UNIVERSIDADE ESTADUAL PAULISTA - UNESP

Instituto de Biociências, Letras e Ciências Exatas - Campus de São José do Rio Preto

Ricardo Trivizan Fares

Aprendizado de representações de textura com redes neurais randomizadas para análise de imagens de biossensores

São José do Rio Preto

2026

Ricardo Trivizan Fares

Aprendizado de representações de textura com redes neurais randomizadas para análise de imagens de biossensores

Dissertação apresentada à Universidade Estadual Paulista (UNESP), Instituto de Biociências, Letras e Ciências Exatas, São José do Rio Preto, para obtenção do título de Mestre em Ciência da Computação.

Área de Concentração: Computação Aplicada

Orientador: Prof. Dr. Lucas Correia Ribas

São José do Rio Preto

2026

F222a

Fares, Ricardo Trivizan
Aprendizado de representações de textura com redes neurais randomizadas para análise de imagens de biossensores / Ricardo Trivizan Fares. -- São José do Rio Preto, 2026
178 p. : il., tabs.

Dissertação (mestrado) - Universidade Estadual Paulista (UNESP), Instituto de Biociências Letras e Ciências Exatas, São José do Rio Preto
Orientador: Lucas Correia Ribas

1. Aprendizado de representações. 2. Redes neurais randomizadas. 3. Análise de imagens. 4. Análise de texturas. I. Título.

Sistema de geração automática de fichas catalográficas da Unesp. Dados fornecidos pelo autor(a).

IMPACTO POTENCIAL DESTA PESQUISA

O impacto desta pesquisa reside na melhoria do desempenho do diagnóstico de doenças por meio da análise de imagens de biossensores, utilizando técnicas de aprendizado de representações de textura. Para isso, investiga-se e desenvolve-se novos métodos de representações de textura que sejam rápidos, apresentem baixo custo computacional e demandem poucos dados, de modo que possam ser empregados em dispositivos eletrônicos passíveis de serem aplicados diretamente em pontos de atendimento para a realização de diagnósticos rápidos e de baixo custo.

POTENTIAL IMPACT OF THIS RESEARCH

The impact of this research lies in improving the performance of disease diagnosis through the analysis of biosensor images using texture representation learning techniques. To this end, novel texture representation methods are investigated and developed that are fast, exhibit low computational cost, and require low amount of data, so that they can be deployed on electronic devices for direct use at points of care, enabling rapid and low-cost diagnostics.

RICARDO TRIVIZAN FARES

APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA COM REDES NEURAIS RANDOMIZADAS PARA ANÁLISE DE IMAGENS DE BIOSSENSORES

Dissertação apresentada à Universidade Estadual Paulista (UNESP), Instituto de Biociências, Letras e Ciências Exatas, São José do Rio Preto, para obtenção do título de Mestre em Ciência da Computação.

Data da defesa: 10/04/2026

Banca Examinadora:

Prof. Dr. Lucas Correia Ribas
UNESP – Instituto de Biociências, Letras e Ciências Exatas – Campus de São José do Rio Preto

---

Prof. Dr. Arnaldo Candido Junior
UNESP – Instituto de Biociências, Letras e Ciências Exatas – Campus de São José do Rio Preto

---

Prof. Dr. João Batista Florindo
UNICAMP – Instituto de Matemática, Estatística e Computação Científica – Campus de Campinas

AGRADECIMENTOS

A Deus, que me permitiu trilhar este caminho com saúde.

Aos meus pais, que sempre me incentivaram e apoiaram as minhas decisões.

Ao meu irmão, que sempre me encorajou a ter perseverança.

Ao meu melhor amigo, Barroso, que esteve comigo durante toda a graduação e pós-graduação, bem como nos diversos momentos desta trajetória, sendo ele um dos poucos que me entende verdadeiramente.

Ao meu orientador, Lucas, pela dedicação, pelos ensinamentos, pelo apoio nos momentos turbulentos, mas, sobretudo, pela confiança depositada em mim.

Aos colegas do Laboratório de Inovação e Desenvolvimento em Inteligência Artificial (LIDIA), que direta ou indiretamente fizeram parte de um trecho desta trajetória.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

O presente trabalho foi realizado com apoio da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP), Brasil. Processo nº 2024/01744-8.

As opiniões, hipóteses e conclusões ou recomendações expressas neste material são de responsabilidade do(s) autor(es) e não necessariamente refletem a visão da FAPESP.

"The smiles as she walked in the room
Have all turned into frowns, am I too quick to assume
That the love is no longer in bloom?
The tantrums and the tears play a very different tune"
(Arctic Monkeys, Too Much To Ask)

RESUMO

FARES, R. T. **Aprendizado de representações de textura com redes neurais randomizadas para análise de imagens de biossensores**. 2026. 178 p. Dissertação (Mestrado em Ciências da Computação), Instituto de Biociências, Letras e Ciências Exatas, Universidade Estadual Paulista "Júlio de Mesquita Filho", São José do Rio Preto, SP, 2026.

A análise de imagens de biossensores para o diagnóstico de doenças emergiu como uma nova modalidade de detecção em adição às técnicas físico-químicas tradicionais. Essas imagens são dotadas de rico conteúdo microtextural e compõem conjuntos de dados ínfimos em relação aos conjuntos de dados de larga escala conhecidos; necessitando, assim, de técnicas de aprendizado de representações de texturas com baixo custo computacional, que demandem poucos dados e que caracterizem eficientemente os padrões microtexturais. Embora estudos recentes têm investigado o uso de técnicas de redes neurais randomizadas para abordar tais imagens, esses se limitaram a modalidade específica de regressão dessa rede, de modo que, a capacidade total de aprendizado da rede neural randomizada não foi totalmente explorada. Nesse sentido, para abordar tais questões, esta dissertação de mestrado está organizada em duas frentes de pesquisa. A primeira frente investiga novas técnicas com redes neurais randomizadas propondo melhorias em: (i) novas formas de cruzamento de dados, (ii) melhores sumarizações estatísticas dos pesos aprendidos que compõe a representação final; (iii) uso da modalidade multialvo e *autoencoder*; e (iv) alterações arquiteturais com foco em aprendizado textural. A segunda frente tem como objetivo desenvolver técnicas que combinam redes neurais randomizadas e modelos pré-treinados, de modo a angariar o conhecimento desses modelos por meio de transferência de aprendizado, empregando a rede neural randomizada para melhorar a transferibilidade das características no novo domínio de análise de texturas. Para isso, propõe-se a investigação de: (v) novas formas de modelagem dos mapas de ativação profundos; e (vi) melhor seleção dos estágios do modelo para angariar somente as características texturais mais importantes. Os métodos desenvolvidos foram avaliados em quatro conjuntos de dados de textura de *benchmark*, no contexto de tarefas de reconhecimento textural, e aplicados a conjuntos de imagens de biossensores, demonstrando, assim, sua aplicabilidade prática. Além disso, os resultados obtidos evidenciam uma maior capacidade textural e, sobretudo, como as contribuições metodológicas se traduzem na prática para o auxílio de um problema de elevada relevância social. Portanto, torna-se evidente que as abordagens desenvolvidas não apenas contribuem para os campos da visão computacional e do reconhecimento de padrões, como também para as áreas médica, biológica e de ciência dos materiais.

Palavras-chave: aprendizado de representações; redes neurais randomizadas; análise de imagens; análise de texturas.

ABSTRACT

FARES, R. T. **Texture representation learning with randomized neural networks for biosensor image analysis**. 2026. 178 p. Dissertação (Mestrado em Ciências da Computação), Instituto de Biociências, Letras e Ciências Exatas, Universidade Estadual Paulista "Júlio de Mesquita Filho", São José do Rio Preto, SP, 2026.

The analysis of biosensor images for disease diagnosis has emerged as a new detection modality in addition to traditional physicochemical techniques. These images exhibit rich microtextural content and comprise datasets that are small compared to well-known large-scale datasets; thus, they require texture representation learning methods with low computational cost, minimal data requirements, and efficient characterization of microtextural patterns. Although recent studies have investigated the use of randomized neural network techniques to address such images, these have been limited to a specific regression modality of this network, such that the full learning capacity of randomized neural networks has not been yet fully explored. In this regard, to address these issues, this master's dissertation is organized into two research areas. The first research area investigates novel techniques based on randomized neural networks by proposing improvements in: (i) new strategies to cross information; (ii) better statistical summarizations of the learned weights that compose the final representation; (iii) the use of multi-target and autoencoder modalities; and (iv) architectural modifications with a focus on texture learning. The second research area aims to develop techniques that combine randomized neural networks with pre-trained models in order to leverage the knowledge of these models through transfer learning, employing randomized neural networks to improve the transferability of features to the new texture analysis domain. To this end, the investigation focuses on: (v) novel strategies for modeling deep activation maps; and (vi) improved selection of model stages to capture only the most relevant textural features. The developed methods were evaluated on four benchmark texture datasets in the context of texture recognition tasks and applied to biosensor image datasets, thereby demonstrating their practical applicability. Furthermore, the results highlight an enhanced texture characterization capability and, above all, illustrate how the proposed methodological contributions translate into practice to support a problem of a high social relevance. Therefore, it becomes evident that the developed approaches not only contribute to the fields of computer vision and pattern recognition, but also to the medical, biological and material science domains.

Keywords: *representation learning; randomized neural networks; image analysis; texture analysis.*

LISTA DE ILUSTRAÇÕES

Figura 1 – Ilustração do processo do aprendizado autossupervisionado por meio do pré-treinamento do modelo utilizando uma tarefa de pretexto com posterior transferência dos pesos do modelo para aplicação nas tarefas finais. 27
Figura 2 – Ilustrações de tarefas de pretexto comumente utilizadas durante o aprendizado autossupervisionado. As tarefas são a predição de rotação, o preenchimento de regiões, a colorização e a resolução de quebra-cabeças. 28
Figura 3 – Ilustração do conceito de aprendizado por contraste. As representações da imagem âncora e positiva são atraídas no espaço de representações, enquanto da imagem negativa é repelida em relação à imagem âncora. . . 29
Figura 4 – Ilustração da arquitetura de rede neural randomizada, a qual é composta por camada de entrada, uma única camada escondida e camada de saída. . 31
Figura 5 – Ilustração da arquitetura de rede neural convolucional LeNet-5. 35
Figura 6 – Ilustração de gráficos de funções de ativação não lineares. 36
Figura 7 – Ilustração do mecanismo de autoatenção, e do mecanismo de autoatenção multicabeça utilizados nas arquiteturas baseadas em *Transformers*. 40
Figura 8 – Ilustração da arquitetura ViT e do módulo de codificação. 42
Figura 9 – Lista das técnicas citadas de aprendizado de representações com redes neurais randomizadas puras categorizadas com base na modalidade empregada. 46
Figura 10 – Lista das técnicas citadas de codificação de representações profundas categorizadas com base na técnica de transferência de aprendizado empregada. 53
Figura 11 – Ilustração da metodologia para aprendizado de representações de textura a partir de múltiplas projeções aleatórias usando *autoencoders* randomizados. 61
Figura 12 – Ilustração das projeções aleatórias da representação de textura $\vec{\Phi}(49, 9)$. . . 64
Figura 13 – Ilustração da metodologia proposta para aprendizado de representações de textura a partir da modelagem em grafos locais. 66
Figura 14 – Ilustração do processo de obtenção dos graus de saída do grafo local cujos vértices modelam o relacionamento das intensidades da janela apresentada. 67
Figura 15 – Comportamento da acurácia de classificação e do número de atributos da representação de textura parcial $\vec{\Omega}(Q)$. 69
Figura 16 – Ilustração do módulo espacial responsável pelo aprendizado das características e aparências texturais de um dado canal. 79
Figura 17 – Ilustração do módulo espectral responsável pelo aprendizado das características e aparências espectrais entre canais adjacentes. 80
Figura 18 – Ilustração do diagrama de alto nível que sumariza os principais passos da abordagem proposta de representação de textura espaço-espectral. 81

Figura 19 – Visualizações das reconstruções dos canais de cores de uma mesma imagem por parte das componentes espacial e espectral. 83
Figura 20 – Visualizações das reconstruções das imagens coloridas por parte das componentes espacial e espectral. 84
Figura 21 – Ilustração do cubo de cor volumétrico utilizado para a obtenção da informação espaço-espectral local da vizinhança para um dado pixel. 85
Figura 22 – Visão geral da metodologia proposta para aprendizado de representações de texturas coloridas por cubos de cor com *autoencoders* randomizados. . 86
Figura 23 – Comportamento da acurácia da representação parcial $\vec{\Theta}(Q)$ à medida que a dimensão da camada escondida do *autoencoder* randomizado é alterado. 87
Figura 24 – Visualização de duas imagens provenientes da Outex e USPtex e suas respectivas reconstruções a partir de distintas representações parciais $\vec{\Theta}(Q)$. . 89
Figura 25 – Ilustração da metodologia de aprendizado de representações de textura com redes neurais randomizadas puras denominada por Mixer. 91
Figura 26 – Ilustração da técnica de preenchimento de uma imagem. (a) Exemplo abstrato. (b) Exemplo concreto com a técnica de replicação. 92
Figura 27 – Ilustração em (a) da saída do módulo de extração de padrões locais para uma imagem de exemplo contendo um único canal, e (b) a etapa de redimensionamento realizada pelo módulo de projeção aleatória hiperesférica. . 93
Figura 28 – Ilustração da hiperesfera $\mathbb{S}^1$ representada por uma circunferência, e da hiperesfera $\mathbb{S}^2$ representada pela superfície de uma esfera. 95
Figura 29 – Comportamento da acurácia (%) da representação de textura $\vec{\Omega}_{59}(I)$ para todos os conjuntos de dados de *benchmark* à medida que variam os valores de regularização dos submódulos direto e misto. 100
Figura 30 – Comportamento da acurácia média (%) da representação de textura $\vec{\Omega}_{59}(I)$ para os pares de conjuntos de dados da Outex & CUReT e USPtex & MBT à medida que variam os valores de regularização dos ramos direto e misto. . 101
Figura 31 – Comportamento da acurácia média (%) para a representação $\vec{\Omega}_Q(I)$ quando somente um dos submódulos ou ambos são utilizados. 102
Figura 32 – Comportamento da acurácia da representação de textura $\vec{\Omega}_Q(I)$ à medida que altera-se dimensão da camada escondida. 103
Figura 33 – Ilustração da metodologia de aprendizado de representações com redes neurais randomizadas híbridas, denominada por TOCHA. 115
Figura 34 – Histogramas das distribuições dos pesos para o modelo ConvNeXt-T (IN-1K) para todos os estágios (dispostos em cada linha) e para três raios de modelagem (dispostos em cada coluna). 121
Figura 35 – Comportamentos das acurácias de classificação para três valores distintos de raios de modelagem e quatro valores de fluxo para todos os conjuntos de dados de aplicações utilizando o *backbone* ConvNeXt-L (IN-1K). 123

Figura 36 – Comportamento das acurácias de classificação para o conjunto de dados de imunossensores à medida que variam-se os hiperparâmetros $\mathcal{B}$, η e r da representação de textura proposta. 124
Figura 37 – Comportamento das acurácias de classificação para o conjunto de dados de genossensores (PCA3) à medida que variam-se os hiperparâmetros $\mathcal{B}$, η e r da representação de textura proposta. 125
Figura 38 – Comportamento das acurácias de classificação para o conjunto de dados de genossensores (SARS-CoV-2) à medida que variam-se os hiperparâmetros $\mathcal{B}$, η e r da representação de textura proposta. 126
Figura 39 – Comportamento das acurácias médias de classificação à medida que variam-se os hiperparâmetros $\mathcal{B}$, η e r da representação de textura proposta. . . . 127
Figura 40 – Imagens das amostras dos genossensores para o diagnóstico de câncer de próstata com o biomarcador PCA3. As imagens correspondem aos analitos de sequência negativa, branco e às diferentes concentrações de PCA3. . . . 132
Figura 41 – Comportamento das acurácias de classificação no conjunto de dados de Genossensores (PCA3) para o problema no contexto binário ou multiclasse, considerando o classificador escolhido e a dimensão de recorte. 134
Figura 42 – Matrizes de confusão das representações aplicadas ao problema de diagnóstico de câncer de próstata com imagens de genossensores. 138
Figura 43 – Imagens das amostras dos genossensores para o diagnóstico de SARS-CoV-2. As imagens correspondem aos analitos de sequência negativa, HPV16, PCA3 e às diferentes concentrações de ssDNA de SARS-CoV-2. 141
Figura 44 – Comportamento das acurácias de classificação no conjunto de dados de Genossensores (SARS-CoV-2) para o problema no contexto binário ou multiclasse, considerando o classificador escolhido e a dimensão de recorte. . . 142
Figura 45 – Matrizes de confusão das representações aplicadas ao problema de diagnóstico de SARS-CoV-2 com imagens de genossensores. 146
Figura 46 – Imagem das amostras dos imunossensores para o diagnóstico de SARS-CoV-2. As imagens correspondem aos analitos de sequência negativa, branco e às diferentes concentrações do vírus inativado de SARS-CoV-2. 149
Figura 47 – Exemplos de imagens de textura de distintas classes do conjunto de dados Outex. 172
Figura 48 – Exemplos de imagens de textura de distintas classes sob diferentes condições de iluminação do conjunto de dados CUReT. 173
Figura 49 – Exemplos de imagens de textura de distintas classes do conjunto de dados USPtex. 174

LISTA DE TABELAS

Tabela 1 – Acurácias de classificação da representação de textura $\vec{\Phi}(Q, S)$. 63
Tabela 2 – Acurácias de classificação para representação de textura $\vec{\Phi}(Q_1, Q_2)$. 70
Tabela 3 – Comparação das acurácias obtidas pelos métodos de representação de textura em nível de cinza propostos com aqueles presentes na literatura. . . . 73
Tabela 4 – Comparação do tempo de processamento médio necessário para a extração das características de uma imagem em nível de cinza de dimensão 224×224. 76
Tabela 5 – Acurácias de classificação (%) para a representação de textura espaço-espectral $\vec{\Psi}(Q)$ do método proposto denominado SST. 82
Tabela 6 – Acurácias de classificação (%) para a representação de textura colorida $\vec{\Omega}(Q_1, Q_2)$ do método proposto denominado VCTex. 88
Tabela 7 – Comportamento da acurácia (%) da representação de textura $\vec{\Upsilon}_{Q_1,Q_2}(I)$ do método proposto denominado Mixer para todos os conjuntos de dados. 104
Tabela 8 – Comparação das acurácias obtidas pelos métodos de representação de textura colorida propostos com aqueles presentes na literatura. 107
Tabela 9 – Descrição geral do conjunto de dados de genossensores para a detecção do biomarcador PCA3. Exibe-se para cada analito, a classe, a quantidade de imagens em suas respectivas resoluções espaciais, e o total de imagens. . . 131
Tabela 10 – Comparação das acurácias de classificação no conjunto de dados de Genossensores (PCA3) obtidas pelos métodos randomizados puros propostos com aqueles presentes na literatura. 135
Tabela 11 – Comparação das acurácias de classificação no conjunto de dados de Genossensores (PCA3) obtidas pelo método randomizado híbrido proposto com aqueles presentes na literatura. 136
Tabela 12 – Descrição geral do conjunto de dados de genossensores para a detecção de SARS-CoV-2. Exibe-se para cada analito, a classe em ambos contextos de classificação e o seu número de amostras. 140
Tabela 13 – Comparação das acurácias de classificação no conjunto de dados de Genossensores (SARS-CoV-2) obtidas pelos métodos propostos com aqueles presentes na literatura. 143
Tabela 14 – Comparação das acurácias de classificação no conjunto de dados de Genossensores (SARS-CoV-2) obtidas pelo método randomizado híbrido proposto com aqueles presentes na literatura. 144
Tabela 15 – Descrição geral do conjunto de dados de imunossensores para a detecção de SARS-CoV-2. Apresenta-se o número de amostras, se a classe é distinguível das demais, e qual a classe representante durante o contexto de classificação binário. 148

Tabela 16 – Comparação das acurácias de classificação no conjunto de dados de Imunossensores (SARS-CoV-2) obtidas pelos métodos randomizados puros propostos com aqueles presentes na literatura. 151
Tabela 17 – Comparação das acurácias de classificação no conjunto de dados de Imunossensores (SARS-CoV-2) obtidas pelo método randomizado híbrido proposto com aqueles presentes na literatura. 153
Tabela 18 – Descrição geral dos conjuntos de imagens de texturas empregados nesta dissertação para a análise das abordagens de representação de textura desenvolvidas. 175

LISTA DE ABREVIATURAS E SIGLAS

CNN	*Convolutional Neural Network*
DCNN	*Deep Convolutional Neural Network*
GAP	*Global Average Pooling*
IN-1K	*ImageNet-1K*
IN-22K	*ImageNet-22K*
KNN	*K-Nearest Neighbors*
LCG	*Linear Congruential Generator*
LDA	*Linear Discriminant Analysis*
LOO	*Leave-One-Out*
PCA3	*Prostate Cancer Antigen 3*
PFU	*Plaque-Forming Unit*
PoC	*Point of Care*
RNA	Rede Neural Artificial
RNN	*Randomized Neural Network*
RSV	*Respiratory Syncytial Virus*
ssDNA	*single-stranded DNA*
SEM	*Scanning Electron Microscope*
SVM	*Support Vector Machine*
TL	*Transfer Learning*
ViT	*Vision Transformer*

LISTA DE SÍMBOLOS

$\mathbb{N}$	Conjunto dos números naturais
$\mathbb{Z}$	Conjunto dos números inteiros
$\mathbb{R}$	Conjunto dos números reais
$\mathbb{R}^n$	Conjunto dos vetores reais n-dimensionais
$\mathbb{R}^{m \times n}$	Conjunto das matrizes reais de dimensão $m \times n$
$\mathbb{F}(\mathbb{R}^m, \mathbb{R}^n)$	Conjunto das funções de $\mathbb{R}^m$ em $\mathbb{R}^n$
$\mathbb{F}(\mathcal{X}, \mathcal{Y})$	Conjunto das funções de $\mathcal{X}$ em $\mathcal{Y}$
$\mathcal{X}^n$	Conjunto das n-tuplas com elementos em $\mathcal{X}$
$\mathcal{X}^{m \times n}$	Conjunto das matrizes com elementos em $\mathcal{X}$ de dimensão $m \times n$
$\mathcal{X} \times \mathcal{Y}$	Produto cartesiano entre os conjuntos $\mathcal{X}$ e $\mathcal{Y}$
$\lvert\mathcal{X}\rvert$	Cardinalidade do conjunto $\mathcal{X}$
(a, b)	Intervalo aberto dos números reais entre a e b
$[a, b]$	Intervalo fechado dos números reais entre a e b
$(a, b)_{\mathcal{X}}$	Intervalo aberto dos números em $\mathcal{X} \subseteq \mathbb{R}$ e entre a e b
$[a, b]_{\mathcal{X}}$	Intervalo fechado dos números em $\mathcal{X} \subseteq \mathbb{R}$ e entre a e b
$\lVert\cdot\rVert_2$	Norma euclidiana
$\lVert\cdot\rVert_F$	Norma de Frobenius
$\mathcal{N}(\mu, \sigma^2)$	Distribuição normal unidimensional com média μ e desvio padrão σ
$\mathcal{N}(0, 1)$	Distribuição normal padrão unidimensional
$[\vec{a}]_k$	Valor na k-ésima entrada do vetor $\vec{a}$
$[A]_{i,j}$	Valor na entrada (i, j) da matriz A

SUMÁRIO

1 INTRODUÇÃO 18
1.1 OBJETIVOS 21
1.2 ORGANIZAÇÃO DO TEXTO 21

2 FUNDAMENTAÇÃO TEÓRICA 23
2.1 APRENDIZADO DE REPRESENTAÇÕES 23
2.1.1 Paradigmas de Aprendizado 24
2.2 REDES NEURAIS ARTIFICIAIS 29
2.2.1 Redes Neurais Randomizadas 30
2.2.2 Redes Neurais Convolucionais 34
2.2.3 Vision Transformers 39

3 TRABALHOS RELACIONADOS 45
3.1 APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA COM REDES NEURAIS RANDOMIZADAS PURAS 45
3.1.1 Estratégias de Aprendizado Autossupervisionado e Aprendizado de Representações com Redes Neurais Randomizadas Puras 45
3.1.2 Estratégias com Medidas Estatísticas para a Sumarização dos Pesos Aprendidos de Redes Neurais Randomizadas 50
3.2 APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA COM REDES NEURAIS RANDOMIZADAS HÍBRIDAS 52
3.2.1 Estratégias de Codificação de Representações Profundas 52
3.3 CONSIDERAÇÕES FINAIS 57

4 APRENDIZADO DE REPRESENTAÇÕES COM REDES NEURAIS RANDOMIZADAS PURAS 58
4.1 APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA EM NÍVEL DE CINZA 59
4.1.1 Aprendizado de Representações de Textura por Ensemble de Autoencoders Randomizados 59
4.1.2 Aprendizado de Representações de Grafos Locais para Análise de Texturas . 65
4.1.3 Comparação com Métodos da Literatura 71
4.2 APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA COLORIDA . . . 77
4.2.1 Aprendizado de Representações Espaço-Espectrais com Redes Neurais Randomizadas 78
4.2.2 Aprendizado de Representações Volumétrica com Autoencoders Randomizados 84

4.2.3 Aprendizado de Representações com Rede Neural Randomizada de Incorporação Mista Hiperesférica . . . 90
4.2.4 Comparação com Métodos da Literatura . . . 106
4.3 CONSIDERAÇÕES FINAIS . . . 111

5 APRENDIZADO DE REPRESENTAÇÕES COM REDES NEURAIS RANDOMIZADAS HÍBRIDAS . . . 112
5.1 APRENDIZADO DE REPRESENTAÇÕES POR CODIFICAÇÃO ALEATÓRIA DE CARACTERÍSTICAS DE BACKBONES CONVOLUCIONAIS . . . 113
5.1.1 Aprendizado de Representações de Textura por Caracterização Topológica e Codificação Aleatória de Mapas de Ativação Profundos . . . 113
5.2 CONSIDERAÇÕES FINAIS . . . 129

6 APLICAÇÕES . . . 130
6.1 DIAGNÓSTICO DE CÂNCER DE PRÓSTATA COM BIOMARCADOR PCA3 POR IMAGENS DE MICROSCOPIA ELETRÔNICA DE GENOSSENSORES 130
6.1.1 Descrição do Conjunto de Dados . . . 131
6.1.2 Análise Experimental e Discussões . . . 133
6.2 DIAGNÓSTICO DO VÍRUS SARS-COV-2 POR IMAGENS DE MICROSCOPIA ELETRÔNICA DE GENOSSENSORES . . . 139
6.2.1 Descrição do Conjunto de Dados . . . 139
6.2.2 Análise Experimental e Discussões . . . 140
6.3 DIAGNÓSTICO DO VÍRUS SARS-COV-2 POR IMAGENS DE MICROSCOPIA ÓPTICA DE IMUNOSSENSORES . . . 147
6.3.1 Descrição do Conjunto de Dados . . . 148
6.3.2 Análise Experimental e Discussões . . . 150
6.4 CONSIDERAÇÕES FINAIS . . . 154

7 CONCLUSÕES . . . 155
7.1 TRABALHOS FUTUROS . . . 157
7.2 LISTA DE PUBLICAÇÕES . . . 157

REFERÊNCIAS . . . 160

A CONJUNTO DE DADOS DE TEXTURA . . . 171
A.1 CONJUNTOS DE DADOS DE TEXTURA EM NÍVEL DE CINZA . . . 171
A.2 CONJUNTOS DE DADOS DE TEXTURA COLORIDA . . . 172

DADOS CURRICULARES . . . 176

1 INTRODUÇÃO

Biossensores são dispositivos analíticos compostos por um biorreceptor (e.g., enzimas, anticorpos, células, aptâmeros etc.), que tem como função interagir quimicamente com o analito (i.e., substância de interesse na investigação), e um transdutor responsável por converter os sinais resultantes dessa interação em sinais elétricos mensuráveis (Raymundo-Pereira et al., 2020). Esses dispositivos analíticos são amplamente estudados na literatura devido à elevada capacidade de aplicá-los no âmbito prático, sendo utilizados para: diagnóstico de doenças, monitoramento ambiental e controle de qualidade alimentar (Turner, 2013). Em particular, os biossensores têm desempenhado papel essencial para o diagnóstico rápido e de baixo custo de doenças como SARS-CoV-2 (Soares et al., 2021; Oiticica et al., 2025) e câncer de próstata (Rodrigues et al., 2021), permitindo o início precoce do tratamento do indivíduo acometido pela doença e, consequentemente, possibilitando o aumento das chances de recuperação. Nesse sentido, um dos objetivos com relação aos avanços de técnicas biossensoriais é permitir que o diagnóstico de doenças seja rápido e acessível à população, de modo que, seja capaz de ser aplicado, sem a necessidade de pessoal especializado, diretamente nos pontos de atendimento (Rodrigues et al., 2021; Oiticica et al., 2025; Carvalho et al., 2025).

Entretanto, dentre as etapas de diagnóstico, é necessário mensurar a intensidade da interação química que ocorre no biossensor e, para isso, são necessários instrumentos de medição físico-químicos (e.g., espectroscopia de impedância eletroquímica), os quais são custosos e cujas análises não são triviais, podendo, assim, limitar sua adoção em locais com recursos limitados (Docherty et al., 2025). Isso posto, os estudos de Soares et al. (2021), Rodrigues et al. (2021) e Oiticica et al. (2025) investigaram, além das técnicas tradicionais, um novo paradigma de detecção baseado na análise de imagens e em aprendizado de máquina. Nessas investigações, foram observadas que, uma vez que a unidade sensorial (i.e., biossensor) é exposta à amostra biológica a ser detectada, tem-se que o biossensor sofre mudanças morfológicas e microtexturais, as quais são capazes de serem detectadas por meio da combinação de técnicas de análise textural com algoritmos de aprendizado de máquina. Nesse contexto, torna-se evidente a atratividade dos métodos de representações de textura para abordar o problema de diagnóstico de doenças a partir de imagens de biossensores, as quais são capturadas com o emprego de microscópios ópticos ou eletrônicos. Ainda, essas são pouco exploradas na literatura, uma vez que grande parte dos estudos se restringe à análise das próprias imagens das amostras biológicas, e não das imagens dos biossensores em si (Rodrigues et al., 2021).

Comumente, os métodos de representação de textura podem ser divididos em duas categorias: abordagens clássicas ou manuais (i.e., *handcrafted*) e abordagens baseadas em aprendizado. Em particular, as abordagens clássicas têm como característica principal o fato de serem manualmente construídas por especialistas, a partir da especificação de uma sequência

de etapas ou procedimentos bem definidos, de modo a conduzir a imagem de entrada até sua respectiva representação, essa consistindo de uma sequência de números cujo objetivo é carregar informações essenciais sobre a imagem. Consequentemente, essas abordagens possuem como vantagens uma maior interpretabilidade de seu funcionamento e certa invariância ao tamanho do conjunto de dados, desde que grande parcela das técnicas obtém a representação individualmente por imagem, i.e., sem a necessidade de outras imagens do conjunto, assim, sendo úteis em contextos em que o conjunto de dados é ínfimo, como abordado em Scabini et al. (2026). No entanto, devido ao certo grau de inflexibilidade inerente de tais abordagens, essas apresentam limitações para a extração de características e padrões visuais mais complexos e não lineares, resultando, assim, em uma redução de sua capacidade discriminativa. Dessa forma, por essas razões, a aplicabilidade das técnicas clássicas mostra-se limitada no contexto de imagens de biossensores, mesmo que estas sejam dotadas de rico conteúdo microtextural.

Por outro lado, as técnicas baseadas em aprendizado empregam redes neurais artificiais (RNAs), tais como redes neurais convolucionais (em inglês, *Convolutional Neural Networks* (CNNs)) e *Vision Transformers* (ViTs), as quais aprendem automaticamente um mapeamento entre a imagem de entrada e sua representação, com a finalidade de melhorar o desempenho em tarefas finais (Le-Khac; Healy; Smeaton, 2020). Entretanto, embora tais técnicas tenham obtido, continuadamente, resultados expressivos em diversas tarefas do campo da visão computacional, como classificação de imagens (Zhang et al., 2025), segmentação de imagens médicas (He et al., 2025), super-resolução de imagens (Chen et al., 2025), essas apresentam elevados custos computacionais (i.e., processamento e memória), financeiros (e.g., aquisição de *hardware* especializado e custos energéticos) e de tempo, além de demandarem grandes volumes de dados para o treinamento. Consequentemente, no contexto atual, tais características limitam a adoção dessas técnicas não apenas pelo custo, mas também, sobretudo, pelo fato de que os conjuntos de imagens de biossensores são pequenos, devido ao tempo necessário para a aquisição das amostras combinado aos altos custos dos microscópios necessários.

Em contrapartida, diferentemente das técnicas apresentadas, estudos têm investigado métodos de representação de textura utilizando redes neurais randomizadas (em inglês, *Randomized Neural Networks* (RNNs)), empregadas em diversas tarefas, como análise de texturas (Florindo; Backes; Neckel, 2024; Fares; Ribas, 2025b), análise de vídeos (Ribas et al., 2022) e análise de contornos (Ribas; Bruno, 2024). Essa arquitetura, em sua forma mais simples, é composta por apenas uma camada escondida cujos pesos são gerados aleatoriamente e utiliza uma solução de fórmula fechada para o seu treinamento. Diante disso, a rede neural randomizada mostrou-se essencial, por reunir os benefícios de ambas as categorias de técnicas apresentadas e, ao mesmo tempo, endereçar os problemas inerentes a cada uma delas (Fares; Ribas, 2025a). Precisamente, as abordagens baseadas em RNNs são invariantes ao conjunto de dados, estando, assim, relacionadas às técnicas clássicas, e pelo fato de possuírem uma etapa de treinamento, tais redes são capazes de aprender padrões mais complexos, superando uma limitação inerente às abordagens manuais e, ao mesmo tempo, estabelecendo um para-

lelo com as técnicas baseadas em aprendizado. Do mesmo modo, pelo fato de tal etapa de treinamento ser conduzida por meio de uma solução em fórmula fechada, dispensa-se o uso de mecanismos de retropropagação (i.e., *backpropagation*) tornado-a uma abordagem de baixo custo e rápida. Dessa forma, por tais razões, essa é uma arquitetura chave neste trabalho.

Nesse contexto, nesta dissertação, em uma primeira frente de pesquisa, são investigadas e desenvolvidas novas técnicas de aprendizado de representações por instância com redes neurais randomizadas, em que o aprendizado por instância refere-se à capacidade de uma técnica em aprender a representação para uma única dada imagem independentemente de outras imagens do conjunto de dados. Em tese, o objetivo é aprimorar a caracterização microtextural e a discriminabilidade das imagens de biossensores por meio dos seguintes caminhos: (i) formas mais eficazes de cruzamento de dados na rede; (ii) melhores sumarizações estatísticas dos pesos aprendidos; (iii) uso de redes neurais randomizadas na forma multialvo e de *autoencoders* randomizados; e (iv) alterações arquiteturais. Além disso, para auxiliar na obtenção de tais objetivos, empregam-se conceitos advindos de mecanismos autossupervisionados, como o uso de reconstruções e de múltiplas projeções aleatórias, com a finalidade de enriquecer o conteúdo semântico da representação, tornado-a mais robusta e discriminativa.

Ainda mais, com o sucesso das arquiteturas de redes neurais (e.g., CNNs e ViTs) em outras áreas de visão computacional e reconhecimento de padrões, essas foram empregues em problemas de análise textural por meio de transferência de aprendizado (em inglês, *Transfer Learning* (TL)), seja como extratores de características ou para ajuste fino (em inglês, *fine-tuning*) (Cimpoi; Maji; Vedaldi, 2015; Xue; Zhang; Dana, 2018; Zhai et al., 2020; Chen et al., 2024). No contexto de ajuste fino, a rede pré-treinada é ajustada ao novo domínio por um novo processo de treinamento iterativo, entretanto, a depender do modelo, isso requisitará conjuntos de dados de larga escala para prevenir problemas de sobreajuste (Scabini et al., 2025), evidenciado-se uma limitação desta técnica para os dados de biossensores abordados. Dessa maneira, a TL como extratora de características torna-se a abordagem mais atrativa, entretanto, os modelos pré-treinados cujo treinamento é realizado sobre um conjunto de dados de objetos e cenas (e.g., ImageNet (Deng et al., 2009)) podem apresentar limitada transferibilidade para aplicações de análise textural (Matos et al., 2023; Chen et al., 2024).

Nesse sentido, para abordar essa questão, nesta segunda frente de pesquisa, investigam-se técnicas de aprendizado de representações de textura com redes neurais randomizadas para a codificação aleatória dos mapas de ativação/características profundos de grandes modelos pré-treinados, com a finalidade de melhorar a transferibilidade desses modelos. Dessa forma, de maneira análoga à primeira frente, o acoplamento do módulo de codificação aleatória aos grandes modelos culmina em um aprendizado de representações por instância, uma vez que não há qualquer forma de treinamento iterativo, tampouco a necessidade de utilizar outras imagens do conjunto para a extração da representação. Logo, torna-se possível aproveitar o aprendizado desses grandes modelos para a construção de representações discriminativas.

1.1 OBJETIVOS

A seguir, são elencados os objetivos principais desta dissertação: (i) investigar e desenvolver métodos de aprendizado de representações por instância com redes neurais randomizadas, com foco no aprimoramento da caracterização microtextural de imagens de biossensores; (ii) melhorar a transferibilidade de grandes modelos por meio de codificação aleatória; e (iii) aplicar os métodos desenvolvidos à análise de imagens de biossensores para o diagnóstico de doenças. Para tanto, esta dissertação está organizada em duas frentes de pesquisa:

- **Frente de Pesquisa I**: Esta frente de pesquisa está associada ao objetivo (i) e visa investigar novos métodos de aprendizado com redes neurais randomizadas. Para atingir tal objetivo, assim como descrito anteriormente, serão investigadas: formas de cruzamento de dados, novas sumarizações estatísticas dos pesos aprendidos, emprego de *autoencoders* randomizados e alterações arquiteturais com foco na análise de texturas.
- **Frente de Pesquisa II**: Esta frente de pesquisa está associada ao objetivo (ii) e visa a melhoria da transferibilidade de grandes modelos por meio de codificação aleatória por RNNs. Para atingir tal objetivo, serão investigadas: melhores formas de modelagem dos mapas de ativação profundos e novas maneiras de realizar a seleção e agregação das características profundas extraídas em diferentes camadas do modelo pré-treinado.

Por fim, visando atingir o objetivo (iii), as abordagens propostas nas Frentes I e II serão aplicadas a imagens de biossensores para fins de diagnóstico de doenças de forma rápida e de baixo custo. Essas aplicações consistem de problemas reais com elevado potencial de contribuição à sociedade, sobretudo quando essas abordagens são integradas a tecnologias de diagnóstico, principalmente, aquelas empregadas diretamente em pontos de atendimento.

1.2 ORGANIZAÇÃO DO TEXTO

Este trabalho está organizado em sete capítulos. Neste Capítulo 1, inicialmente, a introdução da dissertação foi apresentada, bem como suas motivações e objetivos, principalmente, aquela que refere-se à aplicação das abordagens desenvolvidas para o diagnóstico de doenças.

Em seguida, o Capítulo 2 apresenta os fundamentos teóricos essenciais utilizados na investigação e desenvolvimento das abordagens propostas, com o objetivo de possibilitar seu pleno entendimento. Em particular, são apresentados conceitos de aprendizado de máquina, aprendizado de representações e arquiteturas de redes neurais artificiais (e.g., CNNs e ViTs).

No Capítulo 3, uma revisão da literatura é conduzida e são abordados os trabalhos e metodologias mais relacionadas e relevantes com aqueles investigados nesta dissertação.

Esses trabalhos são apresentados, exibindo suas contribuições para a literatura, suas limitações existentes e como as abordagens investigadas e desenvolvidas preenchem tais lacunas.

Na sequência, o Capítulo 4 apresenta as abordagens propostas de aprendizado de representações de textura com redes neurais randomizadas puras. Esse capítulo está dividido em duas seções, as quais tratam das técnicas de aprendizado de representações de textura para imagens em nível de cinza e coloridas, respectivamente. Além disso, ao fim de cada seção, é conduzida a comparação das abordagens propostas com os métodos da literatura seguida da discussão dos resultados, com a finalidade de exibir os benefícios das abordagens investigadas.

No Capítulo 5, são apresentadas as abordagens propostas de aprendizado de representações de textura com redes neurais randomizadas híbridas. Especificamente, este capítulo dedica-se à apresentação do método investigado pertencente à segunda frente de pesquisa cuja finalidade é melhorar a transferibilidade de grandes modelos pré-treinados (i.e., *backbones*).

Logo após, o Capítulo 6 aborda a aplicação das abordagens desenvolvidas nos capítulos anteriores em problema reais, com a finalidade de exibir como as contribuições metodológicas dessas abordagens se traduzem no âmbito prático, principalmente, em uma aplicação tão importante que é o diagnóstico de doenças, em particular, SARS-CoV-2 e câncer de próstata.

Por fim, o Capítulo 7 apresenta as conclusões desta dissertação.

2 FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, apresenta-se a fundamentação teórica necessária para o pleno entendimento e desenvolvimento das abordagens de aprendizado de representações propostas nesta dissertação. Para isso, primeiramente, na Seção 2.1, introduz-se o conceito de aprendizado de representações, bem como sua importância e suas diferenças em relação às técnicas extratoras de características anteriores à popularização das redes neurais artificiais e do aprendizado aprofundo. Em seguida, discutem-se algumas das principais formas de aprendizado existentes para o treinamento de redes neurais artificiais, as quais estão mais diretamente relacionadas a esta dissertação. Por fim, na Seção 2.2, são apresentadas as redes neurais randomizadas, que constituem a arquitetura de rede neural artificial central desta dissertação; as redes neurais convolucionais, em conjunto com conceitos de transferência de aprendizado; e os *Vision Transformers*, que correspondem à arquitetura mais recente abordada.

2.1 APRENDIZADO DE REPRESENTAÇÕES

Segundo Bengio, Courville & Vincent (2013), a capacidade de algoritmos de aprendizado de máquina de realizarem eficientemente suas tarefas dependem inerentemente da robustez e do poder discriminativo da representação dos dados escolhida, devendo-se ao fato de que cada representação abstrai diferentemente os dados podendo, assim, contribuir ou minar a efetividade do algoritmo. Dessa forma, torna-se evidente as razões pelas quais diversos estudos buscam investigar e desenvolver novas técnicas de representação de dados.

Especificamente, as técnicas de representação, também denominadas por extratores de características, podem ser representadas por uma função $f \in \mathbb{F}(\mathcal{X}, \mathcal{R})$ cujo objetivo consiste em obter uma representação $f(\vec{x}) \in \mathcal{R}$ para uma dada amostra $\vec{x} \in \mathcal{X}$, na qual se assume, preferencialmente, que $\mathcal{R}$ seja um espaço de dimensão significativamente menor que $\mathcal{X}$. Em particular, espera-se que a dimensão do espaço de saída seja menor, uma vez que uma parcela substancial do conteúdo de entrada pode ser redundante ou não essencial para a tarefa final.

Isso posto, conforme destacado em LeCun, Bengio & Hinton (2015), tem-se que no início das técnicas de representação, essas eram compostas por uma série de etapas bem definidas, as quais eram estabelecidas por especialistas no domínio. No contexto de análise de texturas, métodos que se enquadram nessa categoria, os quais são denominados por métodos clássicos ou manuais, apresentam como exemplos o GLCM (Haralick; Shanmugam; Dinstein, 1973) e o LBP (em inglês, *Local Binary Pattern*) (Ojala; Pietikainen; Maenpaa, 2002). Entretanto, por serem compostas por uma sequência de etapas, tais técnicas tornam-se inflexíveis para tratar de cenários mais complexos, assim, limitando sua capacidade discriminativa.

Nesse contexto, com a finalidade de obter melhores representações, é interessante que métodos de representação sejam menos dependentes da necessidade de um especialista para sua especificação (Bengio; Courville; Vincent, 2013). Por essa razão, estudos de Rumelhart, Hinton & Williams (1986), LeCun et al. (1989) e Hinton, Osindero & Teh (2006) investigaram a utilização de redes neurais artificias para aprenderem automaticamente a extrair as informações essenciais dos dados de entrada para a construção da representação, com o objetivo de solucionar uma dada tarefa. Desse modo, isso possibilitou dois aspectos principais: a capacidade de extração de características da própria rede de se adequar à complexidade presente nos dados e a redução da necessidade de especificação manual de etapas, concentrando-se no projeto de arquiteturas mais eficazes (Le-Khac; Healy; Smeaton, 2020).

Por fim, embora esse passo fundamental tenha dado início à área denominada aprendizado de representações, as limitações que implicaram de seu surgimento continuam a ser investigadas até o dias atuais. Por exemplo, uma das características estudadas refere-se à obtenção de melhores representações, bem como à capacidade do modelo generalizar adequadamente, em cenários nos quais se dispõe de uma grande quantidade de dados ou, de forma oposta, em cenários nos quais há escassez de dados rotulados. Logo, para tratar os diferentes contextos nos quais o aprendizado de representações podem ocorrer, foram desenvolvidas distintas formas de aprendizado para abordar tal problema, as quais são apresentadas a seguir.

2.1.1 Paradigmas de Aprendizado

Aprendizado Supervisionado. Essa forma de aprendizado refere-se ao treinamento de redes neurais artificiais em contextos em que o conjunto de dados pode ser expresso no formato $\mathcal{D} = \{(\vec{x}_k, \vec{y}_k)\}_{k=1}^{N} \subset \mathcal{X} \times \mathcal{Y}$, em que esse é composto pelas amostras $\vec{x}_k \in \mathcal{X}$ associadas aos seus respectivos rótulos $\vec{y}_k \in \mathcal{Y}$ determinados por humanos, em que $N \in \mathbb{N}_*$ é a quantidade desses pares (Jing; Tian, 2020). Em outras palavras, toda amostra possui associado a ela um rótulo determinado por um humano. Isso posto, o aprendizado supervisionado foi uma das primeiras formas de aprendizado a surgir na literatura, sendo investigado em Rumelhart, Hinton & Williams (1986), no qual se apresentou o procedimento para o treinamento de redes neurais artificias por meio do algoritmo de retropropagação, bem como em LeCun et al. (1989), em que se demonstrou uma aplicação prática dessa técnica no reconhecimento de dígitos.

Em particular, conforme apresentado em LeCun, Bengio & Hinton (2015), embora tenha existido um período na literatura em que pouco se investigou o uso de redes neurais artificiais para o aprendizado de representações, em virtude de acreditar que o procedimento de aprendizado não seria capaz de conduzir a rede a aprender adequadamente, observou-se posteriormente o oposto. Desse modo, a técnica de retropropagação de erros permanece sendo empregada até os dias atuais no treinamento de arquiteturas modernas, as quais são aplicadas, por exemplo, nos campos da visão computacional e do processamento de linguagem natural. Além disso, atualmente, embora existam distintas abordagens para conduzir o treinamento

dessas redes neurais, tem-se que esses se derivam do método original de gradiente descendente. Portanto, com base na relevância que a técnica exibe, essa é brevemente discutida a seguir.

Sejam $f_\theta \in \mathcal{F}$ um modelo de rede neural artificial, em que $\mathcal{F}$ é sua arquitetura com θ sendo seus parâmetros, e $\ell \in \mathbb{F}(\mathcal{Y}^2, \mathbb{R})$ uma função de perda cujo objetivo é mensurar a partir de um número real o erro cometido pelo modelo em uma dada amostra. Isso posto, o erro total do modelo f_θ é expresso pela seguinte fórmula, denominada risco populacional:

$$R(f_\theta) = \mathbb{E}_{(\vec{x},\vec{y})\sim P_\mathcal{D}} \left[\ell(f(\vec{x}), \vec{y})\right] , \tag{1}$$

em que $P_\mathcal{D}$ é a distribuição dos dados, e $\mathbb{E}[\cdot]$ indica o valor esperado. Entretanto, na prática a distribuição dos dados não é conhecida, de modo que, em vez disso, calcula-se o risco empírico:

$$\hat{R}(f_\theta) = \frac{1}{|\mathcal{D}|} \sum_{(\vec{x},\vec{y})\in\mathcal{D}} \ell(f_\theta(\vec{x}), \vec{y}) . \tag{2}$$

Dessa forma, embora o melhor modelo, sob o contexto em que se conhece a distribuição dados, fosse expresso por $f_\theta^* = \arg\min_{f_\theta\in\mathcal{F}} R(f_\theta)$, tem-se que esse modelo, conforme destacado por Vapnik (1999), pode ser aproximado pelo modelo que minimiza o risco empírico, isto é:

$$\hat{f}_\theta = \arg\min_{f_\theta\in\mathcal{F}} \hat{R}(f_\theta) . \tag{3}$$

No entanto, para o entendimento das condições necessárias e suficientes para que tal processo seja válido, o leitor é aconselhado a consultar o trabalho de Vapnik (1999, cáp. 2, pp. 35–57). Isso posto, utiliza-se o método do gradiente descendente para encontrar uma sequência de modelos $f_{\theta_1}, f_{\theta_2}, \ldots, f_{\theta_\eta}$, com o objetivo de, no final, obter um modelo com o menor risco empírico. Para tanto, os parâmetros θ do modelo são atualizados da seguinte forma:

$$\theta_{k+1} = \theta_k - \alpha \nabla \hat{R}(f_{\theta_k}) , \tag{4}$$

em que $\alpha > 0$ é a taxa de aprendizado, a qual estabelece o tamanho da atualização do gradiente, e $\nabla\hat{R}(\cdot)$ corresponde ao gradiente do risco empírico.

Aprendizado Semissupervisionado. Utiliza-se essa forma de aprendizado quando o conjunto de dados pode ser expresso na forma $\mathcal{D} = (\mathcal{D}_L, \mathcal{D}_U)$, em que $\mathcal{D}_L = \{(\vec{x}_k, \vec{y}_k)\}_{k=1}^{N_L} \subset \mathcal{X} \times \mathcal{Y}$ corresponde ao conjunto de dados rotulados, contendo $N_L \in \mathbb{N}_*$ pares, e $\mathcal{D}_U = \{\vec{x}_k\}_{k=1}^{N_U} \subset \mathcal{X}$ refere-se ao conjunto de dados não rotulados, com $N_U \in \mathbb{N}_*$ amostras, sendo, em geral, $N_U > N_L$. Isto é, o conjunto de dados em questão é constituído majoritariamente por dados não rotulados, havendo uma pequena parcela de amostras com rótulos anotados manualmente. Dessa forma, o objetivo do aprendizado semissupervisionado é aproveitar os dados não rotulados para ter a capacidade de compreender a estrutura da distribuição das amostras, com a finalidade de melhorar seu poder preditivo (Oliver et al., 2018).

Aprendizado Não Supervisionado. A configuração do aprendizado não supervisionado é empregue em cenários cujo conjuntos de dados é formulado por $\mathcal{D} = \{\vec{x}_k\}_{k=1}^{N} \subset \mathcal{X}$, em

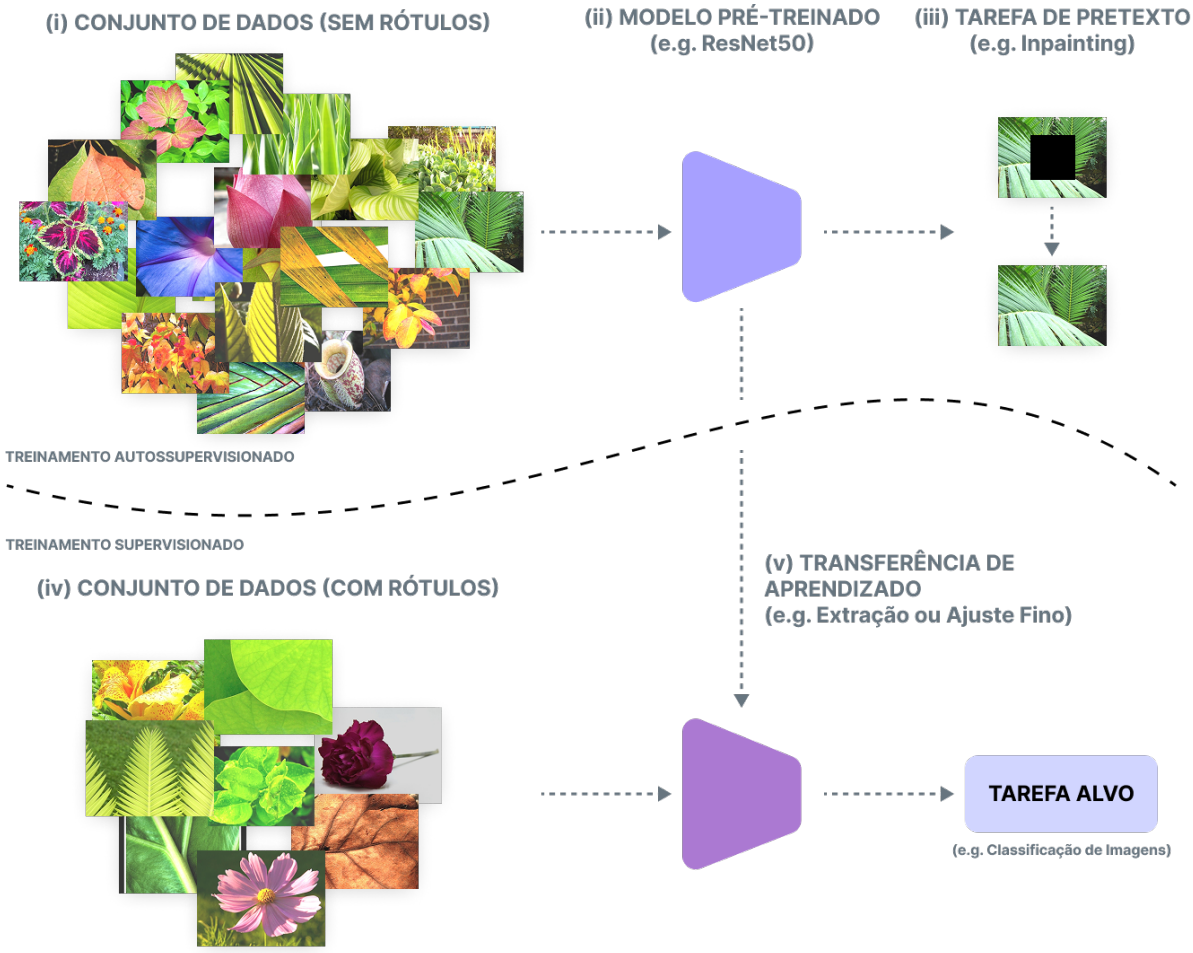
que $N \in \mathbb{N}_*$ é o número de amostras. Dessa maneira, basicamente, o conjunto de dados é composto por amostras sem rótulos. Assim, dentre os objetivos dessa forma de aprendizado está o melhor entendimento da estrutura subjacente da distribuição dos dados, e.g., utilizando algoritmos de agrupamento como *k-means* ou por meio de técnicas de redução de dimensionalidade; ou aprender representações de modo não supervisionado, para aproveitar a maior quantidade de dados não rotulados, e transferi-las posteriormente (Zhan et al., 2020).

Aprendizado Autossupervisionado. À medida que as arquiteturas de redes neurais artificias profundas com maiores capacidades de aprendizado, mais robustas e generalizáveis foram sendo propostas na literatura, esses avanços foram acompanhados pela necessidade de conjuntos de dados cada vez maiores para o treinamento dessas arquiteturas. Especificamente, no campo da visão computacional, famílias de arquiteturas, como ResNet (He et al., 2016), DenseNet (Huang et al., 2017), ViT (Dosovitskiy et al., 2021), ConvNeXt (Liu et al., 2022), BEiT (Bao et al., 2022), DeiT v3 (Touvron; Cord; Jégou, 2022) e ConvNeXt V2 (Woo et al., 2023), foram treinadas com conjuntos de dados como ImageNet (1K e 22K) (Deng et al., 2009), JFT-300M (Sun et al., 2017) e LAION (2B e 5B) (Schuhmann et al., 2022) cujos tamanhos variam de um milhão até cinco bilhões de imagens.

A aquisição de enormes conjuntos de dados demonstra-se uma operação extremamente custosa e demorada, de modo que, isso restringe a dimensão dos conjuntos de dados e, consequentemente, o treinamento de arquiteturas maiores e mais complexas, assim, limitando seu desempenho (Jing; Tian, 2020; Gui et al., 2024). Em particular, conjuntos de dados com rótulos anotados manualmente limitam-se, geralmente, a algumas unidades até poucas dezenas de milhões de imagens. Desse modo, dentre os conjuntos de dados anteriormente citados, tem-se como exemplo a ImageNet-1K, a qual possui aproximadamente um milhão de imagens divididas em 1.000 classes. Por outro lado, embora a JFT-300M com trezentas milhões de imagens também seja rotulada, os autores em Sun et al. (2017) indicam que tal processo de anotação foi realizado automaticamente com um algoritmo, possuindo um erro de aproximadamente 20% de precisão. Em relação a LAION (2B e 5B), as amostras do conjunto não apresentam exatamente um rótulo específico que as categorizem (Schuhmann et al., 2022).

Nesse sentido, o paradigma denominado aprendizado autossupervisionado (em inglês, *Self-Supervised Learning* (SSL)) tem como objetivo aprender representações a partir de enormes conjuntos de dados não rotulados (Gui et al., 2024). Para isso, os modelos são pré-treinados para a resolução das chamadas tarefas de pretexto, as quais são responsáveis por gerarem pseudorrótulos a partir das próprias amostras, os quais são utilizados como sinais supervisórios durante o treinamento, formando, assim, o mecanismo de autossupervisão. A Figura 1 ilustra esse processo para a tarefa de *inpainting*. As tarefas de pretexto permitem que o modelo aprenda a extrair representações mesmo na ausência de rótulos, uma vez que, ao ser forçado a resolver o objetivo proposto, torna-se necessário compreender a estrutura e o contexto nos quais os dados estão inseridos, a fim de viabilizar sua correta solução (Jing;

Figura 1 – Ilustração do processo do aprendizado autossupervisionado por meio do pré-treinamento do modelo utilizando uma tarefa de pretexto com posterior transferência dos pesos do modelo para aplicação nas tarefas finais.



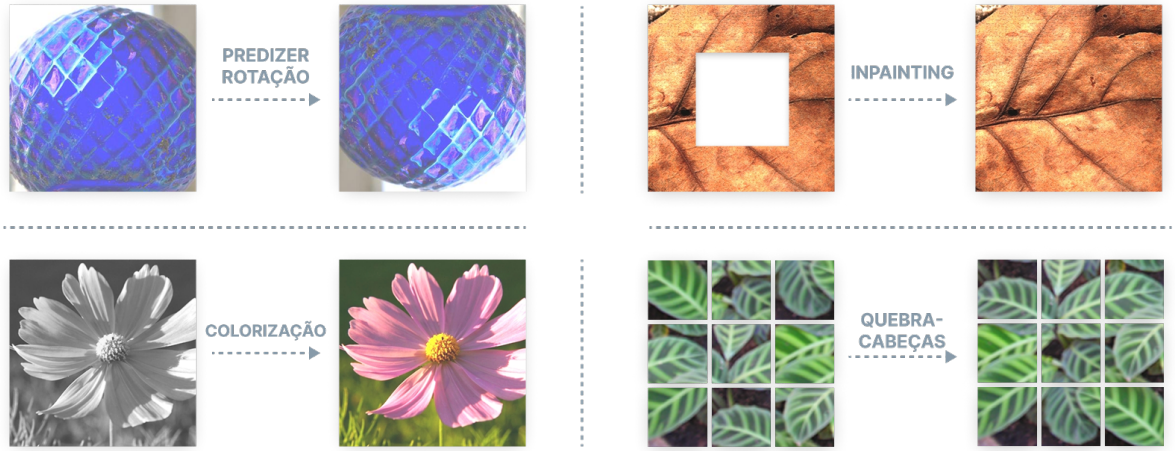
Fonte: Elaborada pelo autor. As imagens de folhagem utilizadas são provenientes do conjunto de dados FMD proposto em Sharan, Rosenholtz & Adelson (2014).

Tian, 2020). Dessa forma, uma vez pré-treinado, o modelo pode ser transferido para as tarefas finais, nas quais é empregado em um contexto supervisionado, fazendo uso de dados rotulados.

Isso posto, é evidente que as tarefas de pretexto possuem um papel fundamental dentro do paradigma de aprendizado autossupervisionado. Com base nisso, diversas investigações na literaturam propuseram distintas estratégias para a especificação dessa tarefa, tais como a predição de rotação (Gidaris; Singh; Komodakis, 2018), o preenchimento de partes faltantes da imagem (i.e., *inpainting*) (Pathak et al., 2016), a colorização de imagens (Zhang; Isola; Efros, 2016) e a resolução de um quebra-cabeças (i.e., *jigsaw puzzle*) (Noroozi; Favaro, 2016), que se encontram ilustradas na Figura 2.

Além disso, como destacado em Gui et al. (2024), outras tarefas que constroem a saída desejada intrinsecamente partir dos dados, tendo como principal exemplo os *autoencoders*, também são consideradas mecanismos autossupervisionados. Em particular, propostas

Figura 2 – Ilustrações de tarefas de pretexto comumente utilizadas durante o aprendizado autossupervisionado. As tarefas são (a) predição de rotação (b) preenchimento de regiões (c) colorização e (d) resolução de quebra-cabeças.



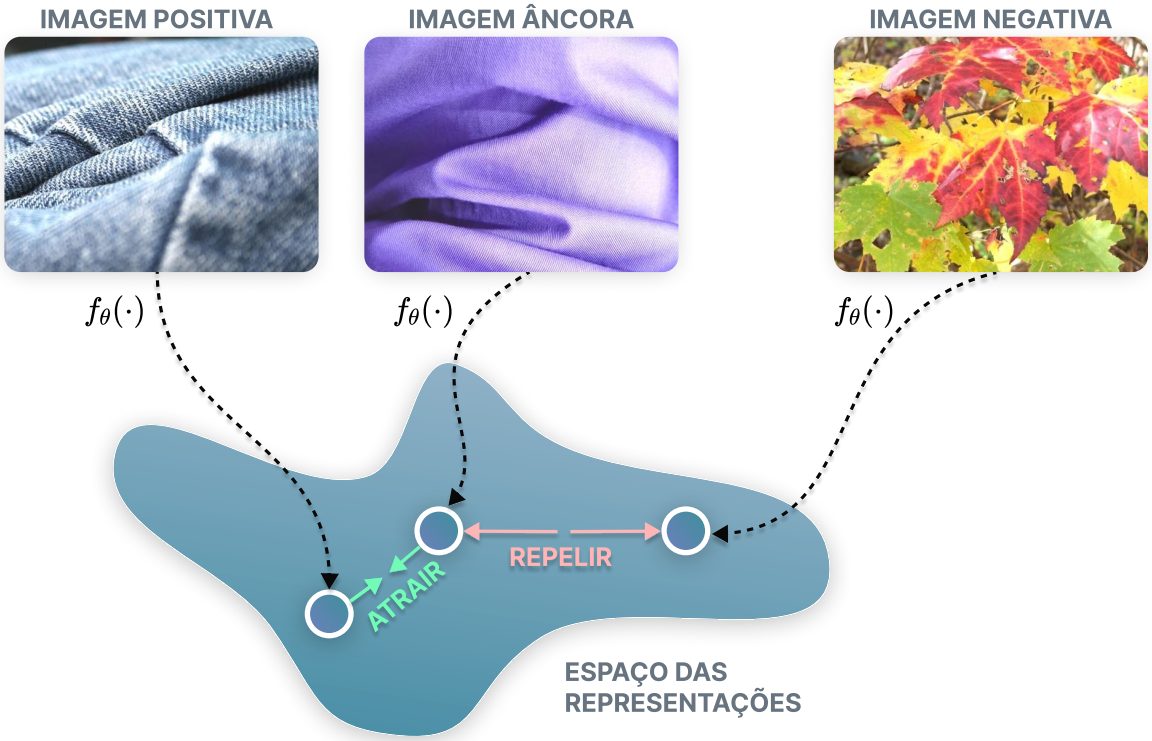
Fonte: Elaborada pelo autor.

que empregaram *autoencoders* foram em Zhang, Isola & Efros (2017), na qual os autores propuseram uma arquitetura de *autoencoder* que prediz uma parte da entrada a partir da remanescente, e nas investigações de He et al. (2022) e Chen et al. (2024). Dessa maneira, isso exibe a conexão entre os mecanismos autossupervisionados e as técnicas investigadas nesta dissertação, as quais empregaram exaustivamente *autoencoders* e técnicas de predições de relacionamentos entre diferentes partes dos dados, visando aprender melhores representações.

Em seguida, outra vertente do aprendizado autossupervisionado que obteve êxito consiste no aprendizado por contraste. Essa técnica tem como objetivo induzir o modelo a, simultaneamente, maximizar a similaridade entre representações de dados semanticamente similares e minimizar a similaridade entre representações de dados semanticamente distintos (Le-Khac; Healy; Smeaton, 2020). A ideia desse processo de aprendizado está ilustrada na Figura 3. Essa exibe as imagens âncora e positiva, denominada par positivo, e o qual corresponde às imagens que são semanticamente similares e, que no caso em questão, são imagens de tecidos. Na mesma figura, também é exibida a imagem negativa, de modo que, o par formado pela imagem âncora e negativa é denominado par negativo, o qual representa às imagens que são semanticamente distintas, nesse caso, imagem de tecido e de folhagem. A partir disso, o treinamento é guiado induzindo ao modelo fazer com que as representações provenientes do par positivo se atraiam, enquanto as representações do par negativo se afastem.

Entretanto, um problema natural que emerge no contexto de aprendizado por contraste refere-se à construção dos pares positivo e negativo, uma vez que como os dados não são rotulados, como determinar quais amostras são semanticamente similares ou distintas. Para isso, investigações em He et al. (2020) e Chen et al. (2020b) empregaram técnicas de ampliação de dados (e.g., rotações, recortes, distorções de cor etc.) aplicadas nas amostras

Figura 3 – Ilustração do conceito de aprendizado por contraste. As representações provenientes da imagem âncora e positiva (tecidos) são atraídas no espaço de representações, enquanto da imagem negativa (folhagem) é repelida em relação à imagem âncora.



Fonte: Elaborada pelo autor. As imagens positiva, âncora e negativa são provenientes do conjunto de dados FMD proposto em Sharan, Rosenholtz & Adelson (2014).

com a finalidade de construírem os pares positivos. Especificamente, em Chen et al. (2020b) para cada amostra $\vec{x}_k \in \mathcal{B}$, em que $\mathcal{B}$ é o *mini-batch* de imagens, tem-se que são obtidas amostras correspondentes $\vec{x}_{2k-1} = t(\vec{x}_k)$ e $\vec{x}_{2k} = t'(\vec{x}_k)$ a partir do emprego de duas ampliações distintas $t(\cdot)$ e $t'(\cdot)$, de modo que, essas amostras formam o par positivo, enquanto as combinações com o restante das imagens no *mini-batch* formam os pares negativos.

Enfim, o aprendizado autossupervisionado mostra-se como um paradigma de aprendizado extremamente robusto, o qual permitiu que arquiteturas de redes modernas fossem pré-treinadas adequadamente a partir de imensos conjuntos de dados não rotulados e aplicadas por meio de transferência de aprendizado em tarefas finais (e.g., análise de imagens médicas) com conjuntos mais restritos. Dessa forma, essa técnica possibilitou a expansão do limite de aprendizado dessas arquiteturas, tornando-se uma forma de aprendizado fundamental.

2.2 REDES NEURAIS ARTIFICIAIS

Esta seção apresenta as redes neurais artificias mais relevantes para esta dissertação. Em primeiro lugar, a Subseção 2.2.1 exibe a arquitetura da rede neural randomizada, a qual

representa a principal arquitetura empregada nesta dissertação. Nessa subseção, são apresentadas a sua importância, formulação matemática e sua complexidade computacional. Em seguida, a Subseção 2.2.2 apresenta a arquitetura de redes neurais convolucionais, exibindo os mecanismos biológicos em que foram inspirados, as primeiras arquiteturas propostas e seus componentes funcionais. Por fim, a Subseção 2.2.3 introduz a arquitetura de *Vision Transformers*, mostrando, principalmente, como essa ganhou enorme relevância a partir de um novo paradigma completamente distinto das redes convolucionais, por meio do mecanismo de autoatenção, que é empregado majoritariamente para processamento de linguagem natural.

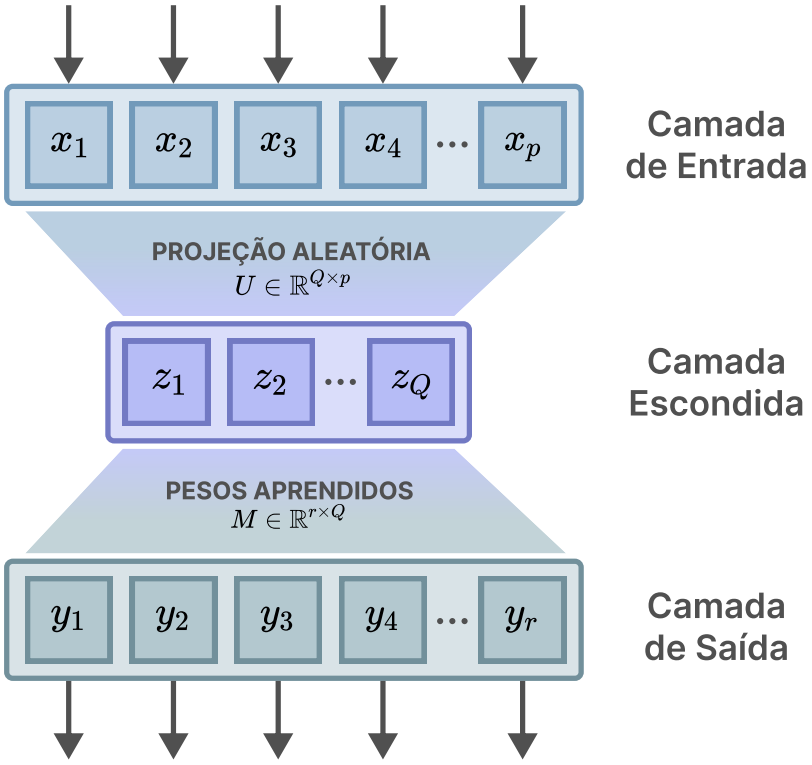
2.2.1 Redes Neurais Randomizadas

As redes neurais randomizadas investigadas em Schmidt, Kraaijveld & Duin (1992), Pao & Takefuji (1992), Pao, Park & Sobajic (1994), Huang, Zhu & Siew (2004) e Huang, Zhu & Siew (2006) detém essa nomenclatura, pois são redes geralmente compostas por uma única camada escondida cujos pesos são gerados aleatoriamente por meio de uma distribuição de probabilidades ou por um gerador de números pseudoaleatório (e.g., gerador congruencial linear), os quais permanecerão inalterados ao longo da utilização dessa rede neural. Essa camada escondida tem como finalidade realizar uma projeção aleatória dos dados de entrada em um outro espaço dimensional, geralmente maior, seguida de uma ativação não linear (Fares; Ribas, 2025a). Posteriormente, os pesos da camada de saída são aprendidos por meio de uma solução em fórmula fechada, especificamente utilizado o método dos mínimos quadrados. Desse modo, essa etapa de treinamento dispensa o uso de mecanismos de retropropagação (i.e., *backpropagation*), consistindo na execução de um pequeno conjunto de operações matriciais realizadas uma única vez, o que a torna rápida e de baixo custo computacional.

Essas redes têm sido empregadas em diversas tarefas de aprendizado de máquina, sendo essas aplicadas majoritariamente em problemas de regressão e classificação (Fares; Ribas, 2025a). No entanto, pelas características atrativas anteriormente exibidas, essas redes começaram a ser aplicadas, principalmente, em tarefas de visão computacional, tais como: análise de texturas (Sá Junior; Backes, 2016; Ribas et al., 2020; Ribas et al., 2024a; Fares; Guerra; Ribas, 2025), análise de vídeos (Sá Junior; Ribas; Bruno, 2019; Ribas et al., 2022), análise de contorno (Sá Junior; Backes; Bruno, 2018; Ribas; Bruno, 2024), análise de nuvem de pontos (Fujiwara; Hashimoto, 2020), entre outros. Em particular, em cenários de análise de textura, vídeos e contornos, a rede tem sido empregada unicamente para cada amostra (e.g., imagem de textura) com o objetivo de aprender uma representação, i.e., a partir de cada amostra são construídos os dados de entrada e as saídas esperadas da rede, a qual é então treinada e após essa etapa, os pesos aprendidos são pós-processados e utilizados como representação. Dessa forma, note-se que não há o uso de qualquer outra amostra, de modo que, a rede é capaz de aprender uma representação por instância, ou seja, para cada amostra.

Na literatura, os estudos têm investigado com sucesso redes neurais randomizadas sob

Figura 4 – Ilustração da arquitetura de rede neural randomizada, a qual é composta por camada de entrada, uma única camada escondida e camada de saída.



Fonte: Elaborada pelo autor.

duas arquiteruas, denominadas *Random Vector Functional Link* (RVFL) e *Extreme Learning Machine* (ELM). Ambas as arquiteturas realizam os mesmos passos de projeção e treinamento anteriormente descritos, entretanto, diferem na forma de conexão da camada de saída. Em relação à RVFL, os neurônios de saída estão conectados tanto aos neurônios da camada escondida quanto os neurônios de entrada. Por outro lado, em relação à ELM, os neurônios de saída estão conectados somente aos neurônios da camada escondida (Fares; Ribas, 2025a). Isso posto, uma vez que os estudos de análise de textura, vídeos e contorno utilizam, majoritariamente, a rede neural randomizada com os neurônios de saída apenas conectados com os da camada escondida, tem-se que esse será tratado durante o restante da subseção. Por fim, a Figura 4 apresenta a ilustração da rede neural randomizada.

Formulação Matemática. Seja $\mathcal{D} = \{(\vec{x}_k, \vec{y}_k)\}_{k=1}^{N} \subset \mathcal{X} \times \mathcal{Y}$ um conjunto de dados formados pelos pares de vetores de atributos de entrada $\vec{x}_k \in \mathcal{X}$, e os vetores de atributos de saída $\vec{y} \in \mathcal{Y}$, em que, geralmente, define-se $\mathcal{X} = \mathbb{R}^p$ e $\mathcal{Y} = \mathbb{R}^r$, de modo que,

$$X = \text{hconcat}(\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_N\}) \in \mathbb{R}^{p \times N} \tag{5}$$

$$Y = \text{hconcat}(\{\vec{y}_1, \vec{y}_2, \ldots, \vec{y}_N\}) \in \mathbb{R}^{r \times N}\,, \tag{6}$$

representam as matrizes resultantes da concatenação horizontal dos vetores de atributos de entrada e saída, respectivamente. Além disso, seja $U \in \mathbb{R}^{Q \times p}$ a matriz dos pesos aleatórios

da camada escondida, em que $Q \in \mathbb{N}_*$ corresponde ao número de neurônios da camada escondida. Dessa forma, uma vez com os pesos aleatórios fixos, esses são utilizados para projetar, de forma não linear, os dados de entrada no espaço Q-dimensional. Para isso,

$$Z = \alpha(UX') \in \mathbb{R}^{Q \times N}\,, \tag{7}$$

em que $\alpha(\cdot)$ representa a função de ativação não linear, aplicada elemento a elemento, a qual é escolhida, comumente, como a função sigmoide, definida por $\alpha(x) := 1/(1+e^{-x})$, e $X' \in \mathbb{R}^{p \times N}$ corresponde a matriz X após o processo de padronização, sendo definida por:

$$[X']_{ij} = \frac{[X]_{ij} - \overline{[X]}_i}{s_i + \epsilon}\,, \tag{8}$$

em que $\epsilon > 0$ corresponde a um pequeno valor para prevenir instabilidades numéricas (e.g., divisão por zero), $\overline{[X]}_i = \frac{1}{N}\sum_{j=1}^{N}[X]_{ij}$ corresponde à media amostral do i-ésimo atributo, e $s_i^2 = \frac{1}{N-1}\sum_{j=1}^{N}([X]_{ij} - \overline{[X]}_i)^2$ corresponde à variância amostral do i-ésimo atributo.

Isso posto, a rede neural randomizada segue para a etapa de treinamento, a qual consiste em resolver o seguinte problema de minimização:

Problema de Otimização da Rede Neural Randomizada

Otimização 1 (Rede Neural Randomizada)**:** Seja $g_M(Z) = MZ$ um preditor linear parametrizado pela matriz $M \in \mathbb{R}^{r \times Q}$, em que $g_M \in \mathbb{F}(\mathbb{R}^{Q \times N}, \mathbb{R}^{r \times N})$. Busca-se aprender uma matriz $\hat{M} \in \mathbb{R}^{r \times Q}$ que satisfaça o seguinte problema de minimização:

$$\hat{M} = \arg\min_{M} \|Y - g_M(Z)\|_F^2 + \lambda\,\Omega(M)\,, \tag{9}$$

em que $\|\cdot\|_F$ representa a norma de Frobenius, $\lambda > 0$ representa o parâmetro de regularização, e $\Omega \in \mathbb{F}(\mathbb{R}^{r \times Q}, \mathbb{R})$ representa o termo de regularização.

Dessa forma, sob o caso sem regularização, i.e., $\Omega(M) = 0$, tem-se que a solução em fórmula fechada do problema de minimização expresso na Equação 9 é dado por:

$$\hat{M} = YZ^{\dagger}\,, \tag{10}$$

em que $Z^{\dagger} = Z^T(ZZ^T)^{-1}$ representa a pseudoinversa de Moore-Penrose (Penrose, 1955). Além disso, conforme discutido em Fares & Ribas (2025a), existem situações em que se faz necessária a inclusão de um termo de regularização imposto sobre os parâmetros do preditor linear. Nesses casos, o termo de regularização mais comumente utilizado, e que ainda preserva a solução em fórmula fechada, é dado por $\Omega(M) = \|M\|_F^2$, o qual corresponde à regularização ℓ_2. Nesse contexto, a solução do problema de otimização na Equação 9 é expressa por:

$$\hat{M} = YZ^T(ZZ^T + \lambda I)^{-1}\,, \tag{11}$$

em que $I \in \mathbb{R}^{Q \times Q}$ representa a matriz identidade. Além disso, com o objetivo de simplificar a notação, tem-se que $\text{RNN}(X, Y; U)$ corresponde ao resultado da Equação 10, enquanto

Algoritmo 1 Treinamento da Rede Neural Randomizada (RNN)

Entrada: A matriz de entrada X, a matriz de saída Y e a matriz dos pesos U
Saída: A matriz de pesos aprendidos da camada de saída M

1: $X' \leftarrow \text{zscore}(X)$ ▷ Padronização dos vetores de atributos de entrada
2: $Z \leftarrow \alpha(UX')$ ▷ Ativação não linear da camada escondida
3: $M \leftarrow YZ^T(ZZ^T)^{-1}$ ▷ Calcula os pesos da camada de saída
4: **retorne** M

RNN$(X, Y; U, \lambda)$ corresponde ao resultado da Equação 11. Ainda mais, na modalidade de *autoencoder*, quando $Y = X$, troca-se RNN$(X, Y; U, \cdot)$ por RAE$(X; U, \cdot)$. Por fim, o Algoritmo 1 apresenta o pseudocódigo para o treinamento de uma rede neural randomizada sem regularização, sem a adição de *bias* e com *zscore* representando a Equação 8.

Complexidade Computacional. A seguir, analisa-se a complexidade computacional do algoritmo de treinamento da rede neural randomizada, com o objetivo de derivar seu limitante superior assintótico. Para isso, primeiramente, suponha que as operações matriciais de multiplicação e inversão de matrizes se apresentam em suas formas mais ingênuas, de modo que, as operações AB e C^{-1} tenham as complexidades computacionais de $O(mnp)$ e $O(m^3)$, em que $A \in \mathbb{R}^{m\times n}$, $B \in \mathbb{R}^{n\times p}$ e $C \in \mathbb{R}^{m\times m}$. Dessa forma, a partir do Algoritmo 1, pode-se definir a seguinte função de tempo de execução:

$$T(p, N, Q, r) = T_1(p, N) + T_2(p, N, Q) + T_3(N, Q, r)\,, \tag{12}$$

em que as funções de tempo de execução T_1, T_2 e T_3 referem-se aos custos da primeira, segunda e terceira linha do algoritmo, respectivamente.

Isso posto, com relação à T_1, tem-se que o custo referente ao cálculo da média dos valores para um atributo em específico é $O(N)$, e repetindo tal processo para todos os p atributos de entrada, resulta em um custo total de $O(pN)$ para a obtenção das médias amostrais de cada atributo. De modo similar, o mesmo custo $O(pN)$ é necessário para o cálculo das variâncias amostrais de cada atributo. Em seguida, cada valor da matriz é atualizado com a Equação 8, de modo que, tal custo também é $O(pN)$. Nesse contexto, com certo abuso de notação, tem-se que $T_1(p, N) = O(pN) + O(pN) + O(pN) = O(pN)$. Em relação à T_2, tem-se que a multiplicação matricial $UX' \in \mathbb{R}^{Q\times N}$, em que $U \in \mathbb{R}^{Q\times p}$ e $X' \in \mathbb{R}^{p\times N}$ possui o custo de $O(QpN)$, enquanto que a aplicação da função de ativação $\alpha(\cdot)$ possui o custo de $O(QN)$, desde que é aplicada elemento a elemento no argumento de entrada. Dessa forma, tem-se que a complexidade computacional é dada por $T_2(p, N, Q) = O(QpN) + O(QN) = O(QpN)$.

Por outro lado, o custo em relação a T_3 será dividido em três partes, primeiramente, tem-se que a multiplicação matricial YZ^T possui custo $O(rNQ)$, desde que $Y \in \mathbb{R}^{r\times N}$ e $Z^T \in \mathbb{R}^{N\times Q}$. Em seguida, a multiplicação matricial ZZ^T seguida de sua inversão possui o custo $O(QNQ) + O(Q^3) = O(Q^2N)$, sob a hipótese de que $N > Q$. Desse modo, resta o custo da multiplicação matricial entre YZ^T e $(ZZ^T)^{-1}$ cujo custo computacional é $O(rQ^2)$.

Assim, o custo computacional de T_3 é dado por $T_3(N, Q, r) = O(rNQ)+O(Q^2N)+O(rQ^2) = O(rNQ) + O(Q^2N) = O(NQ(Q + r))$. Assim, tem-se que o tempo de execução total é:

$$T(p, N, Q, r) = O(pN) + O(QpN) + O(NQ(Q + r)) = O(NQ(p + Q + r)) . \quad (13)$$

Por fim, note-se que uma vez que p, Q e r são mantidos fixos, tem-se que o tempo de execução dependerá linearmente em N, exibindo, assim o seu bom custo computacional.

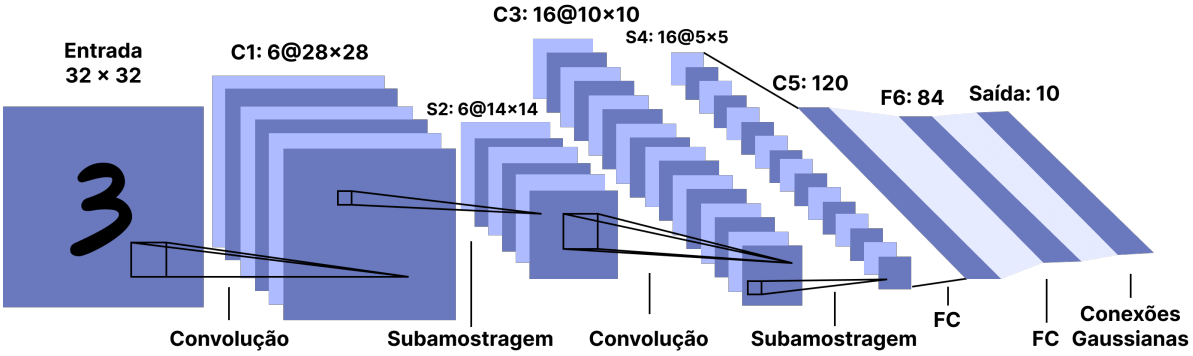
2.2.2 Redes Neurais Convolucionais

As redes neurais convolucionais são redes neurais artificiais formadas por diferentes componentes funcionais e cujo principal mecanismo de operação é a convolução. Essas redes se inspiraram no funcionamento do córtex visual de animais (Gu et al., 2018; Li et al., 2021; Cong; Zhou, 2023), baseados na investigação de Hubel & Wiesel (1959), os quais exibiram que estímulos luminosos em forma de fenda, projetados no campo receptivo das células do córtex estriado de gatos, respondem diferentemente à medida que esses estímulos variam em forma, tamanho e orientação. Nesse contexto, a operação de convolução corresponde artificialmente a esse mecanismo a partir da ativação de pequenas regiões (i.e., janelas, trechos ou *patches*), representando o campo receptivo, que varrem a imagem completamente, buscando entender como as regiões estão ativando a depender do filtro (i.e., *kernel*) desta operação de convolução, o qual é aprendido automaticamente por meio do aprendizado por gradiente descendente.

As primeiras ocorrências mais proeminentes de redes neurais convolucionais na literatura foram em Fukushima (1980) e LeCun et al. (1989). A primeira investigação apresenta uma arquitetura de rede neural fortemente influenciada pelo modelo hierárquico de visão proposto por Hubel & Wiesel (1962), de modo que, a arquitetura proposta é capaz de reconhecer diferentes estímulos (ou padrões) com invariância posicional, i.e., o reconhecimento do estímulo é pouco afetado por sua posição. Por outro lado, a segunda investigação propõe uma arquitetura convolucional composta por cinco camadas para o reconhecimento de dígitos escritos à mão sendo treinada por meio do algoritmo de retropropagação em uma abordagem supervisionada. À título de ilustração, a Figura 5 apresenta a arquitetura da rede neural convolucional, denominada por LeNet-5, proposta em LeCun et al. (1998) para o reconhecimento de dígitos.

Nessa figura, a imagem de entrada, de dimensão 32×32, é submetida à primeira camada convolucional C_1, a qual contém seis filtros (ou *kernels*) de dimensão 5×5. Isso posto, quando tais filtros são aplicados à imagem por meio da operação de convolução, esses filtros produzem seis mapas de características (i.e., *feature maps*) de dimensão 28×28, cuja informação é expressa na figura pela notação 6@28×28. Em seguida, os mapas de características resultantes são submetidos a camada de subamostragem S_2, a qual varre a imagem com um campo receptivo de 2×2 sem sobreposição, de modo que, os mapas de características resultantes possuem dimensão espacial 14×14 (LeCun et al., 1998). Posteriormente, tais mapas de características são submetidos à camada convolucional C_3 e à camada de subamostragem

Figura 5 – Ilustração da arquitetura de rede neural convolucional LeNet-5 proposta em LeCun et al. (1998) para o reconhecimento de dígitos.



Fonte: Adaptada de LeCun et al. (1998). © 1998 IEEE.

S_4 e, novamente, a camada convolucional C_5. Ao fim da rede, um conjunto de camadas totalmente conectadas tem a responsabilidade de transformar a representação de alto nível com 120 dimensões, em uma saída com apenas 10 conexões, com o objetivo de indicar o número que a rede convolucional reconheceu como presente na imagem de entrada.

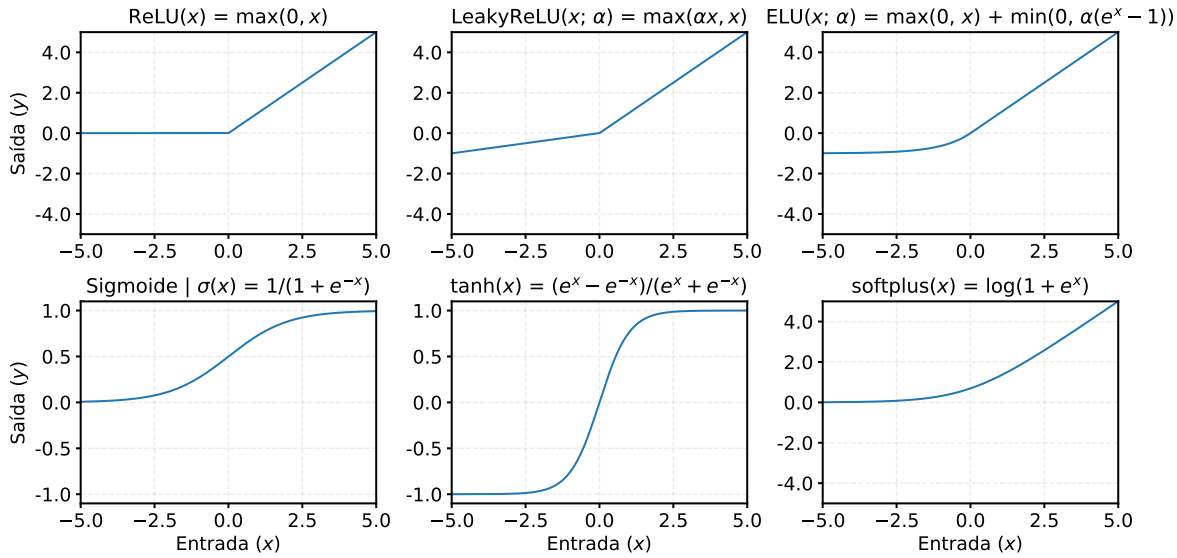
Por fim, embora muitos estudos têm continuadamente utilizado as redes neurais convolucionais para a resolução de problemas, tem-se que essa adoção em massa foi, em grande parte, possível por três razões. Em primeiro lugar, o maior poder de processamento possibilitado por aceleradores de *hardware*, tal como unidades de processamento gráficas (em inglês, *Graphical Processing Units* (GPUs)), as quais permitem processar grandes quantidades de dados simultaneamente e de forma rápida; e com o crescimento da Internet, permitiu-se construir maiores conjuntos de dados para o treinamento dessas redes (Gu et al., 2018; Cong; Zhou, 2023). Em segundo lugar, tem-se que a popularização das redes neurais convolucionais deve-se ao estudo de Krizhevsky, Sutskever & Hinton (2012), em que os autores propuseram uma arquitetura convolucional, posteriormente denominada por AlexNet, que venceu a competição de reconhecimento de imagens da ImageNet. Em terceiro lugar, destacam-se as bibliotecas de código aberto, como TensorFlow (Abadi et al., 2016) e PyTorch (Paszke et al., 2019), as quais facilitam a implementação e o treinamento (e.g., via gradiente descendente) dessas arquiteturas, abstraindo do usuário as complexidades inerentes desses processos.

Componentes Funcionais. A seguir, são apresentados os componentes funcionais mais comumente presentes em diversas arquiteturas de redes neurais convolucionais.

- **Camada Convolucional**: Essa camada possui a responsabilidade de aplicar um conjunto de filtros à imagem de entrada ou aos mapas de características da camada anterior por meio da operação de convolução. Em particular, os hiperparâmetros mais comuns dessa camada são, $C_{\text{out}} \in \mathbb{N}_*$ que corresponde à quantidade de filtros, $K_H \in \mathbb{N}_*$ que corresponde à altura do filtro, e $K_W \in \mathbb{N}_*$ que corresponde à largura do filtro.

  Isso posto, seja $X \in \mathbb{R}^{C_{\text{in}} \times H_{\text{in}} \times W_{\text{in}}}$ a entrada da camada convolucional, e seja $Y \in$

Figura 6 – Gráficos de funções de ativação não lineares. Exibem-se as funções ReLU, LeakyReLU, ELU, Sigmoide, tanh e softplus.



Fonte: Elaborada pelo autor.

$\mathbb{R}^{C_{\text{out}} \times H_{\text{out}} \times W_{\text{out}}}$ o mapa de características resultante (i.e., a saída dessa camada), desse modo, tem-se que o k-ésimo canal $[Y]_{k,:,:}$ representa o resultado da convolução da entrada X com o k-ésimo filtro $F_k \in \mathbb{R}^{C_{\text{in}} \times K_H \times K_W}$, com $k \in \{1, 2, \ldots, C_{\text{out}}\}$.

Assim, a finalidade é que por meio da utilização de múltiplos filtros, os quais são treinados, a camada convolucional possa extrair mais características presentes nas informações e abstraí-las cada vez mais à medida que aprofunda-se na rede (LeCun et al., 1989).

- **Camada de Ativação**: A camada de ativação é responsável por receber a saída da camada anterior, denotada por $X \in \mathbb{R}^{C \times H \times W}$, e aplicar uma função de ativação não linear $\alpha \in \mathbb{F}(\mathbb{R}, \mathbb{R})$ a cada elemento de X, de modo que, a saída da camada de ativação, denotada por $Y \in \mathbb{R}^{C \times H \times W}$, tem seus elementos definidos por:

$$[Y]_{i,j,k} = \alpha([X]_{i,j,k})\,. \tag{14}$$

A importância de empregar funções de ativações não lineares, como ReLU, sigmoide, tanh, softplus, LeakyReLU (Maas et al., 2013), ELU (Clevert; Unterthiner; Hochreiter, 2016), entre outras, reside no fato de que elas introduzem não linearidades na rede, ampliando, assim, sua capacidade de aprendizado (Li et al., 2021). Como exemplo, se as funções de ativação fossem lineares, então uma rede de profundidade arbitrária, composta por camadas totalmente conectadas, poderia ser representada por uma rede com uma única camada, de modo que, a profundidade, nesse caso, não produz efeito. A Figura 6 apresenta os gráficos de algumas funções de ativação não lineares.

- **Camada de Subamostragem**: A camada de subamostragem, também conhecida como camada de *pooling* ou *downsampling*, realiza a redução da dimensão espacial dos mapas

de características, geralmente, por meio da aplicação de uma função de *pooling* sobre os valores de janelas de dimensão $J_1 \times J_2$ que varrem a imagem sem sobreposição.

Nesse contexto, a função de *pooling* tem como finalidade obter um valor representativo da região local definida pela janela, em que, mais comumente, a saída dessas funções correspondem ao valor máximo (i.e., *max pooling*) da região ou a média de seus valores (i.e., *average pooling*).

- **Camada Totalmente Conectada**: O término de uma arquitetura de rede neural convolucional é, geralmente, composto por um conjunto de camadas totalmente conectadas, cuja finalidade é receber a representação abstrata de alto nível obtida a partir das camadas convolucionais e transformá-la em um vetor $\vec{z} \in \mathbb{R}^N$, em que $N \geq 2$ é o número de classes, que será mapeado em uma distribuição de probabilidades para a posterior determinação da classe predita pela rede (Zhang et al., 2019).

Por completude, em um contexto de classificação multiclasse com N classes, sejam $\vec{x}_i$ uma amostra de entrada e f_θ um modelo de rede neural, em que θ é os parâmetros da rede. Dessa forma, uma vez que o modelo tomar $\vec{x}_i$ como entrada, tem-se que esse produzirá como saída a classe predita $f_\theta(\vec{x}_i) \in \{1, 2, \ldots, N\}$ para tal amostra. Para que isso ocorra, considere que $\vec{z}_i \in \mathbb{R}^N$ seja a saída da última camada totalmente conectada. Essa saída é então submetida à função *softmax*, a qual transforma os valores de $\vec{z}_i$ em probabilidades interpretáveis (Gu et al., 2018), resultando no vetor de saída $\vec{s}_i \in [0, 1]^N$, com:

$$[\vec{s}_i]_j = \frac{e^{[\vec{z}_i]_j}}{\sum_{k=1}^{N} e^{[\vec{z}_i]_k}}\,, \tag{15}$$

em que $[\vec{s}_i]_j$ indica a probabilidade da amostra $\vec{x}_i$ pertencer à classe $j \in \{1, 2, \ldots, N\}$. Nesse sentido, seleciona-se como saída a classe associada à maior probabilidade, isto é:

$$f_\theta(\vec{x}_i) = \underset{j \in [1,N]_{\mathbb{N}}}{\arg\max}\, [\vec{s}_i]_j\,. \tag{16}$$

Transferência de Aprendizado. A transferência de aprendizado refere-se à técnica ou procedimento que emprega os pesos de uma rede neural artificial pré-treinados em um conjunto de dados fonte para a resolução de uma tarefa-alvo; e os utiliza para a solução de uma tarefa-alvo em um conjunto de dados alvo, com a finalidade de aproveitar o conhecimento adquirido pela rede durante a resolução da tarefa-fonte para a solução da tarefa-alvo e, desse modo, possivelmente, melhorar seu desempenho (Yamashita et al., 2018; Zhuang et al., 2020).

Especificamente, afirma-se que o desempenho pode, possivelmente, melhorar, uma vez que a transferência dos pesos pré-treinados em um conjunto de dados fonte para a resolução de uma tarefa-alvo pode a afetar negativamente, ocorrendo o fenômeno denominado por transferência negativa (Zhuang et al., 2020). Especificamente, o fenômeno em questão pode ocorrer em função de diversos fatores inerentes aos domínios considerados, ao contexto em

que se encontram inseridos, bem como à forma e à extensão dos pesos transferidos para o modelo alvo, entre outros aspectos relevantes (Yosinski et al., 2014; Zhuang et al., 2020).

Em particular, dentre esses aspectos, tem-se que com relação aos domínios considerados, a transferência negativa pode ocorrer com mais proeminência, principalmente, em situações nas quais o domínio-alvo difere substancialmente do domínio-fonte, de modo que, os pesos pré-treinados pouco contribuirão para a resolução da tarefa-alvo (Yamashita et al., 2018; Zhuang et al., 2020). Nesse caso, em específico, é dito que os pesos pré-treinados apresentam baixa transferibilidade da tarefa-fonte para a tarefa-alvo, de modo que, para contornar tal questão outros procedimentos são necessários para aprimorar sua transferibilidade.

Por outro lado, a depender da extensão da transferência dos pesos pré-treinados, i.e., se apenas as primeiras camadas são transferidas e congeladas, enquanto as últimas são treinadas, ou se todas as camadas são transferidas e congeladas; pode-se ocorrer a transferência negativa, uma vez que as características aprendidas em diferentes profundidades da rede neural convolucional no domínio-fonte podem não se mostrar úteis o suficiente no domínio-alvo. Especificamente, isso deve-se ao fato de que as camadas iniciais de redes neurais convolucionais tendem a extrair características mais genéricas, enquanto que as camadas finais tendem a extrair características mais específicas do domínio-fonte (Yosinski et al., 2014).

Além disso, a transferência de aprendizado é empregada em cenários nos quais o domínio-alvo é composto por conjuntos de dados ínfimos, como em contextos de análise de imagens médicas, nos quais a limitação na obtenção das amostras decorre, em geral, do elevado custo de aquisição dos próprios dados ou, quando esse não é o caso, do custo associado ao processo de anotação, o qual constitui um procedimento laborioso e que, ao depender do contexto, demanda pessoal especializado (Yamashita et al., 2018). Dessa forma, a seguir, são apresentadas as duas estratégias de transferência de aprendizado comumente utilizadas:

- **Extração de Características**: A transferência de aprendizado pelo método de extração de características refere-se ao emprego de uma rede neural convolucional pré-treinada, com seus pesos mantidos congelados. Isso posto, torna-se possível empregar a rede congelada de duas maneiras distintas. Em primeiro lugar, substituem-se as camadas totalmente conectadas pré-treinadas por novas camadas inicializadas aleatoriamente, as quais são treinadas especificamente para ajustar a cabeça de classificação para a tarefa-alvo, sem que se realize qualquer modificação sobre os pesos das camadas anteriores, i.e., mantendo a rede extratora, a qual é composta majoritariamente pelas camadas convolucionais, intacta (Yamashita et al., 2018). Por outro lado, pode-se remover as camadas totalmente conectadas e acoplar a saída da rede extratora diretamente a classificadores tradicionais (e.g., LDA, KNN e SVM) (Scabini et al., 2023).
- **Ajuste Fino**: A transferência de aprendizado pelo método de ajuste fino (em inglês, *fine-tuning*) é caracterizada pelo uso de uma rede neural convolucional pré-treinada, a qual é adaptada para a resolução da tarefa-alvo por meio do treinamento adicional ou

recalibração dos parâmetros de um subconjunto de camadas, ou mesmo de todas as camadas da rede. Além disso, observa-se que, da mesma maneira como na primeira forma da técnica por extração de características, se faz necessária a substituição da cabeça de classificação da rede original por uma nova cabeça de classificação, de modo que esta seja treinada para a tarefa-alvo (Yosinski et al., 2014; Yamashita et al., 2018).

Por fim, embora as formas de transferência de aprendizado discutidas, i.e., extração de características ou ajuste fino, tenham sido apresentadas no contexto de redes neurais convolucionais, observa-se que tais abordagens também são aplicáveis a outras arquiteturas de redes neurais, como os *Vision Transformers*, os quais são apresentados a seguir.

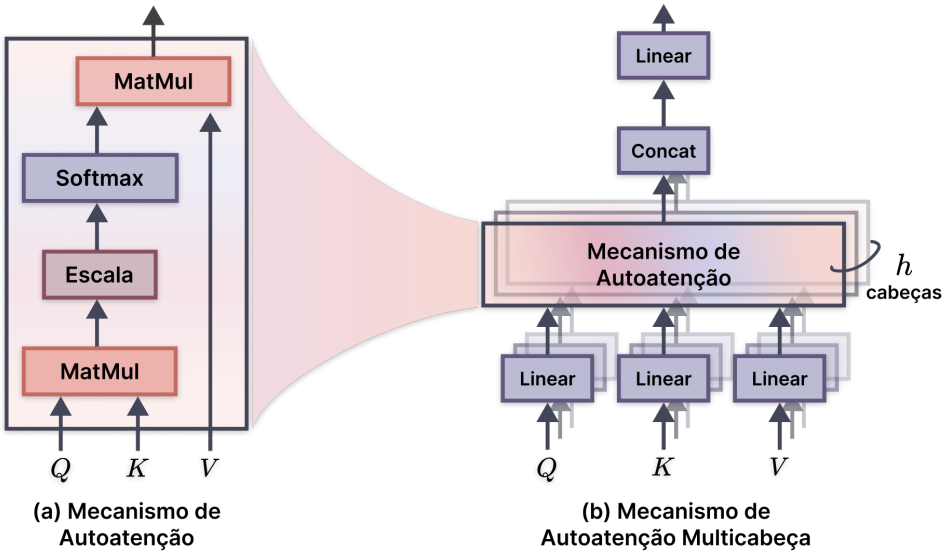
2.2.3 Vision Transformers

O *Vision Transformer* (ViT) consiste em uma nova arquitetura de rede neural artificial proposta em Dosovitskiy et al. (2021) empregada para classificação de imagens, a qual alcançou ampla notoriedade por adotar um paradigma de funcionamento totalmente distinto daquele observado nas redes neurais convolucionais, uma vez que estas últimas empregam múltiplas camadas convolucionais empilhadas com a finalidade de extrair representações progressivamente mais abstratas e de alto nível. Por outro lado, os ViTs empregam como principal operação o mecanismo de autoatenção o qual é capaz de capturar a forma como os elementos em uma sequência influenciam uns aos outros por meio do qual conseguem entender o contexto e extrair características globais da imagem (Han et al., 2022; Khan et al., 2022).

Especificamente, essa arquitetura de rede foi amplamente influenciada pelo sucesso da arquitetura de *Transformers* proposta em Vaswani et al. (2017) para aplicações de processamento de linguagem natural (em inglês, *Natural Language Processing* (NLP)), como análise de sentimento (Yang et al., 2023), tradução automática (Yao; Wan, 2020), entre outras. Ainda mais, como abordado e discutido em Han et al. (2022) e Khan et al. (2022), dois modelos baseados nesta arquitetura, que obtiveram os resultados do estado da arte em diferentes conjuntos de *benchmark* no momento em que foram propostos, são o BERT (Devlin et al., 2019) e o GPT-3 (Brown et al., 2020), e, atualmente, representantes de grande destaque que empregam essa arquitetura incluem os grandes modelos de língua (em inglês, *Large Language Models* (LLM)), como o GPT-4 (Achiam et al., 2023) e o Phi-4 (Abdin et al., 2024).

Nesse contexto, embora algumas investigações anteriores à proposta dos *Visions Transformers* estivessem analisando o uso do mecanismo de autoatenção em áreas da visão computacional, como nos estudos em Chen et al. (2020a) e Cordonnier, Loukas & Jaggi (2020), observa-se que os ViTs, conforme proposto em Dosovitskiy et al. (2021), exibiram-se robustos e generalizáveis em virtude dos resultados obtidos em *benchmarks* de classificação de imagens quando comparado às arquiteturas convolucionais. Entretanto, embora os ViTs tenham se mostrado competitivos em relação a essas arquiteturas, o fato de os ViTs apresen-

Figura 7 – Ilustração do (a) mecanismo de autoatenção, e do (b) mecanismo de autoatenção multicabeça utilizados nas arquiteturas baseadas em *Transformers*.



Fonte: Adaptada de Vaswani et al. (2017).

tarem um menor viés indutivo, como, por exemplo, a ausência de codificação de localidade e de invariância translacional, faz com que se torne necessário o treinamento desses modelos em grandes conjuntos de dados de imagens (Han et al., 2022; Khan et al., 2022).

Desse modo, conforme discutido em Khan et al. (2022), é comum que os ViTs sejam, inicialmente, pré-treinados em grandes conjuntos de dados (e.g., JFT-300M), a fim de que o modelo passe a capturar e compreender os conceitos inerentes às imagens, uma vez que apresenta baixo viés indutivo e, em seguida, por meio de transferência de aprendizado, que os pesos pré-treinados sejam empregues para a resolução da tarefa-alvo, de modo a permitir que o modelo se torne viável para utilização em conjuntos de dados de imagens menores.

Por fim, a seguir, são apresentados os funcionamentos dos mecanismos de autoatenção e sua variante multicabeça, as quais representam as principais operações em ambas as arquiteturas discutidas, com o objetivo de melhorar a compreensão do funcionamento tanto dos próprios mecanismos quanto da compreensão dos componentes funcionais dos ViTs.

Mecanismo de Autoatenção. Seja $X = \text{vconcat}(\{\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_N\}) \in \mathbb{R}^{N \times D}$, em que $\vec{x}_k \in \mathbb{R}^D$, uma matriz que corresponde a concatenação vertical de uma sequência de N *tokens* D-dimensionais. Em seguida, utilizam-se três matrizes de parâmetros treináveis, denotadas por $W^Q \in \mathbb{R}^{D \times D_q}$, $W^K \in \mathbb{R}^{D \times D_k}$ e $W^V \in \mathbb{R}^{D \times D_v}$, em que $d_v = d_k$, para projetar linearmente a sequência de *tokens* X em um outro espaço dimensional, de modo que são obtidas as seguintes matrizes resultantes dessa projeção

$$Q = XW^Q \in \mathbb{R}^{N \times D_q}, \qquad K = XW^K \in \mathbb{R}^{N \times D_k}, \qquad V = XW^V \in \mathbb{R}^{N \times D_v}, \tag{17}$$

as quais são denominadas por matriz das *queries*, *keys* e *values*, respectivamente. Desse

modo, de acordo com Vaswani et al. (2017), uma vez com essas matrizes obtidas, tem-se que a pontuação de autoatenção é calculada por meio da seguinte fórmula:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) V \in \mathbb{R}^{N \times D_v}, \tag{18}$$

a qual está ilustrada na Figura 7(a). Ainda, tem-se que esse mecanismo de autoatenção também é denominado por *Scaled Dot-Product Attention* (SDPA) (Vaswani et al., 2017).

Essa terminologia, conforme discutido em Han et al. (2022), é compreendida a partir da análise das etapas que compõe o cálculo desta fórmula. Em primeiro lugar, observa-se que o termo *dot product* decorre do fato de que multiplicação matricial $QK^T \in \mathbb{R}^{N \times N}$ representa o produto escalar (em inglês, *dot-product*) entre todos os pares de *tokens* do tipo *query* e *key*. Em segundo lugar, o termo *scaled* refere-se ao fato da matriz QK^T ser multiplicado pelo valor $1/\sqrt{d_k}$, o qual tem como finalidade estabilizar os gradientes (Vaswani et al., 2017).

Por fim, a função $\text{softmax}(\cdot)$, aplicada independentemente a cada linha, tem como finalidade transformar as pontuações intermediárias obtidas, de modo que se torne possível compreender, de forma explícita, a relação que um *token query* $\vec{q}_i$ estabelece com um *token key* $\vec{k}_j$. Nesse contexto, é comumente afirmado que a etapa $A = \text{softmax}(QK^T/\sqrt{d_k}) \in \mathbb{R}^{N \times N}$, isto é, ainda sem a multiplicação pela matriz V, armazena os pesos de atenção; e, consequentemente, a multiplicação por V é empregada para atualizar os valores dos *tokens values* com base nos pesos de atenção calculados, incorporando, assim, a informação contextual.

Mecanismo de Autoatenção Multicabeça. O mecanismo de autoatenção realiza a projeção dos *tokens* da sequência de entrada em apenas um único subespaço específico, de modo que, isso pode restringir a capacidade de aprendizado da arquitetura (Vaswani et al., 2017; Han et al., 2022; Liu et al., 2023). Nesse contexto, com a finalidade de permitir com que a rede compreenda e capture características ainda mais complexas entre os diferentes *tokens* presentes na sequência, emprega-se, portanto, o mecanismo de autoatenção múltiplas vezes, o qual é executado paralelamente (Khan et al., 2022). Desse modo, os *tokens* da sequência de entrada são projetados linearmente em distintos subespaços, o que possibilita à rede extrair mais informações (Vaswani et al., 2017; Liu et al., 2023).

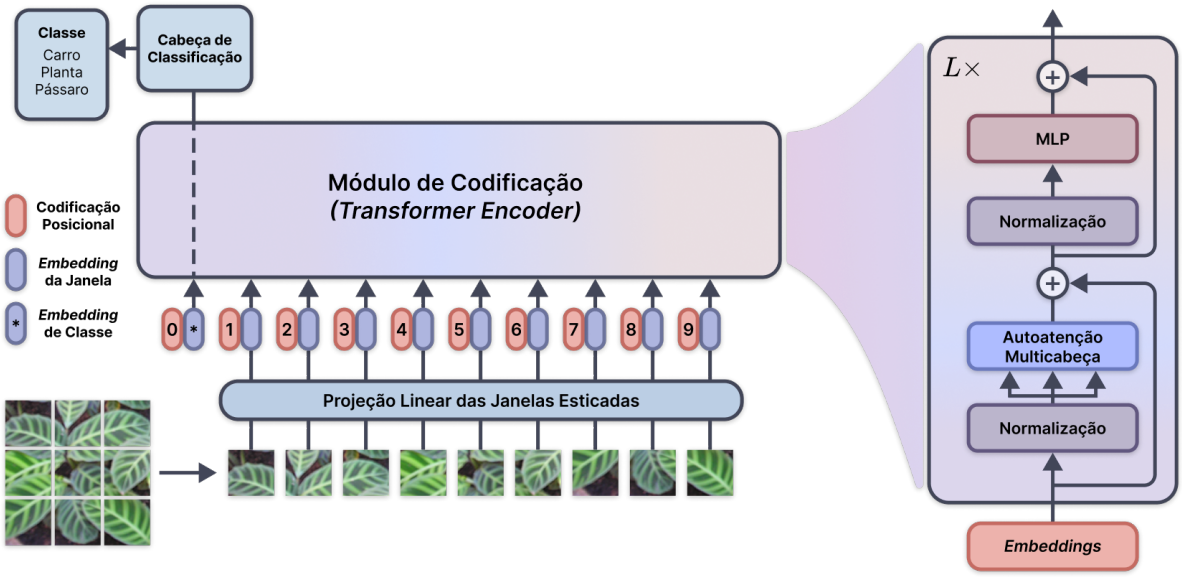
Para isso, seja $h \in \mathbb{N}_*$ o número de cabeças do mecanismo de autoatenção que são executados paralelamente. A partir disso, definem-se as dimensões dos subespaços nos quais os *tokens* da sequência de entrada são projetados como $d'_q = d'_k = d'_v = D/h$ (Vaswani et al., 2017). Isso posto, o mecanismo de autoatenção multicabeça é dado pela seguinte fórmula:

$$\text{MultiHead}(\mathcal{Q}, \mathcal{K}, \mathcal{V}) = \text{hconcat}(\{\text{head}_1, \text{head}_2, \ldots, \text{head}_h\})W^O, \tag{19}$$

$$\text{head}_i = \text{Attention}(Q_i, K_i, V_i), \tag{20}$$

em que $\mathcal{Q} = \{Q_i\}_{i=1}^h$, $\mathcal{K} = \{K_i\}_{i=1}^h$ e $\mathcal{V} = \{V_i\}_{i=1}^N$ correspondem aos conjuntos das matrizes de *queries*, *keys* e *values* resultantes da projeção linear de X por meio das matrizes aprendidas

Figura 8 – Ilustração da (a) arquitetura de *Vision Transformers* e do (b) módulo de codificação.



Fonte: Adaptada de Dosovitskiy et al. (2021).

$W_i^Q \in \mathbb{R}^{D\times d'_q}$, $W_i^K \in \mathbb{R}^{D\times d'_k}$ e $W_i^V \in \mathbb{R}^{D\times d'_v}$, com $i \in [1,h]_{\mathbb{N}}$ e $W^O \in \mathbb{R}^{D\times D}$. Esse mecanismo em questão está ilustrado na Figura 7(b). Por fim, com a finalidade de simplificar a notação desta operação, seja $\text{MSA}(X)$ a saída do $\text{MultiHead}(\mathcal{Q},\mathcal{K},\mathcal{V})$, com o conjunto de matrizes $\mathcal{Q}$, $\mathcal{K}$ e $\mathcal{V}$ obtidos pelas projetações lineares da entrada X em distintos subespaços, conforme anteriormente apontado.

Componentes Funcionais. De modo similar ao que foi realizado durante a apresentação das arquiteturas convolucionais, tem-se que, a seguir, são apresentados os componentes funcionais mais relevantes presentes em uma arquitetura de *Vision Transformers* ilustrada na Figura 8(a).

- **Camada de Entrada (Tokenização)**: Com a finalidade de alimentar o módulo de codificação do *Transformer* é necessário que uma sequência de *tokens* seja especificada. Desse modo, constata-se que não é possível alimentá-lo diretamente com uma imagem, sendo, portanto, necessário um procedimento que transforme a imagem em tal sequência.

  Para isso, seja $X \in \mathbb{R}^{C\times H\times W}$ a imagem de entrada, em que $C \in \mathbb{N}_*$ é o número de canais, $H \in \mathbb{N}_*$ é a altura da imagem e $W \in \mathbb{N}_*$ é a largura da imagem. Essa imagem é dividida em $N = HW/P^2$ janelas de dimensão $P \times P$ sem sobreposição, de modo que, ao esticar cada janela obtém-se a sequência $(\vec{x}_1, \vec{x}_2, \ldots, \vec{x}_N)$, em que $\vec{x}_k \in \mathbb{R}^{P^2C}$.

  Em seguida, essa sequência tem seus *tokens* projetados linearmente, por meio de uma matriz de parâmetros aprendidos $E \in \mathbb{R}^{P^2C\times D}$, em um espaço D-dimensional, de modo que, obtém-se a sequência $(\vec{u}_1, \vec{u}_2, \ldots, \vec{u}_N)$, em que $\vec{u}_k = \vec{x}_k E \in \mathbb{R}^D$.

  A partir dessa última sequência, tem-se que um *token* $\vec{u}_{\text{class}} \in \mathbb{R}^D$ é prefixado a sequência, com a finalidade de usá-lo, posteriormente, para a realização da classificação

por meio da alimentação deste *token* à rede totalmente conectada. Desse modo, obtém-se como resultado a seguinte sequência de *tokens*: $(\vec{u}_{\text{class}}, \vec{u}_1, \vec{u}_2, \ldots, \vec{u}_N)$.

Por fim, monta-se a matriz $U = \text{vconcat}(\{\vec{u}_{\text{class}}, \vec{u}_1, \vec{u}_2, \ldots, \vec{u}_N\}) \in \mathbb{R}^{(N+1)\times D}$ a partir da concatenação vertical de todos os vetores pertencentes à última sequência. Isso posto, adiciona-se à matriz U um codificador posicional $E_{\text{pos}} \in \mathbb{R}^{(N+1)\times D}$, o qual é responsável por manter a informação da posição dos elementos na sequência. Desse modo, obtém-se como resultado a seguinte matriz $Z = U + E_{\text{pos}}$, a qual, em formato matricial, corresponde aos $N+1$ *tokens* D-dimensionais (Dosovitskiy et al., 2021).

- **Camada de Codificação**: A partir da sequência de *tokens*, representado na forma matricial por $Z \in \mathbb{R}^{(N+1)\times D}$, obtidos da camada de entrada após a "tokenização" da imagem, tem-se que esses são utilizados como entrada para o módulo de codificação.

  De acordo com Dosovitskiy et al. (2021), este módulo, ilustrado na Figura 8(b), por sua vez, realiza as seguintes etapas sobre esta sequência:

$$Z'_\ell = \text{MSA}(\text{LN}(Z_{\ell-1})) + Z_{\ell-1}\,, \tag{21}$$

$$Z_\ell = \text{MLP}(\text{LN}(Z'_\ell)) + Z'_\ell\,, \tag{22}$$

  em que $\text{MSA}(\cdot)$ corresponde ao mecanismo de autoatenção multicabeça, $\text{LN}(\cdot)$ corresponde à etapa de normalização, $\text{MLP}(\cdot)$ corresponde a uma rede neural multicamada totalmente conectada e $\ell \in \mathbb{N}_*$ indica a camada desse codificador.

  Em particular, tem-se que a camada totalmente conectada pode ser representada por:

$$\text{MLP}(X_\ell) = \sigma(X_\ell W^1_\ell) W^2_\ell\,, \tag{23}$$

  em que W^1_ℓ e W^2_ℓ são matrizes treináveis, e $\sigma(\cdot)$ é a função de ativação não linear GELU (do inglês, *Gaussian Error Linear Units*) (Hendrycks; Gimpel, 2016).

- **Camada de Saída**: Supondo que a arquitetura em questão seja composta por $L \in \mathbb{N}_*$ camadas, tem-se que, após a sequência de *tokens* de entrada Z_0 ser submetida a uma sequência de L codificadores, a saída resultante, denotada por $Z_L \in \mathbb{R}^{(N+1)\times D}$, é utilizada na etapa de classificação. Em particular, tal saída é composta por uma sequência de $N+1$ *tokens* D-dimensionais codificados. Entretanto, apenas o primeiro *token*, denotado por $\vec{z}_L^{\,\text{class}} = [Z_L]_{1,:} \in \mathbb{R}^D$, o qual foi inserido na camada de entrada para a finalidade de classificação, é utilizado para alimentar a camada totalmente conectada.

Por fim, pela forte capacidade de aprendizado dessa arquitetura e de suas limitações existentes, tem-se que novas investigações, como em d'Ascoli et al. (2021), Touvron, Cord & Jégou (2022), Bao et al. (2022) e Chen et al. (2024), propuseram melhores formas de treinamento, uso de destilação de conhecimento, alterações arquiteturais, entre outras características. Nesse contexto, com a continuação da expansão dessa capacidade de aprendizado e da habilidade de extrair robustas representações, essas arquiteturas têm sido continuadamente

utilizadas em distintos problemas de visão computacional, como classificação de imagens (Zhu et al., 2023), detecção de objetos (Fang et al., 2023) e segmentação (Li et al., 2024).

3 TRABALHOS RELACIONADOS

Neste capítulo, são apresentados os trabalhos relacionados às metodologias investigadas e desenvolvidas nesta dissertação de mestrado. O capítulo é dividido em duas seções principais, sendo cada uma dedicada à discussão de trabalhos relacionados a tipos específicos de estratégias de aprendizado de representações com redes neurais randomizadas. Desse modo, em cada seção, são analisados os estudos mais recentes e relevantes da literatura, destacando suas contribuições científicas, limitações e relevância para as abordagens desenvolvidas nesta dissertação. Por fim, à medida que os trabalhos relacionados são apresentados e discutidos, são evidenciadas e explicadas as lacunas identificadas, bem como a forma pela qual os métodos propostos as preenchem, contribuindo, assim, para o avanço da literatura.

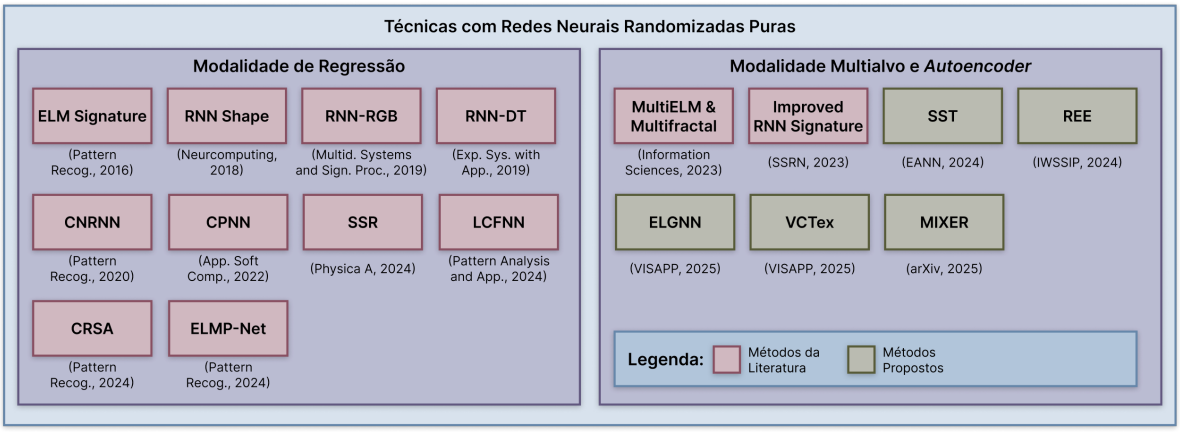
3.1 APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA COM REDES NEURAIS RANDOMIZADAS PURAS

3.1.1 Estratégias de Aprendizado Autossupervisionado e Aprendizado de Representações com Redes Neurais Randomizadas Puras

Esta subseção divide a apresentação das estratégias em dois tópicos principais. Em primeiro lugar, são apresentadas as técnicas relacionadas ao aprendizado de representações com redes neurais randomizadas puras, i.e., que não utilizam *backbone* durante o processo de aprendizado. Em particular, essas técnicas podem ocorrer em duas modalidades denominadas por regressão e multialvo, de modo que a Figura 9 ilustra a categorização das técnicas citadas. Em segundo lugar, são apresentadas as técnicas de aprendizado autossupervisionado que estão intimamente relacionadas às redes neurais randomizadas empregadas para o desenvolvimento das metodologias. Isso posto, em ambos os tópicos são apresentadas as contribuições de cada tipo de estratégia, e, em seguida, são apresentadas as lacunas preenchidas pelas abordagens propostas.

Aprendizado de Representações com Redes Neurais Randomizadas. Até onde se tem conhecimento, Sá Junior & Backes (2016) propuseram a primeira técnica de aprendizado de representações de textura em nível de cinza com redes neurais randomizadas na modalidade de regressão. Em particular, para cada pixel da imagem de entrada, os autores extraem uma vizinhança em relação ao pixel e usam as intensidades dos pixels presentes na vizinhança como entrada $\vec{x}$, enquanto que a intensidade do pixel central era usada como saída alvo y. Dessa maneira, o objetivo foi aprender representações forçando a rede a predizer a intensidade do pixel central com base somente nas intensidades de sua vizinhança. Além disso, os autores empregaram múltiplos tamanhos de vizinhança com a finalidade de analisar regiões maiores

Figura 9 – Lista das técnicas citadas de aprendizado de representações com redes neurais randomizadas puras categorizadas com base na modalidade empregada.



Fonte: Elaborada pelo autor.

da imagem de entrada. Posteriormente, em Sá Junior, Backes & Bruno (2019), os autores estenderam essa técnica para análise de texturas de imagens coloridas.

Em sequência, Ribas et al. (2020) apresentaram a primeira técnica de aprendizado de representações de textura com RNNs combinando-a com redes complexas. Em particular, os autores modelam a imagem de textura em nível de cinza em múltiplas redes complexas, em que cada pixel torna-se um vértice de cada rede complexa. A partir disso, para um dado pixel, o vetor de entrada $\vec{x}$ correspondente refere-se aos graus de saída do vértice que modela o pixel nas múltiplas redes complexas, enquanto o valor de saída y corresponde à intensidade do pixel associado. Esse processo é repetido para outras medidas, tais como força de entrada e força de saída. Nesse contexto, o objetivo consiste em explorar a dinâmica da rede complexa, a partir da variação do parâmetro da rede, com a finalidade de predizer a intensidade do pixel em questão exclusivamente por meio de características topológicas da rede.

Posteriormente, Florindo & Neckel (2023) construíram uma técnica de aprendizado que combina redes neurais randomizadas multialvo com uma abordagem multifractal, na qual o processo de extração de representações ocorre tanto sobre a imagem original quanto a imagem após a transformada fractal. Em particular, para o treinamento da rede, os autores definem o vetor de entrada $\vec{x}$ a partir das intensidades dos pixels em uma dada vizinhança, enquanto que $\vec{y}$ corresponde às intensidades dos pixels associadas a vizinhanças maiores. Assim, a rede será forçada a aprender as relações entre as intensidades de uma vizinhança em relação a outra para ambas as imagens original e transformada, respectivamente, assim aprimorando a discriminabilidade da representação de textura. Além disso, sob a mesma perspectiva de redes multialvo, Sá Junior & Backes (2023) propuseram, até onde se tem conhecimento, a primeira técnica que utilizou um *autoencoder* randomizado para a construção de representações de textura para imagens em nível de cinza, em que a construção dos vetores de entrada e saída se basearam no emprego das intensidades dos pixels nas bordas de cada vizinhança.

Em seguida, novamente com o intuito de obter representações discriminativas a partir da análise da dinâmica de redes complexas, Ribas et al. (2024a) empregaram uma modelagem de rede complexas multicamadas espaço-espectral, na qual modela-se não apenas as características espaciais (i.e., texturais) da imagem, mas também suas características espectrais (i.e., relações cromáticas entre os canais). A modelagem em questão resulta em três redes complexas, denominadas por B, W e N, em que B corresponde à rede complexa formada exclusivamente pelas interações entre os vértices de um mesmo canal, W é resultante das interações entre vértices presentes em canais distintos, enquanto que N agrega as interações intra- e inter-canal. Dessa forma, em um caso particular, para cada pixel e, consequentemente, para cada vértice, o vetor de entrada $\vec{x}$ é composto pelos graus de saída desse vértice nas diferentes redes complexas obtidas a partir da variação do raio de conexão $1, 2, \ldots, R-1$, enquanto que a saída y é definida pelo grau do vértice na rede complexa construída com raio de conexão R. Assim, como anteriormente, esse processo é repetido para as diferentes redes B, W e N com diferentes medidas topológicas. Portanto, os autores forçam a rede a predizer, por exemplo, o grau de saída de um vértice em uma rede com raio de conexão R, utilizando somente as informações dos graus de saída desse mesmo vértice obtidas para raios de conexão menores, assim, guiando a rede a aprender a dinâmica de evolução da rede complexa.

Ainda, Ribas et al. (2024b) propuseram não analisar a dinâmica da rede complexa, mas o comportamento das suas características e comportamentos locais. Para isso, dada uma imagem de entrada em nível de cinza, essa é modelada em uma rede complexa em que janelas de dimensão 3×3 serão centralizadas em cada vértice da rede, de modo que, para cada vértice, constrói-se o vetor de entrada $\vec{x}$ como sendo os graus de saída dos vértices na borda da janela, enquanto y refere-se ao grau de saída do vértice central. Nesse contexto, os autores guiam o aprendizado da rede forçando-a predizer o grau de saída do vértice central utilizando os graus de saída dos vértices vizinhos. Dessa forma, busca-se aprender boas representações a partir da análise do relacionamento local de cada vértice, i.e., da análise de sua vizinhança. Além disso, com a finalidade de enriquecer a representação, os autores repetiram tal processo empregando outras medidas topológicas, tais como: força de saída e força de entrada.

Além disso, em outra vertente, Florindo, Backes & Neckel (2024) propuseram uma técnica de extração de representações baseada na aplicação de sucessivas transformações locais sobre a imagem. Em particular, dada uma imagem de textura em nível de cinza, para cada pixel dessa, obtém-se uma vizinhança $K \times K$, sendo $K \in \mathbb{N}$ ímpar, de tal forma que as intensidades dos pixels na borda da vizinhança são empregues para compor o vetor de entrada $\vec{x} \in \mathbb{N}^{K^2-1}$, enquanto que a saída $y \in \mathbb{N}$ é composta pela intensidade do pixel central. A partir disso, os autores treinam a rede neural randomizada e usam os pesos aprendidos para projetar a matriz resultante da saída da camada escondida. Essa projeção resulta em uma nova imagem, que a depender do valor de K, pode descartar detalhes visuais, mas ainda mantendo suas características globais (Florindo; Backes; Neckel, 2024). Assim, a partir de sucessivas aplicações desse processo, cada imagem projetada é submetida à extração de características

utilizando somente os componentes de sinal e magnitude do CLBP (Guo; Zhang; Zhang, 2010a), bem como os pesos aprendidos a partir do treinamento da rede neural randomizada.

Nesse contexto, a partir desses estudos, observa-se que a maior parte das abordagens para aprendizado de representações com redes neurais randomizadas existentes na literatura, tais como Sá Junior, Ribas & Bruno (2019), Ribas et al. (2022), Ribas & Bruno (2024), Ribas et al. (2024a), Florindo, Backes & Neckel (2024) as empregaram na forma de regressão, em que as abordagens referidas arquitetam o processo de aprendizado, de modo que, a entrada da rede é um vetor $\vec{x} \in \mathbb{R}^p$ e a rede é forçada a predizer um número real $y \in \mathbb{R}$, que representa a saída esperada da rede neural randomizada. Dessa forma, ao guiar a rede, sob a modalidade de regressão, a predizer diferentes relacionamentos entre $\vec{x}$ e y, implicitamente a forçará a embutir o relacionamento aprendido nos pesos da camada de saída da RNN, os quais são utilizados para compor a representação de textura. A partir disso, é evidente que muitos estudos investigaram formas distintas para moldar tal relacionamento aproveitando a capacidade de expressão da rede. Assim, os métodos citados obtiveram êxito em suas respectivas áreas, tais como análise de contornos (Ribas; Bruno, 2024), análise de vídeos (Sá Junior; Ribas; Bruno, 2019; Ribas et al., 2022) e análise de texturas (Ribas et al., 2024a; Florindo; Backes; Neckel, 2024).

Isso posto, nas investigações discutidas, identifica-se uma primeira lacuna na literatura no que diz respeito da pouca investigação do emprego de redes neurais randomizadas em sua forma multialvo e, sobretudo, como *autoencoders* randomizados. Nesses casos, a ideia consiste em que os métodos especifiquem não apenas o vetor de entrada $\vec{x} \in \mathbb{R}^p$, mas também o vetor de saída $\vec{y} \in \mathbb{R}^r$, i.e., múltiplos valores de saída alvo para uma mesma amostra. Assim, uma vez que permitir que a rede explore, de forma conjunta, múltiplas relações entre os vetores de entrada $\vec{x}$ e de saída $\vec{y}$ pode-se potencialmente ampliar sua capacidade de aprendizado e, consequentemente, obter melhores pesos para a construção da representação de textura.

Por fim, embora as abordagens Florindo & Neckel (2023) e Sá Junior & Backes (2023) tenham dado os primeiros passos no emprego de redes neurais randomizadas multialvo e *autoencoders* randomizados, respectivamente, observa-se que, por se tratar de uma modalidade ainda pouco investigada, há amplo espaço para o desenvolvimento de novas abordagens. Ademais, observa-se potenciais ainda não explorados, tais como: (i) novas formas de construção do conteúdo de entrada e saída da rede com foco na caracterização microtextural para a aplicação em imagens de biossensores; e (ii) mudanças arquiteturais na rede neural randomizada focada ao aprendizado de representações de textura. Desse modo, os métodos desenvolvidos nesta dissertação foram construídos com base em preencher essas lacunas.

Aprendizado de Representações Autossupervisionado. Zhang, Isola & Efros (2017) argumentam que *autoencoders* tradicionais podem ser ineficientes para abordar novas tarefas por meio de transferência de aprendizado. Em particular, os autores discutem que, embora a etapa de gargalo (em inglês, *bottleneck*) seja necessária para impedir que a rede aprenda a função identidade, a qual não é uma boa codificadora, essa compressão do conteúdo de

entrada em uma representação latente pode comprometer a caracterização das informações.

Nesse contexto, os autores propuseram o chamado *Split-Brain Autoencoders*, que refere-se a uma nova arquitetura formada por dois autoencoders f_1 e f_2, de modo que, para um conteúdo de entrada $X \in \mathbb{R}^{C \times H \times W}$, esse é dividido em $X_1 \in \mathbb{R}^{C_1 \times H \times W}$ e $X_2 \in \mathbb{R}^{C_2 \times H \times W}$, e de modo simplificado, tem que a rede é treinada a reconstruir X_1 por meio de $f_2(X_2)$ e X_2 por meio de $f_1(X_1)$. Matematicamente, o processo de otimização pode ser formulado por:

$$f_1^* = \text{argmin}_{f_1} \, \ell(f_1(X_1), X_2) \,, \tag{1}$$

$$f_2^* = \text{argmin}_{f_2} \, \ell(f_2(X_2), X_1) \,, \tag{2}$$

em que $\ell(\cdot, \cdot)$ representa a função de perda e, sob a notação utilizada nesta dissertação, pode ser expressa na seguinte fórmula $\ell(X, Y) = \frac{1}{2} \sum_{h=1}^{H} \sum_{w=1}^{W} \| [X]_{:,h,w} - [Y]_{:,h,w} \|_2^2$.

Desse modo, embora os autores tenham contribuído com uma técnica de aprendizado autossupervisionado relevante para substituir a simples etapa de compressão, tem-se que a sua dependência da etapa de treinamento dificulta sua aplicação em conjuntos de dados com poucas amostras, tais como os conjuntos de imagens de biossensores abordados nesta dissertação. Nesse sentido, faz-se necessário o desenvolvimento de métodos que empregam essa forma de cruzamento de dados, em que prediz-se o conteúdo de uma parte somente a partir da outra, forçando a rede a aprender relações, possivelmente, mais complexas.

Além disso, Sui et al. (2024) argumentam que as ampliações (e.g., rotação, recortes etc.) utilizadas para a construção das tarefas de pretexto no aprendizado autossupervisionado estão limitadas em relação à modalidade (e.g., imagem, texto, vídeo etc.) dos dados em questão. Como exemplo evidente, não é possível aplicar transformações de imagem (e.g., *color jitter*) a dados textuais. Além disso, de modo ainda mais sutil, os autores demonstram que, mesmo quando a técnica de ampliação é adequada à modalidade em questão, podem ocorrer situações em que a própria aplicação impõe restrições que inviabilizem o seu uso. Nesses casos, tais limitações tendem a serem reconhecidas caso haja a participação de um especialista no domínio durante o processo de construção do método. Como exemplo, Sui et al. (2024) cita que imagens histológicas obtidas após processo de coloração possuem baixa variação de cores, de tal forma que ampliações de imagens agressivas podem alterar o conteúdo semântico da imagem, sendo assim inadequadas para o processo de treinamento da rede.

Nesse contexto, os autores propuseram que representações discriminativas dos dados de entrada podem ser obtidas guiando a rede neural a reconstruir múltiplas projeções aleatórias desses dados. Assim, removendo a necessidade de realizar ampliações pré-definidas e as quais são de certa forma atreladas a alguma modalidade de dado (e.g., adição de ruído em imagens). Em tese, os autores constroem funções de projeção aleatórias, g, cujo objetivo é projetar aleatoriamente os dados em outro espaço dimensional. A partir disso, os autores mostram que representações discriminativas $f_\theta(x_i)$ para um dado de entrada x_i serão obtidas caso seja possível treinar um grupo de funções de predição h, com limitada capacidade, que aplicadas à

representação $f_\theta(x_i)$ obterão a saída de $g(x_i)$ (Sui et al., 2024). Especificamente, os autores buscam em resolver o seguinte problema de minimização.

$$\underset{\theta,\,\phi}{\operatorname{argmin}} \sum_{x_i} \sum_k \mathcal{D}\left(g^{(k)}(x_i), h_\phi^{(k)}(f_\theta(x_i))\right) + \lambda_1\Omega_1(\theta) + \lambda_2\Omega_2(\phi)\,,$$

em que $\mathcal{D}(\cdot,\cdot)$ é uma métrica de similaridade, e $\Omega(\cdot)$ representa os termos de regularização.

Logo, note-se que este trabalho está intimamente relacionado às abordagens desenvolvidas nesta dissertação, desde que o conceito do objetivo de minimização está associado às arquiteturas de redes neurais randomizadas (*autoencoders* randomizados), as quais tentam predizer (reconstruir) a partir de projeções aleatórias, com a finalidade de obter boas representações dos dados de entrada, fundamentando ainda mais a investigação de técnicas de redes neurais randomizadas. Além disso, os métodos investigados têm como principal contribuição em relação às técnicas tradicionais autossupevisionadas os seguintes pontos: (i) baixo custo computacional (i.e. rápido e pouco uso de memória); (ii) sem necessidade de treinamento iterativo (i.e., *backpropagation*); e (iii) invariância ao tamanho do conjunto de dados.

3.1.2 Estratégias com Medidas Estatísticas para a Sumarização dos Pesos Aprendidos de Redes Neurais Randomizadas

As abordagens de representação de textura baseadas em redes neurais randomizadas que possuem número de neurônios e atributos de saída superiores a um requerem a etapa de sumarização ou compressão para a obtenção da representação de textura. Essa etapa, obrigatória e essencial, corresponde ao emprego de uma função $f \in \mathbb{F}(\mathbb{R}^{m\times n}, \mathbb{R}^{g(m,n)})$ cuja finalidade é sumarizar a matriz dos pesos aprendidos de dimensão $m \times n$ em um vetor m- ou n-dimensional, em que $g \in \mathbb{F}(\mathbb{N}_*^2, \mathbb{N}_*)$ é a função que definirá a dimensão final desse vetor.

Primeiramente, tal etapa é obrigatória, pois as representações devem ser expressas em termos de vetores, i.e., tensores unidimensionais, enquanto que os pesos aprendidos da camada de saída das redes neurais randomizadas, sob as restrições impostas, são matrizes, i.e., tensores bidimensionais, inadequadas nesse contexto para servir como representação. Em segundo lugar, essa etapa é essencial, pois, a depender da função de sumarização f adotada, podem ser observadas diferenças tanto no desempenho quanto na dimensionalidade da representação, características que apresentam certa correlação e que são importantes para a tarefa final.

Em particular, como visto anteriormente, grande parcela das abordagens baseadas em redes neurais randomizadas, tais como em Sá Junior & Backes (2016), Sá Junior, Backes & Bruno (2018), Sá Junior, Backes & Bruno (2019), Sá Junior, Ribas & Bruno (2019), Ribas et al. (2020), Ribas et al. (2022), Ribas et al. (2024a), Ribas et al. (2024b), Florindo, Backes & Neckel (2024), empregam a rede na modalidade de regressão e, consequentemente, a dimensão da matriz dos pesos aprendidos estará na forma $1 \times n$. Por outro lado, embora as investigações

Scabini et al. (2023), Florindo & Neckel (2023), Scabini et al. (2025)[1] empreguem a rede neural randomizada na forma multialvo, essas o fazem com apenas um único neurônio na camada escondida, de forma que, a dimensão da matriz dos pesos estará na forma $m \times 1$.

Em ambos os casos, embora as investigações citadas não explicitem o uso ou o conceito da função de sumarização, desde que esse é exatamente definido nesta dissertação e fogem das restrições acima, tem-se que a função de sumarização f nesses estudos corresponderia àquela que remove a dimensão unitária. Isso posto, por essas razões, esses métodos estão limitados, de certa maneira, a essa forma da função de sumarização, impossibilitando, intrinsecamente, de permitir a construção de representações mais robustas e discriminativas, por meio da investigação e construção de formas mais apropriadas para a função f.

Por outro lado, estudos como os de Sá Junior & Backes (2023) e Sá Junior & Backes (2024), os quais propuseram uma técnica de representação com rede neural randomizada multialvo com múltiplos neurônios e o seu uso na aplicação prática para detecção de câncer colorretal, respectivamente, exibiram, até onde se tem conhecimento, a primeira utilização de uma medida estatística para a sumarização dos pesos, o qual foi a média das colunas da matriz. Em particular, matematicamente, essa medida pode ser expressa na seguinte fórmula:

$$f(A) = [\mu(\vec{a}_1), \mu(\vec{a}_2), \ldots, \mu(\vec{a}_n)] \in \mathbb{R}^n \,, \tag{3}$$

em que $\vec{a}_k \in \mathbb{R}^m$ denota a k-ésima coluna da matriz $A \in \mathbb{R}^{m \times n}$ e $\mu(\vec{a}_k) \in \mathbb{R}$ denota a média do vetor $\vec{a}_k$, a qual é definida por $\mu(\vec{a}_k) = \frac{1}{m} \sum_{i=1}^{m} [\vec{a}_k]_m$. Além disso, em um estudo preliminar, Sá Junior, Backes & Bruno (2020) empregaram as medidas estatísticas de média (μ) e desvio padrão (σ) para a sumarização de uma matriz. Entretanto, no estudo apresentado, voltado ao aprendizado de representações de formas com redes neurais randomizadas, essas medidas estatísticas não foram empregadas para sumarizar uma matriz dos pesos aprendidos de uma rede neural randomizada multialvo e com múltiplos neurônios. Neste método, os autores construíram a matriz por meio da justaposição vertical de diferentes vetores de pesos, os quais foram aprendidos a partir de distintas configurações dos hiperparâmetros do método, empregando a rede neural randomizada na forma de regressão. Nesse caso, embora o método não se enquadre estritamente sobre o uso da função de sumarização, observa-se que o estudo evidencia que o emprego de medidas estatísticas mostra-se relevante.

Nesse contexto, constata-se uma lacuna na literatura relacionada ao fato de que outras medidas estatísticas, como assimetria (γ), a curtose (κ), outras estatísticas de alta ordem e suas combinações, foram pouco investigadas. Além disso, não é necessário restringir-se ao uso somente de medidas estatísticas, de modo que outras funções podem ser empregadas, como, por exemplo, a norma euclidiana. Por fim, com a finalidade de preencher essa lacuna,

[1] Note que ambas as investigações Scabini et al. (2023) e Scabini et al. (2025) relacionam-se ao aprendizado com redes neurais randomizadas híbridas, e não puras. Entretanto, esses trabalhos são citados, pois a função de sumarização está associada à arquitetura em si, independentemente de sua combinação, ou não, com *backbones*. Dessa forma, esses trabalhos também são abordados e discutidos nesta subseção.

foram propostas novas abordagens de aprendizado de representações de textura utilizando diferentes medidas estatísticas e de norma para a sumarização dos pesos aprendidos.

3.2 APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA COM REDES NEURAIS RANDOMIZADAS HÍBRIDAS

As abordagens de textura com redes neurais randomizadas híbridas referem-se às técnicas que combinam as redes neurais randomizadas com grandes modelos pré-treinados (i.e., *backbones*). Nesse sentido, o objetivo do uso da rede neural randomizada é codificar aleatoriamente as representações profundas de uma ou mais camadas do *backbone* após alguma forma de pré-processamento, com a finalidade de aprimorar a transferibilidade da rede. No entanto, existem diferentes técnicas de codificação de representações profundas propostas na literatura, de modo que, nem todas são baseadas em redes neurais randomizadas. Portanto, a seguir, são apresentadas as técnicas de codificação mais relevantes e relacionadas.

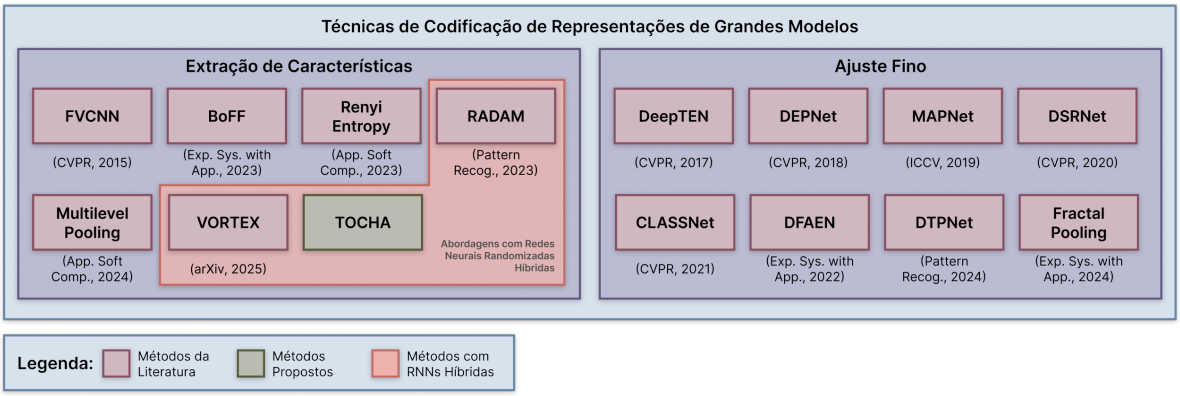
3.2.1 Estratégias de Codificação de Representações Profundas

As técnicas de codificação de representações profundas surgiram com a finalidade de abordar imagens de texturas obtidas em ambientes descontrolados (i.e., imagens capturadas fora de laboratório), em que cuja característica comum é que o conteúdo textural da imagem nem sempre a preenche completamente (Cimpoi; Maji; Vedaldi, 2015). Nesse sentido, os descritores clássicos (i.e., *handcrafted*) mostraram-se limitados quanto à sua capacidade de caracterizar adequadamente o conteúdo textural dessas imagens. Dessa forma, as abordagens presentes na literatura para abordar tais imagens obtidas em ambientes descontrolados apoiaram-se predominantemente no uso de grandes modelos pré-treinados.

Essas abordagens empregaram distintas estratégias de codificação (e.g., técnicas de *pooling*) com a finalidade de aprimorar a transferibilidade da rede para tarefas de análise textural, uma vez que estas diferem consideravelmente da tarefa fonte do pré-treinamento. Em particular, como ilustrado na Figura 10, as técnicas de codificação podem ser divididas em duas categorias: extratoras de características (Cimpoi; Maji; Vedaldi, 2015; Florindo; Laureano, 2023; Florindo, 2023; Scabini et al., 2023; Lyra; Fabris; Florindo, 2024; Scabini et al., 2025) e aprendizado fim a fim (Zhang; Xue; Dana, 2017; Xue; Zhang; Dana, 2018; Zhai et al., 2019; Zhai et al., 2020; Chen et al., 2021; Yang et al., 2022; Chen et al., 2024; Florindo, 2024). A principal diferença entre essas categorias é que, na primeira, tem-se que não há necessidade de treinamento ou ajuste fino da rede, de modo que, emprega-se o *backbone* com os pesos congelados e um classificador tradicional (e.g., LDA, K-NN, SVM etc.) é utilizado à parte. Por outro lado, na segunda categoria, a rede é ajustada (i.e., treinada).

Estratégias de Codificação de Aprendizado Fim a Fim. Zhai et al. (2019) propuseram

Figura 10 – Lista das técnicas citadas de codificação de representações profundas categorizadas com base na técnica de transferência de aprendizado empregada.



Fonte: Elaborada pelo autor.

a técnica DeepTEN, pioneira na codificação de características por meio de aprendizado fim a fim. Diferentemente de FV-CNN (Cimpoi; Maji; Vedaldi, 2015), que emprega *backbones* congelados para a extração de representações e Fisher Vector (FV) para sua codificação, a técnica DeepTEN integra essas etapas em um único *pipeline* de aprendizado, no qual tanto as características quanto o codificador são aprendidos dinamicamente. Em seguida, Xue, Zhang & Dana (2018) argumentaram a necessidade de técnicas que preservassem certa informação espacial do conteúdo textural, uma vez que existem imagens do mundo real que apresentam tal característica (e.g., imagens de terreno). Desse modo, os autores propuseram uma rede neural, denominada por DEPNet, a qual é composta por dois ramos de aprendizado responsáveis por capturar o conteúdo textural e sua informação espacial, respectivamente.

Em seguida, Zhai et al. (2019) exibiram que não somente uma mesma imagem pode ser descrita por distintos atributos texturais (e.g., marmorizado, com nervuras etc.), mas imagens de diferentes classes podem compartilhar um subconjunto de atributos texturais. Nesse contexto, os autores construíram uma rede neural, denominada por MAPNet, que incorporassem as características visuais da textura com a finalidade de melhorar seu reconhecimento. Sob a mesma perspectiva, Zhai et al. (2020) demonstraram que uma imagem pode conter diferentes primitivas texturais a depender de sua localização; entretanto, tais primitivas apresentam uma dependência estrutural inerente, i.e., uma primitiva pode emergir a partir da outra. Desse modo, os autores propuseram a DSRNet com a finalidade de abordar tal questão.

Chen et al. (2021) argumentaram que, embora o *Global Average Pooling* (GAP) seja um mecanismo para a agregação das características extraídas por CNNs, robusto a transformações espaciais, era necessária a formulação de um novo módulo que preservasse tal robustez, mas que ampliasse a capacidade de reconhecimento. Desse modo, os autores propuseram um novo módulo, denominado CLASS, o qual, diferentemente de outras técnicas, emprega não apenas características extraídas de diferentes camadas da rede, mas as interpreta

como uma hipersuperfície, utilizando como descritor o histograma dos valores calculados a partir de uma medida da geometria fractal. Na sequência, Yang et al. (2022) propuseram uma nova rede com o objetivo de mitigar a redundância de características extraídas e agregar informações de primeira e segunda ordem de textura, aprimorando, assim, sua representação.

Posteriormente, Chen et al. (2024) discutem que muitas técnicas de codificação extraem características somente da última camada do *backbone*, descartando informações capturadas de camadas anteriores, as quais podem ser mais úteis, uma vez que tais camadas aprendem características de baixo nível (e.g., texturas e formas). Nesse sentido, os autores propuseram a rede denominada DTPNet, a qual agrega características extraídas de múltiplas camadas da rede e as codificam por meio de padrões binários locais tridimensionais, aprimorando a captura de detalhes texturais mais finos. Em seguida, Florindo (2024) exibe que, embora a grande parcela das técnicas de codificação de representações profundas utilize o GAP para a agregação espacial de características, esse mecanismo não apenas destrói os relacionamentos locais de textura, como também apresenta capacidade limitada para capturar informações mais complexas. Desse modo, o autor propõe uma técnica de *pooling* baseada na dimensão fractal aplicada sobre os mapas de ativação residuais, os quais mostraram-se mais discriminativos em comparação à simples média das ativações nas dimensões espaciais.

Isso posto, foram duas as principais contribuições das técnicas apresentadas as quais se mostram mais diretamente relacionadas a esta dissertação. Em primeiro lugar, destaca-se o entendimento de que são necessárias técnicas capazes de integrar tanto a componente resultante da agregação das ativações espaciais (i.e., *orderless component*) quanto aquela que preserva a informação espacial da textura. Em segundo lugar, é ressaltado a necessidade de técnicas que extraiam informações de múltiplas camadas e que não restrinjam a extração apenas à última camada do *backbone*, uma vez que esta é predominantemente composta por características de alto nível (i.e., mais próximas da tarefa fonte do pré-treinamento).

Por fim, sob o contexto dos problemas abordados nesta dissertação, que se referem à diagnóstico de doenças em conjuntos de dados de imagens de biossensores, os quais são caracterizados por poucas amostras e pelo fato do conteúdo microtextural ser essencial para sua caracterização. Identifica-se que essas técnicas baseadas em aprendizado fim a fim apresentam limitações no que diz respeito ao ajuste fino, uma vez que existem poucas amostras para o treinamento dos modelos convolucionais, por exemplo, os quais apresentam elevada capacidade e grande quantidade de parâmetros, demandando, portanto, maiores conjuntos de dados. Nesse contexto, por tais limitações, a investigação de métodos de representação com *backbones* nesta desta dissertação focou, portanto, em técnicas extratoras de características.

Estratégias de Codificação de Extratores de Características. A técnica FV-CNN, proposta por Cimpoi, Maji & Vedaldi (2015), foi uma das primeiras abordagens de destaque para a representação de texturas, atuando como extratora de características. Essa técnica extrai características profundas a partir da última camada do *backbone* convolucional congelado,

constrói um vocabulário a partir dessas e utiliza a técnica de *Fisher Vector* (FV), atuando como um mecanismo de *pooling*, para a construção da representação final.

Posteriormente, Florindo & Laureano (2023) abordam a característica *tudo ou nada* da técnica de quantização vetorial, empregada em abordagens como a FV-CNN, durante o mecanismo de codificação. Como exemplo da abordagem de quantização tradicional. Seja $\mathcal{C} = \{\vec{c}_1, \vec{c}_2, \ldots, \vec{c}_K\} \subset \mathbb{R}^D$ o vocabulário aprendido, em que $K \in \mathbb{N}_*$ denota o número de *clusters* e $D \in \mathbb{N}_*$ a dimensionalidade dos descritores locais. Nesse caso, para um dado descritor local externo $\vec{x}_i \in \mathbb{R}^D$, sua contribuição para o histograma ocorre com uma única unidade apenas para o *cluster* ao qual $\vec{x}_i$ está mais próximo, evidenciando, assim, tal característica. Desse modo, os autores propuseram utilizar a lógica *fuzzy* com a finalidade de quantificar, no intervalo unitário, o pertencimento de um descritor local em relação aos diferentes *clusters*. Logo, por exemplo, mesmo que $\vec{x}_i$ esteja mais próximo de $\vec{c}_j$, $j \in [1, K]_{\mathbb{N}}$, tem-se que $\vec{x}_i$ irá contribuir com algum valor $\gamma_k \in [0, 1]$ nas outras componentes do histograma.

Em seguida, Scabini et al. (2023) propuseram a primeira abordagem de representação de textura baseadas em redes neurais randomizadas híbridas, denominada por RADAM. Em particular, os autores apresentaram uma abordagem que, primeiramente, agrega as saídas dos diferentes mapas de ativação profundos, assim capturando características visuais de baixo e alto nível, e, em seguida, tal agregação é redimensionada em uma matriz, seguida da adição de codificação posicional (em inglês, *positional encoding*), a qual é utilizada para alimentar um conjunto de *autoencoders* randomizados (i.e., uma sopa de *autoencoders*) cujo objetivo é realizar a codificação aleatória dos mapas de ativação agregados sob diferentes projeções aleatórias. Em sequência, os pesos aprendidos dos decodificadores são agregados e utilizados como representação de textura. Ainda mais, note-se que, mesmo havendo o treinamento *autoencoder* randomizado, a técnica de aprendizado como um todo atua como um extrator de características, uma vez que tal *autoencoder* é treinado por imagem (i.e., por instância).

Ainda, Florindo (2023) desenvolveu um mecanismo baseado na entropia de Renyi com a finalidade de substituir a operação GAP. Em particular, é comum que representações sejam obtidas a partir da aplicação da operação GAP nos mapas de ativação da última camada convolucional. Entretanto, como visto anteriormente, a simples média das ativações, embora relevante, pode descartar informações importantes para a construção da representação. Nesse contexto, o autor, baseado nos conceitos da teoria da informação, propôs a utilização do cálculo da entropia de Renyi sobre cada mapa de ativação da última camada do *backbone*, substituindo, assim, a simples média e agregando informações adicionais, como características associadas à complexidade da textura, desse modo, melhorando sua caracterização.

Além disso, Lyra, Fabris & Florindo (2024) argumentaram que, embora existam métodos que empreguem características de camadas anteriores, esses pertencem à categoria de aprendizado fim a fim, a qual apresenta limitada aplicabilidade em determinados contextos reais, como aqueles caracterizados por conjuntos de dados ínfimos. Nesse contexto, os autores

preenchem tal lacuna ao propor uma nova técnica que combina as características extraídas em múltiplos blocos do *backbone*, com a finalidade de agregar não apenas características de baixo nível (i.e., mais genéricas), mas também aquelas de alto nível (i.e., mais específicas). Em seguida, as características extraídas e adequadamente processadas são submetidas a um processo de codificação do tipo *orderless* por meio da técnica de *Fisher Vector*.

Na sequência, Scabini et al. (2025) propuseram a segunda técnica de aprendizado de representações com redes neurais randomizadas híbridas, porém, a primeira que emprega a arquitetura de *Vision Transformers* como *backbone*, denominada por VORTEX. Os autores argumentaram que embora ViTs tenham sido muito utilizados para tarefas de visão computacional, pouco foi explorado seu uso para o reconhecimento textural. Nesse sentido, guiado pelas boas representações obtidas pela primeira abordagem, os autores propuseram um novo módulo, o qual é acoplado ao *backbone* ViT e cuja responsabilidade é codificar aleatoriamente suas representações. Para isso, tem-se que, primeiramente, uma vez que o *backbone* é alimentado com a imagem de entrada, o módulo proposto realiza agregação dos *token embeddings* provenientes das diferentes camadas do *backbone*. Em seguida, esse bloco de *tokens* agregados tem suas colunas normalizadas, e, é utilizado como entrada para um conjunto de *autoencoders* randomizados, os quais codificam aleatoriamente os *tokens* extraídos, e, em seguida, os pesos dos decodificadores aprendidos são agregados e empregados como representação.

Desse modo, assim como anteriormente, também foram duas as mais relevantes contribuições dentre os trabalhos relacionados. Em primeiro lugar, cita-se não somente os novos mecanismos de *pooling* cujo objetivo foi enriquecer o conteúdo semântico da representação, tal como apresentado em Florindo (2023), mas também os novos mecanismos de codificação, principalmente, aqueles que empregam redes neurais randomizadas, tais como Scabini et al. (2023) e Scabini et al. (2025), os quais são os mais relacionados com esta dissertação. Em segundo lugar, de modo mais relacionado às redes neurais randomizadas, tem-se como contribuição a forma como os mapas de ativação das múltiplas camadas são agregados, redimensionados e pós-processados, com o objetivo de fornecê-lo para a rede neural randomizada o codificar aleatoriamente. Essa é uma contribuição importante por dois motivos: (i) as saídas das diferentes camadas do *backbone* são tensores tridimensionais, enquanto que a entrada da rede neural randomizada é um tensor bidimensional; e (ii) a depender da forma como a agregação é realizada e pós-processada, isso pode impactar o desempenho das representações.

Por fim, dentre esses trabalhos, foram identificadas duas lacunas, entre elas: (i) falta de investigações sobre a possível modelagem dos mapas de ativação (e.g., modelagem em grafos), com a finalidade de extrair características adicionais sobre possíveis relacionamentos entre as ativações; e (ii) não foram investigadas seleções de subconjuntos de camadas do *backbone*, desde que, em geral, as investigações ou realizam a agregação de todas as camadas ou selecionam apenas a última camada para a extração da representação. Nesse contexto, a abordagem randomizada híbrida proposta nesta dissertação visou preencher tais lacunas.

3.3 CONSIDERAÇÕES FINAIS

Neste capítulo, foram apresentadas as abordagens de representações de textura com redes neurais randomizadas puras e híbridas, as quais são mais relevantes a esta dissertação. Em particular, apresentou-se um panorama geral da literatura sobre essas técnicas exibindo suas contribuições e as respectivas lacunas a serem preenchidas pelos métodos investigados. Especificamente, para as abordagens puras, exibiu-se que grande parcela da literatura focou em técnicas que empregaram a rede neural randomizada pura em sua modalidade de regressão, assim, deixando lacunas a serem exploradas na modalidade multialvo e com *autoencoders*. Além disso, mesmo com relação à única técnica de *autoencoder*, tem-se que essa deixou lacunas para investigação de novas formas de construção das matrizes de entrada e saída do *autoencoder*, bem como a investigação do uso de novas medidas estatísticas ou de norma.

Ainda mais, com relação às estratégias de aprendizado autossupervisionado e estratégias de codificação de representações profundas por meio de aprendizado fim a fim, identificou-se que a necessidade de treinamento iterativo (i.e., *backpropagation*) por parte dessas abordagens limitam sua aplicabilidade a contextos com conjuntos de dados com poucas amostras, tais como os conjuntos de imagens de biossensores abordados nesta dissertação. Por fim, em relação às técnicas de codificação extratoras de características, especificamente, com base naquelas que empregam redes neurais randomizadas, as quais são denominadas por híbridas, tem-se que essas deixaram como lacunas tanto a falta de modelagem das ativações do *backbone* quanto a imaleabilidade com relação à agregação total de todas as saídas dos mapas de ativação, as quais podem carregar características de alto nível (i.e., mais específicas da tarefa de pré-treinamento) não interessantes para as tarefas de reconhecimento microtextural.

4 APRENDIZADO DE REPRESENTAÇÕES COM REDES NEURAIS RANDOMIZADAS PURAS

Neste capítulo, serão apresentadas as técnicas de aprendizado de representações de texturas com redes neurais randomizadas puras. Especificamente, tais técnicas são adjetivadas como puras, pois realizam o aprendizado de representações de textura sem o emprego de modelos de aprendizado profundo pré-treinados, i.e., sem a utilização de *backbones* dentro do processo de aprendizado. Dessa forma, a qualidade da representação obtida é dependente de três aspectos: dos conteúdos de entrada e saída fornecidos à rede neural randomizada; da capacidade arquitetural dessa de aprender, por meio do cruzamento de dados, os padrões texturais intrínsecos presentes em tais conteúdos; e a forma de sumarização dos pesos aprendidos da rede neural randomizada. Portanto, durante este trabalho foram investigadas e propostas técnicas de aprendizado de representações de textura com redes neurais randomizadas puras, as quais serão apresentadas a seguir e que atacam tais problemáticas, com a finalidade de construir representações de textura mais robustas e discriminativas.

Nesse contexto, este capítulo está dividido em duas seções. A Seção 4.1 apresenta as técnicas de aprendizado de representações de textura em nível de cinza. Embora exista uma imensa quantidade de imagens coloridas e hiperespectrais atualmente, tem-se que as imagens de textura em nível de cinza são de extrema importância, desde que diversas áreas, tais como: medicina, biologia, ciência dos materiais, entre outras, são compostas por inúmeras aplicações, sensores, técnicas de microscopia, responsáveis por processar ou gerar imagens em nível de cinza. Desse modo, seguindo os mesmos estágios iniciais do campo de estudo da análise textural que fora realizado sobre imagens em nível de cinza; a investigação, o desenvolvimento e o avanço de tais técnicas em nível de cinza continuam sendo essenciais em aplicações práticas.

Por outro lado, a Seção 4.2 apresenta as abordagens de aprendizado de representações de texturas coloridas. Em particular, embora as imagens coloridas possam ser representadas em diversos espaços de cor, o principal espaço selecionado para o desenvolvimento e análise das abordagens propostas foi o espaço de cores RGB. A justificativa para tal escolha é que grande parte das imagens coloridas existentes são originalmente representadas no espaço de cor RGB, permitindo, portanto, a fácil aplicação das técnicas de representações propostas, e, além disso, os conjuntos de dados de *benchmark* para a investigação e análise das metodologias, bem como os conjuntos de dados das aplicações são representados em tal espaço de cor. No entanto, mesmo que as técnicas propostas tenham sido construídas sobre o espaço de cor RGB, essas podem ser aplicadas em qualquer espaço de cor munido de três canais. Além disso, como será visto posteriormente, uma das técnicas coloridas propostas pode ser aplicada em imagens com uma quantidade arbitrária de canais sendo, portanto, extremamente adaptável.

Por fim, dentro de cada seção, as abordagens de aprendizado de representações propostas são apresentadas em distintas subseções detalhadamente. Essas apresentações são constituídas por um parágrafo de introdução, a qual apresenta de forma concisa a metodologia da técnica proposta, como essa está preenchendo uma lacuna na literatura, e a ideia pela qual o funcionamento da abordagem de representação está fundamentada. Em seguida, a metodologia da técnica de representação de textura é apresentada no tópico denominado por **Construção da Representação** que apresentará como a representação de textura é obtida a partir de uma dada imagem. Na sequência, os tópicos intitulados **Análise** ou **Avaliação Experimental** dedicam-se, conforme a própria denominação sugere, à investigação e à discussão experimental da metodologia proposta. Logo após, com a finalidade de terminar a apresentação da metodologia, o tópico de **Contribuições e Limitações** exibe como a abordagem apresentada contribui nas áreas de análise textural e reconhecimento de padrões e quais são as limitações existentes que podem ser abordadas em trabalhos futuros.

4.1 APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA EM NÍVEL DE CINZA

Esta seção dedica-se à apresentação das metodologias de aprendizado de representações de textura em nível de cinza com redes neurais randomizadas puras. Dessa forma, a Subseção 4.1.1 exibe a metodologia proposta em Fares et al. (2024) que emprega múltiplos *autoencoders* randomizados inicializados com diferentes sementes aleatórias com o objetivo de construir uma representação mais discriminativa e generalizável baseando-se no fato de que cada *autoencoder* aprenderá uma perspectiva diferente, porém consensual e complementar, dos padrões texturais brutos presente na imagem de entrada. Em seguida, a Subseção 4.1.2 apresenta a abordagem proposta em Fares, Guerra & Ribas (2025), que utiliza uma modelagem em grafos livre de parâmetros para os padrões texturais locais seguida da codificação das informações topológicas extraídas a partir de tais grafos por uma rede neural randomizada cujos pesos aprendidos são posteriormente sumarizados por um conjunto de medidas estatísticas e de norma para a construção da representação de textura final. Por fim, a Subseção 4.1.3 dedica-se à comparação das técnicas propostas com as abordagens existentes na literatura.

4.1.1 Aprendizado de Representações de Textura por Ensemble de Autoencoders Randomizados

Esta subseção descreve o método proposto de aprendizado de representações de textura por *ensemble* de *autoencoders* randomizados publicado em Fares et al. (2024). Este artigo tem como inovação a proposta de uso de múltiplos *autoencoders* randomizados que projetam distintamente os padrões texturais da imagem em um outro espaço dimensional e aprendem a reconstruir os padrões texturais de entrada a partir dessas distintas projeções. Dessa forma, como as projeções são distintas, então os padrões texturais codificados também

serão distintos, portanto, formando diferentes perspectivas desses padrões (i.e., produzindo informações complementares). No entanto, também deve-se notar que tais padrões codificados são consensuais (i.e., possuem informações relacionadas, compartilhadas), desde que, partiram do mesmo padrão textural de entrada. Nesse contexto, ao forçar os *autoencoders* randomizados a reconstruírem os padrões texturais originais, esses consequentemente, aprendem diferentes características da imagem de textura de entrada, as quais estão codificadas nos pesos do decodificador. Logo, os pesos provenientes de cada decodificador também formam um conjunto de informações consensuais e complementares, que serão utilizados para construir uma representação de textura em nível de cinza mais descritiva e robusta.

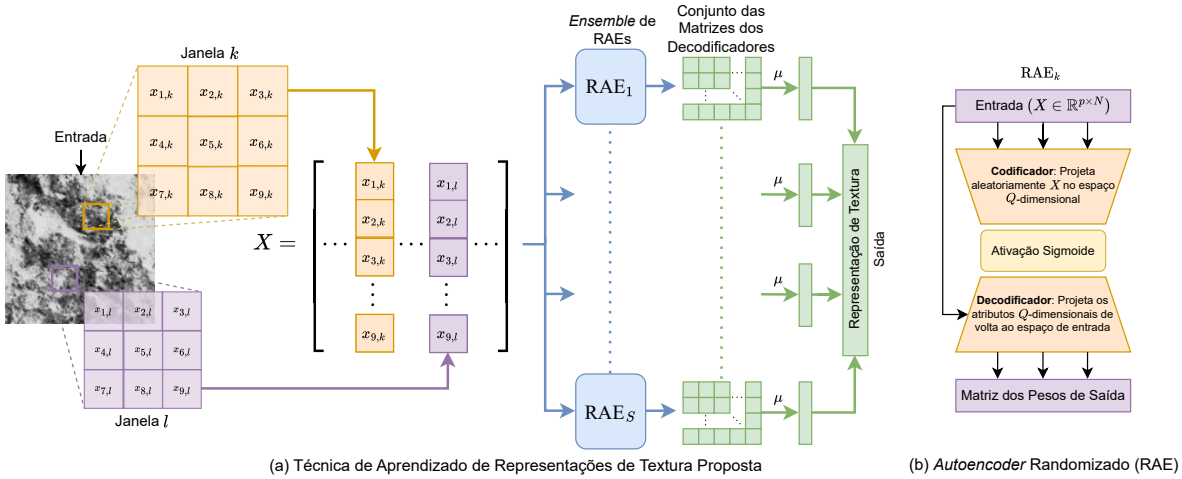
Construção da Representação. A seguir, são listados os passos necessários para a construção da representação de uma imagem de textura em nível de cinza. Dessa forma, seja $I \in \mathbb{R}^{H \times W}$ uma imagem em nível de cinza, em que H e W são a altura e largura, respectivamente. A partir dela, será construída a matriz de atributos de entrada X, que alimentará os *autoencoders* randomizados. Para isso, janelas 3×3 são centralizadas em todos os pixels da imagem e, em seguida, são esticadas (formando vetores 9-dimensionais) e justapostas horizontalmente compondo, assim, a matriz de atributos de entrada X, como ilustrado na Figura 11. Essa etapa é essencial, pois permite construir tal matriz contendo as informações locais de textura da imagem de entrada, as quais serão úteis para o treinamento dos *autoencoders* randomizados (Fares et al., 2024). Para fornecer um exemplo da dimensionalidade, dada uma imagem de textura de dimensão 128×128, então haverá 16.384 pixels a serem percorridos para a centralização das janelas de tamanho 3×3 e, portanto, tem-se que $X \in \mathbb{R}^{9 \times 16384}$.

Dessa forma, uma vez que X está construído, deve-se gerar o seguinte conjunto $\mathcal{U} = \{U_1, U_2, \ldots, U_S\} \subset \mathbb{R}^{Q \times (p+1)}$ que comporta as matrizes inicializadas aleatoriamente de cada *autoencoder* randomizado, em que $p = 9$ é o tamanho da janela esticada, e S é o tamanho do *ensemble* (i.e., a quantidade de *autoencoders* randomizados). Nesse contexto, tem-se que a matriz U_k inicializada aleatoriamente será empregada pelo k-ésimo *autoencoder* para projetar os vetores de atributos de entrada da matriz X no espaço Q-dimensional. Para construir tal conjunto, primeiramente, constrói-se um vetor $V \in \mathbb{R}^L$, em que $L = S \cdot Q \cdot (p+1)$. Esse vetor será populado empregando o gerador congruente linear (em inglês, *Linear Congruential Generator* (LCG)) seguindo o mesmo procedimento aplicado em Sá Junior & Backes (2016), de modo que os elementos de V seguem a seguinte fórmula de recorrência:

$$V(n+1) = (a \cdot V(n) + b) \bmod c\,, \tag{1}$$

em que as seguintes inicializações de parâmetros $V(0) = L+1,\ a = L+2,\ b = L+3,\ c = L^2$, também seguem como aplicado em Sá Junior & Backes (2016). Logo, uma vez que V está populado, esse vetor é padronizado e redimensionado nas dimensões $(S, Q, p+1)$, de tal forma que, formará um tensor com três dimensões representando as S matrizes de dimensão $Q \times (p+1)$ do conjunto $\mathcal{U}$. Por fim, deve-se notar que o $+1$ no número de atributos de entrada p, refere-se ao peso do *bias* adicionado durante a projeção aleatória.

Figura 11 – Ilustração da metodologia proposta em Fares et al. (2024) para aprendizado de representações de textura a partir de múltiplas projeções aleatórias utilizando um *ensemble* de *autoencoders* randomizados. (a) Ilustração do processo para obtenção da representação. (b) *Autoencoder* Randomizado (RAE).



Fonte: Adaptada de Fares et al. (2024).

Portanto, com as matrizes aleatórias geradas, serão calculados os pesos do decodificador, $M_k \in \mathbb{R}^{p \times (Q+1)}$, de cada *autoencoder* randomizado no *ensemble*. Para isso, tem-se que a matriz de atributos de entrada X será aplicada como entrada para cada *autoencoder*, como ilustrado na Figura 11, combinado com sua própria matriz aleatória. Nesse contexto, os pesos do decodificador do k-ésimo *autoencoder* é calculado da seguinte forma:

$$M_k = \text{RAE}(X; U_k, \lambda)\,, \tag{2}$$

em que $\lambda = 10^3$ é o valor da regularização de Tikhonov empregado neste método. Logo, com os pesos de cada decodificador calculados, são construídas as representações parciais, $\vec{\Omega}_k(Q)$, obtidas para cada *autoencoder*. Para isso, define-se a representação parcial da seguinte forma:

$$\vec{\Omega}_k(Q) = [\mu(\vec{m}_{1,k}), \mu(\vec{m}_{2,k}), \ldots, \mu(\vec{m}_{Q+1,k})] \in \mathbb{R}^{Q+1}\,, \tag{3}$$

em que $\vec{m}_{j,k} \in \mathbb{R}^p$ representa a j-ésima coluna da matriz dos pesos de saída M_k, e $\mu(\vec{m}_{j,k}) \in \mathbb{R}$ representa a média dos valores desta coluna (Fares et al., 2024). Dessa forma, como cada representação parcial $\vec{\Omega}_k(Q)$, carrega em si distintas informações aprendidas devido às distintas projeções, então a representação de textura final é formulada da seguinte forma:

$$\vec{\Phi}(Q, S) = \left[\vec{\Omega}_1(Q), \vec{\Omega}_2(Q), \ldots, \vec{\Omega}_S(Q)\right] \in \mathbb{R}^{S \cdot (Q+1)}\,, \tag{4}$$

em que essa representação agrega todos os pesos aprendidos sumarizados pela média a partir das distintas projeções promovendo, portanto, um descritor mais robusto.

Análise de Parâmetros. A representação de textura proposta possui dois parâmetros, sendo esses: Q e S, representando o número de neurônios na camada escondida de cada *autoencoder*

randomizado, e o tamanho do *ensemble*, respectivamente. Isso posto, é analisada a qualidade das representações à medida que tais parâmetros são alterados. Para isso, tem-se que Q será analisado dentro do conjunto de valores $\mathcal{Q} = \{9, 19, 29, 39, 49, 59\}$, e S será analisado dentro do conjunto de valores $\mathcal{S} = \{1, 9, 18, 27\}$. Além disso, para que não haja quantidade de atributos exorbitantes nas representações analisadas, serão, portanto, investigados somente os pares ordenados (Q, S) cujo número de atributos da representação correspondente não exceda 1.000, ou seja, apenas aqueles pertencentes ao seguinte espaço de parâmetros:

$$\mathcal{E}_{\text{REE}} = \{(Q, S) \in \mathcal{Q} \times \mathcal{S} \mid S \cdot (Q + 1) \leq 1000\}\,. \tag{5}$$

Dessa forma, uma vez definido o espaço de parâmetros a ser investigado, os desempenhos das representações correspondentes são mensurados utilizando como métrica a acurácia, a qual é calculada utilizando o classificador *Linear Discriminant Analysis* (LDA) e a estratégia de validação cruzada *Leave-One-Out* (LOO). As bases de dados usadas na análise de parâmetros foram: Outex (Ojala et al., 2002), USPtex (Backes; Casanova; Bruno, 2012), Brodatz (Brodatz, 1966) e MBT (Abdelmounaime; Dong-Chen, 2013), de modo que suas características e desafios estão apresentadas em detalhe no Apêndice A.

A Tabela 1 apresenta as acurácias de classificação da representação $\vec{\Phi}(Q, S)$ para todas as configurações dentro do espaço de parâmetros definido previamente. A partir dessas, é possível identificar uma melhora na acurácia média das representações à medida que Q e S são ampliados, pois há um aumento na quantidade de atributos aprendidos compondo tal configuração de representação. No entanto, deve-se notar que tal aumento do número de atributos não corresponderá sempre a aumentos da acurácia média. Como exemplo, tem-se que a configuração $\vec{\Phi}(49, 9)$ obtém uma acurácia média de 95,73% com 450 atributos, enquanto que a configuração $\vec{\Phi}(29, 27)$ obtém uma acurácia média de 95,64% (Fares et al., 2024). Dessa forma, a primeira apresentou-se como uma representação mais compacta (redução de 360 atributos) e maior acurácia média em relação à representação comparada.

Em seguida, ao analisar as representações $\vec{\Phi}(19, 27)$ (95,79%), $\vec{\Phi}(49, 9)$ (95,73%) e $\vec{\Phi}(59, 9)$ (95,72%), em que os valores entre parênteses referem-se às acurácias médias obtidas pela configuração, tem-se que essas foram as que obtiveram os três melhores desempenhos médios, respectivamente. Em particular, é possível identificar uma característica interessante das configurações apresentadas. A configuração com melhor desempenho é aquela que possui um projetor aleatório com baixa dimensão e um grande tamanho de *ensemble*. Isso indica, na média e para as bases analisadas, que o método proposto foi capaz de agregar as pequenas porções informações complementares aprendidas dos muitos *autoencoders* para construir um descritor mais discriminativo. Entretanto, no extremo oposto, tem-se que as representações $\vec{\Phi}(49, 9)$ e $\vec{\Phi}(59, 9)$ possuem projetores aleatórios de dimensões elevadas ao passo que possuem um pequeno *ensemble*. Nesse caso, isso sugere que boas representações de textura foram obtidas a partir da agregação de maiores porções de informações complementares provenientes de poucos *autoencoders*.

Tabela 1 – Acurácias de classificação da representação de textura $\vec{\Phi}(Q, S)$. Os resultados **destacados** são aqueles que apresentam as maiores acurácias médias. ↑ indica que maiores resultados são preferíveis.

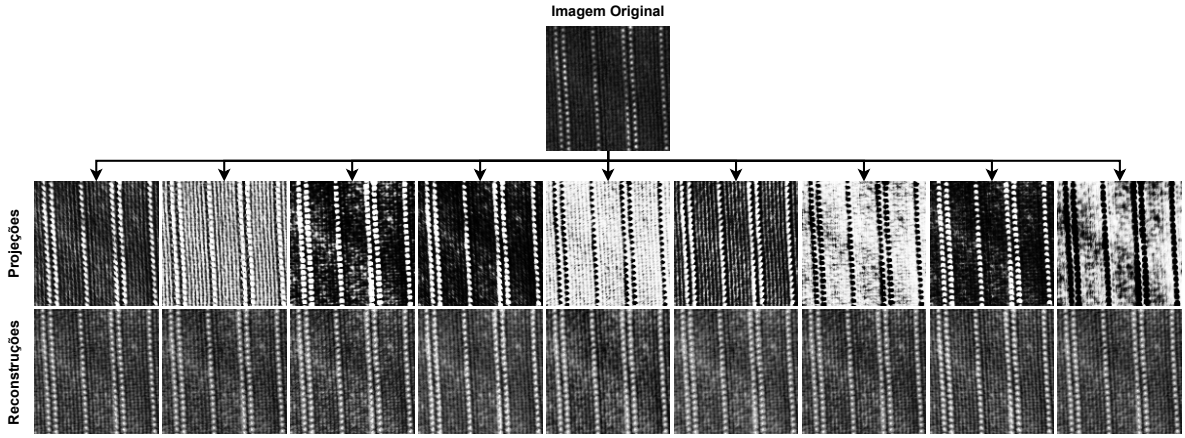
(Q, S)	# de Atributos	Acurácia (↑)				
		Outex	USPtex	Brodatz	MBT	Média
(09, 1)	10	87,28	81,46	84,23	66,52	79,87
(19, 1)	20	89,56	86,61	87,78	80,15	86,03
(29, 1)	30	90,59	90,49	89,53	84,90	88,88
(39, 1)	40	90,81	92,67	92,17	87,70	90,84
(49, 1)	50	90,88	93,63	92,79	87,66	91,24
(59, 1)	60	91,84	94,28	93,24	88,35	91,93
(09, 9)	90	92,13	95,11	95,21	91,11	93,39
(19, 9)	180	93,01	96,99	96,45	93,30	94,94
(29, 9)	270	93,46	97,47	96,34	92,94	95,05
(39, 9)	360	93,46	97,77	97,13	92,94	95,33
(49, 9)	**450**	**93,82**	**98,08**	**97,07**	**93,95**	**95,73**
(59, 9)	**540**	**93,97**	**97,95**	**97,47**	**93,47**	**95,72**
(09, 18)	180	92,72	95,03	95,10	90,79	93,41
(19, 18)	360	93,60	97,60	97,02	93,63	95,46
(29, 18)	540	92,94	98,17	97,30	93,59	95,50
(39, 18)	720	92,79	97,95	97,41	93,55	95,43
(49, 18)	900	92,06	97,82	97,69	93,30	95,22
(09, 27)	270	92,94	97,60	96,79	92,61	94,99
(19, 27)	**540**	**93,68**	**98,43**	**97,24**	**93,79**	**95,79**
(29, 27)	810	93,24	97,99	97,52	93,79	95,64

Fonte: Adaptada de Fares et al. (2024).

Por fim, com o propósito de realizar uma comparação justa do método proposto com outros métodos da literatura, serão adotadas as configurações que obtiveram as três maiores acurácias médias entre todos os conjuntos de dados analisados. Portanto, como apresentado, tais representações propostas neste método a fim de comparação são: REE$_a$ ($\vec{\Phi}(19, 27)$), REE$_b$ ($\vec{\Phi}(49, 9)$) e REE$_c$ ($\vec{\Phi}(59, 9)$), as quais obtiveram acurácia média de 95,79%, 95,73% e 95,72%, respectivamente.

Análise Qualitativa. A fim de analisar qualitativamente o método proposto, serão analisadas as visualizações das múltiplas projeções aleatórias realizadas pelos *autoencoders* no *ensemble*, e suas respectivas visualizações das reconstruções da informação original. Nesse contexto, a Figura 12 apresenta uma visualização de tais projeções e reconstruções para a representação de textura $\vec{\Phi}(49, 9)$. Nessa figura, nota-se que a partir da imagem original é realizado nove projeções aleatórias distintas dos padrões locais de textura cujas visualizações são apresentadas na primeira linha. A partir dessa, é possível perceber como as projeções aleatórias afetaram os padrões de textura variadamente realçando, assim, distintas características da imagem original. Essa variação nos realces, devido aos *autoencoders* serem inicializados com sementes aleatórias

Figura 12 – Ilustração das múltiplas projeções aleatórias da representação de textura $\vec{\Phi}(49, 9)$. A primeira linha corresponde as visualizações das projeções aleatórias da imagem de entrada. A segunda linha corresponde as visualizações das reconstruções.



Fonte: Adaptada de Fares et al. (2024).

distintas, permitem que esses vejam de forma consensual e complementar a informação do conteúdo textural bruto contida na imagem original.

Em seguida, ao analisar a segunda linha, a qual contém as visualizações das reconstruções dos padrões de textura de entrada, é possível perceber que tais reconstruções possuem grande semelhança entre si e com a informação original da imagem de entrada. Essa observação reforça o fato de que o conteúdo das representações parciais aprendidas $\vec{\Omega}_k(Q)$ carregam informações complementares sobre os padrões de textura da imagem original, pois mesmo a partir de distintas projeções aleatórias, cada *autoencoder* foi capaz de reconstruir o conteúdo original com certa fidelidade, como visto por inspeção visual. Nesse contexto, a representação final $\vec{\Phi}(Q, S)$, constituída de todas as representações parciais, engloba tais informações complementares obtendo-se, portanto, um descritor de textura mais robusto.

Contribuições e Limitações. As principais contribuições da metodologia proposta são: (i) estar baseada em conceitos fundamentais do aprendizado autossupervisionado, tal como guiar o aprendizado de representações de textura com base nos próprios dados de entrada (Schiappa; Rawat; Shah, 2023; Gui et al., 2024), desde que a arquitetura específica de *autoencoder* randomizado está sendo utilizada para tal. Dessa forma, os pesos do decodificador são forçados a aprender os relacionamentos inerentes pertencentes a esses padrões de textura, tornando-se, portanto, útil como representação. (ii) Emprego de conceitos do aprendizado multivisão utilizando, nesse caso, as múltiplas projeções aleatórias como fonte de informações complementares (i.e., únicas) e consensuais (i.e., compartilhadas), fazendo, para isso, o emprego de um *ensemble* de *autoencoders* randomizados, permitindo, assim, que cada *autoencoder* possa aprender informações distintas sobre os padrões de textura de entrada.

No entanto, a metodologia proposta está munida com algumas limitações. A principal

limitação dessa é que o número de atributos da representação de textura final $\vec{\Phi}(Q,S) \in \mathbb{R}^{S\cdot(Q+1)}$ está diretamente relacionada com o tamanho do *ensemble* (S). Nesse sentido, tornaria-se inviável o estudo do desempenho da representação quando a dimensionalidade da camada escondida e o tamanho do *ensemble* forem relativamente elevados, desde que geraria $\vec{\Phi}(Q,S)$ com alto número de atributos. Além disso, uma outra limitação, é a possível falta de cruzamento de informações. Nesse caso, como trata-se de um *autoencoder*, os mesmos dados de entrada são usados como saída. Todavia, o emprego de distintas informações sobre os padrões de textura entre os dados de entrada e saída pode permitir que a rede aprenda informações ainda mais discriminativas, pois é forçada a realizar o cruzamento de tais dados.

4.1.2 Aprendizado de Representações de Grafos Locais para Análise de Texturas

Esta subseção apresenta o método proposto de aprendizado de representações de textura baseado em modelagem de textura em grafos locais e codificação com redes neurais randomizadas publicado em Fares, Guerra & Ribas (2025). Esse método possui como principal contribuição a utilização de uma modelagem de textura em grafos livre de parâmetros, distinguindo-se das abordagens que necessitam de (muitos) parâmetros, como em Scabini, Ribas & Bruno (2020), Ribas et al. (2020), Ribas et al. (2024a), Ribas et al. (2024b). Além disso, nessas abordagens, a modelagem de textura em grafos empregada constrói um único ou vários grandes grafos, os quais são construídos tendo cada pixel i da imagem original como um vértice v_i, culminando assim em grafos com $H \cdot W$ vértices para uma imagem $I \in \mathbb{R}^{H\times W}$. Por outro lado, a modelagem em grafo da metodologia proposta diferentemente de construir unidades até dezenas de enormes grafos, foca em construir milhares de minúsculos grafos permitindo, portanto, uma modelagem mais fina dos padrões de textura da imagem de entrada. A seguir, é apresentada como essa modelagem em grafos locais é realizada.

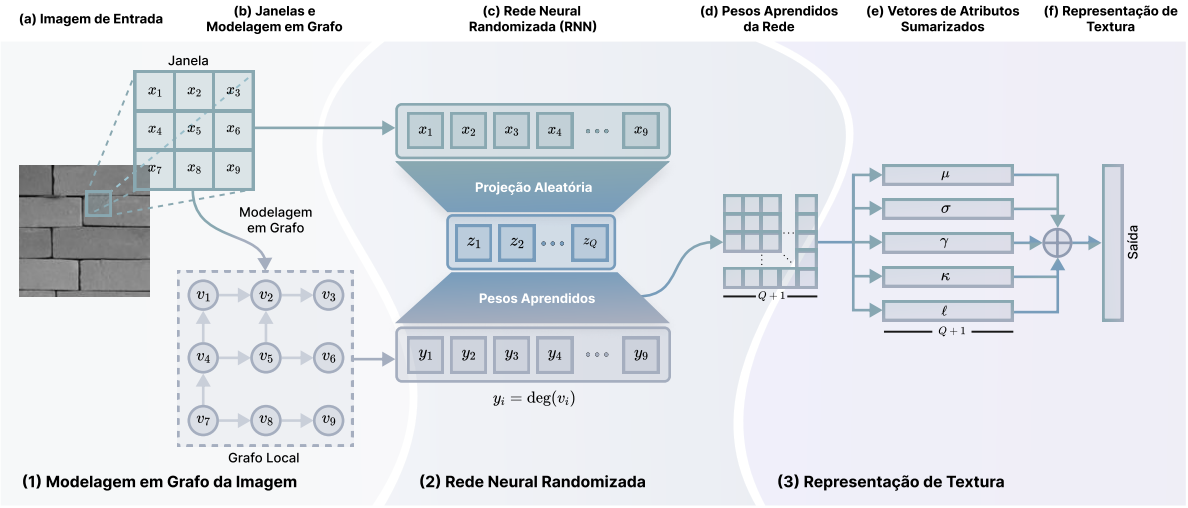
Modelagem em Grafo. Seja $T^{(k)} \in \mathbb{R}^{J\times J}$ uma janela ou trecho (i.e., um *patch*) de tamanho $J \times J$ obtido a partir de uma imagem $I \in \mathbb{R}^{H\times W}$. A partir desse trecho, esse é esticado produzindo o vetor $\vec{t}^{(k)} \in \mathbb{R}^{J^2}$, e constrói-se o grafo $G^{(k)} = (V^{(k)}, E^{(k)})$, em que para cada pixel $t_i^{(k)}$ na janela, existe um vértice $v_i^{(k)} \in V^{(k)}$ correspondente, e existirá uma aresta direcionada do vértice $v_i^{(k)}$ para $v_j^{(k)}$, desde que a intensidade do pixel referente ao vértice de saída seja estritamente maior que o vértice de entrada. Matematicamente, tem-se que:

$$e_{ij}^{(k)} \in E^{(k)} \iff P(v_i^{(k)}) > P(v_j^{(k)})\,, \tag{6}$$

em que $P(v_i^{(k)})$ é a intensidade do pixel na janela $T^{(k)}$ modelado pelo vértice $v_i^{(k)}$.

Construção da Representação. Seja $I \in \mathbb{R}^{H\times W}$ uma imagem em nível de cinza. A partir dessa imagem, janelas de tamanho 3×3 são centralizados em cada pixel k da imagem I, com objetivo de capturar os padrões mais locais na textura, e, em que $T^{(k)} \in \mathbb{R}^{3\times 3}$ denomina a janela centrada no pixel k. Em seguida, com as janelas centralizadas em todos os pixels da imagem, obtém-se o conjunto $\mathcal{T} = \{T^{(1)}, T^{(2)}, \ldots, T^{(H\cdot W)}\}$. A partir desse, será construída

Figura 13 – Ilustração da metodologia proposta em Fares, Guerra & Ribas (2025) para aprendizado de representações de textura a partir de modelagem em grafos locais.



Fonte: Adaptada de Fares, Guerra & Ribas (2025).

as matrizes dos vetores de atributos de entrada e saída da rede neural randomizada, a qual fornecerá os pesos da camada de saída que serão utilizados como representação de textura após pós-processamento. Nesse sentido, para a construção da matriz dos atributos de entrada, tem-se que cada janela $T^{(k)} \in \mathcal{T}$ será esticada, de modo a formar a matriz coluna de entrada $\vec{x}^{\,(k)} \in \mathbb{R}^{9\times 1}$ e horizontalmente concatenada, de modo que a matriz de entrada é dada por:

$$X = \left[\vec{x}^{\,(1)}, \vec{x}^{\,(2)}, \ldots, \vec{x}^{\,(H\cdot W)}\right] \in \mathbb{R}^{9\times HW}\,. \tag{7}$$

Dessa forma, os atributos de entrada referem-se as intensidades dos padrões de locais de textura provenientes das janelas centradas em cada pixel. Por outro lado, para a construção da matriz dos atributos de saída, tem-se que para cada janela $T^{(k)}$ é montado o grafo $G^{(k)} = (V^{(k)}, E^{(k)})$ empregando a modelagem apresentada anteriormente, e constrói-se a matriz coluna $\vec{y}^{\,(k)} = [y_1^{(k)}, y_2^{(k)}, \ldots, y_9^{(k)}]^T \in \mathbb{R}^{9\times 1}$, em que $y_i^{(k)} = \deg(v_i^{(k)})$, e

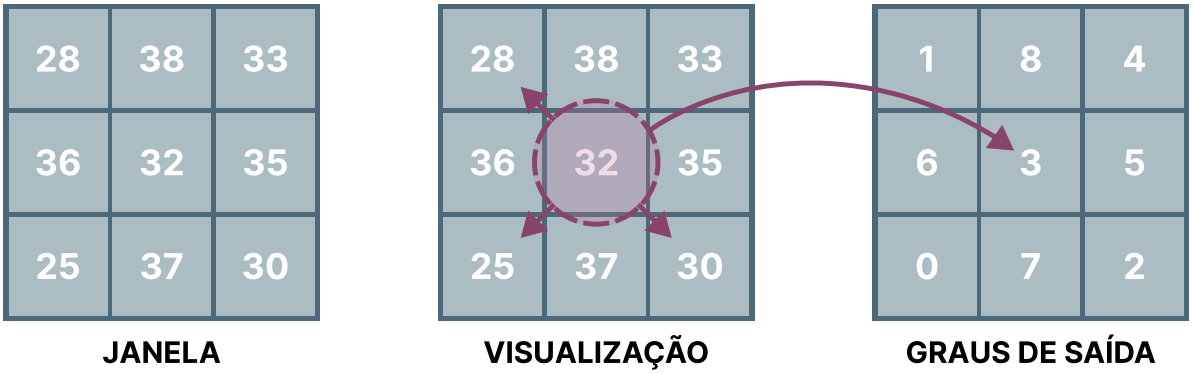
$$\deg(v_i^{(k)}) = \sum_j \begin{cases} 1, & \text{se } e_{ij}^{(k)} \in E^{(k)} \\ 0, & \text{caso contrário} \end{cases}\,. \tag{8}$$

Nesse contexto, os atributos de saída referem-se aos graus de saída dos vértices que estão modelando os pixels provenientes dos padrões locais de textura na imagem, como ilustrado na Figura 13(b). Portanto, a matriz de saída é formulada a partir da concatenação horizontal das matrizes coluna de saída, sendo a matriz de saída expressa por:

$$Y = \left[\vec{y}^{\,(1)}, \vec{y}^{\,(2)}, \ldots, \vec{y}^{\,(H\cdot W)}\right] \in \mathbb{R}^{9\times HW}\,. \tag{9}$$

A Figura 14 apresenta uma visualização de como os graus de saída, calculados a partir da modelagem em grafos locais apresentada anteriormente, se relacionam com a janela de entrada modelada pelo grafo local.

Figura 14 – Ilustração do processo de obtenção dos graus de saída do grafo local cujos vértices modelam o relacionamento das intensidades da janela apresentada.



Fonte: Elaborada pelo autor.

Isso posto, a ideia do método é forçar a arquitetura de rede neural randomizada a predizer o grau de saída dos vértices provenientes da modelagem em grafo apresentada, somente a partir das intensidades dos pixels modelados por tais vértices. À vista disso, a rede é forçada a cruzar tais informações visando que essa aprenda pesos da camada de saída que contenham informações mais discriminativas dos padrões de textura de entrada.

Assim, antes de aplicar efetivamente a rede neural randomizada, é necessário gerar a matriz de pesos aleatórios, a qual é gerada utilizando a mesma técnica apresentada na Subseção 4.1.1, porém, diferentemente, será utilizado $L = Q \cdot (p+1)$, e o vetor $V \in \mathbb{R}^L$ resultante será redimensionado para $(Q, p+1)$, obtendo, portanto, a matriz de pesos aleatórios $U \in \mathbb{R}^{Q\times(p+1)}$. Dessa forma, com as matrizes de entrada, de saída e de pesos aleatórios construídas, tem-se que os pesos da camada de saída M são obtidos da seguinte forma:

$$M = \mathsf{RNN}(X, Y; U, \lambda) \in \mathbb{R}^{p\times(Q+1)}\,, \tag{10}$$

em que $\lambda = 10^3$ é o valor da regularização de Tikhonov utilizado neste método. Essas etapas são ilustradas na Figura 13(c) e (d). A partir dos pesos de saída, esses são utilizados para construir o vetor de atributos parcial sumarizado ilustrado na Figura 13(e), e expresso por:

$$\vec{\Theta}_f(Q) = [f(\vec{m}_1), f(\vec{m}_2), \ldots, f(\vec{m}_{Q+1})] \in \mathbb{R}^{Q+1}\,, \tag{11}$$

em que $\vec{m}_k \in \mathbb{R}^p$ representa a k-ésima coluna de M, e $f(\vec{m}_k) \in \mathbb{R}$ representa o valor sumarizado da k-ésima coluna pela função de sumarização $f : \mathbb{R}^p \to \mathbb{R}$. A partir disso, com a finalidade de introduzir diferentes perspectivas dos pesos da camada de saída, pode-se usar múltiplas medidas de sumarização, tais como medidas estatísticas ou de norma, as quais se exibiram úteis (Fares; Ribas, 2024b). Dessa forma, a seguinte representação de textura parcial é construída a partir da agregação de distintas medidas estatísticas e de norma, tal como apresentada em Fares, Guerra & Ribas (2025), e a qual é expressa por:

$$\vec{\Omega}(Q) = \left[\vec{\Theta}_\mu(Q), \vec{\Theta}_\sigma(Q), \vec{\Theta}_\gamma(Q), \vec{\Theta}_\kappa(Q), \vec{\Theta}_\ell(Q)\right] \in \mathbb{R}^{5\cdot(Q+1)}\,, \tag{12}$$

em que μ, σ, γ e κ denotam as medidas estatísticas de média, desvio padrão, assimetria e curtose, respectivamente, e, ℓ denota a norma euclidiana quadrada.

Por fim, como investigado em Sá Junior & Backes (2016), Ribas et al. (2020), Ribas et al. (2024a), Ribas et al. (2024b), adota-se como representação final aquela que agrega as representações parciais obtidas a partir de projeções realizadas em espaços de diferentes dimensionalidades, permitindo que diferentes informações sobre o conteúdo textural de entrada possam ser capturadas. Dessa forma, a representação final é expressa por:

$$\vec{\Phi}(Q_1, Q_2, \ldots, Q_N) = \left[\vec{\Omega}(Q_1), \vec{\Omega}(Q_2), \ldots, \vec{\Omega}(Q_N)\right] \in \mathbb{R}^m\,, \tag{13}$$

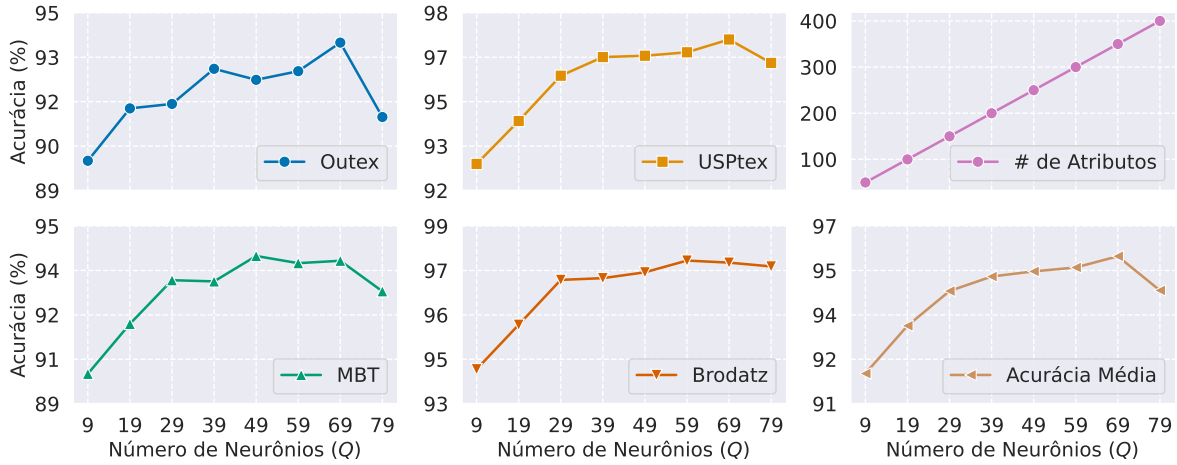
em que $m = 5 \cdot \sum_{i=1}^{N}(Q_i + 1)$ é a dimensionalidade da representação proposta.

Análise de Parâmetros. A análise dos parâmetros da metodologia proposta usou procedimentos semelhantes àqueles empregados na análise de parâmetros do método proposto na Subseção 4.1.1. Isto é, empregou-se como métrica principal a acurácia, a qual é calculada utilizando o classificador *Linear Discriminant Analysis* com a estratégia de validação cruzada *Leave-One-Out*, e obteve-se o desempenho de classificação nos conjuntos de dados da Outex (Ojala et al., 2002), USPtex (Backes; Casanova; Bruno, 2012), Brodatz (Brodatz, 1966) e MBT (Abdelmounaime; Dong-Chen, 2013), esses apresentados em detalhe no Apêndice A.

Uma vez que a configuração experimental está definida, primeiramente, analisou-se o comportamento do desempenho de classificação da representação de textura parcial $\vec{\Omega}(Q)$ à medida que alterou-se a dimensão da camada escondida. Para isso, o espaço de parâmetros em que Q varia é especificado por $\mathcal{Q} = \{9, 19, \ldots, 79\}$. Dessa forma, a Figura 15 apresenta o comportamento do desempenho para cada conjunto de dados analisado e do número de atributos da representação em questão. A partir dela, é possível notar que até $Q = 29$ a representação parcial proposta apresenta contínuos crescimentos, sendo esses mais acentuados para os conjuntos de dados da USPtex, Brodatz e MBT. Entre $Q = 29$ e $Q = 69$, embora o desempenho de classificação apresente ligeiros aumentos nos conjuntos da USPtex e Brodatz e certa flutuação nos conjuntos de dados da Outex e MBT, a acurácia em $Q = 69$ foi maior que $Q = 29$ em todos os conjuntos analisados. Nesse contexto, pode-se afirmar que o desempenho da representação de textura parcial foi bem comportado e apresentou bons resultados até $Q = 69$. Isso sugere que os padrões texturais foram melhor capturados, à medida que os atributos de entrada foram projetados em espaços de dimensões maiores, permitindo que mais características (i.e., atributos) pudessem ser aprendidas, as quais se mostraram úteis no processo de reconhecimento de texturas.

No entanto, com $Q = 79$ tem-se uma queda no desempenho de classificação em todos os conjuntos de dados analisados, sendo esta queda acentuada para a Outex, moderada para a MBT e USPtex, e sutil para Brodatz. Isso indica que a formulação da representação parcial não lidou adequadamente com dimensões mais altas da camada escondida do que $Q = 69$, de maneira que, não necessariamente o aumento de Q implicou em aumento de acurácia

Figura 15 – Comportamento da acurácia de classificação e do número de atributos da representação de textura parcial $\vec{\Omega}(Q)$ proposta em Fares, Guerra & Ribas (2025).



Fonte: Adaptada de Fares, Guerra & Ribas (2025).

(Fares; Guerra; Ribas, 2025). Nesse contexto, para aproveitar os atributos aprendidos pelas representações parciais com distintas dimensões, foi proposta a representação de textura final $\vec{\Phi}(Q_1, Q_2, \ldots, Q_N)$, que será analisada a seguir:

Como Q varia até 79, tem-se que uma única representação parcial obterá no máximo até $5 \cdot (79+1) = 400$ atributos. Dessa forma, considerando que a representação final consiste na agregação das representações parciais e visando obter representações mais compactas, analisou-se somente a agregação de duas representações parciais distintas. Nesse contexto, o espaço de parâmetros considerado na análise é expresso por:

$$\mathcal{E}_{\text{ELGNN}} = \{(Q_1, Q_2) \in \mathcal{Q} \times \mathcal{Q} \mid Q_1 < Q_2\}. \tag{14}$$

Logo, as representações finais a serem analisadas são apenas aquelas cujos parâmetros pertencem ao conjunto $\mathcal{Q}$ e cujo primeiro parâmetro é estritamente menor que o segundo.

A Tabela 2 apresenta o desempenho da representação de textura $\vec{\Phi}(Q_1, Q_2)$, exibindo uma melhora na acurácia média das representações com diversas combinações, obtendo acima de 96%, enquanto que na análise da representação parcial $\vec{\Omega}(Q)$ obteve-se no máximo 95,74% de acurácia média. Nesse contexto, como esperado, a combinação de representações parciais com dimensões menores apresentou melhor caracterização dos padrões texturais em relação as representações parciais com elevadas dimensões, mesmo quando ambas têm o mesmo número de atributos. Como exemplo, enquanto a representação parcial $\vec{\Omega}(59)$ obteve acurácia média de 95,37%, a representação final $\vec{\Phi}(19, 39)$ obteve acurácia média de 95,55%, ambas com 300 atributos. Assim, isso demonstra que a complementaridade advinda da caracterização dos mesmos padrões texturais a partir de distintas projeções e cruzando as informações topológicas dos grafos modelados foi benéfica para a construção de uma representação mais discriminativa.

Em conclusão, para efeito de comparação com outros métodos da literatura em seções

Tabela 2 – Acurácias de classificação para representação de textura $\vec{\Phi}(Q_1, Q_2)$. Os resultados em **destaque** obtiveram as maiores acurácias médias. ↑ indica que maiores resultados são preferíveis.

(Q_1, Q_2)	# de Atributos	Acurácia (↑)				
		Outex	USPtex	Brodatz	MBT	Média
(09, 19)	150	91,54	95,90	96,45	93,38	94,32
(09, 29)	200	92,57	96,64	97,30	94,28	95,20
(09, 39)	250	93,09	97,21	97,18	95,05	95,63
(09, 49)	300	92,79	97,08	97,47	94,81	95,54
(09, 59)	350	93,16	97,08	97,35	94,28	95,47
(09, 69)	400	93,60	97,73	97,75	94,81	95,97
(09, 79)	450	91,69	96,86	97,52	93,79	94,97
(19, 29)	250	92,65	96,99	97,24	94,40	95,32
(19, 39)	300	93,46	97,03	96,96	94,76	95,55
(19, 49)	350	92,72	97,16	97,30	95,05	95,56
(19, 59)	400	93,16	97,51	97,86	94,64	95,79
(19, 69)	450	93,75	97,86	97,24	95,09	95,99
(19, 79)	500	92,65	97,47	97,35	94,07	95,39
(29, 39)	350	93,31	97,34	97,58	95,01	95,81
(29, 49)	400	92,65	97,47	97,86	94,72	95,68
(29, 59)	450	92,79	97,60	97,97	94,44	95,70
(29, 69)	500	93,75	97,82	97,58	94,68	95,96
(29, 79)	550	92,35	97,51	97,97	94,20	95,51
(39, 49)	**450**	**94,49**	**97,77**	**97,75**	**94,60**	**96,15**
(39, 59)	**500**	**93,53**	**97,56**	**97,97**	**94,93**	**96,00**
(39, 69)	550	93,60	97,77	97,92	94,72	96,00
(39, 79)	600	92,65	97,60	97,64	94,44	95,58
(49, 59)	550	93,31	97,56	97,47	94,85	95,80
(49, 69)	**600**	**93,53**	**97,99**	**97,69**	**94,93**	**96,04**
(49, 79)	650	92,72	97,47	97,64	95,13	95,74
(59, 69)	650	92,94	97,86	98,09	94,64	95,88
(59, 79)	700	92,50	97,82	97,52	94,60	95,61
(69, 79)	750	93,01	97,64	97,97	94,20	95,71

Fonte: Adaptada de Fares, Guerra & Ribas (2025).

posteriores, selecionam-se como representações propostas aquelas que obtiveram os três melhores desempenhos médios. Essas são, ELGNN$_a$ ($\vec{\Phi}(39, 49)$), ELGNN$_b$ ($\vec{\Phi}(49, 69)$) e ELGNN$_c$ ($\vec{\Phi}(39, 59)$)[1], as quais obtiveram as acurácias de 96,15%, 96,04% e 96,00%, respectivamente.

Contribuições e Limitações. As principais contribuições do método proposto são duas. A primeira relaciona-se ao emprego de uma modelagem em grafos locais a partir de janelas de tamanho 3×3, permitindo a captura de detalhes texturais mais finos e, portanto, melhor

[1] Deve-se notar que ambas as representações finais $\vec{\Phi}(39, 59)$ e $\vec{\Phi}(39, 69)$ obtiveram a mesma acurácia média (após arredondamentos). No entanto, a primeira foi selecionada, pois obteve mínima diferença de acurácia com uma quantidade menor de atributos (redução de aproximadamente 9,09%).

caracterização textural da imagem de entrada. A segunda diz respeito ao cruzamento de informações imposto à rede neural randomizada, a qual é forçada a cruzar as informações entre as intensidades dos pixels das janelas de entrada e os graus de saída dos vértices que modelam tais pixels, assim, propiciando o aprendizado de características mais complexas. Além disso, ainda que de forma secundária, há uma terceira contribuição, que consiste no uso de medidas estatísticas de ordem mais alta (e.g., curtose) e no uso de normas (e.g., norma euclidiana), com a finalidade de sumarizar a matriz dos pesos aprendidos da camada de saída.

Apesar das contribuições, o método proposto apresenta as seguintes limitações. Primeiramente, a ausência de uma análise multiescala, conduzida pelo emprego de janelas de múltiplos tamanhos a serem modeladas. Nessa perspectiva, a abordagem tenderia a analisar tanto padrões extremamente locais quanto aqueles mais globais. Em segundo lugar, trabalhos futuros podem-se beneficiar do emprego de outras medidas de grafo (e.g., força), permitindo não só a inserção de informações essenciais sobre a caracterização textural, mas também a oportunidade de realizar distintos cruzamentos de informações (e.g., intensidade→força, graus→força, força→graus etc.).

4.1.3 Comparação com Métodos da Literatura

Esta subseção dedica-se à análise comparativa e discussão entre os resultados alcançados pelas técnicas de aprendizado de representações de textura em nível de cinza propostas nas subseções anteriores e aqueles reportados na literatura. Nesse contexto, a seguir, serão apresentados o protocolo experimental empregado na análise comparativa, os conjuntos de dados de *benchmark* utilizados e as abordagens da literatura consideradas para comparação.

Protocolo Experimental. Com a finalidade de garantir consistência na comparação dos resultados obtidos pelos métodos propostos e aqueles apresentados na literatura, tem-se que todos os resultados apresentados na Tabela 3 foram obtidos empregando o classificador *Linear Discriminant Analysis* (LDA) com a estratégia de validação cruzada *Leave-One-Out* (LOO), exceto para a abordagem CLBP (Guo; Zhang; Zhang, 2010a) que empregou o classificador *1-nearest-neighbor* (i.e., 1-NN). Em seguida, para a análise comparativa entre as abordagens foram utilizados quatro conjuntos de imagens de textura, sendo eles: Outex (Ojala et al., 2002), USPtex (Backes; Casanova; Bruno, 2012), Brodatz (Brodatz, 1966) e MBT (Abdelmounaime; Dong-Chen, 2013). Em particular, os conjuntos da Outex, USPtex e MBT consistem originalmente por imagens coloridas. Dessa forma, por se tratar da análise de técnicas de aprendizado de representações de textura em nível de cinza, as imagens são convertidas para esse formato antes de serem submetidas ao processo de extração de características.

Quanto às abordagens da literatura incluídas na comparação, tem-se que essas foram divididas em três categorias, sendo elas: métodos construídos por especialistas ou artesanais (do inglês, *handcrafted*), métodos baseados em redes neurais randomizadas (RNNs, do inglês

Randomized Neural Networks) e métodos baseados em redes neurais convolucionais profundas (DCNNs, do inglês *Deep Convolutional Neural Networks*). Dessa forma, obtém-se uma comparação com um amplo conjunto de abordagens fundamentadas em técnicas distintas, consequentemente, enriquecendo o processo de análise e discussão dos métodos propostos.

Nesse contexto, na categoria de métodos construídos por especialistas ou artesanais, tem-se que os empregados na comparação são: GLCM (Haralick; Shanmugam; Dinstein, 1973), GLDM (Weszka; Dyer; Rosenfeld, 1976), Filtros de Gabor (Manjunath; Ma, 1996), Fractal (Backes; Casanova; Bruno, 2009), Fractal Fourier (Florindo; Bruno, 2012), LBP (Ojala; Pietikainen; Maenpaa, 2002), LBPV (Guo; Zhang; Zhang, 2010b), CLBP (Guo; Zhang; Zhang, 2010a), AHP (Zhu et al., 2015), BSIF (Kannala; Rahtu, 2012), LCP (Guo; Zhao; Pietikäinen, 2011), LFD (Maani; Kalra; Yang, 2013), LPQ (Ojansivu; Heikkilä, 2008), CNTD (Backes; Casanova; Bruno, 2013) e SSN (Scabini; Ribas; Bruno, 2020). Por outro lado, na categoria dos métodos baseados em redes neurais randomizadas, as abordagens são: ELM (Sá Junior; Backes, 2016), CNRNN (Ribas et al., 2020), LCFNN (Ribas et al., 2024b) e SSR (Ribas et al., 2024a). Por fim, para a categoria dos métodos baseados em redes neurais convolucionais profundas, as arquiteturas consideradas na comparação são: VGG19 (Simonyan; Zisserman, 2014), ResNet50 (He et al., 2016), InceptionV3 (Szegedy et al., 2016), InceptionResNetV2 (Szegedy et al., 2017) e ConvNeXt V2 (*Nano* e *Large*) (Woo et al., 2023). Os modelos dessas arquiteturas, pré-treinados na ImageNet-1K (Deng et al., 2009), foram utilizados como extratores de características, empregando-se como representação de textura o *Global Average Pooling* (GAP) da última camada anterior à cabeça de classificação.

Análise Comparativa. Analisando-se a Tabela 3, para o conjunto de dados da Outex, observa-se que os três melhores resultados foram obtidos pelas representações propostas neste trabalho de mestrado, sendo essas: ELGNN$_a$, REE$_c$ e REE$_b$, as quais alcançaram as acurácias de 94,49%, 93,97% e 93,82%, respectivamente. Em particular, entre as abordagens manuais, ao contrastar a melhor representação geral ELGNN$_a$ com a SSN (Scabini; Ribas; Bruno, 2020), que obteve a maior acurácia de 89,04%, observa-se que o incremento de 5,45% pela técnica proposta demonstra os benefícios das abordagens de representação de texturas baseadas em aprendizado, sobretudo aquelas que empregam redes neurais randomizadas. Por outro lado, em relação às abordagens baseadas em DCNNs, tem-se que a ConvNeXt V2-*Nano* (Woo et al., 2023) foi aquela que obteve a maior acurácia de 88,01%, de modo que, ELGNN$_a$ exibe um aumento de 6,48%. Especificamente, sabe-se que as técnicas baseadas exclusivamente em DCNNs revelam-se ineficientes quando aplicadas à Outex, dado que esse conjunto é composto predominantemente por padrões microtexturais (Scabini; Ribas; Bruno, 2020; Ribas et al., 2024a). Dessa forma, a representação proposta neste trabalho exibiu-se discriminativa o suficiente na caracterização microtextural, como mostrado nos resultados. Por fim, no que se refere à comparação com as técnicas baseadas em RNNs, ponto central desta análise, observa-se que a ELGNN$_a$ apresentou um acréscimo de 2,36% em relação à LCFNN, que registrou a maior acurácia de 92,13% entre os métodos comparados dessa categoria.

Tabela 3 – Comparação das acurácias obtidas pelos métodos propostos com aqueles presentes na literatura. Uma parte dos resultados apresentados foram obtidos em Ribas et al. (2024b) e Fares, Guerra & Ribas (2025). Em cada coluna, o maior valor está em **destaque**, o valor em azul indica o melhor resultado da literatura, e em vermelho indica resultados iguais ou superiores por parte dos métodos propostos.

Método	# de Atributos	Acurácia (↑)			
		Outex	USPtex	Brodatz	MBT
Métodos Artesanais					
GLCM	24	80,73	83,64	90,43	85,88
GLDM	60	86,76	92,06	94,43	92,78
Filtros de Gabor	48	81,91	89,22	89,86	89,94
Fourier	63	81,91	67,50	75,90	67,82
Fractal	69	80,51	78,27	87,16	74,07
Fractal Fourier	68	68,38	59,47	71,96	70,29
LBP	256	81,10	85,43	93,64	84,74
LBPV	555	75,66	54,97	86,26	73,54
CLBP	648	85,80	91,14	95,32	80,28
AHP	120	88,31	94,85	94,88	85,35
BSIF	256	77,43	77,66	91,44	79,71
LCP	81	86,25	91,14	93,47	84,13
LFD	276	82,57	83,55	90,99	80,20
LPQ	256	79,41	85,12	92,51	74,64
CNTD	108	86,76	91,71	95,27	83,70
SSN (nível de cinza)	108	89,04	94,37	95,50	90,83
Métodos Baseados em RNNs					
ELM	180	89,71	95,11	95,27	–
CNRNN_c	240	91,32	96,95	96,06	91,23
LCFNN	330	92,13	97,21	98,09	93,22
SSR_a (nível de cinza)	630	90,80	95,80	96,51	90,10
SSR_b (nível de cinza)	990	91,60	96,30	96,40	91,00
Métodos Baseados em DCNNs					
VGG19	512	76,62	93,19	96,79	87,42
ResNet50	2048	65,66	62,30	81,98	85,51
InceptionV3	2048	86,40	96,77	98,54	82,14
InceptionResNetV2	1536	85,88	96,34	**98,99**	88,60
ConvNeXt V2-Nano	648	88,01	97,95	98,59	90,79
ConvNeXt V2-Large	1536	86,62	97,51	96,06	93,30
Métodos Propostos					
REE_a	540	93,68	**98,43**	97,24	93,79
REE_b	450	93,82	98,08	97,07	93,95
REE_c	540	93,97	97,95	97,47	93,47
ELGNN_a	450	**94,49**	97,77	97,75	94,60
ELGNN_b	600	93,53	97,99	97,69	**94,93**
ELGNN_c	500	93,53	97,56	97,97	**94,93**

*O hífen indica que o valor não estava disponível ou não foi possível obtê-lo.

Fonte: Dados da pesquisa.

Em relação à USPtex, tem-se que as representações de maior desempenho também foram aquelas propostas no presente trabalho de mestrado, sendo essas: REE_a, REE_b e ELGNN_b, as quais obtiveram as acurácias de 98,43%, 98,08% e 97,99%, respectivamente. Ao comparar com a representação AHP (Zhu et al., 2015), que apresenta a maior acurácia de 94,85% entre os métodos manuais, observa-se que a melhor representação proposta, REE_a, supera esse resultado em 3,58%. Com relação às abordagens baseadas em DCNNs, a melhor representação dessa categoria foi obtida pela ConvNeXt V2-*Nano* (Woo et al., 2023) que alcançou 97,95%, evidenciando uma representação muito competitiva. No entanto, a representação REE_a ainda supera tal desempenho em 0,48%, com um número menor de atributos e prescindindo de um modelo de rede neural profunda pré-treinado em um conjunto contendo milhões de imagens. Por fim, entre as abordagens baseadas em RNNs, LCFNN (Ribas et al., 2024b) apresentou o melhor desempenho da categoria, de 97,21%. Nesse contexto, de forma análoga ao observado na Outex, tem-se uma melhoria atingida pelas representações de textura propostas, sendo que REE_a registrou um aumento de desempenho de 1,22%.

Para a Brodatz, as melhores representações foram aquelas obtidas por métodos baseados em DCNNs, sendo elas: InceptionResNetV2 (Szegedy et al., 2017), ConvNeXt V2-*Nano* (Woo et al., 2023) e InceptionV3 (Szegedy et al., 2016), as quais alcançaram as acurácias de 98,99%, 98,59% e 98,54%, respectivamente. Dessa forma, a discriminabilidade textural na Brodatz beneficiou-se do poder de representação dos modelos de redes neurais convolucionais profundas pré-treinados em conjuntos massivos de dados. No entanto, nota-se que a representação ELGNN_c, que dentre as representações propostas obteve a maior acurácia de 97,97% com apenas 500 atributos, está, aproximadamente, 1% abaixo em relação à InceptionResNetV2 (Szegedy et al., 2017) que obteve o melhor desempenho com 1536 atributos. Nesse contexto, fica claro que, embora as DCNNs tenham obtido os melhores resultados, as técnicas propostas entregaram resultados competitivos utilizando abordagens leves sem a necessidade de treinamento por retropropagação. Por fim, em relação às abordagens baseadas em RNNs, as técnicas propostas foram superadas exclusivamente por LCFNN (Ribas et al., 2024b), cuja acurácia de 98,09% indica uma sutil diferença de 0,12% em relação a ELGNN_c.

Com relação à MBT, observa-se de maneira similar à Outex e USPtex, que as representações com as melhores acurácias foram obtidas pelas abordagens propostas, especificamente: ELGNN_c, ELGNN_b, ELGNN_a e REE_b, obtendo as acurácias de 94,93%, 94,93%, 94,60% e 93,95%, respectivamente. Em particular, ELGNN_c será selecionada como representação principal a ser comparada desde que obteve a maior acurácia com menor número de atributos. Nesse contexto, com relação às técnicas manuais, a técnica ELGNN_c obteve um incremento de 2,15% em relação à técnica GLDM (Weszka; Dyer; Rosenfeld, 1976). Esta última, aliás, representa a melhor técnica manual, alcançando 92,78%. Com relação às abordagens baseadas em DCNNs, tem-se que ELGNN_c superou a ConvNeXt V2-*Large* (Woo et al., 2023), que é a melhor técnica entre as DCNNs. Observa-se que essa apresentou um aumento de 1,63% e, adicionalmente, empregou 1036 atributos a menos, o que indica uma redução de

67,45% em sua dimensionalidade, assim, reforçando a eficiência da representação. Por fim, em relação às técnicas baseadas em RNNs, a representação ELGNN$_c$ alcançou um acréscimo de 1,71% em relação à técnica LCFNN (Ribas et al., 2024b), a qual por sua vez representa a melhor abordagem comparada, obtendo um desempenho de 93,22%.

Análise de Eficiência Computacional. Para a análise da eficiência computacional dos métodos propostos e comparados, adotou-se a mesma metodologia descrita em Fares & Ribas (2025b). Nesse contexto, utilizou-se na comparação os tempos médios de processamento obtidos em 100 repetições após dez repetições de descarte (i.e., repetições de aquecimento). A utilização dessas repetições de aquecimento foram com o objetivo de evitar medições discrepantes. Dessa forma, buscou-se conferir maior robustez ao processo de análise de desempenho. Com relação à configuração experimental do servidor utilizado durante a análise da eficiência computacional, tem-se que esse é munido de um Intel(R) Core(TM) i9-14900KF @ 3.2 GHz com 128 GB de memória RAM e executando o sistema operacional Ubuntu 22.04.03 LTS x86_64. Além disso, todos os métodos comparados de natureza artesanal ou baseados em RNNs foram implementados em MATLAB (R2023b), ao passo que as técnicas propostas foram implementadas e executadas em Python v3.11.5, utilizando o pacote PyTorch v2.6.0.

A Tabela 4 apresenta os tempos de processamento de um subconjunto[2] dos métodos comparados. Em particular, observa-se que os métodos artesanais apresentaram os menores tempos de processamento, sendo esses: LBP (Ojala; Pietikainen; Maenpaa, 2002), Fractal Fourier (Florindo; Bruno, 2012) e GLDM (Weszka; Dyer; Rosenfeld, 1976) cujos tempos foram de 6,1 ms, 7,7 ms e 9,2 ms, respectivamente. No entanto, já era esperado que determinados métodos artesanais apresentassem os menores tempos de processamento, uma vez que, em geral, constituem técnicas com baixa demanda computacional. Dessa forma, embora esses métodos apresentem os melhores desempenhos em termos de eficiência computacional, suas acurácias mostram-se menos expressivas quando comparadas às das técnicas propostas, em todos os conjuntos de dados analisados. Por exemplo, na Outex, as técnicas LBP, Fractal Fourier e GLDM obtiveram 81,10%, 68,38% e 86,76%, ao passo que as três representações propostas mais rápidas (REE$_b$, REE$_c$ e ELGNN$_a$) alcançaram 93,82%, 93,97% e 94,49%, respectivamente. Portanto, abordagens mais robustas serão aquelas que apresentam melhor equilíbrio entre eficiência computacional e desempenho em tarefas finais (e.g., acurácia).

Nesse contexto, como parte principal desta análise, serão comparados os métodos propostos com aqueles baseados em RNNs. Desse modo, a partir da Tabela 4, nota-se que as representações propostas possuem os menores tempos de processamento. Especificamente, enquanto a abordagem comparada com menor tempo de execução é a LCFNN (Ribas et al., 2024b), com 343,1 ms, a técnica proposta mais eficiente, REE$_b$ requer apenas 20,8 ms, resultando em uma redução de tempo equivalente a um fator de 16,5×. Ainda, além da maior

[2] Este subconjunto é constituído pelos métodos da literatura para os quais o código-fonte está disponível, possibilitando a realização das medições necessárias para a análise de eficiência computacional.

Tabela 4 – Comparação do tempo de processamento médio necessário para a extração das características de uma imagem em nível de cinza de dimensão 224×224. Os resultados nos seguintes formatos indicam o **primeiro**, **segundo** e terceiro melhor resultado. A unidade de medida utilizada é segundos (s).

Método	Tempo de CPU (↓)
Métodos Artesanais	
GLCM	0,0133 ± 0,0024
GLDM	0,0092 ± 0,0020
Filtros de Gabor	0,0492 ± 0,0046
Fourier	0,0182 ± 0,0029
Fractal	0,9519 ± 0,0144
Fractal Fourier	**0,0077 ± 0,0020**
LBP	**0,0061 ± 0,0018**
LBPV	0,1364 ± 0,0075
CLBP	0,0753 ± 0,0059
AHP	0,0617 ± 0,0049
BSIF	0,0099 ± 0,0017
LCP	0,0391 ± 0,0047
LFD	0,0720 ± 0,0056
LPQ	0,0724 ± 0,0053
CNTD	0,0865 ± 0,0032
SSN (nível de cinza)	0,0547 ± 0,0027
Métodos Baseados em RNNs	
CNRNN_c	0,3896 ± 0,0276
LCFNN	0,3431 ± 0,0252
SSR_a (nível de cinza)	2,1886 ± 0,1240
SSR_b (nível de cinza)	2,4809 ± 0,2564
Métodos Propostos	
REE_a	0,0775 ± 0,0023
REE_b	0,0208 ± 0,0019
REE_c	0,0220 ± 0,0015
ELGNN_a	0,0447 ± 0,0030
ELGNN_b	0,0465 ± 0,0015
ELGNN_c	0,0463 ± 0,0026

Fonte: Elaborada pelo autor.

eficiência computacional, a técnica REE_b obtém uma acurácia média[3] de 95,73%, enquanto LCFNN alcança 95,16%. Portanto, tais resultados reforçam as vantagens das técnicas propostas em termos de eficiência computacional e desempenho.

Discussões. Desse modo, as técnicas propostas REE e ELGNN demonstraram ótimos de-

[3] A acurácia média em questão é calculada a partir da média dos resultados obtidos para cada conjunto de dados analisado.

sempenhos, evidenciando superioridade em relação às demais técnicas baseadas em RNNs, tais como: CNRNN, LCFNN e SSR. Em particular, as abordagens desenvolvidas obtiveram as três representações de maior acurácia em três dos quatro conjuntos de dados analisados, sendo esses: Outex, USPtex e MBT. Especificamente, na Brodatz, as técnicas perderam apenas para as baseadas em DCNNs e para a LCFNN, esta última por uma diferença sutil.

Além disso, deve-se notar que a técnica proposta ELGNN emprega uma modelagem em grafos dos padrões texturais locais presente na imagem de textura. Nesse sentido, os desempenhos superiores alcançados pela ELGNN, em termos de acurácia, em comparação às demais técnicas baseadas em grafos, como CNRNN, LCFNN e SSR, substanciam as vantagens dessa modelagem, a qual é capaz de capturar de forma mais efetiva um conteúdo textural mais fino. Além disso, os desempenhos superiores obtidos pela abordagem REE, quando comparada às demais técnicas, evidenciam que o aprendizado do conteúdo textural local por meio de múltiplos *autoencoders* randomizados também foi vantajoso.

Por fim, além dos desempenhos expressivos obtidos pelas técnicas investigadas, observa-se que essas também apresentaram elevada eficiência computacional, superando todas as demais abordagens baseadas em RNNs e, de modo geral, mostrando-se competitivas em eficiência em relação aos métodos artesanais. Desse modo, tem-se que as análises e discussões apresentadas mostram as contribuições que as abordagens desenvolvidas trouxeram para as áreas da análise de texturas e reconhecimento de padrões.

4.2 APRENDIZADO DE REPRESENTAÇÕES DE TEXTURA COLORIDA

Nesta seção, serão apresentadas as técnicas de aprendizado de representações de texturas coloridas usando redes neurais randomizadas puras. Em particular, as metodologias investigadas e desenvolvidas neste trabalho focaram em abordagens que aprendessem a informação espaço-espectral presente na textura colorida. Nesse contexto, ao alavancar simultaneamente tais informações, tem-se a obtenção de uma representação de textura mais discriminativa. Desse modo, a Subseção 4.2.1 apresentará uma abordagem de representação de hiperparâmetro único que emprega dois módulos denominados espacial e espectral cuja responsabilidade é aprender as características espaciais e espectrais, respectivamente. Para isso, o módulo espacial aprende os padrões texturais intra-canal a partir do cruzamento dos padrões texturais presentes no próprio canal, enquanto o módulo espectral o faz por meio do cruzamento dos padrões e relacionamentos texturais presentes em canais adjacentes.

Em seguida, a Subseção 4.2.2 exibirá uma técnica simples e direta que utiliza apenas um único *autoencoder* randomizado para o aprendizado dos padrões texturais espaço-espectral. Especificamente, a principal contribuição dessa metodologia não está relacionada à arquitetura, mas como a informação espaço-espectral da imagem colorida é capturada para, posteriormente, alimentar a rede. Em particular, tal informação é capturada a partir da varredura de cubos de

cor de dimensões $3 \times 3 \times 3$ sobre a imagem colorida capturando, assim, simultaneamente a informação espaço-espectral bruta presente na imagem de entrada.

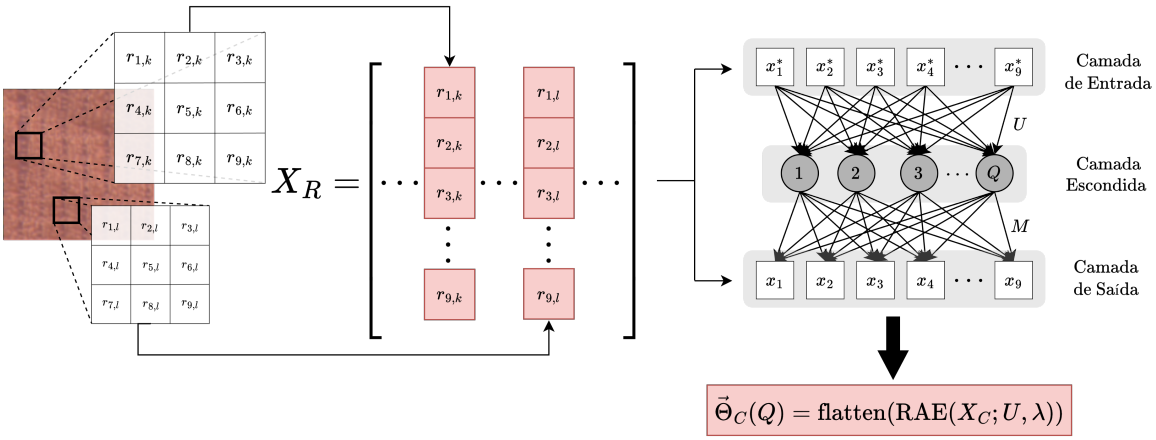
Posteriormente, a Subseção 4.2.3 apresentará a metodologia denominada por Mixer cujas duas principais contribuições referem-se ao módulo de projeção aleatória hiperesférica e ao módulo de aprendizado. Em primeiro lugar, o módulo de projeção aleatória hiperesférica tem como responsabilidade obter as incorporações aleatórias hiperesféricas a partir dos padrões texturais locais, de modo que tais incorporações hiperesféricas, i.e., projetadas sobre a superfície de uma hiperesfera, ajudam a estabilizar o treinamento da rede. Em segundo lugar, o módulo de aprendizado é munido de dois submódulos denominados direto e misto cujas responsabilidades são aprender as características intra- e inter-canal, respectivamente. Em particular, o submódulo misto é munido de uma nova fórmula de otimização que alavanca os padrões texturais de todos os canais para predizer os padrões presentes em um canal específico, assim, enriquecendo o processo de aprendizado. Por fim, a Subseção 4.2.4 realizará a comparação das técnicas propostas com aquelas presentes na literatura.

4.2.1 Aprendizado de Representações Espaço-Espectrais com Redes Neurais Randomizadas

Nesta subseção, detalha-se o método proposto em Fares & Ribas (2024a) para aprendizado de representações espaço-espectrais de textura colorida. Este artigo propõe uma representação espaço-espectral de parâmetro único, a qual é obtida a partir do emprego de dois módulos de aprendizado: o módulo espacial e o módulo espectral. O módulo espacial é responsável por aprender as características e os relacionamentos dos padrões texturais presentes na textura colorida. Para tal, utilizam-se três *autoencoders* randomizados, sendo cada um aplicado ao aprendizado dos aspectos texturais de cada canal da imagem. Ao contrário disso, o módulo espectral tem como função o aprendizado das características espectrais da imagem. Com esse objetivo, empregam-se três redes neurais randomizadas, as quais são treinadas para predizer os relacionamentos locais das intensidades entre canais adjacentes. Nesse contexto, o módulo fundamenta-se na hipótese de que os padrões de textura de um canal podem ser previstos a partir de outro adjacente. Consequentemente, a representação de textura final incorpora informações espaciais e espectrais úteis para caracterização de texturas coloridas.

Construção da Representação. Seja $I \in \mathbb{R}^{3\times H\times W}$ uma imagem colorida no espaço de cor RGB, e $I_C \in \mathbb{R}^{H\times W}$ o canal de cor da imagem I, em que $C \in \{R, G, B\}$. Considerando o canal de cor I_C, estabelece-se a matriz $X_C \in \mathbb{R}^{9\times HW}$ a qual é composta pela justaposição de janelas 3×3 esticadas, que varrem todos os pixels do canal de cor I_C. Essa etapa, a exemplo dos métodos anteriores, desempenha papel fundamental na captura dos padrões de textura, viabilizando a incorporação dessas informações por X_C. Nesse contexto, munido das informações texturais de cada canal de cor, constroem-se dois módulos de aprendizado de representações, os quais são apresentados a seguir:

Figura 16 – Ilustração do módulo espacial responsável pelo aprendizado das características e aparências texturais de um dado canal proposto em Fares & Ribas (2024a).



Fonte: Adaptada de Fares & Ribas (2024a).

- **Módulo Espacial** $\vec{\Theta}_C(Q)$: O módulo espacial, ilustrado na Figura 16, é responsável pelo aprendizado das características e aspectos texturais de cada canal de cor I_C. Para tal, o módulo aplicará um *autoencoder* randomizado para cada canal de cor I_C, empregando-se como entrada as informações texturais contidas em X_C. Nesse contexto, a representação de textura espacial consiste no esticamento dos pesos da camada de saída do decodificador, o qual foi forçado a aprender a reconstruir os padrões de intensidades em I_C. Assim, tem-se que a representação do módulo espacial é dada por:

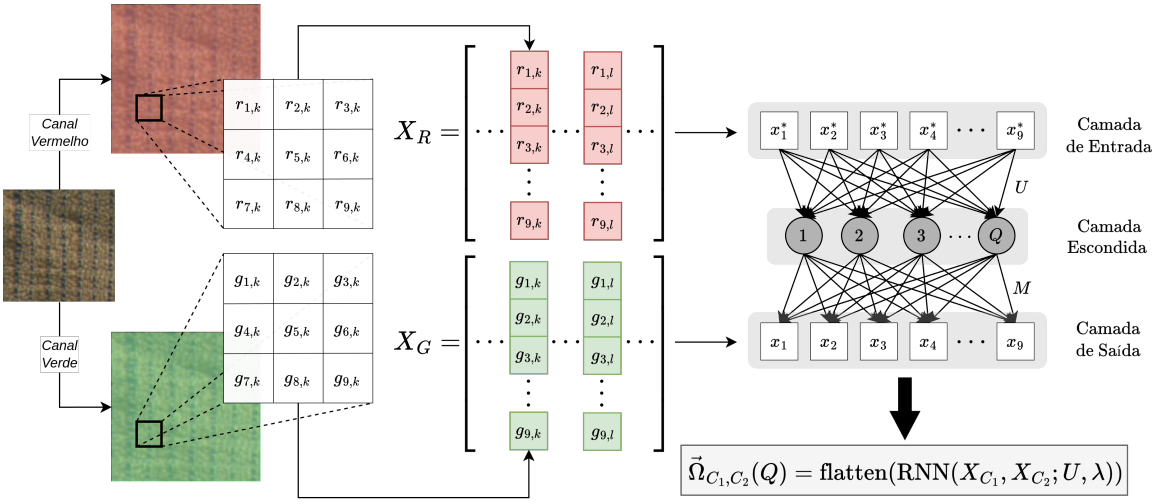
$$\vec{\Theta}_C(Q) = \text{flatten}(\text{RAE}(X_C; U, \lambda)) \in \mathbb{R}^{9\cdot(Q+1)}\,. \tag{15}$$

- **Módulo Espectral** $\vec{\Omega}_{C_1,C_2}(Q)$: O módulo espectral, ilustrado na Figura 17, é responsável pelo cruzamento de informações entre canais de cor adjacentes, com a finalidade de aprender não só características texturais mas também espectrais (i.e., cor). Para isso, dados dois canais de cores adjacentes C_1 e C_2, tem-se que uma rede neural randomizada é treinada forçando a rede a predizer as intensidades dos pixels dos padrões de textura do canal C_2 a partir de C_1. Dessa forma, a rede cruza tais informações e, consequentemente, os pesos da camada de saída capturam as informações essenciais sobre a relação das intensidades entre tais canais. Nesse sentido, a representação espectral é expressa por:

$$\vec{\Omega}_{C_1,C_2}(Q) = \text{flatten}(\text{RNN}(X_{C_1}, X_{C_2}; U, \lambda)) \in \mathbb{R}^{9\cdot(Q+1)}\,. \tag{16}$$

Além disso, tem-se que as matrizes de pesos aleatórios $U \in \mathbb{R}^{9\times(Q+1)}$ de ambos os módulos são construídas pelo mesmo procedimento apresentado na Subseção 4.1.2, e o valor da regularização de Tikhonov empregado é $\lambda = 10^{-3}$, com a finalidade de, em grande parte, prevenir matrizes singulares durante o processo de inversão matricial.

Figura 17 – Ilustração do módulo espectral responsável pelo aprendizado das características e aparências espectrais entre canais adjacentes proposto em Fares & Ribas (2024a).



Fonte: Adaptada de Fares & Ribas (2024a).

A partir dos módulos espacial e espectral, os quais caracterizam os aspectos texturais e espectrais da imagem, respectivamente. Constrói-se a representação de textura espaço-espectral final a partir da agregação das representações aprendidas aplicadas em cada canal e par de canais adjacentes. Portanto, a representação espaço-espectral é expressa por:

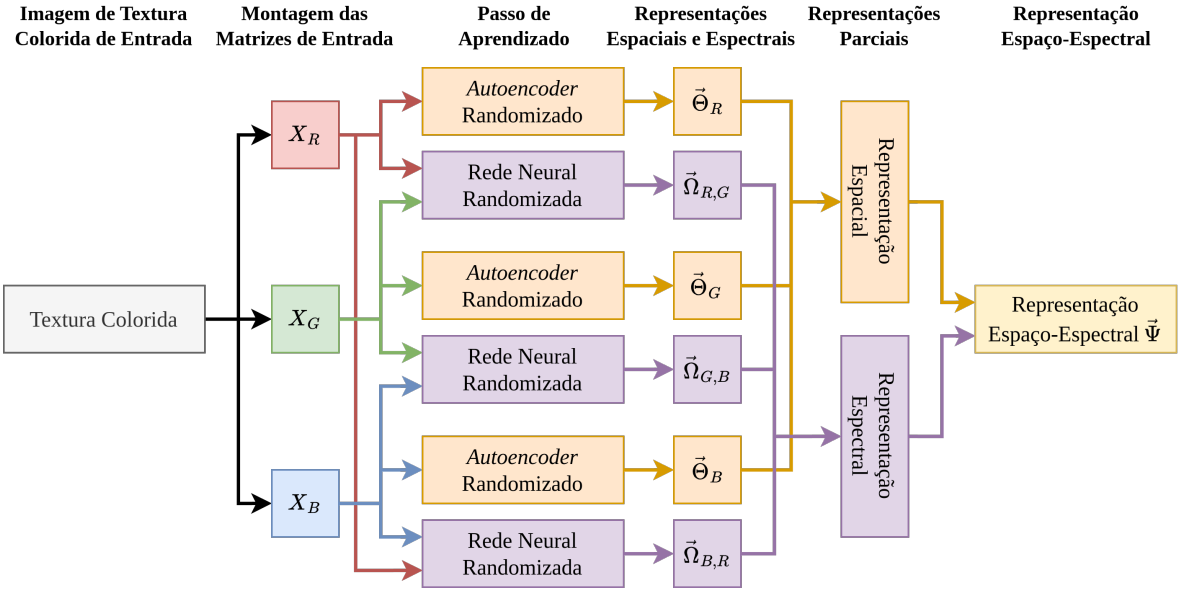
$$\vec{\Psi}(Q) = \Big[\underbrace{\vec{\Theta}_R(Q), \vec{\Theta}_G(Q), \vec{\Theta}_B(Q)}_{\text{Caracterização Espacial}}, \underbrace{\vec{\Omega}_{R,G}(Q), \vec{\Omega}_{G,B}(Q), \vec{\Omega}_{B,R}(Q)}_{\text{Caracterização Espectral}}\Big] \in \mathbb{R}^{54\cdot(Q+1)}\,. \tag{17}$$

Por fim, embora a Figura 16 e Figura 17 detalhem as funcionalidades de seus módulos, tem-se que a Figura 18 apresenta um diagrama de alto nível sumarizando os principais passos para a obtenção da representação de textura espaço-espectral proposta.

Análise de Parâmetros. Neste método, a configuração experimental segue as utilizadas nos métodos apresentados nas subseções 4.1.1 e 4.1.2. Portanto, utiliza-se a acurácia como métrica de desempenho obtida empregando-se o classificador *Linear Discriminant Analysis* com validação cruzada *Leave-One-Out*. Os conjuntos de dados de textura colorida adotados compreendem: Outex (Ojala et al., 2002), USPtex (Backes; Casanova; Bruno, 2012) e MBT (Abdelmounaime; Dong-Chen, 2013), os quais estão detalhados no Apêndice A.

Com relação à análise da representação proposta, essa apresenta-se como um descritor simples que contém apenas um único parâmetro a ser calibrado. Nesse contexto, a Tabela 5 apresenta o comportamento do desempenho da representação à medida que a dimensão da projeção aleatória, Q, é alterada. Ao se analisar o desempenho médio, observa-se que a representação apresenta crescimento contínuo até $Q = 5$, sofrendo uma ligeira queda de 0,1% em $Q = 7$. Em seguida, o desempenho médio retoma seu crescimento, até apresentar uma

Figura 18 – Ilustração do diagrama de alto nível que sumariza os principais passos da abordagem proposta de representação de textura espaço-espectral.



Fonte: Adaptada de Fares & Ribas (2024a).

queda mais expressiva de 0,7% em $Q = 15$. De certa forma, a acurácia média aumenta à medida que Q aumenta, com certas quedas. No entanto, quando Q eleva-se muito, ocorre uma queda mais brusca em comparação com a queda anterior, exibindo certa estabilização da representação e um *trade-off* entre poder descritivo e número de atributos aprendidos.

Além disso, analisando cada conjunto de dados, na Outex, a representação espaço-espectral obteve bom desempenho, obtendo um resultado máximo de 96,8% com a configuração $\vec{\Psi}(13)$ com 756 atributos. Com relação à USPtex, embora as diversas configurações em geral para $Q \geq 5$ obtiveram pouca variação de acurácia indicando certa estabilidade em seu poder descritivo, obteve-se um desempenho máximo de 99,2% com a configuração $\vec{\Psi}(09)$ composta por 540 atributos. Quanto à MBT, a representação espaço-espectral mostrou-se robusta para a caracterização textural desse conjunto de dados, apresentando a configuração $\vec{\Psi}(13)$ com uma acurácia de 99,8%, quase saturando tal conjunto. Diante disso, a representação proposta sugere ser adequada às diferentes características e aparências texturais nas imagens coloridas, exibindo, assim, a robustez de tal caracterização espaço-espectral.

Por fim, com finalidade de comparar de forma justa o método proposto com outros presentes na literatura, serão selecionados as representações de textura que obtiveram as três maiores acurácias médias segundo a Tabela 5. Dessa forma, as representações selecionadas foram: SST_a ($\vec{\Psi}(13)$), SST_b ($\vec{\Psi}(09)$) e SST_c ($\vec{\Psi}(05)$) cujas acurácias obtidas foram 98,6%, 98,3% e 98,2%, respectivamente.

Análise Qualitativa. A fim de complementar a análise anterior, a seguir apresenta-se uma análise qualitativa de visualizações das reconstruções realizadas pelas componentes espacial

Tabela 5 – Acurácias de classificação (%) para a representação de textura espaço-espectral $\vec{\Psi}(Q)$. Os resultados em **destaque** obtiveram as maiores acurácias médias. ↑ indica que maiores resultados são preferíveis.

Q	# de Atributos	Acurácia (↑)			
		Outex	USPtex	MBT	Média
01	108	93,1	98,3	97,7	96,4
03	216	96,0	98,7	98,9	97,9
05	**324**	**95,9**	**99,1**	**99,5**	**98,2**
07	432	95,7	98,9	99,6	98,1
09	**540**	**96,2**	**99,2**	**99,6**	**98,3**
13	**756**	**96,8**	**99,1**	**99,8**	**98,6**
15	864	94,9	99,1	99,7	97,9

Fonte: Adaptada de Fares & Ribas (2024a).

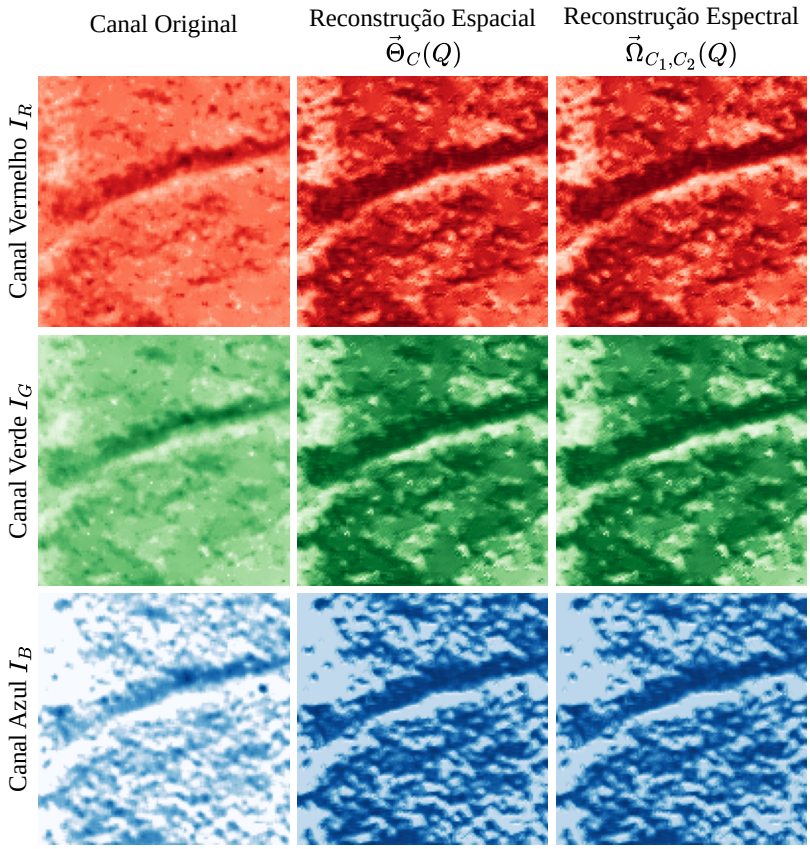
$\vec{\Theta}_C(Q)$, e espectral $\vec{\Omega}_{C_1,C_2}(Q)$. Embora a qualidade das visualizações das reconstruções do conteúdo original por parte das representações de textura não sirvam diretamente como um auxílio para a análise do desempenho dessas representações, a qual foi conduzida quantitativamente, reconhece-se que tal investigação qualitativa é essencial para compreender o que foi aprendido pelos pesos da camada de saída das redes neurais randomizadas utilizadas e do porquê esses pesos compõe a representação (Fares; Ribas, 2024a).

Nesse sentido, a Figura 19 e a Figura 20 apresentam, respectivamente, as visualizações das reconstruções de cada canal e de cada imagem colorida para as componentes espacial e espectral. Para a Figura 19, tem-se que cada linha apresenta um canal de cor distinto da mesma imagem e cada coluna apresenta a imagem original, a visualização da reconstrução espacial e espectral, respectivamente. Por outro lado, para a Figura 20 tem-se que cada linha apresenta uma imagem colorida distinta e cada coluna apresenta a imagem original e as respectivas visualizações das reconstruções realizadas pelos módulos espacial e espectral.

A partir dessas figuras, é possível notar satisfatória similaridade entre as visualizações das reconstruções e seus respectivos conteúdos originais. Dessa forma, isso indica que os pesos aprendidos da camada de saída das redes neurais randomizadas utilizadas capturaram as características essenciais sobre o conteúdo textural intra- e inter-canal. Assim, esses pesos foram capazes, portanto, de mapear com fidelidade as projeções aleatórias das janelas (ou trechos) de entrada de volta para o conteúdo do trecho original (módulo espacial), ou de volta para o conteúdo do trecho do canal adjacente (módulo espectral). Portanto, tal análise exibiu não somente a capacidade do método proposto em capturar tais informações, mas também sua robustez em relação às características das diferentes imagens de texturas coloridas exibidas durante a condução dessa análise qualitativa (Fares; Ribas, 2024a).

Contribuições e Limitações. O método proposto possui as seguintes contribuições. Em

Figura 19 – Visualizações das reconstruções dos canais de cores de uma mesma imagem por parte das componentes espacial e espectral. Em cada linha constam o canal original, a reconstrução espacial e a reconstrução espectral, respectivamente.

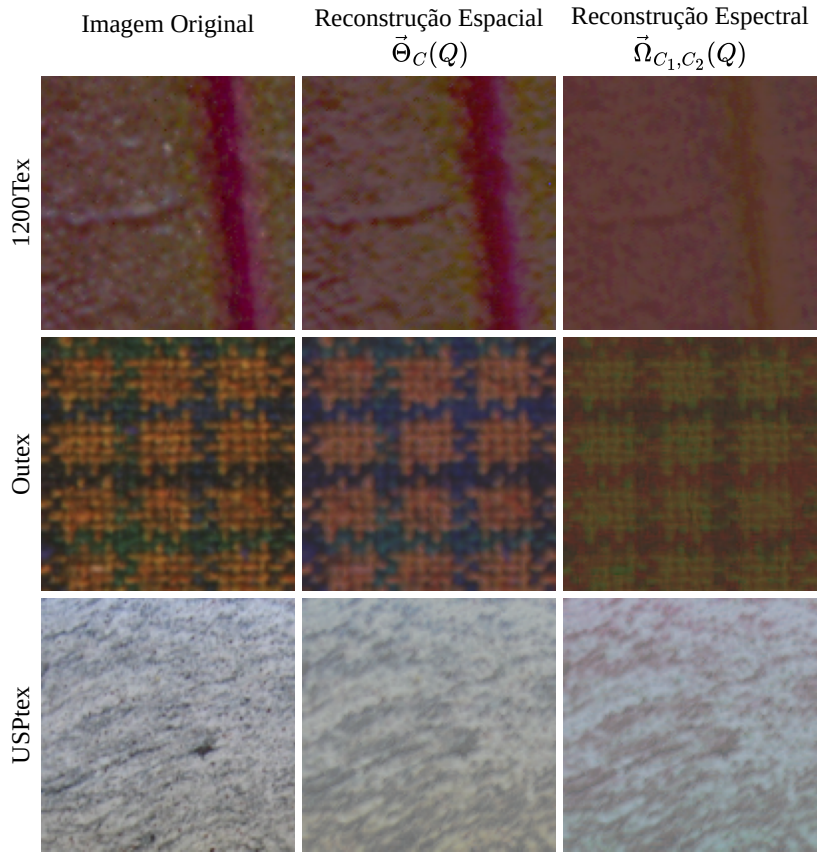


Fonte: Adaptada de Fares & Ribas (2024a).

primeiro lugar, contribui com uma técnica de representação simples e eficaz, possuindo apenas um único parâmetro a ser calibrado facilitando, portanto, a adequação em outras tarefas de reconhecimento de padrões e visão computacional. Em segundo lugar, os módulos espacial e espectral propostos visam aprender características essenciais da imagem de textura colorida e foram substanciados pelos resultados apresentados, demonstrando, assim, que o emprego de *autoencoders* (i.e., redes que aprendem a partir dos próprios dados) no módulo espacial e o cruzamento de informações no módulo espectral foram extremamente importantes para a construção de uma representação de textura colorida robusta e discriminativa.

Entretanto, o método apresenta as seguintes limitações. Primeiramente, o método possui, relativamente, uma restrição no conjunto de valores que o hiperparâmetro da representação de textura pode assumir, devido à elevada constante que a multiplica para o cálculo do número de atributos que compõe a representação. Como segundo ponto, tem-se que a arquitetura da representação proposta está limitada especificamente aos espaços de cores com apenas três bandas, sendo essa analisada efetivamente no espaço de cor RGB. Nesse sentido, trabalhos futuros podem abordar extensões, permitindo que funcionem de forma eficaz em

Figura 20 – Visualizações das reconstruções das imagens coloridas por parte das componentes espacial e espectral. Em cada linha constam a imagem original, a reconstrução espacial e a reconstrução espectral, respectivamente.



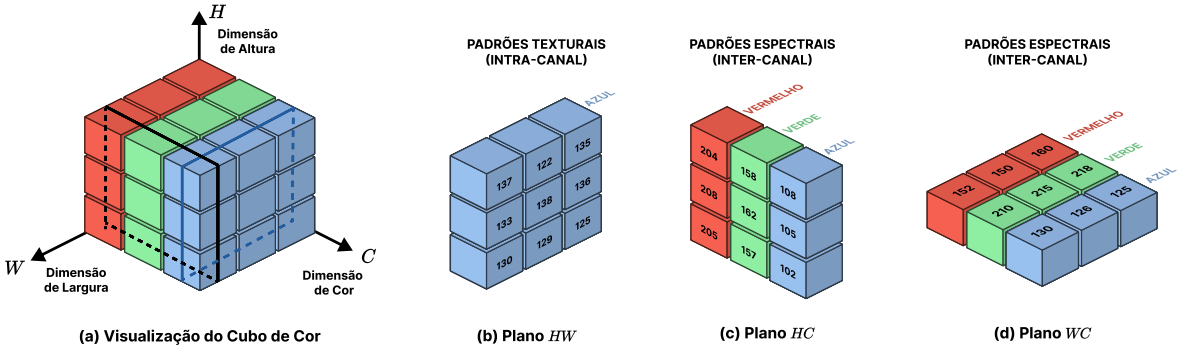
Fonte: Adaptada de Fares & Ribas (2024a).

imagens com mais bandas (*e.g.*, hiperespectrais).

4.2.2 Aprendizado de Representações Volumétrica com Autoencoders Randomizados

Esta subseção dedica-se à apresentação do método de representação de texturas coloridas usando cubos volumétricos proposto em Fares & Ribas (2025b). De modo geral, os métodos de textura colorida existentes aplicam técnicas em nível de cinza em cada canal e agregam as representações obtidas, ou esses não são capazes de capturar eficientemente as informações espaço-espectrais da imagem de textura colorida (Fares; Ribas, 2025b). Nesse contexto, a técnica proposta visa construir representações de textura colorida a partir do esticamento dos pesos dos decodificadores que aprenderam a reconstruir cubos volumétricos de cor, que contêm não somente a informação textural intra-canal, mas também a informação inter-canal. Portanto, a técnica foge de uma abordagem integrativa, permitindo uma análise simultânea de toda a informação presente na imagem colorida, e o que permite tal feito é a extração do conteúdo da imagem colorida a partir da varredura de cubos de cor sobre toda a

Figura 21 – Ilustração do cubo de cor volumétrico utilizado para a obtenção da informação espaço-espectral local da vizinhança para um dado pixel.



Fonte: Adaptada de Fares & Ribas (2025b).

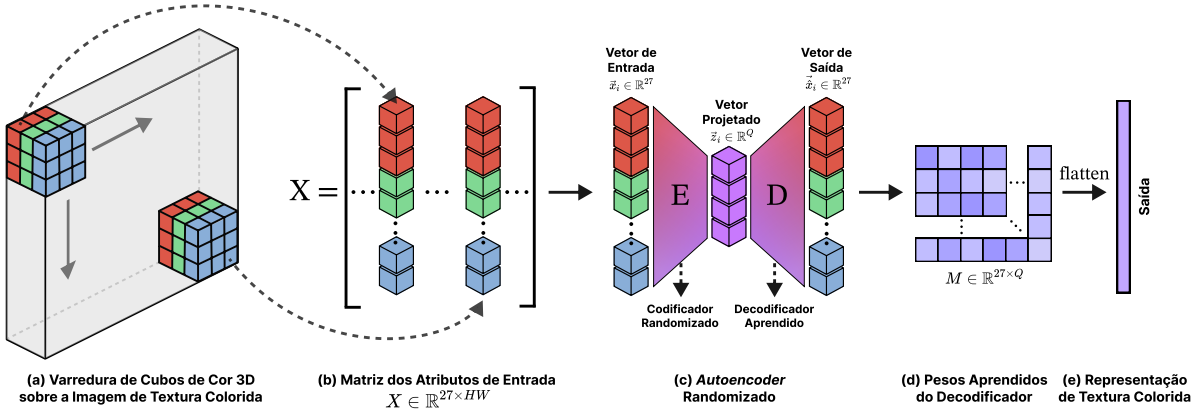
imagem seguido de sua codificação por um único *autoencoder* randomizado.

Construção da Representação. Seja $I \in \mathbb{R}^{3\times H\times W}$ uma imagem colorida. Para cada pixel k da imagem I, é centralizado um cubo $C^{(k)} \in \mathbb{R}^{3\times 3\times 3}$ que agrega simultaneamente toda informação textural (intra-canal) e espectral (inter-canal) presente na vizinhança daquele pixel. Em particular, note que o pixel central do cubo de cor é sempre posicionado sobre os pixels do canal verde para que sempre seja possível capturar a informação de todos os três canais simultaneamente. Nesse sentido, com a finalidade de introduzir uma visualização do cubo de cor volumétrico e como esse é capaz de capturar tanto informação espacial quanto espectral, tem-se que a Figura 21 apresenta a visualização desse cubo de cor. Logo, desde que o procedimento de centralização é realizado sobre todos os pixels do canal verde, obtém-se o conjunto $\mathcal{C} = \{C^{(1)}, C^{(2)}, \ldots, C^{(HW)}\} \subset \mathbb{R}^{3\times 3\times 3}$, o qual agrega a informação espaço-espectral local da imagem integralmente. A partir desse conjunto constrói-se a matriz de atributos de entrada X por meio do esticamento das intensidades dos pixels presentes em cada cubo volumétrico seguido de sua justaposição horizontal (Fares; Ribas, 2025b). Desse modo, a matriz de atributos de entrada é expressa da seguinte forma:

$$X = \left[\text{flatten}(C^{(1)}), \text{flatten}(C^{(2)}), \ldots, \text{flatten}(C^{(HW)})\right] \in \mathbb{R}^{27\times HW}. \tag{18}$$

Logo, do mesmo modo como no conjunto $\mathcal{C}$, tem-se que a matriz de atributos de entrada X carrega a informação espaço-espectral local para toda a imagem, e tal matriz é utilizada para alimentar o *autoencoder* randomizado, o qual é responsável por projetar aleatoriamente e de maneira não linear tais informações espaço-espectrais em outro espaço dimensional e reconstruí-las de volta ao espaço original. Nesse contexto, no processo de decodificação, se o *autoencoder* é capaz de reconstruir simultaneamente tais informações espaço-espectrais com certa fidelidade, o qual será visto em breve, então tem-se que os pesos do decodificador carregam informações úteis e discriminativas sobre os padrões texturais intra- e inter-canal da imagem de textura colorida de entrada, servindo, portanto, como uma fonte útil para a representação de imagem colorida (Fares; Ribas, 2025b).

Figura 22 – Visão geral da metodologia proposta para aprendizado de representações de texturas coloridas por cubos de cor com *autoencoders* randomizados.



Fonte: Adaptada de Fares & Ribas (2025b).

Dessa forma, para a obtenção dos pesos do decodificador, primeiramente, obtém-se a matriz de pesos aleatórios $U = \mathsf{LCG}(Q, p+1)$, em que $p = 27$, representando a quantidade de pixels presentes no cubo de cor volumétrico de dimensões $3 \times 3 \times 3$, o qual é o utilizado na técnica apresentada. Em seguida, os pesos do decodificador é calculado da seguinte forma:

$$M = \mathsf{RAE}(X; U, \lambda_{\mathsf{VCTex}}) \in \mathbb{R}^{27\times(Q+1)}\,, \tag{19}$$

em que $\lambda_{\mathsf{VCTex}} = 10^3$ é o valor da regularização de Tikhonov utilizado no método proposto. Isso posto, a representação parcial do método proposto é obtida a partir do esticamento da matriz dos pesos do decodificador, sendo expressa por: $\vec{\Theta}(Q) = \mathsf{flatten}(M) \in \mathbb{R}^{27\cdot(Q+1)}$. Por fim, da mesma forma como na abordagem proposta na Subseção 4.1.2, e nos métodos da literatura (Sá Junior; Backes, 2016; Ribas et al., 2020), tem-se que é interessante e vantajoso realizar a agregação dos pesos aprendidos a partir da projeção dos dados de entrada em distintas dimensões da camada escondida. Portanto, a representação final é formulada por:

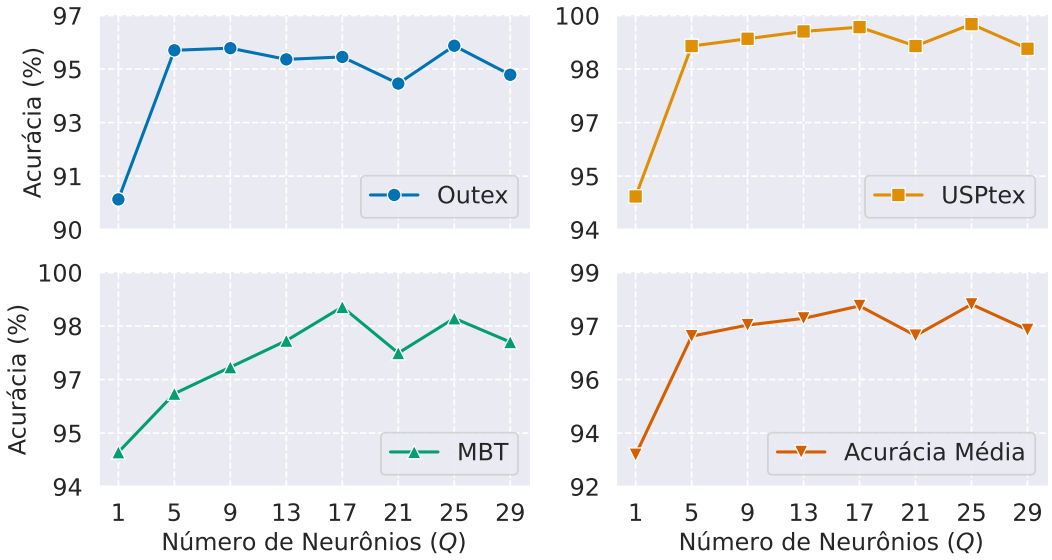
$$\vec{\Omega}(Q_1, Q_2, \ldots, Q_N) = \left[\vec{\Theta}(Q_1), \vec{\Theta}(Q_2), \ldots, \vec{\Theta}(Q_N)\right] \in \mathbb{R}^m, \tag{20}$$

em que $m = 27 \cdot \sum_{i=1}^{N}(Q_i + 1)$ é a dimensão da representação final proposta.

Análise de Parâmetros. O desempenho em tarefas de reconhecimento de texturas utilizando a representação da técnica apresentada será analisado seguindo a mesma configuração experimental do método apresentado na Subseção 4.2.1. Nesse sentido, a técnica será analisada em três conjuntos de dados de imagens coloridas, sendo eles: Outex (Ojala et al., 2002), USPtex (Backes; Casanova; Bruno, 2012) e MBT (Abdelmounaime; Dong-Chen, 2013). Ademais, a métrica de desempenho utilizada é a acurácia, sendo essa obtida por meio do classificador *Linear Discriminant Analysis* com a técnica de validação cruzada *Leave-One-Out*.

Primeiramente, será analisado o desempenho da representação parcial $\vec{\Theta}(Q)$ à medida que é alterada a dimensão da camada escondida, representada pelo parâmetro Q. Em

Figura 23 – Comportamento da acurácia da representação parcial $\vec{\Theta}(Q)$ à medida que a dimensão da camada escondida do *autoencoder* randomizado é alterado.



Fonte: Adaptada de Fares & Ribas (2025b).

particular, tal parâmetro será analisado dentro do seguinte espaço de parâmetros $\mathcal{Q}_{\text{VCTex}} = \{1, 5, 9, 13, 17, 21, 25, 29\}$. Assim, do mesmo modo que em métodos apresentados anteriormente, esse espaço de parâmetros é escolhido para que não haja o surgimento de representações com grande números de atributos o que é indesejável. Dessa forma, durante a análise da representação parcial, essa terá um máximo de 810 atributos, a qual é obtida quando $Q = 29$. Isso posto, a Figura 23 apresenta o comportamento do desempenho da representação parcial $\vec{\Theta}(Q)$ para todos os conjuntos de dados analisados juntamente com o desempenho médio. A partir dela, é possível notar que há um crescimento contínuo do desempenho nos conjuntos USPtex e MBT entre $Q = 1$ e $Q = 17$, refletindo também no desempenho médio. Por outro lado, na Outex, tem-se um crescimento em $Q = 5$, porém até $Q = 17$ há, no geral, uma queda de desempenho. No entanto, analisando o desempenho de todos os conjuntos de dados, juntamente com o desempenho médio, no restante do espaço de parâmetros, i.e., $Q > 17$, tem-se que, embora ocorra uma flutuação no desempenho das representações, a técnica resultante com $Q = 25$ destacou-se como a de maior desempenho médio.

Em seguida, uma segunda análise é realizada, porém agora sobre a representação final $\vec{\Omega}(Q_1, Q_2, \ldots, Q_N)$. Em particular, serão analisadas apenas as representações finais resultantes a partir da agregação de duas representações parciais distintas cujo tamanho da dimensão da camada escondida da primeira representação seja estritamente menor que o da segunda representação. A Tabela 6 apresenta as acurácias de classificação para todas as combinações resultantes do espaço de parâmetros definido anteriormente. Em particular, três configurações $\vec{\Omega}(05, 17)$, $\vec{\Omega}(01, 17)$ e $\vec{\Omega}(17, 29)$ emergem como aquelas que obtiveram os maiores desempenhos médios. Além disso, note-se que a representação parcial $\vec{\Theta}(17)$ está presente em tais

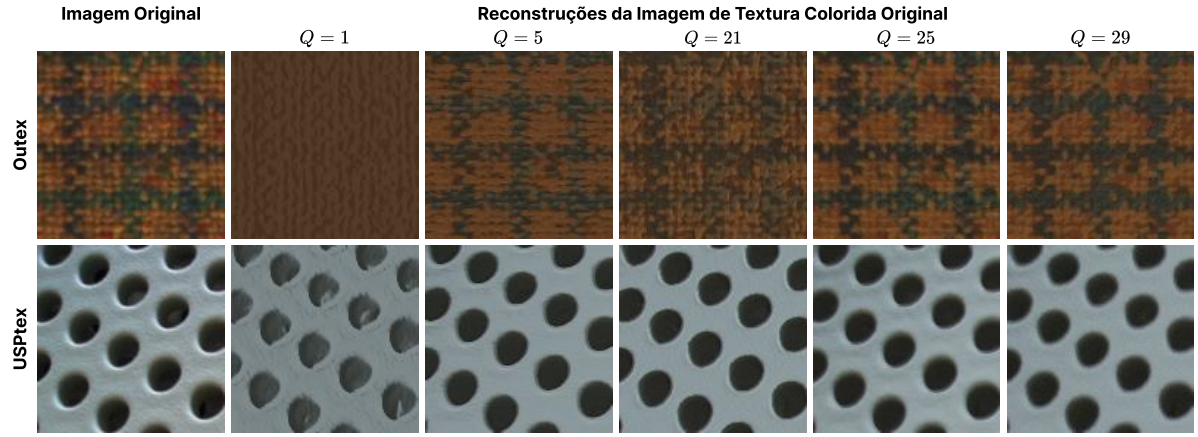
Tabela 6 – Acurácias de classificação (%) para a representação de textura colorida $\vec{\Omega}(Q_1, Q_2)$. Os resultados em **destaque** obtiveram as maiores acurácias médias. ↑ indica que maiores resultados são preferíveis.

(Q_1, Q_2)	# de Atributos	Acurácia (↑)			
		USPtex	Outex	MBT	Média
(01, 05)	216	99,0	95,7	97,2	97,3
(01, 09)	324	99,5	96,1	97,9	97,8
(01, 13)	432	99,5	95,4	98,2	97,7
(01, 17)	**540**	**99,6**	**95,8**	**98,9**	**98,1**
(01, 21)	648	99,2	94,3	97,7	97,1
(01, 25)	756	99,7	95,9	98,7	98,1
(01, 29)	864	99,1	94,9	98,3	97,4
(05, 09)	432	99,7	96,0	97,3	97,7
(05, 13)	540	99,7	95,2	97,8	97,6
(05, 17)	**648**	**99,6**	**96,0**	**99,1**	**98,2**
(05, 21)	756	99,6	95,2	97,9	97,6
(05, 25)	864	99,6	95,2	98,7	97,8
(05, 29)	972	99,4	96,0	98,3	97,9
(09, 13)	648	99,7	96,1	97,9	97,9
(09, 17)	756	99,7	94,9	98,9	97,8
(09, 21)	864	99,5	94,9	98,3	97,6
(09, 25)	972	99,7	94,9	98,7	97,8
(09, 29)	1080	99,6	95,7	98,4	97,9
(13, 17)	864	99,7	94,9	98,9	97,8
(13, 21)	972	99,7	95,0	98,6	97,8
(13, 25)	1080	99,7	93,8	98,7	97,4
(13, 29)	1188	99,7	95,7	98,4	97,9
(17, 21)	1080	99,7	95,1	99,2	98,0
(17, 25)	1188	99,7	93,2	98,9	97,3
(17, 29)	**1296**	**99,7**	**95,4**	**99,2**	**98,1**
(21, 25)	1296	99,7	94,6	98,9	97,8
(21, 29)	1404	99,4	94,8	98,6	97,6
(25, 29)	1512	99,7	95,2	98,9	98,0

Fonte: Adaptada de Fares & Ribas (2025b).

configurações. Dessa forma, isso mostra que a representação parcial $\vec{\Theta}(17)$ caracteriza eficientemente os padrões texturais espaço-espectrais das imagens coloridas e, isso está substanciado pelo bom desempenho médio como exibe a Figura 23. Nesse sentido, agregando outras boas representações parciais à $\vec{\Theta}(17)$ permitiu construir uma representação final ainda mais discriminativa, tais como as citadas anteriormente. No entanto, deve-se notar que nem sempre a agregação com $\vec{\Theta}(17)$ produziu melhor desempenho como, por exemplo, o caso da $\vec{\Omega}(9, 17)$ que obteve 97,8%, enquanto $\vec{\Theta}(17)$ obteve 98,0% no desempenho médio. Enfim, para questões de comparação com os métodos da literatura posteriormente, tem-se que as representações finais

Figura 24 – Visualização de duas imagens originais provenientes da Outex e USPtex e suas respectivas reconstruções a partir de distintas representações parciais $\vec{\Theta}(Q)$.



Fonte: Adaptada de Fares & Ribas (2025b).

$\vec{\Omega}(Q_1, Q_2)$ selecionadas são aquelas que obtiveram os maiores desempenhos médios, sendo elas: VCTex$_a$ ($\vec{\Omega}(05, 17)$), VCTex$_b$ ($\vec{\Omega}(01, 17)$) e VCTex$_c$ ($\vec{\Omega}(17, 29)$)[4].

Análise Qualitativa. A Figura 24 apresenta uma visualização das imagens originais provenientes dos conjuntos de dados da Outex e USPtex e suas respectivas reconstruções obtidas a partir do decodificador do *autoencoder* randomizado utilizado para a obtenção da representação parcial $\vec{\Theta}(Q)$. Em particular, a visualização apresenta a reconstrução para distintas dimensões da camada escondida, i.e., da dimensão do espaço em que as intensidades dos pixels presentes em cada cubo de cor são projetados. Dessa forma, é possível notar que, de modo geral, dimensões mais baixas tendem a não capturar toda a informação espaço-espectral presente na imagem de entrada, tal que isso pode ser visualizado, por exemplo, pela reconstrução imprecisa da representação parcial $\vec{\Theta}(01)$. Esse efeito é esperado pelo fato da compressão agressiva que incide sobre as intensidades dos pixels do cubo de cor, as quais são transformadas em um único (i.e., $Q = 1$) valor real. Por outro lado, por uma inspeção visual, é possível notar que, de modo geral, observa-se uma melhora na qualidade da reconstrução, por exemplo, pela satisfatória imagem reconstruída a partir da representação parcial $\vec{\Theta}(25)$.

Nesse contexto, de maneira análoga ao método apresentado na Subseção 4.2.1, tem-se que a notável qualidade das reconstruções, principalmente aquelas que possuem uma dimensão da camada escondida de moderada à elevada, indicam que os pesos do decodificador aprendidos, para a imagem de textura de entrada, estão sendo capazes de mapear com precisão as projeções aleatórias das intensidades dos cubos de cor nas próprias intensidades originais, i.e., mapear com precisão o conteúdo do espaço projetado de volta ao espaço de entrada. Isso sugere que os pesos do decodificador carregam informações essenciais sobre a imagem de

[4] Deve-se notar que a representação $\vec{\Omega}(01, 25)$ também obteve o desempenho média de 98,1%. No entanto, como $\vec{\Omega}(01, 17)$ foi selecionado e possui uma quantidade de atributos menor que $\vec{\Omega}(01, 25)$, então foi dado preferência à próxima representação, sendo essa a $\vec{\Omega}(17, 29)$.

textura de entrada, sendo substanciado pela análise quantitativa apresentada (Fares; Ribas, 2025b). Logo, esta análise completa a justificativa de que os pesos aprendidos do decodificador são úteis e eficazes como forma de representação de imagens de textura.

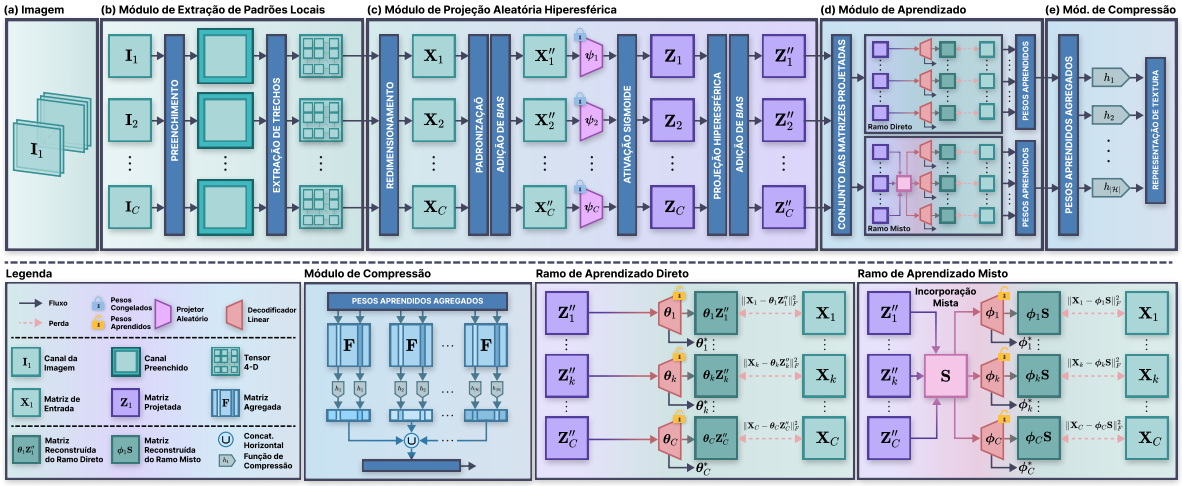
Contribuições e Limitações. A principal contribuição do método proposto é a maneira como a matriz de entrada da rede neural randomizada é construída, de forma a capturar única e simultaneamente o conteúdo espaço-espectral da imagem de entrada. Como visto anteriormente, isso foi possível de ser realizado a partir da varredura de cubos volumétricos de dimensão $3 \times 3 \times 3$ sobre a imagem de entrada. Nesse contexto, a partir do treinamento da rede com a finalidade de aprender a reconstrução de tais cubos volumétricos, foi possível, portanto, a obtenção de representações de texturas coloridas espaço-espectrais discriminativas.

No entanto, como analisado em Fares & Ribas (2025b) uma limitação do método proposto é sua sensibilidade à presença de ruído na imagem. Embora com um nível leve de ruído o método proposto ainda é capaz de caracterizar eficientemente o conteúdo textural da imagem, percebeu-se que com alta carga de ruído outros métodos baseados em redes neurais randomizadas, tal como o SSR (Ribas et al., 2024a), foram capazes de ser mais robustos à presença de ruído. Acredita-se que isso se deve à forma como a imagem de textura é modelada como a matriz de entrada, pois o método proposto usa diretamente os valores dos pixels, os quais são sujeitos a sofrer com a carga ruidosa, enquanto métodos como o SSR podem ser mais robustos devido a sua modelagem em redes complexas.

4.2.3 Aprendizado de Representações com Rede Neural Randomizada de Incorporação Mista Hiperesférica

Apresenta-se nesta subseção a técnica de reconhecimento de texturas usando uma nova formulação de rede neural randomizada proposta em Fares & Ribas (2025a). Atualmente, muitos dos métodos propostos para aprendizado de representações de textura focaram em melhorar a modelagem das matrizes de entrada e saída da rede neural randomizada com a finalidade de aprender pesos mais robustos, e pouco focou-se em melhorar a arquitetura existente da rede neural randomizada para aprender melhores representações de textura (Fares; Ribas, 2025a). Nesse contexto, propôs-se uma arquitetura de rede neural randomizada com uma nova formulação do problema de otimização original, porém voltada para aprender robustas representações de textura. Para isso, a técnica proposta utiliza-se de incorporações aleatórias hiperesféricas melhorando o treinamento e a estabilidade numérica, seguida de um novo módulo de aprendizado munido com dois submódulos especializados que aprendem em conjunto as relações de textura intra- e inter-canal da imagem de entrada. A partir disso, os pesos aprendidos a partir desse novo módulo de aprendizado sofrem uma operação de fusão e são comprimidos (i.e., sumarizados) em uma representação de textura a partir da utilização de distintas medidas estatísticas.

Figura 25 – Ilustração da metodologia de aprendizado de representações de textura denominada por Mixer proposta em Fares & Ribas (2025a).



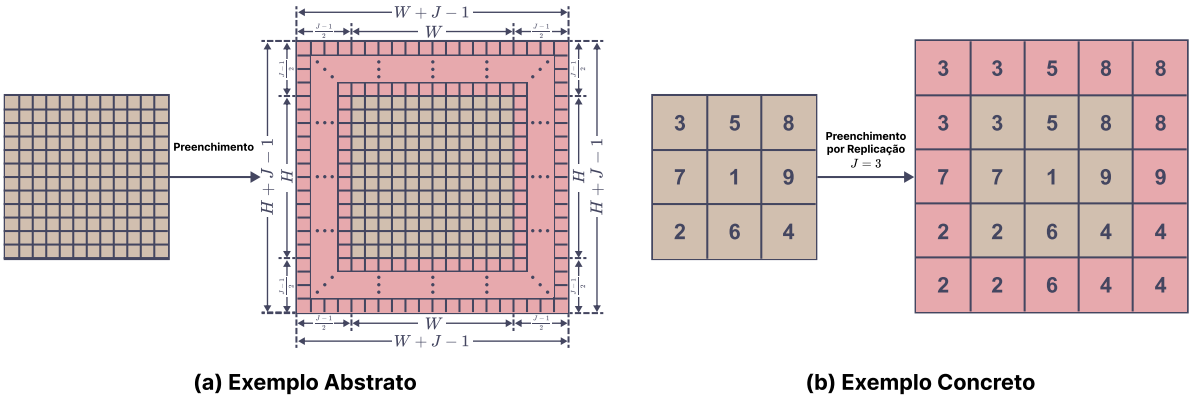
Fonte: Adaptada de Fares & Ribas (2025a).

Construção da Representação. Seja $I \in \mathbb{R}^{C \times H \times W}$ uma imagem com uma quantidade arbitrária de canais. Essa imagem será utilizada como entrada do *pipeline* da técnica de aprendizado de representações proposta, ilustrada na Figura 25, e, a qual consiste de quatro módulos principais, sendo eles: Módulo de Extração de Padrões Locais (LPE, do inglês *Local Pattern Extractor Module*); Módulo de Projetor Aleatório Hiperesférico (HPR, do inglês *Hypersperical Random Projector Module*); Módulo de Aprendizado (LM, do inglês *Learning Module*) e Módulo de Compressão (CM, do inglês *Compression Module*). Cada módulo possui um papel específico e essencial, com o objetivo de obter representações de texturas generalizáveis e discriminativas. Nesse contexto, a seguir, cada módulo será apresentado detalhadamente.

Módulo de Extração de Padrões Locais. Este módulo, ilustrado na Figura 25(b), é responsável pela extração dos conteúdos texturais brutos presentes em todos os canais da imagem de entrada. Nesse sentido, o módulo captura toda a informação textural preliminar essencial que será utilizada para alimentar o restante da rede, particularmente, alimentar o módulo de projeção aleatória hiperesférica, o qual será discutido posteriormente (Fares; Ribas, 2025a).

Dessa forma, para a extração deste conteúdo textural, o módulo realiza densamente a varredura de uma janela de dimensões $J \times J$ sobre cada canal da imagem independentemente, sendo essa varredura representada pela centralização de tais janelas em todos os pixels da dimensão espacial. Nesse contexto, portanto, surge o primeiro hiperparâmetro da técnica proposta que é J, o qual representa o tamanho do lado da janela de varredura. De forma geral, é conhecido na literatura e também intuitivo que valores pequenos para J, tais como 3, 5 e 7, concentram-se principalmente no conteúdo micro-textural presente na imagem, enquanto valores, tais como 9, 11 e 13 etc., buscam analisar regiões com maior granulosidade em relação ao conteúdo textural. Em particular, a técnica proposta foca em analisar conteúdo

Figura 26 – Ilustração da técnica de preenchimento de uma imagem representada por uma matriz. (a) Exemplo abstrato. (b) Exemplo concreto com a técnica de replicação.



Fonte: Adaptada de Fares & Ribas (2025a).

microtextural, e, portanto, esse módulo tem como valor padrão $J = 3$.

Entretanto, deve-se notar que é inviável a centralização de tais janelas próximos ou na borda da imagem em questão. Nesse sentido, antes da varredura das janelas para a extração do conteúdo textural, a imagem é submetida a um processo de preenchimento de pixels para que seja possível obter todas as $C \cdot H \cdot W$ janelas possíveis (Fares; Ribas, 2025a). Dessa forma, embora o método proposto empregue $J = 3$, tem-se que de forma geral para qualquer valor de J e sendo esse ímpar, então $\frac{J-1}{2}$ pixels são adicionados em cada lado da imagem. Em particular, o processo de preenchimento emprega a técnica de replicação, de modo que os valores dos pixels da borda são rebatidos no preenchimento. Tal processo é ilustrado de forma abstrata e concreta na Figura 26. Nesse contexto, tais etapas de preenchimento e extração de janelas (i.e., trechos) da imagem original são representadas pelas seguintes funções:

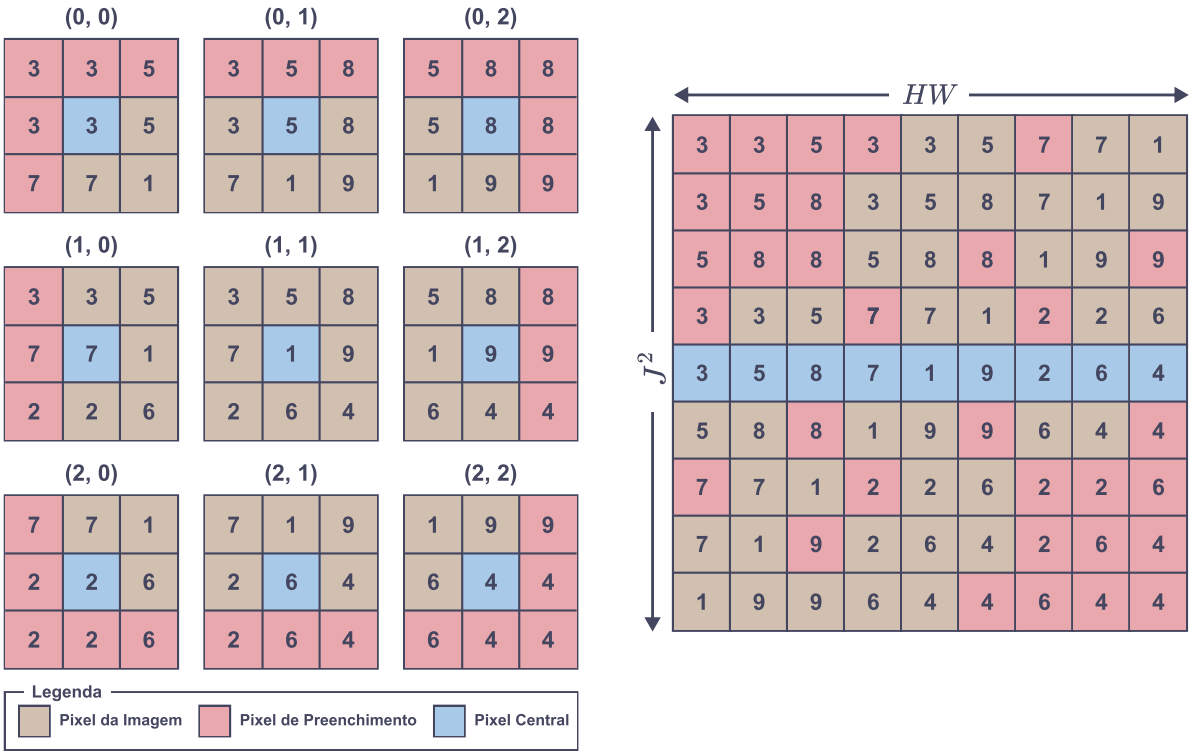
$$\text{Pad}_J(I) \in \mathbb{R}^{C\times(H+J-1)\times(W+J-1)} \tag{21}$$

$$\text{LPE}(\text{Pad}_J(I)) \in \mathbb{R}^{C\times H\times W\times J\times J}\,. \tag{22}$$

Basicamente, a função $\text{Pad}_J : \mathbb{R}^{C\times H\times W} \to \mathbb{R}^{C\times(H+J-1)\times(W+J-1)}$, parametrizada por J, é responsável por receber a imagem de entrada $I \in \mathbb{R}^{C\times H\times W}$ e realizar o processo de preenchimento com a técnica de replicação em cada um dos C canais da imagem independentemente, com tal processo ilustrado na Figura 26(a) para um único canal. Nesse sentido, como $\frac{J-1}{2}$ pixels são adicionados em cada lado da imagem, então $2 \cdot \frac{J-1}{2} = J - 1$ pixels são adicionados em ambas dimensões espaciais, tendo, portanto, tal função uma saída de dimensão de $C \times (H + J - 1) \times (W + J - 1)$. Essa saída é usada para alimentar a função $\text{LPE} : \mathbb{R}^{C\times(H+J-1)\times(W+J-1)} \to \mathbb{R}^{C\times H\times W\times J\times J}$ que tem por finalidade realizar a extração densa de múltiplos trechos de dimensão $J \times J$ sobre a imagem, resultando em um tensor 5-dimensional, que representa todos os trechos $J \times J$ centralizados em todos os pixels da dimensão espacial $H \times W$ para cada um dos C canais da imagem (Fares; Ribas, 2025a).

Por fim, como exemplo, considere a matriz 3×3 da Figura 26(b) como a imagem

Figura 27 – Ilustração em (a) da saída do módulo de extração de padrões locais para uma imagem de exemplo contendo um único canal, e (b) a etapa de redimensionamento realizada pelo módulo de projeção aleatória hiperesférica.



(a) $L = \mathrm{LPE}(\mathrm{Pad}_J(I)) \in \mathbb{R}^{1\times 3\times 3\times 3\times 3}$ **(b)** $X = \mathrm{Reshape}(L) \in \mathbb{R}^{1\times 9\times 9}$

Fonte: Adaptada de Fares & Ribas (2025a).

de entrada de um único canal $I \in \mathbb{R}^{1\times 3\times 3}$. Dessa forma, tem-se que o resultado da função $\mathrm{Pad}_3(I) \in \mathbb{R}^{1\times 5\times 5}$ é representada pela matriz 5×5 da mesma figura. Nesse caso ilustrativo, é possível notar claramente como o preenchimento resulta-se do rebatimento dos valores dos pixels da borda da imagem, o qual é característico da técnica de preenchimento por replicação. Em seguida, tem-se na Figura 27(a) a saída da função $\mathrm{LPE}(\cdot)$ que exibe o conteúdo presente nas distintas janelas de dimensão 3×3 centralizadas em cada um dos nove pixels presentes na imagem original. Logo, esse exemplo ilustrativo busca melhorar o entendimento sobre a ação que este módulo tem sobre a imagem original e seu objetivo.

Módulo do Projetor Aleatório Hiperesférico. Este módulo, ilustrado na Figura 25(c), tem como responsabilidade receber os múltiplos trechos obtidos, os quais carregam o conteúdo textural bruto, e calcular as representações internas correspondentes denominadas por incorporações aleatórias hiperesféricas, as quais são utilizadas posteriormente pelo módulo de aprendizado para o cálculo dos pesos dos decodificadores lineares (Fares; Ribas, 2025a). Em particular, denomina-se por incorporação aleatória, desde que os trechos são, primeiramente, projetados aleatoriamente de maneira não-linear em um outro espaço Q-dimensional, geralmente maior, com a finalidade de aumentar a separabilidade dos dados como afirmado pelo

Teorema de Cover (Cover, 1965). Em seguida, as incorporações aleatórias parciais são projetadas sobre a superfície de uma hiperesfera, permitindo melhorar o treinamento no módulo de aprendizado e sua estabilidade numérica (Fares; Ribas, 2025a).

Para isso, seja $L = \text{LPE}(\text{Pad}_J(I))$ a saída do módulo de extração de padrões locais (i.e., o módulo anterior). Primeiramente, tem-se que L é um vetor 5-dimensional, e, esse deverá sofrer determinados procedimentos para que seja possível realizar a projeção dos padrões texturais locais. Dessa forma, esse tensor é redimensionado a partir da fusão entre as dimensões que carregam os padrões locais (i.e., as duas últimas dimensões), e a fusão das dimensões espaciais (i.e., a segunda e terceira dimensões), seguida, por fim, da permutação entre as dimensões fundidas resultantes. Nesse contexto, essa etapa de redimensionamento é, simplificadamente, representada pela seguinte função:

$$\text{Reshape} : \mathbb{R}^{C\times H\times W\times J\times J} \to \mathbb{R}^{C\times J^2\times HW}\,, \tag{23}$$

em que $X = \text{Reshape}(L)$ representa o tensor 3-dimensional contendo os padrões locais texturais esticados para cada pixel na dimensão espacial e para cada canal da imagem de entrada. Em particular, tal tensor será visualizado como um vetor de matrizes, representando $X_1, X_2, \ldots, X_C$, em que $X_k \in \mathbb{R}^{J^2\times HW}$ representa a matriz contendo os padrões texturais locais presentes no k-ésimo canal da imagem I. A partir disso, essas matrizes serão utilizadas para a obtenção das incorporações aleatórias parciais, as quais são expressas por:

$$Z_k = \alpha(U_k X_k'') \in [0,1]^{Q\times HW}\,, \tag{24}$$

em que $\alpha(\cdot)$ é a função de ativação sigmoide, $U_k \in \mathbb{R}^{Q\times(J^2+1)}$ denota a matriz dos pesos aleatórios, a qual é responsável por realizar a projeção aleatória da seguinte matriz dos atributos de entrada padronizados adicionada de *bias*,

$$X_k'' = \text{vconcat}(\{-1_{HW}, X_k'\}) \in \mathbb{R}^{(J^2+1)\times HW}\,, \tag{25}$$

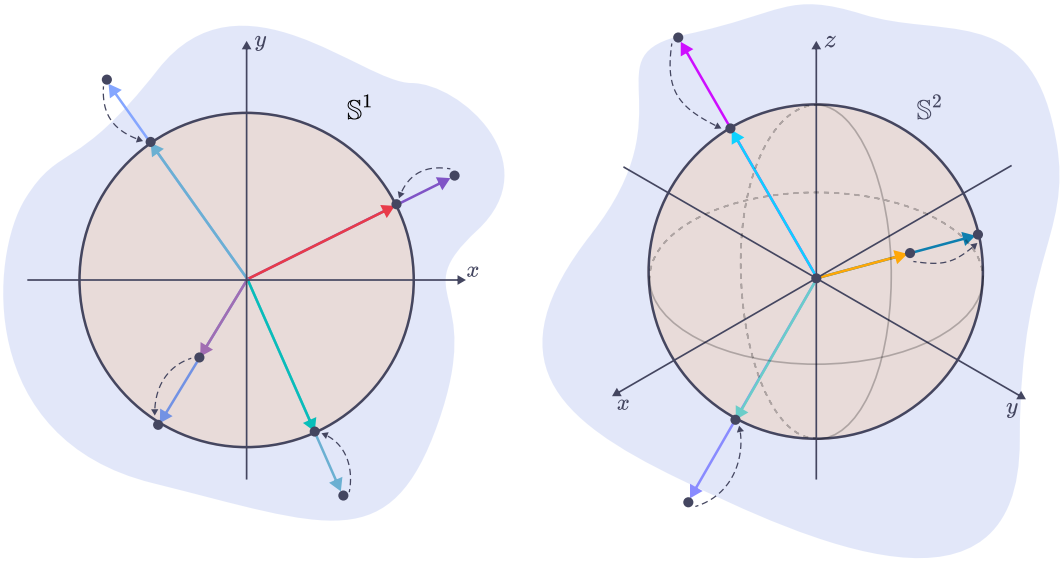
com $\text{vconcat}(\cdot)$ representando a operação de concatenação vertical de matrizes, e X_k' denotando a matriz X_k submetida ao processo de padronização dos atributos de entrada, o qual é expresso da seguinte forma:

$$[X_k']_{ij} = \frac{[X_k]_{ij} - \overline{[X]}_{i,:}}{s_{k,i} + \epsilon}\,, \quad k \in \{1, 2, \ldots, C\}\,, \tag{26}$$

sendo $\overline{[X]}_{i,:} = \frac{1}{HW}\sum_{j=1}^{HW}[X_k]_{ij}$ a média amostral dos valores presentes na i-ésima linha da matriz X_k, $s_{k,i}^2 = \frac{1}{HW-1}\sum_{j=1}^{HW}([X_k]_{i,j} - \overline{[X_k]}_{i,:})^2$ representando a variância amostral, e o valor $\epsilon = 10^{-10}$ sendo utilizado para evitar divisão por zero durante o processo de padronização da matriz X_k (Fares; Ribas, 2025a).

Dessa forma, tem-se que de forma geral a matriz $Z_k = [\vec{z}_1^{(k)}, \vec{z}_2^{(k)}, \ldots, \vec{z}_{HW}^{(k)}] \in \mathbb{R}^{Q\times HW}$ obtida a partir da Equação 24 é composta por todas as incorporações aleatórias

Figura 28 – À esquerda ilustra-se a hiperesfera $\mathbb{S}^1$ representada por uma circunferência, e à direita ilustra-se a hiperesfera $\mathbb{S}^2$ representada pela superfície de uma esfera.



Fonte: Elaborada pelo autor.

parciais Q-dimensionais, em que $\vec{z}_j^{(k)} \in \mathbb{R}^Q$ representa a incorporação aleatória parcial do j-ésimo trecho de padrões texturais locais presentes no k-ésimo canal da imagem I. Em seguida, obtém-se as incorporações aleatórias hiperesféricas correspondentes das incorporações parciais. Para isso, o processo de normalização será aplicado sobre as colunas de cada uma das matrizes projetadas $Z_1, Z_2, \ldots, Z_C$. Em outras palavras, tal processo será expresso por:

$$[Z'_k]_{ij} = \frac{[Z_k]_{ij}}{\max(\|[Z_k]_{:,j}\|_2, \epsilon)}, \quad k \in \{1, 2, \ldots, C\}, \tag{27}$$

em que $\|\cdot\|_2$ denota a norma Euclidiana, e $[Z_k]_{:,j} \in \mathbb{R}^Q$ representa a j-ésima coluna de Z_k. Em particular, deve-se notar que a partir da Equação 27 as incorporações aleatórias parciais são projetadas sobre a hiperesfera $\mathbb{S}^{Q-1} = \{\vec{v} \in \mathbb{R}^Q \mid \|\vec{v}\|_2 = 1\} \subset \mathbb{R}^Q$ (Fares; Ribas, 2025a). Como exemplo, a Figura 28 apresenta exemplos das hiperesferas $\mathbb{S}^1$ e $\mathbb{S}^2$, as quais correspondem, respectivamente, a uma circunferência em $\mathbb{R}^2$, e a uma superfície esférica em $\mathbb{R}^3$, e o processo de projeção de vetores quaisquer sobre a hiperesfera. No passo seguinte, o valor do *bias* também será adicionado as matrizes projetadas normalizadas $Z'_1, Z'_2, \ldots, Z'_C$, de modo que esse processo é expresso por:

$$Z''_k = \text{vconcat}(\{-1_{HW}, Z'_k\}) \in \mathbb{R}^{(Q+1)\times HW}. \tag{28}$$

Por fim, da mesma forma como realizado para o módulo de extração de padrões locais, todo o processo de obtenção da matriz Z''_k correspondendo a projeção aleatória hiperesférica de X_k, será representada por $f_{U_k}(X_k)$, em que f_{U_k} denota o k-ésimo projetor aleatório hiperesférico munido com os pesos aleatórios $U_k \in \mathbb{R}^{Q\times(J^2+1)}$ (Fares; Ribas, 2025a).

Módulo de Aprendizado. Este módulo, ilustrado na Figura 25(d), representa uma peça-chave na técnica proposta, pois é responsável pelo treinamento dos decodificadores lineares cujos pesos serão utilizados posteriormente para comporem a representação de textura final. Nesse contexto, o módulo de aprendizado é composto por dois submódulos ou ramos denominados Direto e Misto, os quais são responsáveis por aprenderem as relações texturais intra- e inter-canal, respectivamente, de tal forma que o módulo misto realiza esse feito por um novo problema de otimização focado no aprendizado de representação de texturas. Dessa forma, a seguir, serão apresentados tais submódulos ou ramos detalhadamente:

- **Ramo Direto**: Este ramo é responsável por aprender as características texturais intra-canal da imagem de textura de entrada. Para isso, esse é treinado com a finalidade de aprender um mapeamento entre o espaço dos atributos projetados (i.e., o espaço contendo as incorporações aleatórias hiperesféricas) e o espaço dos atributos de entrada (i.e., o espaço que carregam as informações texturais brutas) somente a partir do conteúdo textural presente em um dado canal X_k (Fares; Ribas, 2025a). Fazendo isso, este submódulo aprende as relações intrínsecas entre tais espaços, permitindo, portanto, que o módulo aprenda os padrões texturais presentes na imagem.

  Dessa forma, para que tal objetivo seja atingido, o processo de treinamento anteriormente destacado é matematicamente expresso no seguinte problema de otimização:

  Problema de Otimização Convexo do Ramo Direto

  Otimização 2 (Ramo Direto)**:** Seja $g_{M_D}(Z) = M_D Z$ um decodificador linear parametrizado por M_D. Busca-se aprender um decodificador linear $\hat{M}_D \in \mathbb{R}^{J^2 \times (Q+1)}$ para cada canal da imagem a partir da resolução do seguinte problema de minimização:

  $$\min_{M_D^{(k)}} \|X_k - g_{M_D^{(k)}}(f_{U_k}(X_k))\|_F^2 + \lambda_D \|M_D^{(k)}\|_F^2 , \tag{29}$$

  em que $\lambda_D > 0$ é o parâmetro de regularização do ramo direto. Por fim, deve-se notar que este problema de otimização culmina em um *autoencoder* regularizado, de modo que, busca-se aprender parâmetros $\hat{M}_D$ que projete as incorporações aleatórias não-lineares hiperesféricas em $\mathbb{R}^{Q+1}$ de volta ao espaço de entrada original $\mathbb{R}^{J^2}$.

  Em particular, como descrito em Fares & Ribas (2025a) o problema de minimização descrito na Equação 29 é munido de uma solução analítica, de modo que para encontrar um decodificador linear $\hat{M}_D$ adequado, não é necessário nenhum método de retropropagação. Nesse contexto, a solução analítica para tal submódulo ou ramo é dado por:

  $$\hat{M}_D^{(k)} = X_k Z_k''^T (Z_k'' Z_k''^T + \lambda_D I)^{-1} . \tag{30}$$

  Por fim, como este processo de encontrar os pesos do decodificador linear adequado é realizado para cada canal da imagem, este módulo fornecerá como saída um conjunto de

pesos de decodificadores lineares, os quais carregam e incorporam as relações texturais presentes em cada canal da imagem. Dessa forma, tem-se que o seguinte conjunto contendo os pesos dos decodificadores lineares $\mathcal{S}_{\mathcal{D}} = \{\hat{M}_D^{(1)}, \hat{M}_D^{(2)}, \ldots, \hat{M}_D^{(C)}\}$ representa a saída deste submódulo (Fares; Ribas, 2025a).

- **Ramo Misto**: Este ramo tem como atribuição aprender as relações texturais intrínsecas inter-canal. Para isso, diferentemente do ramo direto que utiliza apenas a incorporação aleatória do próprio canal para guiar o aprendizado, este módulo utiliza uma incorporação mista resultante de uma fusão intermediária das incorporações aleatórias de todos os canais da imagem. Nesse sentido, este ramo reconstrói o padrão local textural em uma posição específica de um canal utilizando não só a incorporação aleatória deste padrão textural, mas também as incorporações aleatórias provenientes dos padrões locais texturais na mesma posição específica de todos os canais (Fares; Ribas, 2025a).

  Portanto, para tal feito, esse processo de aprendizado pode ser matematicamente descrito no seguinte problema de otimização:

Problema de Otimização Convexo do Ramo Misto

Otimização 3 (Ramo Misto)**:** Seja $g_{M_S}(Z) = M_S Z$ um decodificador linear parametrizado por M_S. Busca-se aprender um decodificador linear $\hat{M}_S \in \mathbb{R}^{J^2 \times (Q+1)}$ para cada canal da imagem a partir da resolução do seguinte problema de minimização:

$$\min_{M_S^{(k)}} \|X_k - g_{M_S^{(k)}}(S)\|_F^2 + \lambda_S \|M_S^{(k)}\|_F^2 \,, \tag{31}$$

em que $\lambda_S > 0$ é o parâmetro de regularização do ramo misto, e $S \in \mathbb{R}^{(Q+1)\times HW}$ representa a matriz composta das incorporações aleatórias mistas construídas a partir do seguinte processo de fusão intermediária:

$$S = \arg\min_{S'} \sum_{k=1}^{C} \|f_{U_k}(X_k) - S'\|_F^2 = \frac{1}{C} \sum_{k=1}^{C} f_{U_k}(X_k) \,. \tag{32}$$

Assim, da mesma forma que no ramo direto, o problema de otimização representado na Equação 31 admite uma solução analítica, a qual é expressa por:

$$\hat{M}_S^{(k)} = X_k S^T (SS^T + \lambda_S I)^{-1} \,. \tag{33}$$

Por fim, como este processo é realizado para cada canal independentemente, então este submódulo fornecerá como saída o conjunto dos pesos dos decodificadores lineares $\mathcal{S}_S = \{\hat{M}_S^{(1)}, \hat{M}_S^{(2)}, \ldots, \hat{M}_S^{(C)}\}$ obtidos pela nova fórmula de otimização.

Módulo de Compressão. Uma representação para qualquer imagem é, geralmente, representada por um vetor n-dimensional que deve carregar informações essenciais sobre tal imagem.

Nesse contexto, deve-se notar que a saída dos módulos de aprendizado é um conjunto de matrizes $\mathcal{S} = \mathcal{S}_D \cup \mathcal{S}_S \subset \mathbb{R}^{J^2 \times (Q+1)}$, que corresponde aos pesos dos decodificadores lineares aprendidos a partir dos problemas de otimização apresentados. Dessa forma, este módulo, como sugere sua denominação e ilustrado na Figura 25(e), tem como finalidade comprimir (sumarizar) tal conjunto de matrizes em um vetor que carregue informações robustas e discriminativas para que seja possível, posteriormente, o seu uso em tarefas de classificação e reconhecimento de texturas (Fares; Ribas, 2025a).

Para isso, seja $F = \text{vconcat}(\mathcal{S}) \in \mathbb{R}^{|\mathcal{S}|J^2 \times (Q+1)}$ a matriz resultante da concatenação vertical das matrizes representando os pesos dos decodificadores lineares aprendidos, em que $|\mathcal{S}| = 2C$, desde que C decodificadores são aprendidos em cada ramo. Nesse caso, como a matriz F agrega toda informação intra- e inter-canal aprendida no módulo anterior, tem-se que uma representação vetorial da imagem pode ser construída a partir da aplicação de um conjunto $\mathcal{H} \subset \mathbb{F}(\mathbb{R}^{2CJ^2}, \mathbb{R})$ de funções de compressão, os quais tem por finalidade comprimir a matriz F em sua respectiva representação vetorial. Portanto, denotando $\vec{f}_k \in \mathbb{R}^{2CJ^2}$ como a k-ésima coluna da matriz F, tem-se que:

$$\vec{s}_h = (h(\vec{f}_1), h(\vec{f}_2), \ldots, h(\vec{f}_{Q+1})) \in \mathbb{R}^{Q+1} \tag{34}$$

$$\vec{\Omega}_Q(I) = \text{hconcat}(\{\vec{s}_h \,|\, h \in \mathcal{H}\}) \in \mathbb{R}^{|\mathcal{H}|(Q+1)}\,, \tag{35}$$

em que $\vec{s}_h$ denota o vetor $(Q+1)$-dimensional consistindo da compressão (sumarização) da matriz F sob a função de compressão $h \in \mathcal{H}$. Além disso, tem-se que $\vec{\Omega}_Q(I)$ denota a representação vetorial da imagem I utilizando incorporações aleatórias Q-dimensionais, a qual consiste da concatenação horizontal das distintas compressões da matriz F realizadas pelas funções de compressão pertencentes ao conjunto $\mathcal{H}$ (Fares; Ribas, 2025a).

Em particular, para o trabalho proposto, o conjunto de funções de compressão $\mathcal{H}$ consiste das mesmas funções de medidas estatísticas empregadas no método proposto da Subseção 4.1.2. Especificamente, tais medidas estatísticas são: média (h_μ), desvio padrão (h_σ), assimetria (h_γ) e curtose (h_κ). Nesse contexto, tem-se que $\mathcal{H} = \{h_\mu, h_\sigma, h_\gamma, h_\kappa\}$, e, portanto, a dimensionalidade da representação vetorial da imagem I apesentada na Equação 35 é dada por $4(Q+1)$, pois quatro representações vetoriais da forma $\vec{s}_h \in \mathbb{R}^{(Q+1)}$ serão concatenadas horizontalmente, pelo fato de que a matriz F será vista sob quatro perspectivas estatísticas distintas (Fares; Ribas, 2025a).

Por fim, de forma característica em abordagens de aprendizado de representações com redes neurais randomizadas, tem-se que em geral as representações oriundas da concatenação de representações obtidas a partir de incorporações aleatórias distintas (i.e., espaços de projeção com dimensões diferentes), geralmente, são mais discriminativas em tarefas de reconhecimento de texturas. Nesse contexto, a técnica proposta tem como representação de textura final o resultado da fusão tardia das representações na forma da Equação 35 a partir

da utilização de distintos tamanhos de incorporação aleatória. Dessa forma, tem-se que

$$\vec{\Upsilon}_{\mathcal{W}}(Q) = \text{hconcat}(\{\vec{\Omega}_w(I) \,|\, w \in \mathcal{W}\}) \in \mathbb{R}^m \,, \tag{36}$$

em que $\mathcal{W} = \{w_1, w_2, \ldots, w_n\} \subset \mathbb{N}$ é o conjunto das dimensões das incorporações aleatórias que serão utilizadas para compor a representação final $\vec{\Upsilon}_{\mathcal{W}}(I)$, de modo que a dimensionalidade de tal representação final é dada por $m = 4 \cdot \sum_{w \in \mathcal{W}}(w + 1)$ (Fares; Ribas, 2025a).

Avaliação Experimental. Para a avaliação experimental da técnica proposta, serão realizadas três análises: análise de regularização, análise de ablação e análise de parâmetros. Em todos os casos a métrica de desempenho avaliada será a acurácia, a qual é obtida a partir das predições obtidas utilizando o classificador *Linear Discriminant Analysis* juntamente com a técnica de validação cruzada *Leave-One-Out*. Além disso, os conjuntos de dados de textura colorida utilizados são: Outex (Ojala et al., 2002), CUReT (Dana et al., 1999), USPtex (Backes; Casanova; Bruno, 2012) e MBT (Abdelmounaime; Dong-Chen, 2013).

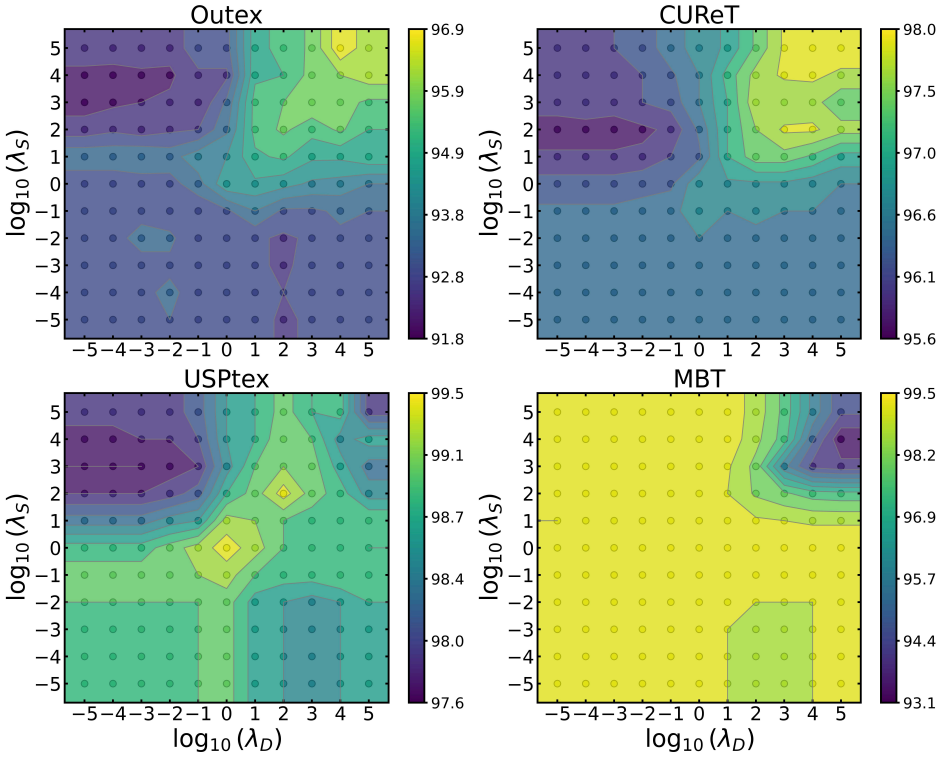
Análise de Regularização. Nesta parte, serão analisados os parâmetros de regularização de ambos submódulos direto e misto, com a finalidade de entender quais são os seus efeitos na discriminabilidade da representação de textura resultante. Isso posto, será analisado o comportamento do desempenho da representação de textura $\vec{\Omega}_{59}(I)$ em todos os conjuntos de dados de *benchmark*, à medida que os parâmetros de regularização λ_D e λ_S variam no espaço de parâmetros $\mathcal{R} \times \mathcal{R}$, em que $\mathcal{R} = \{10^k \mid k \in [-5, 5]_{\mathbb{Z}}\}$ (Fares; Ribas, 2025a).

A partir da Figura 29, tem-se que, para o conjunto de dados da Outex e CUReT, o desempenho da representação $\vec{\Omega}_{59}(I)$ apresentou comportamentos semelhantes. Primeiramente, observa-se uma melhora na acurácia da representação à medida que ambos os parâmetros de regularização são potencializados, com este fato evidenciado pelo aparecimento de uma região mais quente no canto superior direito dos gráficos da Outex e CUReT. Em segundo lugar, outra evidência desse comportamento semelhante é a presença de desempenhos reduzidos nos quadrantes restantes dos gráficos, identificada pela predominância de cores mais frias.

Conforme analisado em Fares & Ribas (2025a), para ambos os conjuntos da Outex e CUReT, a representação de textura beneficiou-se da utilização de submódulos direto e misto munidos com altos valores de regularização. Nesse contexto, isso indica que o módulo de aprendizado foi capaz de aprender representações mais robustas a partir da redução do ajuste que os decodificadores lineares tinham que realizar para reconstruir os padrões locais texturais presentes na imagem de entrada. Dessa forma, é possível que nessa redução de ajuste da reconstrução pela forte regularização, possivelmente, também tenha sido causada a redução do ajuste de ruídos presentes na imagem por parte do decodificador, portanto, resultando na consolidação de representações de textura mais robustas para esses conjuntos de dados.

Em seguida, para o conjunto de dados da USPtex, tem-se que a representação de textura apresentou desempenhos satisfatórios com valores de regularização moderados. Tal observação é evidenciada por regiões de cores mais quentes presentes no centro do gráfico.

Figura 29 – Comportamento da acurácia (%) da representação de textura $\vec{\Omega}_{59}(I)$ para todos os conjuntos de dados de *benchmark* à medida que variam os valores de regularização dos submódulos direto e misto.

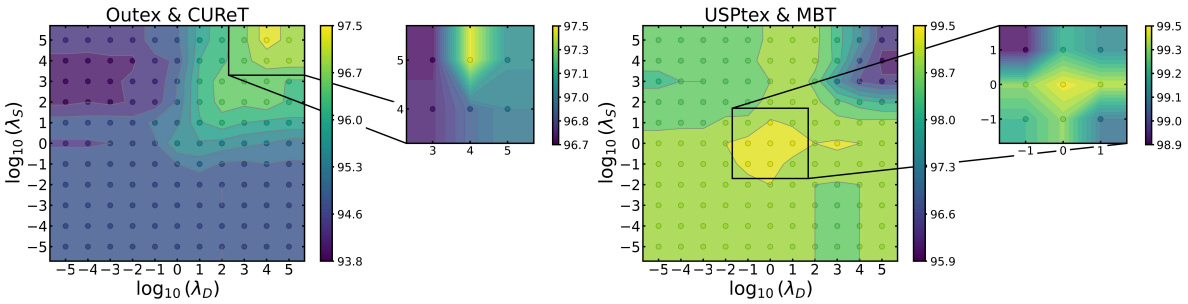


Fonte: Adaptada de Fares & Ribas (2025a).

Dessa forma, isso sugere que uma leve redução no ajuste por parte dos decodificadores lineares foi o suficiente para obter uma representação de textura discriminativa. Além disso, de acordo com Fares & Ribas (2025a), tem-se que para baixos valores de regularização para ambos submódulos, a representação de textura resultante obteve desempenhos moderados. No entanto, deve-se notar que à medida que fixa-se a regularização do submódulo direto e eleva-se apenas o do submódulo misto, observa-se uma redução do desempenho da representação. Isso exibe que o submódulo misto representa um componente essencial na descrição do conteúdo textural presente na imagem de entrada, de modo que, o aumento excessivo da regularização do submódulo misto, possivelmente, ocasionou um subajuste por parte deste submódulo resultando, portanto, em um enfraquecimento da discriminabilidade da representação como ilustrado pela presença de cores mais frias no quadrante superior esquerdo.

Com relação ao último conjunto de dados a ser analisado, sendo este o MBT, tem-se que esse apresentou um comportamento diametralmente oposto àquele apresentado pelos conjuntos de dados da Outex e CUReT. Especificamente, tem-se que quando ambos os submódulos são fortemente regularizados, as representações resultantes apresentam resultados menos expressivos, evidenciado pela presença de região mais fria no quadrante superior direito. Por outro lado, no restante do espaço de valores de regularização, as representações resultantes

Figura 30 – Comportamento da acurácia média (%) da representação de textura $\vec{\Omega}_{59}(I)$ para os pares de conjuntos de dados da Outex & CUReT e USPtex & MBT à medida que variam os valores de regularização dos ramos direto e misto.



Fonte: Adaptada de Fares & Ribas (2025a).

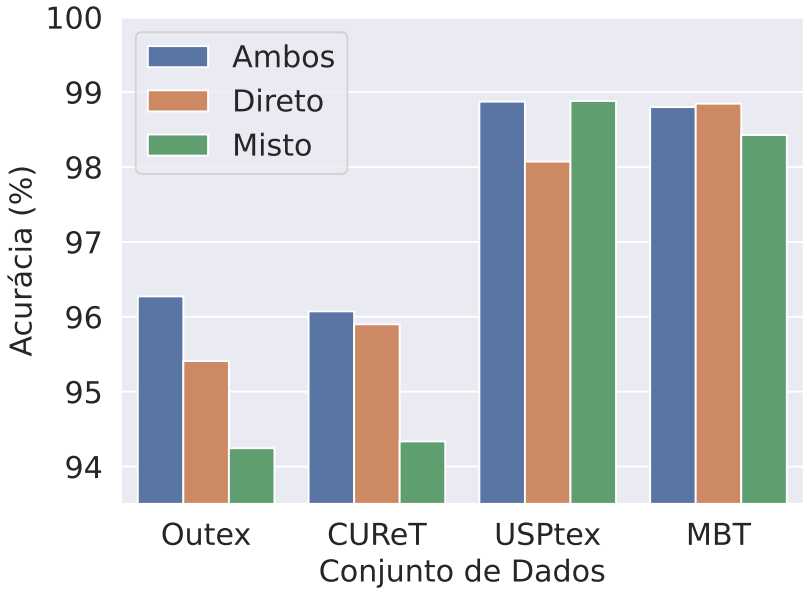
obtiveram resultados satisfatórios, sugerindo que as representações obtidas foram robustas em relação à variação dos parâmetros de regularização (Fares; Ribas, 2025a). Esse comportamento, indica que fortes valores de regularização sobre os ramos direto e misto ocasionam um subajuste, de modo que a representação resultante não contém características discriminativas suficientes para a caracterização eficiente do conteúdo textural presente nas imagens.

Por fim, a partir da análise anterior e dos resultados apresentados na Figura 29, é possível derivar dois pares de valores de regularização dos submódulos direto e misto para cada par de conjunto de dados analisado. Em particular, o primeiro par é composto dos conjuntos da Outex e CUReT, desde que esses apresentaram grande semelhança no comportamento do desempenho ao longo do espaço de parâmetros. Por outro lado, o segundo par é composto dos conjuntos de dados da USPtex e MBT, desde que ambos apresentaram resultados expressivos para valores de regularização moderados. Nesse sentido, a Figura 30 apresenta a média dos desempenhos para cada par de conjunto de dados, juntamente com os gráficos aumentados que exibem claramente os valores de regularização que obtiveram as maiores acurácias médias. Em particular, para o par Outex e CUReT, os melhores valores para regularização foram $\lambda_D = 10^4$ e $\lambda_S = 10^5$, enquanto que para o par USPtex e MBT, os melhores foram $\lambda_D = 10^0$ e $\lambda_S = 10^0$ (Fares; Ribas, 2025a). Nesse contexto, para o restante da apresentação da técnica do método proposto, tal como análise de ablação e análise de parâmetros, tem-se que tais valores de regularização encontrados serão utilizados.

Análise de Ablação. Esta parte dedica-se a analisar como cada submódulo contribui para a qualidade da representação de textura resultante. Dessa forma, na Figura 31 são apresentadas as acurácias médias da representação de textura $\vec{\Omega}_Q(I)$ quando um único submódulo ou ambos são utilizados. Em particular, tal análise é apresentada para todos os conjuntos de dados, e conta com a análise da acurácia média, a qual é obtida a partir da média dos valores de acurácia obtidos pelas representações com Q variando no espaço $\mathcal{Q}_{\text{Mixer}} = \{9, 19, \ldots, 109\}$.

A partir da Figura 31 é possível notar que para os conjuntos de dados da Outex e

Figura 31 – Comportamento da acurácia média (%) para a representação $\vec{\Omega}_Q(I)$ quando somente um dos submódulos ou ambos são utilizados.



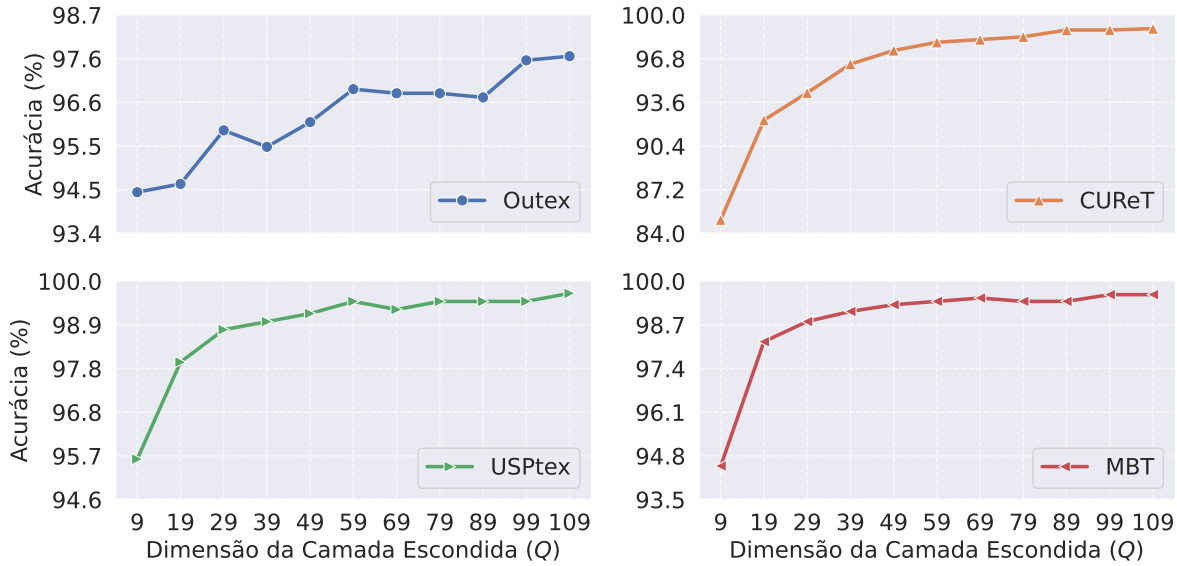
Fonte: Adaptada de Fares & Ribas (2025a).

CUReT, as representações de textura obtidas a partir da utilização de ambos os submódulos obtiveram em média uma acurácia maior em relação à utilização de um único submódulo, com tal aumento mais expressivo para a Outex. Em geral, isso indica que ambos os submódulos foram essenciais no aprendizado de uma representação mais discriminativa, sugerindo que as relações texturais intrínsecas aprendidas pelos decodificadores se complementaram (Fares; Ribas, 2025a).

Por outro lado, para os conjuntos de dados da USPtex e MBT, tem-se que nesses um único submódulo foi suficiente para atingir um bom nível de caracterização textural, obtendo, por exemplo, no MBT, uma acurácia maior do que quando ambos os submódulos são utilizados. No entanto, note-se que a seleção dos submódulos é um hiperparâmetro, e, dessa forma, poderá ser alterado à medida que trabalha-se com outros conjuntos de dados. Além disso, os resultados apresentados pela USPtex e MBT demonstram a importância da existência de tais submódulos, desde que na USPtex, o submódulo misto foi mais eficaz na caracterização textural, enquanto para o MBT o submódulo direto foi mais expressivo (Fares; Ribas, 2025a). Portanto, isso sugere que os submódulos são capazes de adaptarem-se às diferentes características ou desafios texturais presentes nos conjuntos de dados, e, que embora cada conjunto de dados apresente uma preferência por um único ou ambos, o restante da avaliação experimental do método proposto utilizará ambos os submódulos.

Análise de Parâmetros. Nesta parte, observa-se o comportamento do desempenho da representação de textura $\vec{\Omega}_Q(I)$, à medida que varia-se o hiperparâmetro Q, o qual representa a dimensão do espaço intermediário, i.e., a dimensão da camada escondida no qual os padrões

Figura 32 – Comportamento da acurácia da representação de textura $\vec{\Omega}_Q(I)$ à medida que altera-se dimensão da camada escondida.



Fonte: Adaptada de Fares & Ribas (2025a).

texturais brutos são projetados. Além disso, por completude, a análise é realizada sobre todos os conjuntos de dados apresentados, e utiliza-se o espaço de parâmetros $\mathcal{Q}_{\text{Mixer}}$ para a análise da dimensão da camada escondida (Fares; Ribas, 2025a).

A partir da Figura 32, observa-se que, embora, para o conjunto de dados da Outex, o desempenho da representação $\vec{\Omega}_Q(I)$ sofreu duas leves quedas, tem-se que a tendência dentro do espaço de parâmetros analisado é que o desempenho aumentou à medida que a dimensão da camada escondida aumentou, obtendo uma acurácia máxima de 97,6%. Por outro lado, os conjuntos de dados da CUReT, USPtex e MBT apresentaram um comportamento diferente da Outex, porém semelhante entre eles. Este comportamento é caracterizado por um crescimento acentuado do desempenho no início do espaço de parâmetros, porém que decresce à medida que a dimensão da camada escondida aumenta até estabilizar-se. Portanto, a dimensão da camada escondida se apresenta como um hiperparâmetro essencial na discriminabilidade da representação de textura final (Fares; Ribas, 2025a).

Em seguida, analisa-se a representação de textura final $\vec{\Upsilon}_{\mathcal{W}}(I)$, a qual é obtida a partir da concatenação de representações da forma $\vec{\Omega}_Q(I)$ para distintas dimensões da camada escondida. Isso é realizado, como anteriormente citado, pelo fato de que em técnicas de representações de textura utilizando redes neurais randomizadas é comum que representações finais obtidas a partir da concatenação de representações com dimensões menores obtenham maiores acurácias mesmo mantendo o mesmo número de atributos (Fares; Ribas, 2025a). Dessa forma, serão analisadas somente as representações finais resultantes da combinação de apenas duas representações com dimensões distintas para que não ocorra o surgimento de representações com números de atributos excessivos. Portanto, a seguir, o espaço de

Tabela 7 – Comportamento da acurácia da representação de textura $\vec{\Upsilon}_{Q_1,Q_2}(I)$ para todos os conjuntos de dados. Os resultados em **destaque** obtiveram as maiores acurácias médias. ↑ indica que maiores resultados são preferíveis.

	Acurácia (↑)						Acurácia (↑)				
(Q_1, Q_2)	Outex	CUReT	USPtex	MBT	Média	(Q_1, Q_2)	Outex	CUReT	USPtex	MBT	Média
Linha em branco deixada propositalmente						(39,49)	97,1	98,8	99,6	99.5	98,8
(9,19)	95,4	95,4	98,8	98,8	97,1	(39,59)	97,6	99,1	99,6	99.7	99,0
(9,29)	96,2	96,2	99,2	98,9	97,6	(39,69)	97,6	99,1	99,6	99.7	99,0
(9,39)	95,5	97,4	99,2	99,4	97,9	**(39,79)**	**97,9**	**99,2**	**99,5**	**99.6**	**99,1**
(9,49)	96,5	98,0	99,3	99,2	98,3	(39,89)	97,1	99,3	99,7	99.5	98,9
(9,59)	96,8	98,4	99,5	99,5	98,6	(39,99)	97,4	99,4	99,5	99.6	99,0
(9,69)	96,5	98,6	99,5	99,6	98,6	**(39,109)**	**97,8**	**99,5**	**99,7**	**99.7**	**99,2**
(9,79)	96,8	98,7	99,4	99,5	98,6	(49,59)	97,4	99,1	99,7	99.5	98,9
(9,89)	96,9	99,1	99,6	99,4	98,8	(49,69)	97,1	99,1	99,6	99.6	98,9
(9,99)	97,4	99,1	99,5	99,4	98,9	(49,79)	97,5	99,2	99,5	99.6	99,0
(9,109)	97,5	99,3	99,7	99,6	99,0	(49,89)	97,1	99,3	99,5	99.4	98,8
(19,29)	96,4	97,6	99,3	99,2	98,1	(49,99)	97,6	99,4	99,7	99.6	99,1
(19,39)	96,4	98,2	99,2	99,4	98,3	(49,109)	97,6	99,4	99,5	99.6	99,0
(19,49)	96,5	98,3	99,3	99,3	98,4	(59,69)	97,6	99,2	99,7	99.6	99,0
(19,59)	97,1	98,6	99,6	99,5	98,7	(59,79)	97,6	99,5	99,6	99.6	99,1
(19,69)	97,2	98,8	99,5	99,4	98,7	(59,89)	97,4	99,5	99,6	99.6	99,0
(19,79)	97,3	99,0	99,6	99,5	98,9	(59,99)	97,6	99,5	99,5	99.6	99,1
(19,89)	97,1	99,2	99,6	99,5	98,9	(59,109)	97,4	99,6	99,6	99.7	99,1
(19,99)	97,4	99,3	99,5	99,6	99,0	(69,79)	97,8	99,3	99,5	99.8	99,1
(19,109)	97,5	99,2	99,7	99,7	99,0	(69,89)	97,4	99,4	99,7	99.6	99,0
(29,39)	96,6	98,4	99,6	99,4	98,5	(69,99)	97,2	99,5	99,6	99.5	99,0
(29,49)	96,8	98,5	99,5	99,3	98,5	**(69,109)**	**97,7**	**99,6**	**99,6**	**99.7**	**99,2**
(29,59)	97,4	98,9	99,6	99,3	98,8	(79,89)	97,7	99,5	99,7	99.6	99,1
(29,69)	97,3	98,9	99,6	99,6	98,9	(79,99)	97,3	99,6	99,6	99.5	99,0
(29,79)	97,6	99,1	99,5	99,5	98,9	(79,109)	97,5	99,6	99,7	99.6	99,1
(29,89)	97,3	99,3	99,7	99,4	98,9	(89,99)	97,3	99,6	99,6	99.6	99,0
(29,99)	97,1	99,3	99,7	99,4	98,9	(89,109)	97,4	99,6	99,7	99.7	99,1
(29,109)	97,8	99,4	99,6	99,5	99,1	(99,109)	97,4	99,7	99,7	99.7	99,1

Fonte: Adaptada de Fares & Ribas (2025a).

parâmetros a ser analisado é expresso por:

$$\mathcal{E}_{\text{Mixer}} = \{(Q_1, Q_2) \in \mathcal{O} \times \mathcal{O} \mid Q_1 < Q_2\}. \tag{37}$$

A Tabela 7 apresenta as acurácias obtidas pelas diferentes configurações da representação de textura $\vec{\Upsilon}_{Q_1,Q_2}(I)$. Dessa forma, conforme exibido em Fares & Ribas (2025a), a representação de textura combinada $\vec{\Upsilon}_{Q_1,Q_2}(I)$ obteve maiores acurácias em relação à representação de textura $\vec{\Omega}_Q(I)$, mesmo embora tal aumento tenha sido discreto. Nesse sentido, isso sugere que a representação $\vec{\Omega}_Q(I)$ é capaz de caracterizar eficientemente o conteúdo textural presente na imagem sem a necessidade da realização de combinações, de modo que, isso facilita sua adaptação em outras tarefas de classificação, desde que não será necessário investigações sobre distintas combinações de configurações da técnica proposta. Notadamente, o ganho mais expressivo foi observado no conjunto de dados da CUReT, com 99,7% ($\vec{\Upsilon}_{99,109}(I)$) em comparação a 99,0% ($\vec{\Omega}_{109}(I)$). Em seguida, destacam-se os resultados obtidos pela Outex, que alcançou 97,9% ($\vec{\Upsilon}_{39,79}(I)$) frente aos 97,7% ($\vec{\Omega}_{109}(I)$), e na MBT, que registrou

99,8% ($\vec{\Upsilon}_{69,79}(I)$) em contraste com 99,6% ($\vec{\Omega}_{109}(I)$). Por outro lado, no conjunto de dados da USPtex não se verificou qualquer ganho de desempenho (Fares; Ribas, 2025a).

Além disso, a Tabela 7 exibe a existência de instâncias de configurações de representação de textura final $\vec{\Upsilon}_{Q_1,Q_2}(I)$, as quais obtêm acurácias maiores em relação a representações de textura da forma $\vec{\Omega}_Q(I)$ ainda que mantido o mesmo número de atributos. Nesse contexto, tais observações conforme discutido em Fares & Ribas (2025a) está em concordância com outros estudos em aprendizado de representações de textura com redes neurais randomizadas. Em particular, no conjunto de dados da Outex, tem-se que a representação de textura $\vec{\Upsilon}_{29,59}(I)$ obteve uma acurácia de 97,4% com 360 atributos, ao passo que a representação $\vec{\Omega}_{89}(I)$ obteve 96,7% também com 360 atributos. Como segundo exemplo, na CUReT, a representação de textura $\vec{\Upsilon}_{9,19}(I)$ alcançou acurácia de 95,4% com 120 atributos, enquanto que a representação $\vec{\Omega}_{29}(I)$ obteve 94,3% com o mesmo número de atributos.

Portanto, com a finalidade de comparação com outros métodos da literatura, os quais serão conduzidos em breve, tem-se que as representações de textura final a serem utilizadas durante a comparação são aquelas da forma $\vec{\Upsilon}_{\mathcal{W}}(I)$ e que obtiveram as maiores acurácias médias na Tabela 7, sendo elas: Mixer$_a$ ($\vec{\Upsilon}_{39,109}$), Mixer$_b$ ($\vec{\Upsilon}_{69,109}$) e Mixer$_c$ ($\vec{\Upsilon}_{39,79}$), as quais obtiveram acurácias médias de 99,2%, 99,2% e 99,1%[5], respectivamente.

Contribuições e Limitações. Embora a metodologia apresentada seja munida de diversas adições e peculiaridades em relação às abordagens com redes neurais randomizadas, tem-se que são duas as principais contribuições da técnica proposta. Primeiramente, a abordagem apresentada propõe um novo módulo de projeção aleatória hiperesférica munido de múltiplas cabeças de projeção, uma para cada canal da imagem, a qual é responsável por construir incorporações aleatórias hiperesféricas, as quais melhoram a estabilidade numérica durante a etapa de treinamento. Por outro lado, a segunda e mais proeminente contribuição é a construção de um módulo de aprendizado munido dos submódulos direto e misto, os quais são responsáveis por aprender os padrões texturais intra- e inter-canal, respectivamente. Em particular, o submódulo misto tem seu treinamento guiado por uma nova fórmula de otimização a qual emprega uma incorporação aleatória fundida intermediariamente, com a finalidade de aprender os padrões texturais presentes em um dado canal da imagem, a partir dos padrões texturais presentes em todos os outros canais da imagem, dessa forma, resultando em um processo de aprendizado substancialmente mais rico em informações.

No entanto, a técnica proposta apresenta algumas limitações arquiteturais as quais podem ser resolvidas como trabalhos futuros. Em primeiro lugar, conforme discutido em Fares & Ribas (2025a), observa-se o desenvolvimento de novas estratégias de construção das matrizes de entrada e saída, com a finalidade de melhorar a descrição do conteúdo textural

[5] Deve-se notar que existem múltiplas configurações da representação $\vec{\Upsilon}_{Q_1,Q_2}(I)$, as quais obtiveram a acurácia média de 99,1%. No entanto, a representação $\vec{\Upsilon}_{39,79}$ será selecionada, pois obtém tal acurácia com o menor número de atributos, sendo composta por 480 atributos.

bruto presente na imagem de entrada, e, ao mesmo tempo, elevando a complexidade e expressividade do aprendizado, assim, tornando o processo de aprendizado mais robusto, uma vez que a rede deve inferir relações entre essas informações. Em segundo lugar, identifica-se a possibilidade de formas alternativas para a construção da incorporação aleatória intermediária que é utilizada pelo submódulo misto para guiar seu aprendizado. Por fim, em terceiro lugar, pode-se conferir maior robustez ao módulo de aprendizado a partir da incorporação de novas formulações de problemas de otimização convexa para o cálculo dos pesos dos decodificadores lineares, tal como o uso de formulações dos mínimos quadrados multiobjetivo.

4.2.4 Comparação com Métodos da Literatura

Nesta subseção, realizam-se a comparação e a análise dos resultados obtidos pelas abordagens de aprendizado de representações de textura colorida previamente descritas em relação àquelas presentes na literatura. Nesse contexto, da mesma forma como realizado na Subseção 4.1.3, serão apresentados, a seguir, o protocolo experimental, os conjuntos de dados analisados e as distintas técnicas da literatura consideradas para comparação e discussão.

Protocolo Experimental. Com relação aos resultados apresentados pelas abordagens propostas e pelas técnicas a serem comparadas, tem-se que esses foram obtidos empregando o classificador *Linear Discriminant Analysis* com a estratégia de validação cruzada *Leave-One-Out*. Ainda, para uma avaliação abrangente das técnicas propostas, empregaram-se quatro conjuntos de dados de imagens de textura colorida com propriedades texturais distintas, permitindo uma análise mais ampla das abordagens propostas. Em particular, tais conjuntos de dados são: Outex (Ojala et al., 2002), CUReT (Dana et al., 1999), USPtex (Backes; Casanova; Bruno, 2012) e MBT (Abdelmounaime; Dong-Chen, 2013).

As técnicas a serem comparadas também estão divididas nas mesmas categorias como realizado na Subseção 4.1.3. Desse modo, as abordagens artesanais comparadas são: Opponent-Gabor (Jain; Healey, 1998), GLCM (Haralick; Shanmugam; Dinstein, 1973), Fourier (Weszka; Dyer; Rosenfeld, 1976), Fractal (Backes; Casanova; Bruno, 2009), Fractal Fourier (Florindo; Bruno, 2012), LBP (Ojala; Pietikainen; Maenpaa, 2002), LPQ (Ojansivu; Heikkilä, 2008), AHP (Zhu et al., 2015), BSIF (Kannala; Rahtu, 2012), CLBP (Guo; Zhang; Zhang, 2010a), CNTD (Backes; Casanova; Bruno, 2013), MCND (Scabini et al., 2019) e SSN (Scabini; Ribas; Bruno, 2020). Com relação às abordagens baseadas em RNNs, essas são: RNN-RGB (Sá Junior; Backes; Bruno, 2019) e SSR (Ribas et al., 2024a). Por fim, as abordagens baseadas em DCNNs são: VGG11 (Simonyan; Zisserman, 2014), ResNet18 e ResNet50 (He et al., 2016), DenseNet121 (Huang et al., 2017) e InceptionResNetV2 (Szegedy et al., 2017).

Análise Comparativa. A Tabela 8 apresenta a comparação das acurácias de classificação obtidas pelas técnicas propostas (SST, VCTex e Mixer) em relação às técnicas da literatura. De

Tabela 8 – Comparação das acurácias de classificação obtidas pelos métodos propostos com aqueles presentes na literatura. Um subconjunto destes resultados foi obtido em Ribas et al. (2024a) e Fares & Ribas (2025b). Em cada coluna, o maior valor está em **destaque**, o valor em azul indica o melhor resultado da literatura, e em vermelho indica resultados iguais ou superiores por parte dos métodos propostos.

Método	# de Atributos	Acurácia (↑)			
		Outex	CUReT	USPtex	MBT
Métodos Artesanais					
Opponent-Gabor	264	93,5	95,8	99,1	97,6
GLCM†	72	91,1	84,5	95,9	97,4
Fourier†	189	85,8	82,7	91,6	96,3
Fractal†	207	86,8	82,0	95,0	97,0
Fractal Fourier†	204	76,2	91,1	81,0	97,5
LBP†	768	82,4	91,4	90,2	96,6
LPQ†	768	80,1	91,7	90,4	95,7
AHP†	360	93,4	95,5	98,7	98,1
BSIF†	768	77,9	93,7	82,9	97,9
CLBP†	1944	89,6	91,8	97,4	98,2
CNTD†	324	92,3	91,9	97,9	98,5
MCND	480**	95,4	97,0	99,0	97,1
SSN (φ^4)	648	96,8	98,6	99,5	99,0
Métodos Baseados em RNNs					
RNN-RGB	180	94,8	–	98,4	–
SSR_a	630	96,7	98,6	99,3	98,2
SSR_b	990	96,8	98,7	99,0	98,0
Métodos Baseados em DCNNs					
VGG11	512	89,9	96,5	99,3	93,8
ResNet18	512	86,7	96,7	98,7	90,8
ResNet50	2048	86,8	98,6	89,4	85,6
DenseNet121	1024	84,7	98,7	99,4	94,1
InceptionResNetV2	1536	82,4	98,5	96,7	89,5
Métodos Propostos					
SST_a	756	96,8	95,6	99,1	**99,8**
SST_b	540	96,2	94,6	99,2	99,6
SST_c	324	95,9	93,2	99,1	99,5
$VCTex_a$	648	96,0	96,0	99,6	99,1
$VCTex_b$	540	95,8	95,4	99,6	98,9
$VCTex_c$	1296	95,4	96,5	**99,7**	99,2
$Mixer_a$	600	97,8	99,5	**99,7**	99,7
$Mixer_b$	720	97,7	**99,6**	99,6	99,7
$Mixer_c$	480	**97,9**	99,2	99,5	99,6

*O hífen indica que o valor não estava disponível ou não foi possível obtê-lo.

**O tamanho do descritor MCND tem 1440 atributos para a CUReT e USPtex.

†Métodos originalmente nível de cinza utilizados na versão integrativa.

Fonte: Dados da pesquisa.

modo geral, observa-se que as representações obtidas por meio das técnicas propostas obtêm majoritariamente as três melhores acurácias em todos os conjuntos de dados analisados. Desse modo, à primeira vista, nota-se a superioridade metodológica dessas abordagens.

Nesse contexto, em relação à abordagem proposta SST, essa obteve as duas melhores representações no conjunto de dados da MBT (SST_a e SST_b). A primeira representação obteve 99,8%, enquanto a segunda 99,6%; representando, assim, um aumento de 0,8% e 0,6%, respectivamente, em relação ao maior resultado anterior obtido, que foi pela SSN (Scabini; Ribas; Bruno, 2020) com 99,0%. Em seguida, comparando tais representações com o SSR (Ribas et al., 2024a), método baseado em RNN que obteve 98,2% e 98,0%, tem-se que essas alcançaram aumentos expressivos de até 1,6% e com uma quantidade menor de atributos quando diz respeito à SST_b. Acerca das abordagens baseadas em DCNNs, a técnica desenvolvida SST alcança incrementos significativos, por exemplo, SST_a obteve 99,8% com 756 atributos, enquanto DenseNet121 alcançou 94,1% com 1024 atributos, um incremento de 5,7% com redução de 268 atributos.

Em particular, para o conjunto da MBT, tais resultados substanciam os benefícios da representação de textura espaço-espectral cuja uma das etapas de treinamento principais é guiada forçando a rede a predizer os padrões texturais de um canal específico a partir de seu adjacente (i.e., treinamento espectral), assim abordando a peculiaridade desse conjunto de dados, em que que cada canal da imagem é formado por um conteúdo textural distinto. Por outro lado, para a Outex e USPtex, a abordagem manteve sua competitividade alcançando 96,8% (SST_a) e 99,2% (SST_b), respectivamente. Por fim, com relação ao conjunto de dados CUReT, a metodologia proposta apresentou limitações para sua caracterização textural, sendo superada pelas abordagens SSN e SSR. Entretanto, a abordagem proposta ainda supera majoritariamente as abordagens artesanais.

Quanto à técnica desenvolvida VCTex, essa apresentou elevada discriminabilidade textural para o conjunto de dados da USPtex, atingindo acurácias de 99,7% (VCTex_c) e 99,6% (VCTex_a e VCTex_b). Esses resultados apresentam um aumento de 0,2% em relação a melhor técnica anterior SSN (Scabini; Ribas; Bruno, 2020) e aumentos consideráveis de até 1,3% em relação às abordagens RNN-RGB (Sá Junior; Backes; Bruno, 2019) e SSR (Ribas et al., 2024a), as quais empregam RNNs. Logo, a VCTex apresentou-se competitiva com a SSN e superior em relação às técnicas RNN-RGB e SSR nesse conjunto. Especificamente, com relação aos últimos dois métodos citados, um dos fatores para o incremento alcançado pela VCTex atribui-se à forma como o conteúdo textural espaço-espectral é capturado, em que se emprega a varredura de cubos de cor $3\times3\times3$ sobre toda imagem, capturando simultaneamente os padrões texturais e espectrais locais da imagem de textura colorida.

Além disso, sob o mesmo fator e sobre a peculiaridade do conjunto MBT discutida anteriormente, a VCTex obteve resultados expressivos como 99,2% (VCTex_c) e 99,1% (VCTex_a), que embora indiquem um aumento sutil em relação à SSN, apresentam um aumento relevante

de até 1,2% em relação à SSR. Para o conjunto da Outex, a VCTex apresentou menor discriminabilidade textural quando comparada à SSN e SSR. Todavia, a VCTex não só supera todas as abordagens artesanais, exceto SSN, como também demonstra aumentos expressivos de 13,6% quando comparada às DCNNs. Em última análise, a VCTex apresentou a mesma peculiaridade da SST no conjunto de dados da CUReT, em que obteve-se desempenhos inferiores, por exemplo, em relação à SSN, SSR e DCNNs. Nesse sentido, acredita-se que esta baixa discriminabilidade esteja relacionada à forma como os pesos das redes neurais randomizadas foram sumarizados nos métodos propostos, desde que, ambos VCTex e SST utilizam a abordagem de esticamento dos pesos[6], que se exibiu ineficiente nesse conjunto de dados.

Com relação à abordagem proposta Mixer, as representações resultantes obtiveram sempre as três melhores acurácias para todos os conjuntos de dados. Especificamente, essa abordagem não somente obtém os melhores desempenhos (i.e., 1º lugar) nos conjuntos de dados da Outex, CUReT e USPtex como também supera todas as abordagens da literatura comparadas. Em particular, para o conjunto da Outex, Mixer foi a única abordagem capaz de ultrapassar 97% de acurácia, situando-se apenas 0,1% da marca dos 98%. Comparando com a SSN e SSR_b, as melhores abordagens comparadas da literatura na Outex, tem-se que Mixer_c apresenta um incremento de 1,1%, além de empregar uma representação mais compacta, desde que SSN e SSR_b utilizam 648 e 990 atributos, respectivamente, enquanto Mixer_c apenas 480.

Em relação à CUReT, tem-se que as representações da técnica proposta Mixer foram as únicas que ultrapassaram a marca dos 99%. Comparando-a com as técnicas SSR_b e DenseNet121 que atingiram a maior acurácia de 98,7% entre os métodos listados, Mixer_b alcança 99,6% superando-os com um incremento de 0,9%, que indica um aumento de aproximadamente 50 imagens classificadas corretamente. Nesse contexto, de forma análoga na Outex, a representação Mixer_b apresenta um incremento considerável utilizando uma representação com menor dimensionalidade, em relação à SSR_b, reforçando sua eficiência e efetiva discriminabilidade textural em um conjunto de dados munido com desafios de distintos pontos de vista e iluminação.

Para o conjunto da USPtex, tem-se que Mixer_a (99,7%) e Mixer_b (99,6%) apresentam aumentos sutis de até 0,2% em relação à SSN (99,5%), que é a melhor abordagem comparada. Além disso, Mixer_c mantém a mesma acurácia de 99,5% da SSN, entretanto, a menor representação para tal, reitera sua eficiência assim como na Outex e CUReT. Por fim, na MBT, as representações de Mixer não alcançam o primeiro lugar apenas pela representação SST_a, a qual também foi proposta neste trabalho de mestrado. No entanto, Mixer_a e Mixer_b alcançam as acurácias de 99,7% as quais superam todos os métodos comparados da literatura. Em particular, tal desempenho indica um incremento de 0,7% em relação à SSN, aumento de até 1,7% em relação às abordagens baseadas em RNNs, e de até 14,1% para as DCNNs.

[6] Para fins de recapitulação, a técnica de sumarização por esticamento de pesos refere-se à função flatten$(\cdot)$ que recebe como entrada um tensor n-dimensional e produz como saída um tensor 1-dimensional contendo todos os elementos desse tensor de entrada dispostos horizontalmente.

Discussões. Com base nos resultados apresentados e na análise comparativa fornecida, observou-se que as abordagens propostas contribuíram para o avanço dos resultados do estado da arte. Em particular, embora as abordagens SST e VCTex alcançaram desempenhos competitivos na comparação geral, essas apresentaram aumentos significativos em conjuntos de dados específicos, tal como MBT e USPtex, respectivamente. Por outro lado, o método proposto Mixer demonstrou claramente sua superioridade metodológica, sendo essa substanciada pela observação de que as representações propostas mantiveram-se, consistentemente, entre os três melhores resultados em todos os conjuntos de dados de textura analisados.

Em particular, com relação à SST, os desempenhos competitivos e o incremento significativo alcançado na MBT devem-se, principalmente, aos módulos de aprendizado espacial e espectral cujas responsabilidades são aprender o conteúdo textural intra- e inter-canal, respectivamente. No entanto, diferentemente do Mixer, a abordagem SST realiza o aprendizado espectral apenas para os canais adjacentes e sem a utilização de qualquer incorporação específica. Além disso, a metodologia empregada para a sumarização dos pesos aprendidos dos *autoencoders* randomizados para a composição da representação são simples em relação ao utilizado na técnica Mixer. Desse modo, acredita-se que tais limitações contribuíram para o Mixer superar SST em três (Outex, CUReT e USPtex) dos quatro conjuntos de dados.

Entretanto, ao comparar o SST com outras abordagens da literatura baseadas em redes neurais randomizadas, observa-se que, em relação à RNN-RGB, o método proposto supera consistentemente todos os resultados obtidos por essa técnica, enquanto que em relação à SSR, mantém desempenho competitivo nos conjuntos Outex e USPtex e apresenta superioridade no MBT. Em particular, as vantagens metodológicas da abordagem podem ser atribuídas, sobretudo, a dois fatores. Primeiramente, o SST realiza o aprendizado exclusivamente a partir de padrões locais finos (i.e., conteúdo microtextural) por meio da extração de pequenas janelas de dimensão 3×3, ao passo que as abordagens comparadas empregam, adicionalmente, regiões maiores, como 5×5 e 7×7 no caso da RNN-RGB. Em segundo lugar, o módulo espectral permite o enriquecimento substantivo da representação final de textura, uma vez que incorpora os pesos obtidos durante o aprendizado textural inter-canal aplicado a todas as combinações de canais adjacentes, o que, por sua vez, eleva a discriminabilidade textural.

Por fim, quanto à abordagem proposta Mixer, como mencionado anteriormente, tem-se que essa evidenciou-se como a melhor representação de textura neste trabalho. Para isso, Mixer propôs diversas melhorias, principalmente arquiteturais, no processo de aprendizado de representações utilizando redes neurais randomizadas. Primeiramente, a abordagem integra um projetor aleatório hiperesférico multicabeça cuja responsabilidade de cada cabeça é projetar aleatoriamente os padrões texturais de cada canal independentemente, assim, incorporando informações de propriedade do canal, e, em seguida, efetuar a projeção das incorporações aleatórias obtidas sobre a superfície de uma hiperesfera com a finalidade de aumentar a estabilidade numérica durante o treinamento. Em segundo lugar, a abordagem contém um novo

módulo de aprendizado subdividido em dois submódulos direto e misto cujas responsabilidades são aprender as características texturais intrínsecas intra- e inter-canal. No entanto, diferentemente das abordagens anteriores, o aprendizado inter-canal é efetuado por meio do emprego de uma nova incorporação aleatória obtida durante uma etapa de fusão intermediária a qual agrega toda a informação textural local presente em cada canal da imagem. Nesse contexto, tais modificações arquiteturais mostraram-se benéficas, as quais foram substanciadas pelos resultados exibidos na Tabela 8.

4.3 CONSIDERAÇÕES FINAIS

Neste capítulo, foram apresentadas as abordagens investigadas e desenvolvidas de representação de texturas com redes neurais randomizadas puras. Especificamente, foram propostas duas abordagens de representações de textura para imagens em nível de cinza, enquanto três foram apresentadas para imagens coloridas. Em particular, foram estudadas, analisadas e desenvolvidas abordagens que abordassem uma ou mais propriedades dentre os tópicos de: modelagem do conteúdo textural das matrizes de entrada e saída, melhoria da capacidade arquitetural da rede e melhores formas de sumarização dos pesos aprendidos durante a etapa de treinamento; tópicos esses que foram citados como foco no começo do capítulo.

As abordagens propostas resultaram em representações de texturas eficientes obtendo acurácias de classificação significativas com representações compactas, i.e., representações com baixo número de atributos. Além disso, as relevantes acurácias de classificação apresentadas nas tabelas de comparação demonstraram a competitividade e superioridade das abordagens propostas em relação aos métodos do estado da arte comparados. Não apenas isso, mas também, os relevantes desempenhos computacionais das abordagens propostas sugerem que essas podem ser empregadas em cenários cujos recursos computacionais são limitados ou que possuam limitações temporais. Nesse contexto, esses pontos coincidem com a alta discriminabilidade textural e eficiência computacional dos métodos propostos, o que reforça ainda mais as contribuições metodológicas investigadas e desenvolvidas nesta dissertação.

Por fim, o capítulo seguinte apresentará as técnicas investigadas e desenvolvidas para aprendizado de representações de texturas com redes neurais randomizadas híbridas.

5 APRENDIZADO DE REPRESENTAÇÕES COM REDES NEURAIS RANDOMIZADAS HÍBRIDAS

Neste capítulo, apresenta-se a abordagem proposta para o aprendizado de representações de textura com redes neurais randomizadas híbridas. Nesse caso, diferentemente das técnicas de aprendizado de representações puras, as abordagens caracterizadas como híbridas são aquelas que empregam, como extratores de características, modelos de aprendizado profundo pré-treinados em conjuntos de dados massivos. Em seguida, utiliza-se uma rede neural randomizada com a finalidade de codificar aleatoriamente as características extraídas, visando à obtenção de uma representação de textura mais robusta. Desse modo, o emprego de tais modelos como extratores de características, ainda que a tarefa final seja substancialmente distinta daquela considerada durante o pré-treinamento, elimina a necessidade de treinamento adicional da rede (i.e., *fine-tuning*), o qual é computacionalmente custoso e pode acarretar problemas de sobreajuste, uma vez que conjuntos de dados de textura são ínfimos quando comparados, por exemplo, à ImageNet (Scabini, 2023). Ademais, a dispensa do treinamento possibilita que as técnicas híbridas operem de maneira análoga às técnicas manuais e puras, permitindo a extração de representações de textura por instância (i.e., para cada imagem).

Além disso, a emergência desta categoria de técnicas híbridas, as quais realizam um pós-processamento ou, mais especificamente, nesse caso, a caracterização e/ou codificação aleatória das características extraídas por meio dos modelos de aprendizado profundo, decorre do fato de que apenas o emprego direto dessas características profundas (i.e., por meio da aplicação do *Global Average Pooling*), sob certos domínios de imagens de textura, torna-se insuficiente para a adequada caracterização textural, sendo ineficiente mesmo em relação às técnicas manuais (Scabini, 2023). Nesse sentido, evidencia-se a necessidade de proposições de abordagens de representações híbridas, como feita nesta dissertação, para que seja possível a extração e o enriquecimento dos padrões extraídos por esses modelos de aprendizado profundo.

Por fim, assim como realizado nas técnicas puras, tem-se que a qualidade da representação resultante é fortemente dependente de dois aspectos centrais: (i) a forma como os mapas de ativação provenientes dos distintos estágios do *backbone* são pré-processados e/ou tratados; e (ii) o modo com as matrizes de entrada e saída da rede neural randomizada são construídas, desde que esse é aspecto relevante, uma vez que determina como as informações são internamente correlacionadas pela pela rede neural e, consequentemente, está extremamente ligada aos pesos aprendidos resultantes que são utilizados para compor a representação. Isso posto, nesta dissertação, investigou-se e desenvolveu-se uma nova estratégia para o tratamento dos diferentes mapas de ativação, bem como uma nova forma de cruzar as informações provenientes diretamente desses mapas com aquelas obtidas após tratamentos do mesmo. No entanto, tais contribuições se elucidarão ao decorrer deste capítulo, à medida que se apresenta

a abordagem de representação de textura proposta. Dessa forma, subdivide-se este capítulo em duas seções. A Seção 5.1 introduz a abordagem de representações híbrida proposta, a qual obtém representações de textura por meio do emprego de codificação aleatória de características profundas provenientes de *backbones* convolucionais. Por outro lado, a Seção 5.2 dedica-se a apresentação das considerações finais com relação à abordagem proposta e a avaliação experimental conduzida, assim, finalizando esse capítulo.

5.1 APRENDIZADO DE REPRESENTAÇÕES POR CODIFICAÇÃO ALEATÓRIA DE CARACTERÍSTICAS DE BACKBONES CONVOLUCIONAIS

5.1.1 Aprendizado de Representações de Textura por Caracterização Topológica e Codificação Aleatória de Mapas de Ativação Profundos

Esta subseção introduz a técnica de aprendizado de representações de texturas por meio da caracterização topológica e codificação aleatória de mapas de ativação profundos, denominado por TOCHA (do inglês, *texture representation learning by TOpological CHAracterization and random encoding of deep activation maps*). Brevemente, a abordagem proposta inova ao empregar a codificação aleatória de características extraídas de uma sequência de estágios do *backbone*, forçando a rede neural randomizada a predizer medidas topológicas (i.e., graus de saída) somente a partir das intensidades das ativações associadas aos vértices modelados. Nesse contexto, ao submeter a rede a essa tarefa, possibilita-se a captura de relacionamentos e padrões inerentes à forma como as ativações se organizam e à conectividade entre os vértices, de modo que tal conhecimento é implicitamente incorporado aos pesos aprendidos, os quais passam a constituir a própria representação de textura. Além disso, com a finalidade de tornar tal abordagem ainda mais robusta, incorpora-se na representação de cada estágio um componente desordenado (em inglês, *orderless*), que descarta a conexão entre as posições relativas ou absolutas e as suas respectivas características visuais locais.

Modelagem em Grafo. Seja $A \in \mathbb{R}^{C \times H \times W}$ um tensor tridimensional que representa os mapas de ativação da saída de um estágio de algum *backbone*. A partir desse, constrói-se o grafo $G = (V, E)$, de modo que, para cada elemento $A_{x,y,z}$, existirá um vértice $v_{x,y,z} \in V$ correspondente. Além disso, uma aresta direcionada $e^{i,j,k}_{x,y,z}$ partindo do vértice $v_{x,y,z}$ para o vértice $v_{i,j,k}$ existirá se, e somente se, a distância euclidiana entre as posições espaciais correspondentes de tais vértices for positiva e no máximo de $r\sqrt{3}$, em que $r \in \mathbb{N}$ é um hiperparâmetro que define o raio da vizinhança do vértice. Matematicamente, tem-se que:

$$e^{i,j,k}_{x,y,z} \in E \iff 0 < \sqrt{(x-i)^2 + (y-j)^2 + (z-k)^2} \leq r\sqrt{3}\,. \tag{1}$$

Note-se que a partir desta condição de existência, é possível inferir dois aspectos. Em primeiro lugar, não existirão laços neste grafo, i.e., arestas que conectam o mesmo vértice, desde que a distância euclidiana entre tais vértices deverá ser positiva. Em segundo lugar, se a aresta $e^{i,j,k}_{x,y,z}$

existir, então a aresta $e_{i,j,k}^{x,y,z}$ também existirá, desde que a métrica de distância é simétrica. Ainda mais, com a finalidade de enriquecer os mapas de ativação modelados, para toda aresta $e_{x,y,z}^{i,j,k} \in E$ será atribuído o peso $w_{x,y,z}^{i,j,k} = A_{x,y,z} - A_{i,j,k}$, o qual carregará a informação da diferença de intensidades das ativações realizadas nas posições correspondentes aos vértices.

Nesse contexto, a modelagem apresentada possibilita o estabelecimento de relações entre as ativações produzidas por um mesmo filtro em distintas posições espaciais, situação na qual o terceiro índice se mantém idêntico; bem como entre as ativações de diferentes filtros do *backbone* em uma mesma região da imagem, situação na qual os dois primeiros índices permanecem idênticos. Dessa forma, no primeiro caso, captura-se a informação de como o mesmo filtro ativa diferentes regiões da imagem, enquanto que no segundo caso, obtém-se a relação de como distintos filtros ativam a mesma região. Entretanto, tais situações constituem casos particulares e, de forma mais geral, a modelagem proposta realiza o relacionamento entre as ativações produzidas por distintos filtros pré-treinados em diferentes regiões da imagem.

Por fim, deve-se notar que para cada vértice $v \in V$ presente no grafo, tem-se que esse está conectado com sua vizinhança com base somente na relação de distância entre os vértices, de modo que, tal forma de conectividade está ausente de informações extraídas por meio do modelo pré-treinado, sendo tal informação (i.e., as intensidades das ativação) presente somente nos pesos. Dessa forma, com a finalidade de introduzir o conteúdo extraído pelo modelo e permitir o surgimento de interações complexas, tem-se que alterações topológicas serão realizadas no grafo original. Para esse fim, define-se o seguinte grafo $G^{(t)} = (V, E^{(t)})$ que representa o grafo resultante a partir do corte das arestas de G por $t \in \mathbb{R}$, em que

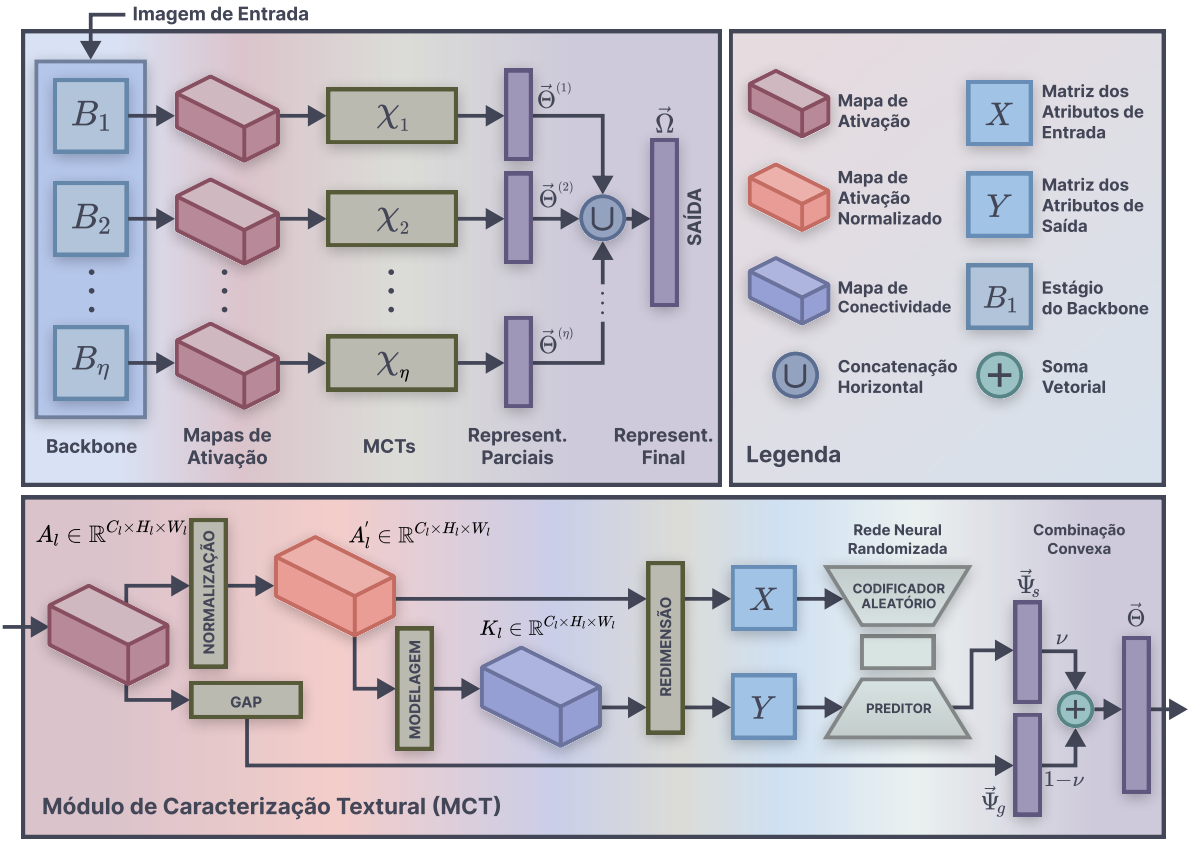
$$E^{(t)} = \{e_{x,y,z}^{i,j,k} \in E \mid w_{x,y,z}^{i,j,k} > t\} \subseteq E\,, \tag{2}$$

denota o conjunto de arestas cujo peso correspondente é estritamente maior que o valor de corte t. Desse modo, a conectividade entre os vértices e aqueles em sua vizinhança original (i.e., em G) são alteradas com base na intensidade das ativações executadas pelos diferentes filtros e em distintas regiões, assim, permitindo o surgimento de propriedades úteis as quais serão obtidas por meio de medidas topológicas, tal como o grau de saída dos vértices.

Construção da Representação. Primeiramente, apresentam-se os quatro hiperparâmetros $\mathcal{B}$, η, r e t da abordagem de representação proposta, ilustrada na Figura 33. Especificamente, $\mathcal{B} = (B_1, B_2, \ldots, B_L)$ denota um *backbone* composto por $L \in \mathbb{N}$ estágios; $\eta \in [1, L]_{\mathbb{N}}$ corresponde ao número de estágios empregados na construção da representação; $r \in \mathbb{N}$ define o raio da vizinhança considerado na modelagem do grafo; e $t \in \mathbb{R}$ representa o valor de corte utilizado para remoção das arestas culminando em alterações topológicas do grafo, e a possibilidade de indução de interações complexas, que podem caracterizar padrões subjacentes.

Desse modo, para uma dada imagem arbitrária $I \in \mathbb{R}^{3\times H\times W}$, para a construção da representação correspondente, tem-se que, primeiramente, a imagem I é redimensionada utilizando interpolação bilinear para a dimensão 224×224, que corresponde ao tamanho

Figura 33 – Ilustração da metodologia de aprendizado de representações com redes neurais randomizadas híbridas, denominada por TOCHA. Inicialmente, a imagem de entrada é fornecida ao *backbone*, de modo que as respectivas saídas dos estágios são fornecidos aos módulos de caracterização textural (MCTs) responsáveis por obterem as representações parciais de cada estágio, os quais durante a fase de agregação serão fundidos para compor a representação de textura final.



Fonte: Elaborada pelo autor.

de entrada padrão adotado por arquiteturas de redes neurais profundas. Ademais, caso a imagem de entrada esteja em nível de cinza, então o canal único dessa imagem é replicado três vezes de forma a compor uma imagem colorida. Em seguida, os *pixels* da imagem resultante do redimensionamento são normalizados no intervalo unitário e padronizados por meio das estatísticas de média e desvio padrão provenientes do conjunto de dados ImageNet (Deng et al., 2009). Além disso, por questão de simplicidade notacional, denota-se por $I' \in \mathbb{R}^{3\times224\times224}$ a imagem final após as transformações descritas. Portanto, após essa etapa de pré-processamento, a imagem I' passa a servir como entrada do *backbone*, de modo que esse produzirá como saída dos estágios os mapas de ativação $A_l \in \mathbb{R}^{C_l\times H_l\times W_l}$, em que $l \in [1, \eta]_{\mathbb{N}}$.

Em seguida, cada mapa de ativação obtido será utilizado como entrada para o módulo, denominado módulo de caracterização textural, cuja responsabilidade é a extração das representações parciais (i.e., nesse caso, aquelas obtidas para cada estágio), as quais serão posteriormente empregadas na fase de agregação para compor a representação de textura final.

Especificamente, o módulo de caracterização textural é composto por dois ramos de extração de características complementares, os quais serão apresentados detalhadamente a seguir.

- **Ramo Topológico**: Este ramo é responsável pela modelagem em grafo do mapa de ativação de entrada, seguida da extração das características topológicas decorrentes dessa construção. O propósito central consiste em empregá-las em uma rede neural randomizada cuja função é codificar, de forma aleatória, as informações topológicas extraídas nos pesos aprendidos pelo preditor. Esses pesos, em tese, carregam informações essenciais sobre a caracterização topológica do grafo e, principalmente, o relacionamento entre as intensidades das ativações e como cada vértice se relaciona com sua vizinhança. Desse modo, tais pesos são utilizados para compor a representação parcial de textura, i.e., a representação de textura decorrente do processamento do mapa de ativação.

  Para isso, esse ramo, ao receber o mapa de ativação de entrada $A_l \in \mathbb{R}^{C_l \times H_l \times W_l}$, obtém a versão correspondente $A'_l \in \mathbb{R}^{C_l \times H_l \times W_l}$, resultante da normalização das dimensões espaciais de cada canal desse mapa, conforme descrito na equação a seguir:

  $$[A'_l]_{k,:,:} = \frac{[A_l]_{k,:,:}}{\max(\|[A_l]_{k,:,:}\|_2, \epsilon)}\,, \quad k \in \{1, 2, \ldots, C_l\}\,, \tag{3}$$

  em que $\epsilon = 10^{-10}$ denota um número pequeno para prevenir divisões por zero. Desse modo, a partir de A'_l, constrói-se o grafo $G_l^{(t)}$ conforme a modelagem previamente descrita e, a partir do grafo construído, define-se um tensor tridimensional $K_l \in \mathbb{R}^{C_l \times H_l \times W_l}$, denominado mapa de conectividade, cujos elementos satisfazem a seguinte condição:

  $$[K_l]_{x,y,z} = \deg(v_{x,y,z}) \in [0, (2r+1)^3 - 1]_{\mathbb{N}}\,, \tag{4}$$

  de tal modo que, $\deg(v_{x,y,z})$ representa o grau de saída do vértice $v_{x,y,z}$, i.e., a quantidade de arestas existentes no grafo $G_l^{(t)}$ que partem de $v_{x,y,z}$.

  Assim, a partir do mapa de ativação normalizado e do mapa de conectividade obtido, esses são redimensionados em tensores bidimensionais (i.e., matrizes), com a finalidade de atuar como matrizes de entrada e saída, respectivamente, da rede neural randomizada. Dessa forma, tal rede será então empregada para a estimação dos pesos da camada de saída do preditor, os quais são ajustados de modo a forçar a predição do grau de saída dos vértices somente a partir dos valores das ativações produzidas pelo *backbone* nas posições espaciais de tais vértices. Logo, para esse fim, seja a seguinte função de redimensionamento Reshape $\in \mathbb{F}(\mathbb{R}^{C \times H \times W}, \mathbb{R}^{C \times HW})$, a qual consiste em realizar a agregação das dimensões espaciais em uma única dimensão. Consequentemente, fica claro que as matrizes de entrada e saída podem ser definidas, respectivamente, como $X_l = \text{Reshape}(A'_l)$ e $Y_l = \text{Reshape}(K_l)$, as quais pertencem ao espaço $\mathbb{R}^{C_l \times H_l W_l}$.

  Desse modo, permanece apenas a codificação aleatória das informações topológicas e das relações entre as ativações e a conectividade dos vértices com suas respectivas vizinhanças. Para isso, primeiramente, assim como nas abordagens propostas com redes

neurais randomizadas puras, realiza-se a obtenção da matriz de pesos aleatórios cuja função consiste em projetar aleatoriamente os dados de entrada em um outro espaço dimensional. Em particular, seguindo abordagens híbridas correlatas, emprega-se o mesmo gerador congruente linear utilizado no RADAM (Scabini et al., 2023) para preencher o vetor $V \in \mathbb{R}^{C_l}$, o qual é posteriormente redimensionado para as dimensões $(1, C_l)$, resultando, portanto, na matriz de pesos aleatórios $U \in \mathbb{R}^{1\times C_l}$. Observa-se que, nesse caso, seguindo também a abordagem do RADAM, utiliza-se apenas um único neurônio na camada escondida, uma vez que a utilização de múltiplos neurônios não se revelou, ainda, vantajosa. Portanto, apresenta-se, a seguir, a representação de textura resultante deste ramo, a qual representa a codificação aleatória das informações topológicas:

$$\vec{\Psi}_s^{(l)} = \text{squeeze}(\text{RNN}(X, Y; U)) \in \mathbb{R}^{C_l}\,. \tag{5}$$

- **Ramo Desordenado**: Este ramo é responsável por extrair uma componente de representação textural do tipo desordenada (do inglês, *orderless*). A principal característica desse tipo de representação consiste no descarte das posições espaciais das características visuais representadas, de modo que a representação resultante se torne mais invariante a transformações ou deformações (Lin; RoyChowdhury; Maji, 2015; Gong et al., 2014). Desse modo, esse ramo passa a contribuir com uma representação textural que atua de forma complementar ao ramo topológico, produzindo, ao final, uma representação que incorpora ambas as fontes de informação e, consequentemente, apresenta maior robustez. Portanto, para a obtenção dessa representação desordenada, a qual descarta as posições absolutas das características visuais, aplica-se a operação de agrupamento médio global (do inglês, *Global Average Pooling*) ao mapa de ativação correspondente. Matematicamente, essa operação é expressa por:

$$\vec{\Psi}_g^{(l)} = (\psi([A_l]_{1,:,:}), \psi([A_l]_{2,:,:}), \ldots, \psi([A_l]_{C_l,:,:})) \in \mathbb{R}^{C_l}\,, \tag{6}$$

em que $\psi \in \mathbb{F}(\mathbb{R}^{m\times n}, \mathbb{R})$ é uma função cuja responsabilidade é calcular a média dos valores em uma matriz e, especificamente, é definida por $\psi(X) = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}[X]_{i,j}$.

Portanto, uma vez que o mapa de ativação A_l alimenta o módulo de caracterização textural, este fornece internamente duas representações, $\vec{\Psi}_s^{(l)}$ e $\vec{\Psi}_g^{(l)}$, resultantes dos ramos topológico e desordenado, respectivamente. Diante disso, torna-se necessária a adoção de uma técnica para a composição dessas representações para fornecer como resultado do módulo e, embora existam diferentes métodos factíveis que poderiam ser empregados, os mais adequados são aqueles que preservam a dimensionalidade original de ambas as representações (deve-se notar que as duas representações possuem a mesma dimensão), de modo que técnicas como a concatenação não se mostram favoráveis. Nesse contexto, uma abordagem adequada para a fusão das representações, mantendo as condições anteriormente estabelecidas, consiste em empregar uma combinação convexa. Assim, um hiperparâmetro adicional $\nu \in [0, 1]$ passa a

indicar o peso relativo com que cada representação contribuirá na composição. Dessa forma, a representação de textura parcial correspondente ao mapa de ativação A_l é expressa por:

$$\vec{\Theta}^{(l)} = \nu\vec{\Psi}_s^{(l)} + (1-\nu)\vec{\Psi}_g^{(l)} \in \mathbb{R}^{C_l}\,. \tag{7}$$

Por fim, como tal processo é realizado para os primeiros η mapas de ativação, i.e., $A_1, A_2, \ldots, A_\eta$, obtém-se a partir do módulo de caracterização textural as representações de textura parciais $\vec{\Theta}^{(1)}, \vec{\Theta}^{(2)}, \ldots, \vec{\Theta}^{(\eta)}$. Desse modo, tais representações seguem para a fase de agregação, cuja responsabilidade consiste em compor a representação final de textura. Assim, essa representação é expressa pela concatenação horizontal das representações parciais extraídas de cada estágio, resultando, portanto, em uma representação robusta que incorpora os padrões capturados ao longo dos primeiros η estágios do *backbone* e que é expressa por:

$$\vec{\Omega} = \text{hconcat}(\{\vec{\Theta}^{(1)}, \vec{\Theta}^{(2)}, \ldots, \vec{\Theta}^{(\eta)}\}) \in \mathbb{R}^q, \tag{8}$$

em que $q = \sum_{k=1}^{\eta} C_k$ representa a dimensionalidade da representação final.

Problema da Seleção do Valor de Corte. Um dos hiperparâmetros da abordagem é o valor do corte $t \in \mathbb{R}$, que é empregado com a finalidade de realizar remoção de arestas cujo peso não é estritamente maior que t induzindo, assim, alterações topológicas e o surgimento de comportamentos complexos no grafo modelado. Embora à primeira vista pareça que a especificação desse valor seja simples, existem complicações, como por exemplo, a possível diferença de magnitude entre o valor especificado e os pesos, desde que a abordagem proposta não fornece acesso direto ao usuário aos pesos modelados, de tal maneira que, dependendo do valor de corte selecionado, esse pode induzir alterações drásticas no grafo, tornando-o não útil para a extração de informações essenciais. Um exemplo de alteração drástica consiste na seleção de um valor de corte $t \geq \max w$, i.e., um valor que seja pelo menos igual ao maior peso entre as arestas modeladas, de modo que o grafo resultante após a aplicação do corte não possuirá quaisquer arestas. Por outro lado, um segundo exemplo corresponde à escolha de $t < \min w$, caso em que nenhuma aresta será removida e, consequentemente, possivelmente nenhum comportamento complexo subjacente do grafo será capturado.

Nesse contexto, propõem-se três abordagens a serem consideradas para a seleção desse hiperparâmetro. Em primeiro lugar, pode-se manter a especificação desse hiperparâmetro de forma totalmente manual; entretanto, em virtude das complicações discutidas anteriormente, tal abordagem não se mostra vantajosa. Em segundo lugar, o hiperparâmetro ainda é especificado pelo usuário, entretanto, são aplicadas transformações estatísticas (i.e., emprego de estatísticas obtidas a partir da distribuição dos pesos) sobre o valor inicialmente definido, de modo a alcançar o valor final do hiperparâmetro. Em particular, a transformação considerada é dada por $t' = \mu + t\sigma$, em que μ e σ representam, respectivamente, a média e o desvio padrão dos pesos, enquanto t corresponde ao valor originalmente especificado. Desse modo, embora tal opção seja melhor que a primeira, essa ainda requer a especificação de

um valor de corte inicial. Em terceiro lugar, a solução considerada mais adequada consiste na seleção automática do valor de corte, de modo que esse valor deixe de ser tratado como um hiperparâmetro, ajustando-se dinamicamente em relação aos demais hiperparâmetros da abordagem e, adicionalmente, reduzindo o espaço de parâmetros a ser investigado.

Para isso, investiga-se como o emprego de distintas medidas estatísticas (i.e., média, desvio padrão, quartis etc.), obtidas a partir da distribuição dos pesos, para a definição do valor de corte, impacta a qualidade da representação de textura obtida. Empiricamente, observa-se que, embora o uso de medidas como a média e o desvio padrão, ou de transformações baseadas em ambas, alcance acurácias razoáveis, identifica-se que a utilização dos quartis se mostra mais vantajosa, uma vez que carrega uma característica essencial da distribuição e é obtida diretamente, sem a necessidade de transformações adicionais. Particularmente, existem três quartis Q_1, Q_2 e Q_3, que dividem a distribuição em quatro partes iguais e indicam que, após cada um de seus valores, encontram-se, respectivamente, 75%, 50% e 25% dos dados[1]. Dessa forma, permanece a necessidade de investigar e selecionar o quartil mais adequado.

No cenário atual, considera-se mais apropriada a seleção do terceiro quartil, uma vez que, com base no conhecimento de domínio acerca da definição dos pesos das arestas, constata-se que aproximadamente 50% dos pesos carregam informações redundantes. Isso ocorre porque, conforme discutido anteriormente, para cada aresta direcionada $e^{i,j,k}_{x,y,z}$ presente no grafo original, existe uma aresta correspondente na direção oposta $e^{x,y,z}_{i,j,k}$. Desse modo, dado que o peso de cada aresta corresponde à diferença entre as ativações associadas às posições espaciais dos vértices de saída e entrada, estabelece-se o seguinte teorema:

Teorema da Simetria dos Pesos

Teorema 5.1.1 (Simetria dos Pesos)**:** Seja $G = (V, E)$ o grafo construído a partir da saída do estágio de um *backbone*, utilizando-se a modelagem apresentada anteriormente. Assim, tem-se que se $e^{i,j,k}_{x,y,z} \in E$ com peso $w^{i,j,k}_{x,y,z} \in \mathbb{R}$, então existe a aresta correspondente $e^{x,y,z}_{i,j,k} \in E$ com peso $w^{x,y,z}_{i,j,k} = -w^{i,j,k}_{x,y,z}$.

Demonstração. Por hipótese, tem-se que $e^{i,j,k}_{x,y,z} \in E$, assim, por definição, para que tal aresta exista, essa deve satisfazer $0 < \sqrt{(x-i)^2 + (y-j)^2 + (z-k)^2} \leq r\sqrt{3}$. Nesse contexto, como a distância euclidiana é simétrica, então tem-se que:

$$\sqrt{(x-i)^2 + (y-j)^2 + (z-k)^2} = \sqrt{(i-x)^2 + (j-y)^2 + (k-z)^2}\,. \tag{9}$$

Assim, a desigualdade composta apresentada será reescrita como:

$$0 < \sqrt{(i-x)^2 + (j-y)^2 + (k-z)^2} \leq r\sqrt{3}\,, \tag{10}$$

[1] Note-se que o texto introduz o valor do quartil com relação a distribuição restante que está após o valor do quartil. Por outro lado, mais comumente, a distribuição resultante até o valor dos quartis Q_1, Q_2 e Q_3 representa 25%, 50% e 75% dos dados.

de modo que, isso implica, por definição, que $e_{i,j,k}^{x,y,z} \in E$. Assim, permanece somente demonstrar que o peso de uma aresta corresponde ao oposto do peso da outra. Dessa forma, por definição, tem-se que o peso da aresta é dado por: $w_{x,y,z}^{i,j,k} = A_{x,y,z} - A_{i,j,k}$. Consequentemente, obtém-se a seguinte relação entre os pesos dessas arestas correspondentes:

$$w_{i,j,k}^{x,y,z} = A_{i,j,k} - A_{x,y,z} = -(A_{x,y,z} - A_{i,j,k}) = -w_{x,y,z}^{i,j,k} \,. \tag{11}$$

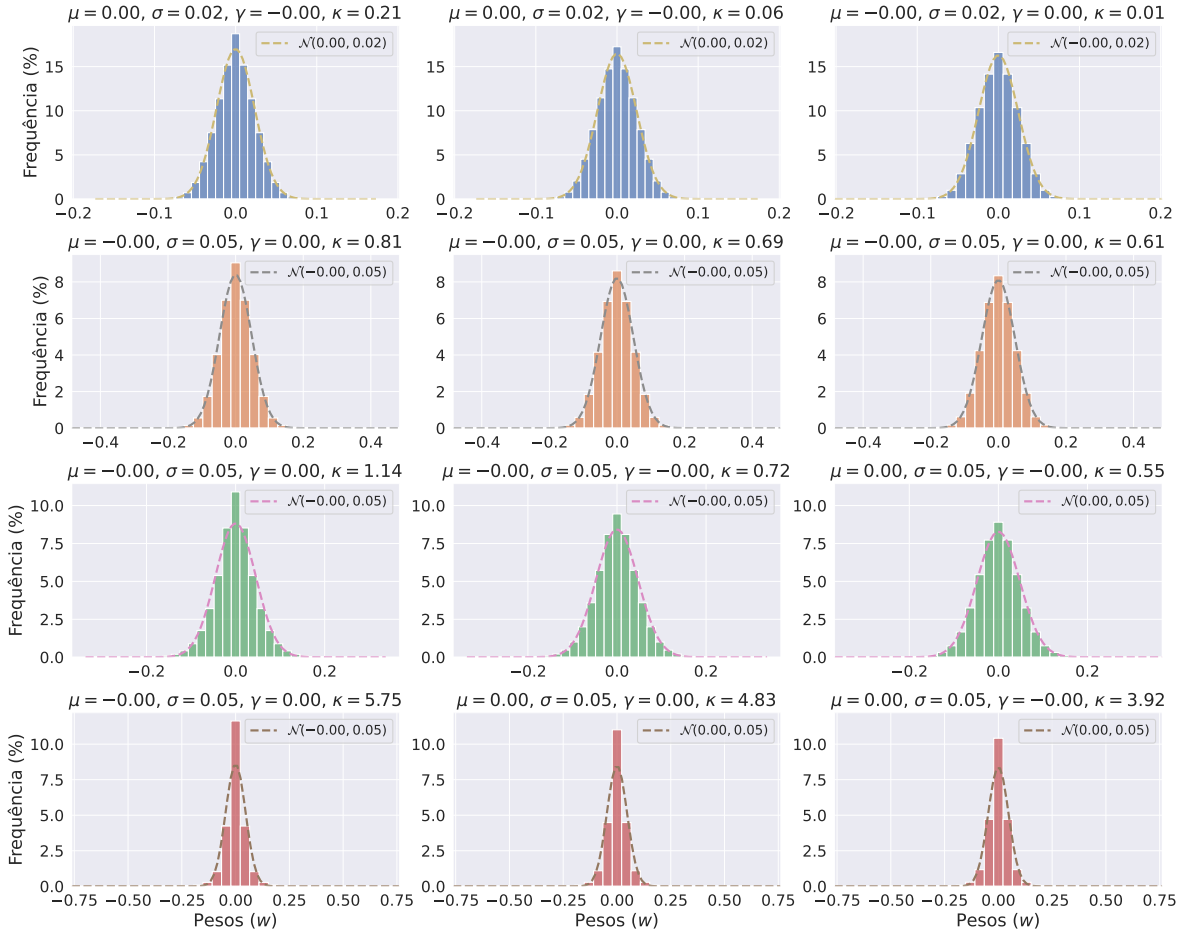
Portanto, conclui-se a demonstração do teorema. ■

Dessa forma, para cada aresta direcionada que possui um determinado peso w, existirá uma aresta correspondente na direção oposta cujo peso é $-w$. Consequentemente, 50% dos pesos passam a carregar essencialmente a mesma informação. Nesse sentido, o primeiro quartil Q_1 corta 25% da distribuição, mantendo, contudo, 25% de informação redundante. Por sua vez, o quartil Q_2, também conhecido como mediana, corta 50% da distribuição, o que, na prática, embora produza alterações topológicas no grafo, ainda não se mostra, provavelmente, capaz de induzir o surgimento de propriedades complexas relevantes, desde que apenas informação redundante é removida. Em vista disso, o terceiro quartil mostra-se o mais apropriado, de modo que, mantém apenas 25% dos maiores pesos da distribuição original.

Entretanto, surge um segundo problema, o cálculo do terceiro quartil a partir da distribuição dos pesos, o qual revela-se computacionalmente custoso. Isso ocorre porque a quantidade de pesos pode variar desde aproximadamente sete milhões (e.g., modelos menores com raio reduzido) até cerca de trezentos milhões (e.g., modelos maiores com raio ampliado). Consequentemente, o tempo de processamento exigido para computar o terceiro quartil torna-se substancial mesmo para uma única imagem, o que torna o processo de extração excessivamente custoso. Assim, a solução proposta para tal problema é realizar a aproximação do terceiro quartil a partir de uma distribuição de probabilidades conhecida, a qual possua esse valor tabelado. Nesse cenário, com a finalidade de investigar a distribuição mais adequada para abordar o problema em questão, a Figura 34 apresenta o histograma das distribuições dos pesos obtidos para todos os estágios para diferentes valores do raio da modelagem.

A partir dela, percebe-se que as distribuições dos pesos apresentadas assemelham-se à distribuição normal. Dessa forma, juntamente com os histogramas, plotam-se as distribuições normais teóricas $\mathcal{N}(\mu, \sigma^2)$ cujos parâmetros de média (μ) e desvio padrão (σ) são estimados diretamente a partir da própria distribuição de pesos. Ademais, com o objetivo de aprimorar a caracterização estatística dessas distribuições, apresentam-se, em conjunto com a média e o desvio padrão, as medidas de assimetria (γ) e curtose (κ) cuja função é caracterizar o formato da distribuição. Em particular, ao analisar tais plotagens, tem-se que a média, dentro do número de casas decimais, é muito próximo de zero. Isso sugere, de modo preliminar, que um grade número das ativações produzidas são próximas entre si. Em seguida, nota-se que para todas as distribuições de pesos apresentadas, essas apresentam um valor de assimetria de zero,

Figura 34 – Histogramas das distribuições dos pesos para o modelo ConvNeXt-T (IN-1K) para todos os estágios (dispostos em cada linha) e para três raios de modelagem (dispostos em cada coluna).



Fonte: Elaborada pelo autor.

o que é esperado, desde que para cada aresta existe uma outra correspondente com o peso oposto ao da primeira, de modo que, isso contribui com relação à simetria da distribuição.

Por fim, ao assumir que a distribuição dos pesos segue uma distribuição normal $w \sim \mathcal{N}(\mu, \sigma^2)$, cujos parâmetros μ e σ são previamente estimados, pode-se proceder à padronização dessa distribuição original, transformando-a na distribuição normal padrão $w' \sim \mathcal{N}(0, 1)$ cujo valor do terceiro quartil é tabelado e dado por $\alpha = 0,67449$. Assim, embora tal valor corresponda ao valor de corte para a distribuição padronizada, a qual não corresponde à distribuição original dos pesos, deve-se relembrar da seguinte relação entre ambas distribuições:

$$w' > \alpha \iff \frac{w - \mu}{\sigma} > \alpha \iff w > \mu + \alpha\sigma \,. \tag{12}$$

Logo, torna-se evidente que o valor de corte que representa, aproximadamente, o terceiro quartil é dado por $t = \mu + \alpha\sigma$, de modo que o cálculo desse valor revela-se substancialmente menos custoso, uma vez que a computação da média e do desvio padrão realiza-se rapidamente.

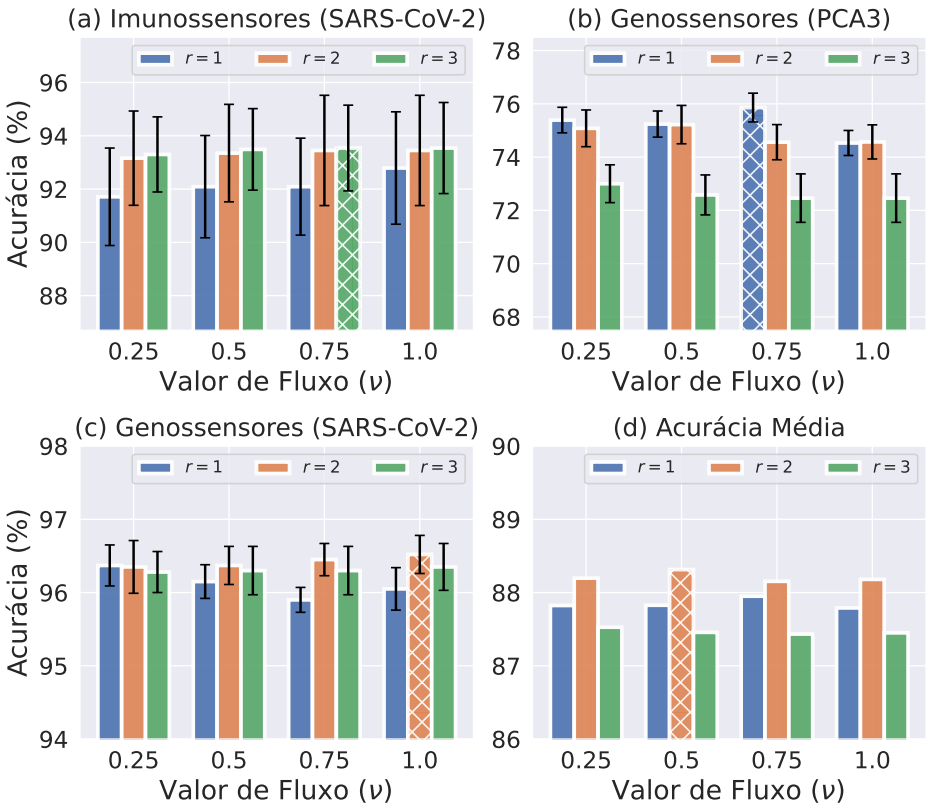
Avaliação Experimental. A abordagem híbrida proposta será submetida a dois processos de avaliação experimental, a saber: a análise do valor de fluxo (ν) e a análise de parâmetros. Para esse propósito, diferentemente das abordagens previamente apresentadas no Capítulo 4, as quais empregam conjuntos de dados de *benchmark* para conduzir as análises, a técnica híbrida proposta nesta seção é concebida diretamente com foco nos conjuntos de dados provenientes das aplicações tratadas nesta dissertação. Desse modo, nessa avaliação experimental, os conjuntos de dados de aplicações empregados são: Imunossensores (SARS-CoV-2) (Oiticica et al., 2025), Genossensores (PCA3) (Rodrigues et al., 2021) e Genossensores (SARS-CoV-2) (Soares et al., 2021). Esses conjuntos de imagens de biossensores estão detalhadamente apresentados no Capítulo 6. Por fim, uma vez apresentados os conjuntos de dados, emprega-se como métrica de desempenho a acurácia, a qual é estimada por meio do classificador *Support Vector Machine* (SVM) com *kernel* linear. No que se refere ao procedimento de validação cruzada, observa-se que, para o conjunto de imunossensores, adota-se a mesma metodologia utilizada em Oiticica et al. (2025), na qual se aplica o esquema de 5-fold estratificado com três repetições. Por outro lado, para os dois conjuntos de genossensores, emprega-se a validação cruzada 10-fold estratificada e agrupada com dez repetições.

Análise do Valor de Fluxo. O hiperparâmetro do valor de fluxo (ν) é responsável por controlar o quanto as características visuais extraídas de um ramo irá fluir em relação ao outro. Em particular, ao analisar a representação de textura parcial expressa na Equação 7, observa-se que à medida que ν tende à um, então somente características extraídas do ramo topológico serão mantidas, assim, descartando o conteúdo do ramo desordenado. Por outro lado, à medida que ν tende à zero, tem-se que somente informações do ramo desordenado serão mantidas. Isso posto, torna-se evidente esse hiperparâmetro deve ser investigado para entender como isso impactará na qualidade (i.e., desempenho) da representação de textura.

Para isso, a Figura 35 apresenta o comportamento das acurácias de classificação à medida que os raios de modelagem e valores de fluxo variam em espaços de parâmetros pré-definidos, de modo que essa análise é apresentada para todos os conjuntos de dados de aplicações por razões de completude. Em particular, com a finalidade de manter um espaço de parâmetros a ser investigado reduzido, os hiperparâmetros com relação ao *backbone* $\mathcal{B} =$ ConvNeXt-T (IN-1K) e número de estágios $\eta = 2$ foram mantidos fixos, enquanto que o raio da modelagem varia no conjunto $r \in \{1, 2, 3\}$ e o valor de fluxo $\nu \in \{\frac{1}{4}, \frac{1}{2}, \frac{3}{4}, 1\}$.

Assim, constata-se que a variação do valor de fluxo (ν) provoca diferenças sutis nas acurácias obtidas para o conjunto de imunossensores e para o conjunto de genossensores (SARS-CoV-2), enquanto variações mais pronunciadas manifestam-se no conjunto de genossensores (PCA3). Ademais, observa-se que as configurações que alcançaram as acurácias máximas em cada conjunto de dados apresentam valores distintos de fluxo, o que sugere que a representação adaptou-se às particularidades das características visuais presentes em cada domínio. Em particular, nota-se que os valores de fluxo tendem a concentrar-se próximos de

Figura 35 – Comportamentos das acurácias de classificação para três valores distintos de raios de modelagem e quatro valores de fluxo para todos os conjuntos de dados de aplicações utilizando o *backbone* ConvNeXt-L (IN-1K). A barra hachurada ▩ indica a configuração da representação com acurácia máxima.



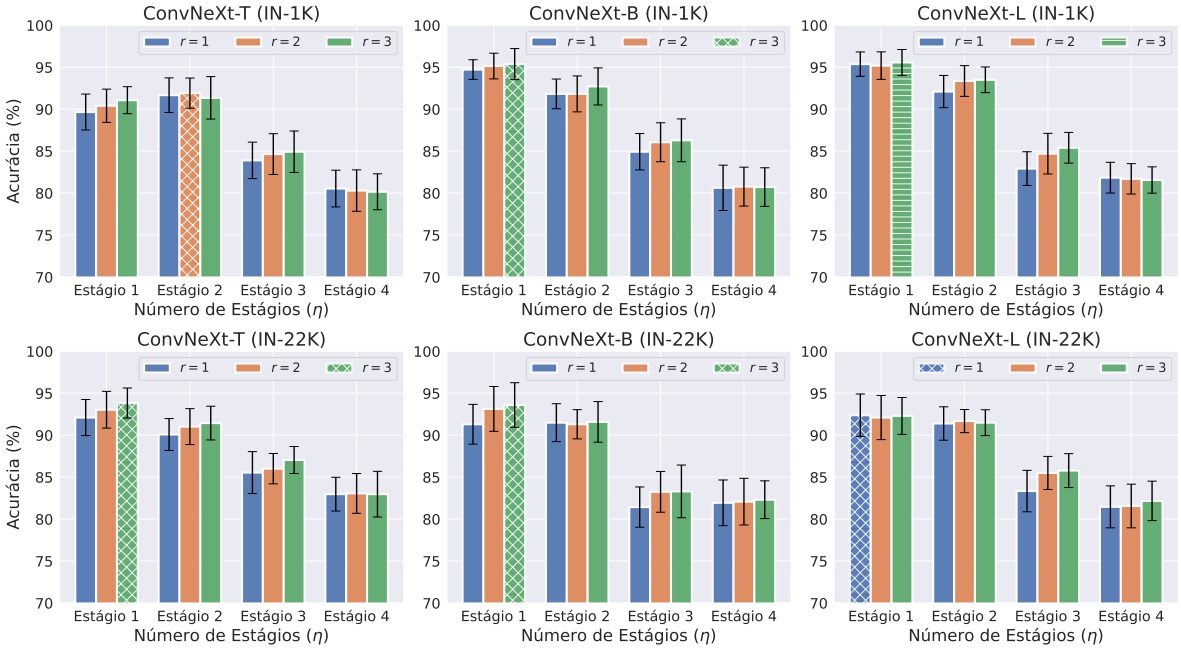
Fonte: Elaborada pelo autor.

um, indicando que se atribui maior peso às características extraídas pelo ramo topológico em relação ao ramo desordenado. Entretanto, ressalta-se que ambos os ramos são relevantes e contribuem de maneira complementar para a construção de uma representação discriminativa.

Por fim, embora os conjuntos de dados de imunossensores e genossensores (PCA3) obtiveram as melhores configurações com valor de fluxo $\nu = \frac{3}{4}$, tem-se que o conjunto de genossensores (SARS-CoV-2) obteve com $\nu = 1$, de modo que, sugere o descarte completo das características obtidas pelo ramo desordenado, mantendo apenas aquelas obtidas pelo ramo topológico. Nesse contexto, para a seleção justa do hiperparâmetro de valor de fluxo, a Figura 35(d) apresenta a acurácia média obtida para os conjuntos de dados analisados. Assim, nesse cenário, emerge como melhor configuração de representação aquela que possui valor de fluxo $\nu = \frac{1}{2}$, a qual será empregada como valor padrão no restante deste capítulo.

Análise de Parâmetros. Investiga-se o comportamento do desempenho da representação de textura proposta à medida que os hiperparâmetros $\mathcal{B}$, η e r, que representam o *backbone*, o número de estágios selecionados e o raio de modelagem, respectivamente, são alterados. A análise de parâmetros é realizada em todos os conjuntos de dados por completude, assim como

Figura 36 – Comportamento das acurácias de classificação para o conjunto de dados de imunossensores à medida que variam-se os hiperparâmetros $\mathcal{B}$, η e r da representação de textura proposta. A barra hachurada no padrão ▩ indica a configuração da representação com acurácia máxima para o *backbone* em questão, enquanto no padrão ▤ indica a melhor configuração geral.



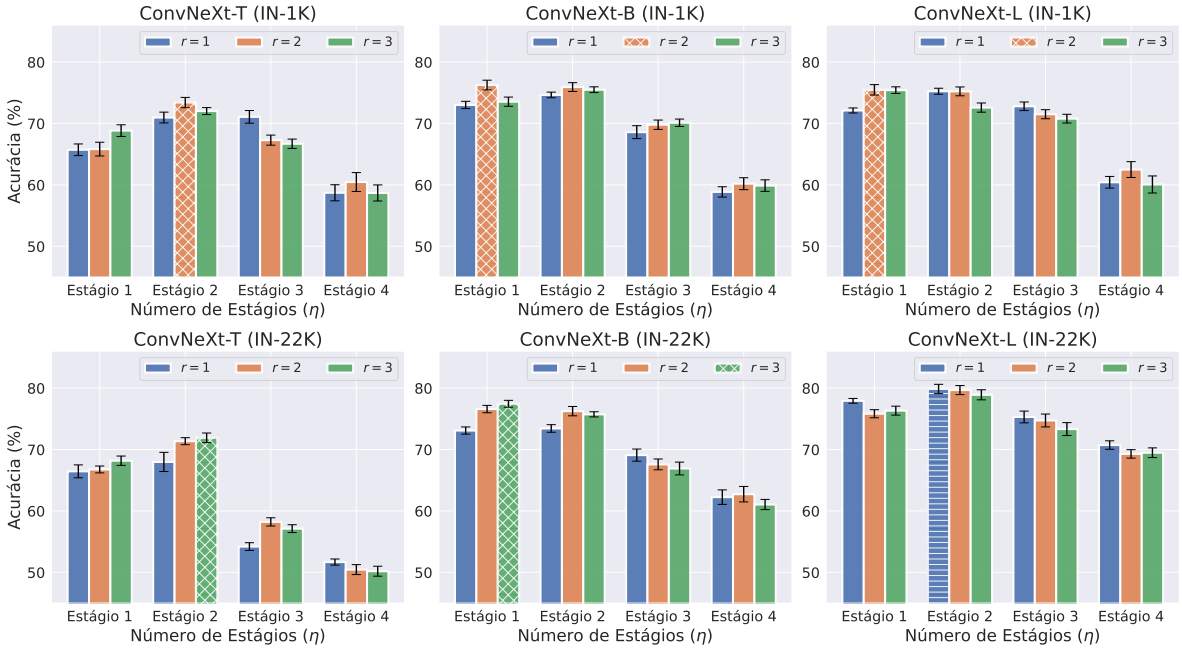
Fonte: Elaborada pelo autor.

feito para a análise do valor de fluxo anteriormente. Dessa forma, tem-se que seis modelos de *backbones* da família ConvNeXt (Liu et al., 2022) são utilizados nessa análise, os quais são: ConvNeXt-T, ConvNeXt-B e ConvNeXt-L pré-treinados, separadamente, nos conjuntos da ImageNet-1K e ImageNet-22K (Deng et al., 2009). Em seguida, com relação ao número de estágios (η), estabelece-se que esse hiperparâmetro percorre o conjunto $\eta \in \{1, 2, 3, 4\}$, uma vez que a família ConvNeXt é composta, no máximo, por quatro estágios. Por fim, quanto ao raio de modelagem (r), define-se que esse hiperparâmetro varia no conjunto $r \in \{1, 2, 3\}$, de modo a possibilitar a análise da qualidade da representação à medida que a modelagem das relações entre os vértices evolui de uma vizinhança mais local ($r = 1$) para uma vizinhança progressivamente mais ampla ($r = 3$) com 26 e 342 vizinhos[2], respectivamente.

As Figuras 36, 37 e 38 apresentam as acurácias de classificação em diferentes configurações da representação proposta para os conjuntos de dados de imunossensores, genossensores (PCA3) e genossensores (SARS-CoV-2), respectivamente. Desse modo, inicia-se a análise da representação proposta por meio da investigação do hiperparâmetro η, o qual define a quantidade de estágios empregados na construção da representação. Nesse contexto,

[2] Durante a modelagem, para um raio de vizinhança r, tem-se que para cada vértice v, o tamanho da vizinhança deste vértice será de $(2r + 1)^3 - 1$. Especificamente, $(2r + 1)^3$ refere-se à quantidade de unidades em um cubo de lado $2r + 1$ centrado em v. Em seguida, a remoção de uma unidade do total refere-se ao fato de que não existem laços no grafo modelado, i.e., o vértice v não conecta com si mesmo.

Figura 37 – Comportamento das acurácias de classificação para o conjunto de dados de genossensores (PCA3) à medida que variam-se os hiperparâmetros $\mathcal{B}$, η e r da representação de textura proposta. A barra hachurada no padrão ▩ indica a configuração da representação com acurácia máxima para o *backbone* em questão, enquanto no padrão ▤ indica a melhor configuração geral.



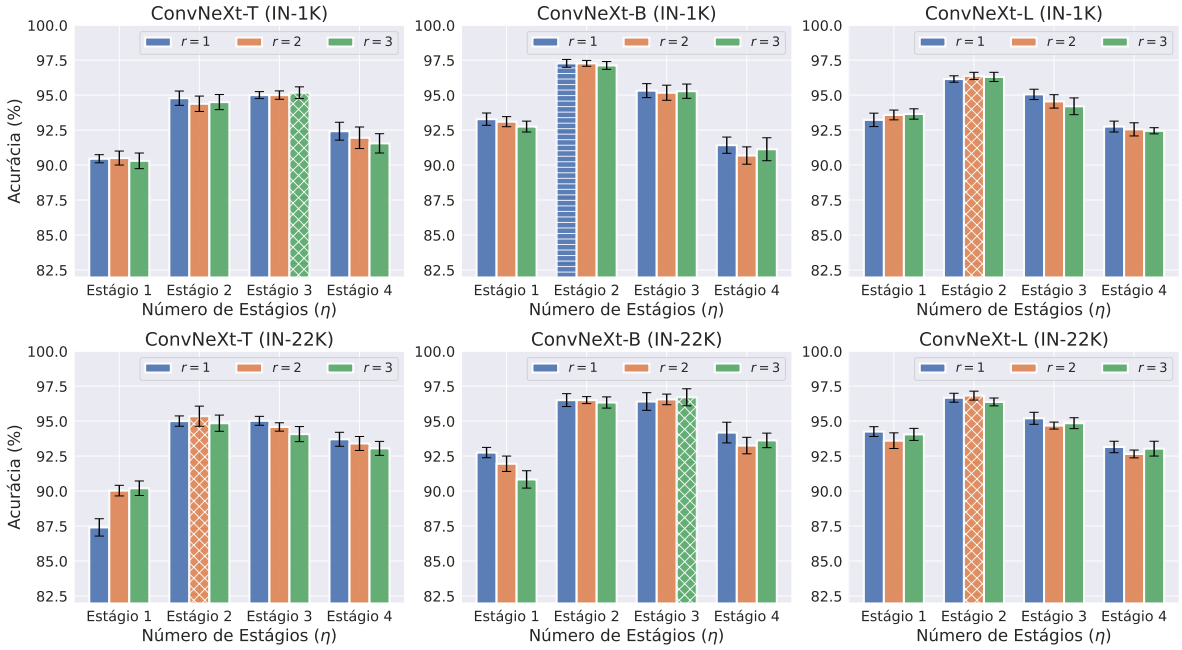
Fonte: Elaborada pelo autor.

ao examinar as figuras mencionadas, observa-se que, de maneira geral, as maiores acurácias são obtidas quando se utilizam apenas os dois primeiros estágios. Ademais, à medida que o número de estágios considerados é aumentando, especialmente quando $\eta \geq 3$, identifica-se uma redução no desempenho da representação, o que indica que a incorporação das características provenientes de estágios finais não contribuem no aumento da discriminabilidade.

Em particular, tal comportamento é esperado, uma vez que as características e os padrões extraídos em camadas finais (i.e., mais profundas) tendem a codificar informações mais específicas e de alto nível (e.g., objetos e cenas), fortemente relacionadas à tarefa considerada durante a etapa de pré-treinamento do modelo. Por outro lado, as camadas iniciais (i.e., mais rasas) são responsáveis pela extração de informações visuais mais genéricas e de baixo nível (e.g., texturas, bordas e formas), sendo essas mais interessantes para a caracterização textural. Desse modo, tornam-se evidentes as razões pelas quais o emprego das características advindas dos primeiros estágios do *backbone* é capaz de alcançar desempenhos superiores em comparação ao cenário em que todos os estágios são agregados, uma vez que, nesse contexto, o interesse concentra-se na obtenção de representações de textura robustas e não na representação ou caracterização de informações visuais sobre objetos e cenas.

Com relação ao tamanho do *backbone*, identifica-se que, de certa forma, arquiteturas maiores (i.e., ConvNeXt-B e ConvNeXt-L) obtêm maiores desempenhos comparados

Figura 38 – Comportamento das acurácias de classificação para o conjunto de dados de genossensores (SARS-CoV-2) à medida que variam-se os hiperparâmetros $\mathcal{B}$, η e r da representação de textura proposta. A barra hachurada no padrão ▩ indica a configuração da representação com acurácia máxima para o *backbone* em questão, enquanto no padrão ▤ indica a melhor configuração geral.

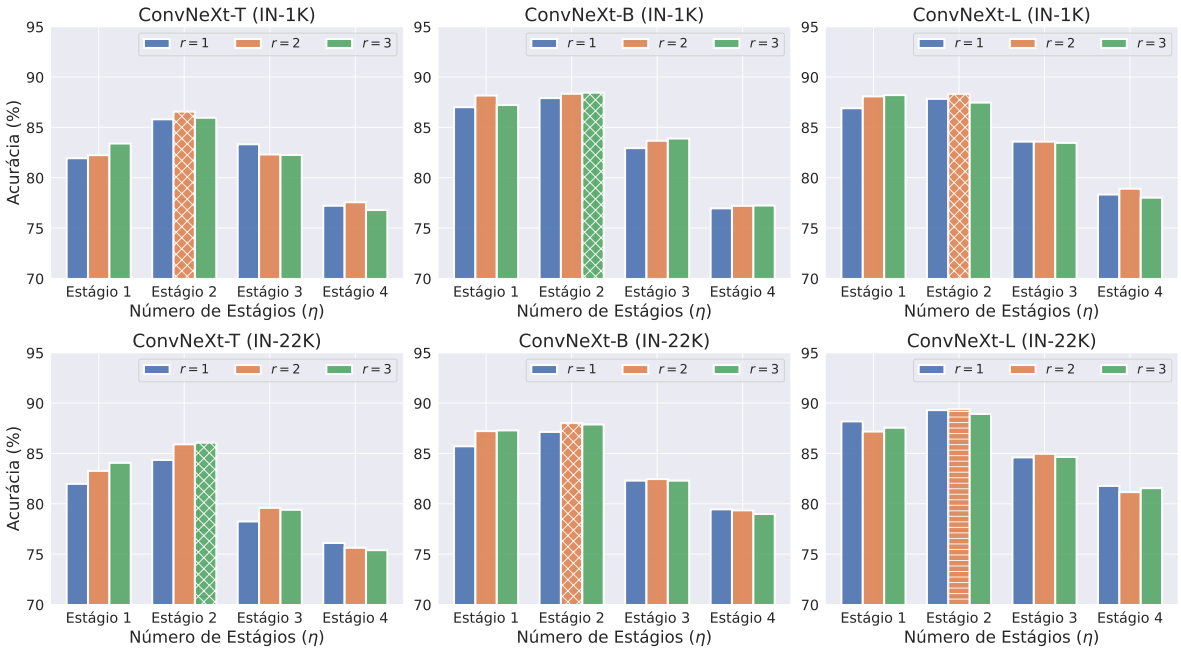


Fonte: Elaborada pelo autor.

à arquiteturas menores (i.e., ConvNeXt-T). Em particular, a título de ilustração, observa-se que as configurações que alcançaram as maiores acurácias empregaram arquiteturas maiores. Especificamente, para os conjuntos de dados de imunossensores e genossensores (PCA3), a arquitetura ConvNeXt-L mostrou-se mais adequada, enquanto que, para o conjunto de genossensores (SARS-CoV-2), o melhor desempenho foi obtido com a ConvNeXt-B. Entretanto, tal melhoria pode estar sujeita a variações não apenas em função da escolha dos demais hiperparâmetros (i.e., η e r), mas também em função do conjunto de dados de pré-treinamentos (i.e., IN-1K ou IN-22K), de modo que, se observam configurações específicas com arquiteturas menores capazes de alcançar desempenhos superiores com os demais hiperparâmetros fixos.

Nesse cenário, o maior poder discriminativo das representações obtidas por meio do emprego de arquiteturas maiores relaciona-se ao fato de que tais arquiteturas apresentam maior capacidade de aprendizagem de representações abstratas e hierárquicas, permitindo, assim, a captura de padrões mais complexos e intrínsecos presentes nas imagens observadas durante a etapa de pré-treinamento, a qual, em geral, consiste na aplicação de aprendizado supervisionado para a classificação das múltiplas categorias do conjunto ImageNet (Deng et al., 2009). Desse modo, os modelos derivados de arquiteturas com maior capacidade de aprendizado mostram-se, consequentemente, mais aptos a extraírem informações complexas das imagens observadas durante a tarefa final. Essas informações encontram-se embutidas nas

Figura 39 – Comportamento das acurácias médias de classificação à medida que variam-se os hiperparâmetros $\mathcal{B}$, η e r da representação de textura proposta. A barra hachurada no padrão ▨ indica a configuração da representação com acurácia máxima para o *backbone* em questão, enquanto no padrão ▤ indica a melhor configuração geral.



Fonte: Elaborada pelo autor.

saídas dos diferentes estágios da rede, o que viabiliza o tratamento posterior desse conteúdo para a construção de uma representação discriminativa, como feito nessa abordagem.

Quanto ao hiperparâmetro de raio de vizinhança (r), observa-se que este apresentou comportamento dependente do conjunto de dados de aplicação considerado. Por exemplo, no conjunto de imunossensores, o raio de vizinhança $r = 3$ apresentou, em grande parte, os melhores desempenhos de classificação quando comparado aos demais valores, considerando diferentes números de estágios (η) e *backbones* ($\mathcal{B}$). Dessa forma, para esse conjunto específico, a construção de representações a partir de uma análise mais ampla da vizinhança de cada vértice, a qual corresponde à modelagem das ativações produzidas por múltiplos filtros em regiões distintas, mostrou-se benéfica para a obtenção de uma representação mais discriminativa. No entanto, diferentemente, para os conjuntos de dados de genossensores (PCA3 e SARS-CoV-2), identifica-se que, embora existam configurações que alcançaram acurácias relevantes com $r \in \{2, 3\}$, as configurações que obtiveram as acurácias máximas para ambos os conjuntos foram aquelas com $r = 1$. Dessa maneira, evidencia-se que a abordagem proposta demonstra capacidade de adaptação às diferentes características visuais presentes nas imagens dos diferentes conjuntos de dados abordados.

Por fim, com relação a cada aplicação analisada, observa-se que a configuração da representação que atingiu a acurácia máxima para o conjunto de imunossensores foi aquela que

empregou ConvNeXt-L (IN-1K), com um único estágio ($\eta = 1$) e raio de modelagem $r = 3$. Analogamente, para o conjunto de genossensores (PCA3), a melhor configuração correspondeu à utilização da ConvNeXt-L (IN-22K), com dois estágios ($\eta = 2$) e raio de modelagem $r = 1$. Por sua vez, para o conjunto de genossensores (SARS-CoV-2), a configuração de maior desempenho foi obtida com a arquitetura ConvNeXt-B (IN-1K), também com dois estágios ($\eta = 2$) e raio de modelagem $r = 1$. Entretanto, para uma seleção justa da configuração da representação (i.e., dos hiperparâmetros) visando à posterior comparação com outros métodos da literatura, a Figura 39 apresenta as acurácias médias obtidas entre os conjuntos de dados analisados para as diferentes configurações investigadas. Nesse contexto, identifica-se que a configuração que alcança a maior acurácia média é aquela que emprega o modelo ConvNeXt-L (IN-22K), com dois estágios ($\eta = 2$) e raio de modelagem $r = 2$. Dessa forma, tal configuração é adotada, juntamente com seus respectivos modelos, durante a etapa de comparação.

Contribuições e Limitações. Em particular, identificam-se três contribuições centrais introduzidas pela abordagem proposta. Como primeira contribuição, destaca-se a modelagem em grafos dos mapas de ativação fornecidos em cada estágio, de modo que se tornou possível caracterizá-los por meio do emprego de medidas topológicas (e.g., grau de saída), após a realização de cortes em suas arestas com a finalidade de induzir interações complexas, as quais podem implicar na emergência de padrões subjacentes passíveis de serem capturados. Além disso, a segunda contribuição refere-se à forma como a modelagem proposta é aplicada. Especificamente, em contraste com algumas outras abordagens híbridas de representações de textura, nas quais a codificação ou processamento decorre da agregação conjunta de múltiplos estágios, o que exige redimensionamentos espaciais, a abordagem proposta realiza uma análise mais fina dos conteúdos provenientes de cada estágio, os quais são modelados individualmente na forma de grafos. Por fim, a terceira contribuição refere-se à possibilidade de selecionar uma sequência de estágios para a composição da representação final. Essa estratégia permite descartar padrões específicos associados às tarefas de pré-treinamento dos modelos empregados, conforme discutido na análise de parâmetros, possibilitando, assim, a construção de uma representação de textura mais discriminativa, que está alinhada com o objetivo da aplicação.

Entretanto, a técnica proposta apresenta algumas limitações, as quais podem ser endereçadas em trabalhos futuros com aprimoramentos metodológicos. Nesse contexto, a principal limitação relaciona-se ao fato de que a abordagem é restrita a arquiteturas convolucionais, desconsiderando, por exemplo, arquiteturas baseadas em *transformers* (e.g., ViT), as quais têm demonstrado, consistentemente, elevado desempenho em tarefas de visão computacional. Ademais, como direções para trabalhos futuros, pode-se considerar a investigação de novas estratégias de modelagem em grafos para os diferentes estágios, a utilização de outras medidas topológicas (e.g., força, intermediação etc.), bem como a proposição e/ou o emprego de outras arquiteturas de redes neurais randomizadas responsáveis pela codificação aleatória do conteúdo proveniente dos mapas de ativação e da caracterização topológica.

5.2 CONSIDERAÇÕES FINAIS

Neste capítulo, foi apresentada a abordagem híbrida, denominada por TOCHA, a qual, de modo geral, obtém a representação de textura a partir da modelagem em grafo de cada mapa de ativação, seguida de sua caracterização topológica e posterior codificação aleatória. Ademais, a metodologia e a avaliação experimental da abordagem proposta, diferentemente das técnicas puras, concentram-se exclusivamente em conjuntos de imagens de biossensores. Assim, a comparação da abordagem proposta com métodos da literatura é apresentada no Capítulo 6. Por fim, no capítulo subsequente, é demonstrada a efetividade e a contribuição das técnicas de representação de texturas propostas para a análise de imagens de biossensores com a finalidade de apoio ao diagnóstico médico.

6 APLICAÇÕES

Este capítulo dedica-se à avaliação das abordagens de representação de textura investigas e desenvolvidas nesta dissertação no âmbito prático. Em particular, os problemas abordados são de extrema importância para a sociedade e, desse modo, esse capítulo tem como finalidade principal exibir de forma clara como as contribuições científicas dos métodos propostos no campo teórico se traduzem na prática para a resolução desses problemas. Especificamente, os problemas abordados nesta dissertação referem-se aqueles relacionados ao diagnóstico de doenças por meio da análise de imagens de biossensores. Essa análise é um novo tipo de técnica ainda pouco explorada na literatura e que está continuamente demonstrando sua efetividade no campo de diagnóstico médico. Em particular, diferentemente das abordagens tradicionais que empregam imagens da própria amostra biológica para a realização do diagnóstico, tem-se que nas aplicações apresentadas, as imagens em questão são dos próprios biossensores obtidos por microscopia eletrônica ou óptica (Rodrigues et al., 2021). Portanto, a seguir, as seções subsequentes apresentarão em detalhes o problema de diagnóstico em questão, realizarão a descrição do conjunto de dados de imagens de biossensores utilizadas, e terminarão realizando a análise experimental e discussão dos resultados obtidos a partir do emprego das diferentes abordagens puras e híbridas propostas nesta dissertação.

6.1 DIAGNÓSTICO DE CÂNCER DE PRÓSTATA COM BIOMARCADOR PCA3 POR IMAGENS DE MICROSCOPIA ELETRÔNICA DE GENOSSENSORES

Segundo Gandaglia et al. (2021), o câncer de próstata é o segundo tipo de câncer mais comumente presente em homens e o quinto maior em mortalidade mundialmente. Essa doença refere-se ao crescimento anormal de células cancerígenas na glândula prostática, em que tal crescimento é geralmente assintomático, o que dificulta sua detecção precoce (Rodrigues et al., 2021). Consequentemente, quando não diagnosticado ou tratado de forma inadequada, pode-se acarretar a progressão da doença para metástases ósseas, hepáticas e linfonodais, assim, podendo levar o indivíduo acometido ao óbito (Ni et al., 2024). Atualmente, conforme estabelecido na literatura, sabe-se que os principais fatores de risco para a incidência do câncer de próstata incluem: o envelhecimento, etnia e histórico familiar. Além disso, políticas de saúde pública e fatores socioeconômicos estão relacionados à variabilidade regional da incidência e mortalidade do câncer de próstata, tornando-se um motivador essencial para a investigação de novas técnicas de detecção dessa enfermidade (Culp et al., 2020; Gandaglia et al., 2021).

Dessa forma, com a finalidade de permitir o diagnóstico do câncer de próstata de maneira rápida e precisa e o amplo acesso à sociedade por meio de testes rápidos e de baixo custo, fica notória a necessidade de técnicas biossensoriais. Além do mais, as técnicas com essas ca-

Tabela 9 – Descrição geral do conjunto de dados de genossensores para a detecção do biomarcador PCA3. Exibe-se para cada analito, a classe, a quantidade de imagens em suas respectivas resoluções espaciais bem como o seu total de imagens.

Analito (Concentração*)	Classe	Binário**	Resolução Espacial		Total
			768×1024	1536×2048	
Sequência Negativa	Negativa	Negativa	2	0	2
Branco/Controle	Controle	Negativa	3	0	3
PCA3 (1×10^{-5})	1×10^{-5}	Positiva	2	2	4
PCA3 (1×10^{-4})	1×10^{-4}	Positiva	1	3	4
PCA3 (1×10^{-3})	1×10^{-3}	Positiva	0	4	4
PCA3 (1×10^{-2})	1×10^{-2}	Positiva	3	3	6
PCA3 (1×10^{-1})	1×10^{-1}	Positiva	5	0	5
PCA3 (1×10^{0})	1×10^{0}	Positiva	4	0	4

*A unidade de concentração utilizada é μ mol L^{-1}.

**Qual classe representa durante o contexto binário.

Fonte: Adaptada de Rodrigues et al. (2021).

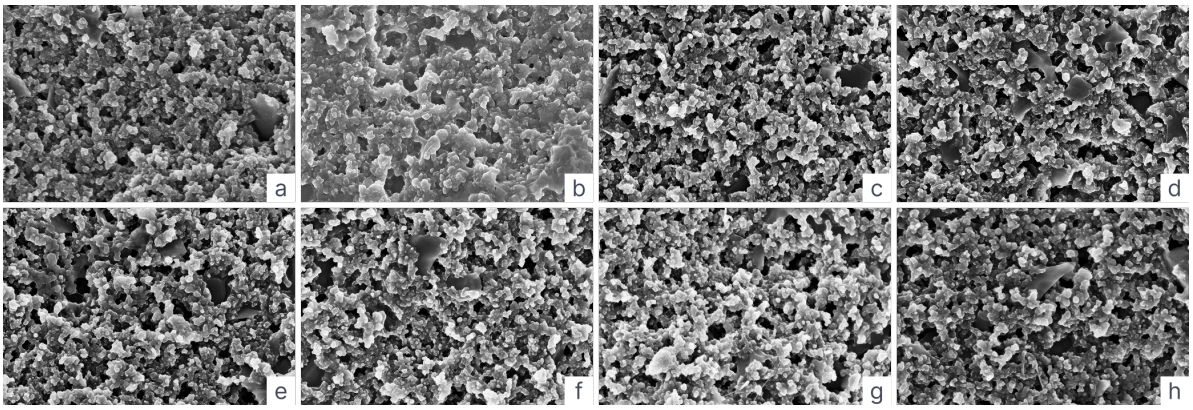
racterísticas essenciais foram amplamente evidenciadas durante a pandemia de COVID-19, em que se tornou necessário o emprego de técnicas de diagnóstico rápidas e aplicáveis diretamente nos pontos de atendimento, possibilitando, assim, diagnósticos precoces de COVID-19.

Nesse contexto, o estudo em Rodrigues et al. (2021) produziu um genossensor, biossensor em que o mecanismo de detecção é por meio de hibridização, capaz de detectar os biomarcadores PCA3, os quais são utilizados para indicar que o indivíduo possui ou desenvolverá câncer de próstata. Em particular, baseado na literatura, este estudo configura-se como o primeiro a aplicar técnicas de aprendizado de máquina e visão computacional sobre as imagens das unidades biossensoriais para fins de detecção. Logo, com a finalidade de avançar a capacidade diagnóstica por meio da análise de imagens de biossensores, serão empregadas as abordagens propostas de representação de textura para imagens em nível de cinza.

6.1.1 Descrição do Conjunto de Dados

O conjunto de dados, denominado Genossensores (PCA3), é composto por 32 imagens de unidades biossensoriais submetidas a diferentes níveis de concentração do biomarcador PCA3 variando de 10^{-5} μ mol L^{-1} até 10^{0} μ mol L^{-1}. Além disso, as amostras incluem os casos de sequência negativa (não complementar) e de medição em branco para fins de controle. Essas imagens foram adquiridas nas escalas de 200 nm e 300 nm por meio do Microscópio Eletrônico de Varredura (em inglês, *Scanning Electron Microscope* (SEM)) DSM 960 da Zeiss, Alemanha, e estão no formato em nível de cinza quantizadas em 8-bits com resoluções espaciais de 768×1024 e 1536×2048 pixels (Rodrigues et al., 2021). Por completude, a Tabela 9 apresenta, detalhadamente, os analitos, a quantidade de imagens em suas respectivas resoluções espaciais

Figura 40 – Imagens das amostras dos genossensores para o diagnóstico de câncer de próstata com o biomarcador PCA3. As imagens correspondem ao analitos (a) sequência negativa (b) branco (controle) e às diferentes concentrações em $\mu\,\mathrm{mol\,L^{-1}}$ de PCA3 (c) 1×10^{-5} (d) 1×10^{-4} (e) 1×10^{-3} (f) 1×10^{-2} (g) 1×10^{-1} e (h) 1×10^{0}.



Fonte: Elaborada pelo autor. As imagens das amostras são provenientes do conjunto de dados proposto em Rodrigues et al. (2021).

e o número total de imagens por analito, enquanto que a Figura 40 apresenta as imagens de amostras das unidades biossensoriais com cada analito.

Isso posto, derivam-se duas categorias de problemas que podem ser abordadas com o conjunto de genossensores em questão. O primeiro problema refere-se à aplicação de técnicas de análise de imagens e de aprendizado de máquina com a finalidade de distinguir os analitos associados a diferentes concentrações do biomarcador PCA3, caracterizando-os como classe positiva, daqueles que apresentam sequência não complementar ou medição em branco, os quais são caracterizados como classe negativa, portanto, resultando em um problema de classificação binário[1]. Por outro lado, o segundo problema refere-se à aplicação das mesmas técnicas em um cenário multiclasse, composto por oito classes, no qual cada analito corresponde a uma categoria distinta. Nesse contexto, o objetivo não se limita à distinção entre as amostras que apresentam ou não o biomarcador, mas abrange também a identificação das suas diferentes concentrações, tornando esse problema de classificação mais desafiador.

Por fim, observa-se a partir da Tabela 9 que a configuração atual do conjunto de dados apresenta um número reduzido de amostras. Nesse contexto, com o objetivo de ampliar o conjunto de dados e, consequentemente, viabilizar uma avaliação mais robusta das métricas de classificação obtidas durante a experimentação, as imagens serão redimensionadas para a resolução espacial de 768×1024 e, em seguida, recortadas em janelas quadradas sem sobreposição, cujas dimensões serão definidas experimentalmente. Ressalta-se que essa estratégia também foi adotada no trabalho original em Rodrigues et al. (2021) que propôs o conjunto

[1] Neste problema de classificação binário, tem-se que a classe positiva é formada pelo agrupamento dos analitos que possuem as diferentes concentrações de biomarcador PCA3 (10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1} e 10^{0} $\mu\,\mathrm{mol\,L^{-1}}$), enquanto que a classe negativa é formada pelos analitos (sequência negativa e branco).

de dados, evidenciando sua adequação para a análise das técnicas consideradas.

6.1.2 Análise Experimental e Discussões

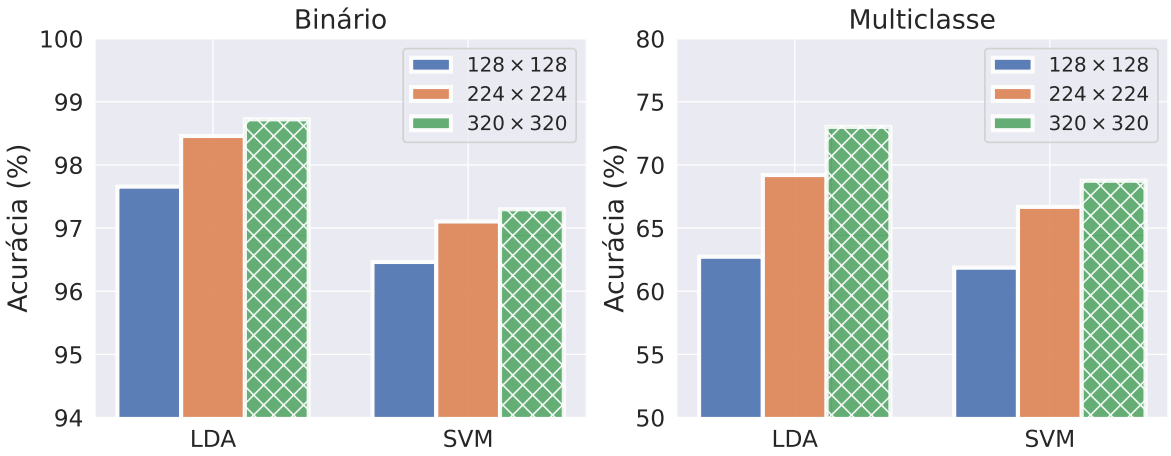
Esta subseção dedica-se à apresentação da avaliação experimental das abordagens de representação de textura propostas em nível de cinza para a detecção de câncer de próstata por meio de imagens de genossensores, bem como à discussão dos resultados obtidos por essas abordagens. Para esse fim, primeiramente, será definido o protocolo experimental utilizado nesta subseção, seguido da realização da análise do tamanho do recorte das imagens originais, que uma vez definido será utilizado como valor padrão durante o restante dessa subseção.

Protocolo Experimental. Para a avaliação dos valores de recorte e comparação entre diferentes técnicas de aprendizado de representações, a métrica de desempenho empregada será a acurácia, a qual será obtida por meio de dois classificadores, sendo eles: *Linear Discriminant Analysis* (LDA) e *Support Vector Machine* (SVM) com *kernel* linear. Além disso, a técnica de validação cruzada é a 10-*fold* estratificada e agrupada com 10 repetições. Especificamente, emprega-se a estratificação com o objetivo de preservar a mesma distribuição de classes do conjunto original, enquanto que o agrupamento é adotado para mitigar problemas de vazamento de dados. Nesse cenário, considerando que se trata de recortes de imagens de biossensores, os recortes provenientes de uma mesma imagem estarão ou no conjunto de treino ou no conjunto de teste, mas não ocorrerão simultaneamente em ambos.

Análise da Dimensão de Recorte. Investiga-se qual a dimensão da janela de recorte que permite um aumento considerável da quantidade de amostras em relação ao conjunto de dados original, bem como a obtenção de um desempenho relevante para abordar o problema em questão. Nesse contexto, são analisadas três dimensões da janela, sendo elas: 128×128, 224×224 e 320×320. Desse modo, após o recorte, os conjuntos de dados resultantes possuem 1536, 384 e 192 amostras, respectivamente. Logo, para fins de análise, considera-se o problema de classificação P (binário ou multiclasse), o tamanho da janela L e o classificador C (LDA ou SVM). Para uma dada configuração, as três melhores representações de cada abordagem proposta para imagens em nível de cinza (REE e ELGNN) são selecionadas, totalizando seis representações. Para cada uma, é computada a acurácia correspondente e, assim, o desempenho da configuração é representado pela acurácia média dessas seis representações.

A Figura 41 apresenta as acurácias médias para cada forma do problema abordado. Em particular, nota-se que para ambos os problemas de classificação (i.e., binário ou multiclasse) tem-se que as melhores acurácias foram obtidas com recortes de dimensão 320×320. Nesse contexto, embora tal dimensão origine um menor número de imagens em relação às dimensões alternativas, as maiores acurácias obtidas em recortes maiores podem estar relacionadas à conservação das características mais discriminativas em uma mesma imagem, permitindo, assim, que o modelo possa usufruir melhor das representações das amostras con-

Figura 41 – Comportamento das acurácias de classificação no conjunto de dados de Genossensores (PCA3) para o problema no contexto binário ou multiclasse, considerando o classificador escolhido e a dimensão de recorte. A barra hachurada no padrão ▨ indica a dimensão de recorte que apresentou a melhor acurácia.



Fonte: Elaborada pelo autor.

tendo tais características visuais, com a finalidade de realizar a correta predição da classe. Portanto, a partir dessa breve análise, seleciona-se 320×320 como a dimensão de recorte padrão a ser utilizada.

Resultados. A Tabela 10 apresenta a comparação entre as acurácias de classificação obtidas pelas abordagens propostas randomizadas puras em nível de cinza (REE e ELGNN) e aquelas presentes na literatura. Em particular, entre as abordagens da literatura consideradas incluem-se LBP e Fourier, que correspondem a técnicas mais tradicionais; CNTD e SSN, que baseiam-se em redes complexas; e CNRNN, LCFNN e SSR, as quais combinam redes complexas com redes neurais randomizadas. Ainda mais, a referida tabela apresenta os resultados para ambos os classificadores selecionados e para ambos os problemas de classificação investigados.

Isso posto, para o problema de classificação binária, que consiste em detectar a presença ou não do biomarcador PCA3 na amostra do biossensor, tem-se que essa é uma tarefa relativamente simples cuja observação é substanciada pelas altas acurácias obtidas. Nesse contexto, assim como exibido em Rodrigues et al. (2021) fica evidente que as imagens das amostras dos genossensores sofrem mudanças texturais suficientes, as quais são capazes de serem detectadas por auxílio das técnicas de inteligência artificial. Em particular, a abordagem proposta REE obteve a maior acurácia de 100,00% com o classificador LDA nas três configurações, superando o maior valor comparado de 99,04% obtido pela SSN. Por outro lado, com o classificador SVM, a maior acurácia de 99,64% foi obtida pela representação LCFNN.

Em relação ao problema de classificação multiclasse, que consiste não só em detectar a presença ou não do biomarcador PCA3 na amostra do biossensor mas também sua concentração, tem-se que tal tarefa é mais desafiadora, sendo tal característica exibida pelas reduções

Tabela 10 – Comparação das acurácias de classificação no conjunto de dados de Genossensores (PCA3) obtidas pelos métodos randomizados puros propostos com aqueles presentes na literatura. Em cada coluna, o maior valor está em **destaque**, o valor em azul indica o melhor resultado da literatura, e em vermelho indica resultados iguais ou superiores por parte dos métodos propostos.

	Binário		Multiclasse	
Método	LDA (↑)	SVM (↑)	LDA (↑)	SVM (↑)
Fourier	94,81±0,83	94,32±0,94	71,31±1,41	66,67±0,88
LBP	95,32±0,38	96,75±0,45	58,96±1,33	54,65±1,01
CNTD	96,58±0,72	94,46±0,65	72,17±0,88	55,44±0,67
SSN‡ (φ^6)	99,04±0,19	94,69±0,40	73,26±0,75	64,62±0,74
CNRNN$_c$	97,17±1,19	97,42±0,19	75,54±1,36	74,00±0,54
LCFNN	97,60±0,39	**99,64±0,36**	71,99±0,90	68,33±1,16
SSR$_a^{\ddagger}$	97,18±0,43	99,11±0,26	73,68±1,26	69,88±0,75
SSR$_b^{\ddagger}$	96,82±0,33	99,46±0,27	74,69±1,07	68,60±0,65
Métodos Propostos				
REE$_a$	**100,00±0,00**	96,29±0,38	**77,38±0,79**	73,65±1,33
REE$_b$	**100,00±0,00**	97,14±0,24	76,06±0,65	72,76±1,00
REE$_c$	**100,00±0,00**	96,78±0,28	76,65±1,00	**74,72±1,35**
ELGNN$_a$	97,33±0,33	97,86±0,36	69,33±1,11	64,08±0,93
ELGNN$_b$	97,67±0,57	97,58±0,49	70,00±1,31	64,06±1,36
ELGNN$_c$	97,33±0,33	98,14±0,24	68,65±0,84	63,21±1,08

‡Método originalmente colorido utilizado na versão em nível de cinza.

Fonte: Elaborada pelo autor.

de acurácia em relação ao problema binário. Entretanto, ainda assim, as abordagens propostas e comparadas são capazes de obterem resultados satisfatórios, de tal forma que, exibi-se a existência de possíveis mudanças texturais entre as diferentes concentrações nas imagens, permitindo, assim como no caso binário, sua detecção por meio de algoritmos de inteligência artificial. Em relação às abordagens e seus resultados obtidos, tem-se que a técnica proposta REE obteve as maiores acurácias de classificação de 77,38% e 74,72% com os classificadores LDA e SVM, respectivamente, superando os melhores valores comparados de 75,54% e 74,00% obtidos pela abordagem CNRNN. Nesse sentido, a abordagem proposta REE, demonstrou-se mais apta a abordar o problema multiclasse em relação às abordagens comparadas.

Em seguida, a Tabela 11 apresenta os resultados obtidos pelo método randomizado híbrido proposto, denominado por TOCHA, como também outros métodos randomizados híbridos, tal como o RADAM (Scabini et al., 2023) e o VORTEX (Scabini et al., 2025). Especificamente, as técnicas RADAM e TOCHA baseiam-se essencialmente em *backbones* convolucionais; nesse contexto, são empregados os seguintes modelos: ConvNeXt-T, ConvNeXt-B, ConvNeXt-L, ConvNeXt-XL e ConvNeXt-XXL. Por outro lado, a técnica VOR-

Tabela 11 – Comparação das acurácias de classificação no conjunto de dados de Genossensores (PCA3) obtidas pelo método randomizado híbrido proposto com aqueles presentes na literatura. Em cada coluna, o maior valor está em **destaque**, o valor em azul indica o melhor resultado da literatura, e em vermelho indica resultados iguais ou superiores por parte dos métodos propostos.

		Binário		Multiclasse	
Método	Backbone	LDA (↑)	SVM (↑)	LDA (↑)	SVM (↑)
RADAM	ConvNeXt-T	98,89±0,37	98,10±0,55	65,79±1,59	55,54±1,30
VORTEX	ViT-S/16	96,38±0,60	96,51±0,62	51,44±1,30	47,13±0,62
TOCHA	ConvNeXt-T	99,96±0,12	99,36±0,34	74,68±0,72	73,40±0,84
RADAM	ConvNeXt-T (IN-22K)	99,22±0,19	99,03±0,18	59,60±1,71	49,75±0,73
VORTEX	ViT-S/16 (IN-22K)	97,03±0,74	98,18±0,35	50,14±1,26	53,12±1,22
TOCHA	ConvNeXt-T (IN-22K)	**100,00±0,00**	**100,00±0,00**	73,88±1,53	71,35±0,57
RADAM	ConvNeXt-B	99,33±0,20	98,69±0,29	66,18±1,07	54,00±0,70
VORTEX	ViT-B/16	95,78±0,63	96,82±0,68	58,51±1,46	55,74±0,66
TOCHA	ConvNeXt-B	99,50±0,41	99,22±0,57	79,26±1,33	75,92±0,72
RADAM	ConvNeXt-B (IN-22K)	99,28±0,38	98,22±0,34	62,94±1,13	52,64±1,28
VORTEX	ViT-B/16 (IN-22K)	97,71±0,43	97,69±0,78	53,79±0,90	54,24±0,88
TOCHA	ConvNeXt-B (IN-22K)	98,87±0,49	99,58±0,00	79,01±1,10	76,24±0,76
RADAM	ConvNeXt-L	99,60±0,15	98,75±0,38	68,53±0,88	58,47±1,07
VORTEX	DeiT3-L/16	97,62±0,54	96,65±0,49	44,39±0,85	40,94±0,58
TOCHA	ConvNeXt-L	99,60±0,37	99,12±0,12	78,96±1,55	75,22±0,72
RADAM	ConvNeXt-L (IN-22K)	99,56±0,06	99,11±0,11	69,36±0,96	65,40±1,17
VORTEX	DeiT3-L/16 (IN-22K)	98,62±0,35	98,46±0,39	46,32±1,32	46,83±1,07
TOCHA	ConvNeXt-L (IN-22K)	**100,00±0,00**	98,79±0,47	**82,24±0,98**	79,67±0,74
RADAM	ConvNeXt-XL (IN-22K)	98,39±0,64	98,75±0,56	67,69±1,66	63,62±1,19
RADAM	ConvNeXt-XXL (LAION-2B)	98,97±0,53	98,58±0,73	73,29±1,49	65,97±0,75
VORTEX	ViT-H/14 (LAION-2B)	98,54±0,32	97,17±0,47	55,38±1,26	50,24±1,00
TOCHA	ConvNeXt-XL (IN-22K)	99,28±0,38	99,88±0,19	80,28±1,49	**80,93±0,50**
TOCHA	ConvNeXt-XXL (LAION-2B)	99,28±0,38	99,01±0,60	79,75±1,12	80,00±0,93

*TOCHA está com a configuração selecionada no Capítulo 5, exceto com a variação dos *backbones*.

Fonte: Elaborada pelo autor.

TEX baseia-se em *Vision Transformers* (ViTs); assim, são utilizados os *backbones* ViT-S/16, ViT-B/16, DeiT3-L/16 e ViT-H/14. Esses modelos empregados são pré-treinados nos conjuntos ImageNet-1K (IN-1K), ImageNet-22K (IN-22K) ou LAION-2B[2].

A partir dessa tabela, nota-se que a abordagem proposta TOCHA supera, consistentemente, os resultados das abordagens comparadas, assim, evidenciando a discriminabilidade da representação de textura proposta aplicada no conjunto de dados abordado. Em parti-

[2] Durante a análise comparativa dos resultados ficará explícito o conjunto de pré-treinamento do modelo em questão. Entretanto, quando não especificado, o conjunto de pré-treinamento é a ImageNet-1K (IN-1K).

cular, no contexto de classificação binária, tem-se que TOCHA (ConvNeXt-T IN-22K)[3] alcança as acurácias de 100,00% em ambos classificadores ultrapassando as melhores acurácias prévias de 99,60% com LDA pelo RADAM (ConvNeXt-L) e 99,11% com SVM pelo RADAM (ConvNeXt-L IN-22K). Por outro lado, em um cenário mais desafiador, representado pelo contexto multiclasse, tem-se que TOCHA (ConvNeXt-L IN-22K) obtém com LDA a maior acurácia de 82,24%, assim, superando em 8,95% o melhor resultado comparado de 73,29% obtido por RADAM (ConvNeXt-XXL LAION-2B) e, da mesma forma, tem-se que TOCHA (ConvNeXt-XL IN-22K) alcança com SVM a maior acurácia de 80,93%, de modo que, ultrapassa em 14,96% a melhor acurácia comparada de 65,97% obtida por RADAM (ConvNeXt-XXL LAION-2B).

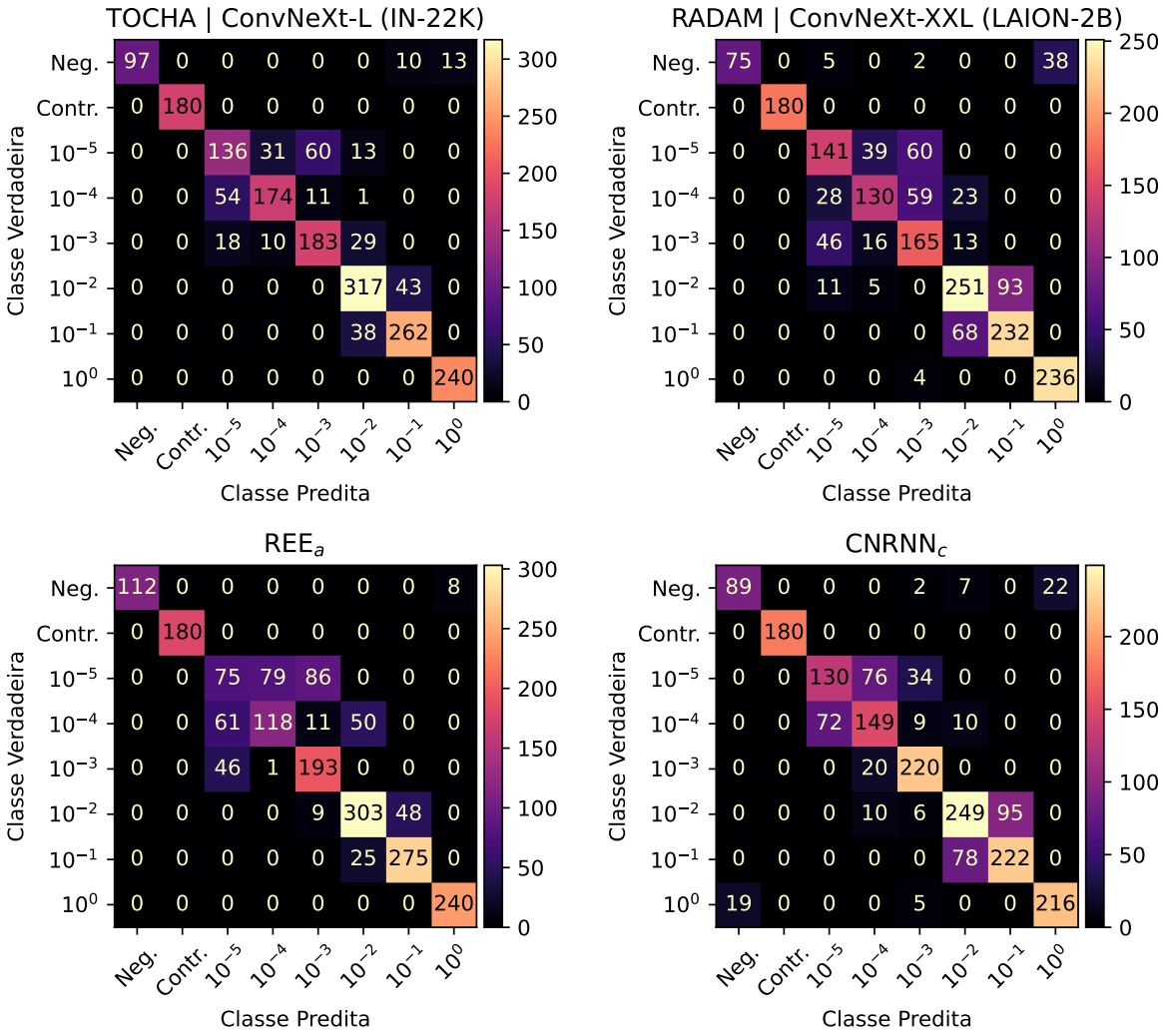
Além disso, ao comparar os resultados obtidos nas Tabelas 10 e 11, as quais se referem, de modo geral, às abordagens tradicionais, puras e híbridas empregadas, observa-se que, no contexto de classificação binária, as representações da abordagem REE e a representação TOCHA (ConvNeXt-T IN-22K) alcançaram as maiores acurácias, ambas de 100%, quando associadas ao classificador LDA. Por outro lado, no cenário multiclasse, a abordagem TOCHA (ConvNeXt-L IN-22K) obteve a maior acurácia geral, de 82,24%, com o mesmo classificador, representando, assim, um aumento de 6,70% em relação à melhor acurácia prévia dentre os métodos da literatura, de 75,54%, alcançada pela representação de textura CNRNN_c.

Ainda, a fim de complementar tal análise, a Figura 42 apresenta, na primeira linha, as matrizes de confusão das representações híbridas TOCHA (ConvNeXt-L IN-22K) e RADAM (ConvNeXt-XXL LAION-2B), as quais alcançaram as melhores acurácias atual e prévia, respectivamente, com o classificador LDA, entre os métodos híbridos. Na sequência, de forma análoga, a segunda linha apresenta as representações REE_a e CNRNN_c, representativas dos métodos puros. Além disso, deve-se notar que os valores nas matrizes de confusão correspondem às predições agregadas para as dez repetições do protocolo de validação cruzada. Assim, em primeiro lugar, observa-se que as representações TOCHA (ConvNeXt-L IN-22K) e REE_a distinguiram completamente a classe de controle e a classe associada à concentração $1{\times}10^{0}$ $\mu\,\text{mol}\,\text{L}^{-1}$ das demais classes. Por outro lado, as representações da literatura RADAM (ConvNeXt-XXL LAION-2B) e CNRNN_c apresentaram distinção completa somente para a classe de controle. Ainda mais, as duas representações propostas apresentam uma confusão consideravelmente menor entre as classes associadas às concentrações $1{\times}10^{-2}$ e $1{\times}10^{-1}$ $\mu\,\text{mol}\,\text{L}^{-1}$, enquanto as abordagens comparadas exibem maior grau de confusão entre essas classes. Em última análise, observa-se que, de modo geral, as representações apresentaram maiores taxas de confusão entre as concentrações mais baixas, evidenciando, assim, a necessidade recorrente de desenvolvimento de novas técnicas de representação de textura.

Por fim, a partir dos resultados apresentados, esses sustentam o desempenho superior das abordagens propostas em relação às abordagens comparadas. Dessa forma, as abordagens investigadas nesta dissertação, focadas na extração e caracterização do conteúdo microtex-

[3] Entre parênteses é fornecido o identificador do *backbone* utilizado na abordagem randomizada híbrida.

Figura 42 – Matrizes de confusão das representações aplicadas ao problema de diagnóstico de câncer de próstata com imagens de genossensores. O cenário em questão é multiclasse sob o classificador LDA. Cada linha apresenta as representações com melhor acurácia atual e prévia, respectivamente. A primeira linha corresponde aos métodos híbridos, enquanto a segunda corresponde aos métodos puros.



Fonte: Elaborada pelo autor.

tural presente nas imagens dos genossensores, representam um passo fundamental no avanço do diagnóstico de câncer de próstata auxiliado por técnicas de visão computacional, principalmente, aquelas auxiliadas por técnicas de representação textural. As técnicas desenvolvidas, munidas de avanços metodológicos e de desempenho, podem permitir a detecção precoce do câncer de próstata, assim, aumentando as chances de recuperação dos indivíduos acometidos por essa doença. Além de tais avanços, cita-se a adaptabilidade das abordagens investigadas em relação aos recursos computacionais disponíveis (i.e., poder de processamento, memória etc.). Especificamente, os métodos propostos são rápidos e possuem baixo custo computacional, principalmente no que diz respeito às metodologias puras. Entretanto, caso haja maior disponibilidade de recursos computacionais, podem ser empregados *backbones* de maior

porte associados à abordagem híbrida proposta, visando à obtenção de representações mais discriminativas, conforme evidenciado nas análises realizadas.

6.2 DIAGNÓSTICO DO VÍRUS SARS-COV-2 POR IMAGENS DE MICROSCOPIA ELETRÔNICA DE GENOSSENSORES

A pandemia de COVID-19 vivenciada nos últimos anos demonstrou, como discutido, a necessidade de testes de diagnóstico rápidos, com baixo custo e fáceis de usar. Essas características essenciais permitem: a diminuição do tempo de resposta para o diagnóstico de SARS-CoV-2; a produção em massa dos testes, assim fornecendo mais acessibilidade à sociedade; e a menor demanda de profissionais especializados para a sua realização (Mattioli et al., 2020; Soares et al., 2021). Diante desse cenário, o estudo em Soares et al. (2021) desenvolveu e investigou um genossensor, munido das características citadas previamente, capaz de detectar a presença de ssDNA (do inglês, *single-stranded* DNA) de SARS-CoV-2.

Especificamente, no estudo em questão, os autores empregaram dentre as distintas técnicas físico-químicas de detecção (e.g., espectroscopia de impedância eletroquímica), metodologias de visão computacional, particularmente, empregando abordagens de representação de textura. Dessa forma, assim como no estudo apresentado em Rodrigues et al. (2021), a investigação obteve um conjunto de imagens dos genossensores por meio de microscopia eletrônica. Entretanto, diferentemente da aplicação anterior que obteve imagens com ampliação de 50.000×, os autores forneceram um passo fundamental na literatura pelo fato de empregarem um aumento de 1.000×, que representa o limite superior de microscópios ópticos. Assim, o objetivo em questão é depender cada vez menos de microscópios eletrônicos custosos com a finalidade de substituí-los pelo uso de microscópios ópticos e até câmeras de *smartphones*, permitindo sua adoção em diferentes cenários. Portanto, com a finalidade de melhorar o diagnóstico de SARS-CoV-2 por meio de técnicas de visão computacional. Nesta seção, serão adotadas as abordagens propostas de representações de imagens de textura em nível de cinza.

6.2.1 Descrição do Conjunto de Dados

O conjunto de dados adotado nesta aplicação, denominado por Genossensores (SARS-CoV-2), é composto por 200 imagens obtidas por microscopia eletrônica de varredura (SEM), utilizando o microscópio eletrônico Zeiss-LEO-440, equipado com o detector 7060 (Oxford Instruments), e com aumento de 1000× (Soares et al., 2021). As imagens foram adquiridas em nível de cinza com resolução espacial de 712×1024 pixels e quantizadas em 8 bits. Ainda mais, essas imagens, a depender do analito, são agrupadas em nove classes, sendo elas: amostras de sequência negativa, interferentes (HPV16 e PCA3) e sete concentrações de ssDNA de SARS-CoV-2 (1×10^{-18}; 1×10^{-16}; 1×10^{-14}; 1×10^{-12}; 1×10^{-10}; 1×10^{-8} e 1×10^{-6} mol L^{-1}). Além disso, a Tabela 12 apresenta uma descrição geral do conjunto de dados, exibindo os analitos,

Tabela 12 – Descrição geral do conjunto de dados de genossensores para a detecção de SARS-CoV-2. Exibe-se para cada analito, a classe em ambos contextos de classificação e o seu número de amostras.

Analito (Concentração*)	Classe	Binário**	# Amostras
Sequência Negativa	Negativa	Negativa	30
HPV16	Interferente	Negativa	10
PCA3	Interferente	Negativa	10
SARS-CoV-2 (1×10^{-6})	1×10^{-6}	Positiva	30
SARS-CoV-2 (1×10^{-8})	1×10^{-8}	Positiva	20
SARS-CoV-2 (1×10^{-10})	1×10^{-10}	Positiva	20
SARS-CoV-2 (1×10^{-12})	1×10^{-12}	Positiva	20
SARS-CoV-2 (1×10^{-14})	1×10^{-14}	Positiva	20
SARS-CoV-2 (1×10^{-16})	1×10^{-16}	Positiva	20
SARS-CoV-2 (1×10^{-18})	1×10^{-18}	Positiva	20

*A unidade de concentração utilizada é mol L^{-1}.

**Qual classe representa durante o contexto binário.

Fonte: Adaptada de Soares et al. (2021).

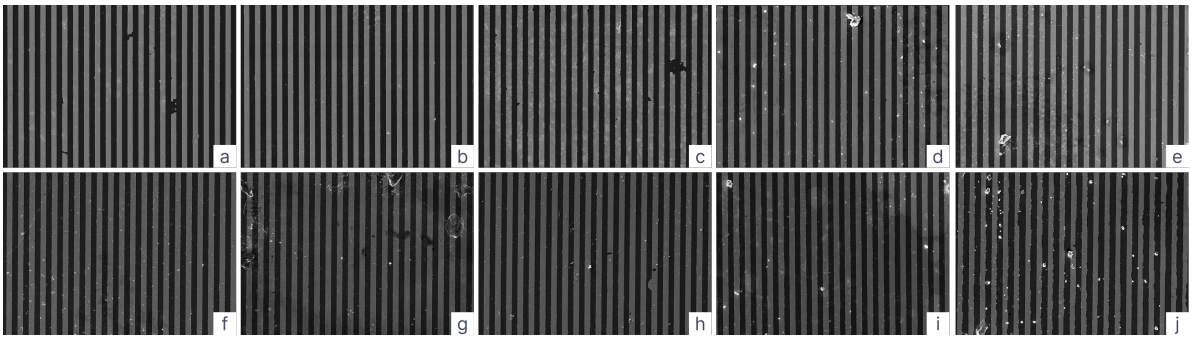
as respectivas classes e o número de amostras associadas a cada analito. Por outro lado, a Figura 43 ilustra as amostras representativas dos biossensores para cada analito avaliado.

Nesse contexto, assim como realizado na subseção anterior, são considerados dois problemas de classificação. Em primeiro lugar, adota-se o cenário binário, no qual o objetivo consiste em identificar as amostras negativas, formadas pelo agrupamento das amostras de sequência negativa e interferentes, e as amostras positivas construídas pelo agrupamento das classes associadas a diferentes concentrações de ssDNA de SARS-CoV-2. Em segundo lugar, considera-se o problema multiclasse cujo objetivo é identificar as classes de sequência negativa, interferentes e as diferentes concentrações de ssDNA de SARS-CoV-2. Portanto, de modo similar, tem-se que no problema multiclasse, investiga-se também indiretamente a capacidade das técnicas de representação posteriormente abordadas de capturarem não só a presença de ssDNA de SARS-CoV-2 como também sua respectiva concentração na amostra. Por fim, embora o conjunto de dados de genossensores para SARS-CoV-2 possua uma quantidade de imagens consideravelmente maior em relação ao conjunto de dados de genossensores para PCA3 anteriormente abordado, tem-se que, para esta aplicação, também será realizado o recorte sem sobreposição das imagens com a finalidade de aumentar a quantidade de dados do conjunto original para uma avaliação mais robusta das técnicas.

6.2.2 Análise Experimental e Discussões

Esta subseção será responsável por três etapas essenciais para a análise das técnicas propostas. Essas etapas consistem da definição do protocolo experimental empregado para

Figura 43 – Imagens das amostras dos genossensores para o diagnóstico de SARS-CoV-2. As imagens correspondem aos analitos (a) sequência negativa (b) HPV16 (c) PCA3 e às diferentes concentrações em mol L^{-1} de ssDNA de SARS-CoV-2 (d) 1×10^{-6} (e) 1×10^{-8} (f) 1×10^{-10} (g) 1×10^{-12} (h) 1×10^{-14} (i) 1×10^{-16} (j) 1×10^{-18}.



Fonte: Elaborada pelo autor. As imagens das amostras são provenientes do conjunto de dados proposto em Soares et al. (2021).

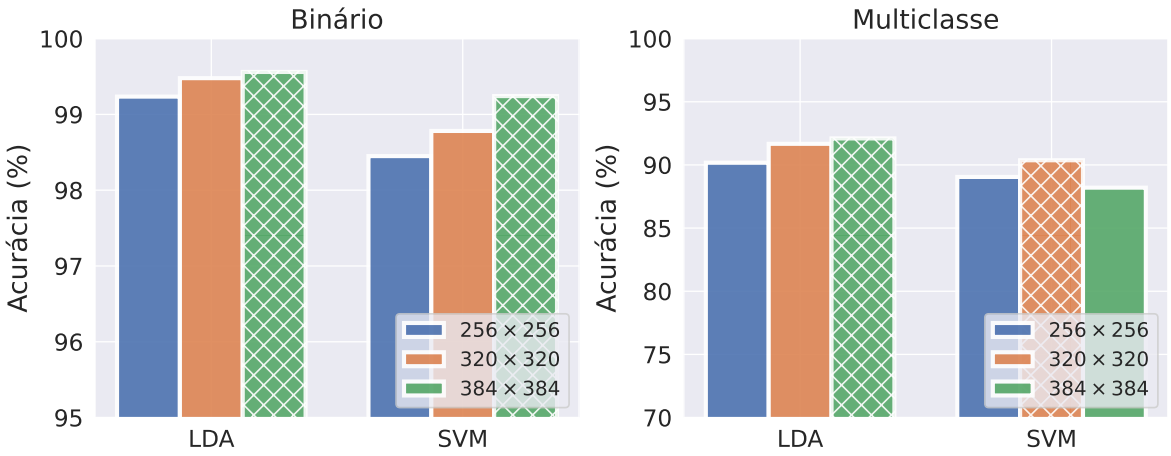
a obtenção dos resultados, a análise da dimensão de recorte das imagens originais com a finalidade de aumento de dados e, por fim, a realização da comparação dos resultados obtidos para as técnicas propostas e aquelas presentes na literatura seguida da discussão dos resultados.

Protocolo Experimental. A métrica empregue na avaliação será a acurácia cujo valor é obtido pelos classificadores LDA e SVM. Ainda mais, a estratégia de validação cruzada empregada é a 10-*fold* estratificada e agrupada com 10 repetições. Logo, assim como realizado na aplicação de genossensores para PCA3, a estratificação é realizada para preservar a proporção original das classes, enquanto que o agrupamento é realizado para prevenir vazamento de dados, desde que as imagens em questão são provenientes de recortes das imagens originais.

Análise da Dimensão de Recorte. A resolução das imagens originais é 712×1024 pixels. Dessa forma, com a finalidade de balancear o acréscimo da quantidade de amostras no conjunto de dados recortado, e a preservação de características visuais mais discriminativas nas imagens recortadas por meio da seleção da dimensão do recorte, então serão selecionadas 256×256, 320×320 e 384×384 como as dimensões de recorte a serem analisadas. Assim, após o recorte, para cada dimensão, o conjunto possuirá 1600, 1200 e 400 amostras, respectivamente. Logo, para a análise do tamanho da janela de recorte, utiliza-se a mesma metodologia adotada na Subseção 6.1.2 em que o valor representativo é dado pela média das seis representações.

A Figura 44 apresenta as acurácias médias obtidas dessas seis representações para cada configuração. A partir dessa figura, observa-se que consistentemente as maiores acurácias foram obtidas com a dimensão 384×384, enquanto que a dimensão 320×320 obteve a melhor acurácia apenas no contexto de classificação multiclasse com o classificador SVM. A característica de preferência com relação a maior resolução pode estar relacionada a preservação das características visuais mais discriminativas, corroborando com a mesma análise realizada na seção anterior. Entretanto, diferentemente da aplicação de genossensores para detecção

Figura 44 – Comportamento das acurácias de classificação no conjunto de dados de Genossensores (SARS-CoV-2) para o problema no contexto binário ou multiclasse, considerando o classificador escolhido e a dimensão de recorte. A barra hachurada no padrão ▨ indica a dimensão de recorte que apresentou a melhor acurácia.



Fonte: Elaborada pelo autor.

de PCA3, tem-se que as imagens desta aplicação foram mais robustas aos recortes, desde que as diferenças de acurácias são mais sutis em relação à aplicação anterior. Apesar disso, a dimensão de recorte de 384×384 pixels será empregue durante o restante desta seção.

Resultados. Primeiramente, inicia-se a análise e discussão dos resultados obtidos pelas técnicas puras propostas para representação de texturas em nível de cinza e pelas técnicas da literatura. Especificamente, as técnicas propostas são: REE e ELGNN, enquanto que as técnicas da literatura comparadas são: Fourier, LBP, CNTD, SSN, CNRNN, LCFNN e SSR.

Com relação ao contexto de classificação binário, observa-se a partir da Tabela 13 que, sob o classificador LDA, a melhor acurácia obtida foi de 99,83% pela representação SSR_a, enquanto que sob o classificador SVM, a acurácia máxima foi de 100,00% obtida pela representação CNRNN_c. Entretanto, embora essas foram as representações com maiores acurácias, tem-se que são ínfimas as diferenças de acurácias entre essas e aquelas obtidas pelas representações de textura propostas. Por exemplo, a representação REE_c obteve a acurácia de 99,75% com o classificador LDA, exibindo uma diferença de apenas 0,08% com relação à SSR_a, enquanto que a representação ELGNN_b obteve a acurácia de 99,50% com o classificador SVM, indicando uma diferença sutil de 0,50% em relação à CNRNN_c. Nesse contexto, mesmo não apresentando desempenhos superiores, as técnicas propostas apresentaram resultados competitivos. Entretanto, devido à saturação dos resultados nesse contexto de classificação (i.e., com a maioria dos valores próximos de 100%), a análise da superioridade dos métodos é mais adequadamente evidenciada no problema de classificação multiclasse apresentado a seguir.

Em relação ao problema multiclasse, a Tabela 13 evidencia de forma clara que as abordagens puras propostas nesta dissertação são aquelas que obtiveram as melhores acurácias.

Tabela 13 – Comparação das acurácias de classificação no conjunto de dados de Genossensores (SARS-CoV-2) obtidas pelos métodos propostos com aqueles presentes na literatura. Em cada coluna, o maior valor está em **destaque**, o valor em azul indica o melhor resultado da literatura, e em vermelho indica resultados iguais ou superiores por parte dos métodos propostos.

	Binário		Multiclasse	
Método	LDA (↑)	SVM (↑)	LDA (↑)	SVM (↑)
Fourier	95,42±0,25	96,12±2,61	38,55±0,72	53,35±2,08
LBP	90,85±0,36	94,35±0,57	70,05±0,71	84,53±0,45
CNTD	98,77±0,31	97,75±0,30	77,77±0,71	77,05±1,04
SSN‡ (φ^6)	99,47±0,17	98,55±0,22	75,10±0,93	57,70±1,19
CNRNN$_c$	99,78±0,08	**100,00±0,00**	87,95±0,43	83,13±0,76
LCFNN	98,62±0,44	99,40±0,20	86,23±0,79	87,10±0,73
SSR$_a^‡$	**99,83±0,11**	97,95±0,40	88,00±0,88	78,58±0,58
SSR$_b^‡$	99,65±0,12	98,17±0,40	89,53±0,68	78,58±0,57
Métodos Propostos				
REE$_a$	99,50±0,00	99,25±0,32	90,75±0,43	86,78±0,75
REE$_b$	99,57±0,16	99,20±0,22	90,15±0,48	86,30±0,94
REE$_c$	99,75±0,11	99,22±0,28	89,92±0,54	85,68±0,72
ELGNN$_a$	99,47±0,07	98,83±0,27	93,55±0,35	89,20±0,63
ELGNN$_b$	99,35±0,12	99,50±0,00	**94,95±0,58**	89,60±0,50
ELGNN$_c$	99,72±0,13	99,45±0,10	93,27±0,45	**91,65±0,61**

‡Método originalmente colorido utilizado na versão em nível de cinza.

Fonte: Elaborada pelo autor.

Em particular, a representação ELGNN$_b$ obteve o maior desempenho de 94,95% sob o classificador LDA, superando em 5,42% a maior acurácia prévia de 89,53% obtida pela SSR$_b$, correspondendo a um acréscimo de aproximadamente 21 imagens corretamente classificadas. Por outro lado, sob o classificador SVM, a maior acurácia de 91,65% foi obtida pela representação ELGNN$_c$ com o classificador SVM, ultrapassando em 4,55% o maior desempenho prévio de 87,10% obtido pela LCFNN, indicando um aumento de aproximadamente 18 imagens corretamente classificadas. Além disso, sob o classificador LDA, nota-se que ambas as abordagens de representação de textura propostas superaram a melhor acurácia prévia em todas as configurações de representação, e sob o classificador SVM, embora apenas as representações da técnica ELGNN superem a melhor acurácia prévia, tem-se que as representações de REE ultrapassaram todos os métodos comparados, exceto a representação de textura LCFNN.

Desse modo, os resultados obtidos substanciam a robustez e a discriminabilidade das abordagens puras desenvolvidas neste trabalho. Assim, novamente, o foco na caracterização do conteúdo microtextural presente nas imagens dos genossensores permite que sejam capturadas com maior facilidade as sutis características visuais presentes nas amostras com diferentes

Tabela 14 – Comparação das acurácias de classificação no conjunto de dados de Genossensores (SARS-CoV-2) obtidas pelo método randomizado híbrido proposto com aqueles presentes na literatura. Em cada coluna, o maior valor está em **destaque**, o valor em azul indica o melhor resultado da literatura, e em vermelho indica resultados iguais ou superiores por parte dos métodos propostos.

		Binário		Multiclasse	
Método	Backbone	LDA (↑)	SVM (↑)	LDA (↑)	SVM (↑)
RADAM	ConvNeXt-T	99,65±0,12	99,50±0,22	94,55±0,47	92,20±0,51
VORTEX	ViT-S/16	99,42±0,11	99,25±0,16	86.17±0,70	82,75±0,78
TOCHA	ConvNeXt-T	99,80±0,10	99,98±0,08	96,03±0,39	94,38±0,55
RADAM	ConvNeXt-T (IN-22K)	99,10±0,25	99,98±0,08	95,05±0,37	92,95±0,66
VORTEX	ViT-S/16 (IN-22K)	98,95±0,10	98,78±0,26	83,45±0,63	84,17±0,95
TOCHA	ConvNeXt-T (IN-22K)	99,52±0,13	99,65±0,12	95,23±0,24	95,35±0,72
RADAM	ConvNeXt-B	99,80±0,10	99,10±0,17	94,58±0,50	89,70±0,58
VORTEX	ViT-B/16	98,78±0,24	99,88±0,20	86,08±0,55	83,87±0,85
TOCHA	ConvNeXt-B	99,65±0,12	99,80±0,10	97,65±0,25	**97,27±0,21**
RADAM	ConvNeXt-B (IN-22K)	**100,00±0,00**	99,83±0,11	95,80±0,40	93,00±0,92
VORTEX	ViT-B/16 (IN-22K)	99,12±0,17	99,35±0,12	86,03±0,31	83,32±0,69
TOCHA	ConvNeXt-B (IN-22K)	99,52±0,24	99,70±0,10	97,40±0,32	96,50±0,25
RADAM	ConvNeXt-L	**100,00±0,00**	99,98±0,08	95,30±0,35	91,45±0,44
VORTEX	DeiT3-L/16	99,35±0,12	99,70±0,10	93,35±0,63	89,85±0,54
TOCHA	ConvNeXt-L	99,78±0,08	**100,00±0,00**	97,23±0,47	96,37±0,26
RADAM	ConvNeXt-L (IN-22K)	99,70±0,15	99,70±0,10	96,05±0,33	92,95±0,31
VORTEX	DeiT3-L/16 (IN-22K)	99,42±0,11	99,00±0,25	94,62±0,30	90,72±0,59
TOCHA	ConvNeXt-L (IN-22K)	**100,00±0,00**	**100,00±0,00**	97,02±0,33	96,82±0,32
RADAM	ConvNeXt-XL (IN-22K)	99,65±0,12	99,60±0,12	96,02±0,26	92,75±0,54
RADAM	ConvNeXt-XXL (LAION-2B)	**100,00±0,00**	99,35±0,23	96.47±0.21	92.70±0.33
VORTEX	ViT-H/14 (LAION-2B)	99,98±0,08	99,57±0,20	94,78±0,34	90,40±0,42
TOCHA	ConvNeXt-XL (IN-22K)	99,60±0,20	99,98±0,08	**98,17±0,16**	96,57±0,25
TOCHA	ConvNeXt-XXL (LAION-2B)	99,78±0,08	99,57±0,11	97,35±0,32	96,15±0,30

*TOCHA está com a configuração selecionada no Capítulo 5, exceto com a variação dos *backbones*.

Fonte: Elaborada pelo autor.

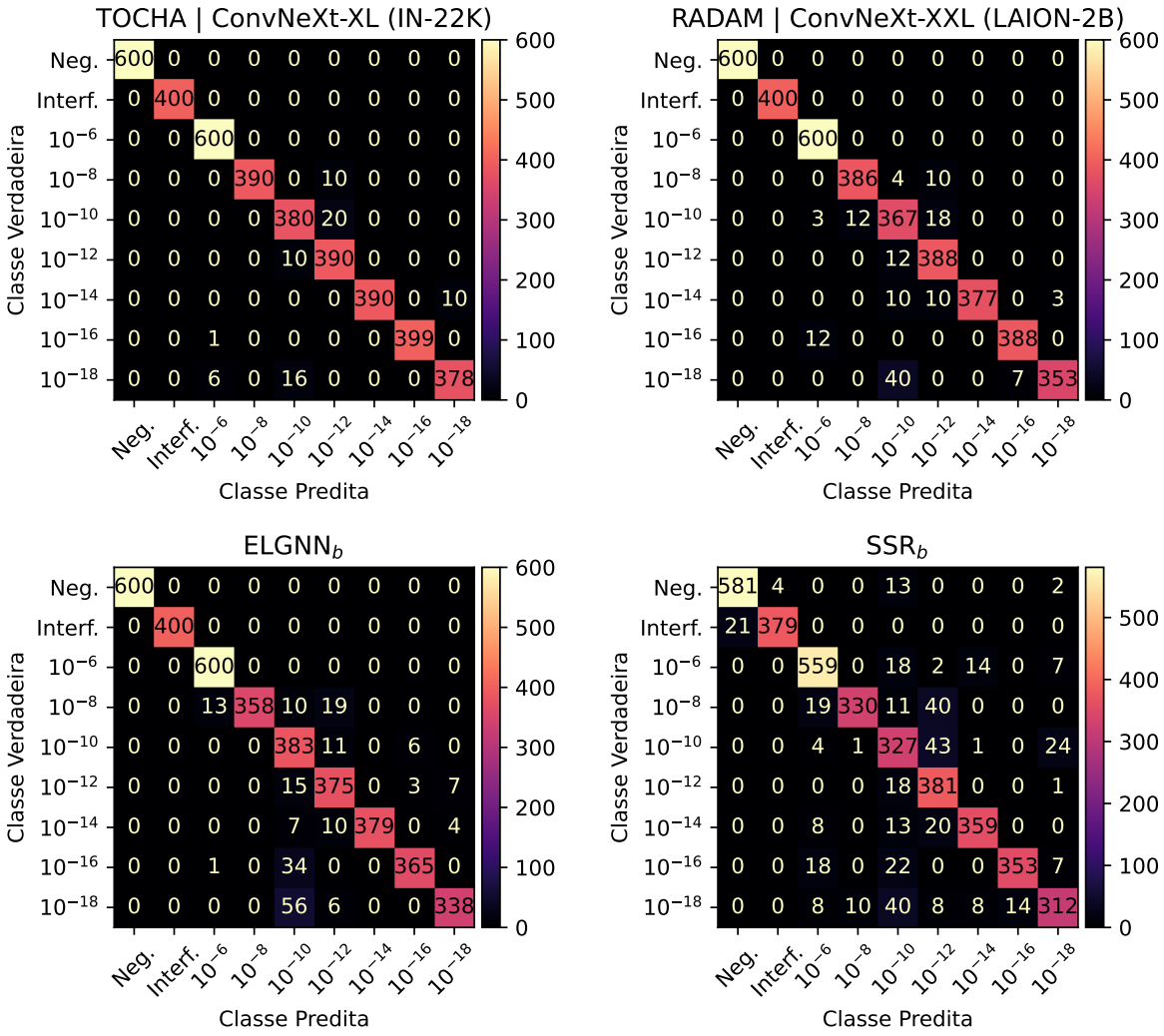
analitos e, principalmente, com relação às amostras com diferentes concentrações de ssDNA de SARS-CoV-2. Além disso, os melhores resultados obtidos pela abordagem ELGNN, que é baseada em grafos, superando outras abordagens da mesma família, tal como CNRNN, LCFNN e SSR, reforçam os benefícios da modelagem em grafos utilizada e, principalmente, como as medidas topológicas do grafo direcionado correspondente (i.e., graus de saída) foram cruzadas dentro da rede neural randomizada, com a finalidade da construção de uma representação mais discriminativa. Portanto, a fim de dar continuidade, a seguir, apresenta-se a análise das abordagens com redes neurais randomizadas híbridas para representação de textura.

A Tabela 14 apresenta as acurácias de classificação obtidas por meio de diferentes técnicas randomizadas híbridas. Especificamente, incluem-se os métodos RADAM e VORTEX, representativos do estado da arte e utilizados para fins de comparação, enquanto a técnica TOCHA corresponde à abordagem híbrida proposta nesta dissertação. Em primeira análise, a partir dessa tabela, observa-se que a abordagem TOCHA alcançou as melhores acurácias em ambos os contextos de classificação e para ambos os classificadores. Em particular, no cenário binário, a representação TOCHA (ConvNeXt-L IN-22K) obteve a acurácia máxima de 100,00% em ambos os classificadores, mantendo-se competitiva em relação às representações RADAM (ConvNeXt-B IN-22K, ConvNeXt-L e ConvNeXt-XLL LAION-2B) as quais também alcançaram a acurácia máxima. Em relação ao classificador SVM, apenas a representação proposta obteve acurácia máxima, entretanto, a diferença entre a melhor acurácia do estado da arte que é de 99,98% obtida pelas representações RADAM (ConvNeXt-T IN-22K e CovnNeXt-L) é ínfima, assim, culminando na mesma adversidade encontrada durante a análise dos métodos puros, a qual é a saturação dos resultados no contexto binário. Desse modo, de maneira similar, o poder discriminativo das representações será melhor analisado no contexto multiclasse.

Na configuração do problema multiclasse, sob o classificador LDA, a maior acurácia de 98,17% foi obtida pela representação TOCHA (ConvNeXt-XL IN-22K), superando em 1,70% a acurácia de 96,47% obtida por RADAM (ConvNeXt-XXL LAION-2B). Por outro lado, sob o classificador SVM, a maior acurácia de 97,27% foi alcançada pela representação TOCHA (ConvNeXt-B), ultrapassando em 4,27% a acurácia de 93,00% obtida por RADAM (ConvNeXt-B IN-22K). Além disso, observa-se a eficiência do poder discriminativo da abordagem proposta notando a existência de representações TOCHA com *backbones* menores alcançando acurácias superiores em relação às representações RADAM e VORTEX com *backbones* maiores. Como exemplo, com o classificador LDA, as representações TOCHA (ConvNeXt-B e ConvNeXt-L) em ambos os conjuntos de pré-treinamento IN-1K e IN-22K alcançaram acurácias superiores ao estado da arte obtido anteriormente por RADAM (ConvNeXt-XXL LAION-2B). Ainda mais, com relação ao classificador SVM, as representações TOCHA (ConvNeXt-T e ConvNeXt-T IN-22K) também obtiveram acurácias superiores, em relação ao estado da arte obtido pelo RADAM, utilizando *backbones* menores. Logo, assim como nas abordagens puras, torna-se evidente a tradução na prática dos benefícios metodológicos incluídos nas abordagens de representação de textura propostas nesta dissertação.

Com relação às matrizes de confusão das representações TOCHA (ConvNeXt-XL IN-22K) e RADAM (ConvNeXt-XXL LAION-2B) apresentadas na Figura 45, observa-se que ambas foram capazes de detectar as amostras negativas, interferentes e com concentração de 1×10^{-6} mol L^{-1} de ssDNA de SARS-CoV-2 completamente. Ainda mais, tal representação da abordagem RADAM apresenta confusões entre as concentrações 1×10^{-8} e 1×10^{-10} mol L^{-1}, e também entre as concentrações 1×10^{-12} e 1×10^{-14} mol L^{-1}, enquanto a representação de TOCHA não apresenta quaisquer dessas confusões. Em geral, as visualizações das menores confusões feitas por TOCHA, também visíveis analisando-se outras classes, corroboram com as

Figura 45 – Matrizes de confusão das representações aplicadas ao problema de diagnóstico de SARS-CoV-2 com imagens de genossensores. O cenário em questão é multiclasse sob o classificador LDA. Cada linha apresenta as representações com melhor acurácia atual e prévia, respectivamente. A primeira linha corresponde aos métodos híbridos, enquanto a segunda corresponde aos métodos puros.



Fonte: Elaborada pelo autor.

métricas de precisão e sensibilidade[4], em que os valores com o classificador LDA são de 98,17% e 97,96%, respectivamente, enquanto que para a representação de RADAM tais valores foram de 96,69% e 96,15%. Desse modo, os desempenhos obtidos por TOCHA nas métricas de acurácia, precisão e sensibilidade demonstram que esse acerta mais amostras no geral e possui menores casos de falsos positivos e falsos negativos, assim, realçando a discriminabilidade do método proposto na aplicação de diagnóstico de SARS-CoV-2 por imagens de genossensores.

Em seguida, ao analisar as matrizes de confusão para a representação ELGNN$_b$ e SSR$_b$, tem-se que a primeira, de modo similar à TOCHA, acerta completamente as classes de

[4] Note que os valores reportados nesta análise são provenientes do emprego do protocolo experimental previamente apresentado e não calculados diretamente das matrizes de confusão.

amostras negativas, interferentes, e de concentração mais alta, enquanto que nenhuma classe é completamente distinguível em SSR_b. Ainda mais, a representação $ELGNN_b$, apresenta uma menor quantidade de falsos negativos em oito das nove classes, quando comparada à SSR_b. Nesse contexto, sob uma perspectiva quantitativa, tem-se que $ELGNN_b$ obtém com LDA precisão e sensibilidade de 95,47% e 94,51%, respectivamente, enquanto que SSR_b obtém 90,56% e 88,82%. Assim, ainda de modo mais expressivo do que quando em relação aos métodos híbridos, exibe-se que $ELGNN_b$ supera SSR_b em 4,91% na métrica de precisão, e em 5,69% na métrica de sensibilidade. Desse modo, fica evidente a redução de falsos positivos e falsos negativos por parte da representação proposta $ELGNN_b$ em relação à SSR_b.

Desse modo, de forma análoga à seção anterior, que abordou imagens de genossensores para o diagnóstico de câncer de próstata, as abordagens puras e híbridas de representação para imagens em nível de cinza, desenvolvidas nesta dissertação, superaram as técnicas do estado da arte. Além disso, o desempenho notório em ambas as aplicações evidencia o funcionamento das técnicas em diferentes contextos de ampliação microscópica, i.e., 50.000× para PCA3 e 1.000× para SARS-CoV-2, reforçando, assim, a adaptabilidade e a capacidade discriminativa das metodologias de caracterização microtextural propostas. Por fim, o desempenho significativo aliado ao baixo custo computacional das técnicas propostas torna cada vez mais viável a aplicação dos métodos de visão computacional a imagens de genossensores em pontos de atendimento (PoC, do inglês *Point of Care*), inclusive em regiões com recursos limitados.

6.3 DIAGNÓSTICO DO VÍRUS SARS-COV-2 POR IMAGENS DE MICROSCOPIA ÓPTICA DE IMUNOSSENSORES

A importância do diagnóstico de SARS-CoV-2 por biossensores foi extensivamente discutida na Seção 6.2, a qual aborda essa mesma doença. Entretanto, diferentemente da seção referida, o problema de diagnóstico abordado nesta seção apresenta duas diferenças fundamentais. Primeiramente, na seção anterior, o SARS-CoV-2 era detectado por meio de genossensores cujo mecanismo de detecção baseia-se na hibridização entre sequências de ssDNA alvo e complementar; nesta seção, por sua vez, emprega-se o uso de imunossensores, cujo mecanismo de detecção ocorre por meio da interação antígeno-anticorpo. Em segundo lugar, de forma mais alinhada ao emprego de técnicas de visão computacional, observa-se que, na seção anterior, as imagens de genossensores foram obtidas por microscopia eletrônica de varredura, a qual implica maior custo de aquisição e manutenção, além de requerer etapas adicionais de preparação das amostras de biossensores. Por outro lado, nesta seção, as imagens de imunossensores são adquiridas por microscopia óptica, a qual é mais acessível e demanda menor preparação, tornando mais prática e barato o diagnóstico de SARS-CoV-2 nos pontos de atendimento (Oiticica et al., 2025).

Tabela 15 – Descrição geral do conjunto de dados de imunossensores para a detecção de SARS-CoV-2. Apresenta-se o número de amostras, se a classe é distinguível das demais, e qual a classe representante durante o contexto de classificação binário.

Analito (Concentração*)	Classe	# Amostras	Distinguível**	Binário***
SARS-CoV-2 (1×10^{5})	CoV(5)	43	Sim	Positiva
SARS-CoV-2 (1×10^{4})	CoV(4)	40	Sim	Positiva
SARS-CoV-2 (1×10^{3})	CoV(3)	21	Sim	Positiva
SARS-CoV-2 (1×10^{2})	CoV(2)	20	Sim	Positiva
SARS-CoV-2 (1×10^{1})	CoV(1)	21	Sim	Positiva
SARS-CoV-2 (1×10^{0})	CoV(0)	66	Sim	Positiva
SARS-CoV-2 (1×10^{-1})	CoV(-1)	40	Não	–
SARS-CoV-2 (1×10^{-2})	CoV(-2)	60	Não	–
SARS-CoV-2 (1×10^{-3})	CoV(-3)	42	Não	–
SARS-CoV-2 (1×10^{-4})	CoV(-4)	40	Não	–
Sonda	Negativa	286	Sim	Negativa
Branco, PBS/$MgCl_2$	Negativa	140	Sim	Negativa
RSV (1×10^{5})	Negativa	39	Sim	Negativa
RSV (1×10^{4})	Negativa	16	Sim	Negativa
RSV (1×10^{3})	Negativa	20	Sim	Negativa

*A unidade da concentração utilizada é PFU/mL.

**Indica se a classe é distinguível por algoritmos de aprendizado de máquina e técnicas de representação de textura utilizadas no artigo Oiticica et al. (2025).

***Qual classe representa durante o contexto binário.

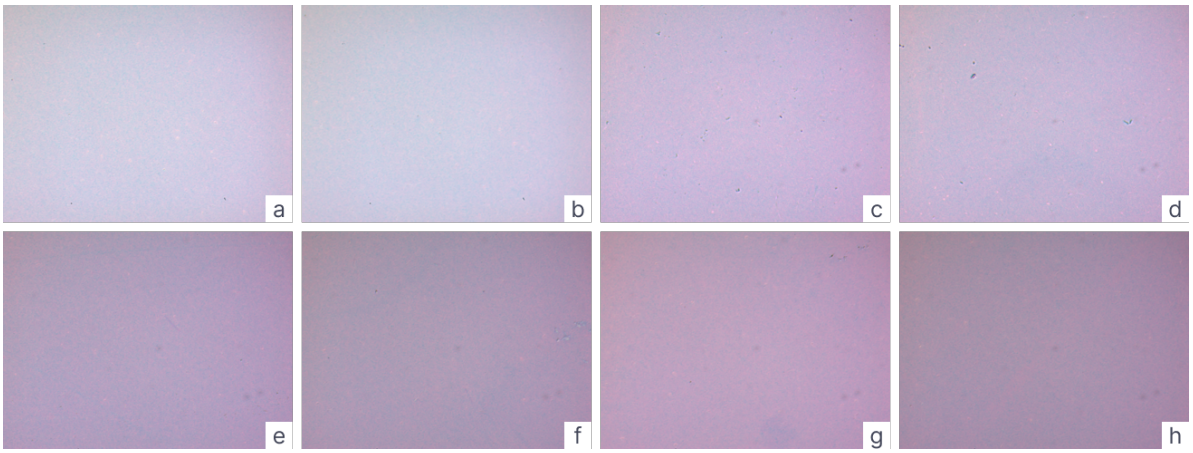
Fonte: Adaptada de Oiticica et al. (2025).

6.3.1 Descrição do Conjunto de Dados

O conjunto de dados de imunossensores proposto em Oiticica et al. (2025) é composto por 894 amostras de imagens de microscopia óptica de imunossensores feitos em substratos plasmônicos. Inicialmente, as imagens estão divididas em 11 classes, em que dez dessas classes correspondem às imagens das amostras dos imunossensores submetidas à diferentes concentrações do vírus inativado de SARS-CoV-2, que variam de 1×10^{5} PFU/mL a 1×10^{-4} PFU/mL, e a única classe restante corresponde a denominada classe negativa, a qual é formada pelo agrupamento das amostras dos imunossensores anterior aos testes, amostras em branco, e amostras de imunossensores submetidas às concentrações de 1×10^{3} PFU/mL a 1×10^{5} PFU/mL do vírus inativo de RSV (do inglês, *Respiratory Syncytial Virus*). Essas imagens foram obtidas utilizando o microscópio óptico de transmissão da Zeiss Axio Lab.A1 com a lente objetiva Zeiss A-Plan $40\times/0.65$ com $400\times$ de aumento. As imagens coloridas das amostras dos imunossensores obtidas possuem resolução de 1920×2560 pixels, representando um campo de visão de 165×220 μm, quantizadas em 8-bits por canal (Oiticica et al., 2025)[5].

[5] Informações mais detalhadas do processo de preparo dos imunossensores e da metodologia completa de aquisição das imagens das amostras dos imunossensores podem ser consultadas em Oiticica et al. (2025).

Figura 46 – Imagem das amostras dos imunossensores para o diagnóstico de SARS-CoV-2. As imagens correspondem aos analitos (a) sequência negativa (b) branco e às diferentes concentrações em PFU/mL do vírus inativado de SARS-CoV-2 (c) 1×10^5 (d) 1×10^4 (e) 1×10^3 (f) 1×10^2 (g) 1×10^1 (h) 1×10^0.



Fonte: Elaborada pelo autor. As imagens das amostras são provenientes do conjunto de dados proposto em Oiticica et al. (2025).

A Tabela 15 apresenta uma descrição geral do conjunto de dados de imunossensores, e a Figura 46 apresenta as imagens dos imunossensores para um subconjunto das classes. À primeira vista, com relação à tabela, apresentam-se os analitos, a classe, o número de amostras por classe, e a classe correspondente do analito sob o contexto de classificação binário. Dessa forma, fica evidente que tal problema de classificação será abordado em dois contextos, sendo eles: binário e multiclasse. O problema binário consiste na detecção da presença do vírus inativado de SARS-CoV-2 na amostra, enquanto que, no contexto multiclasse, busca-se não apenas identificar essa presença, mas também determinar a sua concentração na amostra, tornando o problema mais desafiador. Em segunda análise, a Tabela 15 também apresenta a coluna que determina se a classe em questão é distinguível das demais, em que essa distinção foi investigada e analisada em Oiticica et al. (2025), e tem como objetivo a remoção das classes que não foram possíveis de serem distinguidas com os algoritmos e técnicas empregadas no estudo. Em particular, como esperado, as classes não possíveis de serem distinguidas foram aquelas com concentrações mais baixas. Dessa forma, tal etapa foi conduzida no estudo em questão com o objetivo de avaliar a capacidade de algoritmos de inteligência artificial de detectar diferentes concentrações, inclusive as mais baixas. Entretanto, conforme cautelosamente apontado em Oiticica et al. (2025), não se deve interpretar a detecção da menor concentração como o limite de detecção do método, uma vez que não existe, até o momento, uma definição precisa de limite de detecção aplicável a algoritmos de aprendizado de máquina.

Nesse contexto, com a finalidade de manter a consistência entre a investigação do estudo em questão e a análise conduzida nesta dissertação, adota-se a mesma configuração do conjunto de dados empregada no referido artigo. Especificamente, essa configuração corres-

ponde à remoção das amostras associadas às classes indistinguíveis, resultando em um conjunto de dados composto por 712 amostras, no qual seis das sete classes representam as maiores concentrações do vírus inativado de SARS-CoV-2 (i.e., concentrações variando de $1{\times}10^5$ a $1{\times}10^0$ PFU/mL) e a classe remanescente corresponde à classe negativa (i.e., agrupamento das amostras negativas, em branco e com diferentes concentrações do vírus RSV inativo). Assim, esse conjunto de dados será utilizado ao longo da experimentação e da análise dos resultados obtidos pelas técnicas de representação de texturas.

6.3.2 Análise Experimental e Discussões

Esta subseção apresenta a análise experimental referente a aplicação das técnicas puras e híbridas investigadas e desenvolvidas nesta dissertação para diagnosticar a presença ou não do vírus inativado de SARS-CoV-2 em amostras de imagens de imunossensores bem como sua concentração com a finalidade de auxiliar o diagnóstico por meio de inteligência artificial. Para isso, a seguir, o protocolo experimental adotado é apresentado, que será responsável pela obtenção dos resultados, e, em seguida, a discussão dos resultados obtidos será realizada.

Protocolo Experimental. A acurácia é utilizada como métrica de desempenho, a qual é obtida a partir de três classificadores: *Linear Discriminant Analysis* (LDA), K-Nearest Neighbors (KNN) com $k = 1$ e Support Vector Machine (SVM) com *kernel* linear. Especificamente, todos os três são utilizados na análise dos métodos tradicionais e randomizados puros, enquanto que somente o primeiro e o último são utilizados para a análise das metodologias randomizadas híbridas. Ainda mais, adota-se a mesma estratégia de validação cruzada utilizada no estudo original, que corresponde ao 5-*fold* estratificado com três repetições, em que a estratificação é empregada com o objetivo de preservar a proporção original das classes.

Resultados. A Tabela 16 apresenta os resultados comparativos entre as abordagens randomizadas puras propostas para imagens coloridas nesta dissertação e abordagens da literatura. Especificamente, as técnicas comparadas são: Fourier, LBP, CNTD, SSN e SSR, enquanto que os métodos puros propostos para representação de imagens coloridas são: SST, VCTex e MIXER. A partir dessa tabela, à primeira vista, nota-se que as maiores acurácias em ambos os contextos de classificação (i.e., binário e multiclasse) e para todos os classificadores foram obtidas pelos métodos propostos neste trabalho. Os resultados substanciam o poder discriminativo das abordagens investigadas e desenvolvidas para a caracterização microtextural presente nas imagens de imunossensores, uma vez que todos os resultados foram obtidos utilizando tais imagens em sua resolução original de $1920{\times}2560$ pixels. Nesse contexto, a seguir, analisam-se os resultados correspondentes a cada problema de classificação apresentado.

Em relação ao contexto de classificação binária, tem-se que a técnica MIXER superou, consideravelmente, todos os resultados prévios para todos os classificadores, enquanto as abordagens SST e VCTex superaram significativamente tais resultados com o classifica-

Tabela 16 – Comparação das acurácias de classificação no conjunto de dados de Imunossensores (SARS-CoV-2) obtidas pelos métodos randomizados puros propostos com aqueles presentes na literatura. Em cada coluna, o maior valor está em **destaque**, o valor em azul indica o melhor resultado da literatura, e em vermelho indica resultados iguais ou superiores por parte dos métodos propostos. A dimensão da imagem utilizada é 1920×2560×3.

	Binário			Multiclasse		
Método	LDA (↑)	KNN (↑)	SVM (↑)	LDA (↑)	KNN (↑)	SVM (↑)
Fourier†	80,66±2,80	83,62±3,12	63,29±7,89	72,19±2,94	74,48±3,57	59,37±7,87
LBP†	83,61±2,57	92,04±2,03	90,08±2,24	77,48±3,94	86,94±2,39	89,84±1,72
CNTD†	80,90±3,65	86,24±2,92	76,08±2,63	75,23±2,53	78,79±3,16	73,60±1,33
SSN (φ^4)	93,02±2,18	83,29±2,90	85,95±3,90	86,70±2,49	74,58±2,68	81,83±2,36
SSR_a	94,33±1,84	92,14±1,76	91,01±2,17	92,14±1,49	86,61±2,50	90,54±2,21
SSR_b	94,61±1,75	92,23±1,82	91,67±1,83	92,88±1,33	86,66±2,51	90,40±2,55
Métodos Propostos						
SST_a	87,03±2,57	98,64±1,22	91,43±2,72	83,19±1,90	96,11±1,43	93,35±2,10
SST_b	87,64±1,71	98,50±1,09	86,70±6,28	81,23±2,10	95,79±1,00	93,72±2,42
SST_c	85,39±2,10	98,55±1,35	79,39±12,54	78,84±1,60	95,50±1,36	93,07±1,55
$VCTex_a$	89,75±1,53	98,36±1,23	86,24±6,26	83,85±1,79	95,97±1,55	92,18±2,15
$VCTex_b$	89,14±1,60	**98,73±1,07**	80,45±9,24	83,62±1,48	95,97±1,57	90,92±1,61
$VCTex_c$	88,16±2,28	98,41±0,91	78,53±13,10	80,99±2,19	**96,25±1,02**	90,31±1,62
$MIXER_a$	96,16±1,78	98,45±1,15	94,52±1,78	92,14±1,51	94,99±1,52	94,24±1,36
$MIXER_b$	**97,00±1,32**	98,36±1,20	**95,13±1,86**	**93,07±1,51**	94,94±1,59	**94,62±1,01**
$MIXER_c$	94,90±1,90	98,27±1,12	93,96±1,97	91,11±1,44	94,76±1,87	93,68±1,34

†Métodos originalmente nível de cinza utilizados na versão integrativa.

Fonte: Elaborada pelo autor.

dor KNN. Em particular, a representação $MIXER_b$ obteve as maiores acurácias de 97,00% e 95,13% com os classificadores LDA e SVM, respectivamente, de modo que, aumentou em 2,39% e 3,46% as melhores acurácias prévias de 94,61% e 91,67% obtidas pela representação SSR_b. Ainda, além do maior desempenho obtido por $MIXER_b$, deve-se notar a eficiência e a compacidade da representação, desde que $MIXER_b$ obteve os melhores resultados com 720 atributos, enquanto SSR_b possui 990 atributos, representando uma redução de 27,27%.

Por outro lado, ainda no mesmo contexto de classificação, mas com relação ao classificador KNN, tem-se que a representação $VCTex_b$ alcançou a maior acurácia de 98,73%, representando um aumento significativo de 6,50% em relação a SSR_b e cujo aumento representa um acréscimo de aproximadamente 46 imagens corretamente classificadas. Além disso, similar à análise anterior, novamente exibem-se a eficiência e a discriminabilidade da representação sob este classificador, desde que a representação $VCTex_b$ possui apenas 540 atributos, assim, representando uma redução de 45,45% em relação à representação de textura SSR_b.

Com relação ao contexto de classificação multiclasse, tem-se que a representação

MIXER$_b$ superou os melhores resultados prévios em todos os classificadores. Além disso, todas as abordagens propostas foram capazes de superar os resultados prévios sob os classificadores KNN e SVM, exceto para a VCTex$_c$ que não superou sob o classificador SVM. Portanto, assim como no contexto binário, os resultados significativos realçam a discriminabilidade das abordagens puras propostas. Especificamente, a representação MIXER$_b$ obteve as melhores acurácias de 93,07% e 94,62% com LDA e SVM, respectivamente, assim, superando os melhores resultados prévios de 92,88% e 90,54% obtidos pelas técnicas SSR$_b$ e SSR$_a$.

Além disso, ainda no contexto multiclasse, a representação VCTex$_c$ obteve o melhor resultado de 96,25% sob o classificador KNN, representando um aumento considerável de 9,31% em relação ao LBP que obteve o melhor resultado prévio de 86,94%. Esse aumento corresponde ao acréscimo de aproximadamente 66 imagens corretamente classificadas. Logo, ao analisar os resultados obtidos em ambos os contextos de classificação e todos os classificadores abordados, torna-se evidente o benefício da aplicação das técnicas de caracterização microtextural propostas para o diagnóstico de SARS-CoV-2 por meio de imagens de microscopia óptica. Dessa forma, a seguir, são apresentados e discutidos os resultados obtidos com relação às abordagens de representação de texturas com redes neurais randomizadas híbridas.

A Tabela 17 apresenta as acurácias obtidas com os classificadores LDA e SVM por meio de abordagens randomizadas híbridas, i.e., aquelas que empregam uma combinação de redes neurais randomizadas com *backbones* pré-treinados. Em particular, são consideradas duas técnicas híbridas para comparação, sendo essas: RADAM e VORTEX, as quais são comparadas em relação à técnica proposta TOCHA. Ainda, deve-se observar que as representações resultantes por meio dessas técnicas, considerando diferentes *backbones*, foram obtidas a partir de imagens de imunossensores na resolução de 224×224, a qual é significativamente inferior à resolução original. Essa etapa foi conduzida por dois motivos principais. Primeiramente, essa resolução corresponde ao padrão adotado por diferentes tipos de *backbone* e, portanto, foi escolhida por consistência. Em segundo lugar, a depender do *backbone* empregado e da resolução da imagem (i.e., original ou reduzida), pode ocorrer o esgotamento da memória da unidade de processamento gráfico, inviabilizando a extração das representações de textura. Dessa forma, em razão desses fatores, mantém-se a resolução padrão de 224×224 pixels.

Assim, a partir da Tabela 17, nota-se que a representação TOCHA (ConvNeXt-XXL LAION-2B) foi a que obteve as melhores acurácias em ambos os contextos de classificação e para os dois classificadores. Em particular, com relação ao contexto de classificação binário, a representação em questão obteve as acurácias de 94,99% e 95,55% sob o LDA e SVM, respectivamente. Esses valores indicam um aumento de 2,29% e 2,20% em relação às melhores acurácias prévias de 92,70% e 93,35% obtidas por RADAM (ConvNeXt-XXL LAION-2B). Ainda mais, nessa tabela é possível notar a eficiência da abordagem proposta TOCHA, desde que tal técnica alcançou resultados superiores ao estado da arte mesmo utilizando *backbones* menores, os quais possuem menor custo computacional. Como exemplo, a representação TO-

Tabela 17 – Comparação das acurácias de classificação no conjunto de dados de Imunossensores (SARS-CoV-2) obtidas pelo método randomizado híbrido proposto com aqueles presentes na literatura. Em cada coluna, o maior valor está em **destaque**, o valor em azul indica o melhor resultado da literatura, e em vermelho indica resultados iguais ou superiores por parte dos métodos propostos.

		Binário		Multiclasse	
Método	Backbone	LDA (↑)	SVM (↑)	LDA (↑)	SVM (↑)
RADAM	ConvNeXt-T	89,05±2,02	88,20±2,16	83,38±2,09	78,37±1,89
VORTEX	ViT-S/16	91,01±1,55	91,39±1,92	86,14±2,54	87,73±2,90
TOCHA	ConvNeXt-T	91,10±1,68	91,76±1,87	89,23±1,69	91,90±1,80
RADAM	ConvNeXt-T (IN-22K)	91,43±1,97	88,76±2,00	83,99±2,73	79,77±1,71
VORTEX	ViT-S/16 (IN-22K)	89,56±2,26	90,59±2,27	85,91±2,39	89,42±2,42
TOCHA	ConvNeXt-T (IN-22K)	91,34±1,75	87,59±2,45	87,31±2,05	91,01±2,14
RADAM	ConvNeXt-B	90,78±1,89	89,51±1,85	84,79±2,06	79,96±2,57
VORTEX	ViT-B/16	92,60±1,89	93,02±1,77	86,75±2,36	90,68±2,29
TOCHA	ConvNeXt-B	93,30±2,38	94,38±2,48	90,31±1,78	91,81±2,14
RADAM	ConvNeXt-B (IN-22K)	87,74±2,99	88,53±1,85	81,51±2,56	80,62±1,88
VORTEX	ViT-B/16 (IN-22K)	91,57±1,96	91,99±1,79	90,58±2,80	90,03±2,46
TOCHA	ConvNeXt-B (IN-22K)	90,21±1,51	89,19±2,37	85,96±2,23	91,29±1,74
RADAM	ConvNeXt-L	90,92±2,41	89,93±1,94	85,16±1,98	80,62±2,49
VORTEX	DeiT3-L/16	85,82±2,59	88,86±2,44	79,68±2,53	80,20±2,52
TOCHA	ConvNeXt-L	94,43±1,56	94,71±1,59	92,04±2,69	93,35±1,83
RADAM	ConvNeXt-L (IN-22K)	88,63±2,81	88,16±2,77	84,03±2,61	81,55±2,53
VORTEX	DeiT3-L/16 (IN-22K)	91,48±1,72	90,87±2,38	81,70±2,73	80,94±1,94
TOCHA	ConvNeXt-L (IN-22K)	92,93±2,60	92,18±2,70	87,03±2,08	91,67±1,38
RADAM	ConvNeXt-XL (IN-22K)	91,11±1,69	90,59±2,22	83,43±2,75	82,49±2,34
RADAM	ConvNeXt-XXL (LAION-2B)	92,70±2,08	93,35±1,87	85,25±2,78	84,18±2,98
VORTEX	ViT-H/14 (LAION-2B)	88,58±1,85	88,62±1,65	82,40±2,51	82,96±2,09
TOCHA	ConvNeXt-XL (IN-22K)	92,70±2,36	93,31±1,07	88,63±2,43	91,25±1,47
TOCHA	ConvNeXt-XXL (LAION-2B)	**94,99**±0,91	**95,55**±1,39	**92,13**±1,96	**93,45**±1,42

*TOCHA está com a configuração selecionada no Capítulo 5, exceto com a variação dos *backbones*.

Fonte: Elaborada pelo autor.

CHA (ConvNeXt-B) que emprega um *backbone* de 15,4 GFLOPS e 87,6 M de parâmetros obteve 93,30% e 94,38% com LDA e SVM, respectivamente, enquanto RADAM (ConvNeXt-XXL LAION-2B) emprega um *backbone* de 151,7 GFLOPS com 843,4 M de parâmetros, obteve 92,70% e 93,35% como previamente apresentado. Por outro lado, com relação à configuração multiclasse, a mesma representação TOCHA (ConvNeXt-XXL LAION-2B) alcançou as melhores acurácias de 92,13% e 93,45% com LDA e SVM, respectivamente, assim, ultrapassando em 1,55% e 2,77% as melhores acurácias prévias obtidas pelas representações de textura VORTEX (ViT-B/16 IN-22K) e VORTEX (ViT-B/16), respectivamente.

Por fim, a partir da análise e discussão dos resultados apresentados, evidencia-se que as abordagens de caracterização microtextural de imagens coloridas propostas nesta dissertação mostraram-se eficazes na abordagem do problema de diagnóstico de SARS-CoV-2, agora utilizando imagens de imunossensores obtidos por microscopia óptica, a qual apresenta fator de ampliação inferior em relação aos problemas das seções anteriores. Assim, os resultados obtidos reforçam de maneira ainda mais consistente o potencial da área de análise textural de imagens de biossensores no contexto do diagnóstico médico.

6.4 CONSIDERAÇÕES FINAIS

Neste capítulo, foram abordadas três aplicações cujo objetivo é o diagnóstico de doenças por meio da análise de imagens de biossensores. A primeira aplicação abordada refere-se ao diagnóstico de câncer de próstata com o biomarcador PCA3 por meio de imagens de genossensores, enquanto a segunda e a terceira aplicações dizem respeito ao diagnóstico de SARS-CoV-2 a partir de imagens de genossensores e imunossensores, respectivamente. As duas primeiras aplicações empregaram microscópios eletrônicos para a obtenção de imagens em nível de cinza, resultando em maior fator de ampliação e maior custo. Por outro lado, a terceira aplicação adotou microscopia óptica para a aquisição de imagens coloridas, apresentando menor fator de ampliação e menor custo. Nesse contexto, a aplicação das abordagens investigadas nesta dissertação em três cenários distintos, i.e., envolvendo diferentes biossensores, fatores de ampliação e tipos de microscopia, reforça que os métodos propostos foram avaliados sob configurações variadas, evidenciando sua robustez e generalização.

Associado a isso, observa-se que os resultados apresentados na três aplicações demonstram, de forma consistente, que as abordagens propostas superaram aquelas do estado da arte. Esses resultados indicam que as metodologias desenvolvidas, guiadas pela busca por representações mais robustas e discriminativas por meio da caracterização do conteúdo microtextural dessas imagens, realizadas por diferentes formas, entre elas: múltiplas projeções aleatórias; modelagem em grafo e cruzamento de informações topológicas da rede; módulos de aprendizado espaço-espectral; e extração simultânea de informação de cor e textura por cubos volumétricos, configuraram-se como um caminho promissor no diagnóstico médico auxiliado por inteligência artificial. Nesse contexto, torna-se claro como as contribuições científicas dos métodos desenvolvidos traduziram-se na prática, a partir não apenas da abordagem de problemas de elevada relevância médica e social, mas também do emprego de imagens de biossensores, as quais ainda são pouco exploradas na literatura.

7 CONCLUSÕES

Esta dissertação aborda a investigação e o desenvolvimento de abordagens de aprendizado de representações de textura baseadas em redes neurais randomizadas para a análise de imagens de biossensores, com ênfase no diagnóstico de doenças, tais como SARS-CoV-2 e câncer de próstata. A utilização de imagens de biossensores para o diagnóstico assistido por aprendizado de máquina emergiu recentemente como uma nova modalidade de detecção, em adição às abordagens físico-químicas tradicionalmente empregadas. Especificamente, investigações recentes exibiram que uma vez que os biossensores são expostos à amostra biológica, esses sofrem mudanças morfológicas e texturais, de modo que são detectáveis por algoritmos de aprendizado de máquina, uma vez que boas representações de textura são utilizadas. Dessa forma, evidencia-se a importância dos métodos de representação de textura desenvolvidos nesta dissertação, os quais são aplicados em um problema real relevante para a sociedade.

Entretanto, ao abordar tais imagens de biossensores, os desafios inerentes a essas imagens referem-se à presença de padrões microtexturais complexos e o tamanho ínfimo do conjunto de dados, de modo que se fez necessário a investigação e desenvolvimentos de técnicas de representação de textura rápidas, de baixo custo computacional, que demandassem poucos dados e que fossem capazes de caracterizar tais imagens por meio de representações de textura compactas e discriminativas. Nesse contexto, fica claro os motivos pelos quais as redes neurais randomizadas foram empregues nesta dissertação para o aprendizado de representações de textura, uma vez que essa combina características de abordagens clássicas e baseadas em aprendizado simultaneamente. Em particular, tais características vantajosas são o baixo custo computacional o qual é decorrente do fato que a etapa de treinamento da rede é conduzida por uma solução de fórmula fechada e, principalmente, o fato dessa rede ser capaz de aprender representações por instância, i.e., a aprender e extrair representações de imagens individualmente, sem a necessidade de ter visto outras imagens do conjunto de dados.

Contudo, com o objetivo de extrair representações mais ricas, fez-se necessário a identificação das lacunas presentes nas atuais técnicas de aprendizado de representações de textura com redes neurais randomizadas, juntamente com a proposição de novas abordagens, adjetivadas por puras, cuja finalidade foi preencher as lacunas existentes. Para isso, a primeira frente de pesquisa foi responsável por tais ações, de modo que lacunas existentes, tais como: novas formas de cruzamento de dados pela rede, emprego da rede neural randomizada na forma multialvo e *autoencoder*, emprego de conceitos advindos de mecanismos autossupervisionados e, principalmente, alterações arquiteturais, foram totalmente preenchidas pelos métodos desenvolvidos. Em particular, cinco técnicas de representação puras foram desenvolvidas, as quais duas delas são para imagens de textura em nível de cinza, e três para imagens coloridas.

Por outro lado, visando angariar o conhecimento aprendido dos modelos pré-treinados em conjuntos de dados de larga escala, tem-se que as redes neurais randomizadas foram utilizadas para codificar aleatoriamente os mapas de ativação profundos com o objetivo de melhorar a transferibilidade dessas redes, assim, resultando em representações ainda mais discriminativas. Desse modo, a segunda frente de pesquisa investigou lacunas das técnicas de codificação aleatória com redes neurais randomizadas do estado da arte, resultando na proposição de uma abordagem a qual preenche tais lacunas por meio de uma nova modelagem em grafos dos mapas de ativação, uma forma alternativa de cruzamento de dados por parte da rede e, principalmente, a possibilidade de selecionar uma subsequência dos estágios do *backbone*, de modo a angariar apenas as características mais importantes para textura.

Diante desse contexto, as abordagens puras de representação de texturas foram submetidas a tarefas de reconhecimento textural utilizando quatro conjuntos de dados de *benchmark*, os quais apresentam distintos desafios e características visuais, possibilitando, assim, a avaliação do método proposto em diferentes cenários e, consequentemente, a construção de uma análise experimental mais robusta. Isso posto, considerando os experimentos conduzidos, tem-se que os desempenhos alcançados substanciam que as abordagens propostas apresentam resultados expressivos, sendo capazes de superar diversas técnicas do estado da arte baseadas em redes neurais randomizadas em diferentes conjuntos de dados de *benchmark*, evidenciando, de modo geral, uma maior capacidade de caracterização textural proporcionada pelos métodos propostos. Além disso, no experimento de eficiência computacional conduzido para as abordagens puras em nível de cinza, mostrou-se que essas apresentaram, de forma consistente, menores tempos de execução em comparação às abordagens do estado da arte da mesma categoria, evidenciando, assim, que os métodos investigados contribuíram simultaneamente para a redução do custo computacional e para a obtenção de melhores desempenhos médios.

Por fim, ambas abordagens puras e híbridas desenvolvidas nesta dissertação foram aplicadas à análise de imagens de biossensores, as quais foram obtidas por microscópios ópticos ou eletrônicos, para o diagnóstico de SARS-CoV-2 e câncer de próstata assistido por inteligência artificial. A partir disso, os desempenhos alcançados demonstram que as abordagens propostas superam todos os métodos da literatura considerados, reforçando, assim, a elevada capacidade de descrição microtextural dessas abordagens e, sobretudo, evidenciando como as contribuições metodológicas apresentadas se traduzem no âmbito prático, em especial no contexto de um problema tão relevante. Além disso, a capacidade de detectar, com maior acurácia, tanto a doença quanto diferentes concentrações do analito reforça a utilidade do emprego de abordagens de aprendizado de máquina como técnica de detecção baseada em análise de imagens. Dessa forma, tornam-se evidentes as contribuições teóricas das abordagens propostas, bem como sua aplicabilidade prática, as quais podem ser estendidas a diversas outras tarefas de visão computacional, uma vez que permitem a extração de representações de imagens de textura quaisquer, evidenciando, assim, que os métodos propostos contribuem para o avanço dos campos de visão computacional e reconhecimento de padrões, especialmente no

que diz respeito ao desenvolvimento de técnicas rápidas e de baixo custo computacional.

7.1 TRABALHOS FUTUROS

Embora as abordagens propostas apresentem distintas contribuições metodológicas, tanto no contexto de técnicas puras quanto híbridas, tem-se que essas também apresentam algumas limitações, as quais foram discutidas ao longo da dissertação e podem ser exploradas em trabalhos futuros para o desenvolvimento de técnicas de representação ainda mais robustas e discriminativas, contribuindo, assim, para o avanço dos campos de visão computacional e análise de texturas. Nesse contexto, a seguir, são listados tópicos para investigações futuras:

- **Medidas de Caracterização Topológicas**: As abordagens de textura que empregaram caracterizações topológicas restringiram-se ao uso dos graus de saída dos vértices modelados pelos grafos locais. Nesse contexto, resta investigar outras medidas topológicas tais como o grau de entrada, força de saída, força de entrada, entre outras.
- **Novos Problemas de Otimização**: Uma das mais importantes etapas da rede neural randomizada para um bom aprendizado de representações refere-se ao problema de otimização convexo. Dessa forma, torna-se interessante a investigação de outros problemas de minimização, principalmente aqueles convexos e que possuam soluções de fórmula fechada, entre eles o de mínimos quadrados multiobjetivo.
- **Novas Formas de Modelagem de Mapas de Ativação**: Até onde se tem conhecimento, TOCHA foi a primeira técnica híbrida de que propôs a modelagem dos mapas de ativação em grafos seguida de sua codificação aleatória. Nesse sentido, com os bons resultados alcançados, fica-se aberto o caminho para a investigação de outras formas de modelagem que possam extrair padrões ainda mais complexos e subjacentes.

7.2 LISTA DE PUBLICAÇÕES

A partir do estudo e desenvolvimento das abordagens propostas de aprendizado de representações de textura ao longo do período de mestrado, apresentam-se, a seguir, os trabalhos publicados e submetidos, os quais se encontram direta ou indiretamente relacionados ao presente trabalho.

Publicados. A seguir, são apresentados os artigos publicados.

- Scabini, L., Zielinski, K., **Fares, R.**, Konuk, E., Miranda, G., Kolb, R., Ribas, L., & Bruno, O. (2024). Deep Texture Feature Aggregation on Leaf Microscopy Images for Brazilian Plant Species Recognition. In *Proceedings of the 9th International Conference on Machine Learning Technologies* (pp. 209-213).

- **Fares, R. T.**, & Ribas, L. C. (2024). A New Approach to Learn Spatio-Spectral Texture Representation with Randomized Networks: Application to Brazilian Plant Species Identification. In *International Conference on Engineering Applications of Neural Networks* (pp. 435-449).
- **Fares, R. T.**, Vicentim, A. C. M., Scabini, L., Zielinski, K. M., Jennane, R., Bruno, O. M., & Ribas, L. C. (2024). Randomized encoding ensemble: A new approach for texture representation. In *31st International Conference on Systems, Signals and Image Processing* (pp. 1-8).
- **Fares, R. T.**, & Ribas, L. C. (2024). Randomized Autoencoder-based Representation for Dynamic Texture Recognition. In *31st International Conference on Systems, Signals and Image Processing* (pp. 1-7).
- Ribas, L. C., Casaca, W., & **Fares, R. T.** (2025). Conditional Generative Adversarial Networks and Deep Learning Data Augmentation: A Multi-Perspective Data-Driven Survey Across Multiple Application Fields and Classification Architectures. *AI*, 6(2), 32.
- **Fares, R. T.**, Guerra, L. B., & Ribas, L. C. (2025). Exploring Local Graphs via Random Encoding for Texture Representation Learning. In *20th International Conference on Computer Vision Theory and Applications*, (pp. 200-209).
- **Fares, R. T.**, & Ribas, L. C. (2025). Volumetric Color-Texture Representation for Colorectal Polyp Classification in Histopathology Images. In *20th International Conference on Computer Vision Theory and Applications*, (pp. 210-221).
- Scabini, L., Zielinski, K. M., Fernandes, M., **Fares, R. T.**, Ribas, L. C., Kolb, R. M., & Bruno, O. M. (2026). Complex networks for modeling texture and spectral features of hyperspectral images for environmental analysis. *Computers in Biology and Medicine*, 204, 111504.

Submetidos. A seguir, são apresentados os trabalhos submetidos.

- Scabini, L., Zielinski, K. M., Konuk, E., **Fares, R. T.**, Ribas, L. C., Smith, K., & Bruno, O. M. (2025). VORTEX: Challenging CNNs at Texture Recognition by using Vision Transformers with Orderless and Randomized Token Encodings. *arXiv* preprint arXiv:2503.06368.
- **Fares, R. T.**, & Ribas, L. C. (2025). MIXER: Mixed Hyperspherical Random Embedding Neural Network for Texture Recognition. *arXiv* preprint arXiv:2510.03228.
- Kobal, M., Camacho, S., Toledo, K., **Fares, R.**, Scabini, L., Ribas, L., Bruno, O., Oliveira, O., & Aoki, P. H. Deep Learning-Assisted Classification of Phototherapy-Induced Cellular Responses Mediated by Gold Shell-Isolated Nanoparticles.

Em redação. Os trabalhos em redação são listados a seguir:

- **Fares, R. T.** & Ribas, L. C. TOCHA: Texture Representation Learning by Topological Characterization and Random Encoding of Deep Activation Maps.

REFERÊNCIAS

ABADI, M. et al. {TensorFlow}: a system for {Large-Scale} machine learning. In: **12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)**. [S.l.: s.n.], 2016. p. 265–283.

ABDELMOUNAIME, S.; DONG-CHEN, H. New brodatz-based image databases for grayscale color and multiband texture analysis. **International Scholarly Research Notices**, Wiley Online Library, v. 2013, n. 1, p. 876386, 2013.

ABDIN, M. et al. Phi-4 technical report. **arXiv preprint arXiv:2412.08905**, 2024.

ACHIAM, J. et al. Gpt-4 technical report. **arXiv preprint arXiv:2303.08774**, 2023.

BACKES, A. R.; CASANOVA, D.; BRUNO, O. M. Plant leaf identification based on volumetric fractal dimension. **International Journal of Pattern Recognition and Artificial Intelligence**, World Scientific, v. 23, n. 06, p. 1145–1160, 2009.

BACKES, A. R.; CASANOVA, D.; BRUNO, O. M. Color texture analysis based on fractal descriptors. **Pattern Recognition**, Elsevier, v. 45, n. 5, p. 1984–1992, 2012.

BACKES, A. R.; CASANOVA, D.; BRUNO, O. M. Texture analysis and classification: A complex network-based approach. **Information Sciences**, Elsevier, v. 219, p. 168–180, 2013.

BAO, H. et al. Beit: Bert pre-training of image transformers. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2022.

BENGIO, Y.; COURVILLE, A.; VINCENT, P. Representation learning: A review and new perspectives. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 35, n. 8, p. 1798–1828, 2013.

BRODATZ, P. Textures: A photographic album for artisters and designers. **Dover Publications**, 1966.

BROWN, T. et al. Language models are few-shot learners. **Advances in Neural Information Processing Systems**, v. 33, p. 1877–1901, 2020.

CARVALHO, C. L. C. et al. Sustainable electrochemical-magnetic biosensor fabricated from recycled materials for label-free detection of sars-cov-2 in human saliva. **ACS sensors**, ACS Publications, v. 10, n. 3, p. 1970–1985, 2025.

CHEN, B. et al. Adversarial diffusion compression for real-world image super-resolution. In: **Proceedings of the Computer Vision and Pattern Recognition Conference**. [S.l.: s.n.], 2025. p. 28208–28220.

CHEN, J. et al. Vitamin: Designing scalable vision models in the vision-language era. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2024. p. 12954–12966.

CHEN, M. et al. Generative pretraining from pixels. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2020. p. 1691–1703.

CHEN, T. et al. A simple framework for contrastive learning of visual representations. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2020. p. 1597–1607.

CHEN, X. et al. Context autoencoder for self-supervised representation learning. **International Journal of Computer Vision**, Springer, v. 132, n. 1, p. 208–223, 2024.

CHEN, Z. et al. Deep texture recognition via exploiting cross-layer statistical self-similarity. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2021. p. 5231–5240.

CHEN, Z. et al. Enhancing texture representation with deep tracing pattern encoding. **Pattern Recognition**, Elsevier, v. 146, p. 109959, 2024.

CIMPOI, M.; MAJI, S.; VEDALDI, A. Deep filter banks for texture recognition and segmentation. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2015. p. 3828–3836.

CLEVERT, D.-A.; UNTERTHINER, T.; HOCHREITER, S. Fast and accurate deep network learning by exponential linear units (elus). In: **International Conference on Learning Representations**. [S.l.: s.n.], 2016.

CONG, S.; ZHOU, Y. A review of convolutional neural network architectures and their optimizations. **Artificial Intelligence Review**, Springer, v. 56, n. 3, p. 1905–1969, 2023.

CORDONNIER, J.-B.; LOUKAS, A.; JAGGI, M. On the relationship between self-attention and convolutional layers. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2020.

COVER, T. M. Geometrical and statistical properties of systems of linear inequalities with applications in pattern recognition. **IEEE Transactions on Electronic Computers**, EC-14, n. 3, p. 326–334, 1965.

CULP, M. B. et al. Recent global patterns in prostate cancer incidence and mortality rates. **European Urology**, Elsevier, v. 77, n. 1, p. 38–52, 2020.

DANA, K. J. et al. Reflectance and texture of real-world surfaces. **ACM Transactions On Graphics (TOG)**, ACM New York, NY, USA, v. 18, n. 1, p. 1–34, 1999.

DENG, J. et al. Imagenet: A large-scale hierarchical image database. In: IEEE. **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.], 2009. p. 248–255.

DEVLIN, J. et al. Bert: Pre-training of deep bidirectional transformers for language understanding. In: **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**. [S.l.: s.n.], 2019. p. 4171–4186.

DOCHERTY, N. et al. Maximising the translation potential of electrochemical biosensors. **Chemical Communications**, Royal Society of Chemistry, v. 61, n. 71, p. 13359–13377, 2025.

DOSOVITSKIY, A. et al. An image is worth 16x16 words: Transformers for image recognition at scale. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2021.

D'ASCOLI, S. et al. Convit: Improving vision transformers with soft convolutional inductive biases. In: PMLR. **International Conference on Machine Learning**. [S.l.], 2021. p. 2286–2296.

FANG, Y. et al. Unleashing vanilla vision transformer with masked image modeling for object detection. In: **Proceedings of the IEEE/CVF International Conference on Computer Vision**. [S.l.: s.n.], 2023. p. 6244–6253.

FARES, R. T.; GUERRA, L. B.; RIBAS, L. C. Exploring local graphs via random encoding for texture representation learning. In: **20th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications**. [S.l.]: SciTePress, 2025. v. 3, p. 200–209.

FARES, R. T.; RIBAS, L. C. A new approach to learn spatio-spectral texture representation with randomized networks: Application to brazilian plant species identification. In: SPRINGER. **International Conference on Engineering Applications of Neural Networks**. [S.l.], 2024. p. 435–449.

FARES, R. T.; RIBAS, L. C. Randomized autoencoder-based representation for dynamic texture recognition. In: IEEE. **2024 31st International Conference on Systems, Signals and Image Processing (IWSSIP)**. [S.l.], 2024. p. 1–7.

FARES, R. T.; RIBAS, L. C. Mixer: Mixed hyperspherical random embedding neural network for texture recognition. **arXiv preprint arXiv:2510.03228**, 2025.

FARES, R. T.; RIBAS, L. C. Volumetric color-texture representation for colorectal polyp classification in histopathology images. In: **20th International Conference on Computer Vision Theory and Applications**. [S.l.: s.n.], 2025. p. 210–221.

FARES, R. T. et al. Randomized encoding ensemble: A new approach for texture representation. In: **2024 31st International Conference on Systems, Signals and Image Processing (IWSSIP)**. [S.l.]: IEEE, 2024. p. 1–8.

FLORINDO, J. B. Renyi entropy analysis of a deep convolutional representation for texture recognition. **Applied Soft Computing**, Elsevier, v. 149, p. 110974, 2023.

FLORINDO, J. B. Fractal pooling: A new strategy for texture recognition using convolutional neural networks. **Expert Systems with Applications**, Elsevier, v. 243, p. 122978, 2024.

FLORINDO, J. B.; BACKES, A. R.; NECKEL, A. Elmp-net: The successive application of a randomized local transform for texture classification. **Pattern Recognition**, Elsevier, v. 153, p. 110499, 2024.

FLORINDO, J. B.; BRUNO, O. M. Fractal descriptors based on Fourier spectrum applied to texture analysis. **Physica A: Statistical Mechanics and its Applications**, Elsevier, v. 391, n. 20, p. 4909–4922, 2012.

FLORINDO, J. B.; LAUREANO, E. E. Boff: A bag of fuzzy deep features for texture recognition. **Expert Systems with Applications**, Elsevier, v. 219, p. 119627, 2023.

FLORINDO, J. B.; NECKEL, A. A randomized network approach to multifractal texture descriptors. **Information Sciences**, Elsevier, v. 648, p. 119544, 2023.

FUJIWARA, K.; HASHIMOTO, T. Neural implicit embedding for point cloud analysis. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2020. p. 11734–11743.

FUKUSHIMA, K. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. **Biological Cybernetics**, Springer, v. 36, n. 4, p. 193–202, 1980.

GANDAGLIA, G. et al. Epidemiology and prevention of prostate cancer. **European Urology Oncology**, Elsevier, v. 4, n. 6, p. 877–892, 2021.

GIDARIS, S.; SINGH, P.; KOMODAKIS, N. Unsupervised representation learning by predicting image rotations. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2018.

GONG, Y. et al. Multi-scale orderless pooling of deep convolutional activation features. In: SPRINGER. **European Conference on Computer Vision**. [S.l.], 2014. p. 392–407.

GU, J. et al. Recent advances in convolutional neural networks. **Pattern Recognition**, Elsevier, v. 77, p. 354–377, 2018.

GUI, J. et al. A survey on self-supervised learning: Algorithms, applications, and future trends. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, 2024.

GUO, Y.; ZHAO, G.; PIETIKÄINEN, M. Texture classification using a linear configuration model based descriptor. In: **British Machine Vision Conference**. [S.l.: s.n.], 2011. p. 1–10.

GUO, Z.; ZHANG, L.; ZHANG, D. A completed modeling of local binary pattern operator for texture classification. **IEEE Transactions on Image Processing**, IEEE, v. 19, n. 6, p. 1657–1663, 2010.

GUO, Z.; ZHANG, L.; ZHANG, D. Rotation invariant texture classification using lbp variance (lbpv) with global matching. **Pattern Recognition**, Elsevier, v. 43, n. 3, p. 706–719, 2010.

HAN, K. et al. A survey on vision transformer. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 45, n. 1, p. 87–110, 2022.

HARALICK, R. M.; SHANMUGAM, K.; DINSTEIN, I. Textural features for image classification. **IEEE Transactions on Systems, Man, and Cybernetics**, SMC-3, n. 6, p. 610–621, 1973.

HE, K. et al. Masked autoencoders are scalable vision learners. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2022. p. 16000–16009.

HE, K. et al. Momentum contrast for unsupervised visual representation learning. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2020. p. 9729–9738.

HE, K. et al. Deep residual learning for image recognition. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2016. p. 770–778.

HE, Y. et al. Vista3d: A unified segmentation foundation model for 3d medical imaging. In: **Proceedings of the Computer Vision and Pattern Recognition Conference**. [S.l.: s.n.], 2025. p. 20863–20873.

HENDRYCKS, D.; GIMPEL, K. Gaussian error linear units (gelus). **arXiv preprint arXiv:1606.08415**, 2016.

HINTON, G. E.; OSINDERO, S.; TEH, Y.-W. A fast learning algorithm for deep belief nets. **Neural Computation**, MIT Press, v. 18, n. 7, p. 1527–1554, 2006.

HUANG, G. et al. Densely connected convolutional networks. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2017. p. 4700–4708.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: a new learning scheme of feedforward neural networks. In: IEEE. **2004 IEEE International Joint Conference on Neural Networks**. [S.l.], 2004. v. 2, p. 985–990.

HUANG, G.-B.; ZHU, Q.-Y.; SIEW, C.-K. Extreme learning machine: theory and applications. **Neurocomputing**, Elsevier, v. 70, n. 1-3, p. 489–501, 2006.

HUBEL, D. H.; WIESEL, T. N. Receptive fields of single neurones in the cat's striate cortex. **The Journal of physiology**, v. 148, n. 3, p. 574, 1959.

HUBEL, D. H.; WIESEL, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. **The Journal of Physiology**, v. 160, n. 1, p. 106, 1962.

JAIN, A.; HEALEY, G. A multiscale representation including opponent color features for texture recognition. **IEEE Transactions on Image Processing**, IEEE, v. 7, n. 1, p. 124–128, 1998.

JING, L.; TIAN, Y. Self-supervised visual feature learning with deep neural networks: A survey. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 43, n. 11, p. 4037–4058, 2020.

KANNALA, J.; RAHTU, E. Bsif: Binarized statistical image features. In: IEEE. **Proceedings of the 21st International Conference on Pattern Recognition**. [S.l.], 2012. p. 1363–1366.

KHAN, S. et al. Transformers in vision: A survey. **ACM Computing Surveys (CSUR)**, ACM New York, NY, v. 54, n. 10s, p. 1–41, 2022.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. **Advances in Neural Information Processing Systems**, v. 25, 2012.

LE-KHAC, P. H.; HEALY, G.; SMEATON, A. F. Contrastive representation learning: A framework and review. **IEEE Access**, IEEE, v. 8, p. 193907–193934, 2020.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **Nature**, Nature Publishing Group UK London, v. 521, n. 7553, p. 436–444, 2015.

LECUN, Y. et al. Handwritten digit recognition with a back-propagation network. **Advances in Neural Information Processing Systems**, v. 2, 1989.

LECUN, Y. et al. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, n. 11, p. 2278–2324, 1998.

LI, X. et al. Omg-seg: Is one model good enough for all segmentation? In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2024. p. 27948–27959.

LI, Z. et al. A survey of convolutional neural networks: analysis, applications, and prospects. **IEEE Transactions on Neural Networks and Learning Systems**, IEEE, v. 33, n. 12, p. 6999–7019, 2021.

LIN, T.-Y.; ROYCHOWDHURY, A.; MAJI, S. Bilinear cnn models for fine-grained visual recognition. In: **Proceedings of the IEEE International Conference on Computer Vision**. [S.l.: s.n.], 2015. p. 1449–1457.

LIU, Y. et al. A survey of visual transformers. **IEEE Transactions on Neural Networks and Learning Systems**, IEEE, v. 35, n. 6, p. 7478–7498, 2023.

LIU, Z. et al. A convnet for the 2020s. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2022. p. 11976–11986.

LYRA, L. O.; FABRIS, A. E.; FLORINDO, J. B. A multilevel pooling scheme in convolutional neural networks for texture image recognition. **Applied Soft Computing**, Elsevier, v. 152, p. 111282, 2024.

MAANI, R.; KALRA, S.; YANG, Y.-H. Noise robust rotation invariant features for texture classification. **Pattern Recognition**, Elsevier, v. 46, n. 8, p. 2103–2116, 2013.

MAAS, A. L. et al. Rectifier nonlinearities improve neural network acoustic models. In: **International Conference on Machine Learning**. [S.l.: s.n.], 2013. v. 30, n. 1, p. 3.

MANJUNATH, B.; MA, W. Texture features for browsing and retrieval of image data. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, v. 18, n. 8, p. 837–842, 1996.

MATOS, J. d. et al. Large-margin representation learning for texture classification. **Pattern Recognition Letters**, Elsevier, v. 170, p. 39–47, 2023.

MATTIOLI, I. A. et al. On the challenges for the diagnosis of sars-cov-2 based on a review of current methodologies. **ACS sensors**, ACS Publications, v. 5, n. 12, p. 3655–3677, 2020.

NI, X. et al. From biology to the clinic—exploring liver metastasis in prostate cancer. **Nature Reviews Urology**, Nature Publishing Group UK London, v. 21, n. 10, p. 593–614, 2024.

NOROOZI, M.; FAVARO, P. Unsupervised learning of visual representations by solving jigsaw puzzles. In: SPRINGER. **European Conference on Computer Vision**. [S.l.], 2016. p. 69–84.

OITICICA, P. R. et al. Using machine learning and optical microscopy image analysis of immunosensors made on plasmonic substrates: Application to detect the sars-cov-2 virus. **ACS Sensors**, ACS Publications, v. 10, n. 2, p. 1407–1418, 2025.

OJALA, T. et al. Outex-new framework for empirical evaluation of texture analysis algorithms. In: **2002 International Conference on Pattern Recognition**. [S.l.]: IEEE, 2002. v. 1, p. 701–706.

OJALA, T.; PIETIKAINEN, M.; MAENPAA, T. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. **IEEE Transactions on Pattern Analysis and Machine Intelligence**, IEEE, v. 24, n. 7, p. 971–987, 2002.

OJANSIVU, V.; HEIKKILÄ, J. Blur insensitive texture classification using local phase quantization. In: SPRINGER. **International Conference on Image and Signal Processing**. [S.l.], 2008. p. 236–243.

OLIVER, A. et al. Realistic evaluation of deep semi-supervised learning algorithms. **Advances in Neural Information Processing Systems**, v. 31, 2018.

PAO, Y.-H.; PARK, G.-H.; SOBAJIC, D. J. Learning and generalization characteristics of the random vector functional-link net. **Neurocomputing**, Elsevier, v. 6, n. 2, p. 163–180, 1994.

PAO, Y.-H.; TAKEFUJI, Y. Functional-link net computing: theory, system architecture, and functionalities. **Computer**, v. 25, n. 5, p. 76–79, 1992.

PASZKE, A. et al. Pytorch: An imperative style, high-performance deep learning library. **Advances in Neural Information Processing Systems**, v. 32, 2019.

PATHAK, D. et al. Context encoders: Feature learning by inpainting. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2016. p. 2536–2544.

PENROSE, R. A generalized inverse for matrices. In: CAMBRIDGE UNIVERSITY PRESS. **Mathematical Proceedings of the Cambridge Philosophical Society**. [S.l.], 1955. v. 51, n. 3, p. 406–413.

RAYMUNDO-PEREIRA, P. A. et al. Polyphenol oxidase-based electrochemical biosensors: A review. **Analytica Chimica Acta**, Elsevier, v. 1139, p. 198–221, 2020.

RIBAS, L. C.; BRUNO, O. M. Learning a complex network representation for shape classification. **Pattern Recognition**, Elsevier, v. 154, p. 110566, 2024.

RIBAS, L. C. et al. Learning graph representation with randomized neural network for dynamic texture classification. **Applied Soft Computing**, Elsevier, v. 114, p. 108035, 2022.

RIBAS, L. C. et al. Fusion of complex networks and randomized neural networks for texture analysis. **Pattern Recognition**, Elsevier, v. 103, p. 107189, 2020.

RIBAS, L. C. et al. Color-texture classification based on spatio-spectral complex network representations. **Physica A: Statistical Mechanics and its Applications**, Elsevier, v. 635, p. 129518, 2024.

RIBAS, L. C. et al. Local complex features learned by randomized neural networks for texture analysis. **Pattern Analysis and Applications**, Springer, v. 27, n. 1, p. 23, 2024.

RODRIGUES, V. C. et al. Electrochemical and optical detection and machine learning applied to images of genosensors for diagnosis of prostate cancer with the biomarker pca3. **Talanta**, Elsevier, v. 222, p. 121444, 2021.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning representations by back-propagating errors. **Nature**, Nature Publishing Group UK London, v. 323, n. 6088, p. 533–536, 1986.

Sá Junior, J. J. d. M.; BACKES, A. R. Elm based signature for texture classification. **Pattern Recognition**, Elsevier, v. 51, p. 395–401, 2016.

Sá Junior, J. J. d. M.; BACKES, A. R. An improved randomized neural network signature for texture classification. **Available at SSRN**, p. 1–18, 2023.

Sá Junior, J. J. d. M.; BACKES, A. R. Colorectal image classification using randomized neural network. In: **19th International Conference on Computer Vision Theory and Applications**. [S.l.: s.n.], 2024. p. 800–805.

Sá Junior, J. J. d. M.; BACKES, A. R.; BRUNO, O. M. Randomized neural network based descriptors for shape classification. **Neurocomputing**, Elsevier, v. 312, p. 201–209, 2018.

Sá Junior, J. J. d. M.; BACKES, A. R.; BRUNO, O. M. Randomized neural network based signature for color texture classification. **Multidimensional Systems and Signal Processing**, Springer, v. 30, n. 3, p. 1171–1186, 2019.

Sá Junior, J. J. d. M.; BACKES, A. R.; BRUNO, O. M. Shape classification using randomized neural network descriptors extracted from the fourier spectrum. In: IEEE. **2020 International Conference on Systems, Signals and Image Processing (IWSSIP)**. [S.l.], 2020. p. 279–284.

Sá Junior, J. J. d. M.; RIBAS, L. C.; BRUNO, O. M. Randomized neural network based signature for dynamic texture classification. **Expert Systems with Applications**, Elsevier, v. 135, p. 194–200, 2019.

SCABINI, L. et al. A comparative survey of vision transformers for feature extraction in texture analysis. **Journal of Imaging**, MDPI, v. 11, n. 9, p. 304, 2025.

SCABINI, L. et al. Complex networks for modeling texture and spectral features of hyperspectral images for environmental analysis. **Computers in Biology and Medicine**, Elsevier, v. 204, p. 111504, 2026.

SCABINI, L. et al. Vortex: Challenging cnns at texture recognition by using vision transformers with orderless and randomized token encodings. **arXiv preprint arXiv:2503.06368**, 2025.

SCABINI, L. et al. Radam: Texture recognition through randomized aggregated encoding of deep activation maps. **Pattern Recognition**, Elsevier, v. 143, p. 109802, 2023.

SCABINI, L. F. et al. Multilayer complex network descriptors for color–texture characterization. **Information Sciences**, Elsevier, v. 491, p. 30–47, 2019.

SCABINI, L. F.; RIBAS, L. C.; BRUNO, O. M. Spatio-spectral networks for color-texture analysis. **Information Sciences**, Elsevier, v. 515, p. 64–79, 2020.

SCABINI, L. F. d. S. **Patterns and randomness in networks for computer vision: from graphs to neural networks**. 2023. Tese (Doutorado) — Universidade de São Paulo, 2023.

SCHIAPPA, M. C.; RAWAT, Y. S.; SHAH, M. Self-supervised learning for videos: A survey. **ACM Computing Surveys**, ACM New York, NY, v. 55, n. 13s, p. 1–37, 2023.

SCHMIDT, W. F.; KRAAIJVELD, M. A.; DUIN, R. P. Feed forward neural networks with random weights. In: IEEE COMPUTER SOCIETY PRESS. **International Conference on Pattern Recognition**. [S.l.], 1992. p. 1–1.

SCHUHMANN, C. et al. Laion-5b: An open large-scale dataset for training next generation image-text models. **Advances in Neural Information Processing Systems**, v. 35, p. 25278–25294, 2022.

SHARAN, L.; ROSENHOLTZ, R.; ADELSON, E. H. Accuracy and speed of material categorization in real-world images. **Journal of Vision**, The Association for Research in Vision and Ophthalmology, v. 14, n. 9, p. 12–12, 2014.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409.1556**, 2014.

SOARES, J. C. et al. Detection of a sars-cov-2 sequence with genosensors using data analysis based on information visualization and machine learning techniques. **Materials Chemistry Frontiers**, Royal Society of Chemistry, v. 5, n. 15, p. 5658–5670, 2021.

SUI, Y. et al. Self-supervised representation learning from random data projectors. In: **International Conference on Learning Representations**. [S.l.: s.n.], 2024.

SUN, C. et al. Revisiting unreasonable effectiveness of data in deep learning era. In: **Proceedings of the IEEE International Conference on Computer Vision**. [S.l.: s.n.], 2017. p. 843–852.

SZEGEDY, C. et al. Inception-v4, inception-resnet and the impact of residual connections on learning. In: **Proceedings of the AAAI Conference on Artificial Intelligence**. [S.l.: s.n.], 2017. v. 31, n. 1.

SZEGEDY, C. et al. Rethinking the inception architecture for computer vision. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2016. p. 2818–2826.

TOUVRON, H.; CORD, M.; JÉGOU, H. Deit iii: Revenge of the vit. In: SPRINGER. **European Conference on Computer Vision**. [S.l.], 2022. p. 516–533.

TURNER, A. P. Biosensors: sense and sensibility. **Chemical Society Reviews**, Royal Society of Chemistry, v. 42, n. 8, p. 3184–3196, 2013.

VAPNIK, V. **The Nature of Statistical Learning Theory**. [S.l.]: Springer: New York, 1999.

VASWANI, A. et al. Attention is all you need. **Advances in Neural Information Processing Systems**, v. 30, 2017.

WESZKA, J. S.; DYER, C. R.; ROSENFELD, A. A comparative study of texture measures for terrain classification. **IEEE Transactions on Systems, Man, and Cybernetics**, IEEE, n. 4, p. 269–285, 1976.

WOO, S. et al. Convnext v2: Co-designing and scaling convnets with masked autoencoders. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)**. [S.l.: s.n.], 2023. p. 16133–16142.

XUE, J.; ZHANG, H.; DANA, K. Deep texture manifold for ground terrain recognition. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2018. p. 558–567.

YAMASHITA, R. et al. Convolutional neural networks: an overview and application in radiology. **Insights into imaging**, Springer, v. 9, n. 4, p. 611–629, 2018.

YANG, J. et al. Confede: Contrastive feature decomposition for multimodal sentiment analysis. In: **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**. [S.l.: s.n.], 2023. p. 7617–7630.

YANG, Z. et al. Dfaen: Double-order knowledge fusion and attentional encoding network for texture recognition. **Expert Systems with Applications**, Elsevier, v. 209, p. 118223, 2022.

YAO, S.; WAN, X. Multimodal transformer for multimodal machine translation. In: **Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics**. [S.l.: s.n.], 2020. p. 4346–4350.

YOSINSKI, J. et al. How transferable are features in deep neural networks? **Advances in Neural Information Processing Systems**, v. 27, 2014.

ZHAI, W. et al. Deep structure-revealed network for texture recognition. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2020. p. 11010–11019.

ZHAI, W. et al. Deep multiple-attribute-perceived network for real-world texture recognition. In: **Proceedings of the IEEE/CVF International Conference on Computer Vision**. [S.l.: s.n.], 2019. p. 3613–3622.

ZHAN, X. et al. Online deep clustering for unsupervised representation learning. In: **Proceedings of the IEEE/CVF conference on computer vision and pattern recognition**. [S.l.: s.n.], 2020. p. 6688–6697.

ZHANG, H.; XUE, J.; DANA, K. Deep ten: Texture encoding network. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2017. p. 708–717.

ZHANG, J. et al. 2dmamba: Efficient state space model for image representation with applications on giga-pixel whole slide image classification. In: **Proceedings of the Computer Vision and Pattern Recognition Conference**. [S.l.: s.n.], 2025. p. 3583–3592.

ZHANG, Q. et al. Recent advances in convolutional neural network acceleration. **Neurocomputing**, Elsevier, v. 323, p. 37–51, 2019.

ZHANG, R.; ISOLA, P.; EFROS, A. A. Colorful image colorization. In: SPRINGER. **European Conference on Computer Vision**. [S.l.], 2016. p. 649–666.

ZHANG, R.; ISOLA, P.; EFROS, A. A. Split-brain autoencoders: Unsupervised learning by cross-channel prediction. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2017. p. 1058–1067.

ZHU, L. et al. Biformer: Vision transformer with bi-level routing attention. In: **Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition**. [S.l.: s.n.], 2023. p. 10323–10333.

ZHU, Z. et al. An adaptive hybrid pattern for noise-robust texture analysis. **Pattern Recognition**, Elsevier, v. 48, n. 8, p. 2592–2608, 2015.

ZHUANG, F. et al. A comprehensive survey on transfer learning. **Proceedings of the IEEE**, IEEE, v. 109, n. 1, p. 43–76, 2020.

APÊNDICE A - CONJUNTO DE DADOS DE TEXTURA

Neste apêndice, são apresentados todos os conjuntos de dados de textura usados para análise experimental das abordagens de representações propostas nesta dissertação. Em particular, diversos conjuntos de dados foram empregados na análise, uma vez que, dessa forma, torna-se possível avaliar não apenas a capacidade discriminativa dos métodos propostos, mas também sua capacidade de adaptação às características texturais dos diferentes conjuntos, analisando-se, assim, de maneira implícita, sua capacidade de generalização. Portanto, isso torna o processo de análise experimental das técnicas investigados mais robusto.

Especificamente, os conjuntos de textura empregados, os quais são apresentados posteriormente, são caracterizados como puros, uma vez que as imagens texturais que os compõem foram obtidas em ambientes controlados (e.g., laboratório), com o conteúdo textural preenchendo integralmente o campo de visão da imagem. Exemplos de conjuntos de dados classificados como puros incluem: Brodatz, Outex, USPtex, MBT, CUReT, entre outros.

Esses conjuntos diferem daqueles denominados selvagens, ou *in-the-wild*, os quais são caracterizados por serem obtidos em ambientes não controlados (e.g., na natureza). Nesses cenários, o conteúdo textural nem sempre preenche a imagem completamente e, além disso, múltiplas texturas podem estar presentes, assim como diferentes objetos. Em particular, exemplos de conjuntos de textura selvagens incluem: DTD, FMD, GTOS, GTOS-M.

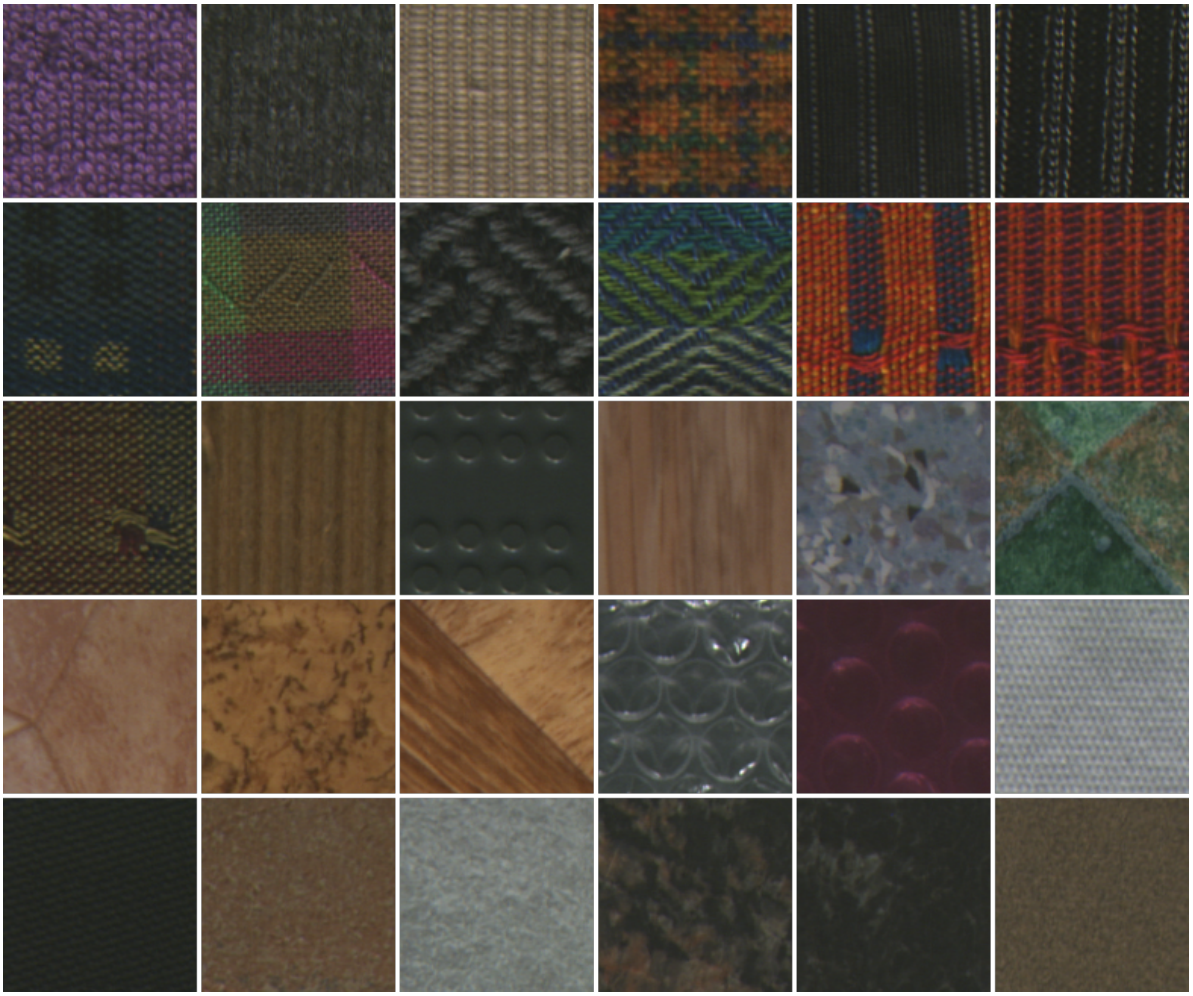
Isso posto, nesta dissertação, para a análise experimental, foram utilizadas somente os conjuntos de dados puros, uma vez que esses apresentam maior similaridade, em termos de aquisição das imagens, com os conjuntos de dados de biossensores, cujas imagens são obtidas em ambiente laboratorial. Dessa forma, a Seção A.1 introduz o conjunto de textura em nível de cinza empregado, enquanto a Seção A.2 apresenta os conjuntos de dados de textura colorida.

A.1 CONJUNTOS DE DADOS DE TEXTURA EM NÍVEL DE CINZA

Esta seção apresenta o conjunto de dados Brodatz, que corresponde ao único conjunto de texturas originalmente em nível de cinza empregado. Contudo, embora esse seja o único conjunto originalmente nesse formato, para a análise experimental dos métodos de aprendizado de textura em nível de cinza, além do Brodatz, também foram utilizados outros conjuntos de textura originalmente coloridos, os quais foram convertidos para a escala de cinza.

Brodatz. Este conjunto de dados é composto por 1776 imagens de textura de dimensão 128×128 divididas em 111 classes, de modo que, cada classe contém 16 imagens. Especificamente, essa configuração é a mesma utilizada em (Backes; Casanova; Bruno, 2013). Além disso, as imagens de textura são provenientes do álbum Brodatz proposto em Brodatz (1966).

Figura 47 – Exemplos de imagens de textura de distintas classes do conjunto de dados Outex.



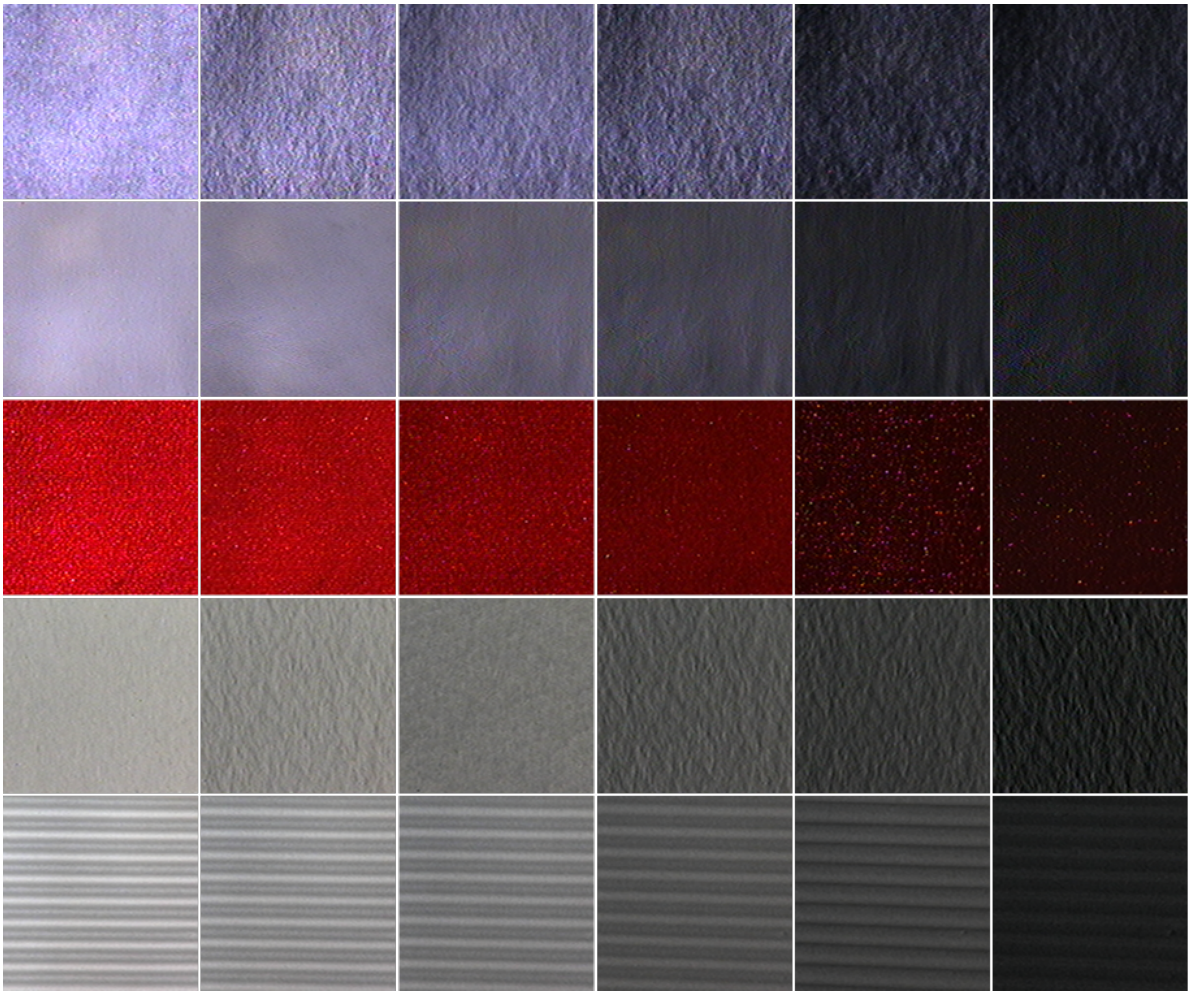
Fonte: Elaborada pelo autor. As imagens são provenientes do décimo terceiro conjunto de dados Outex proposto em Ojala et al. (2002).

A.2 CONJUNTOS DE DADOS DE TEXTURA COLORIDA

Nesta seção, são apresentados quatro conjuntos de dados de textura colorida, sendo esses: Outex, CUReT, USPtex e MBT. Conforme mencionado anteriormente, esses conjuntos de dados apresentam distintos padrões texturais e níveis de desafio, o que os torna apropriados para a avaliação dos métodos propostos. Dessa forma, as configurações empregadas, bem como as principais características de cada conjunto, são apresentadas a seguir.

Outex. Embora existam diferentes configurações do conjunto de dados de textura Outex proposto em Ojala et al. (2002), o conjunto empregado nesta dissertação corresponde ao décimo terceiro conjunto, denominado Outex13. Em particular, trata-se de um conjunto que apresenta rico conteúdo microtextural, de modo que é necessário técnicas mais eficazes para sua discriminabilidade completa (Fares; Ribas, 2025a). A configuração adotada nesta dissertação é a mesma descrita em Backes, Casanova & Bruno (2013), a qual consiste de

Figura 48 – Exemplos de imagens de textura de distintas classes sob diferentes condições de iluminação do conjunto de dados CUReT.



Fonte: Elaborada pelo autor. As imagens são provenientes do conjunto de dados CUReT (Columbia-Utrecht Reflectance and Texture) proposto em Dana et al. (1999).

1360 imagens de textura coloridas de dimensão 128×128 pixels, agrupadas em 68 classes. Por fim, a Figura 47 apresenta amostras de imagens de textura para distintas classes.

CUReT. O conjunto de dados CUReT (do inglês, *Columbia-Utrecht Reflectance and Texture Dataset* foi proposto em Dana et al. (1999) com a finalidade de estudar a aparência visual de superfícies sob diferentes condições de iluminação e pontos de vista. Entre as superfícies do mundo real empregadas para a obtenção das amostras de textura incluem-se folha de alumínio, cortiça, papel amassado, algodão, musgo, grama, entre outras. Nesta dissertação, adota-se a mesma configuração apresentada em Scabini, Ribas & Bruno (2020), na qual o conjunto é composto por 5612 imagens de textura colorida com dimensão 200×200 pixels, distribuídas em 61 classes, de modo que, cada classe contém 92 amostras. A Figura 48 apresenta exemplos de amostras de diferentes classes sob distintas condições de iluminação.

USPtex. Este conjunto de dados, construído pela Universidade de São Paulo e proposto

Figura 49 – Exemplos de imagens de textura de distintas classes do conjunto de dados USPtex.



Fonte: Elaborada pelo autor. As imagens são provenientes do conjunto de dados USPtex proposto em Backes, Casanova & Bruno (2012).

em Backes, Casanova & Bruno (2012), é composto por imagens naturais encontradas no cotidiano (e.g., plantas, paredes e vegetação). Dessa forma, as imagens capturadas apresentam distintos padrões texturais, exibindo diferentes características de irregularidade, periodicidade e aleatoriedade. Isso posto, a configuração adotada nesta dissertação é composta por 2292 imagens de textura colorida com dimensão 128×128 pixels, organizadas em 191 classes, sendo que cada classe contém 12 amostras. A Figura 49 exibe algumas amostras desse conjunto.

MBT. O conjunto MBT (do inglês, *Multi-band Texture*) foi proposto em Abdelmounaime & Dong-Chen (2013). As imagens desse conjunto são derivadas da Brodatz, de modo que as cores das amostras do MBT emergem a partir dos distintos conteúdos texturais presentes em cada canal da imagem, apresentado variações intra- e inter-canal. Dessa forma, a configuração adotada é a mesma descrita em Scabini, Ribas & Bruno (2020), na qual o conjunto é composto por 2464 amostras com dimensão de 160×160 pixels distribuídas em 154 classes.

Comentários Finais. A Tabela 18 apresenta uma descrição geral de todos os conjuntos de

Tabela 18 – Descrição geral dos conjuntos de imagens de texturas empregados nesta dissertação para a análise das abordagens de representação de textura desenvolvidas.

Conjunto	# Amostras	# Classes	# Amostras/Classe	Dimensão	Formato
Brodatz	1776	111	16	128×128	Nível de Cinza
Outex	1360	68	20	128×128	Colorida
CUReT	5612	61	92	200×200	Colorida
USPtex	2292	191	12	128×128	Colorida
MBT	2464	154	16	160×160	Colorida

Fonte: Elaborada pelo autor.

dados de imagens de textura empregados na análise experimental das abordagens de representação de textura desenvolvidas nesta dissertação, tanto em nível de cinza quanto coloridos, os quais possibilitaram a avaliação da capacidade discriminativa e de generalização dessas abordagens. Essa descrição inclui informações referentes ao número de amostras, ao número de classes, ao número de amostras por classe, às dimensões das imagens e ao formato original dos dados (i.e., nível de cinza ou colorido).

DADOS CURRICULARES

IDENTIFICAÇÃO	
	Ricardo Trivizan Fares 12/07/2001
Nacionalidade	Brasileira
Nome em citações bibliográficas:	FARES, R. T.; FARES, RICARDO; FARES, RICARDO T.
Currículo Lattes	http://lattes.cnpq.br/8140393013060301
ORCID	https://orcid.org/0000-0001-8296-8872
FORMAÇÃO ACADÊMICA	
2020/2023	Bacharelado em Ciência da Computação Universidade Estadual Paulista (UNESP) Instituto de Biociências, Letras e Ciências Exatas (IBILCE)
2024/2026	Mestrado em Ciência da Computação Universidade Estadual Paulista (UNESP) Instituto de Biociências, Letras e Ciências Exatas (IBILCE)
PRODUÇÃO BIBLIOGRÁFICA	
FARES, Ricardo T.; MANACERO, A.; BARROSO, João P. R.; GENOVA, Lucas E.; JUIZ, Gustavo T.; LOBATO, Renata S.; SPOLON, Roberta. Simulação de Sistemas Exascale usando Time Warp. In: **Escola Regional de Alto Desempenho de São Paulo (ERAD-SP)**. SBC, 2023. p. 29-32.	
BARROSO, João P. R.; MANACERO, A.; FARES, Ricardo T.; PEREIRA, Daphne L. H.; DE JESUS, Bruno V. V.; LOBATO, Renata S.; SPOLON, Roberta. Interface Gráfica Para Simulação de Sistemas Exascale. In: **Escola Regional de Alto Desempenho de São Paulo (ERAD-SP)**. SBC, 2023. p. 33-36.	
SCABINI, Leonardo; ZIELINSKI, Kallil; FARES, Ricardo; KONUK, Emir; KOLB, Rosana; RIBAS, Lucas; BRUNO, Odemir. Deep Texture Feature Aggregation on Leaf Microscopy Images for Brazilian Plant Species Recognition. In: **Proceedings of the 2024 9th International Conference on Machine Learning Technologies (ICMLT)**. 2024. p. 209-213.	

FARES, Ricardo T.; RIBAS, Lucas C. A new approach to learn spatio-spectral texture representation with randomized networks: Application to brazilian plant species identification. In: **International Conference on Engineering Applications of Neural Networks (EANN)**. Cham: Springer Nature Switzerland, 2024. p. 435-449.

FARES, Ricardo T.; VICENTIM, Ana C. M.; SCABINI, Leonardo; ZIELISNKI, Kallil M.; JENANNE, Rachid; BRUNO, Odemir M.; RIBAS, Lucas C. Randomized encoding ensemble: A new approach for texture representation. In: **2024 31st International Conference on Systems, Signals and Image Processing (IWSSIP)**. IEEE, 2024. p. 1-8.

FARES, Ricardo T.; RIBAS, Lucas C. Randomized Autoencoder-based Representation for Dynamic Texture Recognition. In: **2024 31st International Conference on Systems, Signals and Image Processing (IWSSIP)**. IEEE, 2024. p. 1-7.

RIBAS, Lucas C.; CASACA, Wallace; FARES, Ricardo T. Conditional generative adversarial networks and deep learning data augmentation: a multi-perspective data-driven survey across multiple application fields and classification architectures. **AI**, v. 6, n. 2, p. 32, 2025.

FARES, Ricardo T.; GUERRA, Luan B.; RIBAS, Lucas C. Exploring local graphs via random encoding for texture representation learning. In: **International Conference on Computer Vision Theory and Applications (VISAPP)**. SciTePress, 2025. p. 200-209.

FARES, Ricardo T.; RIBAS, Lucas C. Volumetric Color-Texture Representation for Colorectal Polyp Classification in Histopathology Images. In: **International Conference on Computer Vision Theory and Applications (VISAPP)**. SciTePress, 2025. p. 210-221.

SCABINI, Leonardo; ZIELINSKI, Kallil M.; FERNANDES, Marília; FARES, Ricardo T.; RIBAS, Lucas C.; KOLB, Rosana, M.; BRUNO, Odemir M. Complex networks for modeling texture and spectral features of hyperspectral images for environmental analysis. **Computers in Biology and Medicine**, v. 204, p. 111504, 2026.

PARTICIPAÇÃO EM EVENTOS CIENTÍFICOS

XIV Escola Regional de Alto Desempenho do Estado de São Paulo (ERAD-SP), 2023, (São José dos Campos). Simulação de Sistemas Exascale usando Time Warp. 2023. (Congresso).

XXXV Congresso de Iniciação Científica (CIC), 2023, (São José do Rio Preto). Simulação de Sistemas Exascale usando Time Warp. 2023. (Congresso).