tarefas de classificação aplicadas, utilizando o mesmo banco de dados. De acordo com a referência, para os bancos de dados utilizados e para as tarefas de classificação escolhidas, redes convolucionais se beneficiam de um maior número de convoluções somente até certo ponto. Com base nesta observação e constatando que U-Nets são redes puramente convolucionais com concatenações, comparamos ambas, tendo nossa U-Net um número de convoluções que excede o recomendado pelo artigo.

Construímos a U-Net de tal forma que a única diferença entre ela e a referência seja o número de convoluções (e a presença de concatenações). Mantivemos o mesmo número de neurônios (e mesmo tamanho dos *kernels*) nas convoluções e nas camadas densas, com o intuito de investigar se concatenações impactam positivamente o aprendizado, apesar da rede exceder o número de convoluções recomendadas. Todos os treinamentos foram realizados com algoritmos determinísticos, codificados em PyTorch [41], com a semente dos processos aleatórios fixada. Os resultados dos testes são apresentados ao longo deste capítulo.

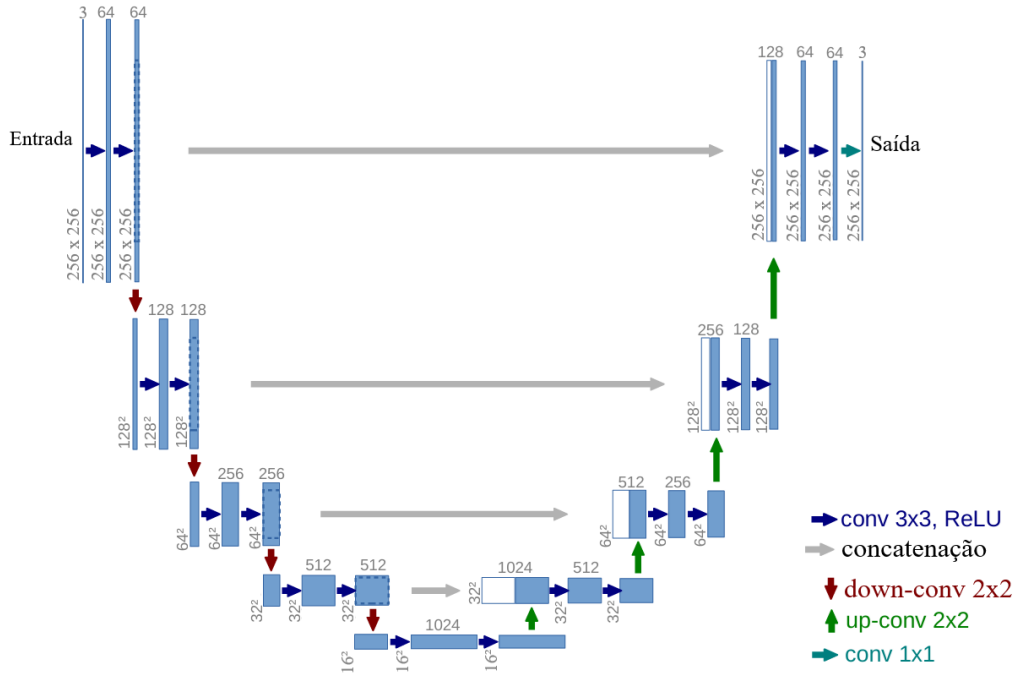


Figura 3.1: Imagem de uma U-Net como descrito pela referência [42], traduzido para o português.

3.2 Teste de Regressão

Neste teste, colocamos as quatro arquiteturas sob a tarefa de classificação de temperatura. As texturas experimentais utilizadas, cristais líquidos E7, foram obtidas por Sigaki et al. [20] por meio de microscopia óptica polarizada. Cada amostra consiste em um capilar retangular ($300 \mu\text{m} \times 4 \text{ mm}$) preenchido com a mistura E7. As amostras foram colocadas sob um sistema de microscópio óptico polarizado acoplado a um controlador de temperatura, que incrementava a temperatura das amostras por passos igualmente espaçados até uma temperatura máxima. Ao final, temos um banco de dados de imagens de texturas de cristais líquidos com temperaturas variando de 40.00 a 67.10°C . Todas as imagens possuem resolução de 2047×1532 pixels (com 24 bits por pixel, RGB) e foram redimensionadas para 510×511 pixels.

No estudo de Sigaki et al., a rede neural, treinando em um banco de dados com 10980 amostras, atingiu bons resultados sob a métrica de coeficiente de determinação (R^2) [43]. No nosso caso, fizemos a mudança de reduzir consideravelmente o número de amostras do banco de dados, de 10980 para 412 amostras, com o intuito de observar o desempenho da U-Net e da rede da referência, para uma mesma configuração de treino, fixa, sob a mesma métrica. Durante o treino, utilizamos transformações aleatórias de inversão horizontal e vertical das imagens, com probabilidade de 50%, para aumentar a variância dos dados, com intuito de prevenir ajuste excessivo nos dois casos. Diferente do artigo original, utilizamos imagens

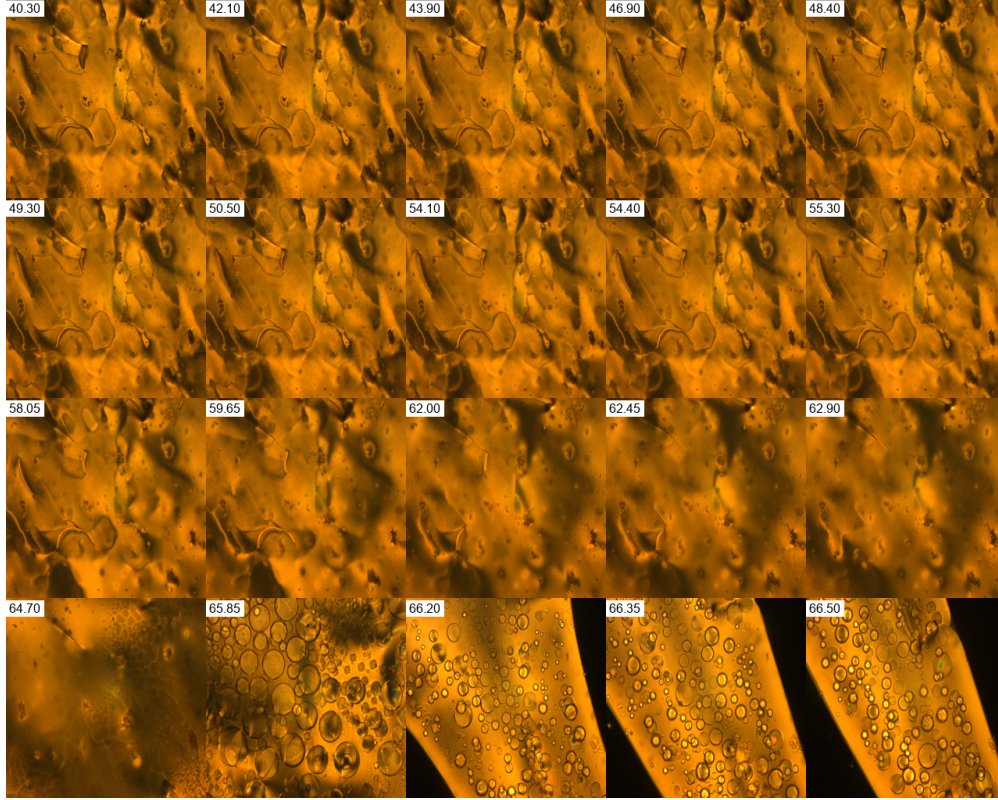


Figura 3.2: Imagens do banco de dados. A temperatura é indicada no canto superior esquerdo de cada amostra, em °C.

coloridas.

Primeiramente, testamos a rede da referência em duas versões: sua versão original; e uma versão modificada. A única diferença entre as duas é que a rede modificada possui normalização em lote (BN¹) [44] a cada duas convoluções. Depois, testamos a rede U-Net em, também, duas versões: uma sem normalização; e outra com BN a cada duas convoluções.

Assim como na referência, treinamos com validação cruzada, garantindo que a rede observa todo o banco de dados. Usamos 5 partições para a validação cruzada, sendo uma parte destinada à validação e quatro partes destinadas ao treino, traduzindo-se em 80% de amostras para treino e 20% para validação. Dada a tarefa de regressão, o modelo foi treinado tendo MSE como função de perda. Com as configurações de treino descritas, mensuramos, para as quatro arquiteturas descritas, o coeficiente de determinação médio, resultado da média dos 5 treinos da validação cruzada, ao longo do treino. Este coeficiente indica a correlação entre as previsões do modelo e o banco de dados. Com valores variando de $(-\infty, 1]$, em que 0.5 indica uma correlação moderada — isto é, o modelo é capaz de explicar aproximadamente 50% da variabilidade dos dados —, enquanto valores próximos a 1 indicam uma correspondência quase perfeita entre previsões e valores reais. Por outro lado, valores próximos ou inferiores a zero sugerem um modelo incapaz de capturar qualquer relação sig-

¹do inglês *Batch Normalization*

nificativa nos dados, equiparando-se, na prática, a prever a média do conjunto de dados ou mesmo a desempenhos piores.

Adicionalmente, optamos por limitar a visualização dos gráficos à faixa $y \in [-1, 1]$. Embora o limite inferior teórico do coeficiente de determinação seja $-\infty$, valores significativamente inferiores a -1 indicam desempenhos extremamente ruins — isto é, modelos cujas previsões são piores do que simplesmente prever a média —, sendo considerados, neste contexto, resultados degenerados. Assim, valores de $R^2 < -1$ foram representados como estando abaixo do limite inferior definido, sendo indicados explicitamente nos gráficos. Esta escolha visa facilitar a interpretação visual dos resultados, uma vez que tais valores extremos não trazem informação adicional relevante além da constatação do fracasso do modelo.

Para os testes da rede referência, observamos queda de performance sob banco de dados reduzido, como pode-se constatar pela figura 3.3.

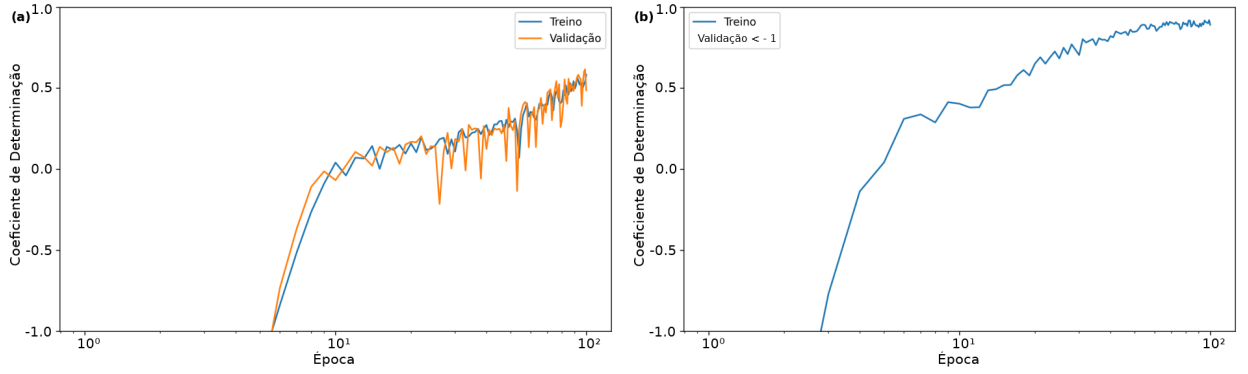


Figura 3.3: Figura (a): Coeficiente de Determinação Médio versus épocas da rede da referência (não modificada). Na figura (b): Coeficiente de Determinação Médio versus épocas da rede da referência (normalizada).

Já no treino da rede U-Net, observamos aumento de performance significativa, especialmente na arquitetura normalizada, como se pode ver na figura 3.4.

Com base nos resultados das quatro arquiteturas, constatamos que, sob exatamente a mesma configuração de treino, a concatenação de convoluções proporciona melhor convergência na validação para o problema de regressão.

Diferente da arquitetura da referência, a arquitetura U-Net é bastante dependente de normalizações. Sob a mudança de arquitetura para BN a cada duas convoluções, a rede U-Net performou bem e alcançou métrica R^2 satisfatória na validação. Como performamos treinos de validação-cruzada, sendo as métricas resultantes médias das partes, os resultados também indicam que, para o problema de regressão, a rede não satura [45] sob mudança dos dados iniciais.

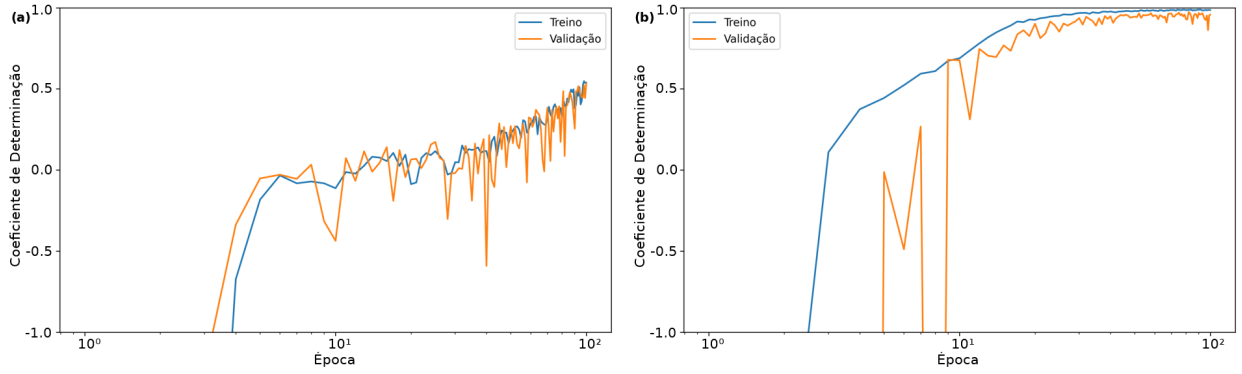


Figura 3.4: Figura (a): Coeficiente de Determinação Médio versus épocas da rede U-Net (não normalizada). Na figura (b): Coeficiente de Determinação Médio versus épocas da rede U-Net (normalizada).

3.3 Teste de Classificação Categórica

Nesta seção, apresentamos os testes realizados para o problema de classificação categórica. Utilizamos as mesmas texturas colestéricas de Sigaki et al. [20], obtidas a partir de simulações numéricas baseadas no modelo de Landau-de Gennes [46], fundamentado na teoria elástica contínua. As simulações foram realizadas em uma malha uniforme de 200×200 pontos, com condições de contorno periódicas, e parâmetros ajustados para replicar as propriedades do cristal líquido 5CB [8]. As texturas ópticas correspondentes foram calculadas utilizando o método de Jones 2×2 [47], resultando em imagens preto-e-branco visualmente semelhantes às obtidas experimentalmente [20]. O conjunto de dados contempla valores de passo $\eta \in \{15, 17, 19, \dots, 29\}$ nm. Na figura 3.5, apresentamos amostras aleatórias representativas desse banco de dados.

Para este banco de dados, composto por 8000 amostras, avaliamos o desempenho das redes na tarefa de classificação do passo η do cristal líquido, utilizando a métrica de acurácia [48]. Esta métrica é calculada como a razão entre o número de previsões corretas e o total de amostras avaliadas, tanto no conjunto de treino quanto no de validação. Assim como no teste apresentado na seção anterior, o número de amostras do banco de dados foi significativamente reduzido (para 520) para a realização deste experimento. Reservamos 10% do banco de dados exclusivamente para o cálculo das matrizes de confusão.

As redes foram treinadas sob exatamente as mesmas configurações do teste anterior (seção 3.2), incluindo o uso de validação cruzada com 5 subdivisões. A única exceção foi a função de perda adotada: neste caso, utilizou-se a entropia cruzada (CE^2) [49] como função de perda, adequada para problemas de classificação.

Nas figuras 3.6 e 3.7, apresentamos os resultados obtidos pela rede original e pela arquitetura U-Net, respectivamente, segundo a métrica de acurácia.

²do inglês *Cross Entropy*

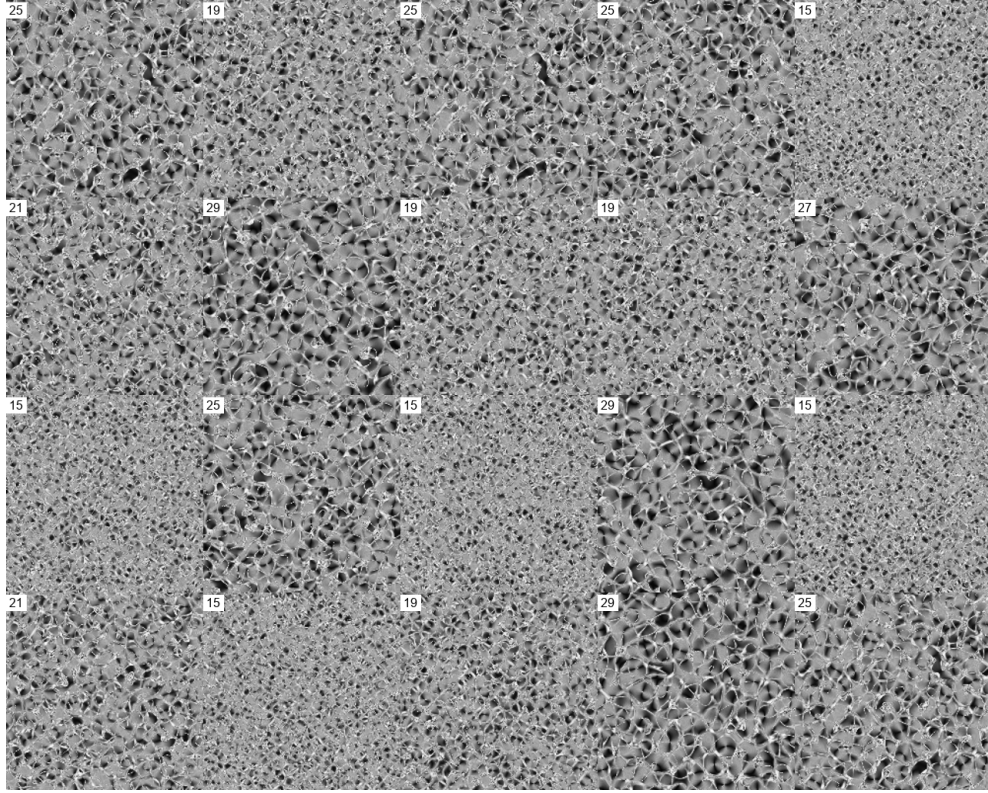


Figura 3.5: Imagens do banco de dados. O passo η é indicado no canto superior esquerdo de cada amostra.

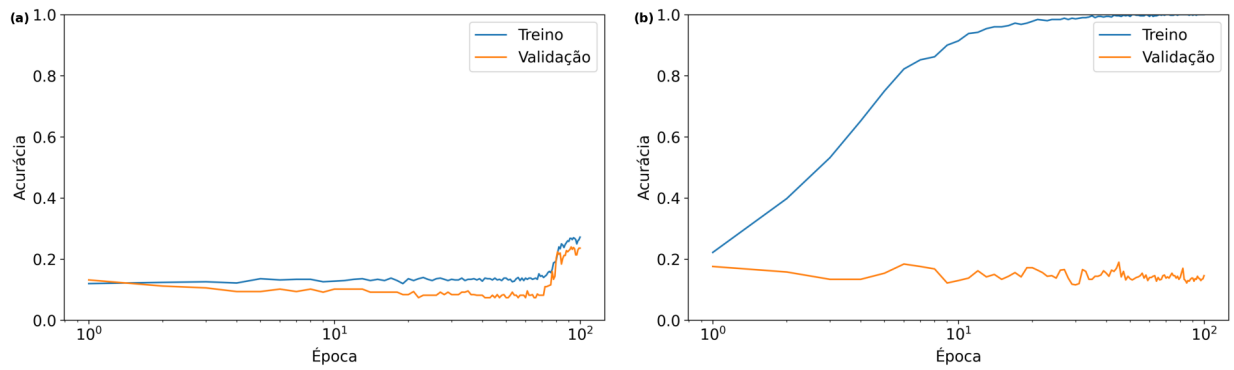


Figura 3.6: Figura (a): Acurácia Média versus épocas da rede da referência (não modificada). Na figura (b): Acurácia Média versus épocas da rede da referência (normalizada).

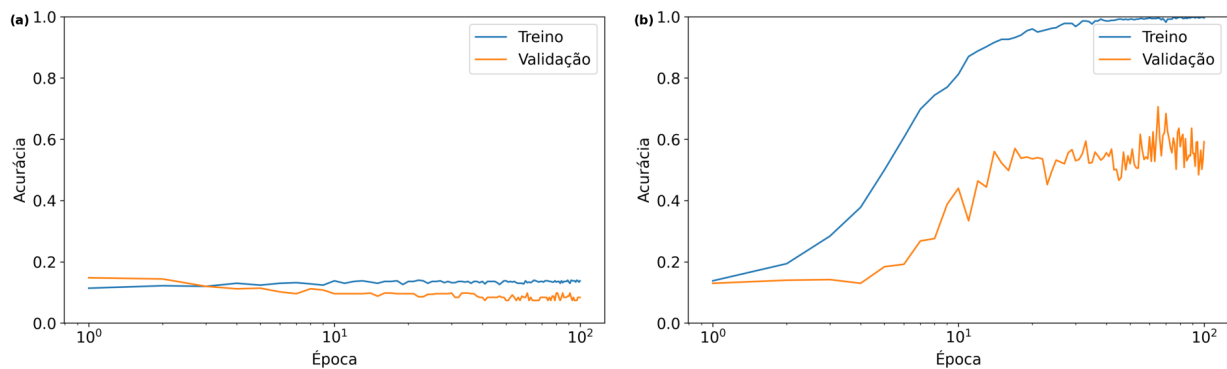


Figura 3.7: Figura (a): Acurácia Média versus épocas da rede U-Net (não normalizada). Na figura (b): Acurácia Média versus épocas da rede U-Net (normalizada).

Os resultados obtidos foram considerados insatisfatórios. Hipotetizamos que a necessidade de alterar a função de perda para a tarefa de classificação — no caso, a adoção de CE — comprometeu a capacidade das redes em aprender de forma eficaz quando treinadas sobre um banco de dados significativamente reduzido, tanto no caso da rede da referência quanto da U-Net. Ressalta-se, contudo, que, mesmo nesse cenário restrito, a U-Net apresentou acurácias de validação superiores às da rede original.

Observa-se também uma diferença sistemática entre as versões normalizadas e não normalizadas das arquiteturas: a U-Net normalizada alcançou melhor desempenho, enquanto, no caso da rede convolucional da referência, a versão não normalizada apresentou resultados superiores. Cabe observar que estes resultados também são influenciados pela configuração de treino, de modo que concluímos estes resultados enquanto relativos à configuração de treino adotada.

Analisando individualmente as partições da validação cruzada, observamos que a rede convolucional não normalizada, quando considerada sua melhor partição, atinge desempenho comparável ao da U-Net normalizada em sua melhor partição, no contexto da tarefa categórica. A figura 3.8 ilustra este resultado, apresentando lado a lado as matrizes de confusão referentes às melhores partições de cada arquitetura.

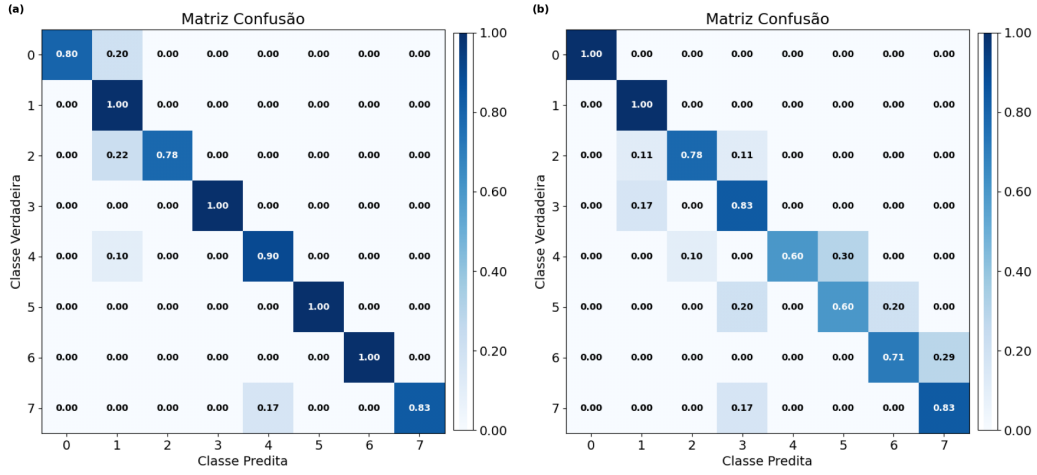


Figura 3.8: Figura (a): matriz confusão da U-Net normalizada da melhor partição. Figura (b): matriz confusão da rede convolutiva não normalizada da melhor partição.

Dessa forma, observamos que, mesmo sob um banco de dados reduzido, a U-Net apresentou desempenho médio superior ao da rede convolutiva. Contudo, conforme ilustrado na figura 3.8, ambas as arquiteturas foram capazes de atingir resultados satisfatórios em suas melhores partições de validação. Ainda assim, a U-Net manteve vantagem consistente, evidenciando melhor capacidade de ajuste sob as condições restritas de treino analisadas.

3.4 Escolha da Arquitetura do Modelo de Difusão

No problema de regressão, a arquitetura da referência apresentou desempenho inferior à U-Net quando treinada sobre um conjunto de dados reduzido. Já na tarefa de classificação dos passos, ambas as redes obtiveram resultados insatisfatórios. Como o modelo de difusão caracteriza-se como um problema de regressão [15], esses resultados motivaram a escolha da U-Net como base para a arquitetura desenvolvida neste trabalho.

Inspirados pela arquitetura de DDPM [16], incorporamos, à nossa arquitetura, mecanismos de atenção [50] e camadas de *embeddings* sinusoidais (temporais e condicionais, responsáveis pelos parâmetros t , do processo, e y , da classe, opcional).

O mecanismo de atenção permite que diferentes regiões da entrada interajam entre si, ponderando a importância relativa de cada posição no espaço. Em redes neurais aplicadas a imagens, a atenção possibilita que cada pixel incorpore informações globais, o que é especialmente útil em modelos de difusão, onde a estrutura espacial está degradada por ruído. De forma geral, a atenção é implementada por meio de operações entre projeções chamadas de consultas, chaves e valores, que determinam como as informações devem ser agregadas em cada posição. Embora a ideia básica seja simples, sua implementação eficiente envolve diversas sutilezas, especialmente no contexto de atenção multi-cabeça e convolutiva. Para uma

descrição detalhada do funcionamento do mecanismo de atenção, recomenda-se a leitura da referência [50], onde o método foi originalmente proposto no contexto de transformadores.

Embeddings sinusoidais são utilizados para codificar a informação de t ou y em modelos de difusão. Essa codificação transforma um escalar discreto, correspondente ao passo de difusão ou à classe, em um tensor contínuo que é passado pela rede neural. A ideia é fornecer ao modelo uma representação suave e diferenciável daquele escalar, permitindo que ele aprenda a condicionar suas previsões à etapa atual t do processo de difusão e, opcionalmente, ao condicional y .

A forma do *embedding* sinusoidal é [50]:

$$\gamma(t)_i = \begin{cases} \sin\left(t/10000^{\frac{2i}{d}}\right), & \text{se } i \text{ par} \\ \cos\left(t/10000^{\frac{2i}{d}}\right), & \text{se } i \text{ ímpar} \end{cases} \quad (3.1)$$

Em que $t \in \mathbb{N}$ é o passo de tempo da difusão; $i \in \{0, 1, \dots, d-1\}$ é o índice da dimensão do *embedding*; e d é a dimensão do *embedding*. Essa construção gera padrões oscilatórios com diferentes frequências, de modo que cada dimensão i representa uma escala temporal distinta. Assim, o modelo consegue representar tanto relações de curto quanto de longo alcance do processo de difusão.

A figura 3.9 apresenta o diagrama da arquitetura proposta, adaptado de [51] e ajustado para refletir fielmente a empregada neste trabalho. Os blocos de atenção foram inseridos apenas nas camadas internas da U-Net, onde as resoluções espaciais são as menores. A estrutura da nossa U-Net segue a ilustrada na figura 3.1. A seguir, descrevemos, em detalhes, a estrutura de nossa arquitetura escolhida.

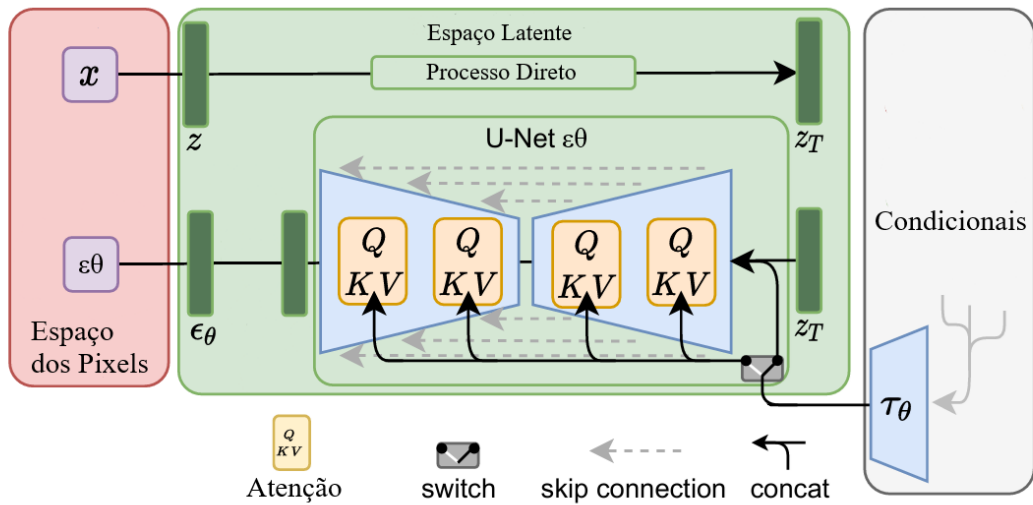


Figura 3.9: Figura de [51], com pequenas simplificações para este trabalho, traduzida para o português. Esta figura adaptada é a arquitetura utilizada neste trabalho.

3.5 Arquitetura do Modelo de Difusão

Nossa arquitetura foi construída em *PyTorch*³. A U-Net implementada possui uma estrutura em forma de “U” (figura 3.1) composta por blocos, que seguem, contínua e respectivamente, dois caminhos: O de contração, que reduz progressivamente a resolução espacial da imagem e aumenta o número de canais; e o de expansão, que reconstrói a resolução da imagem a partir das representações do caminho de contração, utilizando conexões de atalho.

Cada bloco da rede é composto por convoluções, normalizações, funções de ativação, *embeddings* sinusoidais temporais e condicionais, e, opcionalmente, mecanismos de atenção. A entrada da rede é uma imagem ruidosa x_t , corrompida pelo processo de difusão, acompanhada de dois parâmetros: t , representando o instante temporal do processo, e y , um índice inteiro da classe (usado no treinamento condicional), que pode ser ativado ou desativado para o treinamento. O fluxo de dados segue as seguintes etapas: Primeiro, a imagem de entrada x_t passa por uma convolução que projeta os dados para um espaço latente com mais canais, iniciando o processo de codificação. Depois, passa-se por uma U-Net: A imagem é processada por uma sequência de blocos convolucionais de contração. Em cada bloco, uma convolução reduz a resolução espacial pela metade e aumenta o número de canais. Nos blocos, os *embeddings* de tempo t e classe y são computados por módulos separados (baseados em *embeddings* sinusoidais aprendíveis) e somados ao mapa de ativação. No código, as ativações de cada bloco de contração são armazenadas e, posteriormente, utilizadas para concatenação com os blocos de expansão correspondentes. Após o caminho de contração, inicia-se o de expansão, que segue, também, uma sequência de blocos: Em cada bloco, uma convolução transposta, de expansão, dobra a resolução espacial e reduz o número de canais. As ativações armazenadas durante a contração são recuperadas e concatenadas com a entrada do respectivo bloco de expansão. Essa concatenação permite que detalhes de baixa resolução preservados anteriormente sejam reutilizados na reconstrução. Após passar por todos os blocos de expansão, a saída da rede (já com a mesma resolução espacial da entrada), é processada por uma sequência de convoluções, normalizações e ativações, resultando em uma imagem com mesmo número de canais e mesmas dimensões da entrada. Esta imagem, seguindo a teoria de DDPM, é a previsão do ruído $\epsilon_\theta(t, y)$.

A figura 3.9 resume a estrutura geral da arquitetura utilizada. O modelo final foi treinado em imagens 256×256 , com e sem condicionamento y , mantendo um custo computacional acessível e resultados satisfatórios na geração de imagens realistas de cristais líquidos.

Esta arquitetura que propomos é a continuação de uma primeira, a qual descartamos. A estrutura da versão anterior era praticamente idêntica à descrita nesta seção, com exceção do formato adotado nas camadas de atenção. Naquele momento, como ainda não pretendíamos trabalhar com imagens coloridas, utilizamos uma atenção de única cabeça, aplicada apenas

³*PyTorch* é uma biblioteca do *Python* voltada ao aprendizado de máquina, com suporte a redes neurais profundas e computação numérica em GPU.

na direção horizontal da imagem. Em outras palavras, a atenção era restrita às linhas: para uma imagem de dimensões $w \times h$, o mecanismo operava apenas sobre h .

Posteriormente, adotamos a atenção aplicada em cada pixel, fazendo com que a dimensão considerada passasse a ser $w \times h$. Nesse sentido, a arquitetura anterior era mais leve em termos computacionais, mas não se mostrava adequada para lidar com imagens coloridas, que constituíam o foco do nosso trabalho.

Assim, a única diferença entre a arquitetura aqui apresentada e a anterior reside na forma como as atenções foram implementadas. Os resultados obtidos com aquela versão preliminar, bem como alguns comentários adicionais, encontram-se registrados no Apêndice [A](#).

4.1 Introdução

Neste capítulo, apresentamos e discutimos os principais resultados obtidos ao longo deste trabalho, aplicando DDPM ao problema de geração de imagens de cristais líquidos. As imagens utilizadas foram obtidas experimentalmente por meio de microscopia óptica polarizada (POM), constituindo um banco de dados composto por texturas ópticas com ampla variação visual.

Para a geração das imagens, implementamos duas abordagens distintas: uma não condicional, em que o modelo aprende a gerar amostras do banco de dados sem receber condicionais como entrada; e uma condicional: em que o modelo recebe condicionais de classe.

Ambas as versões do DDPM foram desenvolvidas em PyTorch, visando tanto o aprendizado detalhado do funcionamento interno desses modelos quanto a flexibilidade para adaptações específicas necessárias ao banco de dados experimental.

O objetivo desta etapa da pesquisa é avaliar a capacidade dos modelos — condicionais e não condicionais — em gerar imagens de cristais líquidos visualmente coerentes e realistas, comparáveis às imagens reais obtidas por POM. Para isso, as amostras geradas serão analisadas qualitativamente, por inspeção visual, e quantitativamente, por meio da métrica MSE, quando pertinente.

A motivação para esta investigação reside na busca por técnicas generativas capazes de sintetizar imagens de materiais complexos, como cristais líquidos, visando aplicações em modelagem computacional, aumento de base de dados e análise automatizada de texturas.

Nas seções seguintes, detalhamos os resultados obtidos com cada abordagem, destacando suas vantagens, limitações e o potencial de generalização observado em nosso conjunto ex-

perimental.

4.2 Treino Condicional

O banco de dados escolhido para o treino condicional, exposto na figura 4.1, consiste em imagens de cristais líquidos dos tipos: nemático, SmA (tipos A e B), SmC, SmF, SmG; e SmI.

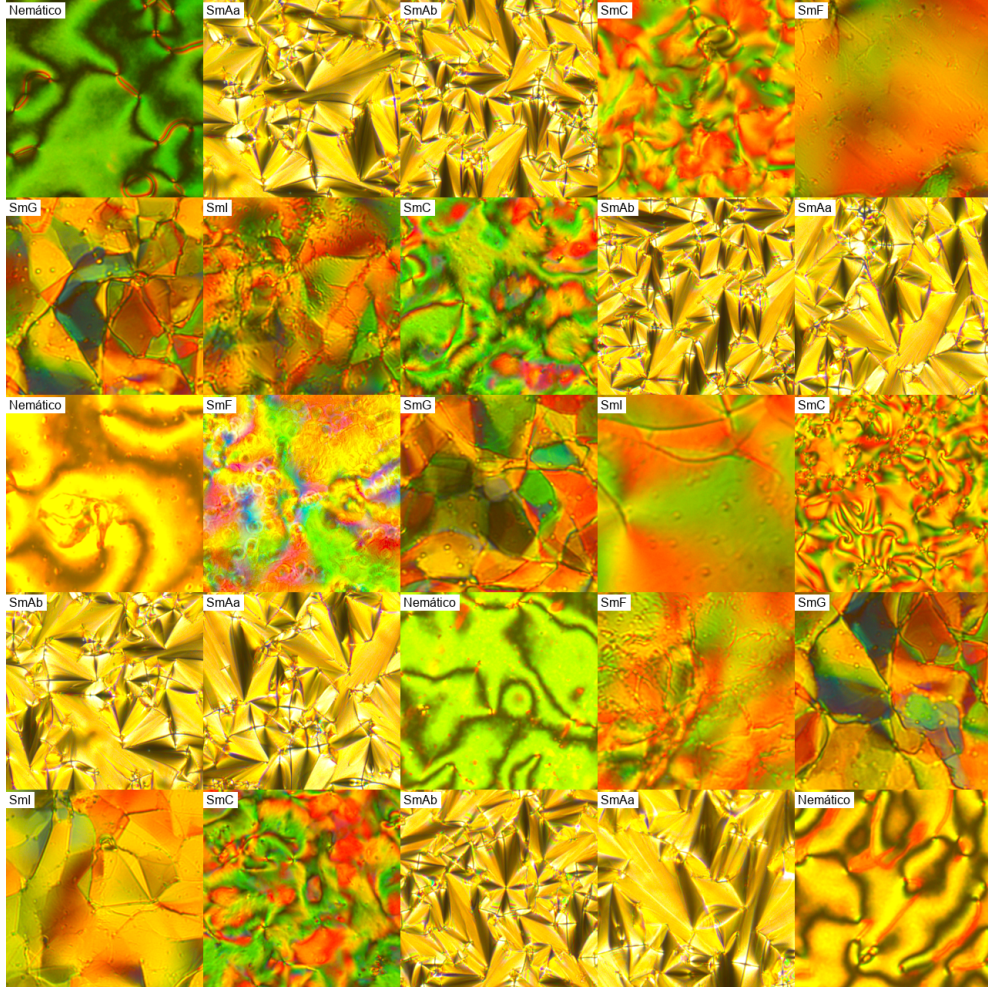


Figura 4.1: Amostras do banco de dados utilizado para o treinamento da rede neural. O tipo está indicado no canto superior esquerdo de cada amostra.

Cada classe recebe um índice de 1 a N , com N igual ao número de classes, que passa pela rede neural da mesma maneira que t . Com 11687 imagens no banco de dados, nosso modelo treinou por aproximadamente 500 épocas, ou, $2 \cdot 10^5$ passos, onde cada passo contém 32 amostras. Treinamos nosso modelo com imagens de resolução 256×256 , sob função de perda MSE. Fixamos a taxa de aprendizado em 10^{-4} para todos os treinos. Observamos, como indicam os resultados da figura 4.2, que nossa rede DDPM não ajusta excessivamente. Os

resultados deste treino condicional para cada classe são expostos e discutidos nas subseções a seguir.

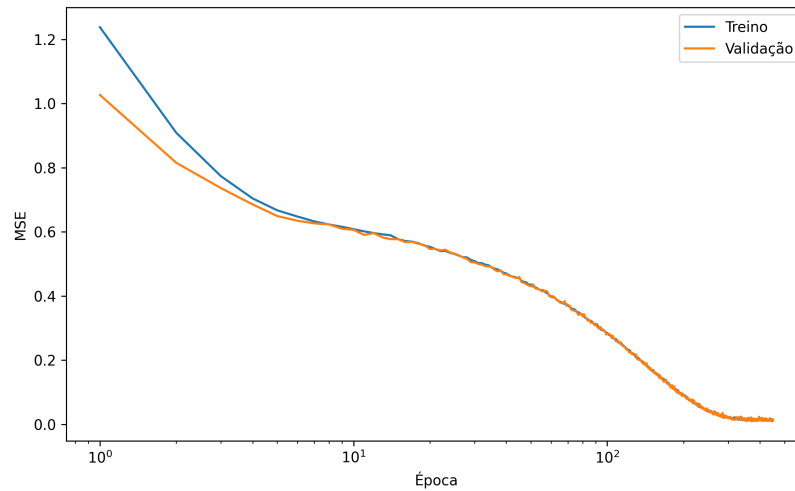


Figura 4.2: MSE versus Épocas. Treino condicional, realizado sob um banco de dados com 11687 imagens.

4.2.1 Nemático

Observando o banco de dados para esta classe, constatamos que as imagens possuem muita diferença estrutural entre si, com grande variância de cores entre as imagens. Na figura 4.3 abaixo, temos um exemplo das melhores amostras geradas. Como o ruído do banco de dados é considerável, e o mesmo também não é extenso, consideramos estes fatores como principais causas para baixa qualidade de imagens amostradas desta classe (figura 4.4).

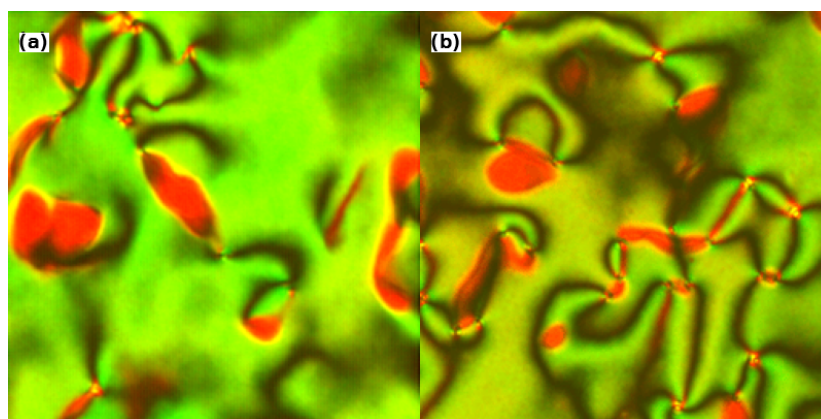


Figura 4.3: Figura (a): Imagem amostrada do modelo de difusão. Figura (b): Imagem real, do banco de dados.

A figura 4.4 mostra 9 amostras aleatórias para a classe de materiais nemáticos. Pela figura, nota-se também que, em nosso banco de dados, as imagens possuem alta variância.

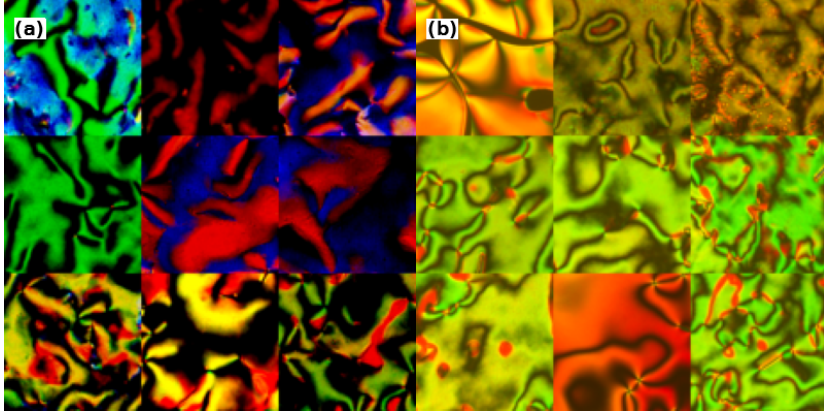


Figura 4.4: Figura (a): Amostras do modelo de difusão. Figura (b): Imagens do banco de dados.

Observamos, também, como a escolha da forma de β_t pode influenciar o processo generativo, visto que, assim como no processo direto (figura 1.1 do capítulo 1), o processo reverso também é majoritariamente ruidoso, e observamos a classe de forma nítida somente nas etapas finais da remoção de ruído. Embora a escolha de β_t linear seja satisfatória, outras escolhas como β_t cossinoidal [52] melhoram a qualidade das amostras. Na figura 4.5 abaixo, pode-se observar o processo generativo como um todo, feito pela rede neural, para um material nemático.



Figura 4.5: Figura contendo 20 imagens x_t geradas por um único processo reverso, igualmente separadas, e aclopadas de forma contínua, da esquerda para direita. No canto superior esquerdo, temos x_{T-1} ; e no canto inferior direito, x_0 .

4.2.2 SmA

As imagens, do banco de dados, da classe SmA possuem predominância de uma única cor, como pode-se notar na figura 4.1, e observamos que a rede neural tende a distribuir cores da mesma maneira ao longo de todas as classes, o que não é o comportamento desejado. Ainda assim, as amostras são coesas e lembram as imagens da classe SmA. Observamos ao longo de nossos testes que a rede neural converge mais rápido para esta classe no treino incondicional.

Tipo A

Na figura 4.6, pode-se observar, lado a lado, uma amostra gerada pelo modelo de difusão condicional (a) e uma amostra real do banco de dados (b), sob o treino com taxa de aprendizado reduzida.

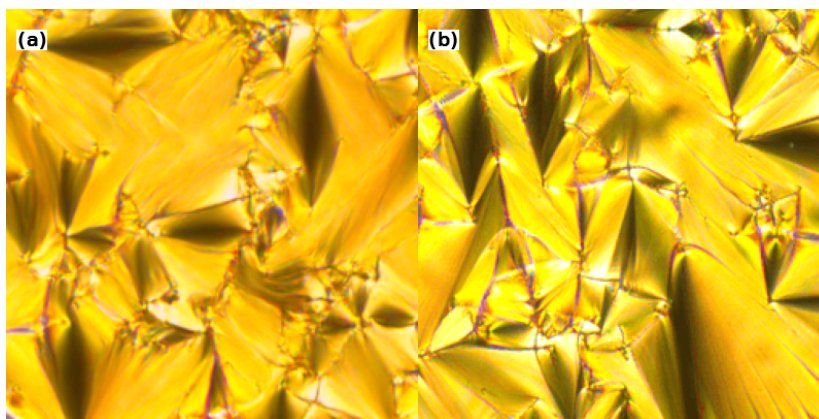


Figura 4.6: Figura (a): Imagem amostrada do modelo de difusão. Figura (b): Imagem real, do banco de dados.

Na figura 4.7 abaixo, temos um exemplo de amostras da rede para SmA treinado condicionalmente. Como mencionado, os resultados não são como o desejado pois há confusão de cores. Ainda assim, as amostras se parecem, estruturalmente, com a classe respectiva.

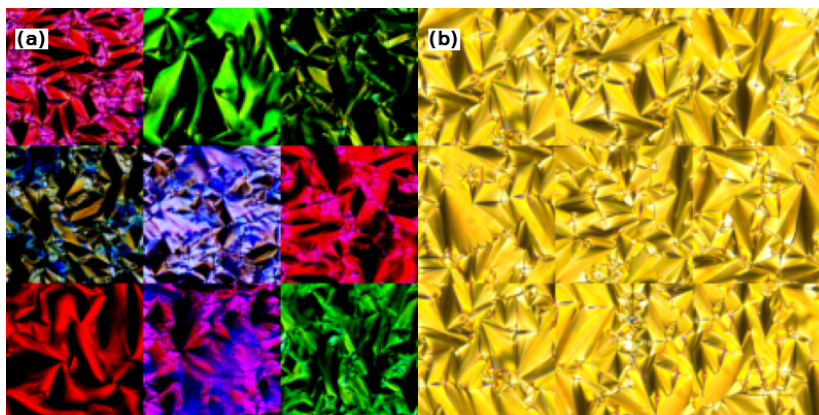


Figura 4.7: Figura (a): Amostras do modelo de difusão. Figura (b): Imagens do banco de dados.

Tipo B

Os dois tipos de SmA são muito parecidos, e possuem estrutura característica. Embora a rede condicional erre bastante as cores da classe, como pode-se observar pela figura 4.8, as formas ainda são condizentes e coesas com a classe em questão. Na figura 4.9, observa-se a clara distinção de cores entre banco de dados real e amostras, baseado no treino condicional.

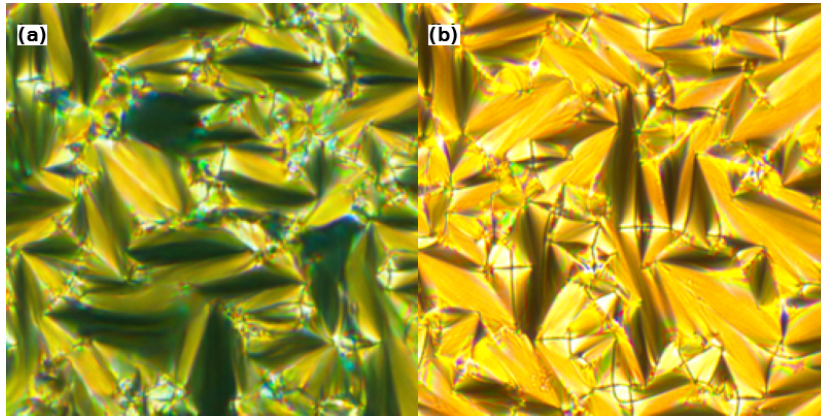


Figura 4.8: Figura (a): Imagem amostrada do modelo de difusão. Figura (b): Imagem real, do banco de dados. Diferente da amostragem da figura 4.6, o modelo usado nesta foi treinado com todas as classes.

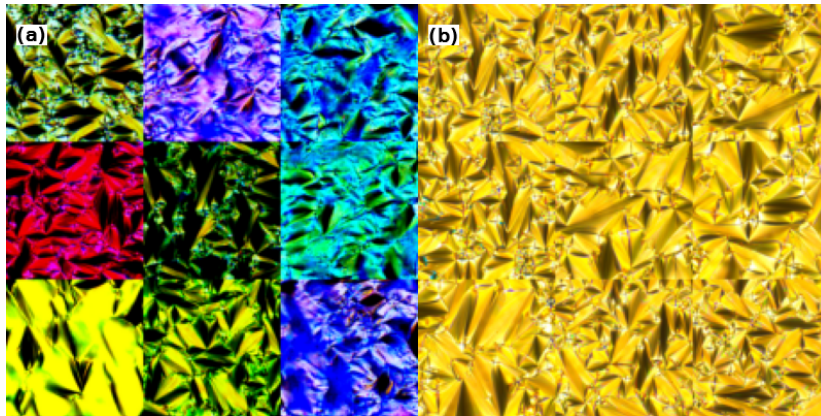


Figura 4.9: Figura (a): Amostras do modelo de difusão. Figura (b): Imagens do banco de dados.

4.2.3 SmC

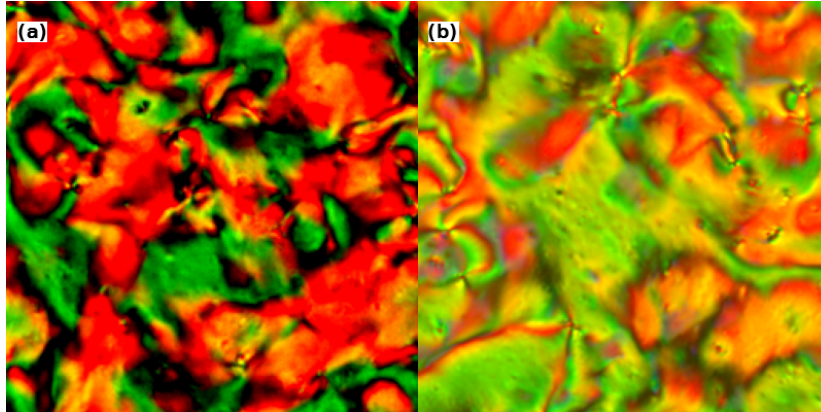


Figura 4.10: Figura (a): Imagem amostrada do modelo de difusão. Figura (b): Imagem real, do banco de dados.

Na figura 4.11 abaixo, temos 9 imagens amostradas do modelo, para a classe SmC, em comparação com 9 imagens do banco de dados real.

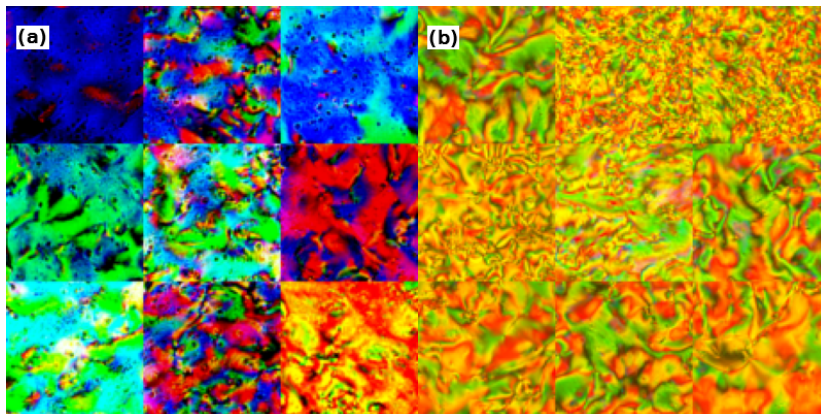


Figura 4.11: Figura (a): Amostras do modelo de difusão. Figura (b): Imagens do banco de dados.

4.2.4 SmF

Assim como para classes outras anteriormente mencionadas, a rede gera imagens com cores pouco parecidas em relação ao banco de dados, mas, quando gera uma boa amostra, desenha o material de forma bastante semelhante. Notamos, observando empiricamente o banco de dados, que muitas imagens de SmC possuem um efeito de borrão, como se a imagem estivesse embaçada. A rede desenha bem essa característica quando condicionada à classe SmF. Na figura 4.12, vemos um exemplo desse tipo de amostra.

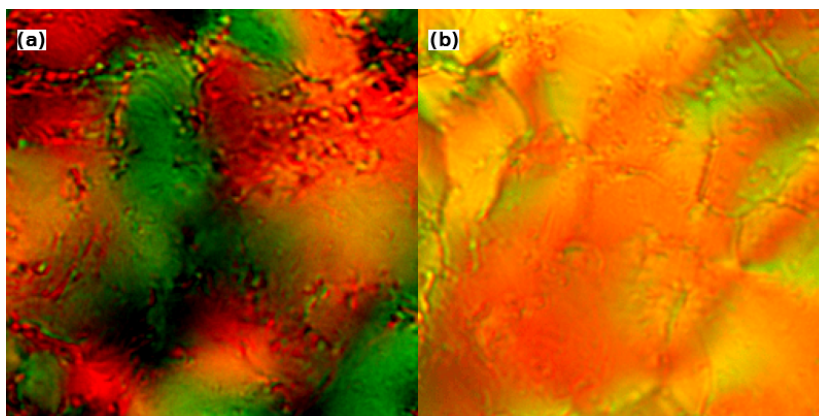


Figura 4.12: Figura (a): Imagem amostrada do modelo de difusão. Figura (b): Imagem real, do banco de dados.

Na figura 4.13 abaixo, temos 9 imagens amostradas do modelo, para a classe SmF, em comparação com 9 imagens do banco de dados real.

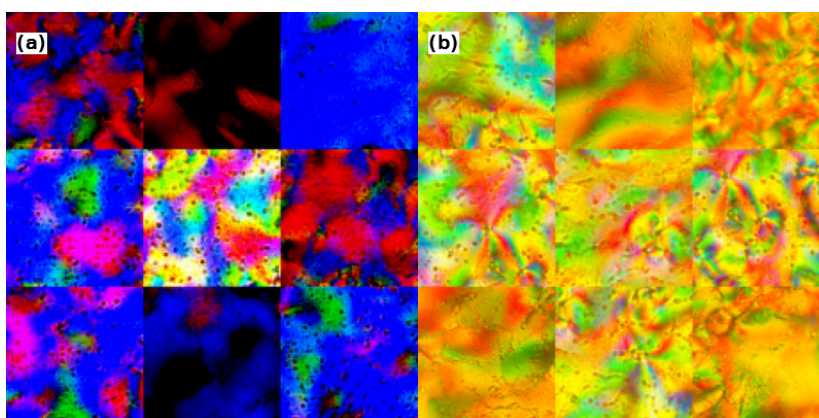


Figura 4.13: Figura (a): Amostras do modelo de difusão. Figura (b): Imagens do banco de dados.

4.2.5 SmG

Os materiais SmG são visualmente bastante distintos das outras classes. Para esta classe, a rede também apresenta confusão de cores mas desenha muito bem as características estruturais. Na figura 4.14 abaixo, temos, à esquerda, um exemplo de destaque do nosso modelo; e à direita, uma imagem do banco de dados real.

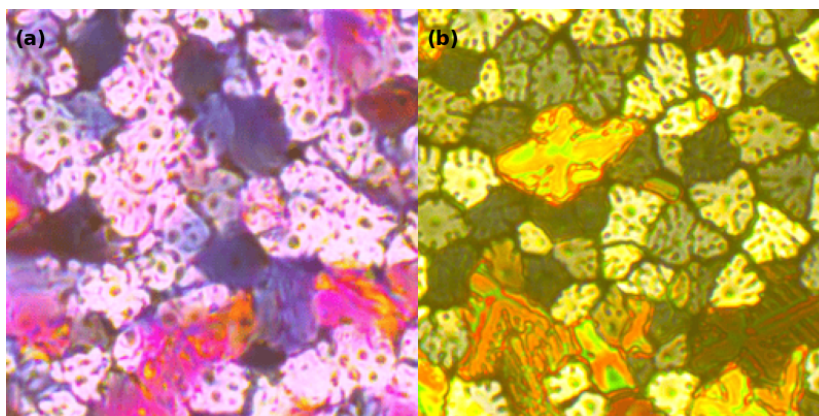


Figura 4.14: Figura (a): Imagem amostrada do modelo de difusão. Figura (b): Imagem real, do banco de dados.

Na figura 4.15 abaixo, temos 9 imagens amostradas do modelo, para a classe SmG, em comparação com 9 imagens do banco de dados real.

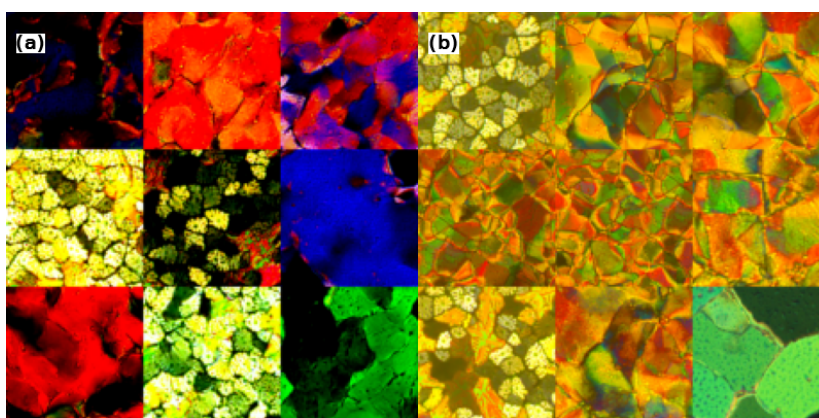


Figura 4.15: Figura (a): Amostras do modelo de difusão. Figura (b): Imagens do banco de dados.

4.2.6 SmI

As imagens do banco de dados do material SmI também apresentam efeitos de lente como os do material SmF (figura 4.12). Novamente, a alta variância de cores das imagens do banco de dados e sua extensão reduzida contribuem para a qualidade insatisfatória das amostras, como pode-se ver na figura 4.16.

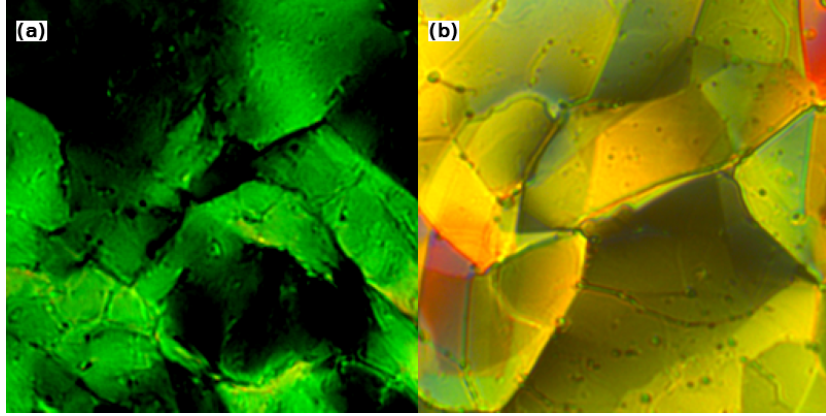


Figura 4.16: Figura (a): Imagem amostrada do modelo de difusão. Figura (b): Imagem real, do banco de dados.

Na figura 4.17 abaixo, temos 9 imagens amostradas do modelo, para a classe SmI, em comparação com 9 imagens do banco de dados real.

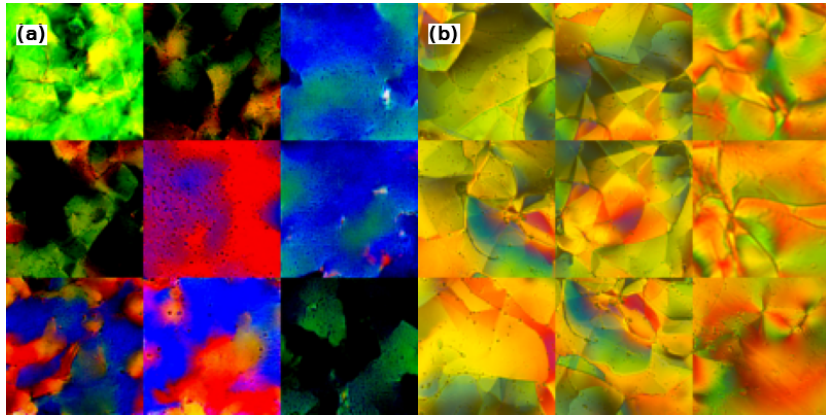


Figura 4.17: Figura (a): Amostras do modelo de difusão. Figura (b): Imagens do banco de dados.

4.3 Treino Condicionado por Campo Elétrico

Dispomos, também, de um banco de dados de imagens com 167 imagens de um cristal sob efeito de mudanças de campo elétrico. Os valores de campo, mensurados em volts, V , do banco de dados, variam de 0.0 a 20.0, todos igualmente espaçados em intervalos de 0.5. Definimos estes valores de campo como nossas classes, e treinamos condicionalmente usando resolução 128×128 , desta vez em preto-e-branco. Utilizamos exatamente as mesmas configurações de treino expostas nessa seção. Durante amostragem, extrapolamos os condicionais de classe para acrescentar os valores de campo 21, 22, e 23. Na figura 4.18 abaixo, apresentamos os resultados da amostragem sob as condições descritas.

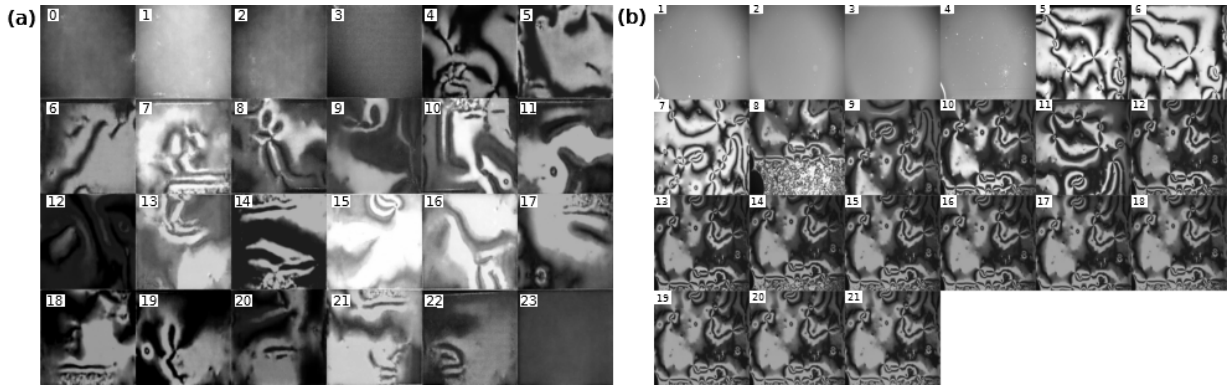


Figura 4.18: Figura (a): amostras do modelo para os valores de campo mencionados. No canto superior esquerdo de cada imagem, é indicado o condicional referente. Figura (b): amostras do banco de dados. No canto superior esquerdo de cada imagem, é indicado o valor de campo referente.

4.4 Treino Incondicional

Com o mesmo banco de dados descrito na seção anterior, treinamos incondicionalmente duas redes neurais, uma para nemático e outra para SmA. Estes testes confirmaram a capacidade de aprendizado sob banco de dados reduzido observado no capítulo 3. O algoritmo do treino é o mesmo usado no caso condicional, bem como a arquitetura da rede, desta vez inicializada com condicionais desligados. Por 500 épocas, o modelo treinou sob banco de dados reduzido em imagens de resolução 128×128 . Sob redução, os bancos de dados contém, todos, menos de 500 amostras. No caso do material SmA, devido à própria quantidade reduzida de imagens para esta classe, treinou-se com menos de 300 amostras. Assim como observado para a rede U-Net, a DDPM performou bem sob banco de dados reduzido. Na figura 4.19 vemos os resultados da rede após o treino.

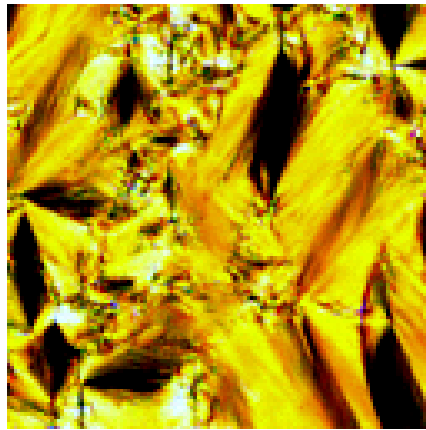


Figura 4.19: Amostra de um material SmA tipo A treinado por 500 épocas.

Notamos também que, assim como no caso condicional, há confusão de cores durante o

treino, como pode-se notar na figura 4.20 mas, no caso incondicional, ao final o modelo gera imagens fiéis ao material.

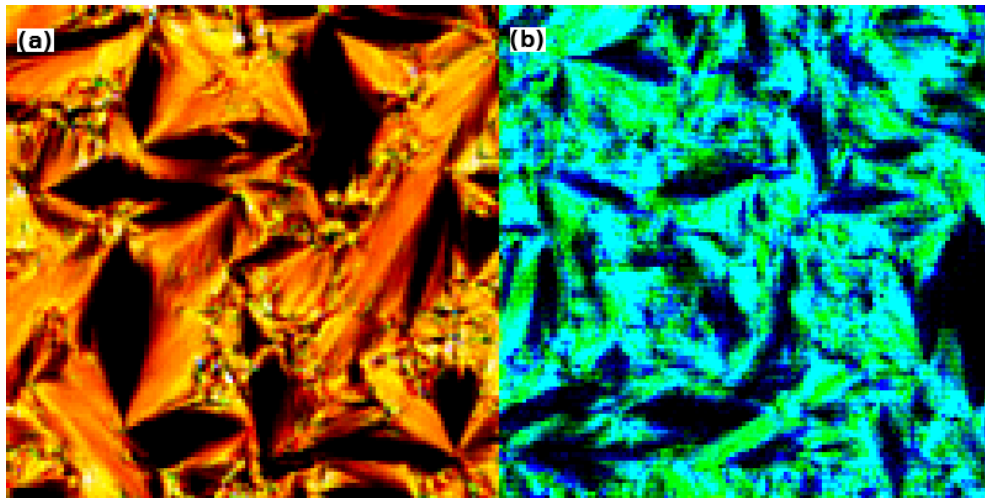


Figura 4.20: Figura (a): Amostragem incondicional de SmA em 300 épocas. Figura (b): Amostragem incondicional de SmA também em 300 épocas.

4.5 Escalabilidade

Nesta seção, apresentamos resultados de treinos incondicionais para a classe nemático, treinados com resolução 256×256 , RGB. Ao amostrar imagens do modelo, desligamos computações de gradientes e, assim, reduzimos o custo computacional da rede. Dentro de nossas limitações computacionais, podemos, neste contexto, amostrar imagens RGB de resolução 512×512 , afim de compará-las àquelas da resolução original do treino. Na figura 4.21 abaixo, comparamos, para a classe nemático, uma amostragem com resolução igual à utilizada no treino, e outra amostragem com resolução 512×512 .

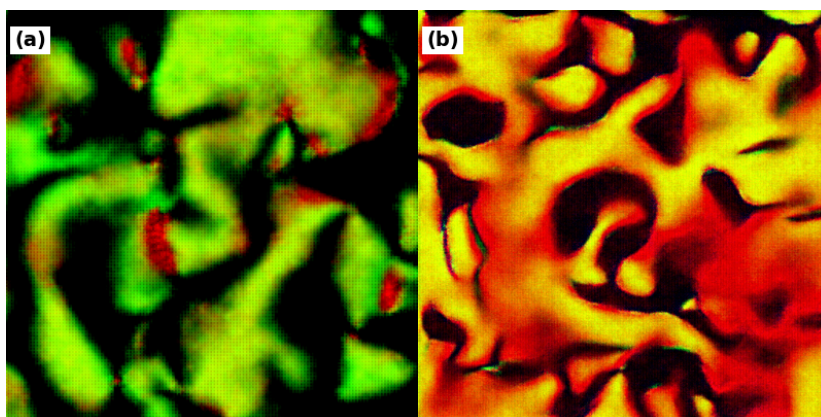


Figura 4.21: Figura (a): Amostragem de um nemático com resolução 256×256 . Figura (b): Amostragem de um nemático com resolução 512×512 .

Neste trabalho, investigamos diversos aspectos dos modelos DDPM. No capítulo 1, discutimos sua relação conceitual com Equações Diferenciais Estocásticas (EDE) e modelos baseados em energia, estabelecendo um panorama teórico inicial. Em seguida, exploramos aspectos arquiteturais relevantes, comparando o desempenho da arquitetura U-Net com redes convolucionais tradicionais baseadas em [20]. Finalmente, implementamos e treinamos uma versão customizada do modelo, aplicada a um banco de dados inédito de cristais líquidos, caracterizado por imagens altamente variadas.

Os resultados apresentados no capítulo 3 indicam que as U-Nets são arquiteturas mais adequadas para tarefas de regressão em imagens, superando redes convolucionais simples, mesmo em cenários de dados limitados. Na seção de resultados (4), demonstramos que nosso modelo customizado foi capaz de gerar amostras sintéticas de cristais líquidos consistentes com a distribuição do banco de dados real, confirmando a capacidade gerativa da abordagem DDPM. Embora a quantidade de boas amostras geradas pela rede condicional tenha sido limitada, observamos que as melhores amostras reproduziam características visuais próximas às observadas no conjunto real.

Como perspectivas futuras, propomos aprofundar o estudo das métricas utilizadas na avaliação de modelos generativos, visando uma análise mais robusta e interpretável da qualidade das amostras geradas. Além disso, pretendemos avançar na compreensão da relação teórica entre modelos baseados em difusão e modelos baseados em energia, buscando esclarecer como essas classes de modelos se conectam no espaço de técnicas generativas e quais vantagens práticas cada abordagem pode oferecer.

Resultados de Arquiteturas Anteriores

Neste trabalho, descartamos uma arquitetura, e seus resultados são apresentados neste apêndice.

Para um banco de dados contendo apenas amostras de materiais colestéricos e nemáticos, treinamos de forma condicionada duas redes diferentes, uma em *RGB* e outra em preto-e-branco. Na figuras [A.1](#) e [A.2](#), para as classes colestérico e nemático, respectivamente: à esquerda, imagens amostradas do modelo descartado; e à direita, imagens do banco de dados.

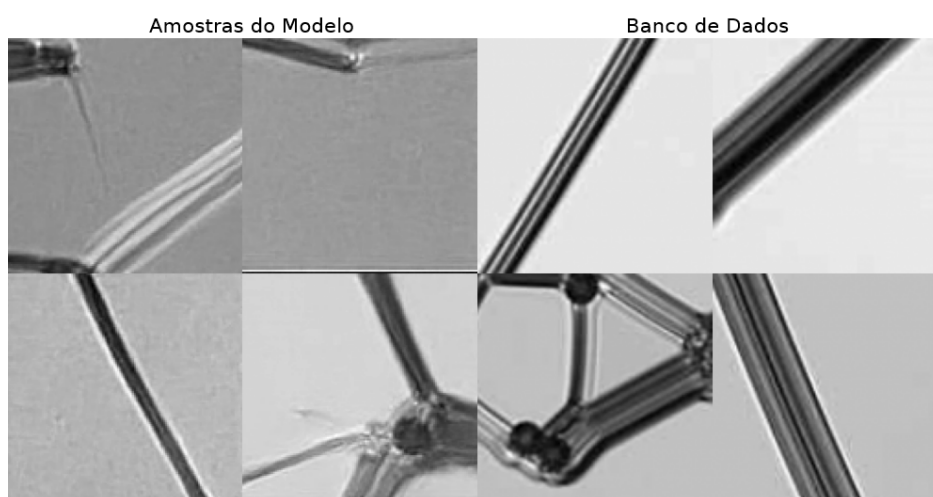


Figura A.1: Materiais colestéricos. No bloco da esquerda, formado por quatro imagens, estão as imagens geradas pelo modelo; no bloco da direita, as imagens do banco de dados.

Podemos notar na figura [A.3](#) que a rede em questão não opera bem em RGB e possui noção espacial ruim.

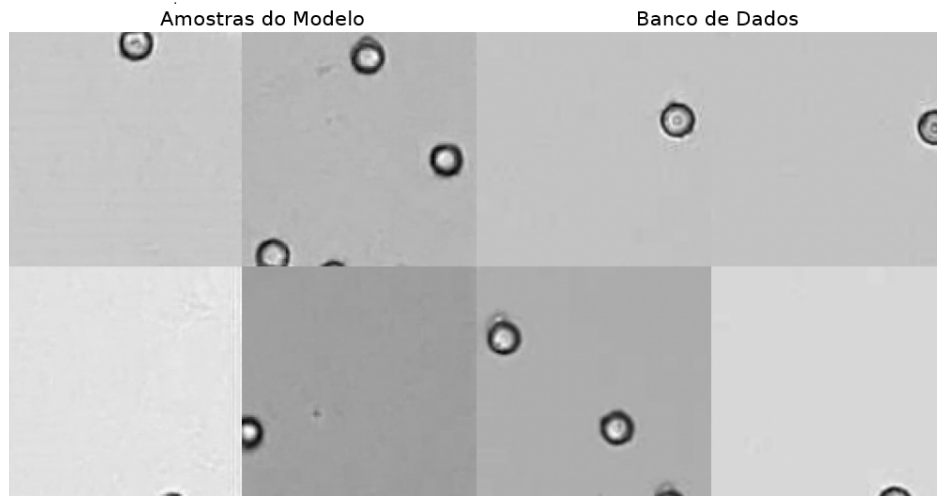


Figura A.2: Materiais nemáticos. No bloco da esquerda, formado por quatro imagens, estão as imagens geradas pelo modelo; no bloco da direita, as imagens do banco de dados.

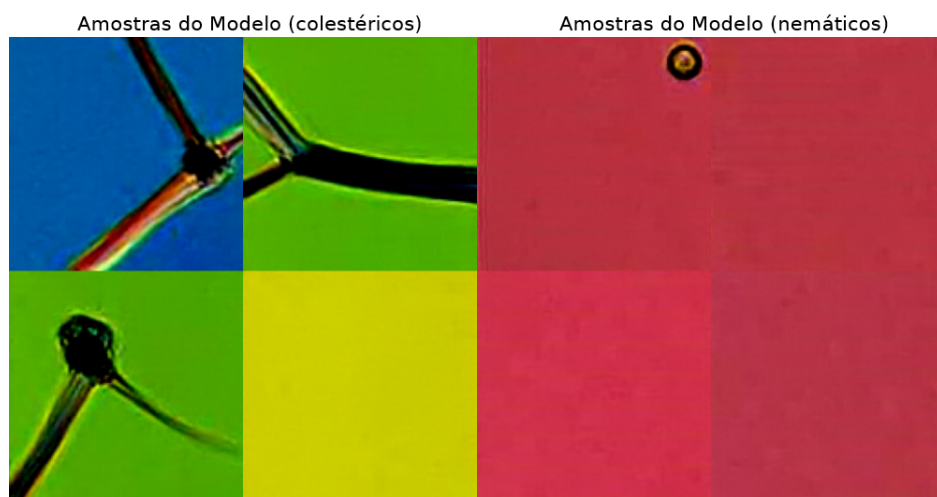


Figura A.3: No bloco da esquerda, formado por quatro imagens, estão imagens de colestéricos geradas pelo modelo; no bloco da direita, imagens de nemáticos geradas pelo modelo.

Referências Bibliográficas

- [1] McCorduck, P. *Machines Who Think: A Personal Inquiry into the History and Prospects of Artificial Intelligence* (CRC Press, 2004). URL <https://books.google.com.br/books?id=KMj1EAAAQBAJ>.
- [2] Russell, S., Russell, S. & Norvig, P. *Artificial Intelligence: A Modern Approach*. Pearson series in artificial intelligence (Pearson, 2020). URL <https://books.google.com.br/books?id=koFptAEACAAJ>.
- [3] Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. *Nature* **323**, 533–536 (1986).
- [4] LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *Nature* **521**, 436–444 (2015).
- [5] Goodfellow, I. J., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. & Bengio, Y. Generative adversarial nets. In Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. & Weinberger, K. (eds.) *Advances in Neural Information Processing Systems*, vol. 27 (Curran Associates, Inc., 2014). URL https://proceedings.neurips.cc/paper_files/paper/2014/file/f033ed80deb0234979a61f95710dbe25-Paper.pdf.
- [6] Clark, R. & Mayer, R. *e-Learning and the Science of Instruction: Proven Guidelines for Consumers and Designers of Multimedia Learning*. Wiley Desktop Editions (Wiley, 2011). URL <https://books.google.com.br/books?id=MOutGGET2VwC>.
- [7] Esteva, A., Kuprel, B., Novoa, R. A., Ko, J., Swetter, S. M., Blau, H. M. & Thrun, S. Dermatologist-level classification of skin cancer with deep neural networks. *Nature* **542**, 115–118 (2017).
- [8] Dierking, I. *Textures of liquid crystals* (John Wiley & Sons, 2003).

- [9] Wilson, M. R. Molecular simulation of liquid crystals: progress towards a better understanding of bulk structure and the prediction of material properties. *Chem. Soc. Rev.* **36**, 1881–1888 (2007).
- [10] National Academies of Sciences, E. & Medicine. *Data Science for Undergraduates: Opportunities and Options* (The National Academies Press, Washington, DC, 2018). URL <https://nap.nationalacademies.org/catalog/25104/data-science-for-undergraduates-opportunities-and-options>.
- [11] Antes, G., Galandi, D. & Bouillon, B. What is evidence-based medicine? *Langenbecks Arch Surg* **384**, 409–416 (1999).
- [12] Vahdat, A., Kreis, K. & Kautz, J. Score-based generative modeling in latent space. *Advances in neural information processing systems* **34**, 11287–11302 (2021).
- [13] Yang, L., Zhang, Z., Song, Y., Hong, S., Xu, R., Zhao, Y., Zhang, W., Cui, B. & Yang, M.-H. Diffusion models: A comprehensive survey of methods and applications. *ACM Computing Surveys* **56**, 1–39 (2023).
- [14] Po, R. *et al.* State of the art on diffusion models for visual computing. In *Computer graphics forum*, vol. 43, e15063 (Wiley Online Library, 2024).
- [15] Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N. & Ganguli, S. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*, 2256–2265 (pmlr, 2015).
- [16] Ho, J., Jain, A. & Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems* **33**, 6840–6851 (2020).
- [17] Kingma, D. P., Welling, M. *et al.* An introduction to variational autoencoders. *Foundations and Trends[®] in Machine Learning* **12**, 307–392 (2019).
- [18] Doersch, C. Tutorial on variational autoencoders. *arXiv preprint arXiv:1606.05908* (2016).
- [19] Azad, R., Aghdam, E. K., Rauland, A., Jia, Y., Avval, A. H., Bozorgpour, A., Karimijafarbigloo, S., Cohen, J. P., Adeli, E. & Merhof, D. Medical image segmentation review: The success of u-net. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2024).
- [20] Sigaki, H. Y., Lenzi, E. K., Zola, R. S., Perc, M. & Ribeiro, H. V. Learning physical properties of liquid crystals with deep convolutional neural networks. *Scientific Reports* **10**, 7664 (2020).

- [21] Gatys, L., Ecker, A. & Bethge, M. A neural algorithm of artistic style. *Journal of Vision* **16**, 326–326 (2016).
- [22] Kingma, D. P. & Welling, M. Auto-encoding variational bayes (2022). URL <https://arxiv.org/abs/1312.6114>. 1312.6114.
- [23] Ramesh, A., Dhariwal, P., Nichol, A., Chu, C. & Chen, M. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125* **1**, 3 (2022).
- [24] Radford, A. *et al.* Learning transferable visual models from natural language supervision. In Meila, M. & Zhang, T. (eds.) *Proceedings of the 38th International Conference on Machine Learning*, vol. 139 of *Proceedings of Machine Learning Research*, 8748–8763 (PMLR, 2021). URL <https://proceedings.mlr.press/v139/radford21a.html>.
- [25] Dhariwal, P. & Nichol, A. Diffusion models beat gans on image synthesis. In Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P. & Vaughan, J. W. (eds.) *Advances in Neural Information Processing Systems*, vol. 34, 8780–8794 (Curran Associates, Inc., 2021). URL https://proceedings.neurips.cc/paper_files/paper/2021/file/49ad23d1ec9fa4bd8d77d02681df5cfa-Paper.pdf.
- [26] Song, J., Meng, C. & Ermon, S. Denoising diffusion implicit models. In *International Conference on Learning Representations* (2021). URL <https://openreview.net/forum?id=StlgiaRCHLP>.
- [27] Rombach, R., Blattmann, A., Lorenz, D., Esser, P. & Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 10684–10695 (2022).
- [28] Song, Y. & Ermon, S. *Generative modeling by estimating gradients of the data distribution* (Curran Associates Inc., Red Hook, NY, USA, 2019).
- [29] Kingma, D. & Gao, R. Understanding diffusion objectives as the elbo with simple data augmentation. *Advances in Neural Information Processing Systems* **36**, 65484–65516 (2023).
- [30] Schulman, P. Bayes’ theorem—a review. *Cardiology clinics* **2**, 319–328 (1984).
- [31] Theis, L., Oord, A. v. d. & Bethge, M. A note on the evaluation of generative models. *arXiv preprint arXiv:1511.01844* (2015).
- [32] Dalalyan, A. Further and stronger analogy between sampling and optimization: Langevin monte carlo and gradient descent. In *Conference on Learning Theory*, 678–689 (PMLR, 2017).

- [33] Huang, X., Dong, H., Hao, Y., Ma, Y.-A. & Zhang, T. Reverse diffusion monte carlo. *arXiv preprint arXiv:2307.02037* (2023).
- [34] Scharf, T. *Polarized Light in Liquid Crystals and Polymers* (Wiley, 2006). URL <https://books.google.com.br/books?id=CQNE13opFucC>.
- [35] de Gennes, P. & Prost, J. *The Physics of Liquid Crystals*. International Series of Monographs on Physics (Clarendon Press, 1993). URL <https://books.google.com.br/books?id=0Nw-dzWz5agC>.
- [36] Neto, A. & Salinas, S. *The Physics of Lyotropic Liquid Crystals: Phase Transitions and Structural Properties*. Monographs on the Physics and Chemistry of Materials (OUP Oxford, 2005). URL <https://books.google.com.br/books?id=d8i1IYQKmekC>.
- [37] Jakli, A. & Saupe, A. *One- and Two-Dimensional Fluids: Properties of Smectic, Lamellar and Columnar Liquid Crystals*. Condensed Matter Physics (CRC Press, 2006). URL <https://books.google.com.br/books?id=rGfLBQAAQBAJ>.
- [38] Zola, R. S. *Effects of electric field, surface alignment and guest materials in cholesteric liquid crystals*. Ph.D. thesis (2012). URL http://rave.ohiolink.edu/etdc/view?acc_num=kent1332428717.
- [39] Majumdar, K. C., Pal, N., Nath, S., Choudhury, S. & Rao, N. V. S. Synthesis, characterization, and mesogenic properties of anisotropic imines. *Molecular Crystals and Liquid Crystals* **461**, 37–51 (2006). <https://doi.org/10.1080/15421400600983531>.
- [40] Osiecka-Drewniak, N., Piwowarczyk, M., Drzewicz, A. & Juszyńska-Gałązka, E. Liquid crystal textures, neural networks and art. *Liquid Crystals* **51**, 128–135 (2024). <https://doi.org/10.1080/02678292.2023.2280813>.
- [41] Paszke, A. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint arXiv:1912.01703* (2019).
- [42] Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention—MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 234–241 (Springer, 2015).
- [43] Nagelkerke, N. J. *et al.* A note on a general definition of the coefficient of determination. *biometrika* **78**, 691–692 (1991).
- [44] Santurkar, S., Tsipras, D., Ilyas, A. & Madry, A. How does batch normalization help optimization? *Advances in neural information processing systems* **31** (2018).

- [45] Tan, H. H. & Lim, K. H. Vanishing gradient mitigation with deep learning neural network optimization. In *2019 7th international conference on smart computing & communications (ICSCC)*, 1–4 (IEEE, 2019).
- [46] Ravnik, M. & Žumer, S. Landau–de gennes modelling of nematic liquid crystal colloids. *Liquid Crystals* **36**, 1201–1214 (2009).
- [47] Yang, D.-K. & Wu, S.-T. *Fundamentals of liquid crystal devices* (John Wiley & Sons, 2014).
- [48] Grandini, M., Bagli, E. & Visani, G. Metrics for multi-class classification: an overview. *arXiv preprint arXiv:2008.05756* (2020).
- [49] Mao, A., Mohri, M. & Zhong, Y. Cross-entropy loss functions: Theoretical analysis and applications. In *International conference on Machine learning*, 23803–23828 (pmlr, 2023).
- [50] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. & Polosukhin, I. Attention is all you need. *Advances in neural information processing systems* **30** (2017).
- [51] Rombach, R., Blattmann, A., Lorenz, D., Esser, P. & Ommer, B. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 10684–10695 (2022).
- [52] Nichol, A. Q. & Dhariwal, P. Improved denoising diffusion probabilistic models. In *International conference on machine learning*, 8162–8171 (PMLR, 2021).