



**UNIVERSIDADE FEDERAL DE UBERLÂNDIA**  
**FACULDADE DE ENGENHARIA ELÉTRICA**  
**PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

**PEDRO MOISÉS DE SOUSA**

**ESTRATÉGIAS DE CLASSIFICAÇÃO DE IMAGENS RADIOLÓGICAS**  
**UTILIZANDO REDES NEURAS CONVOLUCIONAIS E TRANSFOR-**  
**MADA *WAVELET***

Uberlândia

2022

PEDRO MOISÉS DE SOUSA

**ESTRATÉGIAS DE CLASSIFICAÇÃO DE IMAGENS RADIOLÓGICAS  
UTILIZANDO REDES NEURAS CONVOLUCIONAIS E TRANSFOR-  
MADA *WAVELET***

Tese de Doutorado submetida ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial à obtenção do título de Doutor em Ciências.

Área de concentração: Engenharia Biomédica

Orientador: Prof. Dra. Ana Claudia Patrocínio

Uberlândia

2022

Ficha Catalográfica Online do Sistema de Bibliotecas da UFU  
com dados informados pelo(a) próprio(a) autor(a).

S725  
2022 Sousa, Pedro Moises de, 1973-  
ESTRATÉGIAS DE CLASSIFICAÇÃO DE IMAGENS RADIOLÓGICAS  
UTILIZANDO REDES NEURAIAS CONVOLUCIONAIS E TRANSFORMADA  
WAVELET [recurso eletrônico] / Pedro Moises de Sousa. -  
2022.

Orientadora: Ana Claudia Patrocinio.  
Tese (Doutorado) - Universidade Federal de Uberlândia,  
Pós-graduação em Engenharia Elétrica.  
Modo de acesso: Internet.  
Disponível em: <http://doi.org/10.14393/ufu.te.2022.433>  
Inclui bibliografia.

1. Engenharia elétrica. I. Patrocinio, Ana Claudia ,  
1976-, (Orient.). II. Universidade Federal de  
Uberlândia. Pós-graduação em Engenharia Elétrica. III.  
Título.

CDU: 621.3

Bibliotecários responsáveis pela estrutura de acordo com o AACR2:  
Gizele Cristine Nunes do Couto - CRB6/2091  
Nelson Marcos Ferreira - CRB6/3074


**UNIVERSIDADE FEDERAL DE UBERLÂNDIA**

Faculdade de Engenharia Elétrica

Av. João Naves de Ávila, 2121, Bloco 3N - Bairro Santa Mônica, Uberlândia-MG, CEP 38400-902

Telefone: (34) 3239-4701/4702 - www.feelt.ufu.br - feelt@ufu.br


**ATA DE DEFESA - PÓS-GRADUAÇÃO**

Programa de Pós-Graduação em:	Engenharia Elétrica				
Defesa de:	Tese de Doutorado, 302, PPGEELT				
Data:	vinte e nove de julho de dois mil e vinte e dois	Hora de início:	08:00	Hora de encerramento:	11:30
Matrícula do Discente:	11823EEL004				
Nome do Discente:	Pedro Moises de Sousa				
Título do Trabalho:	ESTRATÉGIAS DE CLASSIFICAÇÃO DE IMAGENS RADIOLÓGICAS UTILIZANDO REDES NEURAIIS CONVOLUCIONAIS E TRANSFORMADA WAVELET				
Área de concentração:	Processamento da Informação				
Linha de pesquisa:	Engenharia Biomédica				
Projeto de Pesquisa de vinculação:	N/A				

Reuniu-se por meio de videoconferência a Banca Examinadora designada pelo Colegiado do Programa de Pós-graduação em Engenharia Elétrica, assim composta: Professores Doutores: Adriano de Oliveira Andrade - FEELT/UFU; Tulio Augusto Alves Macedo - FAMED/UFU; Ana Maria Marques da Silva - PUCRS; Júlio César Ferreira - IF GOIANO; Ana Claudia Patrocinio - FEELT/UFU, orientador(a) do(a) candidato(a).

Iniciando os trabalhos o(a) presidente da mesa, Dr(a). Ana Claudia Patrocinio, apresentou a Comissão Examinadora e o candidato(a), agradeceu a presença do público, e concedeu ao Discente a palavra para a exposição do seu trabalho. A duração da apresentação do Discente e o tempo de arguição e resposta foram conforme as normas do Programa.

A seguir o senhor(a) presidente concedeu a palavra, pela ordem sucessivamente, aos(às) examinadores(as), que passaram a arguir o(a) candidato(a). Ultimada a arguição, que se desenvolveu dentro dos termos regimentais, a Banca, em sessão secreta, atribuiu o resultado final, considerando o(a) candidato(a):

Aprovado(a).

Esta defesa faz parte dos requisitos necessários à obtenção do título de Doutor

O competente diploma será expedido após cumprimento dos demais requisitos, conforme as normas do Programa, a legislação pertinente e a regulamentação interna da UFU.

Nada mais havendo a tratar foram encerrados os trabalhos. Foi lavrada a presente ata que após lida e achada conforme foi assinada pela Banca Examinadora.



Documento assinado eletronicamente por **Tulio Augusto Alves Macedo, Professor(a) do Magistério Superior**, em 29/07/2022, às 11:23, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Ana Maria Marques da Silva, Usuário Externo**, em 29/07/2022, às 11:23, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Júlio César Ferreira, Usuário Externo**, em 29/07/2022, às 11:24, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Adriano de Oliveira Andrade, Professor(a) do Magistério Superior**, em 29/07/2022, às 11:26, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



Documento assinado eletronicamente por **Ana Claudia Patrocínio, Professor(a) do Magistério Superior**, em 29/07/2022, às 11:26, conforme horário oficial de Brasília, com fundamento no art. 6º, § 1º, do [Decreto nº 8.539, de 8 de outubro de 2015](#).



A autenticidade deste documento pode ser conferida no site [https://www.sei.ufu.br/sei/controlador\\_externo.php?acao=documento\\_conferir&id\\_orgao\\_acesso\\_externo=0](https://www.sei.ufu.br/sei/controlador_externo.php?acao=documento_conferir&id_orgao_acesso_externo=0), informando o código verificador **3797810** e o código CRC **2BF6EABD**.

*Dedico este trabalho para meu filho Rafael Henrique Ferreira de Sousa*

## AGRADECIMENTOS

Primeiramente agradeço a Deus pela benção de chegar até aqui.

Agradeço ao Frei Beniamino de Luca pela ajuda nos estudos e ensinamentos de vida.

À minha orientadora Ana Patrocínio pela sinceridade, ética, pelo bom coração, por ensinar a trabalhar em equipe, pelo exemplo de respeito pelos alunos, por sua dedicação e por ter me recebido de braços abertos para orientação. Ao professor Adriano de Oliveria Andrade por abrir a oportunidade para eu cursar o doutorado.

À minha esposa Livia Aparecida Ferreira, por ser a pessoa excepcional que é, estando sempre ao meu lado me motivando nos momentos que mais foram necessários.

À minha família por serem as melhores pessoas que alguém poderia ter. A minha mãe Aurora Constantino de Lima, meu irmão William Chaves de Souza Carvalho, minha enteada Laura Danielle Ferreira Silva e meu filho Rafael Henrique Ferreira de Sousa por serem meu braço direito e que cuidam tão bem de mim e me aguentam em todas as crises de ansiedade durante o progresso do doutorado.

Aos meus colegas do Biolab (“Aquário”) pela ajuda durante a execução do trabalho, aos meus amigos Pedro Cunha Carneiro, Gabriel Fernandes Cyrino, Juscelino Cardoso Marciano dos Santos, Leandro Henrique Furtado Pinto Silva, Daniel Stefany Duarte Caetano, Mariane Modesto Oliveira, Gabrielle Macedo Pereira, Carlos Alberto da Costa Junior, Luis Vinicius de Moura, Christian Mattjie que me ajudaram de forma direta/indireta neste trabalho.

Aos grupos de pesquisa "Ciências" e "COVID" por tantas dúvidas respondidas, ideias e sugestões durante as reuniões semanais.

À Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), por ter me ajudado a viabilizar financeiramente este período, uma vez que o presente trabalho foi realizado com seu apoio.

À Universidade Federal de Uberlândia (UFU).

Ao Hospital de Clínicas da UFU (HCU-UFU).

Ao programa de Pós-graduação em Engenharia Elétrica da Universidade Federal de Uberlândia (PPGEELT).

Aos professores do PPGEELT, especialmente o professor Keiji Yamanaka que instigou os estudos sobre Redes Neurais.

À Pontificia Universidade Católica do Rio Grande do Sul (PUCRS), pela parceria e apoio irrestritos.

Ao Hospital São Lucas da PUCRS.

À Universidade Federal de Viçosa - campus Rio Paranaíba, pela lealdade, parceria, amizade, coleguismo, oportunidade e apoio irrestritos.

E a todos aqueles que contribuíram de alguma forma para a execução deste trabalho, cujos nomes me escapam, meu muito obrigado.

## EPÍGRAFE

*Comece fazendo o que é necessário, depois o que é possível, e de repente você estará fazendo o impossível.*

(São Francisco de Assis)

# SUMÁRIO

<i>RESUMO</i> .....	<i>xiii</i>
<i>ABSTRACT</i> .....	<i>xiv</i>
<i>LISTA DE FIGURAS</i> .....	<i>xv</i>
<i>LISTA DE TABELAS</i> .....	<i>xviii</i>
<i>LISTA DE ABREVIATURAS E SIGLAS</i> .....	<i>xx</i>
<i>GLOSSARIO</i> .....	<i>xxii</i>
<b>1. INTRODUÇÃO</b> .....	<b>23</b>
<b>1.1. Objetivo Geral</b> .....	<b>25</b>
<b>1.2. Objetivos Específicos</b> .....	<b>25</b>
<b>1.3. Justificativa</b> .....	<b>25</b>
<b>1.4. Organização da tese</b> .....	<b>26</b>
<b>2. ESTADO DA ARTE</b> .....	<b>27</b>
<b>3. REFERENCIAL TEÓRICO</b> .....	<b>36</b>
<b>3.1. Redes Neurais</b> .....	<b>36</b>
<b>3.1.1. Ideia do Gradiente Descendente</b> .....	<b>36</b>
<b>3.1.2. Gradient Descent (GD)</b> .....	<b>38</b>
<b>3.1.3. Stochastic Gradient Descent (SGD)</b> .....	<b>38</b>
<b>3.1.4. Batch Gradient Descent</b> .....	<b>39</b>
<b>3.1.5. Otimização do GD</b> .....	<b>39</b>
<b>3.1.5.1. AdaGrad</b> .....	<b>40</b>
<b>3.1.5.2. AdaDelta</b> .....	<b>40</b>
<b>3.1.5.3. RMSPro</b> .....	<b>41</b>
<b>3.1.5.4. ADAM</b> .....	<b>41</b>
<b>3.1.6. Backpropagation</b> .....	<b>42</b>
<b>3.1.7. Convolutional Neural Network (CNN)</b> .....	<b>46</b>
<b>3.1.8. Camada Convolutacional</b> .....	<b>47</b>
<b>3.1.9. Funções de ativação</b> .....	<b>49</b>
<b>3.1.10. Camada de pooling</b> .....	<b>50</b>
<b>3.1.11. Fully connected layer</b> .....	<b>51</b>
<b>3.1.12. Dropout</b> .....	<b>51</b>
<b>3.1.13. Treinamento de uma CNN</b> .....	<b>52</b>
<b>3.2. TRANSFORMADAS WAVELET</b> .....	<b>54</b>
<b>3.2.1. Introdução</b> .....	<b>54</b>

3.2.2.	Transformada <i>wavelet</i> contínua (CWT) .....	55
3.2.3.	Transformada discreta de <i>wavelet</i> .....	57
3.2.4.	Conceito de Análise Multiresolução.....	59
3.2.5.	Transformada <i>wavelet</i> em duas dimensões .....	60
<b>4.</b>	<b>PRIMEIRA CONTRIBUIÇÃO: CLASSIFICAÇÃO COVID-19 EM RADIOGRAFIAS DE TÓRAX USANDO UMA NOVA REDE NEURAL CONVOLUCIONAL: CNN-COVID .....</b>	<b>62</b>
4.1.	<i>Introdução</i> .....	62
4.2.	<i>Trabalhos Relacionados</i> .....	64
4.3.	<i>Materiais e métodos</i> .....	66
4.3.1.	Descrição do banco de imagens I.....	66
4.3.2.	Descrição do banco de imagens II .....	67
4.3.3.	Metodologia.....	67
4.3.3.1.	<i>Desenvolvimento de estrutura CNN-COVID</i> .....	67
4.3.3.2.	<i>Treinamento da CNN-COVID</i> .....	68
4.3.3.3.	<i>Parâmetro da entrada CNN-COVID</i> .....	69
4.4.	<i>Resultados</i> .....	70
4.5.	<i>Discussões</i> .....	72
4.6.	<i>Conclusões</i> .....	73
<b>5.</b>	<b>SEGUNDA CONTRIBUIÇÃO: POR UM MODELO DE CLASSIFICAÇÃO USANDO CNN E WAVELETS APLICADOS A IMAGENS DE TC COVID-19 .....</b>	<b>74</b>
5.1.	<i>Introdução</i> .....	74
5.2.	<i>Estrutura da WCNN-COVID</i> .....	76
5.2.1.	Decomposição da Wavelet.....	77
5.2.2.	Modelo WCNN-COVID .....	79
5.2.3.	Treinamento, teste e validação da WCNN-COVID.....	81
5.2.4.	Complexidade do Algoritmo .....	82
5.3.	<i>Workflow de Integração da WCNN-COVID</i> .....	83
5.3.1.1.	<i>Criação da base de imagens</i> .....	83
5.3.1.2.	<i>Pré-processamento da imagem</i> .....	84
5.3.1.3.	<i>Teste de Execução</i> .....	84
5.3.1.4.	<i>Resultados</i> .....	84
5.3.2.	Discussão dos resultados .....	87
5.3.3.	Conclusão dos Resultados .....	88
<b>6.</b>	<b>ESTUDO DE ABLAÇÃO .....</b>	<b>90</b>

<b>6.1.</b>	<b><i>WCNN padrão</i></b> .....	<b>90</b>
<b>6.1.1.</b>	<b><i>Base de dados BIMCV</i></b> .....	<b>91</b>
<b>6.1.2.</b>	<b><i>Teste das famílias wavelet discretas e coeficientes</i></b> .....	<b>92</b>
	6.1.2.1. <i>Escolha da família wavelet</i> .....	92
	6.1.2.2. <i>Análise dos coeficientes da transformada wavelet discreta</i> .....	94
<b>6.1.3.</b>	<b><i>Teste das técnicas de otimização</i></b> .....	<b>95</b>
	6.1.3.1. <i>Stochastic Gradient Descent (SGD)</i> .....	96
	6.1.3.2. <i>RMSPROP</i> .....	96
	6.1.3.3. <i>ADAM</i> .....	97
	6.1.3.4. <i>Resultados da aplicação dos métodos de otimização</i> .....	97
<b>6.1.4.</b>	<b><i>Teste das camadas convolucionais</i></b> .....	<b>98</b>
	6.1.4.1. <i>Teste com 3 camadas convolucionais</i> .....	98
	6.1.4.2. <i>Teste com 2 camadas convolucionais</i> .....	99
	6.1.4.3. <i>Teste com 1 camada convolucional</i> .....	99
<b>6.1.5.</b>	<b><i>Teste de Pooling</i></b> .....	<b>100</b>
<b>6.1.6.</b>	<b><i>Teste das camadas densas</i></b> .....	<b>100</b>
<b>6.2.</b>	<b><i>Teste de eficácia com a base ChestXpert</i></b> .....	<b>105</b>
<b>6.2.1.</b>	<b><i>DenseNet121</i></b> .....	<b>108</b>
<b>6.2.2.</b>	<b><i>Parâmetros</i></b> .....	<b>108</b>
<b>6.2.3.</b>	<b><i>Resultados</i></b> .....	<b>109</b>
<b>6.3.</b>	<b><i>Teste de resizing com a base DRIVE</i></b> .....	<b>109</b>
<b>6.3.1.</b>	<b><i>Primeiro passo</i></b> .....	<b>112</b>
<b>6.3.2.</b>	<b><i>Segundo passo</i></b> .....	<b>112</b>
<b>6.3.3.</b>	<b><i>Terceiro passo</i></b> .....	<b>113</b>
<b>6.3.4.</b>	<b><i>Quarto passo</i></b> .....	<b>113</b>
<b>6.3.5.</b>	<b><i>Quinto passo</i></b> .....	<b>114</b>
<b>6.3.6.</b>	<b><i>Resultados e discussão</i></b> .....	<b>115</b>
<b>6.4.</b>	<b><i>Discussão</i></b> .....	<b>118</b>
<b>6.4.1.</b>	<b><i>Discussão sobre as famílias wavelets discretas e coeficientes</i></b> .....	<b>118</b>
<b>6.4.2.</b>	<b><i>Discussão sobre as técnicas de otimização</i></b> .....	<b>118</b>
<b>6.4.3.</b>	<b><i>Discussão sobre o teste das camadas convolucionais</i></b> .....	<b>119</b>
<b>6.4.4.</b>	<b><i>Discussão sobre as técnicas de Pooling</i></b> .....	<b>119</b>
<b>6.4.5.</b>	<b><i>Discussão sobre o teste das camadas densas</i></b> .....	<b>120</b>
<b>6.4.6.</b>	<b><i>Discussão sobre o teste de eficiência com a base ChestXpert</i></b> .....	<b>121</b>
<b>6.4.7.</b>	<b><i>Discussão sobre o teste de resizing com a base DRIVE</i></b> .....	<b>122</b>

6.4.8.	<i>Arquitetura do modelo resultante da ablação.....</i>	<i>122</i>
7.	<b>O MODELO WCNN .....</b>	<b>125</b>
7.1.	<i>Criação do modelo WCNN.....</i>	<i>125</i>
7.1.1	Ambiente tecnológico .....	127
7.1.2	Wave layer.....	127
7.1.3	Extraction layer .....	128
7.1.4	Fully connected layer.....	128
7.1.5	Treinamento do WCNN .....	129
7.2.	<i>Criação das Bases de Imagens .....</i>	<i>129</i>
7.2.1.	Formato da base.....	129
7.2.2.	Anonimização das imagens .....	129
7.2.3.	Critérios de inclusão .....	130
7.2.4.	Critérios de exclusão .....	130
7.2.5.	Definição de bases internas e/ou externas .....	130
7.2.6.	Definição do percentual para treinamento, teste e validação.....	131
7.3.	<i>Validação do modelo e apresentação dos resultados.....</i>	<i>131</i>
7.3.1.	<i>Matriz de confusão .....</i>	<i>132</i>
7.3.2.	<i>Curva ROC.....</i>	<i>132</i>
8.	<b>TERCEIRA CONTRIBUIÇÃO: UM NOVO MODELO PARA CLASSIFICAÇÃO DE IMAGENS MÉDICAS DE TC USANDO CNN E WAVELET: UM ESTUDO DE CASO COM COVID-19 .....</b>	<b>134</b>
8.1.	<i>Introdução.....</i>	<i>134</i>
8.2.	<i>Trabalhos Relacionados.....</i>	<i>135</i>
8.3.	<i>Wavelet.....</i>	<i>138</i>
8.3.1.	<i>Wavelet mãe e nível de decomposição .....</i>	<i>139</i>
8.3.2.	<i>Análise dos coeficientes da decomposição .....</i>	<i>140</i>
8.4.	<i>Workflow.....</i>	<i>140</i>
8.5.	<i>Criação da Base de Imagens .....</i>	<i>141</i>
8.5.1.	Critérios de inclusão e exclusão.....	141
8.5.2.	Dataset I.....	142
8.5.3.	Dataset II.....	143
8.5.4.	Dataset III .....	144
8.6.	<i>Modelo WCNN.....</i>	<i>145</i>
8.6.1.	Camada Wave .....	145
8.6.2.	<i>Feature extraction .....</i>	<i>147</i>

8.6.3. <i>Flattening layer</i> .....	148
8.6.4. <i>Fully connected layer</i> .....	148
8.7. <i>Parâmetros do treinamento e métricas de avaliação</i> .....	150
8.8. <i>Resultados do Modelo WCNN</i> .....	151
8.8.1. Resultados do <i>dataset I</i> .....	151
8.8.2. Resultados do <i>dataset II</i> .....	153
8.8.3. Resultados do <i>dataset III</i> .....	154
8.9. <i>Resultados Consolidados</i> .....	155
8.10. <i>Discussão</i> .....	156
8.11. <i>Conclusão</i> .....	158
9. CONCLUSÃO .....	159
REFERÊNCIAS.....	162

## RESUMO

A irrupção pandêmica de COVID-19 motivou a efetivação mundial de esforços maciços para enfrentar o problema, como a simplificação de protocolos de acesso a repositórios de dados e metadados do novo vírus e da doença. As buscas por pesquisas sugeriram em diversas áreas desde pesquisas bioquímicas, biológicas, investigação sorológica até engenharia genética e tecnologia da informação. Na área da inteligência artificial, as técnicas de *deep learning* foram utilizadas em busca de ferramentas de apoio que contribuíssem para o enfrentamento da pandemia. A possibilidade de reduzir erros da análise das imagens radiológicas de tórax era uma finalidade, pois elas complementavam o exame médico da doença. O objetivo é propor um novo modelo de classificação de imagens médicas, baseado em CNN, utilizando wavelet como parte da camada de entrada da rede, de maneira a incorporar o pré-processamento das imagens no modelo, evitando as técnicas de redimensionamento de imagem para serem inseridas a rede. Assim, a pandemia foi a primeira motivação para a criação do modelo WCNN (*Wavelet Convolutional Neural Network*) e a segunda motivação foi a observação do uso de técnicas de redimensionamento das imagens médicas para se adequarem a modelos prontos da literatura, o que pode causar distorções ou perda de informações na detecção da doença em estudo. WCNN foi baseado em uma Rede Neural Convolutiva (CNN) e transformada *wavelet*. O modelo propõe uma camada de entrada customizada, chamada *Wave Layer*, que processa as imagens sem redimensioná-las. Para avaliar a WCNN, foi realizado um experimento que exemplifica seu comportamento, utilizando um conjunto de imagens de Tomografia Computadorizada de tórax de indivíduos diagnosticados com COVID-19 e outras infecções pulmonares. O resultado das métricas Acurácia (ACC), Sensibilidade (Sen) e Especificidade (Sp) foram 0,9819, 0,9783 e 0,98, respectivamente. Conclui-se que estes resultados expressivos indicam que a associação de CNNs e transformadas *wavelets* é promissora para a criação de modelos de classificação.

*Palavras-chave:* radiografias de tórax, TC, redes neurais convolucionais, COVID-19, *wavelets*, WCNN

## ABSTRACT

The outbreak of the COVID-19 pandemic has motivated massive worldwide efforts to tackle the problem, such as the simplification of protocols for accessing data repositories and metadata of the new virus and the disease. Searches for research suggest in several areas from biochemical, biological research, serological investigation to genetic engineering and information technology. In the artificial intelligence field, deep learning techniques were used in search of support tools that would contribute to facing the pandemic. The possibility of reducing errors in the analysis of chest radiological images was a goal of such research, as they complemented the medical examination of the disease. The objective is to propose a new medical image classification model, based on CNN, using wavelet as part of the input layer of the network, in order to incorporate the pre-processing of images in the model, avoiding image resizing techniques to be inserted into the network. Thus, the pandemic was the first motivation for the creation of the WCNN model and the second motivation was the observation of the use of techniques for resizing medical images to adapt to ready-made models in the literature, which can cause distortions or loss of information in the detection of the disease under study. WCNN was based on a Convolutional Neural Network (CNN) and wavelet transform. The model proposes a custom input layer, called Wave Layer, which processes the images without resizing them. To assess WCNN, an experiment was performed that exemplifies its behavior, using a set of chest CT (Computed Tomography) images from persons diagnosed with COVID-19 and other lung infections. The result of the metrics Accuracy (ACC), Sensitivity (Sen) and Specificity (Sp) were 0.9819, 0.9783 and 0.98, respectively. Hence, it can be concluded that these expressive results indicate that the association of CNNs and wavelet transforms is promising for the creation of classification models.

*Keywords:* chest X-ray, CT, convolutional neural networks, COVID-19, *wavelets*, WCNN

## LISTA DE FIGURAS

Figura 1 – Ideia do gradiente descendente. ....	37
Figura 2 – Ideia da derivada no gradiente descendente.....	37
Figura 3 – Direção da derivada.....	38
Figura 4 – Diagrama esquemático de uma rede neural, mostrando a notação para as camadas: entrada, oculta e saída, juntamente com os pesos relacionados. Adaptado de (HERTZ, KROGH e PALMER, 2018).....	43
Figura 5 – Diagrama esquemático do algoritmo de <i>backpropagation</i> .....	46
Figura 6 – Exemplo de uma rede neural profunda CNN. Fonte: Adaptado de DUC-JACQUET, 2017 .....	47
Figura 7 – Diagrama esquemático da operação de convolução. Fonte: WANI, BHAT, <i>et al.</i> , 2020. ....	48
Figura 8 – Exemplo de convolução entre uma entrada $5 \times 5$ e um <i>kernel</i> $2 \times 2$ . As figuras (a), (b), (c) e (d) correspondem ao resultado do processamento da convolução da primeira linha (PONTI e DA COSTA, 2018).	48
Figura 9 – Função de ativação <i>sigmóide</i> . Fonte: ( PONTI e DA COSTA, 2018). ....	49
Figura 10 – Função de ativação <i>tanh</i> . Fonte: ( PONTI e DA COSTA, 2018). ....	50
Figura 11 – <i>Rectified Linear Unit</i> (ReLU): Tem saída 0 se a entrada for $< 0$ . Se a entrada for $> 0$ , a saída é igual à entrada. Fonte: (PONTI e DA COSTA, 2018).....	50
Figura 12 – <i>Max pooling</i> . ....	51
Figura 13 – Efeito <i>dropout</i> : a) uma rede neural simples, b) uma rede neural após o <i>dropout</i> . ....	52
Figura 14 – Divisão da base de Imagens .....	53
Figura 15 – (a) <i>Wavelet</i> mãe, (b) $\psi_{t\alpha}: 0 < \alpha < 1$ , (c) $\psi_{t\alpha}: \alpha > 1$ . Fonte: (ACHARYA e RAY, 2005) .....	56
Figura 16 – Transformação <i>wavelet</i> bidimensional. (a) Aplica-se uma etapa de transformação unidimensional a todas as linhas. (b) Em seguida, o mesmo procedimento é repetido para todas as colunas. (c) Nesta etapa, os coeficientes resultantes de uma convolução são expandidos em ambas as direções. Fonte: Adaptado de (JANSEN, 2012) .....	61
Figura 17 – Exemplo de decomposição de um nível da DWT .....	61
Figura 18 – Esquema de classificação da rede neural profunda.....	68
Figura 19 – Escolha do tamanho da entrada da CNN-COVID (200X200 ... 300X300) .....	69

Figura 20 – Conjunto de dados I e II: resultados de cada teste e média geral (Avg) de todos os 10 testes. ....	70
Figura 21 – Matriz de confusão da CNN-COVID em que (A) a é a matriz do conjunto de dados I e (B) é a matriz do conjunto de dados II .....	71
Figura 22 – Curva ROC da CNN-COVID: conjunto de dados I/AUC = 0,972; b conjunto de dados Dataset II/AUC = 0,988. ....	71
Figura 23 – Histograma das Imagens Chest X-ray, componentes aproximação, vertical e diagonal. ....	78
Figura 24 – Processo de decomposição da <i>Wavelet</i> . ....	79
Figura 25 – Esquema de classificação da rede neural profunda.....	79
Figura 26 – Escolha do tamanho da Entrada da WCNN-COVID (200 × 200 ... 300 × 300). ....	81
Figura 27 – Encadeamento dos melhores pesos da WCNN-COVID. ....	82
Figura 28 – Diagrama do método proposto para classificação de COVID-19.....	83
Figura 29 – Matriz de confusão da WCNN-COVID. ....	84
Figura 30 – Curvas ROC do WCNN-COVID: A) Conjunto de imagens sem wavelet/AUC = 0.9229 B) Conjunto de imagens com wavelet /AUC=0.995. ....	85
Figura 31 – Resultado da acurácia da WCNN-COVID: resultados para cada ciclo e média geral para os 10 ciclo. ....	86
Figura 32 – Função Probabilidade Normal.....	86
Figura 33 – Teste com as famílias da <i>wavelet</i> .....	94
Figura 34– Teste com os coeficientes da <i>wavelet</i> .....	95
Figura 35 – Curva ROC da WCNN padrão.....	102
Figura 36 – Curva ROC da WCNN modificada-2 teste.....	103
Figura 37 – Curva ROC da WCNN modificada-3 teste.....	104
Figura 38 – Curva ROC da WCNN modificada-4 teste.....	105
Figura 39 – Radiografia frontal e lateral de um mesmo paciente.....	106
Figura 40 – Exemplo de imagem da base DRIVE com seu respectivo <i>ground truth</i> .....	109
Figura 41 – <i>Workflow</i> do experimento com a base DRIVE .....	111
Figura 42 – Exemplo dos limites da imagem recortada .....	112
Figura 43 – Exemplo da imagem original e suas respectivas ROIs .....	113
Figura 44 – Transformada <i>wavelet</i> e seus coeficientes de uma imagem do olho.....	114
Figura 45 – Alguns resultados da aplicação da WCNN.....	116
Figura 46 – Efeito do resizing do <i>ground truth</i> .....	117

Figura 47 – Sistema <i>deep learning</i> desenvolvido para reconhecimento de rosto humano. Fonte MURALI, 2018 .....	121
Figura 48 – Arquitetura da WCNN padrão .....	124
Figura 49 – A Estrutura de uma CNN é realizada com dois módulos: um extrator de recursos e um classificador treinável (adaptado de LECUN, BOTTOU, <i>et al.</i> , 1998).....	125
Figura 50 – Diagrama ilustrativo da metodologia, fases e etapas da criação do modelo WCNN .....	126
Figura 51 – O espaço ROC e suas interpretações ( PRATES, 2020) .....	133
Figura 52 – Diagrama geral da transformada <i>wavelet</i> .....	139
Figura 53 – Visão geral do <i>workflow</i> e o modelo WCNN.....	141
Figura 54 – Seleção das fatias da TC que serão utilizadas.....	142
Figura 55 – A estrutura base de uma WCNN é composta por 3 módulos, além dos 2 já conhecidos na literatura, o novo módulo denominado WAVE layer, compõe o modelo. ....	145
Figura 56 – Esquema de classificação WCNN.....	145
Figura 57 – Seleção da Região de Interesse (ROI).....	146
Figura 58 – Camada wave da WCN .....	147
Figura 59 – WCNN <i>training loss</i> e <i>training accuracy</i> do <i>dataset I</i> .....	152
Figura 60 – Matriz de confusão do <i>dataset I</i> .....	152
Figura 61 – Curva ROC do <i>dataset I</i> .....	153
Figura 62 – WCNN <i>training loss</i> e <i>training accuracy</i> do <i>dataset II</i> .....	153
Figura 63 – Matriz de confusão do <i>dataset II</i> .....	154
Figura 64 – Curva ROC do <i>dataset II</i> .....	154
Figura 65 – Média da aurácia dos <i>datasets</i> internos ( <i>dataset I</i> e <i>dataset II</i> ) e <i>dataset</i> externo ( <i>dataset III</i> ).....	155
Figura 66 – Exemplo de histograma da imagem original e seus correspondentes coeficientes diagonal, vertical e aproximada .....	157

## LISTA DE TABELAS

Tabela 1 – Exemplos de arquitetura de CNNs pré-treinadas.....	27
Tabela 2 – Quadro de referência e características de trabalhos relacionados à criação de modelos baseados em CNN para detecção e classificação de doenças.....	34
Tabela 3 – Número de imagens do conjunto de dados I.....	66
Tabela 4 – Número de imagens do conjunto de dados II. ....	67
Tabela 5 – Arquitetura da CNN-COVID. A rede contém a entrada (I), Convolução (C), <i>Max pooling</i> (M), Rede Totalmente Conectada (F).....	68
Tabela 6 – Resultados das métricas .....	71
Tabela 7 – Comparação da CNN-COVID com o estado da arte.....	72
Tabela 8 Arquitetura da WCNN-COVID. A rede contém a entrada (I), Convolução (C), <i>Max-Pooling</i> (M), Rede Totalmente Conectada (F) .....	80
Tabela 9: Distribuição da Base de dados utilizada na WCNN-COVID .....	84
Tabela 10: Resultado das métricas .....	85
Tabela 11: Comparação dos valores das métricas do método WCNN-COVID com o estado da arte dos métodos de classificação considerando imagens de tomográfica computadorizada de tórax.....	87
Tabela 12 – Arquitetura padrão da WCNN.....	91
Tabela 13 – Distribuição das imagens do repositório BIMCV .....	92
Tabela 14 – Teste com as tags iniciais e finais de cada família <i>wavelet</i> .....	93
Tabela 15 – Teste da escolha dos Coeficientes .....	95
Tabela 16 – Valor de alfa ( $\alpha$ ) para os testes das técnicas de otimização .....	98
Tabela 17 – Resultado dos testes com três métodos otimizadores .....	98
Tabela 18 – Arquitetura da WCNN: três camadas convolucionais, destacadas em azul .....	98
Tabela 19 – Arquitetura da WCNN: duas camadas convolucionais, destacadas em azul.....	99
Tabela 20 – Arquitetura da WCNN: uma camada convolucional, destacadas em azul .....	99
Tabela 21 – Resultado dos testes utilizando as diferentes técnicas de <i>pooling</i> .....	100
Tabela 22 - Arquitetura da WCNN padrão.....	101
Tabela 23 – Resultados das métricas do teste da WCNN padrão.....	101
Tabela 24 – Arquitetura da WCNN modificada-2 teste .....	102
Tabela 25 – Resultados das métricas da Arquitetura da WCNN modificada-2 teste.....	102
Tabela 26 – Arquitetura da WCNN modificada-3 teste .....	103
Tabela 27 – Resultados das métricas da Arquitetura da WCNN modificada-3 teste.....	104

Tabela 28 – Arquitetura da WCNN modificada-4 teste .....	104
Tabela 29 – Resultados das métricas da Arquitetura da WCNN modificada-3 teste .....	105
Tabela 30 – Divisão da base de dados ChestX-ray14 .....	107
Tabela 31 – Informações sobre a bases de dados ChetXpert.....	108
Tabela 32 – Comparação dos modelos .....	109
Tabela 33 – Teste de verificação da ordem dos coeficientes.....	118
Tabela 34 – Teste nas camadas convolucionais .....	119
Tabela 35 – Teste com número variável de camadas densas.....	120
Tabela 36 – Exemplo de uma matriz de confusão .....	132
Tabela 37 – Distribuição das imagens do repositório BIMCV .....	143
Tabela 38 – Distribuição das imagens do <i>dataset</i> II.....	144
Tabela 39 – Distribuição das imagens do <i>dataset</i> III .....	144
Tabela 40 – Arquitetura do WCNN é composta por <i>Wave Layer</i> (W), <i>Convolucional</i> (C), <i>Max pooling</i> (M) e <i>Fully Connected Layer</i> (F).....	150
Tabela 41 – Métricas de avaliação do modelo.....	151
Tabela 42 – Resultado das métricas para <i>dataset</i> I do BIMCV.....	152
Tabela 43 – Resultado das métricas para <i>dataset</i> II .....	154
Tabela 44 – Resultado das métricas para <i>dataset</i> III- HC-UFU .....	155
Tabela 45 – Resultado consolidado das métricas .....	155
Tabela 46 – Modelo WCNN <i>versus</i> modelos do estado da arte.....	156

## LISTA DE ABREVIATURAS E SIGLAS

2019-nCoV	2019 <i>newCoronavirus</i>
+ssRNA virus	<i>Positive-sense single-stranded RNA virus</i>
2019-nCoV	2019- <i>novel CoronaVirus</i>
2019nCoVr	<i>Novel Coronavirus Resource</i>
ACC	Acurácia
ADAM	<i>ADaptive Moment Estimation</i>
ARN	Ácido Ribonucleico
AUC	<i>Area Under the Curve</i>
BIMCV	<i>Medical Imaging Databank of the Valencia Region</i>
CAD	<i>Computer-Aided Detection</i>
CLBP	<i>Completed Local Binary Patterns (CLBP)</i>
CLCM	<i>Gray-Level Co-occurrence Matrix (GLCM),</i>
CNN	<i>Convolutional Neural Network</i>
CoV	CoronaVirus
Covid-19	<i>COronaVirus Disease 2019</i>
CWT	<i>Continous Wavelet Transform</i>
CXRs	<i>Chest X-Rays</i>
DWT	<i>Discrete Wavelet Transform</i>
FN	Falso Negativo
FP	Falso Positivo
GLCM	<i>Gray Level Co-occurrence Matrix</i>
GLRLM	<i>Gray Level Run Length Matrix</i>
GLSZM	<i>Gray-Level Size Zone Matrix</i>
HCU-UFU	Hospital de Clínicas da Universidade Federal de Uberlândia
HER2	<i>Human Epidermal Growth Factor Receptor 2</i>
HSL-PUCRS	Hospital São Lucas - Pontifícia Universidade Católica do Rio Grande do Sul
IA	Inteligência Artificial
IARC	<i>International Agency for Research on Cancer</i>
LBP	<i>Local Binary Patterns</i>

LDP	<i>Local Directional Pattern</i>
MA	MicroAneurisma
MERS	<i>Middle East Respiratory Syndrome</i>
MET	Microscópio Eletrônico de Transmissão
mini-MIAS	<i>mini-Mammographic Image Analysis Society</i>
NCBI	<i>National Center for Biotechnological Informations</i>
OMS	Organização Mundial de Saúde
P	Precisão
RELU	<i>REctified Linear Unit</i>
RGB	<i>Red, Green, Blue</i>
ROC	<i>Receiver Operating Characteristic</i>
RT-PCR	<i>Reverse Transcription – Polymerase Chain Reaction</i>
SARS	<i>Severe Acute Respiratory Syndrome</i>
SARS-CoV-2	<i>Severe Acute Respiratory Syndrome Coronavirus 2</i>
Se	Sensibilidade
SIFT	<i>Scale Invariant Feature Transform</i>
Sp	Especificidade
STFT	<i>Short Time Fourier Transform</i>
SuReMaPP	<i>Suspicious Region Detection on Mammogram from PP</i>
SVM	<i>Support Vector Machine</i>
TC	Tomografia Computadorizada
TMAD	<i>Stanford Tissue Microarray Database</i>
TWD	Transformada Discreta de <i>Wavelet</i>
UFU	Universidade Federal de Uberlândia
VN	Verdadeiro Negativo
VP	Verdadeiro Positivo
WCNN	<i>Wavelet CNN</i>
WHCV	<i>WuHan CoronaVirus</i>
WHO	<i>World Health Organization</i>
WSI	<i>Whole-Slide Image</i>

## GLOSSÁRIO

<i>Backpropagation</i>	Retropropagação do erro
<i>Batch Size</i>	Tamanho do lote
<i>Data Augmentation</i>	Aumento de dados
<i>Deep learning</i>	Aprendizado profundo
<i>Deep learning network</i>	Rede de aprendizado profundo
<i>Deep neural network</i>	Rede neural profunda
<i>Extraction layer</i>	Camada de extração
<i>Feature extraction</i>	Extração de características
<i>Fully connected layer</i>	Camadas totalmente conectadas
<i>Gradient descent</i>	Gradiente descendente
<i>Ground truth</i>	Valor esperado
<i>Machine learning</i>	Aprendizado de máquinas
<i>Multiple-depth CNNs</i>	CNNs de profundidade múltipla
<i>Neural network</i>	Rede neural
<i>Resizing</i>	Redimensionamento
<i>Training Accuracy</i>	Acurácia no treinamento
<i>Training Loss</i>	Perda no treinamento
<i>Wave layer</i>	Camada Wave

## 1. Introdução

---

Tecnologia e medicina têm seguido caminhos paralelos nas últimas décadas. Avanços tecnológicos estão mudando o conceito de saúde, e as necessidades de saúde estão influenciando o desenvolvimento de tecnologia. A Inteligência Artificial (IA) é composta por uma série de algoritmos lógicos, suficientemente treinados, a partir dos quais as máquinas são capazes de tomar decisões, para casos específicos, com base em regras gerais. Essa tecnologia tem aplicações no diagnóstico e acompanhamento de pacientes com avaliação prognóstica individualizada (ÁVILA-TOMÁS, MAYER-PUJADAS e QUESADA-VARELA, 2020).

A IA começou a ser incorporada na medicina e em áreas relacionadas a ela, para melhorar o atendimento ao paciente, agilizar processos e obter maior precisão diagnóstica, abrindo caminho para fornecer melhor assistência e auxílio à diagnósticos (ÁVILA-TOMÁS, MAYER-PUJADAS e QUESADA-VARELA, 2020).

Uma área que contribui com estudos promissores sobre prognóstico de doenças é a de processamento digital de imagens médicas. Pesquisas nesta área são realizadas, por exemplo, utilizando radiografias de tórax e Tomografia Computadorizada (ZHANG, XIE, *et al.*, 2020, DAI, ZHANG, *et al.*, 2020; OZTURK, OZKAYA e BARSTUGAN, 2020).

Além das contribuições específicas da área de processamento digital de imagens médicas, pesquisadores têm desenvolvido técnicas baseadas em *deep learning*<sup>1</sup>, para classificar imagens radiológicas, as quais permitem, por exemplo, criação de modelos computacionais compostos por camadas de processamento, capazes de representar dados em variados níveis de abstração (ZHANG, XIE, *et al.*, 2020, LECUN, BENGIO e HINTON, 2015; MARTIN, HANSON, *et al.*, 2020).

As Redes Neurais Convolucionais (CNNs), inspiradas em córtex visual (HERTZ, KROGH e PALMER, 2018), se destacam para processar e classificar imagens médicas, mediante reconhecimento de padrões. CNNs são capazes de reconhecer padrões difíceis de serem detectados por especialistas humanos, por exemplo, estágios iniciais de doenças em amostras de tecido (BALAS, ROY, *et al.*, 2019). A capacidade das CNNs foi utilizada por pesquisadores

---

<sup>1</sup> *Deep learning* faz parte de uma ampla família de métodos de aprendizagem de máquina, os quais se baseiam em redes neurais artificiais, que, por sua vez, fazem parte do contexto mais amplo de IA (ZHANG, XIE, *et al.*, 2020, SETHY e BEHERA, 2020; WANG, LIN e WONG, 2020; ABBAS, ABDELSAMEA e GABER, 2020; CHEN, WU, *et al.*, 2020).

para auxiliar no prognóstico da *CoronaVirus Disease 2019* (Covid-19). O primeiro caso oficial da doença reportado foi o de um paciente hospitalizado em 12 de dezembro de 2019, em Wuhan, província de Hubei, China. Os exames mostraram a existência de um vírus no fluido pulmonar do paciente, e isto ocasionou a descoberta de um novo tipo de *CoronaVirus* (CoV) pertencente à família *Coronaviridae*. O vírus foi inicialmente designado *WuHan CoronaVirus* (WHCV), posteriormente 2019-nCoV e finalmente SARS-CoV-2 (WU, HUI, *et al.*, 2020).

A irrupção pandêmica de COVID-19 motivou a efetivação mundial de esforços maciços para enfrentar o problema, como a simplificação de protocolos de acesso a repositórios de dados e metadados do novo vírus e da doença. Alguns repositórios cujo acesso foi permitido são o 2019 *Novel Coronavirus Resource* (2019nCoV) (WU, 2020) e o *National Center for Biotechnological Informations* (NCBI) (SHERRY, WARD, *et al.*) (DOS S RIBEIRO, VAN ROODE, *et al.*, 2018; SIMON, CLAASSEN, *et al.*, 2005; RIBEIRO, KOOPMANS e HARINGHUIZEN, 2018).

A demanda por pesquisas e urgência de soluções, também foi motivação para este trabalho, pensando em ações que poderiam ser conduzidas para auxiliar no enfrentamento da pandemia. Disto resultou a criação do modelo de classificação, objeto desta tese, que foi chamado de WCNN. A título de experimento, ele foi usado para classificar imagens radiológicas de pacientes com COVID-19 e de outras doenças pulmonares, infecciosas ou inflamatórias, como pneumonia, cardiomegalia, derrame pleural, atelectasia, consolidação, etc.

O uso de modelos de classificação baseados em CNN para tratar imagens médicas, exige que as imagens disponíveis sejam processadas para se adequarem aos requisitos de entrada da rede, como tamanho, formato do arquivo e número de canais de cores.(TRIWIJOYOA e ADILA, 2021 URBANIAK e WOLTER, 2021). Por exemplo, quando as imagens são redimensionadas para adequar seu tamanho ao requerido pela CNN, há perda de informações que podem ser relevantes na detecção da doença em questão (TRIWIJOYOA e ADILA, 2021 URBANIAK e WOLTER, 2021), isso pode representar um dos grandes desafios, que é evitar perda de informação e conseguir trabalhar imagens com tamanho adequado para serem processados pela rede. Outro desafio ao se trabalhar com modelos de redes CNN é a generalização do modelo para diversas aplicações, seja classificação ou segmentação de imagens.

Assim, o uso de técnicas de aprendizado de máquinas pode resultar em modelos de classificação promissores e de alta acurácia. O que sempre é bem-vindo em soluções de problemas em imagens médicas e auxílio ao diagnóstico por imagem.

## 1.1. Objetivo Geral

O objetivo geral deste trabalho é propor um novo modelo de classificação de imagens médicas, baseado em CNN, utilizando wavelet como parte da camada de entrada da rede, de maneira a incorporar o pré-processamento das imagens no modelo, evitando as técnicas de redimensionamento de imagem para serem inseridas a rede.

## 1.2. Objetivos Específicos

Os objetivos específicos deste trabalho são:

1. Implementar uma camada de entrada para uma rede neural convolucional que processe imagens sem redimensioná-las.
2. Avaliar o método proposto mediante experimentos, utilizando bases de imagens radiológicas, públicas e privadas.
3. Comparar o modelo desenvolvido com os do estado-da-arte encontrados na pesquisa bibliográfica.
4. Investigar o desempenho do modelo WCNN utilizando uma análise de ablação, utilizando um problema e classificação com imagens de tomografia computadorizada e outro com imagens de radiografia de raios X.

## 1.3. Justificativa

As arquiteturas de rede derivada do padrão CNN exigem que as imagens que serão processadas pelos modelos sejam do mesmo tamanho que o padrão definido para entrada da rede. O tamanho das imagens que cada modelo baseado em CNN processa é pré-definido, e dentre estes modelos destacam-se *AlexNet* (KRIZHEVSKY, SUTSKEVER e HINTON, 2012), *VGG-Net* (SIMONYAN e ZISSERMAN, 2014), *ResNet* (HE, ZHANG, *et al.*, 2016), *GoogLeNet* (SZEGEDY, LIU, *et al.*, 2015), *Inception V2/V3* (SZEGEDY, VANHOUCKE, *et al.*, 2016), *Inception V4* (SZEGEDY, IOFFE, *et al.*, 2017), *Inception-ResNet* (SZEGEDY, IOFFE, *et al.*, 2017), *DenseNet* (HUANG, LIU, *et al.*, 2017) e *Exception* (CHOLLET, 2017).

A Tabela 1, no capítulo Estado da Arte, mostra o tamanho padrão de entrada para cada uma dessas redes. A predefinição do tamanho da entrada exige que as imagens médicas sejam redimensionadas, conforme diversos trabalhos da literatura (BRUNO, ARDIZZONE, *et al.*, 2020, LUO, WEN, *et al.*, 2022, SAVELLI, BRIA, *et al.*, 2020, WANG, SHAO, *et al.*, 2022, ASSARI, MAHLOOJIFAR e AHMADINEJAD, 2022, DEMIR, YILMAZ e KOSE, 2019, KASSANI, KASSANI, *et al.*, 2019, LIN, ZHAO, *et al.*, 2020, HASAN, BAO, *et al.*, 2021,

EMARA, AFIFY, *et al.*, 2019, CHEN, CHEN, *et al.*, 2021). No entanto, ao redimensioná-las, informações relevantes para a classificação podem ser perdidas.

Portanto uma ferramenta de classificação baseada em CNN que não necessite de um tamanho fixo, ou seja, que não seja necessário redimensionamento da imagem é bastante interessante e útil, quando se trata de aplicações com imagens médicas.

#### **1.4. Organização da tese**

O documento é subdividido em capítulos, seções e subseções. O conteúdo de cada capítulo é descrito abaixo:

- Capítulo 2: apresenta o estado da arte da literatura com respeito aos trabalhos relacionados a esta pesquisa.
- Capítulo 3: apresenta o referencial teórico da tese.
- Capítulo 4: apresenta a primeira contribuição, que é “Classificação COVID-19 em imagens de raios X do tórax usando uma nova rede neural convolucional: CNN-COVID”.
- Capítulo 5: apresenta a segunda contribuição, que é “Por um modelo de classificação usando CNN e Wavelets aplicados a imagens de TC de COVID-19”.
- Capítulo 6: apresenta a análise de ablação do modelo.
- Capítulo 7: apresenta o modelo WCNN.
- Capítulo 8: apresenta a terceira contribuição: Um novo modelo para classificação de imagens médicas de TC usando CNN e wavelet: um estudo de caso com COVID-19.
- Capítulo 9: apresenta as conclusões e discute direções futuras para pesquisa.

## 2. Estado da Arte

---

Os métodos de *deep learning* tornaram-se populares por seu alto desempenho na classificação e detecção de eventos em tarefas de visão computacional. CNNs são amplamente utilizadas em atividades de processamento digital de imagens médicas. Neste capítulo são apresentados trabalhos, no estado da arte, em que a arquitetura da solução empregada se baseia em CNNs pré-treinadas.

Para isto, foi conduzida uma revisão bibliográfica sobre as arquiteturas CNN recentes e mais utilizadas atualmente pela comunidade acadêmica: *AlexNet* (KRIZHEVSKY, SUTSKEVER e HINTON, 2012), *VGGNet* (SIMONYAN e ZISSERMAN, 2014), *ResNet* (HE, ZHANG, *et al.*, 2016), *GoogleNet* (SZEGEDY, LIU, *et al.*, 2015), *Inception V2/V3* (SZEGEDY, VANHOUCKE, *et al.*, 2016), *Inception V4* (SZEGEDY, IOFFE, *et al.*, 2017), *Inception-ResNet* (SZEGEDY, IOFFE, *et al.*, 2017), *DenseNet* (HUANG, LIU, *et al.*, 2017) e *Xception* (CHOLLET, 2017). Os dados obtidos estão expostos na Tabela 1.

Tabela 1 – Exemplos de arquitetura de CNNs pré-treinadas

Nome da Arquitetura	Ano de Criação	Tamanho	Técnicas de Regularização	
<i>AlexNet</i>	2012	227×227	<i>Data Augmentation</i>	<i>Dropout</i>
<i>VGGNet</i>	2014	224×224	<i>Data Augmentation</i>	<i>Dropout</i>
<i>ResNet</i>	2014	224×224	<i>Data Augmentation</i>	<i>Dropout</i>
<i>GoogleNet</i>	2015	224×224	<i>Data Augmentation</i>	<i>Dropout</i>
<i>Inception V2/V3</i>	2015	299×299	<i>Data Augmentation</i>	<i>Dropout</i>
<i>Inception V4</i>	2016	299×299	<i>Data Augmentation</i>	<i>Dropout</i>
<i>Inception-ResNet</i>	2016	299×299	<i>Data Augmentation</i>	<i>Dropout</i>
<i>DenseNet</i>	2017	224×224	<i>Data Augmentation</i>	<i>Dropout</i>
<i>Xception</i>	2017	299×299	<i>Data Augmentation</i>	<i>Dropout</i>

As arquiteturas baseadas em CNN criadas entre 2012 e 2017 são exibidas na Tabela 1, porém deve-se ressaltar que outras variações têm sido desenvolvidas ao longo dos anos seguintes. Nesta tabela são exibidos o Nome da Arquitetura, Ano de Criação, Tamanho e Técnicas de Regularização. A partir daqui, são descritos os resultados da revisão bibliográfica feita para identificar o estado da arte dos estudos relacionados a este trabalho.

A apresentação do estado da arte se inicia pelo trabalho realizado por Bruno, Ardizzone *et al.* (BRUNO, ARDIZZONE, *et al.*, 2020), sobre detecção de lesões em mamografias. Entre a população feminina global, o câncer de mama é o câncer mais comumente diagnosticado e o

que mais mata dentre os cânceres. A comunidade científica tem feito muitos esforços, nas últimas décadas, para melhorar a precisão diagnóstica do câncer de mama em mulheres. A leitura de mamografias é um trabalho demorado, cansativo e exigente. Algo em torno de 30% dos cânceres são perdidos em mamografias (falsos negativos), contudo, estudos realizados pela Agência Internacional de Pesquisa sobre o Câncer, ou *International Agency for Research on Cancer* (IARC), mostraram que, quando o médico radiologista submete a mamografia à Sistemas computadorizados de auxílio ao diagnóstico, ou *Computer Aided Diagnostic* (CAD), a sensibilidade do radiologista aumenta.

Dentre as mulheres que moram em regiões que possuem programas de prevenção e rastreamento de câncer de mama, o risco de morrer da doença é de 20% a 40% menor em comparação com mulheres de regiões onde estas ações preventivas não existem (BRUNO, ARDIZZONE, *et al.*, 2020).

Os autores avaliaram as habilidades de inferência de conhecimento a partir de mamografias e propuseram uma nova solução baseada na combinação de descritores de *Scale Invariant Feature Transform* (SIFT) e em *deep learning*, com o objetivo de detectar regiões suspeitas em imagens de mamografia, utilizando *PyramidNet* e *AlexNet*. As redes foram pré-treinadas com mamografias digitais geradas por diferentes equipamentos. Técnicas de *data augmentation* e de *resizing* da imagem foram utilizadas. Para a *PyramidNet*, as imagens foram redimensionadas para  $224 \times 224$  pixels e para a *AlexNet*, o tamanho foi definido como  $227 \times 227$  pixels. As etapas de pré-processamento, extração e seleção de atributos foram executadas pelo método baseado em SIFT, ao passo que a *deep learning network* foi responsável pela validação das regiões suspeitas detectadas pelo método SIFT. As bases mini *Mammographic Image Analysis Society* (*mini-MIAS*) e *Suspicious Region Detection on Mammogram from PP* (*SuReMaPP*) foram utilizadas no experimento, num total de 384 mamografias digitais. Os passos de execução dos experimentos foram seguidos utilizando 384 mamografias digitais dos conjuntos de dados públicos *mini-MIAS* e *SuReMaPP*. A média dos valores obtidos foi 98% de sensibilidade e 90% de especificidade no *SuReMaPP* e 94% de sensibilidade e 91% de especificidade no *mini-MIAS* (BRUNO, ARDIZZONE, *et al.*, 2020).

Outro trabalho, que também utilizou *AlexNet*, foi o de Yingying Gao e seus colegas (LUO, WEN, *et al.*, 2022), que objetivava examinar a capacidade de classificação da disfunção da glândula meibomiana, ou *Meibomian Gland Dysfunction* (MGD) usando ceratografia 5M e *AlexNet*. O trabalho buscou melhorar a capacidade diagnóstica da doença do olho seco, ou *Dry Eye Disease* (DED), uma doença invasiva da superfície ocular que afeta a acuidade visual, causa

desconforto ocular e outros sintomas. A MGD, uma anormalidade crônica e difusa das glândulas meibomianas, comumente caracterizada por obstrução do ducto terminal e/ou alterações qualitativas/quantitativas na secreção glandular. Isso pode resultar em alteração do filme lacrimal e inflamação da superfície ocular, causando irritação ocular e até mesmo causando destruição da função visual por danos na córnea. Como a MGD desempenha papel importante na DED, a mensuração da perda glandular é clinicamente relevante para o diagnóstico da doença do olho seco.

Um total de 4.609 imagens de glândulas meibomianas foram obtidas de dois mil pacientes do Hospital Afiliado da Universidade Médica de Fujian, China. A base de dados para reconhecimento da disfunção da glândula palpebral, foi construída por pré-processamento, rotulagem, corte e aumento das imagens originais. Os efeitos de diferentes métodos de otimização, taxas de aprendizado, métodos de *dropout* e tamanhos de lotes na precisão do reconhecimento foram discutidos. A pesquisa mostrou que o efeito do reconhecimento do modelo é otimizado quando o método de otimização empregado é o *ADaptive Moment Estimation* (ADAM), com 150 épocas, taxa de aprendizado 0,001 e *batch size* igual a 80. Nestas condições, a acurácia média do teste do grau de saúde da pálpebra foi de 94,00 % (LUO, WEN, *et al.*, 2022).

Savelli *et al.* propuseram um novo método para a detecção de pequenas lesões em diferentes modalidades de imagens médicas digitais em (SAVELLI, BRIA, *et al.*, 2020). A abordagem da pesquisa é baseada em um conjunto multicontextual de CNNs, com o objetivo de aprender níveis distintos de contexto espacial da imagem, para melhorar o desempenho da detecção das micro lesões. A principal inovação do modelo é o uso de *multiple-depth CNNs* inspiradas em *VGGNet*, com as redes treinadas individualmente em trechos de imagem de diferentes dimensões, posteriormente combinados. Assim, o conjunto final pode encontrar e localizar anormalidades nas imagens, explorando as características locais e o contexto circundante de uma lesão.

Os experimentos foram focados em dois problemas de detecção médica bem conhecidos: *i*) detecção de microcalcificação (MCs) em mamografias digitais de campo total (FFDM); e *ii*) detecção de microaneurisma (MAs) em imagens de fundo ocular. Os autores utilizaram os repositórios públicos de imagens INbreast e E-optha. O trabalho utilizou técnicas de *data augmentation* e *resizing* para  $192 \times 192$ . A arquitetura proposta alcançou uma sensibilidade média de 83,54% e 81,62% para microcalcificações e micro aneurismas, respectivamente (SAVELLI, BRIA, *et al.*, 2020).

A prevalência e a letalidade do câncer de mama em mulheres, como previamente mencionado por Bruno, Ardizzone *et al.* no trabalho (BRUNO, ARDIZZONE, *et al.*, 2020), são reiteradas por Wang, Shao *et al.* em (WANG, SHAO, *et al.*, 2022), que estudaram o problema de criação do plano de tratamento clínico para pacientes com câncer, o que deve ser feito de acordo com o tipo patológico e grau histológico da doença. Isto é uma tarefa desafiadora, porque a morfologia patológica do câncer de mama é complexa, podendo existir mais de dois tipos de câncer no mesmo tecido canceroso. Assim, para cada tipo de câncer, o tratamento e respectivo prognóstico são diferentes. Atualmente, com o desenvolvimento de *scanners* digitais de alta resolução, a imagem de lâmina inteira, ou *Whole-Slide Image* (WSI) permite obter altas ampliações das lâminas de vidro, o que permite analisar extensivamente as características complexas das células e proteínas. Uma das possibilidades criada pela WSI é a amplificação do gene do receptor do fator de crescimento epidérmico humano-2, ou *Human Epidermal Growth Factor Receptor-2* (HER2). Como a proteína associada ao HER2 existe em um número considerável de pacientes com câncer de mama, sua identificação é interpretada como um indicador de prognóstico independente, que pode desempenhar um papel orientador na escolha do tratamento. Neste cenário, o objetivo do trabalho (WANG, SHAO, *et al.*, 2022) foi classificar imagens de microscopia óptica com imunohistoquímica PARA O HER2. Para isto, os autores utilizaram imagens HER2 do repositório *Stanford Tissue Microarray Database* (TMAD) como dados experimentais e propuseram o algoritmo de classificação de imagens chamado *HER2-ResNet*, o qual usa uma arquitetura inspirada em *ResNet*. Cerca de 3000 imagens de cortes foram utilizadas, sendo 2600 cortes usados na fase de treinamento e 400 na fase de teste. O trabalho utilizou técnicas de *data augmentation* e *resizing* dos cortes para 64×64. O modelo HER2-ResNet alcançou um acurácia de 93%, sensibilidade de 92% e especificidade de 91%.

O trabalho de Assari, Zahra *et al.* (ASSARI, MAHLOOJIFAR e AHMADINEJAD, 2022) propõe um sistema CAD bimodal, baseado em *GoogleNet* usando os repositórios públicos *Digital Database for Mammography Screening* (DDSM) e *Breast Ultrasound Images Dataset* (BUSI). O sistema proposto combina informações de imagens mamográficas e ultrassonográficas para classificar a massa mamária sólida. Cada modalidade é inicialmente treinada usando dois modelos monomodais distintos. Em seguida, um modelo bimodal é treinado, usando os mapas de recursos extraídos de ambas as modalidades. Para que os descritores BI-RADS possam ser explorados plenamente, representações diferentes da imagem de cada massa são obtidas e usadas como entrada. Os experimentos foram realizados utilizando técnicas de *data augmentation* e *resizing* para 100×100. Os resultados obtidos foram acurácia de 90,38%, sensibilidade de 90,91% e especificidade de 89,87%.

O trabalho de Demir, Ahmet *et al.* (DEMIR, YILMAZ e KOSE, 2019) se concentra na identificação de lesões típicas de câncer de pele. Este é o tipo de câncer mais comum no mundo e sua incidência está aumentando gradualmente. A exposição direta à luz do sol é ponto crítico para surgimento de câncer de pele e, portanto, o risco de ocorrência desse tipo de câncer pode ser diminuído por meio da redução do tempo de exposição. A detecção do câncer de pele é feita pelos médicos, inicialmente, por observações das lesões a olho nu, e posteriormente, por exames de dermatoscopia, que geram imagens ampliadas com grande resolução das lesões. A acurácia deste método varia de 75 a 80%, e o diagnóstico exato do câncer de pele depende de exame patológico do tecido. Foram utilizadas imagens da base pública *ISIC-Archive*, sendo 2.437 imagens para treinamento, 660 imagens para teste e 200 imagens para validação. As arquiteturas de *deep learning ResNet-101* e *Inception-V3* foram usadas para a tarefa de classificação. O *resizing* das imagens as padronizaram em  $224 \times 224$ . Os resultados atingiram acurácia 89% e 90% para as redes ResNet101 e Inception V3, respectivamente.

Outro trabalho Emara, Afify *et al.* (EMARA, AFIFY, *et al.*, 2019) também aborda uma pesquisa na detecção do câncer de pele. O trabalho utilizou o modelo *Inception-V4*, adaptado para classificar imagens extraídas do conjunto de dados *Human Against Machine with 10000 Training Images* (HAM 10000). O modelo proposto reutiliza recursos usando uma longa conexão residual em que os elementos extraídos das camadas anteriores são concatenados com aqueles das camadas de alto nível, aumentando, assim, o desempenho de classificação do modelo. O trabalho utilizou as técnicas de *data augmentation* e *resizing* das imagens para  $299 \times 299$ . A arquitetura proposta alcançou acurácia de 94,47%.

O trabalho de Kassani *et al.* visa detectar a retinopatia diabética RD precocemente, pois se a doença for detectada em estágio inicial e tratada imediatamente, a ocorrência de cegueira pode ser reduzida (KASSANI, KASSANI, *et al.*, 2019). A retinopatia diabética (RD) é uma das principais causas de cegueira em pacientes com diabetes tipo II e é caracterizada por dano progressivo crônico da micro vasculatura retiniana. De acordo com dados de 2017, 9,4% da população dos EUA tem diabetes que requer exames regulares para prevenir deficiência visual e cegueira. Dados do Brasil não foram localizados. A RD aparece quando a glicose no sangue danifica os vasos sanguíneos da retina. O aumento do nível de glicose no sangue faz com que as artérias da retina enfraqueçam e vazem para o olho, levando a uma visão embaçada. No estágio seguinte, os vasos sanguíneos fracos recém-formados se rompem e vazam sangue para o olho, levando à perda da visão.

Na pesquisa, os autores utilizaram um método baseado na agregação de camadas profundas, que combina recursos multiníveis de diferentes camadas convolucionais da arquitetura

*Xception*. O desempenho do método proposto foi validado no conjunto de dados do concurso Kaggle APTOS 2019. A classificação e identificação precisas da doença tendem a ser uma tarefa desafiadora devido à existência de diversos tipos de instrumentos de imagem de fundo de olho. Além disso, a qualidade das imagens do fundo de olho com DR podem sofrer alterações por oclusão, sombra, reflexão ou iluminação deficiente, dificultando a diferenciação entre regiões saudáveis e anormais. O trabalho utilizou técnicas de *resizing*, equalizando-as em  $299 \times 299$ . O método desenvolvido foi chamado modelo *Xception* modificado e os testes mostraram que a sua acurácia de classificação foi de 83,09% *versus* 79,59% da arquitetura *Xception* original. A sensibilidade e a especificidade no modelo modificado foram 88,24% e 87,00%, respectivamente, enquanto os valores destas métricas, para a arquitetura original foram 82,35% e 86,32% (KASSANI, KASSANI, *et al.*, 2019).

O trabalho *Lin, Chengchuang et al.* (LIN, ZHAO, *et al.*, 2020), de Zhao, Yang *et al.*, se concentra na classificação cromossômica, porque este é um estágio muito significativo e de trabalho intensivo, na autocariotipagem. Embora muitas pesquisas tenham feito contribuições para esta área, classificar cromossomos com precisão e robustez ainda é uma tarefa desafiadora devido a diversidade de deformações da forma cromossômica<sup>2</sup> e dificuldade em coletar uma grande quantidade de dados rotulados<sup>3</sup>. Os cromossomos humanos contêm informações genéticas comumente usadas para analisar doenças genéticas. Existem 23 pares de cromossomos em um corpo humano saudável, incluindo 22 pares de autossomos e um par de cromossomos sexuais (cromossomo X e Y nas células masculinas e duplo X nas células femininas). A análise do cariótipo é uma abordagem fundamental para os citogeneticistas clínicos diagnosticarem doenças genéticas. Nesta análise, é gerada a disposição dos cromossomos após extraí-los de imagens cromossômicas metafásicas. Para os citogeneticistas, a cariotipagem é trabalhosa; muitos deles se dedicam à autocariotipagem usando técnicas de computação há anos.

Para resolver os problemas acima, os autores propuseram uma abordagem de classificação automática de cromossomos chamada *CIR-Net* baseada na arquitetura *Inception-ResNet*, que é uma versão otimizada da *Inception*. Além disso, para melhorar a aplicabilidade clínica, os autores projetaram um módulo de interface adaptativa de imagem para converter um tamanho arbitrário da imagem de entrada no tamanho de entrada de destino da *Inception-ResNet*, sem nenhum pré-processamento. A base de dados utilizada foi obtida junto ao Centro de Genética

---

<sup>2</sup> Cromossomos mostrados em microfotografias de células coradas têm natureza intrínseca não rígida, por isso é muito comum que cromossomos do mesmo tipo tenham formas e orientações completamente diferentes.

<sup>3</sup> Como as imagens cromossômicas são altamente correlacionadas à privacidade dos pacientes, é muito difícil para os pesquisadores corrigir dados suficientes de instituições médicas para treinar seus classificadores.

Médica e Laboratório de Chaves Metabólico-Genética Materno-Infantil do Hospital Feminino e Infantil de Guangdong, China. O trabalho utilizou técnicas *data augmentation* e *resizing* das imagens para  $299 \times 299$ . O modelo *CIR-Net* alcançou os seguintes resultados acurácia de 95,98%, sensibilidade de 96% e especificidade de 96,7% (LIN, ZHAO, *et al.*, 2020).

O trabalho de Hasan *et al.* (HASAN, BAO, *et al.*, 2021) apresenta uma técnica de prognóstico de pacientes com COVID-19 a partir da imagem de Tomografia Computadorizada (TC) usando CNN. O surto de COVID-19 foi reconhecido como um evento de impacto global na saúde pública. A nova doença foi identificada, pela primeira vez, no final de 2019, em Wuhan, China, e se espalhou para o mundo todo. A Organização Mundial da Saúde (OMS) anunciou uma crise pandêmica em 11/03/2020. Devido à alta transmissibilidade entre humanos, em contato próximo, os profissionais de saúde e os formuladores de políticas públicas não conseguiram controlar a pandemia até o início de 2022. Neste trabalho, a nova abordagem é baseada na arquitetura CNN modificada DenseNet-121. A base de dados de imagens utilizada foi a de um hospital de São Paulo. A detecção precoce de COVID-19 é primordial para tomar medidas proativas para minimizar os riscos e a disseminação de infecções, planejar o tratamento clínico e providenciar suporte de atendimento oportuno. O trabalho utilizou as técnicas de *data augmentation* e *resizing* das imagens para  $224 \times 224$ . O modelo alcançou os resultados 92% para acurácia e 95% para especificidade.

O modelo COVID-19-CNN descrito em Chen, Yao-Mei *et al.* (CHEN, CHEN, *et al.*, 2021) combina o uso das arquiteturas *CNN VGG-19*, *Resnet-101*, *DenseNet-201*, *Inception-v3* e *Inception-ResNet-v2*, previamente treinadas. O treinamento e o teste de performance deste modelo foram conduzidos em um conjunto de dados coletados, que consistiu em 521 imagens de TC de tórax COVID-19, 397 imagens de TC de tórax normal, 76 imagens de TC de tórax de pneumonia bacteriana e 48 imagens de TC de tórax de SARS, todas obtidas por pesquisa exaustiva de dados disponíveis na Internet. Em seguida, esses dados são divididos em três conjuntos, denominados conjunto de treinamento, conjunto de validação e conjunto de teste. Dezesesseis operações de aumento de dados foram projetadas para enriquecer o conjunto de treinamento na fase de treinamento de *deep learning*. Foram empregadas 612 imagens para treinamento, 99 para validação e 91 para teste. O banco de dados não foi expandido, mas a imagem foi redimensionada para resolução espacial de  $224 \times 224$ . O modelo COVID-19-CNN teve acurácia de 96,70 %, sensibilidade de 97,80 % e especificidade de 95,56 %.

A Tabela 2 exibe o extrato de informações do conjunto de trabalhos do estado da arte, descritos na revisão bibliográfica. Os artigos referenciados são aqueles descritos ao longo desse capítulo.

Tabela 2 – Quadro de referência e características de trabalhos relacionados à criação de modelos baseados em CNN para detecção e classificação de doenças

Trabalho	Ano	Modelo CNN	Objetivo do trabalho	Base de dados	D_A?	I_R?	I_S?	P_P?	Métricas
BRUNO, ARDIZZONE, <i>et al.</i> , 2020	2020	PyramidNet / AlexNet	Detecção de massa em mamografia	SuReMaPP/MIAS	Sim	224×224 227×227	Sim	Sim	SuReMaPP: Sen=98%, Sp=90%/ MIAS: Sen=94%, Sp=91%
LUO, WEN, <i>et al.</i> , 2022	2022	AlexNet	Classificação de imagens da glândula meibomiana	2º Hospital Afiliado da Universidade Médica de Fujian	Sim	224×224	Não	Sim	ACC=94%
SAVELLI, BRIA, <i>et al.</i> , 2020	2020	Multiple-depth CNNs (VGG-Net)	Detecção de microcalcificação/detecção de microaneurisma em imagens de fundo ocular.	INBreast/E-ophta	Sim	192×192	Sim	Sim	Sen= 83,54% (MCs) / Sen=81,62% (MAs)
WANG, SHAO, <i>et al.</i> , 2022	2022	HER2-ResNet	Classificação das imagens HER2	Stanford Tissue Microarray (TMAD)	Sim	64×64	Não	Sim	ACC=93%, Sen=92% e Sp=91%
ASSARI, MAHLOOJIFAR AHMADINEJAD, 2022	2022	GoogleNet	Classificação de massa mamária sólida	Digital Database for Mammography Screening (DDSM)/ Breast Ultrasound Images Dataset (BUSI)	Sim	100×100	-	Sim	ACC=90,38%, Sen=90,91% e Sp=89,87%
DEMIR, YILMAZ KOSE, 2019	2019	Inception V3/ResNet101	Classificação do câncer de pele	ISIC-Archive	Não	224×224	Não	Sim	ResNet101: ACC=89% Inception V3: ACC=90%
KASSANI, KASSANI, <i>et al.</i> , 2019	2019	Xception modificado Xception original	Detecção da Retinopatia diabética	Concurso Kaggle APTOS 2019	Não	299×299	Não	Sim	Xception modificado: ACC=83,09%, Sen=88,24% e Sp=87% Xception original: ACC=79,59%,

									Sen=82,35% Sp=86,32%	e
LIN, ZHAO, <i>et al.</i> , 2020	2020	Inception-Res-Net	Classificação automática de cromossomo humano.	Centro Médico Genético e Metabólico Materno-Infantil Laboratório de Chave Genética de Mulheres de Guangdong	Sim	299×299	Não	Sim	Inception-ResNet : ACC=95,98%, Sen=96% e Sp=96%	
HASAN, BAO, <i>et al.</i> , 2021	2021	DenseNet121	Classificação de pacientes com COVID-19	Hospital de São Paulo	Sim	224×224	Não	Sim	ACC=92% e Sp=95%	
EMARA, AFIFY, <i>et al.</i> , 2019	2019	Inception V4	Classificar extraídos do conjunto de dados HAM10000.	Conjunto de dados Human Against Machine with 10000 Training Images (RAM 10000)	Sim	299×299	Não	Sim	ACC=94,47%	
CHEN, CHEN, <i>et al.</i> , 2021	2021	VGG-19, Resnet-101, DenseNet-201 Inception-v3 e Inception-ResNet-v2	classificar imagens de TC de tórax adquiridas de pacientes com COVID-19	Conjuntos de dados obtidos de pacientes com COVID-19 em Hu	Sim	224x224	Não	Sim	ACC= 96,70 %, Sen= 97,80 % e Sp= 95,56 %.	
Modelo Proposto	2022	WCNN	Classificador de pacientes com COVID-19	BIMCV, HSL-PUCRS, HC-UFRJ	Não	Não	Não	Sim	ACC= 98,19 %, Sen= 97,83 e Sp= 98,00 %	

Onde as colunas:

- **D\_A?**: Indica se o Modelo CNN utiliza alguma técnica de *Data Augmentation*.
- **I\_R?**: Indica se o Modelo CNN utiliza alguma técnica de *Image Resizing*.
- **I\_S?**: Indica se o Modelo CNN utiliza alguma técnica de *Image Segmentation*.
- **P\_P?**: Indica se o Modelo CNN utiliza alguma técnica de *Pre Processing*.

## 3. Referencial Teórico

---

### 3.1. Redes Neurais

*Deep neural networks* têm capacidade de aprender e melhorar com a experiência. O processo de aprendizado de máquina começa com os dados brutos que são usados para extrair informações úteis que auxiliam na tomada de decisões. O objetivo principal é permitir que uma máquina aprenda informações úteis, assim como os humanos fazem. Em nível abstrato, o aprendizado de máquina pode ser realizado de acordo com as abordagens: *i) A aprendizagem supervisionada* adapta um sistema de tal forma que, para um dado de entrada, ele produz uma saída de destino. Os dados de aprendizagem são compostos de *tuplas* (atributos, rótulo), onde os *atributos* representam os dados de entrada e *rótulo* representa a saída de destino, também chamado de valor esperado ou *ground truth*. O objetivo aqui é adaptar o sistema de forma que, para uma nova entrada, o sistema possa prever a saída desejada. A aprendizagem supervisionada pode usar tipos contínuos e discretos de entradas de dados (WANI, BHAT, *et al.*, 2020); e, *ii) A aprendizagem não supervisionada* envolve dados que compreendem vetores de entrada sem qualquer saída de destino. Existem diferentes objetivos na aprendizagem não supervisionada, como por exemplo o agrupamento. O objetivo do agrupamento é descobrir grupos de itens de dados semelhantes com base nas semelhanças medidas ou percebidas entre os itens de dados (WANI, BHAT, *et al.*, 2020). Este trabalho utiliza aprendizagem supervisionada, e nas próximas subseções serão abordados conceitos de *neural networks* e *deep neural networks*, relevantes para esta pesquisa.

#### 3.1.1. Ideia do Gradiente Descendente

A ideia intuitiva do gradiente descendente (um algoritmo que permite encontrar o mínimo global de uma função) é representada por um alpinista descendo uma montanha como ilustrado na Figura 1, onde este se move para frente de uma em uma hora, sempre descendo até chegar ao fundo do vale (mínimo global).

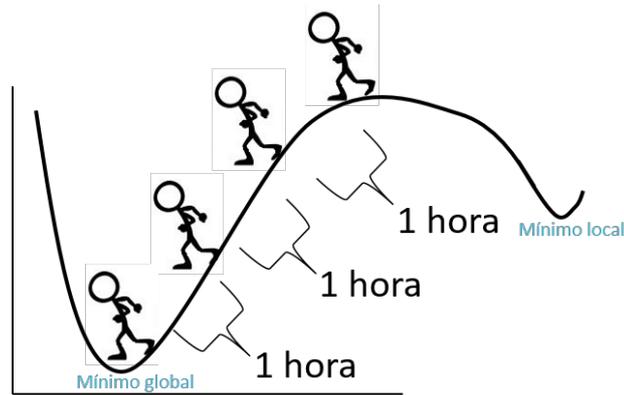


Figura 1 – Ideia do gradiente descendente.

Matematicamente, a inclinação corresponde à derivada do ponto. Então, se a derivada for alta, a inclinação é alta, ao passo que, se a derivada for baixa, a inclinação também é baixa. Caso a derivada seja zero, não existe inclinação, logo tudo é plano (o alpinista pode estar em um mínimo local ou global), conforme exibido na Figura 2.

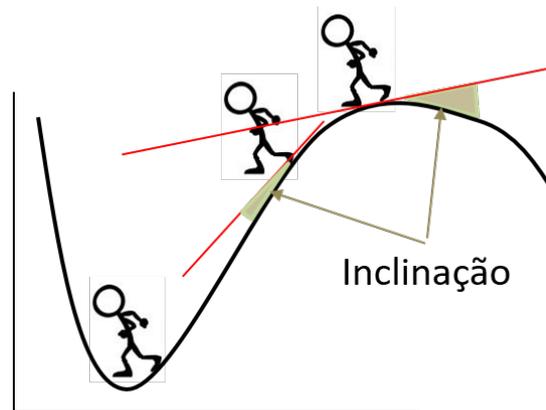


Figura 2 – Ideia da derivada no gradiente descendente.

A direção do gradiente descendente é sempre oposta ao sinal da derivada. Na Figura 3, para ambas as derivadas: azul (lado esquerdo) e vermelho (lado direito), para descer a montanha, multiplica-se a derivada por um sinal negativo para que a direção mude para o fundo do vale (mínimo global).

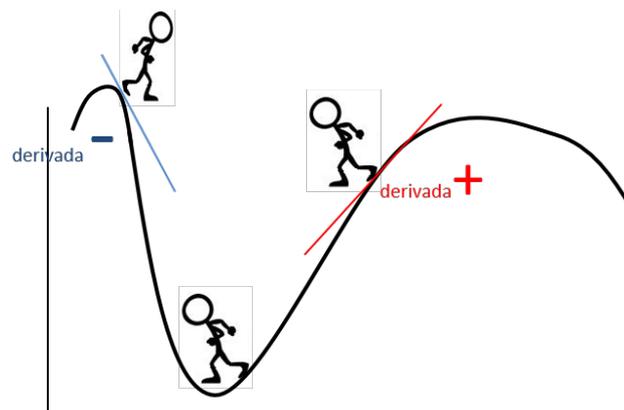


Figura 3 – Direção da derivada.

Tendo o valor da derivada e a direção, o alpinista caminha em direção ao fundo do vale, lembrando que, se ele der passos largos, corre o risco de perder o fundo (mínimo global), passando para o outro lado, e, assim, sucessivamente, nunca alcançando o fundo; e, se ele der passos pequenos, pode levar muito tempo até alcançar o fundo do vale. O passo dado pelo alpinista é chamado *taxa de aprendizagem*  $\alpha$ , que controla a velocidade da descida do gradiente em direção ao mínimo global.

### 3.1.2. *Gradient Descent (GD)*

No *Gradient Descent* (GD) ou descida gradiente em lote, o gradiente de erro em relação ao parâmetro de peso  $w$  é calculado para todo o conjunto de treinamento, seguido pela atualização do parâmetro de peso mostrado na Equação 1 (WANI, BHAT, *et al.*, 2020):

$$w = w - \alpha \cdot \nabla E(w) \quad \text{Equação 1}$$

Onde:

- $\nabla E(w)$  é o gradiente de erro em relação ao peso  $w$ ;
- $\alpha$  é a taxa de aprendizado que define o tamanho do passo a ser percorrido ao longo do gradiente.

A taxa de aprendizagem é um hiperparâmetro que não pode ser muito alto nem muito baixo. Um valor grande da taxa de aprendizado pode perder o valor ideal, e uma taxa de aprendizado muito baixa resultará em um tempo de treinamento lento. É usual que o conjunto de treinamento contenha um número muito grande de instâncias, que pode demandar muita memória, o que torna difícil seu ajuste. Como consequência, o cálculo do gradiente de erro pode se tornar muito lento (WANI, BHAT, *et al.*, 2020).

### 3.1.3. *Stochastic Gradient Descent (SGD)*

O problema do *Gradient Descent* (GD), descrito na seção anterior, acerca do gasto de memória, pode ser corrigido usando *Stochastic Gradient Descent* (SGD), também conhecido como gradiente descendente incremental, onde o gradiente é calculado para um exemplo de treinamento por vez, seguido pela atualização dos valores dos parâmetros. Geralmente é muito mais rápido do que a descida gradiente padrão, pois realiza uma atualização de cada vez, apresentado na Equação 2 (WANI, BHAT, *et al.*, 2020):

$$w = w - \alpha \cdot \nabla E(w, x(i), d(i)) \quad \text{Equação 2}$$

Onde  $\nabla E(w; x(i); d(i))$  é o gradiente de erro, em relação ao peso  $w$  para uma amostra de treinamento  $\{x(i), d(i)\}$ ; em que:

- $x$  são as entradas da rede neural;
- $d$  os valores esperados, também chamados *ground truth*;
- $i$  é a quantidade da amostra.

No SGD, uma única amostra para atualização dos valores dos parâmetros faz com que a função de erro flutue com frequência alta.

### 3.1.4. Batch Gradient Descent

*Batch Gradient Descent*, também conhecida como *Batch SGD*, é uma combinação das técnicas GD e SGD. O *batch SGD* divide todo o conjunto de treinamento em *batches* de  $n$  exemplos de treinamento e executa a atualização dos valores dos parâmetros para cada *batch*. Este tipo de técnica de gradiente descendente aproveita as vantagens das técnicas de descida gradiente padrão GD e SGD, e é uma técnica de otimização comumente usada no aprendizado profundo, conforme Equação 3 (WANI, BHAT, *et al.*, 2020):

$$w = w - \alpha \cdot \nabla E(w, x(i:n); d(i:n)) \quad \text{Equação 3}$$

Sobre o tamanho do *batch*, não há meios exatos de definir tal tamanho. É possível, apenas, tomar alguns cuidados para direcionar sua escolha:

- Os grandes *batches* fornecem gradientes mais precisos, mas, em compensação, têm altos requisitos de memória.
- Por outro lado, os pequenos *batches* oferecerem efeito de regularização, mas requerem uma pequena taxa de aprendizado para manter a estabilidade devido à alta variação na estimativa do gradiente. Isso, por sua vez, aumenta o tempo de treinamento devido à taxa de aprendizado reduzida.

### 3.1.5. Otimização do GD

O principal objetivo da otimização é minimizar o erro. Pois o gradiente descendente é um processo lento de convergência, pois depende de parâmetros escolhidos de forma aleatória. No caso de redes neurais seria a escolha inicial dos pesos. Existem muitos métodos disponíveis que ajudam um algoritmo de otimização a convergir mais rapidamente. Alguns dos métodos comumente usados são discutidos abaixo.

### 3.1.5.1. AdaGrad

No SGD, a taxa de aprendizagem é definida independentemente dos gradientes, o que às vezes pode causar problemas. Por exemplo, se o gradiente for grande, uma grande taxa de aprendizado resultaria em um grande tamanho de etapa, o que significa que pode não atingir o valor ideal, pois pode continuar oscilando em torno do valor ideal e, se a magnitude do gradiente for pequena, um pequeno aprendizado taxa pode resultar em convergência lenta. O problema pode ser resolvido usando alguma abordagem adaptativa para definir a taxa de aprendizagem. AdaGrad é um modelo adaptativo que usa a taxa de aprendizagem adaptativa adicionando o quadrado dos gradientes anteriores ( $\nabla_t^2$ ) e dividindo a taxa de aprendizagem pela raiz quadrada dessa soma, como mostra a Equação 4 (WANI, BHAT, *et al.*, 2020):

$$w_{t+1,i} = w_{t,i} - \frac{\alpha}{\sqrt{G_i}} \cdot \nabla_{t,i} \quad \text{Equação 4}$$

Onde  $\nabla_{t,i}$  é o *gradient descent* da função erro em relação  $w_i$ ,  $G_i = \sum_1^t \nabla_t^2$  e  $\nabla_t$  é o gradiente da iteração  $t$ .

Dessa forma, os parâmetros com gradientes altos terão uma taxa de aprendizado efetiva pequena e os parâmetros com gradientes pequenos receberão uma taxa de aprendizado efetiva aumentada. A principal vantagem do AdaGrad é que a taxa de aprendizagem é ajustada automaticamente e não há necessidade de ajustá-lo manualmente. No entanto, a soma no denominador continua aumentando, o que gradualmente faz com que a taxa de aprendizagem diminua. Esta taxa de aprendizagem descendente pode desacelerar o aprendizado ou interromper completamente o aprendizado.

### 3.1.5.2. AdaDelta

AdaDelta é uma versão modificada do AdaGrad que supera o problema da taxa de inclinação decrescente. AdaDelta limita o número de gradientes anteriores a algum tamanho fixo  $x$  e, em seguida, a média desses gradientes anteriores é armazenada para fins de análise de desempenho. O valor médio  $AVG(\nabla_t^2)$  no tempo  $t$  depende apenas da média anterior e do gradiente atual. A atualização do parâmetro é então feita como mostrado nas Equações 5 e 6 (WANI, BHAT, *et al.*, 2020):

$$w_{t+1,i} = w_{t,i} - \frac{\alpha}{\sqrt{AVG(\nabla_t^2)}} \cdot \nabla_t \quad \text{Equação 5}$$

Uma vez que o denominador é apenas a raiz da média do quadrado (RMS) dos parâmetros (WANI, BHAT, *et al.*, 2020):

$$w_{t+1,i} = w_{t,i} - \frac{\alpha}{RMS(\nabla_t)} \cdot \nabla_t \quad \text{Equação 6}$$

Lembrete Matemático: o valor da raiz da média dos quadrados (RMS) de um conjunto de valores é a raiz quadrada da média aritmética dos quadrados dos valores. Exemplo para o caso de um conjunto de  $n$  valores  $\{x_1 + x_2 + x_3 + \dots + x_n\}$ . O RMS é mostrado na Equação 7:

$$x_{RMS} = \sqrt{\frac{1}{n}(x_1^2 + x_2^2 + x_3^2 + \dots + x_n^2)} \quad \text{Equação 7}$$

### 3.1.5.3.RMSPro

O problema da taxa de aprendizagem do AdaGrad, o qual pode desacelerar/interromper o aprendizado, pode ser corrigida usando RMSProp. É uma versão modificada do AdaGrad que descarta a história do passado distante, introduzindo uma média móvel exponencialmente ponderada. RMSProp usa o sinal do gradiente em vez da magnitude do gradiente para atualizar os pesos. O funcionamento do otimizador RMSProp é o seguinte (KINGMA e BA, 2014; WANI, BHAT, *et al.*, 2020):

- Defina a mesma magnitude de atualizações para todos os pesos (também chamada de *momentum*). Defina as atualizações de peso máximo e mínimo permitido para máximo e mínimo, respectivamente.
- A cada iteração, se os sinais do gradiente corrente e do gradiente anterior forem iguais, aumente a taxa de aprendizado por um fator de 1,2, ou seja,  $\alpha = \alpha + 1,2$ . Portanto, a atualização  $\Delta_{ij}^{t+1}$  torna-se

$$\Delta_{ij}^{t+1} = \min(\alpha + \Delta_{ij}^t, \Delta_{max})$$

- Se os sinais do gradiente atual e do gradiente anterior forem diferentes, diminua a taxa de aprendizado por um fator 0,5, ou seja,  $\alpha = \alpha - 0,5$ . Portanto, a atualização  $\Delta_{ij}^{t+1}$  torna-se

$$\Delta_{ij}^{t+1} = \max(\alpha - \Delta_{ij}^t, \Delta_{min})$$

### 3.1.5.4.ADAM

*ADAM* é um método de aprendizagem adaptativa, o que significa que calcula taxas de aprendizagem individuais para diferentes parâmetros. Seu nome é derivado de estimativa de *momentum* adaptativo, e a razão pela qual é chamado assim é porque Adam usa estimativas de primeiro e segundo *momentum* (o primeiro momento é a média e o segundo momento é a vari-

ância descentralizada (o que significa que a média não foi subtraída durante o cálculo da variância). Assim, ADAM é uma técnica de otimização adaptativa que aproveita as vantagens do AdaGrad e do RMSProp. Como AdaDelta e RMSProp, ADAM salva uma média exponencialmente decrescente de gradientes quadrados anteriores  $v_t$ . Além disso, Adam também calcula a média dos segundos momentos dos gradientes  $m_t$  (KINGMA e BA, 2014; WANI, BHAT, *et al.*, 2020). Os valores da média e variância não centrada  $m_t$  e  $v_t$ , respectivamente, são dados nas Equações 8 e 9:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla_t \quad \text{Equação 8}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) \nabla_t^2 \quad \text{Equação 9}$$

ADAM atualiza as medias móveis exponenciais do gradiente  $\nabla_t$  e o gradiente ao quadrado  $\nabla_t^2$ , onde os hiperparâmetros  $\beta_1, \beta_2 \in [0, 1]$ . Na literatura foi encontrado que usualmente valores com bons resultados de  $\beta_1$  e  $\beta_2$  são 0,9 e 0,99, respectivamente, controlam as taxas de decaimento das médias móveis Equação 10 e Equação 11:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad \text{Equação 10}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad \text{Equação 11}$$

A equação final para atualização é dada pela Equação 12:

$$w_{t+1} = w_t - \frac{\alpha \cdot \hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad \text{Equação 12}$$

Onde  $\alpha$  é a taxa de aprendizagem e  $\epsilon$  uma constante adicionada ao denominador dos métodos de convergência rápida para evitar a divisão por zero (KINGMA e BA, 2014; WANI, BHAT, *et al.*, 2020).

Assim, ADAM tem um desempenho ligeiramente melhor do que outras técnicas adaptativas e converge muito rápido. Ele também supera os problemas enfrentados por outras técnicas de otimização, como taxa de aprendizagem decrescente, alta variação nas atualizações e convergência lenta.

### 3.1.6. Backpropagation

A *backpropagation* é central para muitos trabalhos atuais sobre aprendizagem em redes neurais. O algoritmo fornece uma prescrição para alterar os pesos  $W_{p,q}$  em qualquer rede *feed-forward* para aprender um conjunto de treinamento de pares de entrada-saída  $\{X, O\}$ . Primeiramente, foi considerada uma rede de três camadas (Figura 4).

A camada de saída é denotada por  $O_i$ , a camada oculta por  $V_j$  a camada de entrada por  $X_k$ . Há conexões  $W_{jk}$  entre as camadas de entrada e a camada oculta e conexões  $W_{ij}$  entre as camadas oculta e a camada de saída. O índice  $i$  (Figura 4-azul) se refere a uma unidade da camada de saída,  $j$  (Figura 4-vermelho) a uma unidade da camada oculta,  $k$  (Figura 4-verde) um terminal da camada de entrada e  $\mu$  a camada (HERTZ, KROGH e PALMER, 2018; DA SILVA, SPATTI e FLAUZINO, 2016).

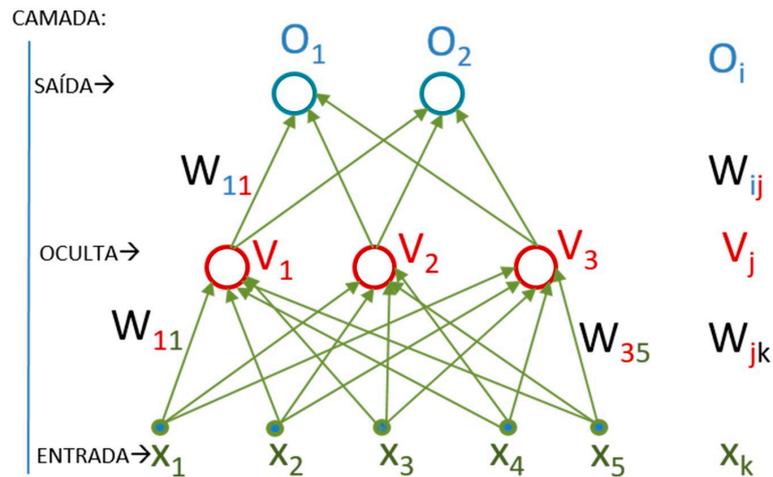


Figura 4 – Diagrama esquemático de uma rede neural, mostrando a notação para as camadas: entrada, oculta e saída, juntamente com os pesos relacionados. Adaptado de (HERTZ, KROGH e PALMER, 2018)

A camada oculta  $V_j$  recebe a camada de entrada  $X_k$ , conforme Equação 13

$$h_j^\mu = \sum_k W_{jk} X_k^\mu \quad \text{Equação 13}$$

e produz a saída utilizando a função de ativação  $g$ , conforme Equação 14

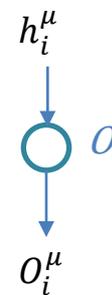
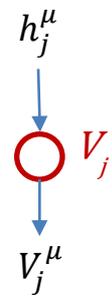
$$V_j^\mu = g(h_j^\mu) = g\left(\sum_k W_{jk} X_k^\mu\right) \quad \text{Equação 14}$$

A camada de saída  $O_i$  recebe conforme Equação 15

$$h_i^\mu = \sum_j W_{ij} V_j^\mu = \sum_j W_{ij} g\left(\sum_k W_{jk} X_k^\mu\right) \quad \text{Equação 15}$$

e produz uma saída, Equação 16

$$O_i^\mu = g(h_i^\mu) = g\left(\sum_j W_{ij} V_j^\mu\right) = g\left(\sum_j W_{ij} g\left(\sum_k W_{jk} X_k^\mu\right)\right) \quad \text{Equação 16}$$



Para facilitar o entendimento, os *thresholds* foram omitidos<sup>4</sup>, mas eles podem ser considerados, como de costume, por uma unidade de entrada extra fixada em  $-1$  e conectada a todas as unidades na rede (HERTZ, KROGH e PALMER, 2018).

O próximo passo para o início da derivação do algoritmo *backpropagation* consiste em definir a função de custo, também chamada de erro ( $E$ ), a qual tem a função de medir os desvios entre as respostas produzidas pelos neurônios de saída da rede em relação aos respectivos valores desejados ( $d$ ). O algoritmo *backpropagation* utiliza o erro quadrático médio, conforme Equação 17 (HERTZ, KROGH e PALMER, 2018):

$$E[W] = \frac{1}{2} \sum_{\mu i} [d_i^\mu - o_i^\mu]^2 \quad \text{Equação 17}$$

Substituindo a Equação 16 na Equação 17, chega-se à Equação 18:

$$E[W] = \frac{1}{2} \sum_{\mu i} \left[ d_i^\mu - g \left( \sum_j W_{ij} g \left( \sum_k W_{jk} X_k^\mu \right) \right) \right]^2 \quad \text{Equação 18}$$

Esta é uma função contínua diferenciável de cada peso, que permite usar um algoritmo do gradiente descendente para aprender os pesos apropriados. Logo, o *backpropagation* ocorre onde o gradiente da função do erro na camada de saída da rede é calculado, e este gradiente será utilizado para aplicar recursivamente a regra da cadeia, ou *chain rule*, para atualizar os pesos da rede, numa etapa também conhecida como *fase de atualização de pesos* (HERTZ, KROGH e PALMER, 2018).

Para as camadas **Ocultas**  $\rightarrow$  **Saída**, a regra de gradiente fornece Equação 19.

$$\Delta w_{ij} = -\alpha \frac{\partial E}{\partial w_{ij}}$$

$$h_i^\mu = \sum_j W_{ij} g \left( \sum_k W_{jk} X_k^\mu \right)$$

$$\Delta w_{ij} = \alpha \sum_{\mu} [d_i^\mu - o_i^\mu] \cdot g'(h_i^\mu) \cdot (V_j^\mu)$$

$$V_j^\mu = g \left( \sum_k W_{jk} X_k^\mu \right)$$

$$\Delta w_{ij} = \alpha \delta_i^\mu (V_j^\mu) \quad \text{Equação 19}$$

$$\delta_i^\mu = g'(h_i^\mu) \cdot [d_i^\mu - o_i^\mu]$$

Onde  $g'$  é a derivada da função  $g$  e  $\delta$  é chamado delta.

<sup>4</sup> *Bias*, em sua tradução literal, significa “viés”, é uma forma de direcionar o aprendizado da rede neural e impedir que ela transmita valores zerados por acidente, fazendo com que ela caia em mínimos locais e não consigam aprender corretamente. *Bias*, que é um parâmetro constante da rede neural que é inserido junto às entradas, ele é tratado como uma entrada e possui pesos, que podem ser atualizados pelo algoritmo de atualização de pesos que estiver sendo utilizado.

Para as camadas **Entrada** → **Ocultas**, a regra de gradiente fornece a Equação 20.

$$\Delta w_{jk} = -\alpha \frac{\partial E}{\partial w_{jk}}$$

$$\Delta w_{jk} = -\alpha \frac{\partial E}{\partial V_j^\mu} \cdot \frac{\partial V_j^\mu}{\partial w_{jk}}$$

$$h_j^\mu = \sum_k W_{jk} X_k^\mu$$

$$\Delta w_{jk} = \alpha \sum_\mu [d_i^\mu - o_i^\mu] \cdot g'(h_i^\mu) \cdot W_{ij} \cdot g'(h_j^\mu) \cdot X_k^\mu$$

$$\Delta w_{ik} = \alpha \sum_{\mu i} \delta_i^\mu \cdot W_{ij} \cdot g'(h_j^\mu) \cdot X_k^\mu$$

$$\Delta w_{ik} = \alpha \underbrace{\sum_{\mu i} \delta_j^\mu \cdot X_k^\mu}_{\text{Equação 20}}$$

$$\delta_j^\mu = g'(h_j^\mu) \sum_i \delta_i^\mu W_{ij}$$

Observe que a Equação 20 tem a mesma forma que a Equação 19, mas com definições diferentes de  $\delta$ . Em geral, com um número arbitrário de camadas, a regra de atualização de retropropagação sempre tem a forma da Equação 21:

$$\Delta W_{pq} = \alpha \cdot \sum \delta_{saída} \cdot V_{entrada} \quad \text{Equação 21}$$

Onde a **saída** e a **entrada** referem-se às duas extremidades  $p$  e  $q$  da conexão em questão e  $V$  representa a ativação da camada de entrada ou oculta. O significado de  $\delta$  depende da camada em questão; para a última camada de conexões, ele é dado pela Equação 19, enquanto, para todas as outras camadas, o significado é dado pela Equação 20. É fácil generalizar este processo através da Equação 21, simplesmente aplicando a regra da cadeia.

O resultado em termos de um procedimento passo a passo/ algoritmo. Foi considerada uma rede com  $M$  camadas  $m = 1, 2, \dots, M$  e  $V_i^m$  para a saída do  $i$ -ésimo neurônio/unidade da  $m$ -ésima camada,  $V_i^0$  corresponde a camada de entrada  $X_i$  e  $w_{ij}^m$  significam a conexão entre  $V_j^{m-1}$  e  $V_i^m$ . Segue o algoritmo *backpropagation*.

1. Inicializar os pesos com pequenos valores aleatórios.
2. Definir as entradas da rede neural para montar a camada de entrada ( $m = 0$ ). Na Equação 22.

$$V_k^0 = X_k, \forall k \quad \text{Equação 22}$$

3. Propagar o sinal para a frente, *forward*, através da rede. Na Equação 23.

$$V_i^m = g(h_i^m) = g\left(\sum_j W_{ij}^m \cdot V_j^{m-1}\right) \quad \text{Equação 23.}$$

Para cada  $i$  e  $m$ . Até a saída final,  $V_i^M$  foram todos calculados.

4. Calcular  $\delta$  para camada de saída. Na Equação 24.

$$\delta_i^M = g'(h_i^M) \cdot [d_i^M - v_i^M] \quad \text{Equação 24.}$$

5. Calcular  $\delta$  para camadas anteriores, propagando o erro *backward*. Na Equação 25.

$$\delta_i^{m-1} = g'(h_i^{m-1}) \sum_j w_{ij}^m \delta_j^m \quad \text{Equação 25.}$$

Para  $m = M, M - 1, \dots, 2$  até  $\delta$  ser calculado para as unidades/neurônios.

6. Use (Equação 26):

$$\Delta W_{ij}^m = \alpha \cdot \delta_i^m v_j^{m-1} \quad \text{Equação 26}$$

Para atualizar todos os pesos de acordo com a Equação 27:

$$W_{ij}^{new} = W_{ij}^{old} + \Delta W_{ij}^m \quad \text{Equação 27}$$

7. Volte para o passo 2 e repita para as próximas entradas.

Resumidamente o algoritmo do *backpropagation*, conhecido também como regra Delta generalizada, opera em uma sequência de dois passos:

1. Primeiro, na fase *forward*, um padrão é apresentado à camada de entrada da rede, a atividade resultante flui através da rede, camada por camada até que a resposta seja produzida pela camada de saída;
2. Segundo, na fase *backward*, a saída obtida é comparada à saída desejada para esse padrão, se esta não estiver correta, o erro é calculado.

O erro é propagado a partir da camada de saída até a camada de entrada, e os pesos das conexões das unidades das camadas internas vão sendo modificados conforme o erro é retro-propagado (FAUSETT, 2006), mostrado na Figura 5.

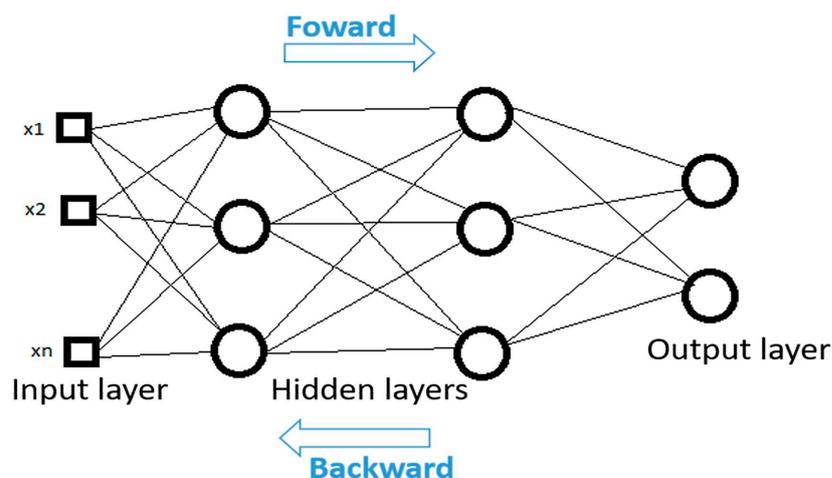


Figura 5 – Diagrama esquemático do algoritmo de *backpropagation*

### 3.1.7. Convolutional Neural Network (CNN)

CNN é uma arquitetura de rede neural que utiliza técnicas de *deep learning*, consistindo no conjunto de várias camadas que extraem padrões de uma entrada. CNNs são inspiradas no

córtex visual biológico, o qual contém pequenas regiões de células que são sensíveis as regiões específicas do campo visual. Assim diferentes neurônios cerebrais respondem a diferentes características, como por exemplo a ativação de alguns neurônios quando são sensibilizados por linhas verticais, enquanto outros são sensibilizados por linhas horizontais ou verticais. Logo, a estratégia de atribuir tarefas específicas para um conjunto especializado de neurônios é a base das CNNs (WANI, BHAT, *et al.*, 2020).

Uma CNN dedicada à classificação é composta por dois estágios: *feature extration* e classificação. Na CNN, as sequências de camadas de convolução e *pooling* atuam como um estágio de *feature extration*, enquanto o estágio de classificação é composto por uma ou mais *fully connected layer* seguidos por uma função de ativação, por exemplo uma função sigmóide (WANI, BHAT, *et al.*, 2020), conforme a Figura 6.

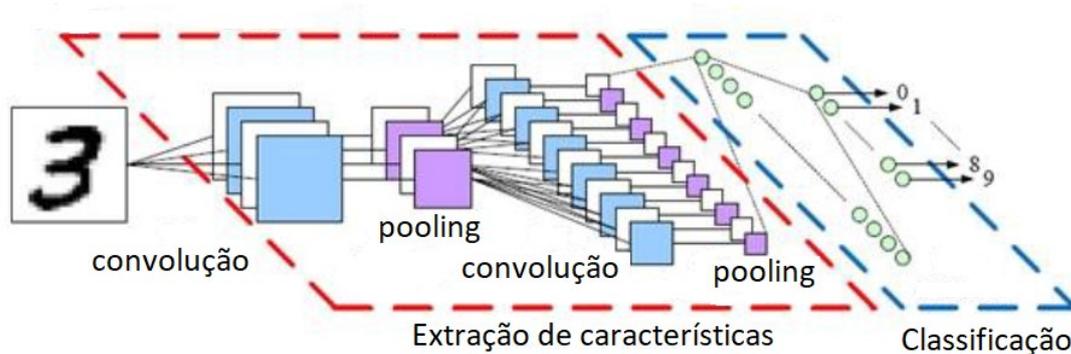


Figura 6 – Exemplo de uma rede neural profunda CNN. Fonte: Adaptado de DUC-JACQUET, 2017

### 3.1.8. Camada Convolutacional

Para a camada convolutacional é definida uma operação de convolução, na qual um *kernel* é usado para mapear as ativações de uma camada para a camada seguinte. A operação de convolução coloca o *kernel* em cada posição possível na imagem, ou camada oculta, para que o *kernel* se sobreponha totalmente à imagem e execute um produto escalar entre os parâmetros do *kernel* e do *local receptive field* que é a área de entrada, correspondente à imagem na qual um *kernel* é aplicado. A operação de convolução é executada em todas as posições possíveis na imagem para definir a próxima camada, na qual as ativações mantêm suas relações espaciais da camada anterior, como pode ser observado na Figura 7 (PONTI e DA COSTA, 2018; AGGARWAL e OTHERS, 2018; LECUN, BENGIO e HINTON, 2015; BALAS, ROY, *et al.*, 2019).

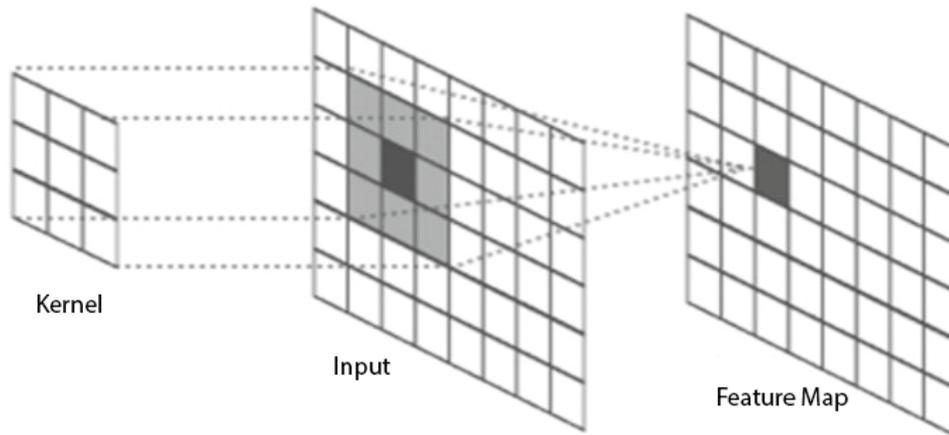


Figura 7 – Diagrama esquemático da operação de convolução. Fonte: WANI, BHAT, *et al.*, 2020.

Na camada convolucional pode haver vários *kernels*. Todo *kernel* evidencia alguma característica, como aresta, canto etc. Durante o processamento (*forward*) cada *kernel* é deslizado pela largura e altura da imagem (ou camada oculta), gerando o *feature map*.

A operação de convolução pode ser aplicada a imagens para realizar várias transformações, quando as imagens são tratadas com funções bidimensionais. Por exemplo, seja um *feature map* (2D) denotado por  $A$ , e um *kernel* 2D de tamanho  $m \times n$  denotado por  $K$ . Esta operação de convolução  $F$  é denotada por  $A \times K$  e é representada matematicamente pela Equação 28 abaixo:

$$F(i, j) = A(i, j) * K(i, j) = \sum_m \sum_n A(m, n)K(i - m, j - n) \quad \text{Equação 28}$$

Um exemplo numérico de convolução entre uma entrada  $5 \times 5$  e um *kernel*  $2 \times 2$  é apresentado na Figura 8.

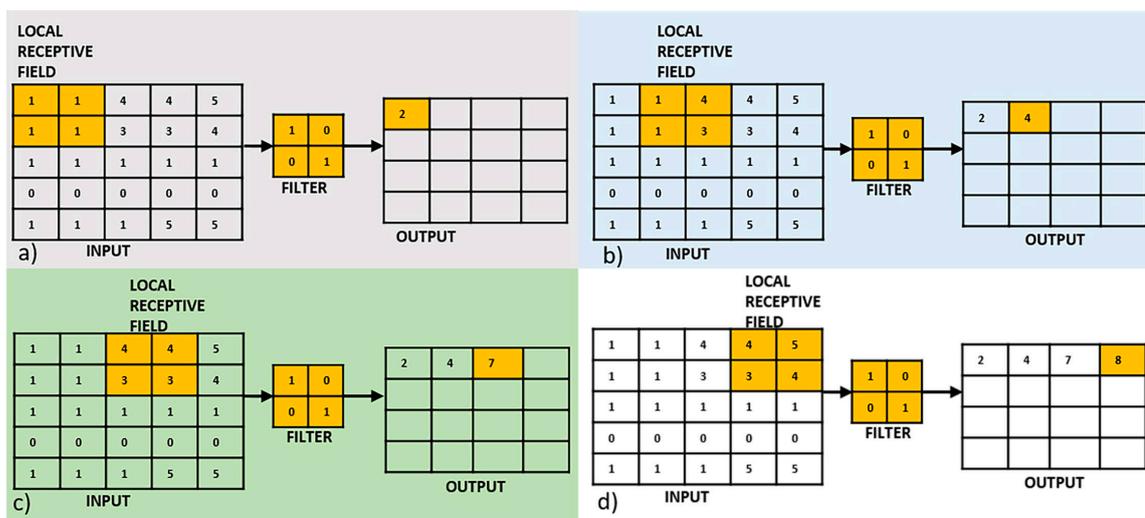


Figura 8 – Exemplo de convolução entre uma entrada  $5 \times 5$  e um *kernel*  $2 \times 2$ . As figuras (a), (b), (c) e (d) correspondem ao resultado do processamento da convolução da primeira linha (PONTI e DA COSTA, 2018).

### 3.1.9. Funções de ativação

A saída das camadas convolucionais é alimentada por uma função de ativação. A camada da função de ativação consiste na função de ativação que utiliza o *feature map* produzido pela camada convolucional e gera o mapa de ativação como saída.

A função de ativação é usada para transformar o nível de ativação de um neurônio em um sinal de saída. Assim, realiza uma operação matemática gerando um nível de ativação do neurônio em um determinado intervalo, por exemplo, de 0 a 1 ou de  $-1$  a 1 (WANI, BHAT, *et al.*, 2020, PONTI e DA COSTA, 2018).

As principais funções de ativação utilizadas em CNN são:

- Função de ativação **logística/sigmóide**: A função sigmóide  $\sigma(x) = \frac{1}{1+e^{-x}}$  é uma curva em forma de S, como mostrado na Figura 9, e restringe a entrada no intervalo  $[0, 1]$  (WANI, BHAT, *et al.*, 2020, PONTI e DA COSTA, 2018).

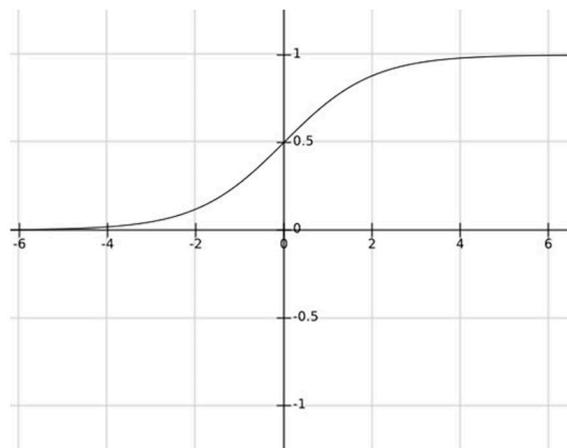


Figura 9 – Função de ativação *sigmóide*. Fonte: ( PONTI e DA COSTA, 2018).

- Função de ativação **tanh**: A função tangente hiperbólica  $\tanh(x) = \frac{2}{(1+e^{-2x})-1}$  é semelhante à função sigmóide, porém, sua saída encontra-se no intervalo  $[-1, 1]$ . A vantagem de **tanh** sobre sigmóide é que as entradas negativas serão mapeadas fortemente negativas, e as entradas zero serão mapeadas perto de zero no gráfico *tanh*, como mostrado na Figura 10 (WANI, BHAT, *et al.*, 2020, PONTI e DA COSTA, 2018).

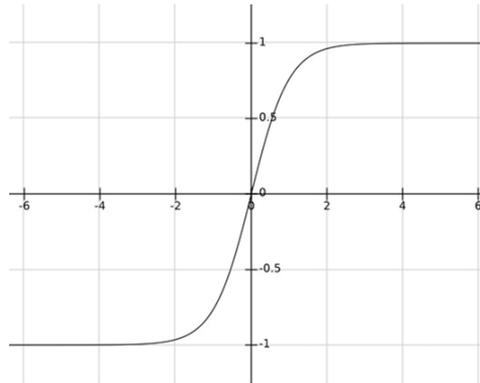


Figura 10 – Função de ativação *tanh*. Fonte: ( PONTI e DA COSTA, 2018).

- Função de ativação ***softmax*** (Função Exponencial): É frequentemente usada na camada de saída de uma rede neural para classificação. Ela é dada por  $\sigma(x_j) = \frac{e^{x_j}}{\sum_{k=1}^n e^{x_k}}$  (WANI, BHAT, *et al.*, 2020, PONTI e DA COSTA, 2018). A função *softmax* é uma função de ativação logística mais generalizada, geralmente usada para classificação multiclasse.
- Função de ativação ***Rectified Linear Unit*** – **ReLU** gera um mapa de ativação não linear, sua fórmula é  $f(x) = \max(0, x)$ , conforme mostra a Figura 11 (WANI, BHAT, *et al.*, 2020, PONTI e DA COSTA, 2018).

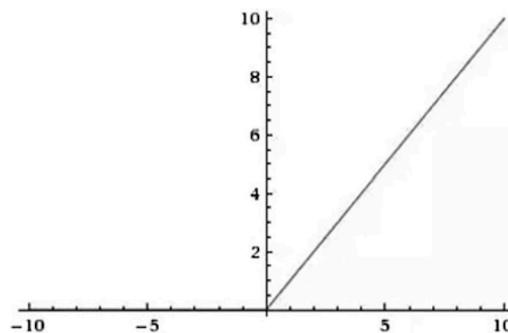


Figura 11 – *Rectified Linear Unit* (ReLU): Tem saída 0 se a entrada for  $< 0$ . Se a entrada for  $> 0$ , a saída é igual à entrada. Fonte: (PONTI e DA COSTA, 2018).

### 3.1.10. Camada de *pooling*

A camada de *pooling* ou *down-sampling* é utilizada para reduzir o tamanho espacial da entrada e, assim, reduzir o número de parâmetros da rede. A camada *pooling* faz uma amostragem reduzida a partir de cada *feature map* da camada convolucional, ou seja, o objetivo é pegar alguma informação que indique o comportamento geral de cada campo receptivo. Tem-se os seguintes tipos de *pooling* (PONTI e DA COSTA, 2018; AGGARWAL e OTHERS, 2018; LECUN, BENGIO e HINTON, 2015):

- *Max pooling* faz uma amostragem com o máximo de cada *feature map*.

- *Min pooling* faz uma amostragem com o mínimo de cada *feature map*.
- *Average pooling* faz uma amostragem com a média de cada *feature map*.

Um exemplo da técnica *max pooling* é apresentado na Figura 12, o qual pode ser estendida para as outras técnicas. A *feature map* tem dimensão  $5 \times 5$  e o *max pooling* tem dimensão  $2 \times 2$ , logo o *max pooling* emitirá o máximo dos quatro valores de entrada, como mostra a Figura 12.

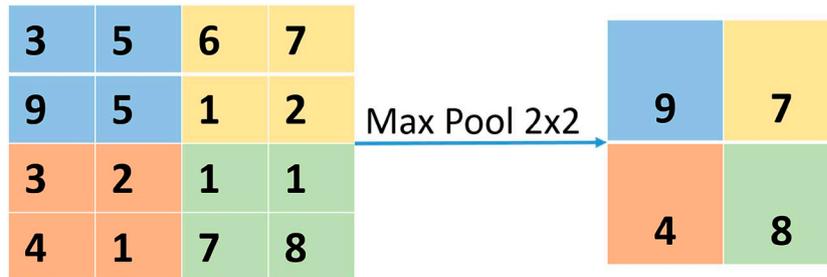


Figura 12 – *Max pooling*.

### 3.1.11. *Fully connected layer*

Em CNN, o próximo passo após as etapas de convolução e *pooling* é o processo de tomada de decisão com base nas características extraídas. Isto é feito adicionando uma *fully connected layer* após o processo de *feature extraction*. Nesta camada, cada neurônio da camada anterior (camada de convolução ou camada de *pooling* ou *fully connected layer*) é conectado a cada neurônio na próxima camada e todo valor contribui para prever a força com que um valor corresponde a uma determinada classe (WANI, BHAT, *et al.*, 2020).

*Fully connected layer* podem ser adicionadas para aprender combinações ainda mais sofisticadas das características. A saída da última *fully connected layer* alimenta uma função de ativação que gera as pontuações da classe. Como citado na subseção 3.1.9, existem várias funções de ativação: sigmóide, tanh, softmax, ReLU etc. A função de ativação escolhida produzirá escores de classe, e a classe com a pontuação mais alta é tratada como a correta (WANI, BHAT, *et al.*, 2020).

*Fully connected layer* utilizam o algoritmo *backpropagation* para atualizar seus pesos, o qual foi apresentado na subseção 3.1.6.

### 3.1.12. *Dropout*

As CNNs são constituídas por várias camadas ocultas seguidas por uma *full connected layer*. Esta característica da CNN pode levar ao desenvolvimento de *overfitting*, quando a rede tem um desempenho tão bom nos dados de treinamento que é incapaz de generalizar para novos dados (WANI, BHAT, *et al.*, 2020). Para reduzir os impactos do *overfitting* na rede treinada,

as CNNs utilizam a técnica de *dropout*, que é o abandono de neurônios de uma rede neural durante seu treinamento<sup>5</sup> (WANI, BHAT, *et al.*, 2020). Os neurônios abandonados não contribuem com *forward pass* nem com *backward pass*. Ao usar o *dropout*, a rede é forçada a aprender as características mais robustas à medida que a arquitetura da rede muda com cada entrada (BALAS, ROY, *et al.*, 2019; WANI, BHAT, *et al.*, 2020). Um exemplo ilustrativo de *dropout* é apresentado na Figura 13.

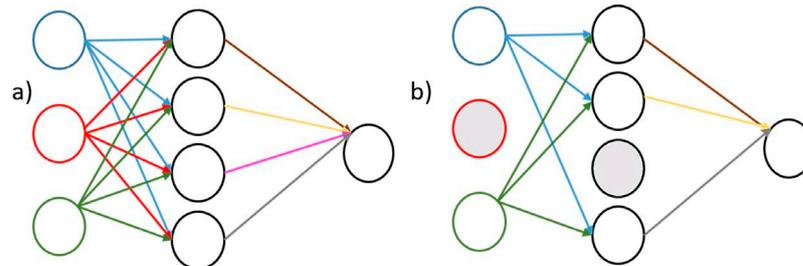


Figura 13 – Efeito *dropout*: a) uma rede neural simples, b) uma rede neural após o *dropout*.

### 3.1.13. Treinamento de uma CNN

O treinamento de uma CNN é baseado no conceito de minimização de uma função de custo. Ou seja, a rede na fase de treinamento procura valores para seus pesos de tal forma que a função custo seja minimizada. O *gradient descent* é uma técnica de otimização utilizada para minimizar o erro pelo cálculo dos gradientes necessários à atualização dos valores dos parâmetros da rede. O algoritmo de aprendizado mais comum, e bem-sucedido, para modelos de aprendizado profundo é o *backpropagation* baseado em *gradient descent*, em que o erro é propagado de trás para a frente, ou seja, da última camada para a primeira discutido na seção 3.6. Assim, no treinamento são definidos aspectos importantes, explicadas a seguir.

- **Divisão da base de imagens/dados em subconjuntos:** treinamento, validação e teste. De acordo com a literatura (WANI, BHAT, *et al.*, 2020, AGGARWAL e OTHERS, 2018, BALAS, ROY, *et al.*, 2019, PEOPLE, 2021) uma boa proporção de divisão podem ser:
  - ✓ 70-15-15, onde 70% de todo o conjunto de imagens será para o treinamento, 15% para validação e 15% para teste.
  - ✓ 80-10-10, onde 80% de todo o conjunto de imagens será para o treinamento, 10% para validação e 10% para teste.

<sup>5</sup> Abandonar um neurônio significa desconectá-lo temporariamente da rede, incluindo todas as suas conexões internas e externas ( WANI, BHAT, *et al.*, 2020).

O subconjunto de treinamento representa os dados que serão usados para treinar o modelo e precisa ser grande o suficiente para obter resultados significativos. O subconjunto de validação representa os dados que foram retidos no treinamento e serão usados para avaliar e ajustar os hiperparâmetros do modelo treinado e, portanto, ajustar o desempenho. Por fim, o subconjunto de testes representa dados que também foram retidos e serão usados para confirmar os resultados do modelo final (WANI, BHAT, *et al.*, 2020, AGGARWAL e OTHERS, 2018, BALAS, ROY, *et al.*, 2019, PEOPLE, 2021) conforme é apresentado na Figura 14 .

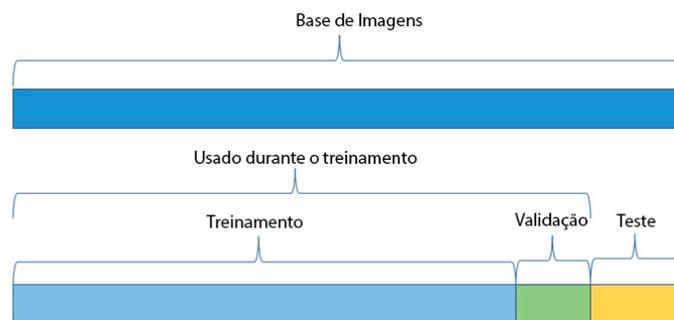


Figura 14 – Divisão da base de Imagens

- **Inicialização:** Os pesos de uma rede neural são inicializados aleatoriamente ou usando uma distribuição de probabilidade. Uma **entrada** é alimentada pela rede para obter a saída. A **saída obtida** e a **saída desejada** são então usadas para calcular o erro usando alguma função de custo, também denominada medida de erro (WANI, BHAT, *et al.*, 2020).
- **Batch size:** devido ao uso de algoritmos de otimização apresentados na seção 3.5 (AdaGrad, AdaDelta, RMSPro, ADAM) e suas variantes, é preciso definir o *batch size* e exemplos a serem utilizados em cada iteração no treinamento. Essa escolha deve levar em consideração restrições de memória, mas também os algoritmos de otimização empregados. Um *batch size* pequeno pode tornar mais difícil a minimização do custo, mas um tamanho muito grande também pode degradar a velocidade de convergência dos algoritmos de otimização (WANI, BHAT, *et al.*, 2020).
- **Número de épocas:** o número de ciclos em que será feito o treinamento (WANI, BHAT, *et al.*, 2020).
- **Critério de parada:** Escolha de um certo número de épocas executadas ou um erro mínimo alcançado (WANI, BHAT, *et al.*, 2020).
- **Taxa de aprendizado:** controla a velocidade da descida do gradiente em direção ao mínimo global (WANI, BHAT, *et al.*, 2020).

## 3.2. TRANSFORMADAS *WAVELET*

### 3.2.1. Introdução

A transformada de Fourier pode ser aplicada diretamente na construção de filtros utilizados no processamento de imagens digitais. O objetivo geral é obter informações sobre as frequências espaciais presentes na imagem e suas orientações e variações das intensidades nos tons de cinza da imagem. Assim, a partir do momento em que estas informações são determinadas é possível aplicar um filtro no domínio da frequência visando eliminar as componentes que apresentam frequências indesejadas (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013).

As técnicas de filtragem no domínio da frequência fornecem uma ferramenta prática para determinadas aplicações, em especial, a segmentação de imagens médicas (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013). A análise de uma imagem no domínio da frequência pode ser realizada com diferentes objetivos, por exemplo, o aguçamento, a suavização ou a redução do nível de ruído presente nas imagens.

A transformada de Fourier recebeu a atenção da comunidade de processamento de sinais no início da década de 1980. A versão discreta da transformada de Fourier utiliza funções exponenciais complexas para o processo de filtragem da imagem. Em sua versão bidimensional a transformada de Fourier fornece informações a respeito das variações presentes nos pixels da imagem. Especificamente a transformada discreta de Fourier é uma transformação de coordenadas que resulta em componentes pertencentes ao conjunto dos números complexos, em que cada coeficiente é obtido pela combinação linear dos elementos de entrada com o núcleo da transformação (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013).

A transformada de Fourier é uma análise do conteúdo de frequência global do sinal. Existem aplicações em processamento de imagens que requerem que a análise seja localizada no domínio espacial. Isso pode ser resolvido introduzindo a frequência espacial na análise de Fourier. A maneira clássica de fazer isso é por meio da chamada Transformada de Fourier em janela. A ideia central de janelamento é refletida na *Short Time Fourier Transform* (STFT) (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013). A transformação em janela de  $f(x)$  é dada como mostrado na Equação 29:

$$F(w, \alpha) = \int_{-\infty}^{+\infty} f(x)g(x - \alpha)\exp^{-jwx} dx \quad \text{Equação 29}$$

Onde  $w$  representa a frequência espacial e  $\alpha$  denota a posição da janela. Pode-se notar aqui que o Gaussiano está bem localizado em torno do tempo  $x = \alpha$ . Portanto, a Equação 29 transforma

o sinal  $f(x)$  em uma pequena janela ao redor de  $\alpha$ . A STFT transmite o componente de frequência localizado presente no sinal durante uma pequena janela espacial. O mesmo conceito pode ser estendido a uma imagem espacial bidimensional onde os componentes de frequência localizados podem ser determinados a partir da transformada em janela. Esta é uma das bases da compreensão conceitual das transformadas *Wavelet* (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013).

Matematicamente, uma “onda” é expressa como uma função sinusoidal (ou oscilante) de tempo ou espaço. A análise de Fourier expande um sinal arbitrário em termos de um número infinito de funções sinusoidais de seus harmônicos e tem sido bem estudada pela comunidade de processamento de sinal por décadas. A representação de Fourier de sinais é conhecida por ser muito eficaz na análise de sinais periódicos invariantes no tempo (estacionários). Em contraste com uma função sinusoidal, uma *wavelet* é uma pequena onda cuja energia é concentrada no tempo. *Wavelets* permitem a análise de tempo e frequência de sinais simultaneamente devido ao fato de que a energia das *wavelets* é concentrada no tempo e ainda possui as características de onda (periódicas). Como resultado, a representação de *wavelet* fornece uma ferramenta matemática versátil para analisar sinais transitórios, variantes no tempo, não estacionários, que não são estatisticamente previsíveis, especialmente na região de descontinuidades, que é uma característica típica de imagens com descontinuidades nas bordas (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013).

### 3.2.2. Transformada *wavelet* contínua (CWT)

*Wavelets* são funções geradas a partir de uma função de base chamada protótipo ou *wavelet* mãe, mediante operações de escalas e translações no domínio do tempo (frequência). Se a *wavelet* mãe é denotada por  $\psi(t)$ , as outras *wavelet*  $\psi_{a,b}(t)$  podem ser representadas de acordo com a Equação 30 (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013; DEBNATH e SHAH, 2017).

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \quad \text{Equação 30}$$

Onde  $a$  e  $b$  são dois números reais arbitrários. As variáveis  $a$  e  $b$  representam os parâmetros para escala e translação, respectivamente, no eixo do tempo. Da Equação 30 decorre que a *wavelet* mãe pode ser essencialmente representada pela Equação 31

$$\psi(t) = \psi_{1,0}(t) \quad \text{Equação 31}$$

Para qualquer  $a \neq 1$  e  $b = 0$  arbitrário, pode-se derivar a Equação 32

$$\psi_{a,0}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t}{a}\right) \quad \text{Equação 32}$$

Conforme mostrado na Equação 32,  $\psi_{a,0}(t)$  nada mais é do que uma escala de tempo (por  $a$ ) e escala de amplitude (por  $\sqrt{|a|}$ ) versão da função *wavelet* mãe  $\psi(t)$  na Equação 31. O parâmetro  $a$  causa contração de  $\psi(t)$  no eixo do tempo quando  $a < 1$  e expansão ou alongamento quando  $a > 1$ . É por isso que o parâmetro  $a$  é chamado de parâmetro de escala. Para  $a < 0$ , a função  $\psi_{a,0}(t)$  resulta na reversão do tempo com dilatação.

Matematicamente, substitui-se  $t$  na Equação 32 por  $t - b$  para causar uma translação ou deslocamento no eixo do tempo, resultando na função *wavelet*  $\psi_{a,b}(t)$  conforme Equação 30. A função  $\psi_{a,b}(t)$  é um deslocamento de  $\psi_{a,0}(t)$  à direita ao longo do eixo do tempo por um valor  $b$  quando  $b > 0$ , enquanto que é um deslocamento à esquerda ao longo do eixo do tempo por um valor  $b$  quando  $b < 0$ . É por isso que a variável  $b$  representa o domínio da translação no tempo (mudança na frequência).

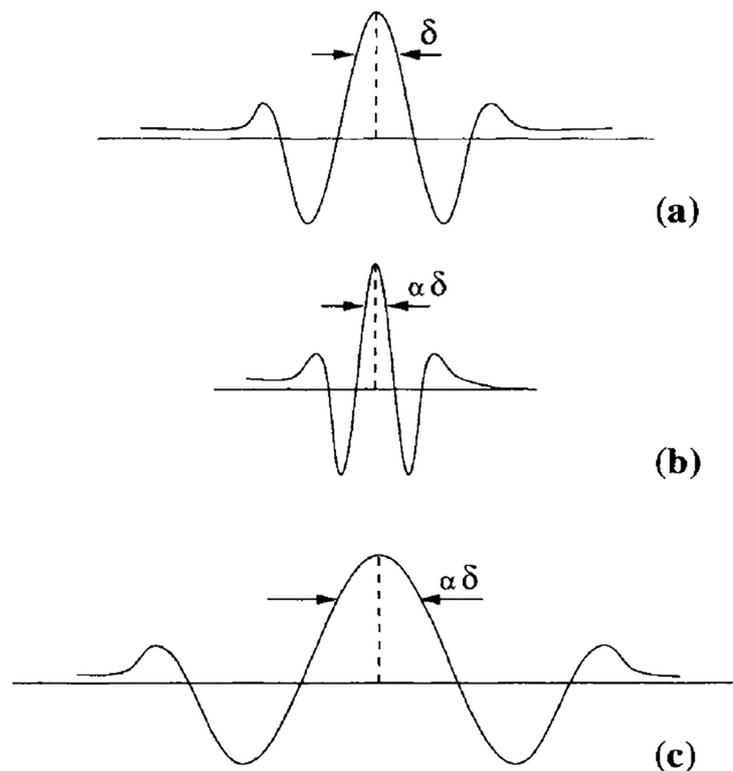


Figura 15 – (a) *Wavelet* mãe, (b)  $\psi\left(\frac{t}{a}\right)$ :  $0 < \alpha < 1$ , (c)  $\psi\left(\frac{t}{a}\right)$ :  $\alpha > 1$ . Fonte: (ACHARYA e RAY, 2005)

A Figura 15 mostra a ilustração de uma *wavelet* mãe e suas dilatações no domínio do tempo com o parâmetro de escala  $a$ . Para a *wavelet* mãe  $\psi(t)$  mostrada na Figura 15 (a), uma contração do sinal no eixo do tempo quando  $a < 1$  é mostrada na Figura 15 (b) e expansão do sinal no eixo do tempo quando  $a > 1$  é mostrada na Figura 15 (c). Com base nesta definição de

*wavelets*, a transformada *wavelet* de uma função (sinal)  $f(t)$  é matematicamente representada pela Equação 33 (ACHARYA e RAY, 2005)

$$Wf(a, b) = \int_{-\infty}^{+\infty} f(t)\psi_{a,b}(t)dt \quad \text{Equação 33}$$

A transformação inversa para reconstruir  $f(t)$  de  $W(a, b)$  é matematicamente representada pela Equação 34

$$f(t) = \frac{1}{c} \int_{a=-\infty}^{+\infty} \int_{b=-\infty}^{+\infty} \frac{1}{|a|^2} W(a, b)\psi_{a,b}(t) da dt \quad \text{Equação 34}$$

Onde

$$c = \int_{-\infty}^{+\infty} \frac{|\psi(\omega)|^2}{|\omega|} d\omega$$

Onde  $\psi(\omega)$  é a transformada de *Fourier* da *wavelet* mãe  $\psi(t)$ .

Se  $a$  e  $b$  são duas variáveis contínuas (não discretas) e  $f(t)$  também é uma função contínua,  $W(a, b)$  é chamada de transformada *wavelet* contínua (CWT). Portanto, o CWT mapeia uma função unidimensional  $f(t)$  para uma função  $W(a, b)$  de duas variáveis reais contínuas  $a$  (escala) e  $b$  (translação).

### 3.2.3. Transformada discreta de *wavelet*

Uma vez que o sinal de entrada, por exemplo, uma imagem digital, é processado por uma máquina de computação digital, é prudente definir a versão discreta da transformada *wavelet*. Antes de definir a transformada *wavelet* discreta, é essencial definir as *wavelets* em termos de valores discretos dos parâmetros de escala e translação  $a$  e  $b$  em vez de serem contínuos. Há muitas maneiras de discretizar  $a$  e  $b$  e, em seguida, representar as *wavelets* discretas. A abordagem mais popular para discretizar os parâmetros da *wavelet* é usar a Equação 35 (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013; DEBNATH e SHAH, 2017)

$$a = a_0^m, b = nb_0 a_0^m \quad \text{Equação 35}$$

Onde  $m$  e  $n$  são inteiros. Substituindo  $a$  e  $b$  na Equação 30 pela Equação 35, a *Wavelet* discreta pode ser representada pela Equação 36

$$\psi_{m,n}(t) = \frac{1}{\sqrt{a_0^m}} \psi\left(\frac{t - nb_0 a_0^m}{a_0^m}\right) \quad \text{Equação 36}$$

Que pode ser reescrita conforme a Equação 37

$$\psi_{m,n}(t) = a_0^{-\frac{m}{2}} \psi(a_0^{-m}t - nb_0) \text{ Equação 37}$$

Existem muitas opções para selecionar os valores de  $a_0$  e  $b_0$ . A escolha mais comum foi selecionada aqui:  $a_0 = 2$  e  $b_0 = 1$ ; portanto,  $a = 2^m$  e  $b = n2^m$ . Isso corresponde à amostragem (discretização) de  $a$  e  $b$  de tal forma que os valores discretos consecutivos de  $a$  e  $b$ , bem como os intervalos de amostragem, diferem por um fator de dois. Esta forma de amostragem é popularmente conhecida como amostragem diádica e a decomposição correspondente dos sinais é chamada de decomposição diádica. Usando esses valores, pode-se representar as ondas discretas como na Equação 38, que constitui uma família de funções de base ortonormais,

Em geral, os coeficientes *wavelet* para a função  $f(t)$  são dados pela Equação 39

$$\underbrace{\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \text{ CWT}}_{a = 2^m \text{ e } b = n2^m}$$

$$\underbrace{\psi_{m,n}(t) = \frac{1}{\sqrt{2^m}} \psi\left(\frac{t - n2^m}{2^m}\right) \text{ DWT}}$$

$$\psi_{m,n}(t) = 2^{-m/2} \psi(2^{-m}t - n) \text{ DWT} \quad \text{Equação 38}$$

$$c_{m,n}(f) = 2^{-\frac{m}{2}} \int f(t) \psi(2^{-m}t - n) dt \quad \text{Equação 39}$$

Isso permite reconstruir o sinal  $f(t)$  a partir dos coeficientes *wavelet* discretos como é apresentado na Equação 40:

$$f(t) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} c_{m,n}(f) \psi_{m,n}(t) \quad (4.11) \quad \text{Equação 40}$$

Em aplicações de processamento de imagem, o sinal de entrada  $f(t)$  precisa ser discreto por causa da amostragem digital dos dados originais, que é representada por um número finito de *bits*. Quando a função de entrada  $f(t)$ , bem como os parâmetros *wavelet*  $a$  e  $b$  são representados de forma discreta, a transformação é comumente referida como a transformada *wavelet* discreta do sinal  $f(t)$  (ACHARYA e RAY, 2005; CHAN, 1994; SHUKLA e TIWARI, 2013; DEBNATH e SHAH, 2017).

Assim, a transformada *wavelet* se baseia em pequenas ondas (*wavelets*) de frequência variada e duração limitada. A transformada *wavelet* discreta pode ser considerada como uma sequência de números que mostra uma certa função contínua.

Quando as imagens digitais são tratadas em várias resoluções, a transformada *wavelet* discreta é uma ferramenta matemática viável. Além de sua estrutura eficiente e altamente intuitiva para representação e armazenamento de imagens multiresolução, a transformada *wavelet* discreta fornece uma visão poderosa das características espaciais e de frequência de uma imagem (JANSEN, 2012; RAFAEL, 2006; WILLIAMS e LI, 2016; CHAN, 1994; DEBNATH e SHAH, 2017).

### 3.2.4. Conceito de Análise Multiresolução

Muitas funções de base de *wavelet* ortonormais da forma  $\psi_{m,n}(t) = 2^{-\frac{m}{2}}\psi(2^{-m}t - n)$  foram desenvolvidas na década de 1980. A teoria da análise de multiresolução apresentou uma abordagem sistemática para gerar as *wavelets*. A ideia da análise de multiresolução é aproximar uma função  $f(t)$  a diferentes níveis de resolução.

Na análise de multiresolução, são consideradas duas funções: a *wavelet* mãe  $\psi_{m,n}$  e a função de escala  $\phi(t)$ . A versão dilatada (dimensionada) e traduzida (deslocada) da função de escala é dada por  $\phi_{m,n}(t) = 2^{\frac{m}{2}}\phi(2^{-m}t - n)$ . Para  $m$ , fixo, o conjunto de funções de escala  $\phi_{m,n}(t)$  são ortonormais.

Assim, ao se ter uma função de escala na resolução  $m - l$ , pode-se decompor em uma soma de funções começando com uma aproximação de resolução mais baixa seguido por uma sequência de funções geradas por dilatações da *wavelet* que representam a perda de informação em termos de detalhes. Seja considerada a representação de uma imagem com cada vez menos *pixels* em  $t$  níveis sucessivos de aproximação. Os coeficientes *wavelet* podem então ser considerados como as informações de detalhes adicionais necessárias para ir de uma aproximação mais grosseira para uma mais fina.

Portanto, em cada nível de decomposição, o sinal pode ser decomposto em duas partes, uma é a aproximação grosseira do sinal na resolução mais baixa e a outra é a informação detalhada que foi perdida por causa da aproximação. Os coeficientes *wavelet* derivados na Equação 39, descrevem a informação (detalhe) perdida ao ir de uma aproximação do sinal na resolução  $2^{m-1}$  para a aproximação mais grosseira na resolução  $2^m$ .

### 3.2.5. Transformada *wavelet* em duas dimensões

A extensão 2D do DWT é essencial para imagens. Um sinal bidimensional (imagem) pode ser representado por uma matriz 2D  $x[M, N]$  com  $M$  linhas e  $N$  colunas, onde  $M$  e  $N$  são números inteiros não negativos. A abordagem simples para a implementação 2D da transformada *wavelet* discreta é realizar a transformada *wavelet* discreta unidimensional, em linha, para produzir um resultado intermediário; e, em seguida, realizar a mesma transformada *wavelet* discreta unidimensional em coluna, neste resultado, intermediário para produzir o resultado. Isso é mostrado na Figura 16 (a) e (b).

Isso é possível porque as funções de escala bidimensional podem ser expressas como funções separáveis que são o produto de duas funções de escala unidimensionais, como  $\phi_2(x, y) = \phi_1(x)\phi_1(y)$ . Ele é verdadeiro para a função  $\psi(x, y)$ . Aplicando a transformação unidimensional em cada linha, são produzidas duas sub-bandas em cada linha. Quando as sub-bandas de baixa frequência de todas as linhas ( $L$ ) são colocadas juntas, parece uma versão fina (de tamanho  $M \frac{N}{2}$ ) do sinal de entrada como mostrado na Figura 16 (a). Da mesma forma, as sub-bandas de alta frequência de todas as linhas foram reunidas para produzir a sub-banda  $H$  (de tamanho  $M \frac{N}{2}$ ) que contém principalmente as informações de alta frequência em torno das discontinuidades (bordas em uma imagem) no sinal de entrada. Em seguida, aplicando uma transformada *wavelet* discreta unidimensional em colunas nessas sub-bandas  $L$  e  $H$  (resultado intermediário), foram produzidas quatro sub-bandas  $LL$ ,  $HL$ ,  $LH$  e  $HH$  mostrado na Figura 16 (b).  $LL$  é uma versão mais grosseira do sinal de entrada original chamada de coeficiente de aproximação.  $HL$ ,  $LH$ , e  $HH$  são a sub-banda de alta frequência que contém as informações detalhadas chamados coeficientes de detalhe. A transformada *wavelet* discreta unidimensional poderia ser aplicada, primeiramente em colunas e depois, em linhas para se obter o mesmo resultado.

A transformada *wavelet* executa primeiro uma etapa da transformada em todas as linhas, produzindo uma matriz em que o lado esquerdo contém os coeficientes *low pass* ( $L$ ) de cada linha e o direito contém os coeficientes *high pass* ( $H$ ). Em seguida, é aplicada a todas as colunas, resultando em quatro tipos de coeficientes, conforme ilustrado na Figura 16 (JANSEN, 2012; RAFAEL, 2006; WILLIAMS e LI, 2016):

- As características diagonais da imagem são geradas a partir de uma convolução com o filtro *high pass* em ambas as direções ( $HH$ );

- As características horizontais da imagem são geradas a partir de uma convolução do filtro *low pass* nas colunas, seguido pelo filtro *high pass* nas linhas (*LH*);
- As características verticais da imagem são geradas a partir de uma convolução do filtro *high pass* nas linhas, seguido pelo filtro *low pass* nas colunas (*HL*);
- O Coeficiente de aproximação da imagem é gerado a partir de uma convolução com o filtro *low pass* (*LL*) em ambas as direções.

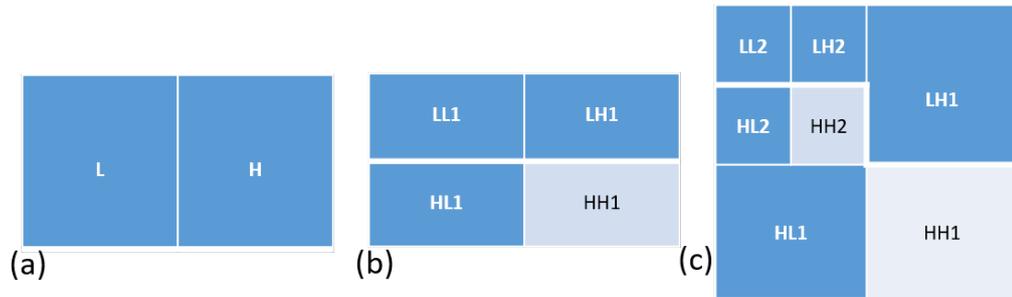


Figura 16 – Transformação *wavelet* bidimensional. (a) Aplica-se uma etapa de transformação unidimensional a todas as linhas. (b) Em seguida, o mesmo procedimento é repetido para todas as colunas. (c) Nesta etapa, os coeficientes resultantes de uma convolução são expandidos em ambas as direções. Fonte: Adaptado de (JANSEN, 2012)

Conforme mostrado na Figura 16(b) e Figura 16(c), a sub-banda *LL1* pode ser posteriormente decomposta em quatro sub-bandas *LL2*, *LH2*, *HL2*, e *HH2* com base no princípio da análise de multiresolução. O mesmo cálculo pode continuar a decompor *LL2* em níveis superiores. A Figura 17 mostra o resultado da transformação *wavelet* em um nível de decomposição de uma radiografia de tórax. Assim a imagem original executa uma etapa da transformada *wavelet* em todas as linhas, produzindo no lado esquerdo o *low pass* (*L*) e no direito, *high pass* (*H*) como mostra a Figura 16 (a). Na sequência, é aplicado a todas as colunas, resultando em quatro tipos de coeficientes aproximada (*LL1*), horizontal (*LH1*), vertical (*HL1*) e diagonal (*HH1*) como e mostrado na a Figura 16 (b) e na Figura 17.

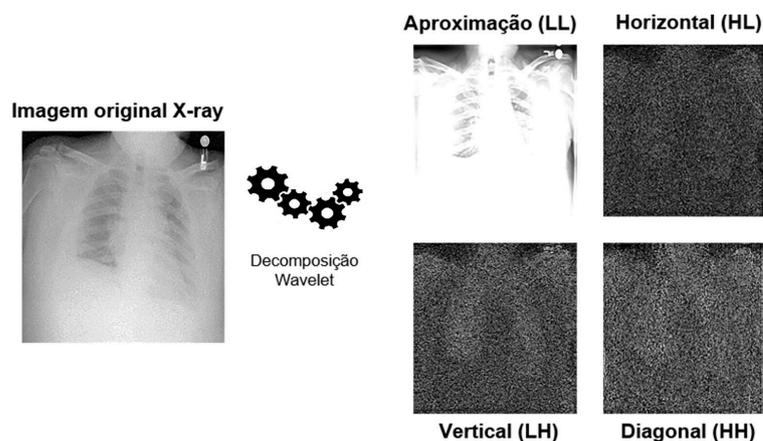


Figura 17 – Exemplo de decomposição de um nível da DWT

## 4. Primeira Contribuição: Classificação COVID-19 em radiografias de tórax usando uma nova rede neural convolucional: CNN-COVID

---

*“Este capítulo é baseado no artigo completo publicado no periódico Research on Biomedical Engineering, Classificação Qualis CAPES 2019 Qualis A4, de autoria do próprio autor desta tese. – DOI: <https://doi.org/10.1007/s42600-020-00120-5>”.*

### 4.1.Introdução

Em dezembro de 2019, um grupo de pacientes com pneumonia atípica, de causa desconhecida, foi associado ao consumo de carne de morcego comprada em um mercado de carne de animais exóticos em Wuhan, Hubei, China. A doença se espalhou rapidamente para outros cantos do mundo e, em 11 de março de 2020, a Organização Mundial da Saúde declarou pandemia de COVID-19, que ainda está em andamento (Zhu et al. 2019; Yang et al. 2020). Ao usar o sequenciamento de amostras de pacientes, foi possível identificar um novo tipo de beta coronavírus. Esse novo coronavírus, então denominado 2019-nCoV, foi comparado aos vírus causadores da Síndrome Respiratória Aguda Grave (SARS) e da Síndrome Respiratória do Oriente Médio (MERS), mas apresentando menor taxa de mortalidade e maior capacidade de transmissão (Zhu et al. 2019; Yang et al. 2020).

Os coronavírus são vírus de RNA, com um envelope lipídico, capazes de causar doenças respiratórias, entéricas, hepáticas e neurológicas em animais domésticos e seres humanos (Zhu et al. 2019; Weiss e Leibowitz 2011). A existência de vírus selvagens na natureza é bem conhecida pelos pesquisadores. Eles são altamente prevalentes, amplamente distribuídos, geneticamente diversos e frequentemente recombinados geneticamente.

Tanto o SARS quanto o MERS foram associados a uma origem zoonótica e transmitidos por civetas e camelos, respectivamente. O aparecimento de doenças como SARS e MERS, supostamente causadas pelo novo coronavírus, tende a se tornar eventos mais frequentes quando não são impostas barreiras entre a sociedade humana e a natureza selvagem (Zhu et al. 2019; Cui et al. 2019).

Como não há medicamento ou vacina específica<sup>6</sup> para tratar o novo coronavírus, a detecção precoce é fundamental para que o paciente seja isolado da população saudável o quanto

---

<sup>6</sup> Na data de publicação deste artigo as vacinas eram inexistentes.

antes (Ai, 2020). Assim, a investigação por métodos de detecção precoce tornou-se essencial para combater este surto pandêmico.

Atualmente, o padrão-ouro do diagnóstico de COVID-19 é feito pela detecção de ácido nucleico viral, usando reação em cadeia da polimerase de transcrição reversa (RT-PCR) em tempo real, embora sua precisão efetiva seja de 30 a 50% (Ai 2020; Zhang et al. 2020; Ozturk et al. 2020).

Um problema subjacente a esse método é a indisponibilidade em várias regiões e países afetados; o que pode gerar problemas logísticos e políticos para fornecer *kits* de teste suficientes, para o número crescente de pacientes suspeitos de ter a doença (Zhang et al. 2020). Além disso, a demora no processamento, obtenção de resultados e o número significativo de falsos negativos incitaram pesquisadores de todo o mundo a tentarem encontrar uma solução para esse problema em diversas áreas do conhecimento (Ozturk et al. 2020).

O processamento de imagens médicas é uma das áreas que vem contribuindo para estudos promissores. Pesquisas nesta área estão sendo feitas para auxiliar clinicamente no diagnóstico da doença para pacientes que desenvolvem pneumonia atípica pulmonar, usando imagens de radiografia de tórax ou Tomografia Computadorizada (TC) (Ozturk et al. 2020). Como vários pacientes com COVID-19 desenvolvem infecção pulmonar, a Tomografia Computadorizada é útil para detectar comprometimento pulmonar, bem como para classificar sua progressão (Zhang et al. 2020; Dai et al. 2020).

As imagens radiológicas de pacientes com COVID-19 podem apresentar semelhanças com as de pacientes com pneumonia bacteriana ou viral, especificamente as causadas por SARS e MERS. Assim, a capacidade de diferenciar doenças com precisão por meio da análise de imagens médicas tornou-se um desafio vital e superá-lo significa ajudar os profissionais de saúde a detectar a doença precocemente e isolar os pacientes afetados o mais rápido possível (Ozturk et al. 2020; Chung et al. 2020).

A literatura da área de processamento de imagens médicas já conta com diversos estudos voltados ao desenvolvimento de métodos de aprendizado de máquina capazes de auxiliar no diagnóstico da COVID-19, seja por meio de imagens de Tomografia Computadorizada ou de radiografia de tórax (Ozturk et al. 2020). Problemas ao usar tomografias computadorizadas em vez de radiografias de tórax incluem baixa disponibilidade de equipamentos, radiologistas e médicos; custo mais alto; e maior tempo para obtenção das imagens (Zhang et al. 2020).

Uma técnica de aprendizado de máquina usada em pesquisas é chamada de *deep learning*. Ela permite que modelos de computador, com várias camadas de processamento, aprendam a representar dados em vários níveis de abstração (Zhang et al. 2020; Lecun et al. 2015;

Martin et al. 2020). Esta técnica permite o projeto de aplicações que podem realizar o reconhecimento, como reconhecimento de fala, reconhecimento visual e detecção de objetos.

Imagens médicas de pacientes com COVID-19 apresentam características comuns que podem apresentar um padrão. *Deep learning* é eficaz usada por pesquisadores para ajudar os profissionais de saúde a analisar grandes volumes de dados gerados por imagens de radiografia de tórax, por exemplo (Zhang et al. 2020; Martin et al. 2020).

Quando se trata de aplicar inteligência artificial e *machine learning* máquina à área médica, as CNNs se destacam. CNNs são redes neurais artificiais profundas que podem ser usadas para classificar imagens, agrupá-las por similaridade e executar o reconhecimento de objetos. Essas redes são inspiradas no processamento do córtex visual humano e usadas para imagens médicas onde irregularidades na morfologia dos tecidos podem ser usadas para classificar tumores. As CNNs podem detectar padrões difíceis de serem encontrados por especialistas humanos, por exemplo, estágios iniciais de doenças em amostras de tecidos (Balas et al., 2020).

Como a investigação e o desenvolvimento de *aprendizado de máquina* são os principais temas de interesse desta pesquisa, outros resultados já haviam sido divulgados. Por exemplo, no artigo *Técnicas de seleção de atributos para classificação de imagens radiográficas*, apresentado e publicado no XXVII Congresso Brasileiro de Engenharia Biomédica, em outubro de 2020 (Costa et al. 2020), foram utilizadas técnicas tradicionais de extração de atributos para verificar a acurácia. Como o resultado do uso das técnicas mostrou-se inadequado, ficou claro a necessidade de utilizar técnicas como *deep learning* para melhorar o resultado, assim foi decidido utilizar *deep neural network* para obter resultados ainda melhores.

Este artigo propõe um novo modelo de Rede Neural Convolutacional chamado CNN-COVID, para classificar imagens de pacientes com COVID-19 e diferenciá-los daqueles que não têm COVID-19, com foco em análise, classificação e alta acurácia. O modelo CNN-COVID e os trabalhos relacionado a ele são apresentados nas seções a seguir.

## 4.2. Trabalhos Relacionados

No estudo de Zang *et al.* (ZHANG, XIE, *et al.*, 2020) foi desenvolvido um modelo de detecção de anomalias, mediante utilização de *deep learning* com o objetivo de realizar uma triagem rápida e confiável de pacientes com COVID-19. Para avaliar o desempenho do modelo, foram utilizadas 100 radiografias de tórax de 70 pacientes com COVID-19 de um repositório no Github e 1431 radiografias de tórax de 1008 pacientes com pneumonia de um conjunto de dados público, o ChestX-ray14. O modelo em (ZHANG, XIE, *et al.*, 2020) é composto de três componentes: *backbone network*, *classification head* e *anomaly detection head*. Desta forma, a

partir de uma radiografia de tórax, o primeiro componente extrai as características em alto nível que, em seguida, vão compor os dados de entrada da *classification head* e posteriormente vão compor os dados de entrada da *anomaly detection head*. Assim, neste trabalho foi possível obter em uma sensibilidade de 90,00% e uma especificidade de 87,84% (quando o parâmetro T, utilizado no estudo, foi de 0,25) ou a sensibilidade de 96,00% e especificidade de 70,65% (quando o parâmetro T foi de 0,15). Entretanto, o modelo apresentou algumas limitações, como a perda de 4% de casos de COVID-19 e aproximadamente 30% de falsos positivos.

Em um estudo similar Sethy *et al.* (SETHY e BEHERA, 2020), foi sugerido um modelo para a detecção do novo coronavírus a partir de radiografias de tórax, utilizando *deep learning*. Assim, o *Support Vector Machine* (SVM) pode classificar as imagens dos pacientes acometidos por essa doença e diferenciá-las das imagens de pacientes que possuíam outras doenças. Foram utilizadas 25 imagens COVID-19 de um repositório no *Github* e 25 imagens de Pneumonia do repositório *Kaggle*. Os resultados obtidos pelo modelo de rede neural profunda *ResNet50* com classificação SVM alcançaram os melhores resultados para classificação e detecção do COVID-19, apresentando acurácia de 95,38%, sensibilidade de 97,2% e especificidade de 93,4%.

Em Wang *et al.* (WANG, LIN e WONG, 2020) foi implementada uma rede CNN adaptada para a detecção de casos de COVID-19, denominado COVID-Net, a partir de imagens de radiografias de tórax que são de código aberto e disponíveis ao público em geral. Assim, foi montado um banco chamado COVIDX formado por 5 bases diferentes (WANG, LIN e WONG, 2020), das quais foram retiradas 100 amostras de CXRs normais, 100 de Pneumonia e 100 imagens COVID-19. A COVID-NET apresentou os seguintes resultados: acurácia de 93,3%, sensibilidade de 91,0% e especificidade de 99,9% na detecção de imagens de radiografia de tórax COVID-19 de casos normais e graves de síndrome respiratória aguda.

Em Abbas *et al.* (ABBAS, ABDELSAMEA e GABER, 2020) o trabalho utiliza a técnica de *transfer learning* que é um mecanismo eficaz que pode fornecer uma solução promissora transferindo conhecimento de tarefas genéricas de reconhecimento de objetos para tarefas específicas do domínio. Desta forma, foi adaptado uma *deep CNN* denominada DeTraC – com a função de Decompor, Transferir e Compor as amostras para a classificação das radiografias de tórax com COVID-19. Foram utilizadas 80 amostras de radiografias de tórax normais (com 4020×4892 *pixels*) da Sociedade Japonesa de Tecnologia Radiológica (JSRT), 105 imagens COVID-19 e 11 imagens SARS, com 4248×3480 *pixels*, de um repositório no *Github*. O DeTraC apresentou os seguintes resultados: uma acurácia de 95.12% com sensibilidade de 97.91% e especificidade de 91.87% na detecção de imagens de radiografias de tórax COVID-19 de casos normais e graves de síndrome respiratória aguda.

### 4.3. Materiais e métodos

O objetivo desta parte é apresentar o desenvolvimento de uma nova rede CNN chamada CNN-COVID para classificar imagens de tórax em COVID e NON-COVID, pertencentes a dois conjuntos de dados diferentes que serão descritos a seguir.

#### 4.3.1. Descrição do banco de imagens I

Duas bases de dados compõem o Conjunto de Dados I. A primeira contém 217 imagens de raios X de tórax da coleção de dados de imagens COVID-19 (COHEN, MORRISON e DAO, 2020). Neste banco de dados, as imagens são de 141 pacientes que tiveram resultado positivo para COVID-19 e foi rotulado como COVID. Do segundo banco de dados, ChestXray14 (WANG, PENG, *et al.*, 2019), foram utilizadas 1126 imagens. Todas essas imagens correspondem a imagens de raios X de tórax marcadas para a presença de 14 observações radiográficas de tórax comuns que, neste artigo, foram marcadas como NON-COVID.

Um subconjunto de 166 imagens, das 217 imagens do conjunto COVID-19, foi selecionado aleatoriamente (COHEN, MORRISON e DAO, 2020). Dessas 166 imagens selecionadas, cerca de 75% foram para a fase de treinamento e 25% para a fase de testes. Do conjunto NON-COVID, 1000 imagens foram selecionadas aleatoriamente, 80% para a fase de treinamento e 20% para a fase de teste.

Para a fase de validação, foram selecionadas 126 imagens do conjunto COVID-19 (de um total de 217 imagens) e 126 imagens do conjunto NON-COVID (de um total de 1126 imagens), conforme mostrado na Tabela 3

Tabela 3 – Número de imagens do conjunto de dados I.

Conjunto I	Total	Treinamento	Teste	Validação
COVID	217	126	40	126
NON-COVID	1126	800	200	126

Uma das dificuldades encontradas durante o estudo foi o número limitado de radiografias de pacientes com COVID-19 usadas para o treinamento de modelos de aprendizado profundo. Para superar esse problema, foi utilizada uma classe `ImageDataGenerator`, que permite a geração de novas imagens no conjunto de treinamento (AGGARWAL e OTHERS, 2018, CHOLLET, 2016). Assim foi possível equilibrar o conjunto de dados das duas classes COVID-19 e NON COVID, nos conjuntos de treinamento e teste.

### 4.3.2. Descrição do banco de imagens II

O conjunto de dados II é composto por imagens do repositório *Valencian Region Medical ImageBank* (BIMCV) (DE LA IGLESIA VAYÁ, SABORIT, *et al.*, 2020) que contém raios X de tórax e imagens de Tomografia Computadorizada (TC) de pacientes com COVID-19+. A partir desse repositório, dois bancos de dados foram criados: COVID e NON-COVID. O COVID tem 2.025 imagens de raios X de tórax de pacientes com teste positivo para COVID-19 e banco de dados NON-COVID, 2.025 imagens de raios X de tórax de pacientes com teste negativo para a doença.

Dentre as 2025 imagens, selecionadas nas bases de dados COVID e NON-COVID, aproximadamente 75% delas foram usadas na fase de treinamento, 15% na fase de teste e 10% na fase de validação, conforme mostra a Tabela 4

Tabela 4 – Número de imagens do conjunto de dados II.

Conjunto II	Total	Treinamento	Teste	Validação
COVID	2025	1419	303	303
NON-COVID	2025	1419	303	303

### 4.3.3. Metodologia

#### 4.3.3.1. Desenvolvimento de estrutura CNN-COVID

CNNs foram propostas para avaliar dados de imagem. O nome vem do operador de convolução, uma maneira simples de fazer operações complexas usando o *kernel* de convolução (RAVI, WONG, *et al.*, 2016)

Muitas variações da CNN foram propostas tais como *AlexNet* (KRIZHEVSKY, SUTSKEVER e HINTON, 2012), *Clarifai* (ZEILER e FERGUS, 2014), *GoogleNet* (SZEGEDY, LIU, *et al.*, 2015). A estrutura da CNN-COVID também é uma variação de uma CNN, que contém a seguinte arquitetura: camada de entrada, camada convolucional, camada densa e camada de saída, apresentada na Figura 18.

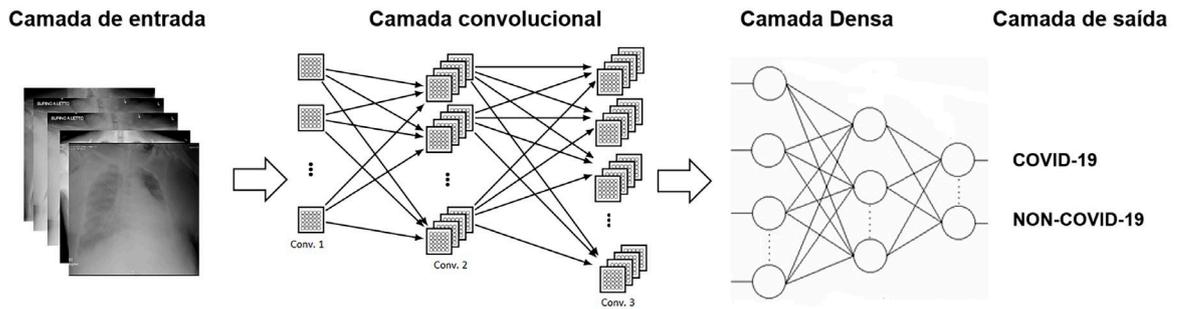


Figura 18 – Esquema de classificação da rede neural profunda

A arquitetura detalhada da rede CNN-COVID para classificação do COVID-19 é mostrada na Tabela 5. A rede consiste de camadas convencionais, incluindo camada de entrada, camada de convolução, camada de *max pooling* e *fully connected layer*.

Além disso, uma função ReLU é usada após cada camada de convolução (1<sup>a</sup>, 3<sup>a</sup>, 5<sup>a</sup> e 7<sup>a</sup>) e nas camadas densas (9<sup>a</sup>, 10<sup>a</sup>, 11<sup>a</sup> e 12<sup>a</sup>). Para diminuir a possibilidade do *overfitting*, o *dropout* é implementado nas quatro primeiras *fully connected layer* (9<sup>a</sup>, 10<sup>a</sup>, 11<sup>a</sup> e 12<sup>a</sup> camadas) com uma porcentagem de 20%.

Tabela 5 – Arquitetura da CNN-COVID. A rede contém a entrada (I), Convolução (C), *Max pooling* (M), Rede Totalmente Conectada (F)

Camada	CNN-Covid		
	Tipo	Dimensão do Filtro	Dimensão da Entrada/Saída
0	I		300x300
1	C	5x5x256	296x296x256
2	M	2x2	148x148x256
3	C	3x3x128	146x146x128
4	M	2x2	73x73x128
5	C	3x3x64	71x71x64
6	M	2x2	35x35x64
7	C	3x3x32	33x33x32x
8	M	2x2	16x16x32
9	F	16x16x32x256	1x256
10	F	1x1x256x128	1x128
11	F	1x1x128x64	1x64
12	F	1x1x64x32	1x32
13	F	1x132x2	1x2

#### 4.3.3.2. Treinamento da CNN-COVID

Na fase de treinamento os pesos são inicializados de forma aleatória. A rede foi treinada de acordo com o modelo ADAM (WANI, BHAT, *et al.*, 2020). Foram utilizados parâmetros

padrão  $\beta_1 = 0,9$  e  $\beta_2 = 0,999$  (KINGMA e BA, 2014), taxa de aprendizado inicial  $\alpha = 0,001$ , decaída por um fator de 10.

#### 4.3.3.3. Parâmetro da entrada CNN-COVID

Para escolher os valores dos parâmetros de entrada e o *batch size* da CNN-COVID foram testadas diversas opções, levando em consideração a capacidade de execução do hardware disponível. Para as entradas 200x200 e 220x220 o *batch size* foi 20. Já para entradas de tamanho maior, o *batch-size* foi 10. Os testes das entradas foram feitos em 500 épocas. O teste que obteve o resultado de melhor acurácia foi o que utilizou entrada de 300x300 e *batch size* de 10, conforme ilustrado na Figura 19.

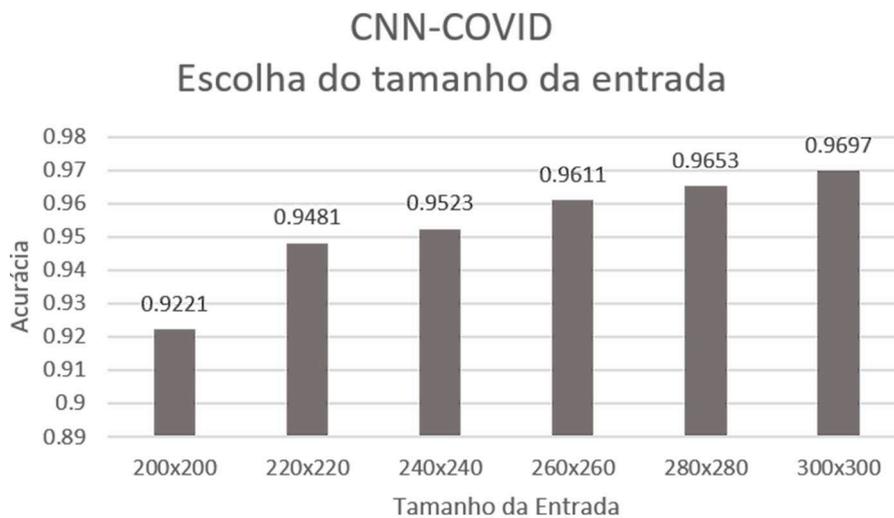


Figura 19 – Escolha do tamanho da entrada da CNN-COVID (200X200 ... 300X300)

#### A. Treinamento, teste e validação da CNN-COVID

Para os conjuntos de dados I e II, as fases de treinamento e teste foram realizadas da seguinte forma:

- Com relação ao conjunto de dados I, a rede foi testada 10 vezes, variando os parâmetros utilizados pela classe `ImageDataGenerator` para gerar novas amostras nas bases de imagens de ambas as classes COVID-19 e NON COVID. Para a fase de validação foi considerada a rede que obteve a melhor acurácia dentre os 10 testes.
- Para o conjunto de dados II, a rede também foi testada 10 vezes, e as imagens do banco de dados de treinamento e teste foram selecionadas aleatoriamente. Para a fase de validação, foi considerada a rede com melhor acurácia na fase de testes.

## 4.4. Resultados

No total, 10 treinamentos e testes foram realizados com o CNN-COVID. Um total de 2.000 épocas foram aplicadas a cada treinamento e teste nos conjuntos de dados I e II. Os resultados são mostrados na Figura 20. A acurácia média dos conjuntos de dados I e II foi de 0,9787 e 0,9839, respectivamente, e a média geral entre os dois conjuntos de dados foi de 0,9813.

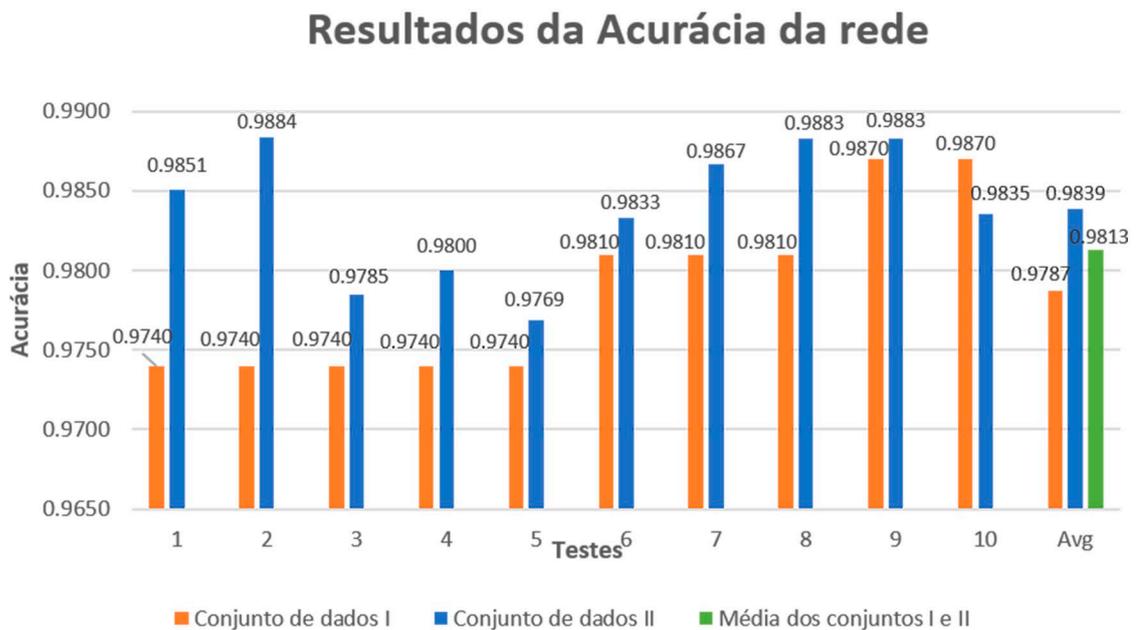


Figura 20 – Conjunto de dados I e II: resultados de cada teste e média geral (Avg) de todos os 10 testes.

Para realizar a fase de validação, foram selecionados os pesos da melhor acurácia dos 10 testes obtidos na CNN-COVID. Assim, a matriz de confusão para o conjunto de dados I foi gerada com 126 imagens COVID-19 e 126 imagens NON-COVID que resultou em 252 imagens para validação do modelo. De forma análoga, a matriz de confusão para o conjunto de dados II foi gerada com 303 imagens COVID-19 e 303 imagens NON-COVID que resultaram em 606 imagens para validação do modelo. A Figura 21 mostra a matriz de confusão gerada para o conjunto de dados I (Figura 21a) e o conjunto de dados II (Figura 21b), respectivamente.

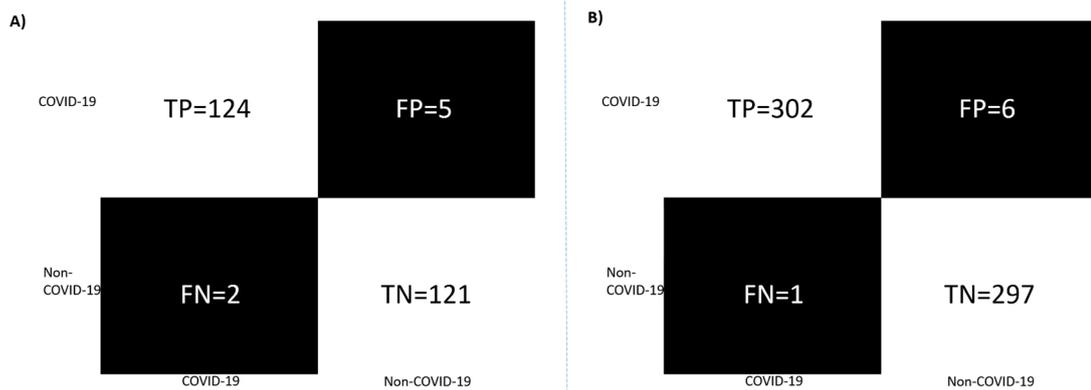


Figura 21 – Matriz de confusão da CNN-COVID em que (A) a é a matriz do conjunto de dados I e (B) é a matriz do conjunto de dados II

Usando os parâmetros TP, TN, FP e FN, a acurácia, a sensibilidade e a especificidade foram calculadas para o conjunto de dados I e o conjunto de dados II, conforme registrado na Tabela 6.

Tabela 6 – Resultados das métricas

Classe	Conjunto de dados	<i>Accuracy</i>	<i>Recall /sensitivity</i>	<i>Specificity</i>
Covid-19	I	0,9722	0,9800	0,9600
Covid-19	II	0,9884	0,9966	0,9801
<b>Média</b>	<b>I e II</b>	<b>0,9803</b>	<b>0,9883</b>	<b>0,9700</b>

De acordo com os resultados apresentados na Tabela 6, a curva ROC foi elaborada para os dois conjuntos de dados como  $(1-SP)$  e  $SE$ , como  $x$  e  $y$ , respectivamente. De acordo com a avaliação ROC, a área (AUC) é mostrada na Figura 22.

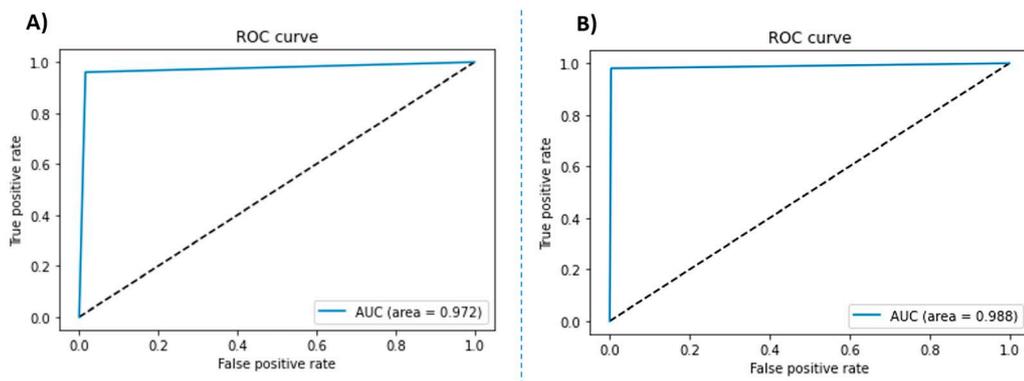


Figura 22 – Curva ROC da CNN-COVID: conjunto de dados I/AUC = 0,972; b conjunto de dados Dataset II/AUC = 0,988.

A partir dos resultados da Tabela 6, calculou-se a média das métricas ACC, Sen e Sp para os dois conjuntos de dados, que foi comparada com os resultados dos métodos de última geração. A CNN-COVID apresentou resultados melhores que os obtidos por outros trabalhos

em relação às métricas Sen e ACC, com valores 98,83% e 98,03%, respectivamente, como exibido na Tabela 7.

Tabela 7 – Comparação da CNN-COVID com o estado da arte

Classe	Método	Accuracy	Sensitivity	Specificity
(Zhang et al. 2020)	Deep Anomaly Detection	0,9518	0,9600	0,7060
(Sethy and Behera 2020)-ResNet50	ResNet50	0,9538	0,9727	0,9347
(Sethy and Behera 2020) -VGG16	VGG16	0,9276	0,9747	0,8805
( WANG, LIN e WONG, 2020)	COVID-NET	0,9333	0,9100	<b>0,9900</b>
(Abbas et al. 2020)	DeTrac	0,9512	0,9791	0,9187
<b>Método proposto (Média dos conjuntos de dados I e II)</b>	<b>CNN-COVID</b>	<b>0,9803</b>	<b>0,9883</b>	0,9700

## 4.5. Discussões

Como as imagens de radiografias de tórax têm tamanhos variados acima de  $1000 \times 1000$  pixels, foi necessário fazer um estudo para qual tamanho as imagens deveriam ser remodeladas de forma a obter a melhor entrada para a rede CNN-COVID. A entrada foi iniciada com  $200 \times 200$ , em seguida foi aumentada de 20 em 20 até a entrada  $300 \times 300$ . Cada entrada foi processada pela CNN-COVID por 500 épocas. Foi observado neste estudo que quanto maior a entrada da rede maior acurácia obtida pela rede, conforme foi mostrado na Figura 19, a qual a entrada  $300 \times 300$  obteve a maior acurácia com o valor de 0,9697.

Para o conjunto de dados I, que apresentava um número reduzido de imagens COVID-19, foi necessário aumentar o banco de dados, para aumentar a generalização do CNN-COVID. O aumento do banco de dados foi feito durante o tempo de execução. Na fase de treinamento, 126 imagens COVID-19 geraram 252.000 novas imagens, e 800 imagens NÃO-COVID geraram 1.600.000 novas imagens, totalizando 1.852.000 imagens. Na fase de testes, 40 imagens COVID-19 geraram 80.000 novas imagens, e 200 imagens NON-COVID geraram 400.000 novas imagens, totalizando 480.000 imagens. Na fase de validação, o gerador de dados não foi utilizado.

Neste trabalho, a CNN-COVID foi treinada e testada 10 vezes, a fim de obter um valor mais confiável para o ACC. A acurácia média dos 10 testes foi de 0,9787 para o conjunto de dados I e 0,9839 para o conjunto de dados II, ilustrada pela Figura 20. Quanto à validação, a acurácia obtida foi de 0,9722 para o conjunto de dados I e 0,9884 para o conjunto de dados II,

como descrito na Tabela 6). Assim, foi constatado que a acurácia da validação do conjunto de dados I ficou próxima da acurácia média dos testes, com percentual de erro menor que 1%.

No conjunto de dados II, a acurácia da validação foi superior à média dos testes, mostrando que a acurácia média é realmente mais confiável. Assim, esses resultados indicam que o investimento de tempo, recursos humanos, financeiros e computacionais na criação e aprimoramento de técnicas baseadas em aprendizado de máquina é uma abordagem promissora para auxiliar os profissionais no prognóstico do novo coronavírus por meio de radiografias de tórax.

## 4.6. Conclusões

Este trabalho propôs uma rede neural profunda, denominada CNNCOVID, para o prognóstico do vírus COVID-19. Dois conjuntos de dados diferentes foram usados: o conjunto de dados I é formado por um repositório de dados de imagem COVID-19 (COHEN, MORRISON e DAO, 2020), bem como o conjunto de dados ChestXray14 (WANG, PENG, *et al.*, 2019), e o conjunto de dados II, por outro lado, é formado pelo repositório BIMCV COVID-19 + (DE LA IGLESIA VAYÁ, SABORIT, *et al.*, 2020). Após a conclusão dos 10 testes, a acurácia média do conjunto de dados I e do conjunto de dados II foi de 0,9787 e 0,9839, respectivamente. Os pesos dos melhores resultados dos testes foram aplicados na validação, obtendo-se valores de acurácia de 0,9722 para o conjunto de dados I e 0,9884 para o conjunto de dados II.

O objetivo desta Parte I era criar um método de classificação utilizando uma rede neural convolucional CNN para classificar imagens de radiografias de tórax. Com os resultados obtidos e mostrados neste capítulo, foi possível validar o novo método CNN, o qual se mostrou uma ferramenta promissora para auxiliar os médicos na classificação das imagens de tórax com pneumonia, considerando a pneumonia atípica por COVID-19 e a pneumonia por outras causas. Espera-se que essa tecnologia aprimore a prestação de serviços de saúde, contribuindo para o prognóstico da doença por meio de exames simples, como a radiografia de tórax, e ampliando o acesso às informações por meio de ferramentas que auxiliem no uso de imagens para o diagnóstico.

## 5. Segunda Contribuição: Por um modelo de classificação usando CNN e Wavelets aplicados a imagens de TC COVID-19

---

“Este capítulo é baseado no artigo completo publicado no periódico Research, Society and Development, Classificação Qualis CAPES 2019 A3, de autoria do próprio autor desta tese. DOI: <http://dx.doi.org/10.33448/rsd-v11i5.27919>”

### 5.1. Introdução

O objetivo deste capítulo é apresentar o desenvolvimento de um novo método para classificação de imagens médicas radiológicas de algumas patologias pulmonares, tais como COVID-19. Para tanto, o resultado da pesquisa bibliográfica relacionada ao tema desta pesquisa é apresentado a seguir.

A pesquisa de Chen *et al.* (CHEN, WU, *et al.*, 2020) foi desenvolvida visando a criação de uma aplicação web baseada num modelo de aprendizado profundo para a detecção de pneumonia por COVID-19 em imagens de Tomografia Computadorizada (TC) de alta resolução do Hospital Renmim da Universidade de Wuhan, província de Hubei, China. A aplicação foi desenvolvida almejando aliviar a pressão de trabalho sobre as equipes do hospital. Para o desenvolvimento e validação do modelo, foram utilizadas 46.096 imagens anônimas de 106 pacientes admitidos no hospital. Os pacientes foram divididos em um grupo de 51 pacientes com diagnóstico de pneumonia por COVID-19 confirmado por exames laboratoriais e um grupo de controle de 55 pacientes diagnosticados com outras doenças pulmonares. Para avaliar e comparar a eficiência do diagnóstico dos especialistas com a do modelo, foram coletados dados retrospectivos de 27 pacientes submetidos ao exame de CT em 05/02/2020 no mesmo hospital. O modelo atingiu acurácia de 95,24%, sensibilidade por paciente de 100% e especificidade de 93,55%. No conjunto de dados dos 27 pacientes atendidos previamente, o modelo chegou à acurácia de 98,85%, sensibilidade por imagem de 94,34% e especificidade de 99,16%. Os autores, assim, mostraram que os resultados do experimento são comparáveis ao desempenho de especialistas médicos.

O trabalho de WU *et al.* (WU, HUI, *et al.*, 2020) desenvolveu um método baseado em redes neurais profundas para identificar com rapidez e precisão pacientes com COVID-19 usando imagens de TC. Para isso foram coletadas, retrospectivamente, imagens de TC de tórax

de 495 pacientes de três hospitais da China. 495 conjuntos de dados foram divididos aleatoriamente em 395 casos (294 do COVID-19, 101 de outras pneumonias) do conjunto de treinamento, 50 casos (37 do COVID-19, 13 de outras pneumonias) do conjunto de validação e 50 casos (37 de COVID-19, 13 de outra pneumonia) do conjunto de testes. O modelo de fusão de múltiplas visualizações foi treinado, usando uma rede de aprendizado profundo para rastrear pacientes com COVID-19 usando imagens de TC com as regiões pulmonares máximas nas vistas axial, coronal e sagital. O desempenho do modelo proposto foi avaliado pelos conjuntos de validação e teste. O modelo de fusão de aprendizado profundo com múltiplas visualizações alcançou a área sob a curva de características operacionais do receptor (AUC) de 0,732, acurácia de 0,700, sensibilidade de 0,730 e especificidade de 0,615 no conjunto de validação. No conjunto de testes, foi obtido AUC, acurácia, sensibilidade e especificidade de 0,819, 0,760, 0,811 e 0,615, respectivamente. Os autores concluem que, com base no método de aprendizado profundo, o modelo de diagnóstico proposto, treinado em imagens de múltiplas imagens de imagens de TC de tórax, mostrou grande potencial para melhorar a eficácia do diagnóstico e mitigar a pesada carga de trabalho dos radiologistas na triagem inicial da pneumonia por COVID-19.

No trabalho Yang *et al.* (YANG, JIANG, *et al.*, 2020) foi utilizado a rede DenseNet para a detecção de características do COVID-19 na Tomografia Computadorizada de alta resolução (TCAR). Duzentos e noventa e cinco pacientes foram incluídos neste estudo (pessoa saudável: 149; pacientes COVID-19: 146), que foram divididos em três cortes não sobrepostas separadas (conjunto de treinamento, 135 (total): 69 (pessoa saudável), 66 (pacientes COVID-19); conjunto de validação 20 (total): 10 (pessoa saudável), 10 (pacientes COVID-19); conjunto de teste 140 (total): 70 (pessoa saudável), 70 (pacientes COVID-19). O modelo do algoritmo DenseNet produziu uma AUC de 0,98 no conjunto de testes. O valor do limiar foi selecionado como 0,8, enquanto para o conjunto de validação, a acurácia foi de 92%, a sensibilidade foi de 97% e a especificidade foi de 87%.

No trabalho Özkaya *et al.* (ÖZKAYA, ÖZTÜRK e BARSTUGAN, 2020) um novo método foi proposto como fusão e *ranking deep features* para detectar COVID-19 na fase inicial. Cortes 16x16 (Subconjunto-1) e 32x32 (Subconjunto-2) foram obtidos de 150 Imagens de CT para gerar subconjuntos de dados. Dentro do escopo do método proposto, 3000 cortes de imagens de foram rotuladas como COVID-19 e No finding, para o uso nas fases de treinamento e teste. O método de fusão e *ranking* foi utilizado para incrementar a performance do modelo proposto. De acordo com outros modelos de rede neural convolucional (CNN) pré-treinados

usados no *transfer learning*, o método proposto mostra uma alta performance no Subset-2 com métricas de 98,27% de acurácia, 98,93% sensibilidade, 97,60% especificidade.

Já no trabalho de Barstugan *et al.* (BARSTUGAN, OZKAYA e OZTURK, 2020) o processo de detecção foi implementado em imagens de Tomografia Computadorizada (TC) abdominal. Para a detecção do COVID-19, quatro conjuntos de dados diferentes foram formados através de amostras de tamanhos  $16 \times 16$ ,  $32 \times 32$ ,  $48 \times 48$ ,  $64 \times 64$  de 150 imagens de TC. O processo de extração de recursos foi aplicado a *patches* para aumentar o desempenho da classificação. Os algoritmos *Gray Level Co-occurrence Matrix* (GLCM), *Local Directional Pattern* (LDP), *Gray Level Run Length Matrix* (GLRLM), *Gray-Level Size Zone Matrix* (GLSZM), and *Discrete Wavelet Transform* (DWT) foram usados como métodos para extração de características e o algoritmo *Support Vector Machines* (SVM) foi usado para classificar essas características. *2-fold*, *5-fold* e *10-fold cross-validations* foram implementadas durante o processo de classificação. Métricas de sensibilidade, especificidade e acurácia foram usadas para avaliar o desempenho da classificação. O melhor valor de acurácia de classificação obtido foi 99.68% com 10-fold cross-validation e o método de extração de características GLSZM.

No trabalho Wang *et al.* (WANG(A), DENG, *et al.*, 2020) foi feito o desenvolvimento de um modelo baseado em aprendizado profundo para o diagnóstico automático de COVID-19 na TC do tórax. Uma estrutura de aprendizado profundo fracamente supervisionada foi desenvolvida usando volumes de TC 3D para classificação COVID-19 e localização de lesões. Para cada paciente, a região pulmonar foi segmentada usando um UNet pré-treinado. A região pulmonar 3D segmentada foi inserida em uma rede neural profunda 3D para prever a probabilidade de infecção por COVID-19; as lesões COVID19 são localizadas combinando as regiões de ativação na rede de classificação e os componentes conectados não supervisionados. Foram utilizados 499 volumes de CT para treinamento e 131 volumes de CT para teste. O algoritmo obteve 0,959 ROC AUC e 0,976 PR AUC. Ao usar um limiar de probabilidade de 0,5 para classificar COVID-positivo e COVID-negativo, o algoritmo obteve uma precisão de 0,901, um valor preditivo positivo de 0,840 e um valor preditivo negativo muito alto de 0,982.

Uma vez estabelecido o objetivo do capítulo e apresentado a relação de trabalhos relacionados, os próximos passos da pesquisa serão abordados subseções a seguir; sendo elas Metodologia, Modelo, Resultados e Discussão.

## 5.2. Estrutura da WCNN-COVID

A estrutura desenvolvida para o modelo WCNN-COVID também é responsável pelo pré-processamento da imagem de entrada da CNN utilizando decomposição de imagens com

transformada *wavelet*. Nesta seção, além dos tópicos mencionados, são abordados, também, aspectos teste e análise da complexidade do algoritmo.

### 5.2.1. Decomposição da Wavelet

A *wavelet* mãe foi utilizada nos trabalhos (DA COSTA JUNIOR e PATROCINIO, 2019, COSTA, PEREIRA, *et al.*, 2020) para redução de ruídos em imagens de mamografia de mamas densas, onde foram testadas as 7 famílias *Daubechies*, *Symlets*, *Coiflets*, *Fejer-Korovkin* e *dMeyer*. A pesquisa mostrou que a família *Coiflets 5* apresentou o resultado relevante para a redução de ruídos em mamografia. Por isso, a função *Coiflets 5* foi o ponto de partida da investigação dos resultados da aplicação da família *coiflets* em imagens de tórax; e a transformada *wavelet* foi utilizada para decomposição multiescala das imagens, de modo que as diferentes escalas pudessem ser manipuladas e usadas como entrada para a WCNN-COVID. Os coeficientes de aproximação, horizontal, vertical e diagonal desta família *coiflets* podem ser usados para inferir mais informações estruturais sobre a imagem. Vale ressaltar que, para a entrada da WCNN-COVID, foram selecionados três coeficientes: aproximação, vertical e diagonal.

Para avaliar a significância dos coeficientes da transformada *wavelet* e escolher três dos quatros tipos de coeficientes resultantes da decomposição, foi feita uma análise estatística que utilizou uma amostra de 500 imagens da base COVID e 500 da base NON-COVID. Foi utilizado o teste de Fiedman para análise de significância, por se tratar de um conjunto de dados heterogêneo, independente e não paramétrico (WILLIAMS e LI, 2016, ZIMMERMAN e ZUMBO, 1993). Esse teste foi feito para averiguar se as classes apresentam valores estatisticamente similares entre si, com relação a cada coeficiente resultante da decomposição por *Wavelet* (coeficientes de aproximação, horizontal, vertical e diagonal das respectivas imagens). Foi utilizado o *software* de análise estatística BioEstat, na versão 5.3, para entrada de dados e testes de hipótese e significância.

Todos os testes consideraram  $\alpha$  de 0,05. Para a análise de Friedman, o desvio padrão obtido da decomposição *wavelet* (desvio padrão dos coeficientes de aproximação, horizontal vertical e diagonal) destacou que todas as combinações de desvio padrão dos coeficientes *wavelet* das imagens da base COVID comparados com as das imagens da base NON-COVID apresentam diferenças estatísticas significativas, exceto para o coeficiente diagonal. Assim, foi utilizado como entrada da WCNN-COVID os 3 coeficientes (aproximação, vertical e diagonal), que apresentaram diferenças intergrupos estatisticamente mais significativa. Os histogramas da Figura 23 ilustram a distribuição esparsa de todas as aproximações dos coeficientes, horizontal e vertical.

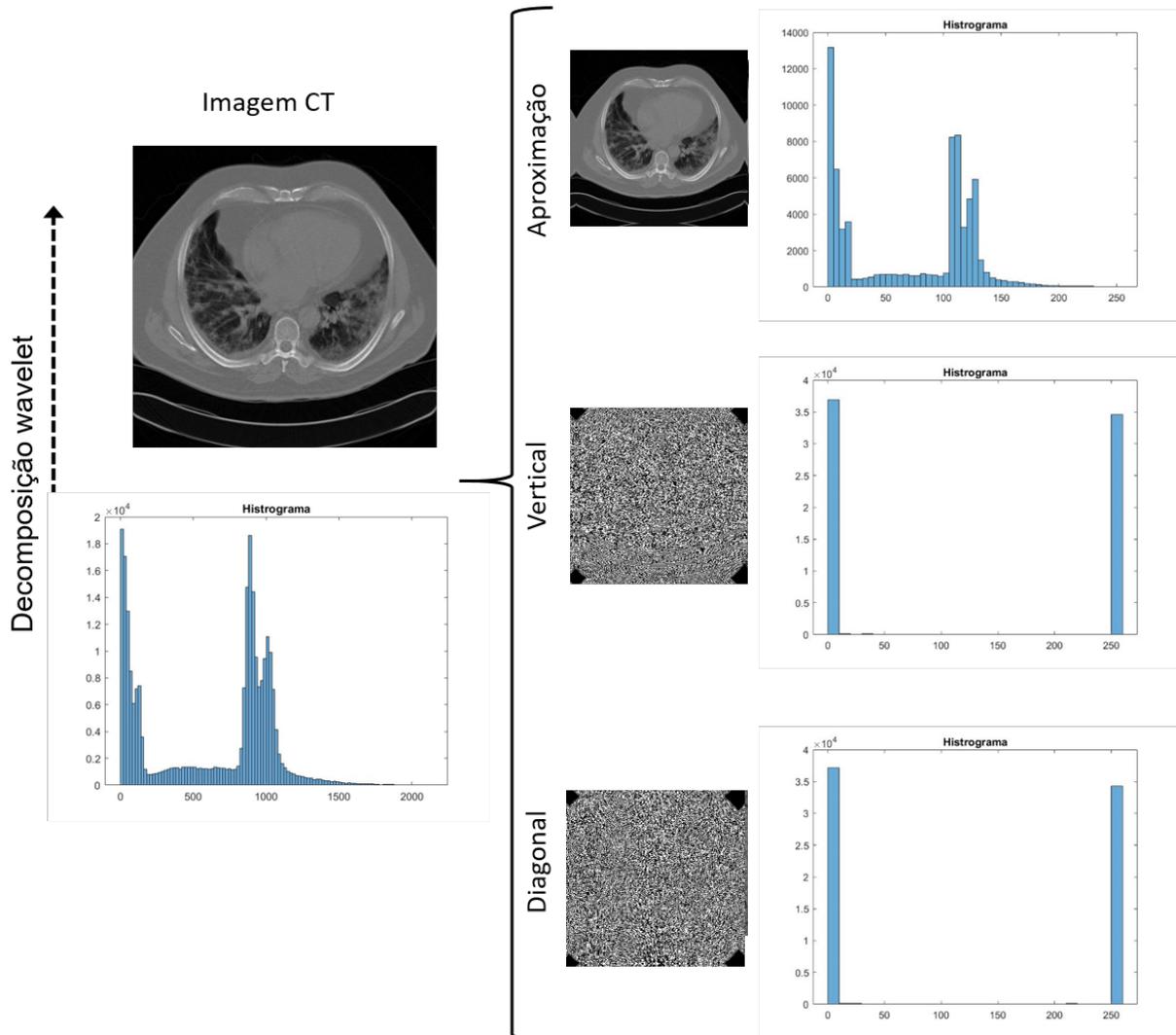


Figura 23 – Histograma das Imagens Chest X-ray, componentes aproximação, vertical e diagonal.

Imagens digitais são compostas de acordo com o espaço *Red, Green, Blue* (RGB), o qual utiliza três componentes distintos; sendo um para cada componente básica da luz: R para vermelho, G para verde e B azul. No presente método, para formar uma nova imagem que será a entrada a ser processada pela WCNN-COVID, substituem-se estes componentes pelos componentes aproximação, vertical e diagonal, respectivamente. A Figura 24 apresenta a formação da entrada utilizando esta abordagem. A imagem TC é decomposta em 3 componentes aproximação, vertical e diagonal (gerados pela *Wavelet* mãe Coiflet 5), os quais serão utilizados para formar a imagem digital que será processada. Cada canal será processado pelas camadas convolucionais e seus respectivos filtros.

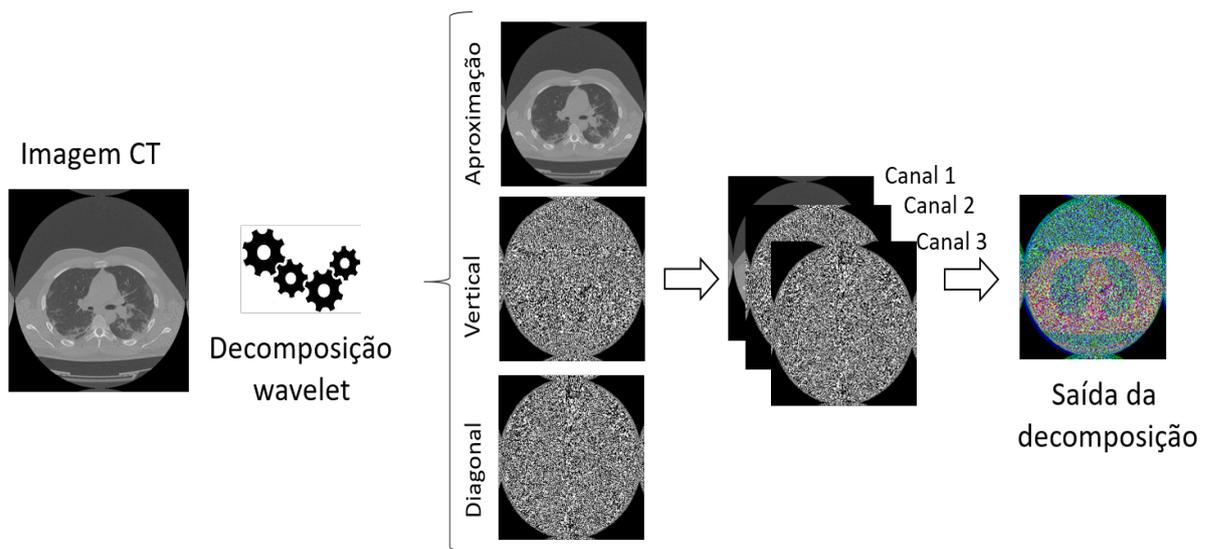


Figura 24 – Processo de decomposição da *Wavelet*.

### 5.2.2. Modelo WCNN-COVID

A estrutura da WCNN-COVID é similar a da CNN-COVID discutida no capítulo quatro, com a diferença na entrada da rede: dimensão da entrada da rede passa a ser  $300 \times 300 \times 3$ . A arquitetura detalhada da rede WCNN-COVID para classificação do COVID-19 é mostrada na Tabela 8. A rede é composta por camadas convencionais, incluindo camada de entrada, camada de convolução, camada de *max-pooling* e camadas totalmente conectadas, como ilustrado na Figura 25.

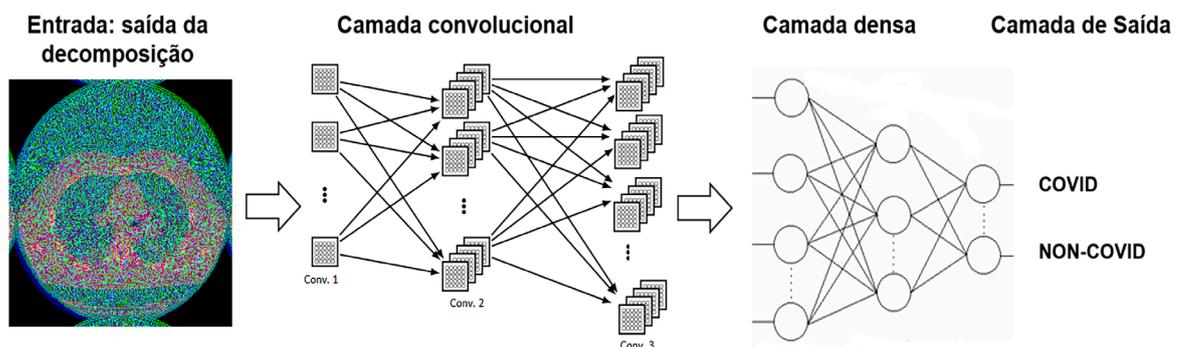


Figura 25 – Esquema de classificação da rede neural profunda

Além disso, uma função de ativação da unidade linear retificada (ReLU) é usada após cada camada de convolução (1ª, 3ª, 5ª e 7ª) e nas camadas densas (9ª, 10ª, 11ª, e 12ª). Para diminuir a possibilidade do *overfitting*, o *dropout* é implementado nas quatro primeiras camadas totalmente conectadas (9ª, 10ª, 11ª e 12ª camadas) com uma porcentagem de 20%.

Tabela 8 Arquitetura da WCNN-COVID. A rede contém a entrada (I), Convolução (C), *Max-Pooling* (M), Rede Totalmente Conectada (F)

Camada	WCNN-COVID		
	Tipo	Dimensões do filtro	Dimensões de Entrada/Saída
0	I		300×300×3
1	C	5×5×256	296×296×256
2	M	2×2	148×148×256
3	C	3×3×128	146×146×128
4	M	2×2	73×73×128
5	C	3×3×64	71×71×64
6	M	2×2	35×35×64
7	C	3×3×32	33×33×32×
8	M	2×2	16×16×32
9	F	16×16×32×256	1×256
10	F	1×1×256×128	1×128
11	F	1×1×128×64	1×64
12	F	1×1×64×32	1×32
13	F	1×132×2	1×2

Para escolher os valores dos parâmetros de entrada e o *batch-size* da WCNN-COVID, foram testadas diversas combinações de entrada. Estes testes foram realizados no repositório do Hospital São Lucas da PUCRS, Porto Alegre, RS. Foi utilizando um *notebook* com configuração I7-8750Hz Intel processor, 2,21GHz CPU, 16.0 GB RAM, placa de vídeo GeForce GTX 1060 com Max-Q Design.

Os testes para definição dos parâmetros levaram em consideração a capacidade de execução do *hardware* disponível. Para as entradas 200×200 e 220×220 o *batch-size* foi 20. Já para entradas de tamanho maior, o *bach-size* foi 10. Os testes das entradas foram feitos em 100 épocas. O teste que obteve o resultado de melhor acurácia foi o que utilizou entrada de 300 × 300 e *batch-size* de dez, conforme ilustrado na Figura 26.

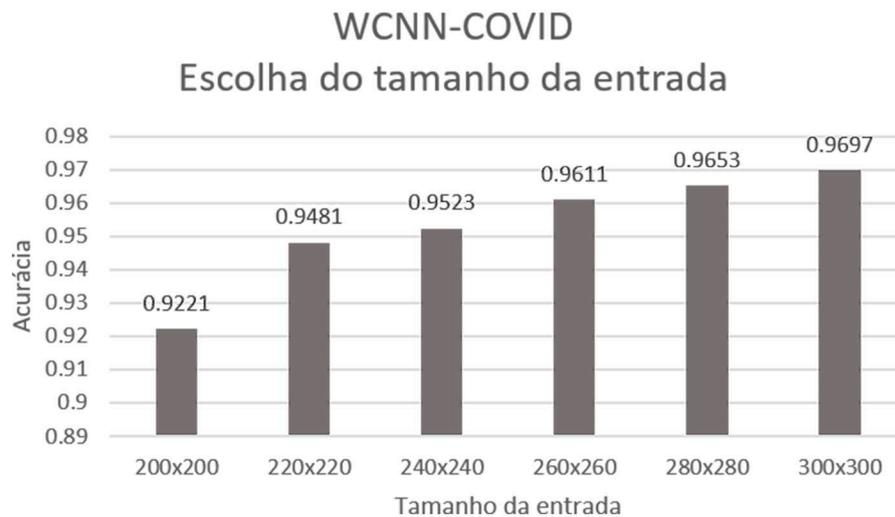


Figura 26 – Escolha do tamanho da Entrada da WCNN-COVID (200 × 200 ... 300 × 300).

Assim a imagem foi redimensionada para o tamanho 300×300. Para o primeiro treinamento os pesos são inicializados de forma aleatória. Os mesmos parâmetros da CNN-COVID, apresentados no Capítulo oito, foram utilizados na WCNN-COVID: A rede foi treinada de acordo com o modelo ADAM (WANI, BHAT, *et al.*, 2020). Foram utilizados parâmetros padrão  $\beta_1 = 0,9$  e  $\beta_2 = 0,999$  (KINGMA e BA, 2014), taxa de aprendizado inicial  $\alpha = 0,001$ , decaída por um fator de 10.

### 5.2.3. Treinamento, teste e validação da WCNN-COVID

O treinamento de redes neurais no domínio *wavelet* pode fazer com que os procedimentos adotados de treino e teste, sejam executados mais rápido. O uso de coeficientes *wavelet* incentiva a dispersão de ativação em camadas ocultas e na camada de saída. Além disso, ao usar os coeficientes, estes ficam mais esparsos. Isso faz com que seja mais fácil para a rede aprender mapas esparsos em vez de mapas densos. Este alto nível de esparsidade reduz ainda mais o tempo de treinamento necessário para a rede, calculando resultados de classificação mais precisos (GUO, SEYED MOUSAVI, *et al.*, 2017). Isso pode ser justificado pela complexidade do algoritmo decrito na seção anterior.

O método foi submetido a dez ciclos de treinamento e teste. Em cada ciclo, as bases de treinamento e teste foram criadas, selecionando imagens COVID e NON-COVID, aleatoriamente, a partir das bases disponíveis e respeitando a proporção de 70% das imagens para treinamento, 15% para teste e 15% para validação. Na etapa de validação, foram os considerados os pesos obtidos pela realização do décimo ciclo de treinamento e teste, conforme mostra a Figura 27.

Conforme a realização dos ciclos avança, os pesos da menor taxa de erro da rede, ou o maior valor da acurácia, no treinamento, são utilizados na inicialização da rede no ciclo subsequente. Por exemplo, os pesos do primeiro ciclo foram utilizados para inicialização da rede no segundo ciclo, e assim sucessivamente. Esta técnica contribuiu para o aumento da acurácia do modelo. A Figura 27 ilustra a estratégia de repasse de pesos entre os ciclos.

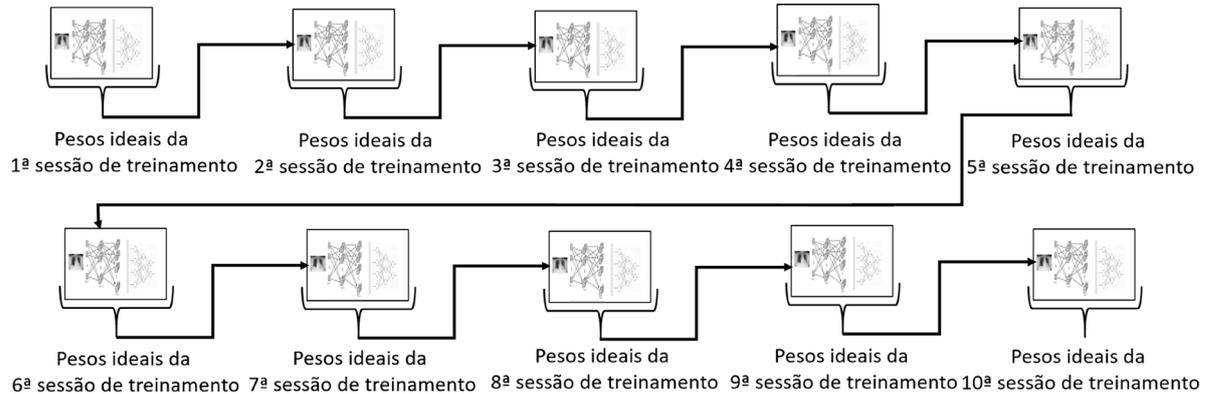


Figura 27 – Encadeamento dos melhores pesos da WCNN-COVID.

#### 5.2.4. Complexidade do Algoritmo

Como sugerido em He e Sun (HE e SUN, 2015), as camadas convolucionais geralmente custam de 90 a 95% do custo computacional. A complexidade computacional das camadas convolucionais foi analisada. Para uma dada CNN, a complexidade total de todas as camadas convolucionais pode ser expressa como:

$$O\left(\sum_{l=1}^d n_{l-1} \cdot s_l^2 \cdot n_l \cdot m_l^2\right) \quad (1)$$

Aqui,  $l$  é o índice de uma camada convolucional e  $d$  é a profundidade (número de camadas convolucionais); e,  $n_l$  é o número de filtros (também conhecido como *largura*) na  $l$ -ésima camada. Sendo que  $n_{l-1}$  é o número de canais de entrada da  $l$ -ésima camada;  $S_l$  é o tamanho espacial (comprimento) do filtro e  $m_l$  é o tamanho espacial do mapa de recursos de saída (*output feature map*).

WCNN-COVID tem, como entrada, a imagem decomposta pela transformada *wavelet*, ou seja, a imagem da entrada da rede tem a metade da resolução da imagem original. Sendo assim, a  $m_l$  também é reduzido à metade. Portanto, em termos espaciais, a rede WCNN-COVID é 50% menos complexa que rede CNN convencional, que utiliza a imagem original, em que a transformada *wavelet* não foi aplicada.

### 5.3. Workflow de Integração da WCNN-COVID

Esta seção apresenta os resultados obtidos pela aplicação do novo método de classificação descrito na seção 5.2. Aqui serão descritas, seguindo os passos do *workflow*, todas as macro-tarefas do método para evidenciar a melhoria significativa obtida na acurácia de classificação das imagens tomográficas de tórax, das bases COVID e NON COVID.

O *workflow* de integração utilizado neste artigo é apresentado na Figura 28. Ele é composto pelas fases **Criação da Base de Imagens, Pré-Processamento de Imagens, Arquitetura da WCNN-COVID e Resultados**.

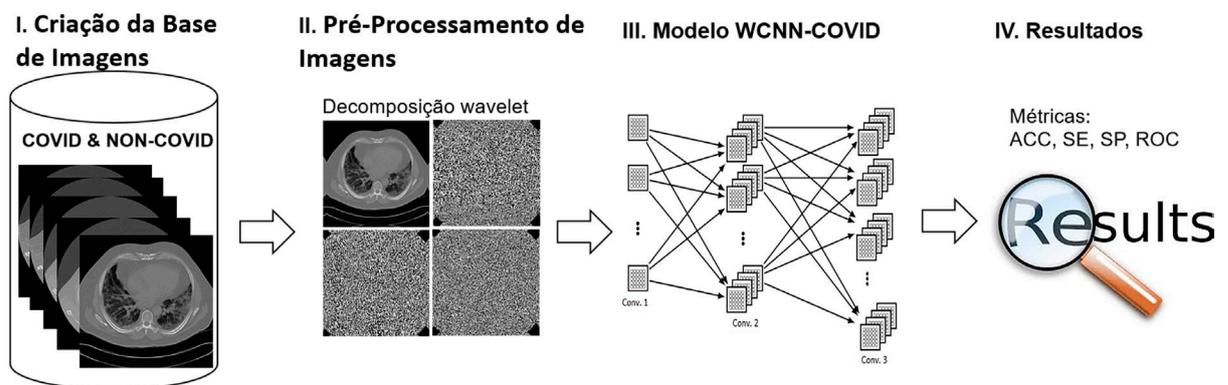


Figura 28 – Diagrama do método proposto para classificação de COVID-19.

#### 5.3.1.1. Criação da base de imagens

A base de imagens usada para treinar e avaliar o método WCNN-COVID é composta por 25.534 imagens de Tomografia Computadorizada (CT), divididas em duas sub-bases: i) COVID, composto por 12.767 imagens testadas positivamente para COVID-19; e, ii) NON-COVID, constituído por 12.767 imagens testadas negativamente. Estas imagens foram obtidas a partir de cinco repositórios:

- Conjunto de dados *COVID-19 CT scans (KAGGLE)* (SETHY e BEHERA, 2020).
- Conjunto de dados *MosMedData: COVID19\_1000 Dataset* (ABBAS, ABDELSAMEA e GABER, 2020).
- Conjunto de dados *NIH Clinical Center: dataset of 32,000 CT images* (SUMMERS, 2020).
- Conjunto de dados *UESTC-COVID-19* (WANG(C), LIU, *et al.*, 2020).
- Conjunto de dados (privado) COVID-19 formado por imagens de CT providas do Hospital São Lucas da PUCRS (HSL-PUCRS).

Das 12.767 imagens da sub-base COVID, 70% delas foram utilizadas na fase de treinamento, 15% na fase de teste e os demais 15%, usados na fase de validação. A mesma distribuição foi feita na sub-base NON-COVID, conforme Tabela 9.

Tabela 9 – Distribuição da Base de dados utilizada na WCNN-COVID

Base	Total	Treinamento	Teste	Validação
COVID	12767	8937	1915	1915
NON-COVID	12767	8937	1915	1915

### 5.3.1.2. Pré-processamento da imagem

Foi utilizada a transformada *wavelet* para decomposição multiescala das imagens, de maneira que as diferentes escalas pudessem ser manipuladas e inseridas como entrada para a WCNN-COVID, conforme detalhado na subseção 5.2.1.

### 5.3.1.3. Teste de Execução

A WCNN-COVID foi submetida a 10 ciclos de treinamento e teste. Em cada ciclo, as bases de treinamento e teste foram criadas, selecionando, aleatoriamente, imagens COVID e NON-COVID. Em cada execução, foram respeitados os critérios detalhados na subseção 5.2.2 e executados conforme restrições documentadas na subseção 5.2.3.

### 5.3.1.4. Resultados

Por meio da evolução de pesos da WCNN-COVID, destacado na seção anterior, foi escolhido a WCNN-COVID do último treinamento para ser aplicado na fase da validação. Assim, a matriz de confusão foi gerada com 1.915 imagens da classe COVID e 1915 da classe NON-COVID. Isso resultou em 3.830 imagens para validação da WCNN-COVID. Desta forma, de acordo com a matriz de confusão da Figura 29, tem-se *True Positive* = 1899, *True Negative* = 1913, *False Positive* = 2 e *False Negative* = 16.

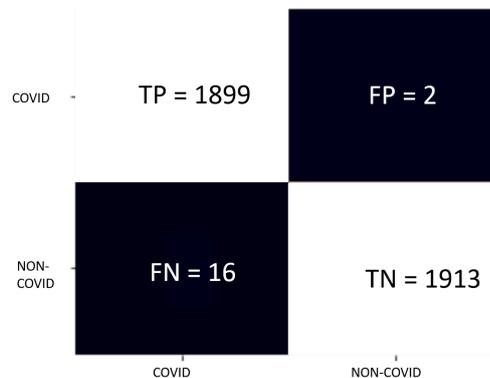


Figura 29 – Matriz de confusão da WCNN-COVID.

Com os parâmetros TP, TN, FP, FN foram calculadas as seguintes métricas Acurácia, Sensitividade e Especificidade, mostradas na Tabela 10.

Tabela 10 – Resultado das métricas

Class	Acurácia	Sensitivite	Especificidade
Covid-19	0,9953	0,9916	0,9989

Com objetivo de avaliar o desempenho do novo método, foram treinadas redes utilizando os bancos de imagens, COVID e NON-COVID, com e sem pré-processamento. Foram construídas curvas ROC para os dois testes, o primeiro não utilizou a metodologia descrita de pré-processamento com decomposição de imagens usando *wavelet* (Figura 30(A)), já o segundo, para todas as imagens, seguiu-se todas as etapas descritas de pre-processamento para entrada da rede (Figura 30 (B)).

As curvas ROC A e B foram calculadas na etapa de validação do décimo ciclo de treinamento e teste. Assim, com base na análise das curvas ROC, a área da curva sem pré-processamento é 0.9229 e a área da curva com pré-processamento é 0.9950, conforme ilustrado na Figura 30.

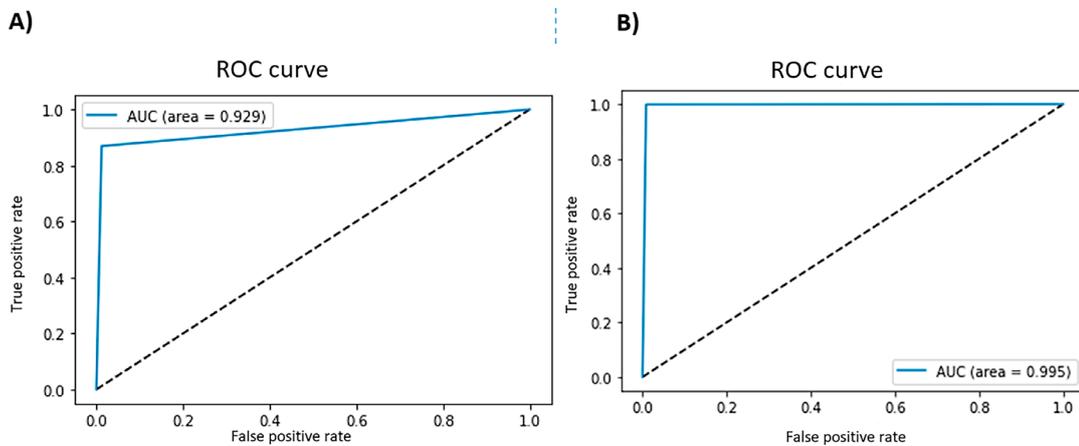


Figura 30 – Curvas ROC do WCNN-COVID: A) Conjunto de imagens sem wavelet/AUC = 0.9229 B) Conjunto de imagens com wavelet /AUC=0.995.

Em cada um dos 10 ciclos de treinamento e teste, foram aplicadas 1000 épocas; gerou os resultados descritos a seguir. A menor acurácia foi obtida no ciclo 1 e foi de 0.9830; já a maior foi obtida no ciclo 8, com valor de 0.9987. A média geral da acurácia dos 10 ciclos foi de 0,9953 e desvio padrão de 0,005284, conforme Figura 31.

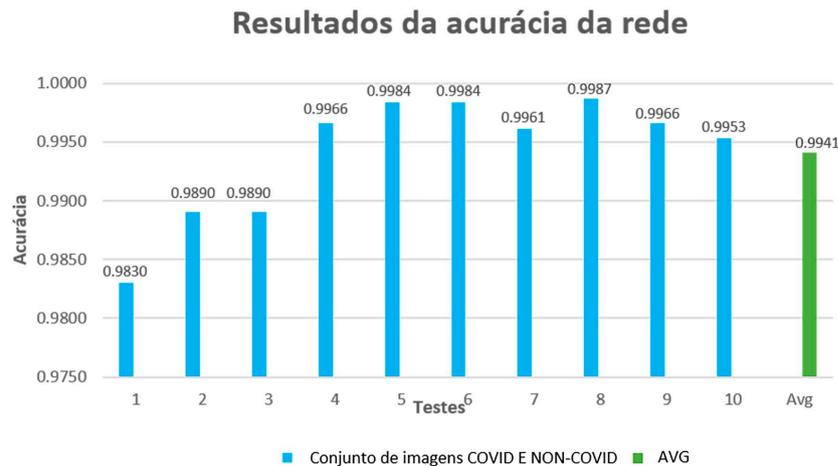


Figura 31 – Resultado da acurácia da WCNN-COVID: resultados para cada ciclo e média geral para os 10 ciclo.

Também foi confirmado que as sub-bases de imagens COVID e NON-COVID segue a distribuição normal. A variável contínua ( $x$ ) representa o sucesso ou fracasso ao classificar as imagens e a função de densidade de probabilidade é denotada pela Equação 41, como segue (PINHEIRO, CUNHA, *et al.*, 2009):

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} \quad \text{Equação 41}$$

A distribuição normal é descrita pelos seguintes parâmetros: média ( $\mu$ ), desvio padrão ( $\sigma$ ); o valor de ( $x$ )  $\in (-\infty, \infty)$ , assim como a curva que é representada pela Figura 32 (PINHEIRO, CUNHA, *et al.*, 2009).

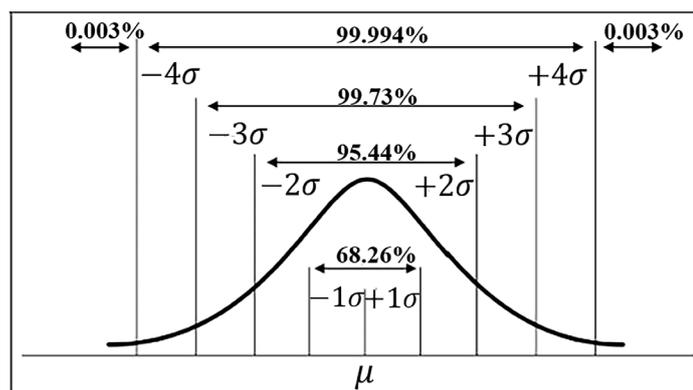


Figura 32 – Função Probabilidade Normal

Para calcular a probabilidade  $P(x)$ , da classificação da imagem ser realizada com sucesso, calcula-se a área da distribuição normal padronizada, entre  $\mu = 0$  e  $\sigma$  (PINHEIRO, CUNHA, *et al.*, 2009). Dado que  $\sigma \cong 0.005284$ ,  $P(x)$  é calculado conforme a Equação 42.

$$P(x) = \int_a^b \frac{1}{\sqrt{2\pi\sigma^2}} \cdot e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2} dx \quad \text{Equação 42}$$

Resolvendo a Equação 42, obtem-se  $(x) \cong \int_0^{0,005284} \frac{1}{\sqrt{2\pi}} \cdot e^{-\frac{1}{2}x^2} dx \cong 0,00210$ . Ou seja, há 0,210% de chance de a imagem ser classificada incorretamente, o que denota alta confiabilidade do método WCNN-COVID.

O resultado da validação do décimo ciclo de treinamento e teste, apresentado na Tabela 10, foi comparado com o estado da arte dos métodos apresentados. O melhor resultado encontrado de acurácia, de outros métodos, foi 99,53%, conforme Tabela 11.

Tabela 11: Comparação dos valores das métricas do método WCNN-COVID com o estado da arte dos métodos de classificação considerando imagens de tomográfica computadorizada de tórax.

Método	Tipo	ACC	Sen	Sp
WU, HUI, <i>et al.</i> , 2020	CT	0,7600	0,8110	0,6150
WANG(A), DENG, <i>et al.</i> , 2020	CT	0,9010	0,8400	0,9820
YANG, JIANG, <i>et al.</i> , 2020	CT	0,9200	0,9700	0,8700
CHEN, WU, <i>et al.</i> , 2020	CT	0,9524	1,000	0,9355
ÖZKAYA, ÖZTÜRK e BARSTUGAN, 2020	CT	0,9827	0,9893	0,97,6
BARSTUGAN, OZKAYA e OZTURK, 2020	CT	0,9968	0,9772	0,9967
<b>Método WCNN-COVID</b>	CT	0,9953	0,9916	0,9989

### 5.3.2. Discussão dos resultados

Os resultados encontrados durante a realização do experimento indicam que o método WCNN-COVID possui características e comportamentos relevantes para a área de processamento digital de imagens médicas, incluindo: *i*) previsibilidade e confiança; *ii*) acurácia inédita dentre modelos de classificação que usam redes neurais convolucionais; e, *iii*) convergência do resultado com poucos testes.

No que se refere ao WCNN-COVID, é importante escolher adequadamente os parâmetros que a rede utilizará. O algoritmo para otimização baseado em gradiente de funções objetivas estocásticas escolhidas foi o *Adaptive Moment Estimation* (ADAM), porque ele é uma técnica de otimização adaptativa que tira vantagens do AdaGrad (a capacidade de AdaGrad de lidar com gradientes esparsos) e RMSProp (a capacidade de RMSProp para lidar com objetivos não estacionários) (WANI, BHAT, *et al.*, 2020). Além disso, o método é simples de implementar e requer pouca memória.

O tamanho padrão da imagem gerada por equipamentos de TC é  $512 \times 512$  pixels. Por isso, foi preciso a condução de um estudo que identificasse o tamanho para o qual as imagens originais de TC deveriam ser convertidas para se adequarem aos parâmetros de entrada da WCNN-COVID. Este estudo mostrou que quanto maior o tamanho da entrada, maior era a acurácia da rede. Conforme mostrado na Figura 26, quando o tamanho da entrada era  $300 \times 300$ , a precisão era 0,7203.

O método que utiliza o pré-processamento das imagens propiciou rápida convergência dos pesos da rede. Essa convergência indica que a decomposição *wavelet*, na fase de pré-processamento, contribuiu com a obtenção dos melhores pesos durante as fases de treinamento e teste do modelo. A execução dos 10 ciclos de treinamento e teste também contribuíram para avaliar a estabilidade dos valores da acurácia do modelo.

O valor médio da acurácia dos 10 ciclos de treinamento e teste foi de 0,9941. Além disso, vale ressaltar que a acurácia de todos os testes ultrapassou 0,98. Os pesos utilizados pelo WCNN-COVID na 10ª sessão de treinamento foram utilizados na etapa de validação, e o valor do ACC foi de 0,9953.

A entrada da WCNN-COVID foi constituída por sub-imagens geradas a partir dos coeficientes de decomposição *wavelet* que são metade do tamanho da resolução da imagem original. Assim, o tamanho espacial do mapa de recursos de saída também é reduzido pela metade, o que acelera o processo de classificação. Além disso, o uso de coeficientes *wavelet* incentiva a dispersão de ativação em camadas ocultas, assim como na camada de saída. Além disso, ao usar resíduos, os próprios coeficientes de *wavelet* se tornam mais esparsos e, portanto, mais fáceis para a rede aprender mapas esparsos em vez de densos, acelerando assim o processo de classificação WCNN-COVID.

O uso de transformadas *wavelet* nas imagens de entrada da rede contribui tanto para o desempenho do treinamento quanto para a acurácia da WCNN-COVID (Figura 30a e Figura 30b). A Figura 30a mostra que a área da curva ROC para os testes, com imagens sem pré-processamento, é 0,9229; ao passo que a Figura 30b mostra que área da curva ROC utilizando pré-processamento é 0,9950, o que representa um aumento de 7,8%.

### 5.3.3. Conclusão dos Resultados

Este trabalho propôs e validou um novo método de classificação automatizado denominado WCNN-COVID. Baseia-se em uma rede neural profunda que usa transformada *wavelet* para extrair recursos de classificação de imagens de pacientes com COVID-19, que já apresentam alterações pulmonares (ou pneumonia).

Os resultados do WCNN-COVID mostraram que, a análise de imagens de TC, usando métodos de aprendizado profundo, proporcionou resultados rápidos e precisos na classificação de pneumonia por COVID-19. O método proposto foi utilizado para classificar um conjunto de imagens reais selecionadas de repositórios públicos e privados. Os resultados foram calculados separadamente para as etapas de teste e validação. Um total de 10 ciclos de treinamento e teste foram realizados.

Os resultados obtidos com este trabalho, até o momento, mostram que a associação de CNNs e *wavelets* é promissora para a criação de métodos de classificação de alta precisão. Estudos futuros nesta área envolvem a realização de pesquisas sobre aprimoramento da rede neural e teste do método com outras modalidades de imagens médicas.

## 6. Estudo de Ablação

---

Em inteligência artificial (IA), particularmente em aprendizagem de máquina, ablação é a remoção de um componente de um sistema de IA. Um estudo de ablação investiga o desempenho do sistema de IA removendo certos componentes para entender a contribuição do mesmo para a totalidade do sistema (MEYES, LU, *et al.*, 2019). O termo *ablação* é uma analogia com a biologia, pois consiste na remoção de componentes de um organismo para averiguar como o indivíduo passa a se comportar (MEYES, LU, *et al.*, 2019).

Ablação é particularmente usada na análise de redes neurais artificiais por analogia com cirurgia cerebral ablativa, na qual segmentos do cérebro do indivíduo são gradualmente seccionados. Este tipo de cirurgia permitiu aos cientistas descobrirem quais áreas cerebrais eram responsáveis pela fala, visão, equilíbrio, memória (MEYES, LU, *et al.*, 2019).

Os estudos de ablação – seja na biologia ou na IA – requerem que um sistema exiba degradação graciosa: o sistema deve continuar a funcionar mesmo quando certos componentes estão ausentes ou degradados. Considerando o crescimento das Redes Neurais Artificiais (RNAs) em tamanho e complexidade, bem como o correspondente crescimento na complexidade dos problemas submetidos a elas, a realização de estudos de ablação teve início neste ambiente. Dessa forma, desde os primórdios da IA, estudo de ablação tem defensores, os quais consideram que os estudos de ablação são uma técnica conveniente na investigação de *constructos* de inteligência artificial, sua durabilidade e resistência a danos estruturais (MEYES, LU, *et al.*, 2019).

Neste capítulo, os resultados do estudo de ablação feito na WCNN são apresentados. Os testes de ablação feitos envolvem a modificação da WCNN padrão, com a base de imagens de TC e dados clínicos, BIMCV (DE LA IGLESIA VAYÁ, SABORIT, *et al.*, 2020): *i*) testes de técnicas de otimização (SGD, RMSprop e ADAM); *ii*) teste das camadas convolucionais; *iii*) testes de ganho de eficiência das técnicas de *pooling* (*max pooling*, *avg pooling* e *min pooling*); *iv*) verificações nas camadas densas ao reconfigurar os parâmetros destas camadas; e, *v*) comparar a eficiência da WCNN com a da rede de (HBORHAN, 2019) ao testar ambas com a base ChestXpert (WANG, PENG, *et al.*, 2019).

### 6.1. WCNN

Este modelo é denominado WCNN, pois é a partir dela é que os valores dos parâmetros são alterados em função dos testes realizados. Os valores são modificados para averiguar o

comportamento da rede em distintos cenários de execução. Os testes foram realizados conforme a arquitetura padrão da WCNN, apresentada na Tabela 12. A WCNN utiliza a função de ativação *Rectified Linear Unit* (ReLU) utilizada após cada camada de convolução e camadas densas.

Para a última camada densa, a função de ativação utilizada é a sigmóide. Uma taxa de uma taxa de *dropout* de 20% foi implementada para as primeiras quatro camadas densas a fim de reduzir a possibilidade de *overfitting*.

A arquitetura foi sendo modificada de acordo com os testes e a configuração utilizada foi devidamente documentada. Após a execução dos cenários de teste, os resultados obtidos foram anotados, para discussão posterior.

Tabela 12 – Arquitetura padrão da WCNN

Camada	Operação	<i>Kernels/channels/Layers</i>	<i>Kernel/input/output size</i>	Função de ativação	Taxa <i>dropout</i>
1	<i>Wave</i>	<i>Channel:1</i>	<i>Input:512×512</i>	-	-
2	Convolução	256	5×5	ReLU	
3	<i>Max pooling</i>	256	2×2	-	-
4	Convolução	128	3×3	ReLU	
5	<i>Max pooling</i>	128	2×2	-	-
6	Convolução	64	3×3	ReLU	
7	<i>Max pooling</i>	64	2×2	-	-
8	Convolução	32	3×3	ReLU	
9	<i>Max pooling</i>	32	2×2	-	-
10	<i>Fully connected</i>	256	-	ReLU	20%
11	<i>Fully connected</i>	128	-	ReLU	20%
12	<i>Fully connected</i>	64	-	ReLU	20%
13	<i>Fully connected</i>	32	-	ReLU	20%
14	<i>Output</i>	<i>Channel:3</i>	<i>Output:1</i>	Sigmóide	-

A arquitetura padrão da WCNN foi configurada com os seguintes parâmetros para o estudo: *i*) os pesos foram inicializados aleatoriamente; *ii*) o otimizador escolhido foi o ADAM (WANI, BHAT, *et al.*, 2020), com  $\beta_1 = 0,9$  e  $\beta_2 = 0,999$ , conforme literatura ( KINGMA e BA, 2014); *iii*) taxa de aprendizado inicial,  $\alpha = 0,001$ , reduzida por um fator de 10; *iv*) o número de épocas foi 200, com tamanho de lote de 32 (*batch-size*); *v*) a técnica de *pooling* adotada foi *Max pooling* com filtro (2×2); e, *vi*) taxa de 20% para a técnica de *dropout*.

### 6.1.1.Base de dados BIMCV

A base de dados, utilizada no estudo dos parâmetros da rede, foi a do repositório público

*Valencian Region Medical ImageBank (BIMCV) (DE LA IGLESIA VAYÁ, SABORIT, et al., 2020).* Ela é subdividida em:

- BIMCV-COVID19+ imagens de pacientes com teste positivo para COVID-19, e inclui, também, achados radiográficos de TC e seus respectivos laudos, exame de reação em cadeia da polimerase (PCR), testes de diagnóstico de anticorpos (imunoglobulina G-IgG e imunoglobulina M-IgM); e,
- BIMCV-COVID19- imagens de pacientes com teste negativo para COVID-19, incluindo achados radiográficos de TC e seus respectivos laudos, incluindo patologias como pneumonia, cardiomegalia, derrame pleural, atelectasia, consolidação.

Quanto à distribuição dos pacientes para os testes, foram selecionados aleatoriamente 174 pacientes, sendo 87 pacientes BIMCV-COVID19+, constituintes da base COVID-19+ e 87 pacientes BIMCV-COVID19-, que compuseram a base COVID-19+. Dos 87 pacientes da base COVID-19+, 70% foram utilizados na fase de treinamento, 15% na fase de teste e os 15% restantes na fase de validação. A mesma distribuição foi usada na base COVID-19-, conforme Tabela 13. As imagens dos pacientes utilizadas na fase de treinamento não foram usadas novamente nas fases de teste e validação.

Tabela 13 – Distribuição das imagens do repositório BIMCV

Sub-base	Treinamento Pacientes/Imagens	Teste Pacientes/Imagens	Validação Pacientes/Imagens
Covid-19+	62/5670	13/1032	13/1032
Covid-19-	62/5670	13/1032	13/1032

## 6.1.2. Teste das famílias *wavelet* discretas e coeficientes

O teste das famílias *wavelet* e coeficientes foi realizado de acordo com o descrito nas próximas subseções.

### 6.1.2.1. Escolha da família *wavelet*

No que se refere à escolha da *wavelet* mãe e do nível de decomposição, a decisão de associar uma transformada *wavelet* discreta da família *Coiflets 5* foi tomada com base nos resultados do trabalho de (DA COSTA JUNIOR e PATROCINIO, 2019, DA COSTA JUNIOR, 2019). Foram testadas as transformadas *wavelet* das famílias *Daubechies*, *Symlets*, *Coiflets*, *Fejer-Korovkin* e *dMeyer*. Dentre estas, a família *Coiflets 5* foi a que apresentou os melhores resultados de redução de ruído em imagens mamográfica de mama densa; e assim, por analogia,

a família *Coiflets 5* foi associada ao modelo WCNN. Já o nível de decomposição foi definido como um para evitar perda de informações necessárias à classificação da imagem.

Neste trabalho foi feita uma análise parcial das famílias *wavelet* discretas para verificar se a *Coiflets 5* também seria a melhor escolha para o modelo WCNN. A análise foi efetuada usando a base BIMCV e a arquitetura padrão WCNN seguindo os passos:

1. Seleção de seis famílias de *wavelet* discretas com base naquelas já implementadas na biblioteca `Pywavelet` do Python: *Biorthogonal*, *Coiflets*, *Daubechies*, “*Discrete*” *FIR approximation of Meyer wavelet*, *Reverse Biorthogonal* e *Symlets*.
2. Seleção das *tags* iniciais e finais de cada família.
3. Execução da WCNN no BIMCV usando as famílias e suas respectivas *tags*.

O resultado obtido ao final do passo 3 é exibido na Tabela 14.

Tabela 14 – Teste com as *tags* iniciais e finais de cada família *wavelet*

<b>Famílias</b>	<b>Tags</b>	<b>Acurácia</b>
<i>Biorthogonal</i>	Bior 1.3	0,9777
	Bior 6.8	0,9714
<i>Coiflets</i>	<b>Coif 5</b>	<b>0,9821</b>
	Coif 17	0,9801
<i>Daubechies</i>	Db 2	0,9733
	Db 20	0,9777
“ <i>Discrete</i> ” <i>FIR approximation of Meyer wavelet</i>	Dmey	0,9404
<i>Reverse biorthogonal</i>	Rbio 1.3	0,9767
	Rbio 6.8	0,9781
<i>Symlets</i>	Sym 2	0,9743
	Sym 20	0,9782

Os resultados também foram apresentados de acordo com outra métrica, que é a curva ROC, que são exibidas na **Erro! Fonte de referência não encontrada.**

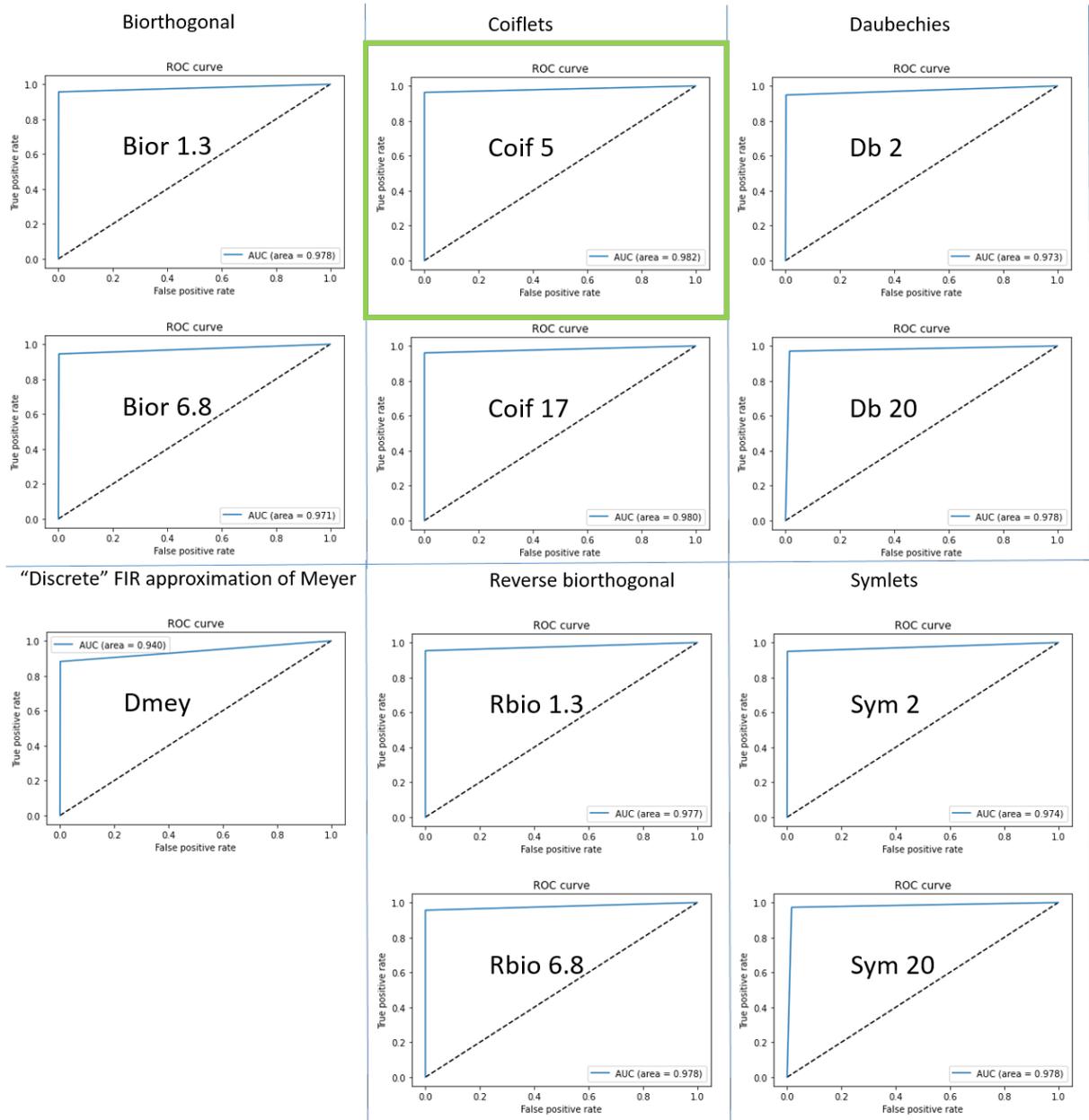


Figura 33 – Teste com as famílias da *wavelet*

### 6.1.2.2. Análise dos coeficientes da transformada *wavelet* discreta

A decomposição em um nível produz quatro sub-bandas, que são *i*) sub-banda LL (aproximada), correspondente à aplicação da função escala sobre o sinal; e, *ii*) as sub-bandas LH, HL e HH<sup>7</sup>, que correspondem, respectivamente, à aplicação das medidas das variações verticais, horizontais e diagonais.

<sup>7</sup> Onde L corresponde ao filtro *down-sampled low pass* e H se refere ao filtro *down-sampled high pass*.

Dos quatro coeficientes foram selecionados três deles mediante aplicação do teste Friedman que mostrou a existência de diferenças estatisticamente significantes entre os coeficientes aproximada, vertical e diagonal.

Em função deste resultado, foi feita a análise enunciada abaixo:

1. O teste foi feito na base BIMCV com a arquitetura WCNN padrão
2. Foi mantido o coeficiente **aproximada**, visto que ele contém mais informações acerca da imagem, sendo considerado uma imagem de baixa resolução.
3. Foram utilizados os coeficientes **aproximada**, horizontal e vertical para avaliação.
4. Foram utilizados os coeficientes **aproximada**, vertical e diagonal para avaliação
5. Foram utilizados os coeficientes **aproximada**, horizontal e diagonal

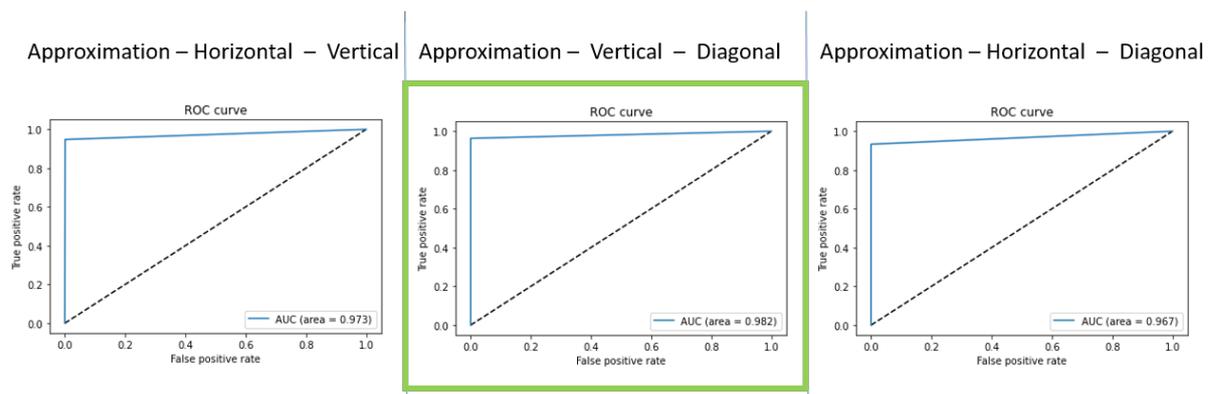
As acurácias obtidas nesse teste são mostradas na Tabela 15.

Tabela 15 – Teste da escolha dos Coeficientes

Coeficientes	Acurácia
<i>Approximation-Horizontal-Vertical</i>	0,9733
<i>Approximation-Vertical-Diagonal</i>	<b>0,9820</b>
<i>Approximation-Horizontal-Diagonal</i>	0,9675

Já as suas respectivas curvas ROC, podem ser vistas na Figura 34.

Figura 34– Teste com os coeficientes da *wavelet*



### 6.1.3. Teste das técnicas de otimização

O uso da técnica *gradient descent* é um processo lento de convergência, pois depende de parâmetros escolhidos de forma aleatória. No caso de redes neurais, está aleatoriedade recai sobre a escolha inicial dos pesos. Diversos métodos disponíveis podem ajudar um algoritmo de otimização a convergir mais rapidamente. Como mencionado, foram testadas as técnicas SGD, RMSprops e ADAM, cujos testes são descritos a seguir.

### 6.1.3.1. Stochastic Gradient Descent (SGD)

Na aplicação do *Stochastic Gradient Descent* (SGD), também conhecido como gradiente descendente incremental, o gradiente é calculado para um exemplo de treinamento por vez, seguido pela atualização dos valores dos parâmetros. Geralmente, o SGD é muito mais rápido que o GD, pois realiza uma atualização de cada vez, o que é apresentado pela Equação 43 (WANI, BHAT, *et al.*, 2020):

$$w = w - \alpha \cdot \nabla E(w, x(i), d(i)) \quad \text{Equação 43}$$

Onde  $\nabla E(w; x(i); d(i))$  é o gradiente de erro, em relação ao peso  $w$  para uma amostra de treinamento  $\{x(i), d(i)\}$ ; em que:

- $x$  são as entradas da rede neural;
- $d$  os valores esperados, também chamados *ground truth*;  $e$ ,
- $i$  é a quantidade da amostra.

No SGD, uma única amostra para atualização dos valores dos parâmetros faz com que a função de erro flutue com frequência alta.

### 6.1.3.2. RMSPROP

O RMSProp é uma versão modificada do AdaGrad que descarta a história do passado distante, introduzindo uma média móvel exponencialmente ponderada. RMSProp usa o sinal do gradiente em vez da magnitude do gradiente para atualizar os pesos. O funcionamento do otimizador RMSProp é o seguinte (KINGMA e BA, 2014; WANI, BHAT, *et al.*, 2020):

- 1) Definir a mesma magnitude de atualizações para todos os pesos (também chamada de *momentum*).
- 2) Definir as atualizações de peso máximo e mínimo permitido para máximo e mínimo, respectivamente.
- 3) A cada iteração, se os sinais do gradiente corrente e do gradiente anterior forem iguais, aumentar a taxa de aprendizado em 1.2, ou seja,  $\alpha = \alpha + 1.2$ . Portanto, a atualização  $\Delta_{ij}^{t+1}$  torna-se

$$\Delta_{ij}^{t+1} = \min(\alpha + \Delta_{ij}^t, \Delta_{max})$$

- 4) Se os sinais do gradiente atual e do gradiente anterior forem diferentes, diminuir a taxa de aprendizado em 0,5; ou seja,  $\alpha = \alpha - 0,5$ . Portanto, a atualização  $\Delta_{ij}^{t+1}$  é:

$$\Delta_{ij}^{t+1} = \max(\alpha - \Delta_{ij}^t, \Delta_{min})$$

### 6.1.3.3. ADAM

*ADAM* é um método de aprendizagem adaptativa, ou seja, ele calcula taxas de aprendizagem individuais para diferentes parâmetros. Seu nome é derivado de estimativa de *momentum* adaptativo, e a razão pela qual é chamado assim é porque Adam usa estimativas de primeiro e segundo *momentum* (o primeiro momento é a média, e o segundo momento é a variância descentralizada (o que significa que não é subtraída a média durante o cálculo da variância)). *ADAM* é uma técnica de otimização adaptativa, que aproveita as vantagens do AdaGrad e do RMSProp. Como AdaDelta e RMSProp, e salva uma média exponencialmente decrescente de gradientes quadrados anteriores  $v_t$ . Além disso, o método também calcula a média dos segundos momentos dos gradientes  $m_t$  (KINGMA e BA, 2014; WANI, BHAT, *et al.*, 2020). Os valores da média e variância não centrada  $m_t$  e  $v_t$ , respectivamente, são dados pelas Equações 44 e 45:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla_t \quad \text{Equação 44}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) \nabla_t^2 \quad \text{Equação 45}$$

ADAM atualiza as médias móveis exponenciais do gradiente  $\nabla_t$  e o gradiente ao quadrado  $\nabla_t^2$ , onde os hiperparâmetros  $\beta_1$  e  $\beta_2 \in [0, 1]$ . Na literatura, usualmente, valores que geram bons resultados utilizam  $\beta_1 = 0,9$  e  $\beta_2 = 0,99$ , os quais controlam as taxas de decaimento das médias móveis, conforme Equação 46 e Equação 47:

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad \text{Equação 46}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad \text{Equação 47}$$

A equação final para atualização é dada pela Equação 48:

$$w_{t+1} = w_t - \frac{\alpha \hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad \text{Equação 48}$$

Onde  $\alpha$  é a taxa de aprendizagem e  $\epsilon$  uma constante adicionada ao denominador dos métodos de convergência rápida para evitar a divisão por zero (KINGMA e BA, 2014; WANI, BHAT, *et al.*, 2020).

### 6.1.3.4. Resultados da aplicação dos métodos de otimização

Os métodos de otimização foram testados considerando a configuração da arquitetura padrão da WCNN. A realização dos testes foi feita considerando os valores de alfa ( $\alpha$ ), segundo Tabela 16.

Tabela 16 – Valor de alfa ( $\alpha$ ) para os testes das técnicas de otimização

<b>Técnica</b>	<b>alfa (<math>\alpha</math>)</b>
SGD	0,01
RMSprop	0,001
ADAM	0,001

Os resultados obtidos nos testes são exibidos na Tabela 17.

Tabela 17 – Resultado dos testes com três métodos otimizadores

<b>Técnica</b>	<b>Acurácia da base de validação</b>
SGD	Estouro de memória
RMSprop	97,38%
ADAM	98,2%

### 6.1.4. Teste das camadas convolucionais

Este teste consistiu na redução gradual das camadas convolucionais para aferir a eficiência da WCNN de acordo com os parâmetros dessas camadas.

#### 6.1.4.1. Teste com 3 camadas convolucionais

Neste primeiro teste, foi reduzido o número de camadas convolucionais de 4 para 3. Após modificar a configuração da arquitetura da WCNN-padrão, foi obtida a arquitetura exibida na Tabela 18.

Tabela 18 – Arquitetura da WCNN: três camadas convolucionais, destacadas em azul

<b>Camada</b>	<b>Operação</b>	<b>Kernels/channels/Layers</b>	<b>Kernel/input/output size</b>	<b>Função de ativação</b>	<b>Taxa Dropout</b>
1	<i>Wave</i>	<i>Channel:1</i>	<i>Input:512×512</i>	-	-
2	<b>Convolução</b>	<b>256</b>	<b>5×5</b>	<b>ReLU</b>	
3	<i>Max pooling</i>	<b>256</b>	<b>2×2</b>	-	-
4	<b>Convolução</b>	<b>128</b>	<b>3×3</b>	<b>ReLU</b>	
5	<i>Max pooling</i>	<b>128</b>	<b>2×2</b>	-	-
6	<b>Convolução</b>	<b>64</b>	<b>3×3</b>	<b>ReLU</b>	
7	<i>Max pooling</i>	<b>64</b>	<b>2×2</b>	-	-
8	<i>Fully connected</i>	256	-	ReLU	20%
9	<i>Fully connected</i>	128	-	ReLU	20%
10	<i>Fully connected</i>	64	-	ReLU	20%
11	<i>Fully connected</i>	32	-	ReLU	20%
12	<i>Output</i>	<i>Channel:3</i>	<i>Output:1</i>	Sigmóide	-

O teste com três camadas convolucionais resultou no total de 6.987.201 parâmetros e na diminuição da acurácia, de 98,2% para 97,77%, o que representa redução real de 0,4379%.

#### 6.1.4.2. Teste com 2 camadas convolucionais

Teste análogo ao anterior, mas as camadas convolucionais foram reduzidas de 4 para 2. Após modificar a arquitetura padrão, a resultante utilizada no teste é exibida na Tabela 19.

Tabela 19 – Arquitetura da WCNN: duas camadas convolucionais, destacadas em azul

Camada	Operação	Kernels/channels/Layers	Kernel/input/output size	Função de ativação	Taxa Dropout
1	Wave	Channel:1	Input:512×512	-	-
2	Convolução	256	5×5	ReLU	
3	Max pooling	256	2×2	-	-
4	Convolução	128	3×3	ReLU	
5	Max pooling	128	2×2	-	-
6	Fully connected	256	-	ReLU	20%
7	Fully connected	128	-	ReLU	20%
8	Fully connected	64	-	ReLU	20%
9	Fully connected	32	-	ReLU	20%
10	Output	Channel:3	Output:1	Sigmóide	-

O teste, com duas camadas convolucionais ao invés de quatro, utilizou 55.442.561 parâmetros e resultou na diminuição da acurácia, de 98,2% para 97,04%, o que representa redução real de 1,1812% na acurácia.

#### 6.1.4.3. Teste com 1 camada convolucional

Este teste consistiu na redução do número de camadas convolucionais, reduzindo-as de 4 camadas para 1 camada. A sua arquitetura utilizada para o teste é apresentada na Tabela 20.

Tabela 20 – Arquitetura da WCNN: uma camada convolucional, destacadas em azul

Camada	Operação	Kernels/channels/Layers	Kernel/input/output size	Função de Ativação	Taxa Dropout
1	Wave	Channel:1	Input:512×512	-	-
2	Convolução	256	5×5	ReLU	
3	Max pooling	256	2×2	-	-
4	Fully connected	256	-	ReLU	20%
5	Fully connected	128	-	ReLU	20%
6	Fully connected	64	-	ReLU	20%
7	Fully connected	32	-	ReLU	20%
8	Output	Channel:3	Output:1	Sigmóide	-

O teste com uma camada convolucional tem um total 451.541.505 parâmetros. Sendo que a redução de uma camada convolucional resultou no estouro de memória devido ao aumento do número de parâmetros da rede. Portanto, com a diminuição das camadas convolucionais, há um aumento significativo dos parâmetros, o que implica em maior custo de processamento e numa redução da acurácia.

### 6.1.5. Teste de *Pooling*

Este teste consistiu na utilização das técnicas de *max pooling*, *avg pooling* e *min pooling*. Como a camada *pooling* faz uma amostragem reduzida a partir da *feature map* da camada convolucional, o objetivo deste teste foi verificar qual técnica impacta mais positivamente a acurácia do modelo. As técnicas consideradas no teste foram (PONTI e DA COSTA, 2018; AGGARWAL e OTHERS, 2018; LECUN, BENGIO e HINTON, 2015):

- *Max pooling* faz uma amostragem com o máximo de cada *feature map*.
- *Min pooling* faz uma amostragem com o mínimo de cada *feature map*.
- *Avg pooling* faz uma amostragem com a média de cada *feature map*.

As técnicas foram testadas considerando a configuração da arquitetura padrão da WCNN. Os resultados obtidos são exibidos na Tabela 21.

Tabela 21 – Resultado dos testes utilizando as diferentes técnicas de *pooling*

Técnica	Acurácia da base de validação
<i>Max pooling</i>	0,9801
<i>Min pooling</i>	-
<i>Avg pooling</i>	0,9700

A técnica *Max pooling* obteve acurácia de 98% e a *Avg pooling*, 97%. A técnica *Min pooling* não foi utilizada pois isto resultaria num mapa de ativação em zero ou próximo de zero. Isto acontece porque, ao repetir o *Min pooling* algumas vezes, os valores de ativação serão zero. Assim, a rede não poderá ser treinada, porque todas as informações uteis teriam sido perdidas.

### 6.1.6. Teste das camadas densas

Este teste consistiu na modificação da quantidade das camadas densas para aferir a eficiência da WCNN de acordo com os parâmetros dessas camadas.

### 6.1.6.1. Primeiro teste com a WCNN padrão

Tabela 22 – Arquitetura da WCNN padrão

<b>Camada</b>	<b>Operação</b>	<b>Kernels/channels Layers</b>	<b>Kernel/input Output size</b>	<b>Função de Ativação</b>	<b>Dropout</b>
1	Wave	<i>Channel:1</i>	<i>Input:512×512</i>	-	-
2	Convolução	256	5×5	ReLU	
3	<i>Max pooling</i>	256	2×2	-	-
4	Convolução	128	3×3	ReLU	
5	<i>Max pooling</i>	128	2×2	-	-
6	Convolução	64	3×3	ReLU	
7	<i>Max pooling</i>	64	2×2	-	-
8	Convolução	32	3×3	ReLU	
9	<i>Max pooling</i>	32	2×2	-	-
10	<i>Fully connected</i>	256	-	ReLU	20%
11	<i>Fully connected</i>	128	-	ReLU	20%
12	<i>Fully connected</i>	64	-	ReLU	20%
13	<i>Fully connected</i>	32	-	ReLU	20%
14	<i>Output</i>	<i>Channel:3</i>	<i>Output:1</i>	Sigmóide	-

A arquitetura do teste da WCNN padrão é apresentado na Tabela 22. A WCNN padrão tem um total de **1.115.745** parâmetros e sua execução gerou as métricas apresentadas na Tabela 23.

Tabela 23 – Resultados das métricas do teste da WCNN padrão

<b>Classe</b>	<b>Accuracy</b>	<b>Recall /sensitivity</b>	<b>Specificity</b>
Covid-19+	0,9820	0,9999	0,9631

Usando os valores da Tabela 23, a curva ROC foi calculada com  $(1-Sp)=0,0369$  e  $Sen=0,9999$ , como eixos  $x$  e  $y$ , respectivamente. Com base na análise ROC, a área (AUC) foi calculada e o valor obtido foi 0,982, como exibido na Figura 35.

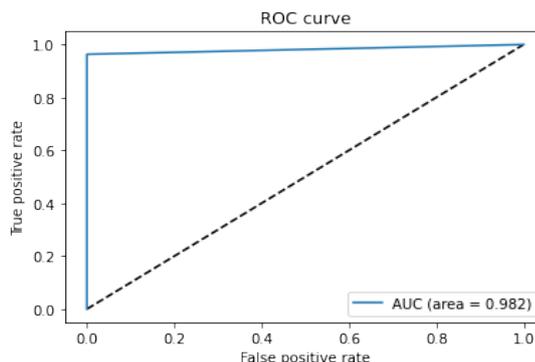


Figura 35 – Curva ROC da WCNN padrão

### 6.1.6.2. Segundo teste com a WCNN modificada

A arquitetura da WCNN modificada, para o segundo teste, é apresentada na Tabela 24, alterando as camadas densas para 256-300-150-50.

Tabela 24 – Arquitetura da WCNN modificada-2 teste

Camada	Operação	<i>Kernels/channels/layers</i>	<i>Kernel/input/output size</i>	Função de Ativação	Taxa de Dropout
1	Wave	<i>Channel:1</i>	<i>Input:512×512</i>	-	-
2	Convolução	256	5×5	ReLU	
3	<i>Max pooling</i>	256	2×2	-	-
4	Convolução	128	3×3	ReLU	
5	<i>Max pooling</i>	128	2×2	-	-
6	Convolução	64	3×3	ReLU	
7	<i>Max pooling</i>	64	2×2	-	-
8	Convolução	32	3×3	ReLU	
9	<i>Max pooling</i>	32	2×2	-	-
10	<i>Fully connected</i>	256	-	ReLU	20%
11	<i>Fully connected</i>	300	-	ReLU	20%
12	<i>Fully connected</i>	150	-	ReLU	20%
13	<i>Fully connected</i>	50	-	ReLU	20%
14	<i>Output</i>	<i>Channel:3</i>	<i>Output:1</i>	Sigmóide	-

Esta configuração da WCNN modificada tem um total de **1.202.331** parâmetros, e a execução do modelo obteve as seguintes métricas, apresentadas na Tabela 25.

Tabela 25 – Resultados das métricas da Arquitetura da WCNN modificada-2 teste

Classe	<i>Accuracy</i>	<i>Recall /sensitivity</i>	<i>Specificity</i>
Covid-19+	0,9816	0,9980	0,9651

Usando os valores da Tabela 25, a curva ROC foi calculada com  $(1-Sp) = 0,0369$  e  $Sen = 0,9999$ , as quais como eixos  $x$  e  $y$ , respectivamente. Com base na análise ROC, a área (AUC) foi calculada e o valor obtido foi 0,982, como exibido na Figura 36.

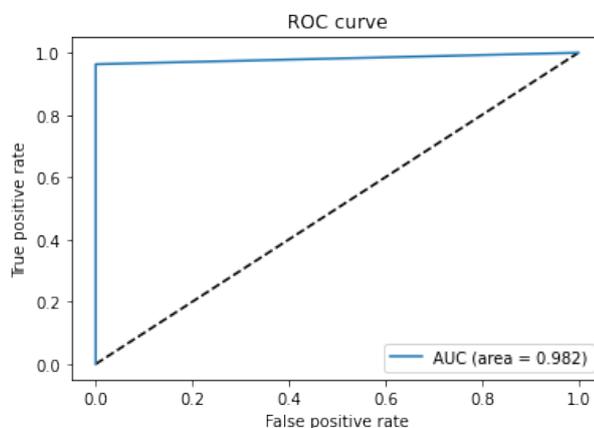


Figura 36 – Curva ROC da WCNN modificada-2 teste

#### A. Terceiro teste com a WCNN modificada

A arquitetura da WCNN modificada, para o terceiro teste, é apresentada na Tabela 26, alterando as camadas densas para 128-64-32-16.

Tabela 26 – Arquitetura da WCNN modificada-3 teste

Camada	Operação	Kernels/channels/layers	Kernel/input/output size	Função de Ativação	Taxa Dropout
1	Wave	Channel:1	Input:512×512	-	-
2	Convolução	256	5×5	ReLU	
3	Max pooling	256	2×2	-	-
4	Convolução	128	3×3	ReLU	
5	Max pooling	128	2×2	-	-
6	Convolução	64	3×3	ReLU	
7	Max pooling	64	2×2	-	-
8	Convolução	32	3×3	ReLU	
9	Max pooling	32	2×2	-	-
10	Fully connected	128	-	ReLU	20%
11	Fully connected	64	-	ReLU	20%
12	Fully connected	32	-	ReLU	20%
13	Fully connected	16	-	ReLU	20%
14	Output	Channel:3	Output:1	Sigmóide	-

Esta configuração da WCNN modificada tem um total de **751.457** parâmetros, e obteve as seguintes métricas, apresentadas na Tabela 27.

Tabela 27 – Resultados das métricas da Arquitetura da WCNN modificada-3 teste

<b>Classe</b>	<b>Accuracy</b>	<b>Recall /sensitivity</b>	<b>Specificity</b>
Covid-19+	0,9790	0,9961	0,9622

Usando os valores da Tabela 27, a curva ROC foi calculada com  $(1-Sp) = 0,0378$  e  $Sen = 0,9980$ , como x e y, respectivamente. Com base na análise ROC, a área (AUC) foi calculada como sendo 0,9820, como exibido na Figura 37.

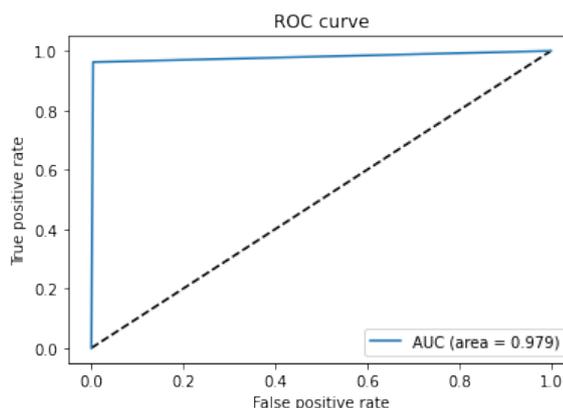


Figura 37 – Curva ROC da WCNN modificada-3 teste

## B. Quarto teste com a WCNN modificada

A arquitetura da WCNN modificada para o quarto teste, envolvendo apenas uma camada densa, apresentada na Tabela 28, alterando a camada densa para 64.

Tabela 28 – Arquitetura da WCNN modificada-4 teste

<b>Camada</b>	<b>Operação</b>	<b>Kernels/channels/layers</b>	<b>Kernel/input/output size</b>	<b>Função de ativação</b>	<b>Taxa de Dropout</b>
1	Wave	<i>Channel:1</i>	<i>Input:512×512</i>	-	-
2	Convolução	256	5×5	ReLU	
3	<i>Max pooling</i>	256	2×2	-	-
4	Convolução	128	3×3	ReLU	
5	<i>Max pooling</i>	128	2×2	-	-
6	Convolução	64	3×3	ReLU	
7	<i>Max pooling</i>	64	2×2	-	-
8	Convolução	32	3×3	ReLU	
9	<i>Max pooling</i>	32	2×2	-	-
<b>10</b>	<b><i>Fully connected</i></b>	<b>64</b>	<b>-</b>	<b>ReLU</b>	<b>20%</b>
14	<i>Output</i>	<i>Channel:3</i>	<i>Output:1</i>	Sigmóide	-

Esta configuração da WCNN modificada tem um total de **574.689** parâmetros, e obteve as seguintes métricas, apresentadas na Tabela 29.

Tabela 29 – Resultados das métricas da Arquitetura da WCNN modificada-3 teste

Classe	Accuracy	Recall /sensitivity	Specificity
Covid-19+	0,9593	0,9910	0,9186

Usando os valores da Tabela 29, a curva ROC foi calculada com  $(1-Sp) = 0,0814$  e  $Sen = 0,9910$ , como x e y, respectivamente. Com base na análise ROC, a área (AUC) foi calculada como sendo 0,9559, como exibido na Figura 38.

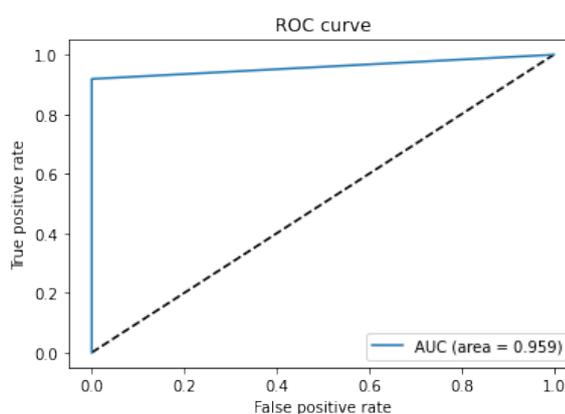


Figura 38 – Curva ROC da WCNN modificada-4 teste

## 6.2. Teste de eficácia com a base ChestXpert

Neste teste, a eficiência da WCNN foi comparada com a da pesquisa realizada por (HBORHAN, 2019); e, neste contexto, eficácia deve ser entendida como AUC com o mesmo conjunto de imagens. Para que a comparação fosse fidedigna, a WCNN foi executada não somente a mesma base de dados replicando os passos seguidos pelo autor acima mencionado. Assim, a base de dados utilizada foi a base de imagens radiografias de tórax chamada ChestXpert WANG, PENG, *et al.*, 2019.

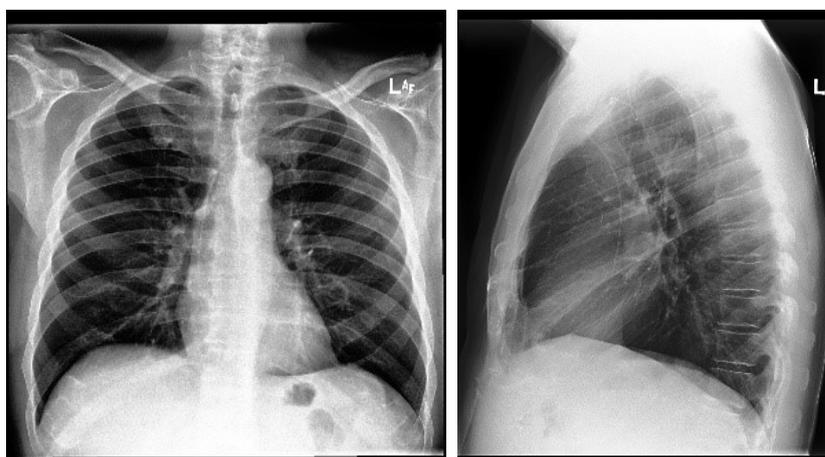


Figura 39 – Radiografia frontal e lateral de um mesmo paciente.

*ChestXpert* WANG, PENG, *et al.*, 2019 é uma base de dados de imagens de radiografia de tórax disponibilizada para um desafio público de Aprendizado de Máquina, consistindo em 224.316 imagens de 65240 pacientes. Como o desafio fica continuamente em andamento, estão disponíveis apenas os conjuntos de treinamento e validação, que possuem 223414 e 234 imagens, respectivamente. As imagens são divididas em 14 classes. Cada imagem recebe o nome de uma doença e seu respectivo laudo radiológico, podendo ser inferida como *positiva*, *incerta* ou *negativa*. A Tabela 30 detalha as divisões do banco de dados ChestX-ray14.

Este banco de dados foi criado no *Stanford Hospital*, no período compreendido entre outubro de 2002 e julho de 2017, e conta com imagens dos centros de internação e dos ambulatórios. A Figura 39 apresenta exemplos de imagens de radiografia de um paciente.

Tabela 30 – Divisão da base de dados ChestX-ray14

Classe	Treinamento			Validação			Teste		
	Positivo	Incerto	Negativo	Positivo	Incerto	Negativo	Positivo	Incerto	Negativo
Normal	15576	0	141051	2163	0	20408	4680	0	39770
Cardiomedial aumentado	7632	8596	140399	1079	1358	20134	2196	2449	39805
Cardiomegalia	18840	5657	132130	2783	828	18960	5445	1602	37403
Opacidade pulmonar	74154	3837	78636	10766	559	11246	20787	1202	22461
Lesão pulmonar	6434	1058	149135	963	141	21467	1790	289	42371
Edema pulmonar	36775	9211	110641	5324	1241	16006	10192	2532	31726
Consolidação pulmonar	10412	19472	126743	1479	2822	18270	2925	5448	36077
Pneumonia	4277	13184	139166	625	1975	19971	1145	3611	39694
Atelectasia	23464	23588	109575	3317	3377	15877	6675	6774	31001
Pneumotórax	13564	2187	140876	1974	333	20264	3918	625	39907
Derrame pleural	60147	8217	88263	8746	1098	12727	17361	2313	24776
Outros problemas pleurais	2409	1834	152384	346	248	21977	769	571	43110
Fratura	6329	471	149827	935	61	21575	1776	110	42564
Dispositivos de suporte	81363	758	74506	11893	107	10571	22852	214	21384

A bases de dados foi dividida da mesma forma que os trabalhos correlatos (RAJPURKAR, IRVIN, *et al.*, 2017, 23,25, WANG, PENG, *et al.*, 2019, etc.) sendo 70% para treinamento, 10% para validação e 20% para teste, como mostrado na Tabela 30. Assim, é possível realizar uma comparação coerente com outros trabalhos disponíveis na literatura. A Tabela 31 apresenta algumas informações relevantes sobre a bases de dados ChetXpert.

Tabela 31 – Informações sobre a bases de dados ChetXpert

Base de Dados	Tipo	Dimensões	Formato	Profundidade	Tamanho
<i>ChestXpert</i> Reduzida	CXR	320×320	JPG	8 bits	~11 GB

### 6.2.1. Aplicação da WCNN em imagens de fundo de olho (DenseNet121)

Vários métodos de aprendizado profundo foram projetados e avaliados neste conjunto de dados participando da competição CheXpert organizada pelo grupo Stanford ML (IRVIN, RAJPURKAR, *et al.*, 2019), e muitos deles alcançaram desempenho em nível de radiologista na detecção de certas doenças relacionadas.

Para a comparação com o desempenho da WCNN, foi escolhido um trabalho com a rede DenseNet121, pois ela é utilizada nos trabalhos correlatos (RANGARAJAN e RAMACHANDRAN, 2022, WANG, PENG, *et al.*, 2019, SALEEM, SHEIKH e KHALID, 2021). Huang *et al.* (HUANG, LIU, *et al.*, 2017) propuseram a DCNN *DenseNet* que tem como objetivo principal garantir o fluxo máximo de informações entre as camadas da rede, ligando todas as camadas diretamente às demais. Para preservar o *feed-forward*, cada camada obtém entradas adicionais de todas as camadas anteriores e passa seus *feature maps* para todas as camadas subsequentes.

### 6.2.2. Parâmetros

O experimento foi realizado conforme todos os parâmetros da competição. Assim, os mesmos parâmetros foram utilizados pela *DenseNet* 121 foram os mesmos da WCNN: *i*) o otimizador ADAM, foi utilizado com uma taxa de aprendizagem 0,001; *ii*) entropia Cruzada Binária como função de perda; *iii*) 100 épocas de duração como limite para a etapa de treinamento; *iv*) as bases de dados foram divididas em 70% para treinamento, 10% para validação e 20% para teste; *v*) não utilização de técnicas de *data augmentation* para a balancear a base, mantendo-a desbalanceada; e, *vi*) uso da métrica AUC como medida de qualidade dos resultados.

### 6.2.3. Resultados

A realização dos testes com a *DenseNet* 121 (HBORHAN, 2019) e a WCNN revelaram os resultados mostrados na Tabela 32.

Tabela 32 – Comparação dos modelos

CNN	AUC
DenseNet121,	80,65%
WCNN	85,96%

Este resultado mostra que a WCNN é 6,584% mais eficiente que a rede baseada em *DenseNet* 121.

### 6.3. Teste de resizing com a base DRIVE

O teste de *resizing* foi baseado no experimento descrito na tese (SANTOS, 2022), o qual utilizou a base DRIVE (HOOVER, KOUZNETSOVA e GOLDBAUM, 2000), uma rede neural baseada em CNN, com arquitetura Unet – criada pelo autor da tese – e técnica de segmentação. A rede neural foi criada para ajudar na detecção de enfermidades como retinopatia diabética, degeneração macular, glaucoma e hipertensão (SANTOS, 2022). A base DRIVE é disponibilizada publicamente para condução de estudos em que é necessário extrair vasos sanguíneos. Este banco de dados contém 40 imagens coloridas do fundo do olho (584×565) que são divididas em um conjunto de teste (20 imagens) e um conjunto de treinamento (20 imagens) (NIEMEIJER, STAAL, *et al.*, 2004). Cada imagem é acompanhada por uma segmentação manual (*ground truth*). Um exemplo de uma imagem com seu respectivo *ground truth* é apresentado na Figura 40.

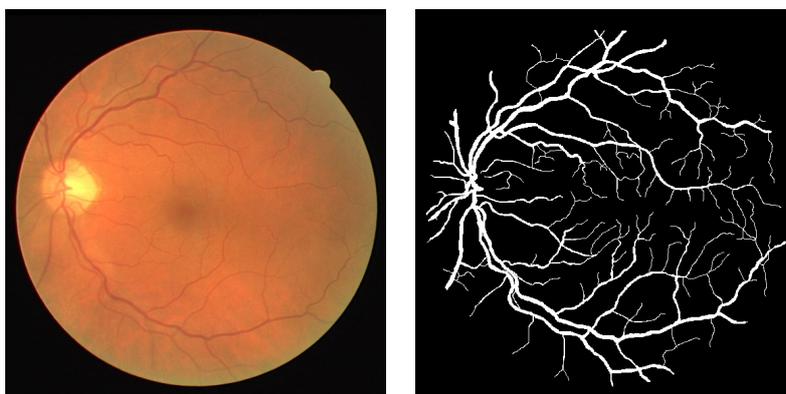


Figura 40 – Exemplo de imagem da base DRIVE com seu respectivo *ground truth*

O teste com a WCNN teste replicou exatamente os passos descritos na tese (SANTOS, 2022). Porém, com a mudança da rede Unet pela WCNN. O objetivo deste teste é demonstrar a versatilidade da *Wave layer* e mostrar o problema do redimensionamento.

A *Wave layer* recebe uma imagem, com resolução espacial de  $565 \times 565$  como entrada e seu respectivo *ground truth*. A imagem é processada pela *Wave layer* em um *workflow* composto por vários passos os quais são mostrados no diagrama da Figura 41.

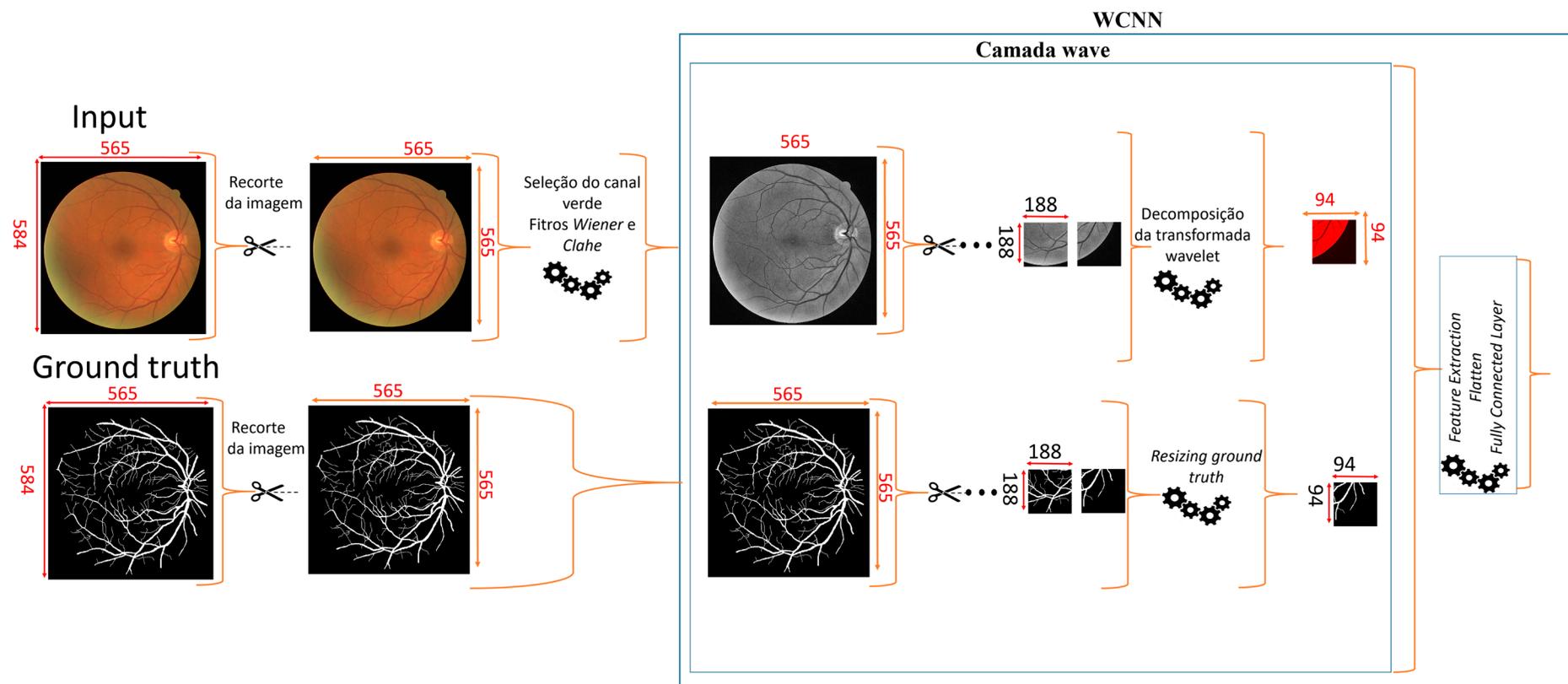


Figura 41 – *Workflow* do experimento com a base DRIVE

A seguir, cada passo do *workflow* será descrito detalhadamente.

### 6.3.1. Primeiro passo

O primeiro passo é responsável pela redução do impacto do fundo. A imagem é recortada retirando-se 10 *pixels* na margem superior e 9 *pixels* na margem superior, o que resulta em uma imagem de 565×565 *pixels*, conforme ilustrado pela Figura 42. O mesmo processo é repetido no respectivo *ground truth* da imagem.

	212	213	214	215	216	217
249	255	255	255	0	0	0
250	255	255	255	0	0	0
251	255	255	255	0	0	0
252	255	0	255	255	255	255
253	0	0	0	0	0	0
254	0	0	0	0	0	0
255	0	0	0	0	0	0
256	0	0	0	0	0	0
257	0	0	0	0	0	0
258	0	0	0	0	0	0
259	0	0	0	0	0	0
260	255	0	0	0	0	0
261	255	0	0	0	0	0
262	255	0	0	0	0	0
263	255	255	0	0	0	0
264	0	255	255	0	0	0
265	0	255	255	255	0	0
266	0	0	255	255	255	255

img22\_gt      Format: %d

Colored cells  
 Resize Automatically

Close

Figura 42 – Exemplo dos limites da imagem recortada

### 6.3.2. Segundo passo

No segundo passo, ocorre a seleção do canal e a definição dos parâmetros dos filtros. No teste om a WCNN, foram utilizados os mesmos canal e valores dos parâmetros descritos na tese de (SANTOS, 2022). Feito isso, a imagem colorida é separada em canais RGB, no entanto, apenas o canal *green* foi considerado no teste, pois este canal resulta na imagem mais detalhada dos vasos sanguíneos (SANTOS, 2022). Em seguida, o ruído do processo de aquisição da imagem é minimizado por meio da filtragem de *Wiener*. Em seguida, o contraste dos vasos sanguíneos foi melhorado pelo CLAHE (DOS SANTOS, CARRIJO, *et al.*, 2020, SANTOS, 2022).

### 6.3.3. Terceiro passo

No terceiro passo a *Wave layer* recorta a imagem e sua respectiva *ground truth* em ROIs de tamanho  $188 \times 188$ . Este tamanho foi considerado na tese (SANTOS, 2022), obtendo bons resultados. O Processo de recorte e apresentado na Figura 43.

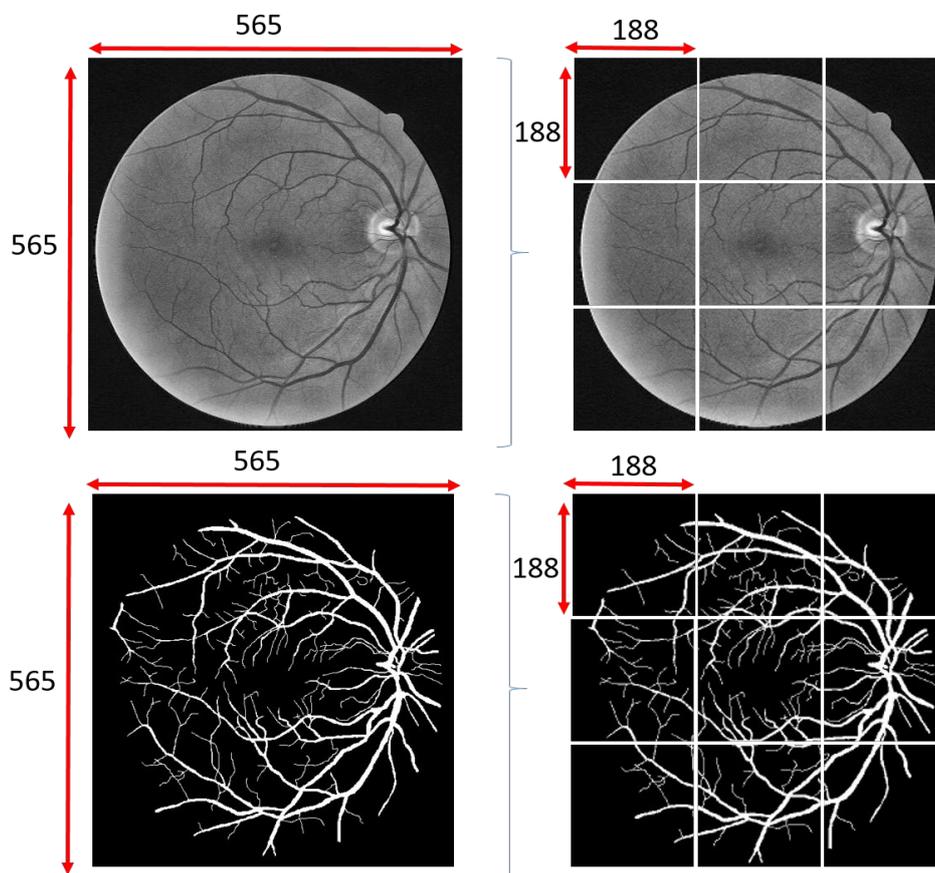


Figura 43 – Exemplo da imagem original e suas respectivas ROIs

### 6.3.4. Quarto passo

No quarto passo, a imagem de entrada é processada pela transformada *wavelet* em um único nível de decomposição, utilizando a *wavelet* mãe da família *Daubechies* com a tag *db1*. A seleção desta família e da *tag* foram feitas de acordo com as métricas encontradas no trabalho (DASH e SENAPATI, 2020). Dos quatro coeficientes gerados (aproximada, horizontal, vertical e diagonal), foram utilizados apenas três deles para formar a imagem digital. Considerando modelos de espaços de imagens como RGB, a decomposição das imagens pela transformada *wavelet* também considerou um espaço de 3 canais: o canal R recebe o coeficiente de aproximação, o canal G recebe o coeficiente horizontal e o canal B recebe o coeficiente vertical, formando uma saída de decomposição da *wavelet* com resolução espacial  $94 \times 94$ .

Todas as variações dos coeficientes foram testadas, mas somente a combinação aproximada, horizontal e vertical forneceu a imagem mais detalhada ao lidar com o problema de segmentação de vasos, como exibido na Figura 44

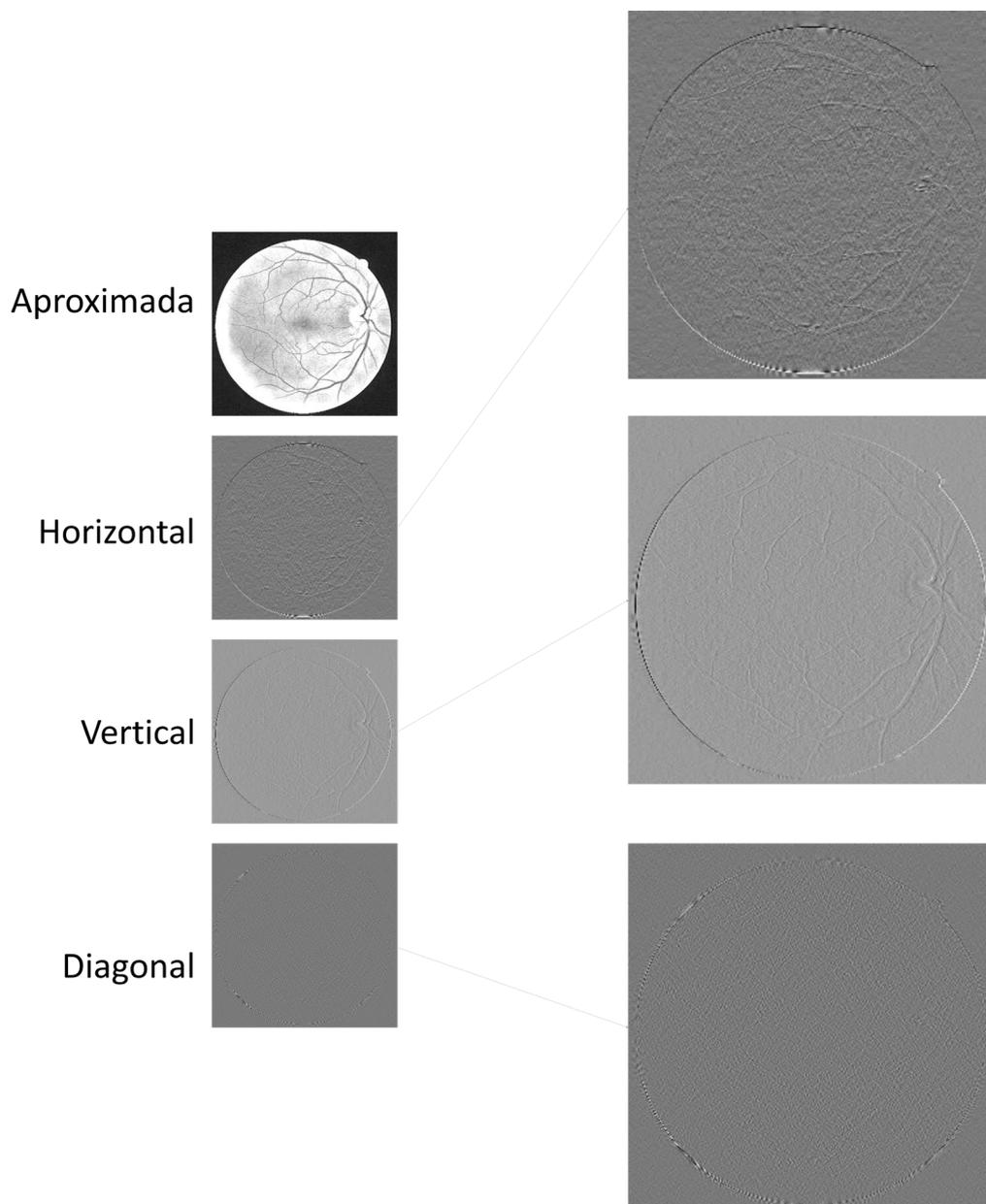


Figura 44 – Transformada *wavelet* e seus coeficientes de uma imagem do olho

Como pode ser visto na Figura 44, os coeficientes aproximada, vertical e horizontal mostram melhor o delineamento dos vasos do olho.

### 6.3.5. Quinto passo

Por fim, no quinto passo, uma imagem com resolução final de  $94 \times 94$  é gerada por interpolação bicúbica, porque o tamanho da imagem de saída da transformada *wavelet* é metade da imagem original. Assim, também foi necessário reduzir o *ground truth*, que tinha resolução

de  $188 \times 188$ , uma vez que seria inviável, para a WCNN, treinar com imagem de entrada e *ground truth* de resoluções diferentes.

A escolha da interpolação bicúbica foi baseada no trabalho (TRIWIJOYOA e ADILA, 2021). O método de interpolação bicúbica neste trabalho, utilizando a mesma base DRIVE teve a menor taxa de erro para redimensionamento de imagem em comparação com os outros dois métodos, ou seja, interpolação bilinear e interpolação do vizinho mais próximo. Assim, a interpolação bicúbica foi utilizada neste experimento para o *resizing* do *ground truth* gerando uma resolução espacial  $94 \times 94$ .A

### **6.3.6.Resultados e discussão**

#### ***6.3.6.1.Resultado da aplicação da WCNN nas imagens ópticas de fundo de olho***

A tese (SANTOS, 2022) utilizou Unet e alcançou acurácia de 0,9713. O modelo WCNN alcançou acurácia de 0,9570. A Figura 45 apresenta alguns resultados do experimento, onde a primeira coluna apresenta a imagem original, a segunda apresenta a imagem filtrada do canal verde, a terceira apresenta o *ground truth* e a última apresenta o resultado da segmentação da WCNN.

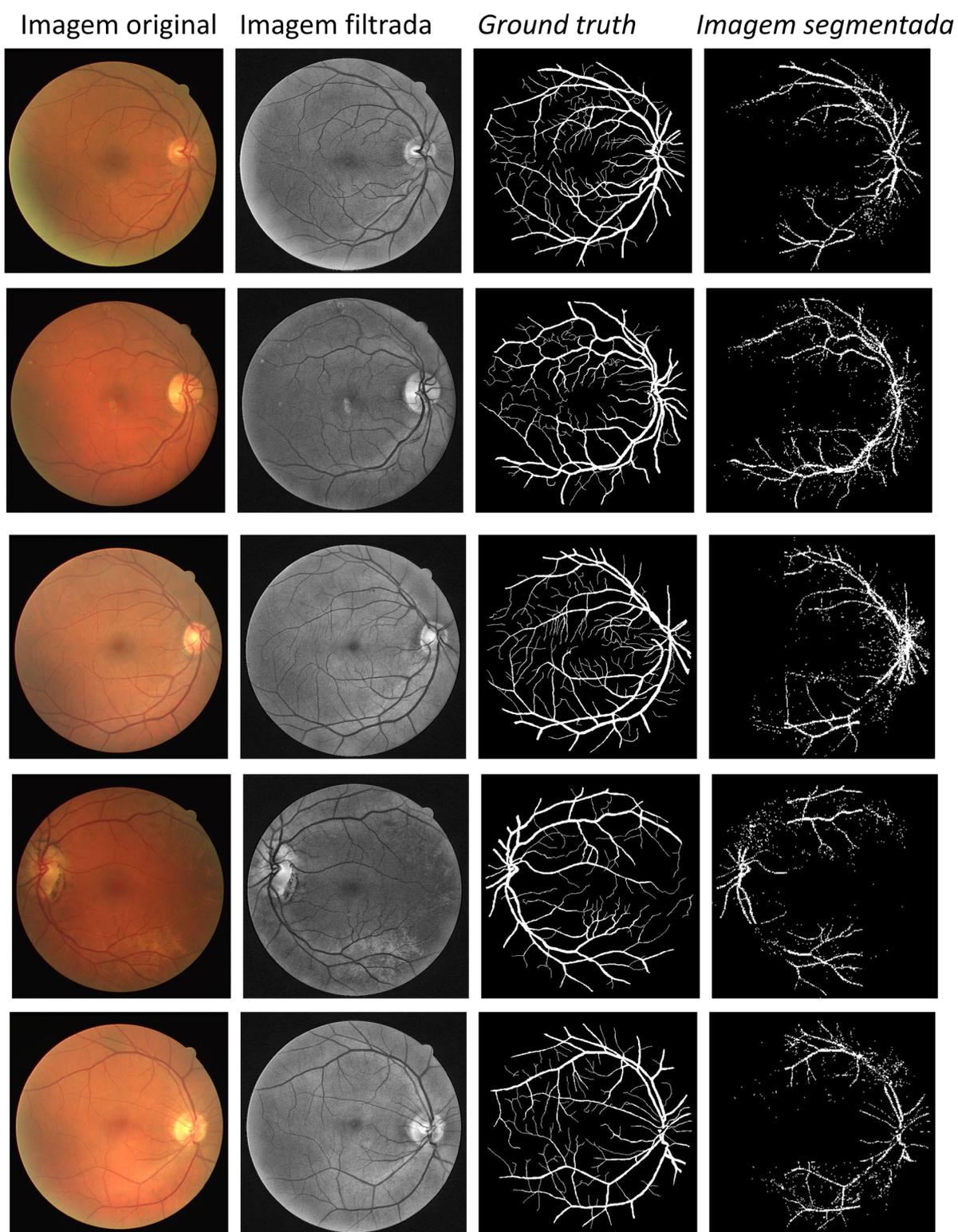


Figura 45 – Alguns resultados da aplicação da WCNN

### 6.3.6.2. Problemas do resizing

A Figura 46 mostra o efeito do *resizing* no *ground truth*.

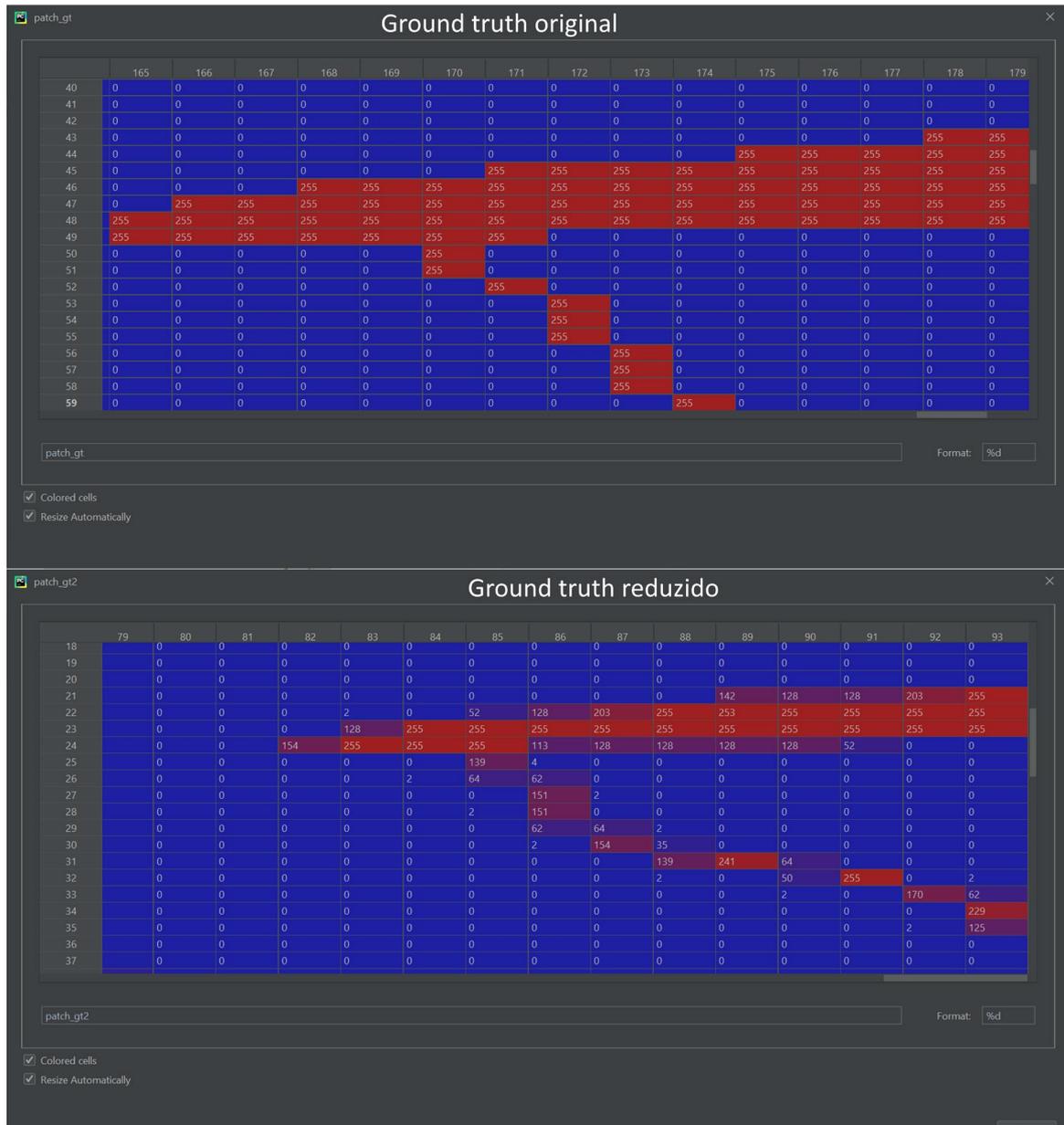


Figura 46 – Efeito do resizing do *ground truth*

A ROI do *ground truth* original de tamanho  $188 \times 188$  tem apenas dois valores 0 e 255. O zero representa o fundo e o 255 representa os pontos brancos, delimitando a segmentação dos vasos. Com o *resizing* do *ground truth* para o tamanho  $94 \times 94$  há inserção de novos valores no intervalo de  $[0, 255]$ . Isto gera problemas de aprendizado para a WCNN, pois ao normalizar o *ground truth* original obter-se-iam apenas dois valores: 0 para o fundo e 1 para os pontos brancos. Mas com o *ground truth* reduzido, haveria mais que dois valores pertencentes ao intervalo  $[0, 1]$ . Desta forma, conclui-se que o *resizing* interferiu na reconstrução de vasos menores como é apresentado na Figura 46.

## 6.4. Discussão

### 6.4.1. Discussão sobre as famílias wavelets discretas e coeficientes

Para ratificar a pesquisa do (DA COSTA JUNIOR e PATROCINIO, 2019, DA COSTA JUNIOR, 2019), foi feita uma análise parcial das *tags* iniciais e finais das famílias *wavelet* discretas *Biorthogonal*, *Coiflets*, *Daubechies*, “*Discrete*” *FIR approximation of Meyer wavelet*, *Reverse biorthogonal* e *Symlets*. Não foi feita a análise de todas as *tags* em função do tempo necessário para isso, uma vez que seriam 90 *tags* no total, que foi o escopo da pesquisa do (DA COSTA JUNIOR e PATROCINIO, 2019, DA COSTA JUNIOR, 2019). Vale ressaltar que o resultado das demais famílias foi próximo daquele obtido pela *Coiflets 5*, variando no intervalo [0,0039 – 0,0417], sendo 0,0039 da *Sym 20* e 0,0417 da *Dmey*. Dessa forma, a escolha da *Coiflets 5* foi mantida.

Já para a escolha de três coeficientes dentre quatro, o teste de Friedmann apresentou diferença estatística significativa para os coeficientes aproximada, vertical e diagonal. Para fazer uma contraprova deste resultado, o coeficiente aproximada foi mantido e os outros três (vertical, horizontal e diagonal) foram variados. Os resultados obtidos são mostrados na Tabela 14, e a melhor acurácia foi obtida pela combinação aproximada-vertical-diagonal, com resultado 0,9820, confirmando, assim, o teste de Friedmann.

Para provar que a ordem dos coeficientes não é relevante, outro teste foi feito, no qual o coeficiente aproximada foi fixado e os outros dois foram variados. Os resultados são apresentados na Tabela 33.

Tabela 33 – Teste de verificação da ordem dos coeficientes

Coeficientes	Acurácia
<i>Approximation-<b>Horizontal</b>-Vertical</i>	0,9733
<i>Approximation-Vertical-<b>Horizontal</b></i>	0,9767

Note que a diferença entre um e outro, 0,0034, está na casa dos décimo-milésimos, o que pode ser considerado irrelevante.

### 6.4.2. Discussão sobre as técnicas de otimização

Foram testadas as três técnicas de otimização: SGD, RMSProp e o ADAM. Para a técnica SGD houve um estouro de memória não conseguindo rodar 200 épocas, com uma base de dados grande com mais de 10000 registros. Foi obtido as seguintes acurácias 97,38% e 98,2% para as técnicas RMSProp e o ADAM, respectivamente.

Como foi visto nos resultados acima, a técnica ADAM tem um desempenho ligeiramente melhor do que outras técnicas adaptativas e converge muito rápido. Ele também supera os problemas enfrentados por outras técnicas de otimização, como taxa de aprendizagem decrescente, alta variação nas atualizações e convergência lenta.

### 6.4.3. Discussão sobre o teste das camadas convolucionais

Sobre a remoção das camadas convolucionais, há o aumento do número de parâmetros, como é mostrado nos testes de remoção de 1, 2, 3 das 4 camadas convolucionais.

Remover algumas camadas convolucionais, significa que o tamanho da imagem/tensor, antes de ser passado para *fully connected layer*, (*flatten layer*), será maior e conterá mais *pixels*, resultando em um número maior de parâmetros na *fully connected layer*, pois um de seus neurônios é conectado a todos os *pixels* da *flatten layer*, como mostra a Tabela 34. Assim, a redução das camadas convolucionais leva a um aumento geral no número de parâmetros da rede.

Tabela 34 – Teste nas camadas convolucionais

Número de camadas convolucionais	Número de parâmetros da 1 camada densa	Total de parâmetros	Acurácia da base de validação
4 camadas	663.808	1.115.745	98,2%
3 camadas	6.553.856	6.987.201	97,77%
2 camadas	55.083.264	55.442.561	97,04%
1 camada	451.477.760	451.541.505	Estouro de memória

A redução s técnicas foram testadas considerando a configuração da arquitetura padrão da WCNN. As acurácias obtidas são exibidas na Tabela 34. As acurácias são 98,2%, 97,77, 97,04% para 4 camadas, 3 camadas e 2 camadas, respectivamente.

A partir dos dados da Tabela 34 pode-se inferir que: *i*) a redução de quatro para três camadas ocasionou perda de 0,438%; *ii*) a modificação de quatro para duas camadas ocasionou perda de 1,181%; e; *iii*) a redução de quatro para uma camada não foi possível computar, pois aconteceu estouro de memória devido ao aumento excessivo do número de parâmetros, conforme Tabela 34.

### 6.4.4. Discussão sobre as técnicas de *Pooling*

Foram testadas as três técnicas de pooling: *max pooling*, *min pooling* e *avg pooling*. A técnica *max pooling* obteve acurácia de 98% e a *avg pooling*, 97%. A técnica *min pooling* não foi utilizada pois isto resultaria num mapa de ativação em zero ou próximo de zero. Isto acontece porque, ao repetir o *min pooling* algumas vezes, os valores de ativação serão zero. Assim, a rede não poderá ser treinada, porque todas as informações uteis teriam sido perdidas. Em

geral, não é correto dizer que um método de agrupamento é melhor que outros, pois o agrupamento nos dá uma certa quantidade de invariância de translação. O *avg pooling* ajuda a extrair recursos suaves, pois em se tratando de imagens, é obtido a combinação de todas as cores apresentadas na região coberta pelo *feature map*. Assim, o método *avg pooling* suaviza a imagem e, portanto, as características nítidas podem não ser identificadas quando esse método de agrupamento é usado. O *max pooling* se concentra apenas nos recursos muito importantes, como arestas, pontos etc. ou se o foco é sobre processamento de imagem, o *max pooling* ajuda a extrair os recursos mais nítidos da imagem e os recursos mais nítidos são a melhor representação de nível inferior da imagem.

Conseqüentemente, a escolha do melhor método de pooling depende das expectativas da camada de *pooling*, da arquitetura da CNN e do problema em questão. No estudo de caso de imagens de TC de pacientes com COVID-19, *max pooling* foi escolhido, por extrair características importantes para a classificação da doença e apresentar bom resultado.

#### 6.4.5. Discussão sobre o teste das camadas densas

A camada densa é uma rede neural que possui conexão profunda, o que significa que cada neurônio dessa camada recebe entrada de todos os neurônios de sua camada anterior. Esta realiza uma multiplicação matriz-vetor, sendo que os valores usados na matriz são parâmetros que treináveis e atualizáveis com ajuda de retropropagação. Sobre a modificação das camadas densas, a redução ou aumento do número de parâmetros de acordo com as modificações da estrutura da camada densa, como exibido na Tabela 35. As vantagens da camada densa, é que ela oferece recursos de aprendizado de todos os recursos combinacionais da camada anterior, mas com a desvantagem do custo computacional.

Tabela 35 – Teste com número variável de camadas densas

Número de camadas convolucionais	Total de parâmetros	Acurácia da base de validação
WCNN Padrão 256-128-64-32	1.115.745	98,2%
I-Modificada 256-300-150-50	1.202.331	98,16%
II-Modificada 128-64-32-16	751.457	97,9%
III-Modificada 64	574.689	95,93%

A Tabela 35 apresenta os resultados obtidos com as modificações: *i*) a modificação I-Modificada mostra uma redução de 0,0407% em relação à WCNN padrão, *ii*) II-Modificada

mostra uma redução de 0,3054% e a III-Modificada mostra uma redução de 2,3116%. As diferenças são pequenas, mas quando a aplicação lida com vidas humanas, qualquer pequeno percentual é importante.

A III-Modificada mostra a importância das camadas ocultas na CNN, pois nessa modificação foi retirado todas as camadas ocultas, sendo que uma a camada oculta nas redes neurais artificiais é uma camada de neurônios, cuja saída está conectada às entradas de outros neurônios e, portanto, não é visível como saída da rede. As camadas ocultas aprendem um grupo de características que ajudam a identificar o objeto de estudo. Como exemplo pode-se citar um sistema de reconhecimento de imagem desenvolvido para reconhecimento de rosto humano (ZANIN, AITYA, *et al.*, 2021). Diferentes facetas do rosto, como linhas, olhos e bocas, e as linhas gerais do rosto são aprendidas por diferentes camadas ocultas (Figura 47).

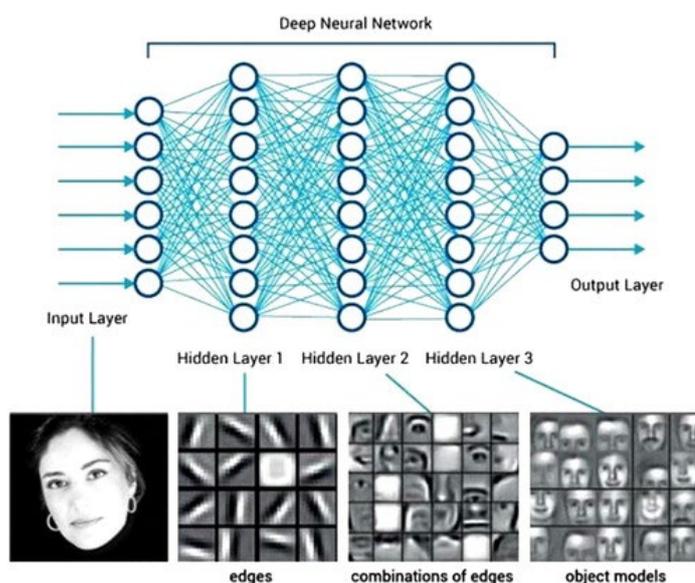


Figura 47 – Sistema *deep learning* desenvolvido para reconhecimento de rosto humano. Fonte MURALI, 2018

#### 6.4.6. Discussão sobre o teste de desempenho com a base ChestXpert

O teste de desempenho com a base ChestXpert, foi realizado para verificar o desempenho da WCNN quando comparada com outra rede CNN. O teste apresentou uma AUC de 85,96% para o modelo WCNN e 80,65% para a rede *DenseNet121*, o que representa que a WCNN é 6,584% mais eficiente que a *DenseNet 121*.

Apesar do experimento ter utilizado uma base não balanceada da ChestXpert, ou seja, sem utilização de qualquer técnica de *data augmentation*, a WCNN apresentou um bom resultado em comparação com a rede DenseNet 121.

Este teste teve o intuito de analisar a WCNN no desempenho de classificação de diversas doenças simultaneamente, obtendo resultado satisfatório. Novos testes desta natureza não foram realizados porque a resolução espacial da base (320×320), o formato da imagem (JPG) e sua profundidade (8 bits) são muito diferentes da realidade encontrada no processamento de imagens médicas reais, que, dentre outras propriedades, usa com imagens de 16 *bits*.

#### 6.4.7. Discussão sobre o teste de resizing com a base DRIVE

Após replicar os passos seguidos pelo autor da tese (SANTOS, 2022), substituindo a Unet pela WCNN e diante dos resultados obtidos por ambos, pode-se citar os seguintes pontos positivos da WCNN, utilizando uma camada de entrada personalizada: *i*) a *Wave layer* pode receber imagens médicas de diferentes resoluções espaciais, pois ela as recorta em fragmentos de mesmo tamanho (no caso deste teste, o tamanho do recorte é 188×188), mantendo, assim, a integridade das imagens médicas, sem utilizar técnicas de *data augmentation*, que podem ocasionar a perda de informações; e, *ii*) possibilidade de utilizar outras técnicas de extração de características além da transformada *wavelet* utilizada no experimento. Com relação aos pontos que precisam ser melhorados, citamos: *i*) a necessidade de selecionar a família *wavelet* mais adequada para cada estudo; *ii*) necessidade de selecionar os três melhores coeficientes para cada caso em estudo. Por exemplo, os melhores coeficientes para o teste com a base DRIVE a *wavelet* mãe foi a db1 e os coeficientes foram aproximada, horizontal e vertical; e, *iii*) *resize* do *ground truth*, ocasionando perda de informações relevantes.

Os resultados obtidos neste reforçam a justificativa da tese, com o fato que o redimensionamento de imagens médicas incorre na perda de informações relevantes; que foi exatamente o que aconteceu neste experimento, ao impossibilitar a reconstrução dos vasos mais delgados, ocasionada pelo redimensionamento dos respectivos *ground truth* das imagens.

#### 6.4.8. Arquitetura do modelo resultante da ablação

As seções anteriores descreveram os testes de ablação e apresentaram as respectivas discussões, e é com base nos conhecimentos gerados por esse conjunto de informações, que foi possível, identificar os componentes e suas inter-relações de dependência, que permitiram montar o quadro completo do que viria a ser a arquitetura primordial da WCNN, conforme ilustrado pela Figura 48. Essa arquitetura é subdivida em três grandes estágios: *wave layer*, *feature extraction* e *classification*.

O estágio *Wave layer* materializa a ideia de criar uma camada customizada para tratar as imagens de entrada da rede convolucional.

O estágio *Features extration* agrega um conjunto de 4 *Convolutional layers* e *pooling layers* utiliza o *max pooling*; com o objetivo combinado de extrair recursos de alto nível e reduzir as dimensões dos dados.

O estágio *Classification* reúne quatro *full connected layers*, compostas por 256, 128, 64 e 32 neurônios, respectivamente. Além disso, uma função de ativação sigmóide é vinculada à última camada *full connected layer*.

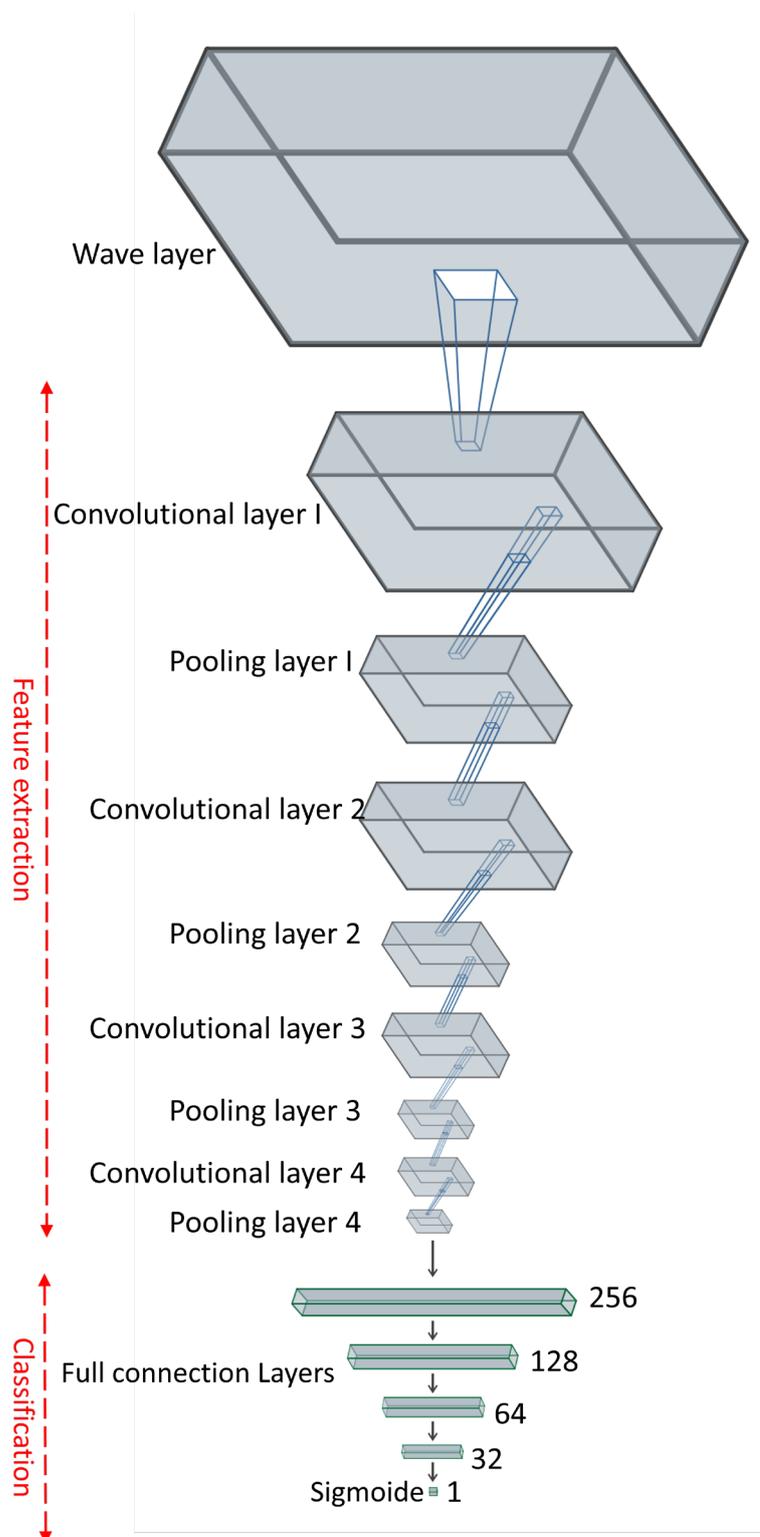


Figura 48 – Arquitetura da WCNN padrão

## 7. O Modelo WCNN

Como continuação do desenvolvimento dos trabalhos publicados e o estudo de ablação, apresentados no Capítulo 4, Capítulo 5 e Capítulo 6, respectivamente, surgiu, então, a proposição de um modelo com o objetivo de suprir alguns desafios enfrentados na primeira rede, como redimensionamento da imagem de entrada. Além da necessidade de tornar o modelo mais generalista, que pudesse ter bons desempenhos em vários tipos de aplicações.

### 7.1. Criação do modelo WCNN

Como mencionado anteriormente, o modelo proposto neste trabalho é baseado na arquitetura de CNNs, que foram propostas para avaliar dados de imagem. O nome vem do operador de convolução, uma maneira fácil de fazer operações complexas usando o *kernel* de convolução (RAVI, WONG, *et al.*, 2016). Muitas variações de CNN já foram propostas, como *AlexNet* (KRIZHEVSKY, SUTSKEVER e HINTON, 2012), *Clarifai* (ZEILER e FERGUS, 2014), *GoogLeNet* (SZEGEDY, LIU, *et al.*, 2015), cuja arquitetura padrão é ilustrada no lado esquerdo da Figura 49.

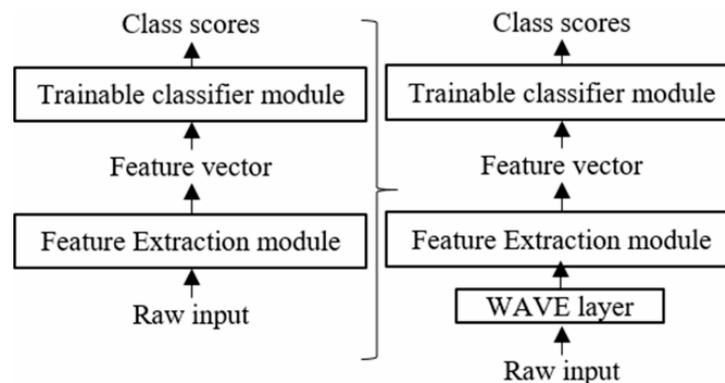


Figura 49 – A Estrutura de uma CNN é realizada com dois módulos: um extrator de recursos e um classificador treinável (adaptado de LECUN, BOTTOU, *et al.*, 1998)

Para o modelo WCNN, foi desenvolvido um mecanismo para entrada das imagens na rede, que chamado de *wave layer*, cujo diagrama esquemático que pode ser visto no lado direito da Figura 49. O diagrama ilustrado pela Figura 50 apresenta a metodologia utilizada nesta tese, incluindo os requisitos, entradas e saídas de cada fase da criação/utilização do modelo WCNN.

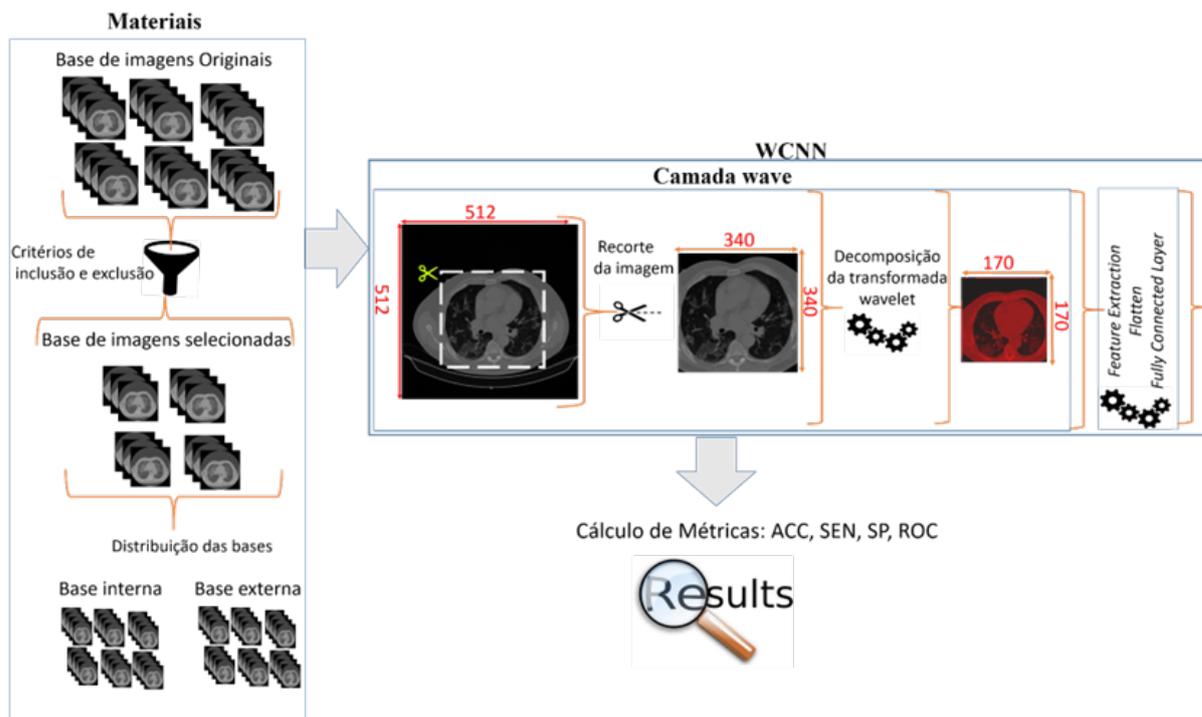


Figura 50 – Diagrama ilustrativo da metodologia, fases e etapas da criação do modelo WCNN

A metodologia desenvolvida para este trabalho é composta por três fases, a saber: *i*) Criação das Bases de Imagem, *ii*) Criação do Modelo; e, *iii*) Validação e Interpretação dos Resultados.

A fase de *Criação das Bases de Imagem* se inicia com a seleção dos repositórios públicos e/ou privados que contiverem as imagens que serão classificadas. Feito isto, as imagens dos repositórios selecionados são semiestruturadas nas categorias de interesse da pesquisa, e submetidas a critérios de inclusão e exclusão, os quais decidirão: *i*) quais delas comporão as bases internas e externas; e, *ii*) percentual das imagens das bases empregadas nas etapas de treinamento, teste e validação do modelo.

Já na fase de *Criação do Modelo WCNN*, as imagens selecionadas na fase anterior são processadas pela *wave layer*, que as recorta convenientemente para a extração dos coeficientes aproximada, vertical e diagonal utilizados no treinamento do modelo.

Por fim, na fase de *Validação do Modelo*, o cálculo das métricas *Accuracy*, *Recall/Sensitivity* e *Specificity* aliado à criação da matriz de confusão e à plotagem da Curva ROC dão condições para a validação e interpretação dos resultados nas bases interna e externa.

Nas próximas seções, as fases *i*) *Criação das Bases de Imagem*, *ii*) *Criação do Modelo*; e, *iii*) *Validação e Interpretação dos Resultados*. Serão descritas em detalhes, em termos de seus requisitos, etapas e passos. As próximas subseções descrevem o ambiente tecnológico e a visão arquitetural das camadas do modelo WCNN.

### 7.1.1 Ambiente tecnológico

A preparação de ambiente tecnológico computacional envolveu tanto a escolha da linguagem de programação, bibliotecas e *hardware* utilizados. O trabalho foi desenvolvido em linguagem Python utilizando biblioteca Tensorflow (CHOLLET, 2016). O equipamento utilizado foi notebook cuja configuração é processador Intel I7-8750H, CPU de 2,21GHz, 16,0 GB de RAM e placa de vídeo GeForce GTX 1060 com Max -Q.

### 7.1.2 Wave layer

Como a *Wave layer* é centrada nos conceitos de transformada *wavelet*, a criação desta camada dependeu de decisões arquiteturais centradas na: *i*) escolha da *wavelet* mãe, *ii*) definição do respectivo nível de decomposição; e, *iii*) na seleção dos coeficientes mais relevantes da transformada *wavelet* para o propósito do modelo. A *wavelet* mãe escolhida foi Coifltes 5, o nível de decomposição foi definido como um para reduzir perda de informações relevantes e os coeficientes selecionados foram *aproximada*, *vertical* e *diagonal*, vale ressaltar que, para cada aplicação, os coeficientes podem ser alterados, de acordo com a significância dos dados para tal aplicação, isto pode ser facilmente decidido utilizando análise estatística e avaliando a significâncias de cada coeficiente no problema.

Uma vez tomadas estas decisões, o funcionamento do modelo segue com a *Wave layer* recebendo a imagem e processando-a, primeiramente, fazendo sua normalização.

A imagem é normalizada para remover as variações causadas por possíveis diferenças entre os equipamentos, considerando a distribuição normal padrão onde a média  $\mu = 0$  e variância  $\sigma^2 = 1$ . A normalização é feita da seguinte forma: calcula-se a média  $\mu$  e a variância  $\sigma^2$ , como na Equação 49 e Equação 50, respectivamente, da imagem  $I$ , que é formada por  $m$  linhas e  $n$  colunas e denotada por  $I_{0,0}, I_{0,1}, \dots, I_{m,n}$  (AGGARWAL e OTHERS, 2018). Assim,  $I_{Normalized}$  é calculada conforme Equação 51.

$$\mu = \frac{\sum_1^m \sum_1^n I_{i,j}}{m*n} \quad \text{Equação 49}$$

$$\sigma^2 = \frac{\sum_1^m \sum_1^n (I_{i,j} - \mu)^2}{m*n} \quad \text{Equação 50}$$

$$I_{Normalized} = \frac{I_{i,j} - \mu}{\sigma} \quad \text{Equação 51}$$

Após a normalização e recorte, tem-se uma imagem que mostra os aspectos relevantes para a classificação. A partir de então, a imagem é processada pela transformada *wavelet*, gerando os coeficientes que substituirão os canais da imagem digital, considerando modelos de espaços de imagens como *Red, Green e Blue* (RGB). A decomposição das imagens por *wavelet* também assume um espaço de três canais, em que o canal R recebe o coeficiente de aproximação, o canal G recebe o coeficiente vertical e o canal B recebe o coeficiente diagonal, formando a *output of decomposition*.

A ablação do modelo, que foi detalhada no capítulo 6, definiu os parâmetros seguintes:

### 7.1.3 Extraction layer

1. A escolha de camadas convolucionais foi feita considerando o resultado de um teste de ablação específico, considerando 1, 2, 3, e 4 camadas, cuja descrição detalhada encontra-se nas subseções **6.1.4. Teste das camadas convolucionais** e **6.4.1 Discussão sobre as famílias wavelets discretas e coeficientes**. Este teste mostrou que a melhor configuração é composta por quatro camadas.
2. A escolha da técnica de *pooling* foi feita considerando o resultado de um teste de ablação específico, considerando *min pooling*, *avg pooling* e *max pooling*, cuja descrição detalhada encontra-se nas subseções **6.1.5 Teste de Pooling** e **6.4.4 Discussão sobre as técnicas de Pooling**. Este teste mostrou que a melhor técnica é a *max pooling*.

### 7.1.4 Fully connected layer

1. A escolha do otimizador foi feita considerando o resultado de um teste de ablação específico, considerando SGD, RMSProp e ADAM, cuja descrição detalha encontra-se nas subseções **6.1.3 Teste das técnicas de otimização** e **6.4.2 Discussão sobre as técnicas de otimização**. Este teste mostrou que o melhor otimizador, para o contexto do WCNN, é o ADAM, com taxa de aprendizagem de  $\alpha = 0,001$ .
2. A escolha das funções de ativação foi feita considerando trabalhos da literatura. Eles recomendam o uso da função sigmóide como função de ativação da saída da *fully connected layer* e o uso da ReLU para camadas interdiárias da CNN (AGGARWAL e OTHERS, 2018, WANI, BHAT, *et al.*, 2020).

### 7.1.5 Treinamento do WCNN

1. Definir para o primeiro treinamento, pesos inicializados aleatoriamente.
2. Fazer a distribuição das bases de treinamento, teste e validação, seguindo uma relação de proporção da literatura, como por exemplo: 70 % treinamento, 15% teste, 15% validação.
3. Treinar a WCNN de acordo com seus conjuntos de treinamento, teste, salvando os melhores pesos encontrados em cada fase de teste.

## 7.2.Criação das Bases de Imagens

A primeira fase da metodologia é responsável pela criação e preparação das bases de imagens que serão utilizadas. Para isto, é necessário, primeiramente, selecionar os repositórios públicos e/ou privados cujas imagens serão classificadas e assegurar sua anonimização por questões de privacidade dos pacientes. A partir daí as imagens pré-selecionadas são semiestruturadas nas categorias de interesse da pesquisa e analisados de acordo com critérios de inclusão e exclusão previamente estabelecidos. Estes critérios, em última instância, decidirão quais imagens serão efetivamente utilizadas. A seguir, as bases internas e externas serão formadas e distribuídas de acordo com o percentual determinado para as etapas de treinamento, teste e validação do modelo.

### 7.2.1. Formato da base

A primeira tarefa desta fase é assegurar que as imagens das bases estejam codificadas em mesmo formato ou formatos compatíveis, preferencialmente em formato DICOM (16 bits) ou *Medical Imaging Data Structure* (MIDS) (16 bits), pois ambas permitem ao usuário organizar os metadados das imagens numa estrutura de diretórios padrão clara e fácil de manipular. Ao utilizar estes formatos compatíveis, é possível haver interoperabilidade de imagens entre equipamentos de fornecedores distintos e diferentes plataformas (MOREIRA, AMARAL, *et al.*, 2012). Com isso, passa-se a ter a vantagem da conservação da qualidade da imagem mesmo após o compartilhamento, devido à preservação da profundidade de pixels, dimensões e números de *bits* por *pixel* (SAEZ e OTHERS, 2009; LEVIN, 2011).

### 7.2.2. Anonimização das imagens

Após a compatibilidade do formato das imagens ter sido garantido, é preciso garantir sua anonimização, caso esta informação tenha sido disponibilizada, casos do padrão DICOM. Isto é feito atribuindo-se um número identificador correlacionado ao identificador único do

prontuário do paciente no hospital, o que elimina o risco de violação à privacidade dos pacientes. Uma vez que o estudo trata de casos retrospectivos, de uma população restrita de indivíduos que já realizaram exames radiológicos (radiografias e Tomografias Computadorizadas) e foram diagnosticados, as informações deles já se encontram protegidas.

### 7.2.3. Critérios de inclusão

Dentre os critérios que levam uma imagem a ser incluída na base, tem-se:

- Inserir nas bases, apenas imagens relacionadas à doença investigada.
- Inserir imagens com mesma resolução de contraste, de 16 *bits*
- Para imagens de TC de tórax, selecionar aquelas com resolução espacial maior ou igual a 512×512 *pixels*.
- Para imagens de radiografias de tórax, selecionar aquelas com resolução espacial maior ou igual a 1000×1000 *pixels*.
- Inserir apenas imagens de pacientes com idade adulta, ou seja, maiores que 18 anos.

### 7.2.4. Critérios de exclusão

Dentre os critérios que levam uma imagem a ser excluída, tem-se:

- Em imagens de TC de toráx, descartar 40% do total de fatias de cada exame, sendo 20% das fatias iniciais e 20% das fatias finais<sup>8</sup>. Esta seleção é importante para que a base seja mais consistente e focada na área a ser estudada (no caso de COVID-19, o pulmão), pois as fatias iniciais e finais contêm poucas informações relevantes sobre o órgão;
- Imagens que não sejam do órgão/tecido em estudo;
- Imagens de pacientes com idade menor que 18 anos;
- Imagens de cuja resolução de contraste seja inferior a 16 bits;
- Imagens que não estejam na posição adequada para o estudo; e,
- Excluir imagens que não atendam os critérios de inclusão.

### 7.2.5. Definição de bases internas e/ou externas

Finalizado a aplicação de critérios de inclusão e exclusão, de acordo com M. Roberts, *et al.* (ROBERTS, DRIGGS, *et al.*, 2021), é necessário estabelecer como a validação intrínseca do algoritmo será feita interna e externamente. Para isto, ele sugere a política de uso de bases

---

<sup>8</sup> Este critério foi estabelecido seguindo orientação do radiologista que nos apoiou.

internas e externas para validar o modelo, porque isto permite obter evidências sobre a generalização do algoritmo. Para isso, são criadas: *i*) bases internas, nas quais as imagens que serão utilizadas na validação são da mesma fonte que as imagens usadas no treinamento; e/ou, *ii*) bases externas, em que as imagens utilizadas na validação sejam de fontes diferentes, uma vez que elas não participaram do balanceamento dos pesos da rede.

### 7.2.6. Definição do percentual para treinamento, teste e validação

Uma vez definidas as bases interna e externa, elas são subdivididas em conjuntos de imagens para treinamento, teste e validação. A cada um deles, atribui-se um percentual do total de imagens, de acordo com propostas da literatura, como 70 % para treinamento, 15% para teste e 15% para validação.

### 7.3. Validação do modelo e apresentação dos resultados

O objetivo final dos modelos de classificação é prever a ocorrência de um evento com base em ocorrências passadas. As técnicas para se fazer isso têm evoluído cotidianamente, porém, de forma geral, elas se valem de um conjunto de dados como entradas e seus respectivos resultados esperados para este conjunto de dados (*ground truth*). Estas informações são usadas para treinar modelos para a predição de novos resultados quando novas entradas surgirem no futuro.

Porém, estes resultados de nada adiantariam se não pudessem ser validados objetivamente, e isto é feito mediante o cálculo de métricas distintas, as quais avaliam consistentemente os resultados gerados pelos modelos. Neste trabalho, as métricas utilizadas para avaliar o desempenho do modelo são:

- *Accuracy* (Acc), taxa de classificação correta em relação ao total de elementos;
- *Recall/Sensitivity* (Sen), taxa de verdadeiros positivos;
- *Specificity* (Sp), taxa de verdadeiros negativos.

Esta métricas são usadas para avaliar o desempenho da classificação (RUUSKA, HÄMÄLÄINEN, *et al.*, 2018; SKANSI, 2018; KHATAMI, KHOSRAVI, *et al.*, 2017). As métricas são calculadas com base nos seguintes parâmetros:

- *True positive* (TP): quando o método diz que a classe é positiva, e ao verificar a resposta, vê-se que a classe era realmente positiva;
- *False positive* (FP): quando o método diz que a classe é positiva, mas ao verificar a resposta, vê-se que a classe era negativa;

- *True negative* (TN): quando o método diz que a classe é negativa, mas ao verificar a resposta, vê-se que a classe era realmente negativa;
- *False negative* (FN): quando o método diz que a classe é negativa, mas ao verificar a resposta, vê-se que a classe era positiva.

### 7.3.1. Matriz de confusão

A matriz de confusão é uma maneira visual para exibição da quantidade de TP, FP, TN e FN. Como exemplo da classificação de duas classes, uma matriz de confusão é apresentada na Tabela 36:

Tabela 36 – Exemplo de uma matriz de confusão

		Classificador detecta	
		<i>True</i>	<i>False</i>
<i>Real</i>	<i>True</i>	TP	FN
	<i>False</i>	FP	TN

Fonte: Adaptado de (SKANSI, 2018)

A partir da matriz de confusão é possível determinar as métricas (NARIN, KAYA e PAMUK, 2020; RUUSKA, HÄMÄLÄINEN, *et al.*, 2018; SKANSI, 2018; KHATAMI, KHOSRAVI, *et al.*, 2017):

- *Accuracy* (*Acc*) 
$$Acc = \frac{TN + TP}{(TN + TP + FN + FP)}$$
- *Recall/Sensitivity* (*Sen*) 
$$Se = \frac{TP}{(TP + FN)}$$
- *Precision* 
$$P = \frac{TP}{(TP + FP)}$$
- *Specificity* (*Sp*) 
$$Sp = \frac{TN}{(TN + FP)}$$
- *F1-Score* 
$$F1 = \frac{2 \times (precision * recall)}{(precision + recall)}$$

### 7.3.2. Curva ROC

Em estatística, uma curva *Receiver Operating Characteristics* (ROC) é um gráfico que ilustra o desempenho de um sistema classificador binário conforme seu limite de discriminação é variado. A análise ROC é frequentemente chamada de taxa de precisão ROC, uma técnica comum para julgar a precisão dos modelos de probabilidade padrão. Um espaço ROC é definido

por  $(1 - Sp)$  e  $Sen$  como eixos  $x$  e  $y$  respectivamente, que representam a *relação* entre verdadeiros positivos (benefícios) e falso positivo (custos).

Cada resultado ou instância de previsão de uma matriz de confusão representa um ponto no espaço ROC. O melhor método de previsão possível produziria um ponto no canto superior esquerdo ou coordenada  $(0,1)$  do espaço ROC, representando 100% de sensibilidade (sem falsos negativos) e 100% de especificidade (sem falsos positivos). O ponto  $(0,1)$  também é chamado de perfeita classificação. Além disso, a diagonal divide o espaço do ROC: os pontos acima da diagonal representam bons resultados de classificação e os pontos abaixo da linha representam resultados ruins (SHIRAZI, CHABOK e MOHAMMADI, 2018), conforme mostra a Figura 51(0).

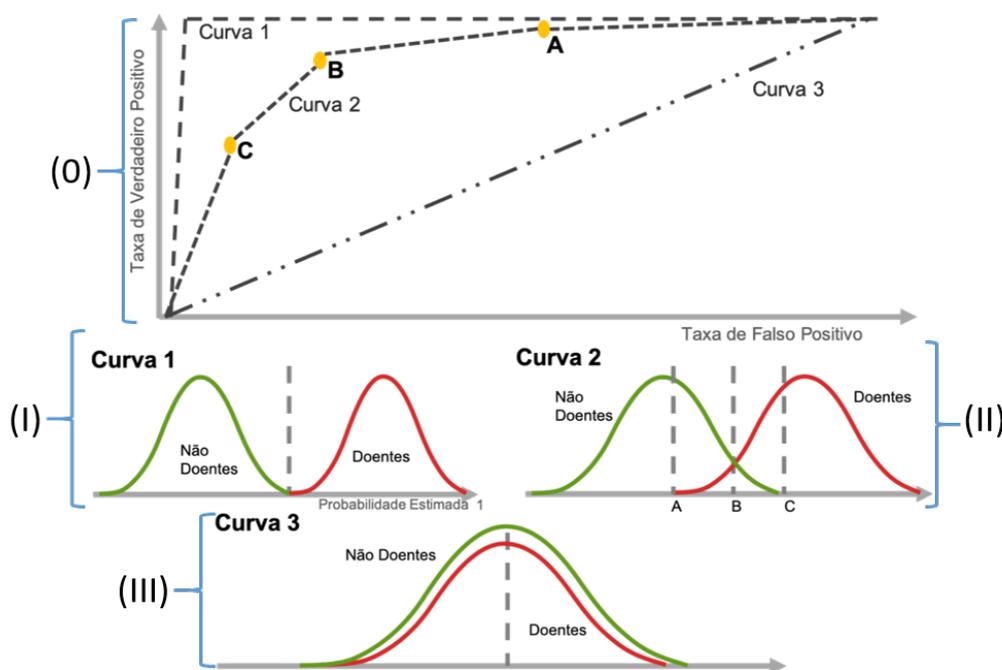


Figura 51 – O espaço ROC e suas interpretações ( PRATES, 2020)

As partes (I), (II) e (III), apresentadas na Figura 51, são exemplos de como se parecem as curvas ROC, para casos hipotéticos, de acordo com as curvas:

- A curva 1, Figura 51(I), representa um modelo perfeito. Esta seria a curva ideal, onde não há nenhum falso positivo; ou seja, a taxa de verdadeiro positivo é 100%.
- A curva 2, Figura 51(II), representa um modelo usual. Esta curva é a usualmente encontrada, onde o modelo indica alguns falsos positivos e falsos negativos.
- Por fim, a curva 3, Figura 51(III), representa um modelo errático. Esta curva é encontrada em modelos que fazem somente previsões aleatórias.

As métricas apresentadas serão utilizadas para avaliar tanto a validação interna como a externa.

## 8. Terceira Contribuição: Um novo modelo para classificação de imagens médicas de TC usando CNN e *wavelet*: um estudo de caso com COVID-19

---

“Este capítulo é baseado no artigo completo, aprovado e em revisão, no periódico *Multimedia Tools and Applications*, *Classificação Qualis CAPES 2019 A2*, de autoria do próprio autor desta tese.”

### 8.1. Introdução

Devido ao impacto da pandemia global do COVID-19, foram feitos esforços internacionais para simplificar o acesso de pesquisadores aos dados sobre a doença por meio de repositórios, como o 2019 *Novel Coronavirus Resource* (2019nCoV.R.). (WU, 2020) e o *National Center for Biotechnology Information* (NCBI) (SHERRY, WARD, *et al.*). Quanto mais acessível forem as informações, é mais provável que um conjunto de contramedidas médicas seja desenvolvido rapidamente para controlar a doença em todo o mundo, como aconteceu com outras doenças em outras ocasiões (DOS S RIBEIRO, VAN ROODE, *et al.*, 2018, SIMON, CLAASSEN, *et al.*, 2005, RIBEIRO, KOOPMANS e HARINGHUIZEN, 2018). Nesse contexto, a Tomografia Computadorizada (TC) tem sido realizada como modalidade de triagem de pacientes, pois permite a visualização detalhada da anatomia anormal (OZTURK, OZKAYA e BARSTUGAN, 2020, DAI, ZHANG, *et al.*, 2020).

Imagens de TC mostram semelhanças entre pacientes com COVID-19 e com outro tipo de pneumonia viral, como SARS e MERS. Ainda assim, a análise das imagens de TC deve ser feita por radiologistas experientes, que analisam, por vezes, milhares de cortes de imagem por paciente. Neste período de pandemia de COVID-19, a pressão por resultados e as limitações de tempo dos radiologistas são elementos que tornam as análises mais propensas a erros.

Como o uso de *deep learning* na pesquisa de imagens de radiologia não é novo, (CHEN, WU, *et al.*, 2020, MARTIN, HANSON, *et al.*, 2020, LECUN, BENGIO e HINTON, 2015, WANG, LIN e WONG, 2020), tão logo a pandemia foi declarada pela OMS em março de 2019, pesquisadores de todo o mundo iniciaram a criação de modelos de computador que processassem imagens radiológicas de pacientes com COVID-19 (OZTURK, OZKAYA e BARSTUGAN, 2020, DAI, ZHANG, *et al.*, 2020, CHEN, WU, *et al.*, 2020, ZHANG, XIE, *et al.*, 2020). Imagens de TC de tórax apresentam características comuns que podem mostrar um

padrão específico para COVID-19, porém, a análise visual consome muito tempo do radiologista e é sujeita a erros. Para acelerar a análise e diminuir a probabilidade de erro, parte-se da premissa que *deep learning* é eficaz na análise de grandes volumes de dados gerados por imagens de TC (MARTIN, HANSON, *et al.*, 2020, ZHANG, XIE, *et al.*, 2020, FAN, FENG, *et al.*, 2022, TER-SARKISOV, 2022, RANGARAJAN e RAMACHANDRAN, 2022, ÖZKAYA, ÖZTÜRK e BARSTUGAN, 2020, KOGILAVANI, PRABHU, *et al.*, 2022, CHEN, CHEN, *et al.*, 2021).

Neste artigo é descrita a proposta de um novo modelo para classificação de imagens baseado em *deep learning*, denominado WCNN. O objetivo é melhorar a diferenciação de imagens de pacientes doentes de COVID-19. Assim, as bases de imagens utilizadas neste trabalho são compostas por TC de tórax de pacientes com algum tipo de doença. Então, o conjunto de imagens foi separado em duas classes: classe pacientes com COVID-19, formando a base COVID-19+, e a classe com outras doenças pulmonares, infecciosas ou inflamatórias, como pneumonia, cardiomegalia, derrame pleural, atelectasia, consolidação, entre outras, formando a base de imagens COVID-19-.

O modelo WCNN foi desenvolvido utilizando abordagens CNN bem estabelecidas na literatura (KRIZHEVSKY, SUTSKEVER e HINTON, 2012, ZEILER e FERGUS, 2014, SZEGEDY, LIU, *et al.*, 2015). No entanto, o modelo proposto evita pré-processamento e uso de técnica de redimensionamento da imagem. Para isto, foi desenvolvida uma camada adicional à rede convolucional, a qual foi denominada *wave*. Esta camada utiliza a transformada *wavelet* para decompor a imagem e extrair suas características, sendo responsável pelo pré-processamento e geração da imagem de saída que será processada pelas demais camadas do modelo.

## 8.2. Trabalhos Relacionados

Tão rapidamente quanto a pandemia se espalhou, governos, organizações supranacionais, institutos de pesquisa, universidades e corporações mobilizaram quantidades inéditas de recursos humanos e financeiros para debelar a crise. Num ápice, tanto SARS-CoV-2 quanto COVID-19 foram alçadas ao cume do interesse da comunidade científica mundial de pesquisas nas mais diversas áreas: como medicina, infectologia, bioquímica, tecnologias da informação, matemática aplicada, inteligência artificial, entre outras. Este trabalho de pesquisa foi conduzido neste contexto, motivado pela urgência de debelar a pandemia; e com a iniciativa prévia de outros pesquisadores, cujos resultados são apresentados e discutidos nesta seção.

O primeiro trabalho Fan *et al.* analisado propõe um algoritmo baseado em transformadas

e CNN para reconhecimento de imagens de TC (FAN, FENG, *et al.*, 2022). Os autores apresentam uma solução com duas ramificações: modelo *Trans-CNN* e módulo *Transformer*. O modelo *Trans-CNN* usa a capacidade de extração de recursos locais da CNN e da extração de recursos globais do *Transformer*. A pesquisa consistiu em 194.922 imagens de TC de tórax, de 3.745 pacientes, de 0 a 93 anos, extraídas do banco de dados COVIDx-CT. As imagens incluem pacientes *i)* saudáveis, *ii)* doentes com COVID-19; e, *iii)* doentes com outras doenças pulmonares. A base foi expandida por rotações de 15°, 45°, 90° e 180°. Os valores obtidos para acurácia, sensibilidade e especificidade foram de 0,9673, 0,9776 e 0,9602, respectivamente.

O modelo *COVID-CT-Mask-Net* também utiliza CNN dos autores Ter-Sarkisov *et al.* (TER-SARKISOV, 2022) e é executado em duas etapas: *i)* a rede *Mask R-CNN* é treinada para localizar e detectar regiões de lesões *Ground Glass Opacity* nas imagens de CT; e *ii)* as imagens das lesões são mescladas para classificar a imagem de entrada. O experimento usou 3.000 imagens de TC de tórax da base COVIDx-CT, cujos pacientes poderiam ser *i)* sadios, *ii)* doentes com COVID-19; ou, *iii)* doentes com outras infecções pulmonares. Os valores da métrica acurácia, sensibilidade e especificidade foram calculados e resultaram em 0,9673, 0,9776 e 0,9602, respectivamente.

O trabalho de Rangarajan *et al.* descreve um modelo híbrido que combina *queezeNet* e *ShuffleNet* utiliza 1.252 imagens de CT COVID-19+ e 1.230 imagens COVID-19- da base pública SARS-COV-2 Ct-Scan. As imagens foram coletadas de pacientes reais de hospitais de São Paulo, Brasil (RANGARAJAN e RAMACHANDRAN, 2022). Os dados foram expandidos pela realização de operações aleatórias de *i)* rotação de  $\pm 5^\circ$ ; *ii)* alteração do valor de intensidade de  $\pm 20$ ; e, *iii)* cisalhamento a  $\pm 20^\circ$ . Além das operações aleatórias, também foi efetuado nas imagens *i)* desfoque; *ii)* inversão; e, e *iii)* redimensionando para resolução espacial de  $224 \times 224$  pixels. Os respectivos resultados para acurácia, sensibilidade e especificidade foram 0,9781, 0,9615 e 0,9608.

Outro exemplo de pesquisa em que foram empregadas redes neurais distintas, que foi descrita no artigo Özkaya *et al* (ÖZKAYA, ÖZTÜRK e BARSTUGAN, 2020). O modelo proposto por Özkaya *et al* combina redes *VGG-16*, *GoogleNet* e *ResNet-50*, cujo objetivo era detectar COVID-19 na fase inicial. Ele obteve acurácia de 0,9827, sensibilidade de 0,9893 e especificidade de 0,9760. Foram utilizadas 150 imagens de TC de tórax, pertencentes à *Società Italiana di Radiologia Medica e Interventistica*. Elas deram origem a 3.000 imagens, agrupadas no Subconjunto-1, chamado “Covid-19”, e Subconjunto-2, “No findings”. A resolução das imagens dos subconjuntos é de  $16 \times 16$  e  $32 \times 32$ , respectivamente.

Um destaque para a pesquisa de Kogilavani *et al.* que compara o desempenho de diversas arquiteturas de redes convolucionais se destaca (KOGILAVANI, PRABHU, *et al.*, 2022). Ela envolve redes *VGG16*, *DenseNet121*, *MobileNet*, *NASNet*, *Xception* e *EfficientNet*. O estudo usou imagens de TC de tórax obtidas de *Kaggle*, sendo 1.958 de pacientes COVID-19+ e 1.915 de COVID-19-. As bases imagens foram expandidas, por redimensionamento, para resolução espacial de  $224 \times 224$ . O modelo foi treinado com 70% das imagens, validado com 15% e testado com os outros 15%. Destas arquiteturas, a *VGG16* apresentou os melhores resultados, com acurácia de 0,9768, sensibilidade de 0,9579 e especificidade de 0,9971.

Outro trabalho relacionado descreve a criação de um aplicativo para detecção de pneumonia causada por COVID-19 mediante análise de CT de alta resolução dos autores Chen *et al.* (CHEN, WU, *et al.*, 2020). O aplicativo foi criado pela equipe do Hospital Renmim, da Universidade de Wuhan, China. O modelo utiliza arquitetura derivada de *UNet++*. Seu desempenho foi mensurado usando 46.096 imagens anônimas de 106 pacientes do hospital, agrupados em dois grupos. O primeiro grupo, com 51 pacientes COVID-19+; e, o segundo, utilizado como grupo de controle, com 55 pacientes COVID-19-. Além disso, os autores utilizaram, retrospectivamente, as imagens de 27 pacientes atendidos antes do início do projeto para comparar a eficácia do diagnóstico feito por especialistas com a eficácia obtida pelo aplicativo. O resultado de acurácia, sensibilidade e especificidade do aplicativo foi, respectivamente, 0,9524, 1,0000 e 0,9355. Considerando as imagens dos 27 pacientes prévios, a acurácia foi de 0,9885%, a sensibilidade 0,9434% e a especificidade 0,9916%. Dessa forma, os autores mostraram que o desempenho do aplicativo é compatível com os resultados obtidos por especialistas médicos.

Já o modelo COVID-19-CNN combina o uso de CNNs previamente treinadas dos autores Chen *et al.* (CHEN, CHEN, *et al.*, 2021). Seu treinamento e teste de desempenho usaram imagens de 405 pacientes COVID-19+ e de 397 pacientes COVID-19-. Foram empregadas 612 imagens para treinamento, 99 para validação e 91 para teste. O banco de dados não foi expandido, mas a imagem foi redimensionada para resolução espacial de  $224 \times 224$ . O modelo COVID-19-CNN teve acurácia de 0,9670, sensibilidade de 0,9780 e especificidade de 0,9556.

De acordo com as características das pesquisas que estão no estado da arte, as principais contribuições desta pesquisa incluem: *i*) sistematização da utilização, concomitante, de bancos de imagens públicos e privados, nas fases de treinamento e teste da rede; *ii*) criação de uma base de testes externa para validação independente do modelo; *iii*) aplicação de critérios objetivos para inclusão e exclusão de imagens dessas bases de dados; *iv*) utilização de imagens alocadas em 16 *bits*, contendo as informações necessárias para a caracterização da doença; e, *v*) WCNN não utiliza aumento de dados ou redimensionamento das imagens para classificar, as

imagens COVID-19+ (FAN, FENG, *et al.*, 2022), RANGARAJAN e RAMACHANDRAN, 2022, KOGILAVANI, PRABHU, *et al.*, 2022, CHEN, CHEN, *et al.*, 2021).

### 8.3. Wavelet

Nesta seção, é apresentada a teoria de transformada *wavelet*, inspirada em pequenas ondas (*wavelets*) de frequência variada e duração limitada. Os trabalhos relacionados mostraram que o uso de redes convolucionais é comum em modelos de classificação de imagens. A utilização de *Discrete Wavelet Transform* (DWT) também não é incomum, porém, o uso que este trabalho fez da DWT deu, à WCNN, características que impactaram seu desempenho positivamente. A DWT apresenta uma sequência de números que mostra uma certa função contínua. Além de sua estrutura eficiente e intuitiva para representação e armazenamento de imagens multiresolução, a transformada *wavelet* fornece uma visão das características espaciais e de frequência de uma imagem (GUO, SEYED MOUSAVI, *et al.*, 2017; RAFAEL, 2006; JANSEN, 2012; MERRY, 2005). Seja a imagem  $f(a, b)$ , a transformada discreta *wavelet* (DWT) dessa imagem é definida pela Equação 52, como (GUO, SEYED MOUSAVI, *et al.*, 2017; RAFAEL, 2006; JANSEN, 2012; MERRY, 2005):

$$Wf(a, b) = \int_{-\infty}^{+\infty} f(t)\psi_{a,b}(t)dt \quad \text{Equação 52}$$

Para um sinal discreto de  $n$  pontos, a integral da DWT pode assumir a forma de somatório, como na Equação 53:

$$Wf(a, b) = \sum_{t=0}^{N-1} f(t)\psi_{a,b}(t) \quad \text{Equação 53}$$

A função *wavelet*  $\psi_{a,b}(t)$  é derivada da função  $\psi(t)$ , pela a transformação mostrada na Equação 54:

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right) \quad \text{Equação 54}$$

sendo  $a \in R^+$ ,  $b \in R$  e  $\psi\left(\frac{t-b}{a}\right)$  é a base da *wavelet*, onde  $b$  representa a translação da *wavelet* e  $a$  é o parâmetro de escala associado à largura.

Existem várias funções que podem ser escolhidas para  $\psi(t)$ , chamada *wavelet* mãe, dentre as quais tem-se *Daubechies*, *Symlets*, *Coiflet*, etc. As versões escalonadas e deslocadas da *wavelet* mãe correspondem a filtros *bandpass* com larguras de banda e durações de tempo diferentes. A transformada *wavelet* executa um passo de transformação em cada linha, produzindo uma matriz; em que o lado esquerdo contém os *down-sampled lowpass coefficients* (L),

de cada linha, e o lado direito contém os *highpass coefficients* (H). Em seguida, executa-se um passo a cada coluna, resultando em quatro tipos de coeficientes, conforme Figura 52 (GUO, SEYED MOUSAVI, *et al.*, 2017; RAFAEL, 2006; JANSEN, 2012; MERRY, 2005):

- Coeficientes que resultam de uma convolução com *highpass* em ambas as direções (HH) representam características diagonais da imagem;
- Coeficientes que resultam de uma convolução com *highpass* nas colunas após uma convolução com *lowpass* nas linhas (HL) correspondem às características horizontais da imagem;
- Coeficientes originados de filtros *highpass* nas linhas seguido por filtro *lowpass* nas colunas (LH) correspondem às características verticais da imagem;
- Coeficientes originados de filtros *lowpass* em ambas as direções (LL) correspondem às características da aproximada da imagem.

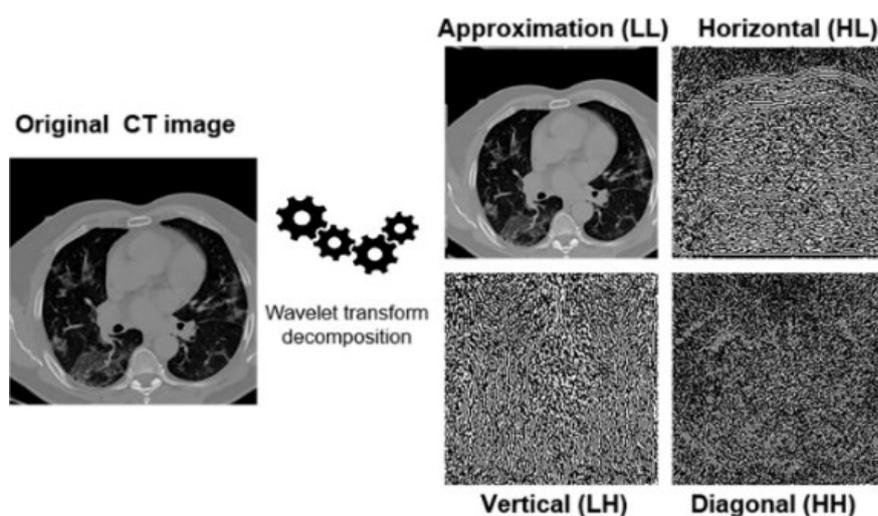


Figura 52 – Diagrama geral da transformada *wavelet*

### 8.3.1. *Wavelet* mãe e nível de decomposição

A decisão de associar uma transformada *wavelet* da família *Coiflets 5* à WCNN foi tomada com base no trabalho de (DA COSTA JUNIOR e PATROCINIO, 2019, DA COSTA JUNIOR, 2019) em que *wavelets* foram utilizadas *wavelets* para reduzir ruído em imagens de radiografias de mama densa. Nesse trabalho, foram testadas as transformadas *wavelet* das famílias *Daubechies*, *Symlets*, *Coiflets*, *Fejer-Korovkin* e *dMeyer*, e a família *Coiflets 5* foi a que apresentou os melhores resultados de redução de ruído.

Para garantir que *Coiflets 5* também seja a família mais adequada para o contexto desta pesquisa, sua escolha foi feita de acordo com o resultado de um teste de ablação específico, o

qual considerou as famílias *Biorthogonal*, *Coiflets*, *Daubechies*, *Discrete FIR approximation of Meyer*, *Reverse biorthogonal* and *Symlets*. A descrição detalhada dos testes encontra-se nas subseções **6.1.2 Teste das famílias wavelet discretas e coeficientes** e **6.4.1 Discussão sobre as famílias wavelets discretas e coeficientes**. Este teste mostrou que a família mais adequada, para o contexto do WCNN, é a *Coiflet 5*. Além disto, foi definido um nível de decomposição para evitar perda de informações necessárias para a classificação da imagem.

### 8.3.2. Análise dos coeficientes da decomposição

A análise dos coeficientes da decomposição utilizou 1.000 imagens, sendo 500 para a base de imagem COVID-19+ e 500 para a COVID-19-. Como os dados são heterogêneos, independentes e não paramétricos, o teste de Fiedman foi usado para analisar a significância entre os grupos (ZIMMERMAN e ZUMBO, 1993, ISHITAKI, ODA e BAROLLI, 2016) e para averiguar se eles apresentavam valores estatisticamente similares entre si, com relação aos coeficientes aproximada, horizontal, vertical e diagonal; resultantes da decomposição *wavelet* das imagens. Foi utilizado o *software* de análise estatística BioEstat, na versão 5.3, para entrada de dados e testes de hipótese e significância, considerando  $\alpha = 0,05$ . O desvio padrão dos coeficientes foi utilizado como parâmetro para teste de significância entre as bases COVID-19+ e COVID-19-, e foi evidenciada a existência de diferenças estatísticas significativas para os coeficientes **aproximada, vertical e diagonal**.

Para garantir que os coeficientes **aproximada, vertical e diagonal** também sejam a escolha mais adequada para o contexto desta pesquisa, sua escolha foi reforçada com um teste de ablação específico, o qual testou outras combinações de coeficientes. A descrição detalhada dos testes encontra-se nas subseções **6.1.2 Teste das famílias wavelet discretas e coeficientes** e **6.4.1 Discussão sobre as famílias wavelets discretas e coeficientes**. Este teste mostrou que a combinação dos coeficientes aproximada, vertical e diagonal é a mais adequada, para o contexto do WCNN.

## 8.4. Workflow

Modelos de classificação baseado em *deep learning* são parte integrante de *workflows* para detecção de padrões em imagens. O *workflow* utilizado nesta pesquisa descreve etapas para criação da base, execução do modelo, consolidação do resultado e cálculo das métricas. A Figura 53 ilustra o *workflow* do qual a execução do modelo WCNN é parte.

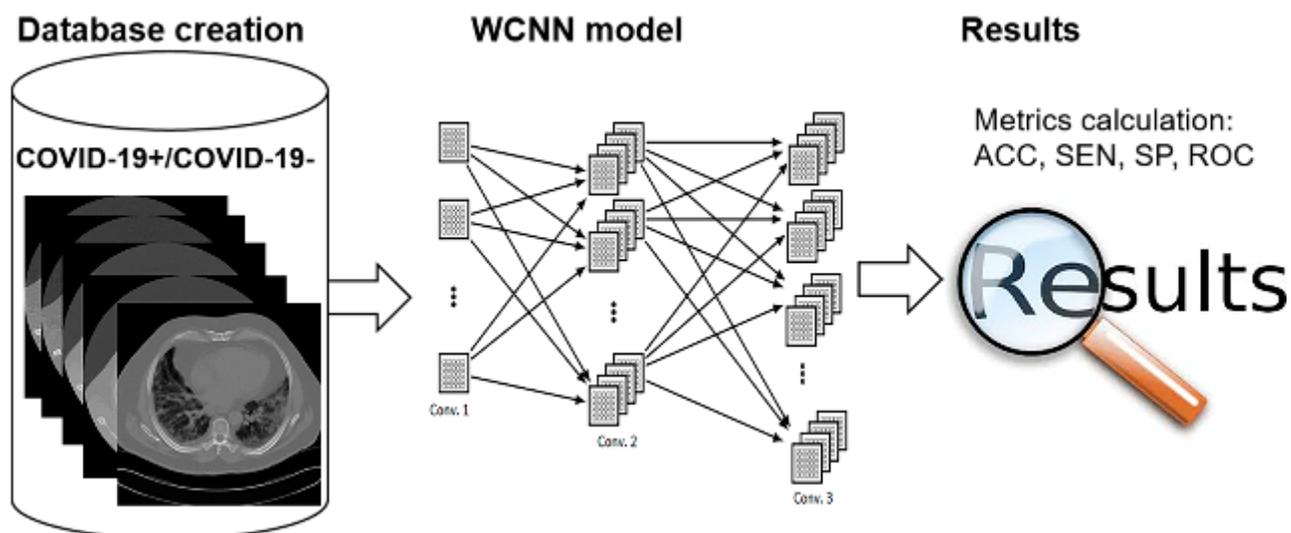


Figura 53 – Visão geral do *workflow* e o modelo WCNN

Nas seções subsequentes são descritas as etapas do *workflow*, começando pela criação da base de imagens.

## 8.5. Criação da Base de Imagens

Esta seção apresenta os critérios de inclusão e exclusão de imagens e os *datasets* utilizados na pesquisa.

### 8.5.1. Critérios de inclusão e exclusão

O primeiro passo para a criação da base de imagens foi a definição dos critérios de inclusão e exclusão, das imagens nas bases COVID-19+ e COVID-19-, o que foi feito com o auxílio de um médico radiologista.

Os critérios de inclusão são:

- A matriz de reconstrução tomográfica formada por  $512 \times 512$  *pixels*;
- Pacientes com idade maior que 18 anos; e,
- Imagens de pacientes com teste positivo para COVID-19.

Os critérios de exclusão são:

- Em imagens de TC de torác, descartar 40% do total de fatias de cada exame, sendo 20% das fatias iniciais e 20% das fatias finais. Esta seleção é importante para que a base seja mais consistente e focada na área a ser estudada (no caso de COVID-19, o pulmão), pois as fatias iniciais e finais contêm poucas informações relevantes sobre o órgão; conforme ilustrado na Figura 54.
- Imagens que não sejam do órgão/tecido em estudo;
- Imagens de pacientes com idade menor que 18 anos;

- Imagens de cuja resolução de contraste seja inferior a 16 bits;e,
- Imagens com não estejam na posição adequada para o estudo.

Como mencionado anteriormente, imagens de pacientes diagnosticados com COVID-19 compõem a base COVID-19+ e imagens de pacientes com teste negativo são inseridas na base COVID-19-. Isto acontece porque, apesar do paciente não estar doente de COVID-19, ele está acometido por outras doenças pulmonares, infecciosas ou inflamatórias, como pneumonia, cardiomegalia, derrame pleural, atelectasia, consolidação.

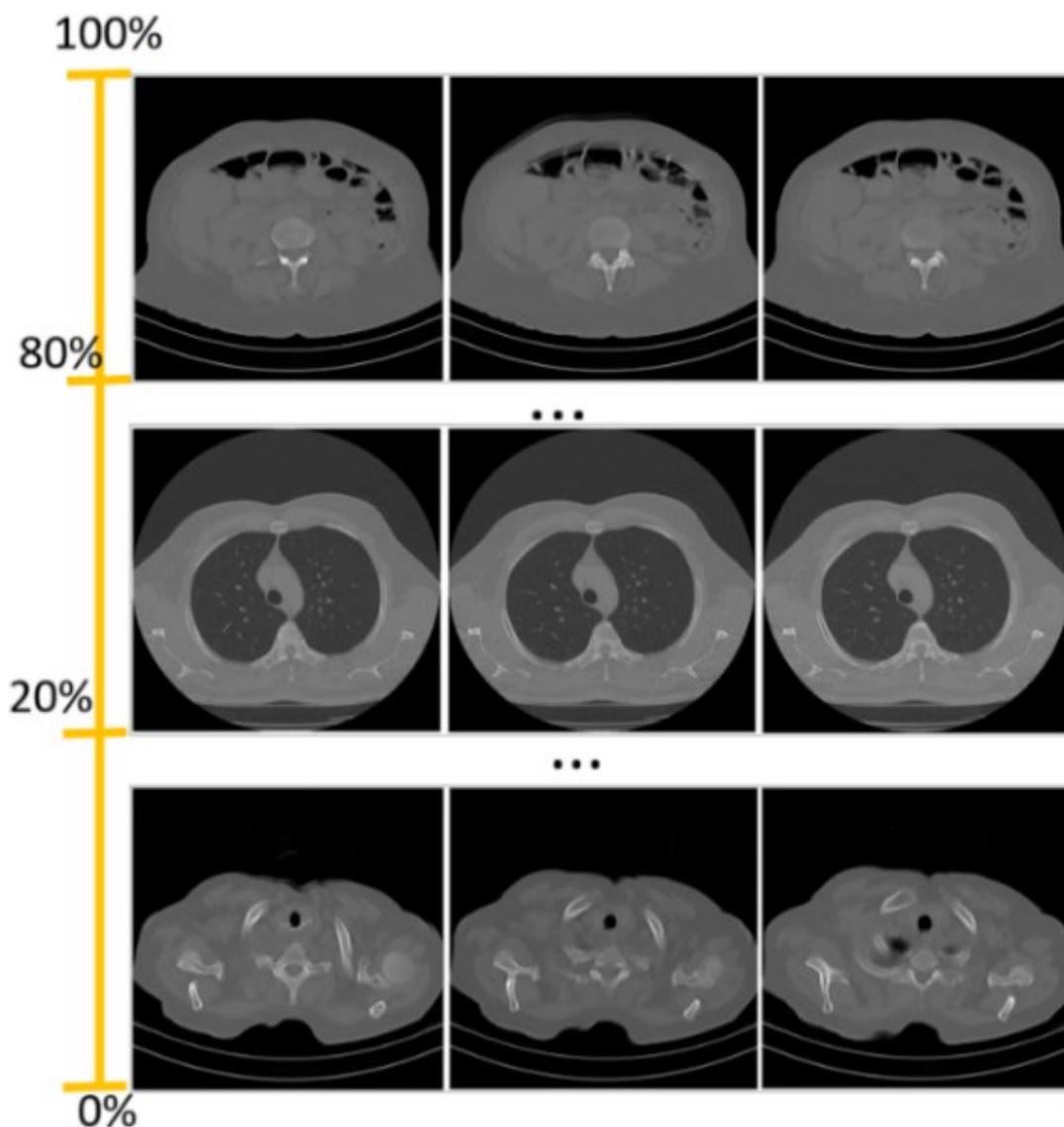


Figura 54 – Seleção das fatias da TC que serão utilizadas

As próximas subseções descrevem os repositórios originais das imagens e os *datasets* criados a partir da utilização dos critérios.

### 8.5.2. Dataset I

O Dataset I contém imagens obtidas no repositório público *Valencian Region Medical*

*ImageBank* (BIMCV) (DE LA IGLESIA VAYÁ, SABORIT, *et al.*, 2020). Elas foram geradas entre 26/02/2020 e 18/04/2020 são subdivididas em:

- BIMCV-COVID19+ imagens de pacientes com teste positivo para COVID-19, e inclui, também, achados radiográficos de TC e seus respectivos laudos, exame de reação em cadeia da polimerase (PCR), testes de diagnóstico de anticorpos (imunoglobulina G-IgG e imunoglobulina M-IgM);
- BIMCV-COVID19- imagens de pacientes com teste negativo para COVID-19, incluindo achados radiológicos de TC e seus respectivos laudos, que apresentam pacientes com patologias como pneumonia, cardiomegalia, derrame pleural, atelectasia, consolidação etc.

Foram utilizados entre 50 e 400 cortes de TC para cada exame, cuja espessura de corte varia entre 1 mm e 7 mm. Os exames radiográficos de TC foram realizados nos seguintes equipamentos: *Konica Minolta 0862*; *GMM ACCORD DR*; *Philips Medical Systems DigitalDiagnost*; *Philips Medical Systems PCR Elevate*; *Siemens Somatom*; *Toshiba Aquilion*; *Philips DigitalDiagnost*; *Philips Brilliance 16*; *Philips Medical Systems Essenta DR*.

Quanto à distribuição dos pacientes do *dataset* I, foram selecionados 174 pacientes, sendo 87 pacientes BIMCV-COVID19+, constituintes da base COVID-19+ e 87 pacientes BIMCV-COVID19-, que compuseram a base COVID-19-. Dos 88 pacientes da base COVID-19+, 70% foram utilizados na fase de treinamento, 15% na fase de teste e os 15% restantes na fase de validação. A mesma distribuição foi usada na base COVID-19-, conforme Tabela 37. As imagens dos pacientes utilizadas na fase de treinamento não foram usadas novamente nas fases de teste e validação, ou seja, as imagens dos pacientes foram atribuídas às fases disjuntamente.

Tabela 37 – Distribuição das imagens do repositório BIMCV

Sub-base	Treinamento Pacientes/Imagens	Teste Pacientes/Imagens	Validação Pacientes/Imagens
Covid-19+	62/5670	13/1032	13/1032
Covid-19-	62/5670	13/1032	13/1032

### 8.5.3. Dataset II

O *dataset* II é composto por imagens de TC obtidas no repositório privado do Hospital São Lucas da Pontifícia Universidade Católica do Rio Grande do Sul (HSL-PUCRS), geradas entre 03/03/2020 e 30/07/2020. Para utilizar as imagens deste repositório, que é privado, foi submetida uma solicitação de uso ao comitê de avaliação da PUCRS, sob o número

30791720.5.0000.5336. O processo seguiu os trâmites normais e, ao final, a demanda foi aprovada.

O Dataset II é composto por pacientes que testaram positivo e negativo para COVID-19, sendo que estes últimos têm outras doenças pulmonares como pneumonia, cardiomegalia, derrame pleural, atelectasia, consolidação. Os exames de TC foram feitos nos equipamentos: *Siemens; GE Medical Systems, Philips Medical Systems, Toshiba*. Cada exame é composto por 50 a 400 fatias de TC, e cada fatia tem espessura que varia de 1 mm a 5 mm.

Quanto à distribuição dos pacientes do HSL-PUCRS, foram selecionados 60 pacientes positivados para COVID-19, da base do Hospital São Lucas da PUCRS. Destes, 70% foram utilizados na fase de treinamento, 15% na fase de teste e as restantes 15% utilizadas na fase de validação, como mostrado na Tabela 38. As imagens dos pacientes utilizadas na fase de treinamento não foram usadas novamente nas fases de teste e validação, ou seja, os pacientes foram divididos de forma disjunta.

Tabela 38 – Distribuição das imagens do *dataset* II

Sub-base	Treinamento Pacientes/Imagens	Teste Pacientes/Imagens	Validação Pacientes/Imagens
Covid-19+	42/3767	9/750	9/740
Covid-19-	42/3767	9/750	9/740

#### 8.5.4. Dataset III

O *dataset* III é composto por imagens de TC obtidas no repositório privado do Hospital de Clínicas da Universidade Federal de Uberlândia (HC-UFU), Brasil. Elas foram geradas entre 08/04/2020 e 12/10/2020. O *dataset* III foi usado para validar o modelo com imagens totalmente diversas daquelas utilizadas no treinamento e teste. Foram coletadas imagens de 20 pacientes positivados para COVID-19, num total de 2.300 imagens e, 20 pacientes negativados para COVID-19, mas positivos para pneumonia viral, também num total de 2.300 imagens. Os mesmos critérios de inclusão e exclusão foram utilizados na base HC-UFU. Além disso, foi feita anonimização das informações. As imagens foram obtidas pelo equipamento *Toshiba CT Scanner*. Os parâmetros de varredura foram: a matriz de reconstrução da janela pulmonar, 512×512; espessura de corte, 1mm-7mm. A Tabela 39 mostra os dados tabulados.

Tabela 39 – Distribuição das imagens do *dataset* III

Sub-base	Validação Pacientes/Imagens
Covid-19+	20/2300
Covid-19-	20/2300

## 8.6. Modelo WCNN

CNNs foram propostas para avaliar dados de imagem. O nome vem do operador de convolução, uma maneira fácil de fazer operações complexas usando o *kernel* de convolução (RAVI, WONG, *et al.*, 2016). Muitas variações de CNN já foram propostas, como *AlexNet* (KRIZHEVSKY, SUTSKEVER e HINTON, 2012), *Clarifai* (ZEILER e FERGUS, 2014), *GoogLeNet* (SZEGEDY, LIU, *et al.*, 2015). O modelo proposto neste trabalho, WCNN, também é uma variação de uma CNN e segue a estrutura básica de uma CNN (LECUN, BOTTOU, *et al.*, 1998), conforme mostrado na Figura 55.

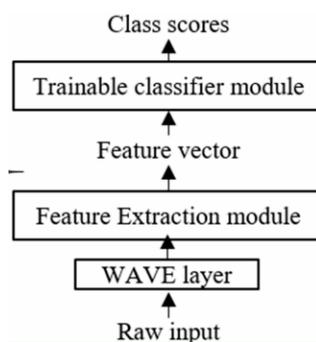


Figura 55 – A estrutura base de uma WCNN é composta por 3 módulos, além dos 2 já conhecidos na literatura, o novo módulo denominado WAVE layer, compõe o modelo.

No modelo WCNN, foi criada a camada WAVE, que utiliza uma transformada *wavelet* para que os dados possam ser processados pelas camadas subsequentes.

A WCNN é composta de quatro estágios: *wave layer*, *feature extraction*, *flatten layer* and *fully connected layer*. Na CNN, as camadas *pooling* e convolução atuam como uma etapa de *feature extraction*, enquanto a etapa de classificação é composta por uma ou mais *fully connected layer* seguidas por uma camada de função sigmóide (WANI, BHAT, *et al.*, 2020). A Figura 56 ilustra o esquema de classificação WCNN.

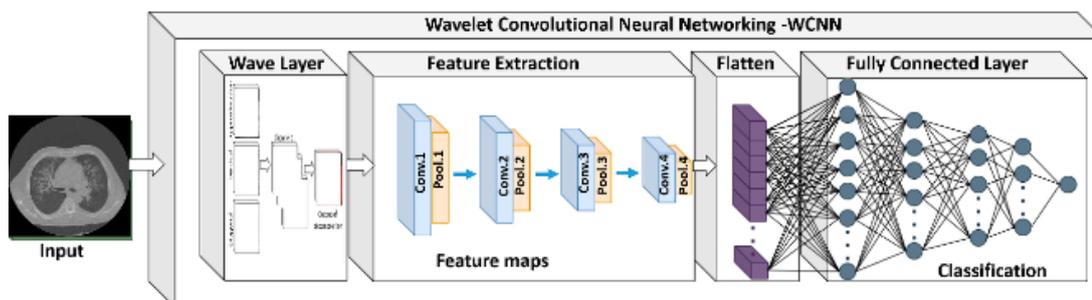


Figura 56 – Esquema de classificação WCNN

### 8.6.1. Camada Wave

A camada Wave imagens de TC, com resolução espacial de  $512 \times 512$  pixels que passam por várias etapas nessa camada, descritas a seguir:

O primeiro passo é responsável por reduzir o impacto do fundo. A imagem é recortada da seguinte maneira: cada corte resulta na retirada de 86 *pixels* nas margens inferior, superior, direita e esquerda. Estas dimensões de corte foram testadas em todas as imagens dos datasets I, II e III para assegurar que não houve perda de informações. A razão pela qual não foi realizado a segmentação pulmonar nas imagens selecionadas é para evitar a remoção de áreas da lesão nos limites pulmonares. O corte resulta em uma imagem de 340×340 *pixels*, conforme Figura 57.

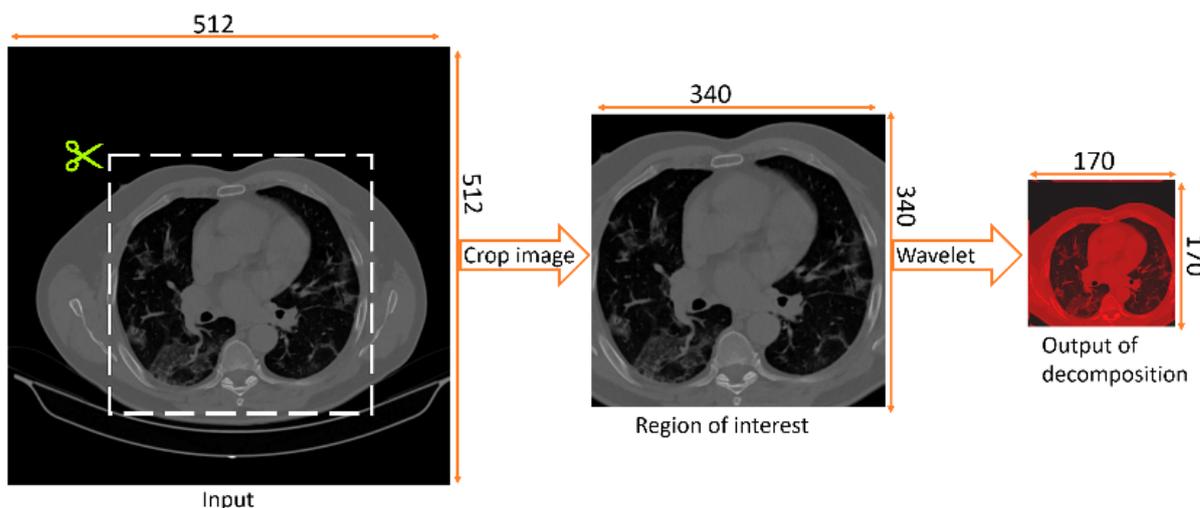


Figura 57 – Seleção da Região de Interesse (ROI)

Na segunda etapa, a imagem é normalizada para remover as variações quaisquer variações causadas por diferentes equipamentos de TC. Suas características são extraídas da distribuição normal padrão, considerando a média  $\mu = 0$  e variância  $\sigma^2 = 1$ . A normalização é feita da seguinte forma: calcula-se média  $\mu$  e a variância  $\sigma^2$ , como na Equação 55 e Equação 56, respectivamente, da imagem  $I$ , que é formada por  $m$  linhas e  $n$  colunas, denotada por  $I_{0,0}, I_{0,1}, \dots, I_{m,n}$  (AGGARWAL e OTHERS, 2018). Assim,  $I_{Normalized}$  é calculada conforme Equação 57.

$$\mu = \frac{\sum_1^m \sum_1^n I_{i,j}}{m*n} \quad \text{Equação 55}$$

$$\sigma^2 = \frac{\sum_1^m \sum_1^n (I_{i,j} - \mu)^2}{m*n} \quad \text{Equação 56}$$

$$I_{Normalized} = \frac{I_{i,j} - \mu}{\sigma} \quad \text{Equação 57}$$

Na terceira etapa a imagem é processada através da decomposição da transformada *wavelet*, em um único nível de decomposição, utilizando a *wavelet* mãe *Coiflets 5*. Dos quatro coeficientes gerados (aproximado, horizontal, vertical e diagonal), neste trabalho apenas três

são utilizados para renderizar a imagem digital. Uma imagem digital é composta pelo espaço *Red, Green e Blue* (RGB), de modo que o canal R recebe o coeficiente aproximado, o canal G recebe o coeficiente vertical e o canal B recebe o coeficiente diagonal, compondo uma saída de decomposição que ser utilizado pelas camadas, conforme mostrado na Figura 58. A imagem recortada pela primeira etapa resulta em uma região de interesse do pulmão que possui resolução espacial de  $340 \times 340$ . A após a decomposição pela transformada *wavelet* a saída da decomposição é uma imagem com resolução espacial de  $170 \times 170$ , como mostrada na Figura 58.

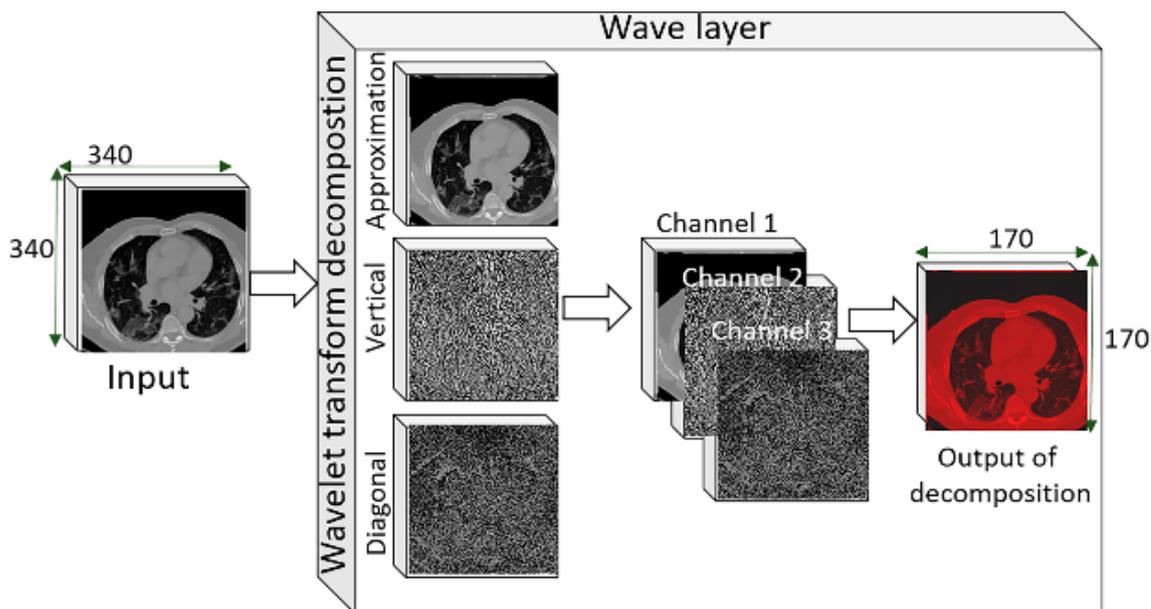


Figura 58 – Camada wave da WCN

### 8.6.2. Feature extraction

A operação de convolução foi estabelecida para a camada convolucional, na qual um *kernel* é usado para mapear as ativações de uma camada para a próxima. A operação de convolução coloca o *kernel* em cada posição possível na imagem (ou camada oculta) para que o *kernel* sobreponha toda a imagem e execute um produto escalar entre os parâmetros do *kernel* e seu campo receptivo correspondente, ao qual o *kernel* é aplicado, na imagem. A operação de convolução é executada em todos os pixels da imagem para definir a próxima camada, na qual as ativações mantêm suas relações espaciais na camada anterior (LECUN, BENGIO e HINTON, 2015, AGGARWAL e OTHERS, 2018, PONTI e DA COSTA, 2018). Pode haver vários *kernels* na camada convolucional. Cada *kernel* revela um recurso, como uma borda ou um canto. Durante a passagem para frente, cada *kernel* é deslizado para a largura e a altura da imagem (ou camada oculta), gerando o *feature map* (LECUN, BENGIO e HINTON, 2015, AGGARWAL e OTHERS, 2018, PONTI e DA COSTA, 2018, BALAS, ROY, *et al.*, 2019).

A camada de *pooling* é usada para reduzir o tamanho espacial do campo receptivo, reduzindo assim o número de parâmetros de rede. A camada de agrupamento seleciona uma amostra reduzida de cada mapa de recursos da camada convolucional. *Max pooling* foi a técnica utilizada para este trabalho; gera o valor máximo no campo receptivo. O campo receptivo é  $2 \times 2$ , portanto, max pooling emitirá o máximo dos quatro valores de entrada (WANI, BHAT, *et al.*, 2020).

### 8.6.3. Flattening layer

Após os processos de convolução e agrupamento, o próximo passo é o *flatten*, que converte todos os mapas de características em uma matriz unidimensional, criando um vetor de entrada para a *fully connected layer* (WANI, BHAT, *et al.*, 2020).

### 8.6.4. Fully connected layer

Nesta camada, cada neurônio da camada anterior está conectado aos neurônios da camada subsequente, e todos os valores contribuem para prever quão fortemente um valor se correlaciona com uma determinada classe (WANI, BHAT, *et al.*, 2020). *Fully connected layer* podem ser colocadas umas sobre as outras para capturar combinações de recursos ainda mais sofisticadas. A saída da última *fully connected layer* é alimentada por uma função de ativação que gera as pontuações da classe. A WCNN utiliza a função de ativação sigmóide, cujo valor de saída varia na faixa  $[0, 1]$ . As entradas da WCNN com um valor de saída acima de 0,5 são classificadas como COVID, e as com valor abaixo de 0,5 referem-se a outras doenças pulmonares (WANI, BHAT, *et al.*, 2020).

WCNN usa *Adaptive Moment Estimation* (ADAM), uma técnica de otimização adaptativa que salva uma média exponencialmente decrescente de gradientes quadrados anteriores  $v_t$ . Além disso, ADAM também calcula a média dos segundos momentos dos gradientes  $m_t$  (WANI, BHAT, *et al.*, 2020, KINGMA e BA, 2014). Os valores de variância média e não centrada  $m_t$  são apresentados na Equação 58 e Equação 59, respectivamente:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1)gt \quad \text{Equação 58}$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2)gt^2 \quad \text{Equação 59}$$

ADAM atualiza as médias móveis exponenciais do gradiente e do gradiente quadrado onde os hiperparâmetros  $\beta_1, \beta_2 \in [0, 1]$  controlam as taxas de decaimento dessas médias móveis (equação 60) e (equação 61):

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t} \text{Equação 60}$$

$$\hat{v}_t = \frac{v_t}{1-\beta_2^t} \text{Equação 61}$$

A equação final para atualização é (Equação 62):

$$w_{t+1} = w_t - \frac{\alpha \cdot \hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \quad \text{Equação 62}$$

onde  $\alpha$  é a taxa de aprendizado e  $\epsilon$  é uma constante adicionada ao denominador para métodos de conversão rápida para evitar a divisão por 0 (WANI, BHAT, *et al.*, 2020, KINGMA e BA, 2014).

A WCNN usa *dropout*, que é a técnica mais popular para reduzir *overfitting*. *Dropout* refere-se ao abandono de neurônios em uma rede neural durante o treinamento. Eliminar um neurônio significa desconectá-lo temporariamente, de todas as suas conexões internas e externas, da rede. Neurônios descartados não contribuem para a passagem para frente nem para a passagem para trás. Ao usar a técnica de *dropout*, a rede é forçada a aprender os recursos mais robustos à medida que a arquitetura da rede muda a cada entrada (WANI, BHAT, *et al.*, 2020, BALAS, ROY, *et al.*, 2019).

A saída de cada camada convolucional é alimentada por uma função de ativação. A camada de função de ativação consiste em uma função de ativação que usa o *feature map* produzido pela camada convolucional e gera o mapa de ativação como saída. A função de ativação é usada para alterar o nível de ativação de um neurônio em um sinal de saída. Assim, ele realiza uma operação matemática e gera o nível de ativação do neurônio em um intervalo específico, por exemplo,  $[0, 1]$  ou  $[-1, 1]$  (WANI, BHAT, *et al.*, 2020). As funções utilizadas foram as seguintes:

1. Sigmóide / função de ativação logística: A função sigmóide  $\sigma(x) = \frac{1}{1+e^{-x}}$  é uma curva com o formato de um S (PONTI e DA COSTA, 2018).
2. A função de ativação  $f(x) = \max(0, x)$  é chamada Rectified Linear Unit (ReLU) (PONTI e DA COSTA, 2018) e gera um mapa de ativação não linear.

A arquitetura detalhada do WCNN é apresentada na Tabela 40. Além disso, uma função de ativação de unidade linear retificada (ReLU) é usada após cada camada de convolução (1ª, 3ª, 5ª e 7ª) e camadas densas (9ª, 10ª, 11ª e 12ª). Para reduzir a possibilidade de *overfitting*, foi utilizada uma taxa de *dropout* de 20% para as primeiras quatro camadas totalmente conectadas (9ª, 10ª, 11ª e 12ª).

Tabela 40 – Arquitetura do WCNN é composta por *Wave Layer* (W), *Convolutacional* (C), *Max pooling* (M) e *Fully Connected Layer* (F)

Camada	WCNN		
	Dimensão	Input/Output	Dimensão
0	W		512×512
1	C	5×5×256	296×296×256
2	M	2×2	148×148×256
3	C	3×3×128	146×146×128
4	M	2×2	73×73×128
5	C	3×3×64	71×71×64
6	M	2×2	35×35×64
7	C	3×3×32	33×33×32
8	M	2×2	16×16×32
9	F	16×16×32 ×256	1×256
10	F	1×1×256×128	1×128
11	F	1×1×128×64	1×64
12	F	1×1×64×32	1×32
13	F	1×1×32×1	1×1

## 8.7. Parâmetros do treinamento e métricas de avaliação

A WCNN foi desenvolvida em *Python* com biblioteca *Tensorflow* (CHOLLET, 2016) rodando em máquina I7-8750H processador Intel, CPU de 2,21GHz, 16,0 GB de RAM e uma placa gráfica GeForce GTX 1060 com Max-Q Design.

Para realizar o primeiro treinamento, os pesos foram inicializados aleatoriamente. A rede foi treinada com ADAM (WANI, BHAT, *et al.*, 2020), utilizando os parâmetros padrão  $\beta_1 = 0,9$  e  $\beta_2 = 0,999$  (KINGMA e BA, 2014). A taxa de aprendizagem inicial foi definida como  $\alpha = 0,001$ , reduzida por um fator de 10. ADAM (WANI, BHAT, *et al.*, 2020) apresenta melhor desempenho em comparação com outras técnicas adaptativas; tem uma taxa de conversão rápida, reduzindo assim as chances de erro e aumentando a acurácia. Ele também supera problemas enfrentados por outras técnicas de otimização, como taxa de aprendizado decrescente, alta variação nas atualizações e convergência lenta (WANI, BHAT, *et al.*, 2020). Para treinamento foram realizadas 200 épocas, com *batch size* de 32. A técnica de *pooling* adotada foi *max pooling* com filtro (2×2) e taxa de 20% para a técnica de *dropout*. As seguintes métricas foram usadas para avaliar o modelo WCNN:

1. *Accuracy* (ACC): taxa de classificação precisa conforme o número total de elementos.

2. *Sensitivity* (Sen): taxa de *True Positive*.
3. *Specificity* (Sp): taxa de *True Negativo*.

Eles são comumente usados para avaliar o desempenho de algoritmos de classificação (RUUSKA, HÄMÄLÄINEN, *et al.*, 2018, SKANSI, 2018, KHATAMI, KHOSRAVI, *et al.*, 2017). Existe uma maneira padrão de mostrar o número de *True Positives* (TP), *False Positives* (FP), *True Negatives* (TN) e *Falses Negatives* (FN) para ser mais visual. Este método é chamado de matriz de confusão.

A matriz de confusão permite determinar as seguintes métricas (RUUSKA, HÄMÄLÄINEN, *et al.*, 2018, SKANSI, 2018, KHATAMI, KHOSRAVI, *et al.*, 2017) como na Tabela 41.

Tabela 41 – Métricas de avaliação do modelo

<b>Accuracy (ACC)</b>	<b>Sensitivity (Sen)</b>	<b>Specificity (Sp)</b>
$ACC = \frac{TN + TP}{TN + TP + FN + FP}$	$Sen = \frac{TP}{TP + FN}$	$Sp = \frac{TN}{TN + FP}$

Também é possível gerar a Curva de Características de Operação do Receptor (curva ROC). A análise ROC é frequentemente chamada de razão de precisão ROC, uma técnica comum para a avaliar a acurácia dos modelos de probabilidade de *default* (SHIRAZI, CHABOK e MOHAMMADI, 2018).

## 8.8. Resultados do Modelo WCNN

Nesta seção são apresentados os resultados da aplicação do modelo WCNN nos *dataset* I e *dataset* II e na base externa *dataset* III, os quais foram obtidos levando em consideração duas abordagens para avaliar o desempenho do modelo WCNN: validação interna e validação externa. Para validação interna, foram usados os *dataset* I e *dataset* II, sendo que os dados utilizados são da mesma fonte que os dados de desenvolvimento e para validação externa são de fontes diferentes (ROBERTS, DRIGGS, *et al.*, 2021).

### 8.8.1. Resultados do *dataset* I

O treinamento feito com o *dataset* I, consistiu em 200 épocas e gerou os resultados exibidos nos gráficos a) *Training Loss* e b) *Training Accuracy*, que compõem a Figura 59.

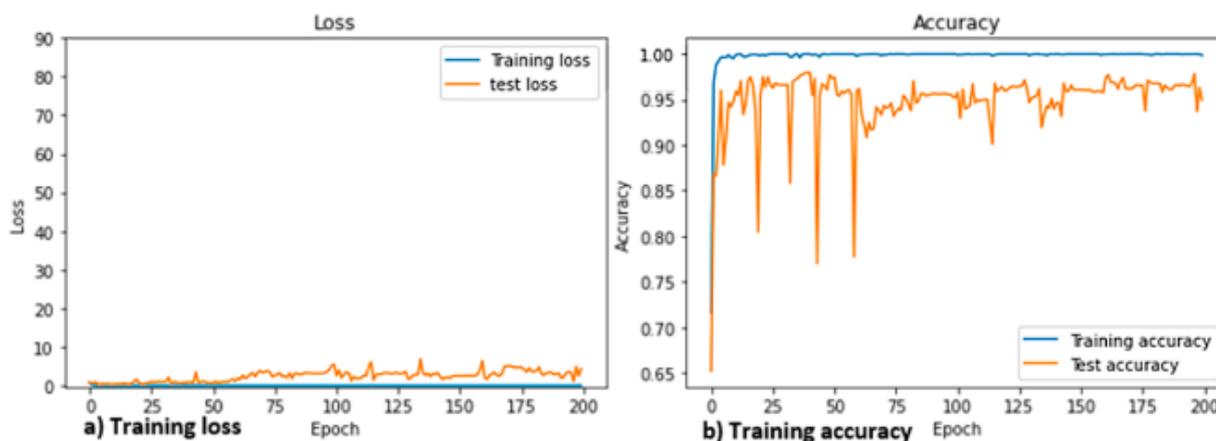


Figura 59 – WCNN *training loss* e *training accuracy* do *dataset I*

A matriz de confusão calculada ao validar a base interna *dataset I*, cuja constituição é apresentada na Tabela 37, e exibida na Figura 60.

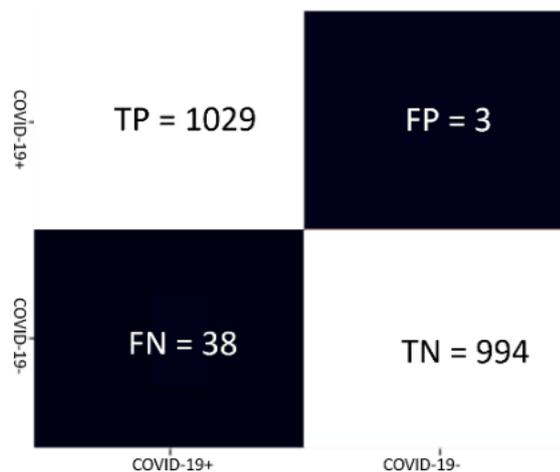


Figura 60 – Matriz de confusão do *dataset I*

A partir dos resultados da Figura 60, as métricas *accuracy*, *sensitivity* e *specificity* foram calculadas e são mostradas na Tabela 42.

Tabela 42 – Resultado das métricas para *dataset I* do BIMCV

Classe	<i>Accuracy</i>	<i>Recall /sensitivity</i>	<i>Specificity</i>
Covid-19+	0,9801	0,9643	0,9969

Usando os valores da Tabela 42, a curva ROC foi calculada com  $(1-Sp) = 0,0031$  e  $sem = 0,9643$ , como  $x$  e  $y$ , respectivamente. Com base na análise ROC, a área (AUC) foi calculada como sendo 0,98, como exibido na Figura 61.

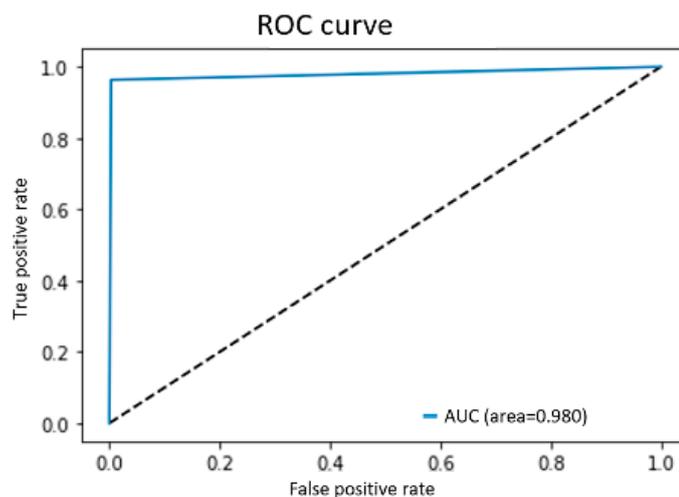


Figura 61 – Curva ROC do *dataset I*

### 8.8.2. Resultados do *dataset II*

O treinamento do *dataset II*, feito em 200 épocas, gerou os resultados exibidos nos gráficos a) *Training Loss* e b) *Training Accuracy*, que compõem a Figura 62.

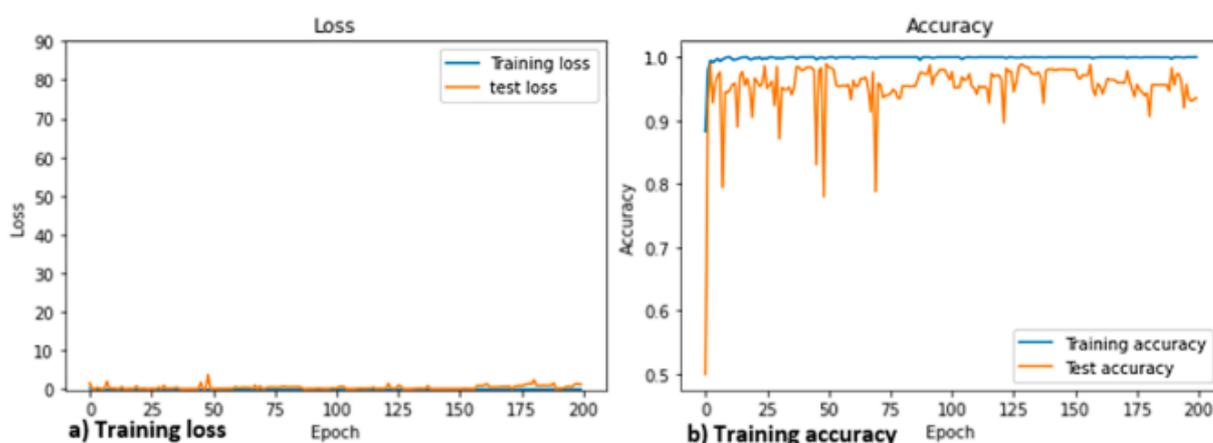


Figura 62 – WCNN *training loss* e *training accuracy* do *dataset II*

A Figura 62a mostra que os valores da função de custo, *Loss*, ao longo das épocas, sempre se manteve próximo de zero. Por outro lado, a Figura 62b, exibe o comportamento dos valores da acurácia, na fase de treinamento, o qual se manteve próximo de 1.

A matriz de confusão, calculada ao validar a base interna *dataset II*, é apresentada na Tabela 38 e exibida na Figura 63.

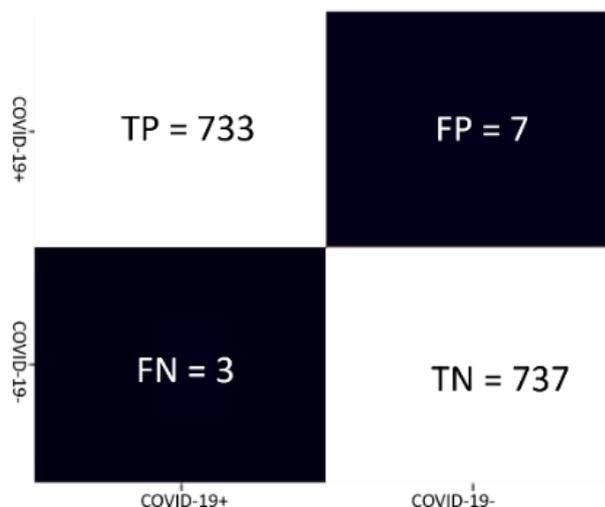


Figura 63 – Matriz de confusão do *dataset II*

Da mesma forma dos resultados do dataset I, os resultados da matriz de confusão para o dataset II foi utilizado para calcular as métricas accuracy, sensitivity e specificity, e estão apresentadas na Tabela 43.

Tabela 43 – Resultado das métricas para *dataset II*

Classe	Accuracy	Recall /sensitivity	Specificity
Covid-19+	0,9932	0,9959	0,9932

De acordo com os valores apresentados na Tabela 43, a curva ROC foi calculada com  $(1-Sp) = 0,0094$  e  $Sen = 0,9959$ , como x e y, respectivamente. Com base na análise ROC, a área (AUC) foi de 0,993. Conforme mostra a Figura 64.

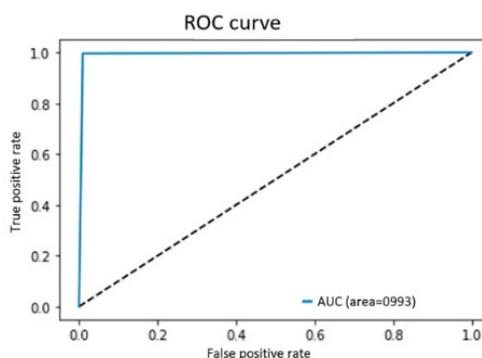


Figura 64 – Curva ROC do *dataset II*

### 8.8.3. Resultados do *dataset III*

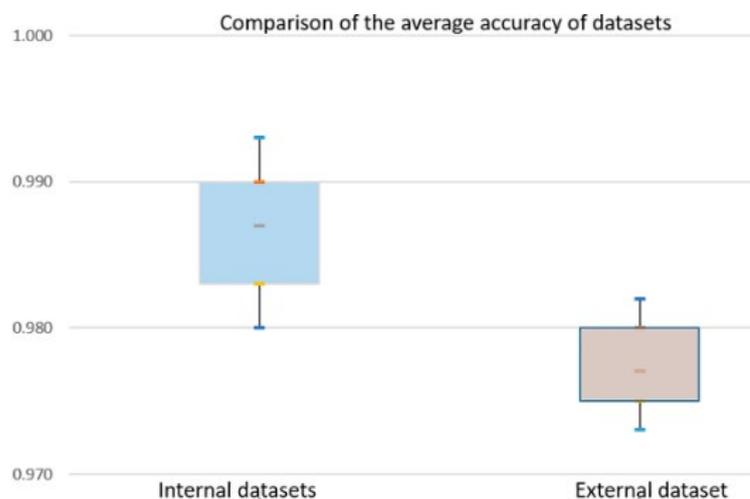
As imagens do *dataset III* foram submetidas ao modelo WCNN em 2 cenários: *i*) WCNN foi executada com os pesos do treinamento do *dataset I*; e, *ii*) WCNN foi executada com os pesos do treinamento do *dataset II*, o que gerou os resultados documentados na Tabela 44

Tabela 44 – Resultado das métricas para *dataset* III- HC-UFU

Class	WCNN trained by	Accuracy	Recall/Sensitivity	Specificity
Covid-19+	<i>dataset</i> I	0,9726	0,9621	0,9835
Covid-19+	<i>dataset</i> II	0,9821	0,9911	0,9735

## 8.9. Resultados Consolidados

Os resultados foram consolidados de acordo com as categorias do ambiente: interno e externo. No gráfico da Figura 65 observa-se que a validação dos *datasets* internos apresentou maior acurácia média que a encontrada na validação do *dataset* externo.

Figura 65 – Média da acurácia dos *datasets* internos (*dataset* I e *dataset* II) e *dataset* externo (*dataset* III)

Como existe discrepância entre os resultados dos *datasets* internos e externo, conforme exibido no gráfico da Figura 65 e nos dados da Tabela 45, para obter um cenário realista, decidiu-se considerar a acurácia média encontrada nestes *datasets* para compará-los com os valores de pesquisas, no estado da arte.

Tabela 45 – Resultado consolidado das métricas

Tipo	Classe	<i>Dataset</i>	<i>Accuracy</i>	<i>Sensitivity</i>	<i>Specificity</i>
Interno	Covid-19+	I	0,9801	0,9643	0,9969
	Covid-19+	II	0,9932	0,9959	0,9932
Externo	Covid-19+	III–(wcnw treinado por <i>dataset</i> I)	0,9726	0,9621	0,9835
	Covid-19+	III–(wcnw treinado por <i>dataset</i> II)	0,9821	0,9911	0,9735
<b>Médias consolidadas</b>			<b>0,9819</b>	<b>0,9783</b>	<b>0,9867</b>

A Tabela 46 apresenta os dados utilizados para comparar o modelo WCNN com os do estado da arte.

Tabela 46 – Modelo WCNN *versus* modelos do estado da arte

Trabalho	Tipo CNN	P/P	P/I	D_A	I_R	ACC	Sen	Sp
FAN, FENG, <i>et al.</i> , 2022,	<i>Trans-CNN</i>	Público	Imagem	Sim	Sim	0,9673	0,9776	0,9602
TER-SARKISOV, 2022	<i>COVID-CT-Mask-Net</i>	Público	Imagem	Não	Não	0,9166	0,9080	0,9162
RANGARAJAN e RAMACHANDRAN, 2022	<i>Hybrid Model (queezeNet e ShuffleNet)</i>	Público	Imagem	Sim	Sim	0,9781	0,9615	0,9608
ÖZKAYA, ÖZTÜRK e BARSTUGAN, 2020	<i>FusionModel (VGG-16, GoogleNet e ResNet-50)</i>	Público	Imagem	Não	Não	0,9827	0,9893	0,9760
KOGILAVANI, PRABHU, <i>et al.</i> , 2022	<i>VGG16, DeseNet121, MobileNet, NASNet, Xception, and EfficientNet</i>	Público	Imagem	Sim	Sim	0,9768	0,9579	0,9971
CHEN, WU, <i>et al.</i> , 2020	<i>UNet++</i>	Privado	Imagem	Não	Não	0,9885	0,9434	0,9916
CHEN, CHEN, <i>et al.</i> , 2021	<i>Covid-19-CNN</i>	Público	Imagem	Não	Sim	0,9670	0,9780	0,9556
	Nosso modelo	Ambos	Paciente	Não	Não	0,9819	0,9783	0,9867

Onde:

- **P/P**: Indica se as bases de imagem são **Públicas/Privadas**
- **P/I**: Indica se as imagens são selecionadas por **Paciente/Imagem**
- **D\_A?**: Indica se o Modelo CNN utiliza alguma técnica de *Data Augmentation*.
- **I\_R?**: Indica se o Modelo CNN utiliza alguma técnica de *Image Resizing*.

## 8.10. Discussão

O desenvolvimento deste trabalho consistiu da utilização do modelo WCNN em três bases de imagens de CT de tórax. Foram duas bases de imagens internas *dataset I* e *dataset II*, que são, respectivamente pública e privada, e a base externa *dataset III*. Como é de se esperar nestes casos, as bases são heterogêneas, possuem quantidade variáveis de pacientes e imagens, foram obtidas por diversos tipos de equipamentos de TC.

Para que os resultados de *performance* do modelo pudessem ser comparados com segurança, foram criados critérios de inclusão e exclusão das imagens (ROBERTS, DRIGGS, *et al.*, 2021), contando com a ajuda de um médico radiologista. Exemplos de critério de inclusão e exclusão, são, respectivamente, uso de imagens de tamanho 512×512 e eliminação das imagens cujos pacientes tenham idade inferior a 18 anos. A distribuição dos dados para treinamento, teste e validação foi feita por paciente e não por imagem, o que elimina certo risco de

utilizar uma mesma imagem em treinamento e teste, por exemplo. Ressalta-se que o modelo WCNN não utiliza os recursos de *data augmentation* e *resizing*, como em alguns trabalhos da literatura, vide Tabela 46. Essa abordagem evita o risco de perda de informações pelo aumento artificial da imagem.

Além dos critérios de inclusão e exclusão, foi criada a camada *wave* para padronizar as imagens. A *wave layer* normaliza as imagens, calcula a saída de decomposição mediante transformada *wavelet*, substituindo os canais RGB originários pelos canais dos coeficientes aproximada, vertical e diagonal e finalmente, compondo uma nova imagem digital que é repassada para as camadas seguintes; o que ajuda a diminuir a diferença da extração das imagens por diversos equipamentos. A camada *wave* processa a imagem mediante *wavelet transform decomposition*, em um único nível de decomposição, utilizado a *wavelet* mãe *Coiflets 5*. Assim, o uso da WCNN no domínio *wavelet*, pode impulsionar os procedimentos de treinamento.

Este impulso se dá porque o uso de coeficientes *wavelet* estimula a dispersão de ativação em camadas ocultas e na camada de saída. Os coeficientes *wavelet* se tornam mais esparsos e, portanto, é mais fácil para a rede aprender mapas esparsos em vez de mapas densos. Os histogramas na Figura 66 ilustram a distribuição esparsa dos coeficientes vertical, diagonal e de aproximação. O alto nível de esparsidade reduz ainda mais o tempo de treinamento necessário para a rede localizar o mínimo global (GUO, SEYED MOUSAVI, *et al.*, 2017).

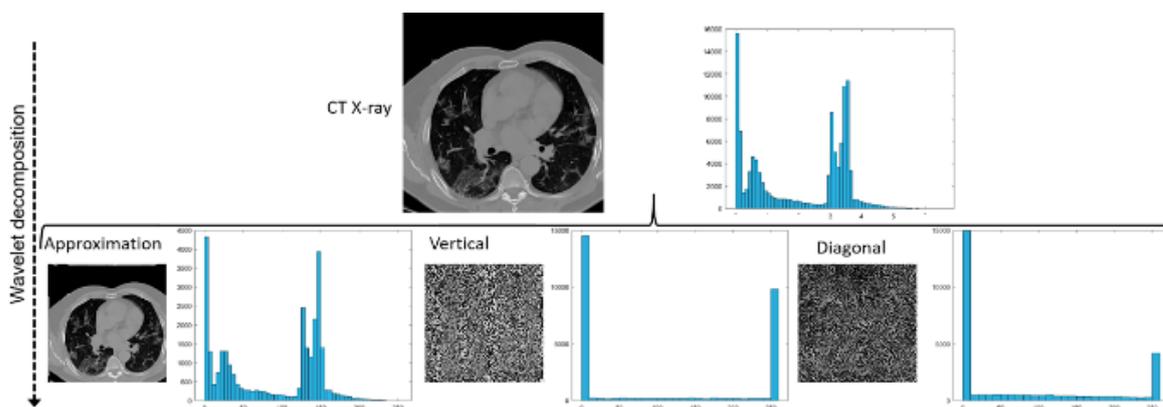


Figura 66 – Exemplo de histograma da imagem original e seus correspondentes coeficientes diagonal, vertical e aproximada

Feito isto, calcula-se a média dos valores das métricas dos *datasets* internos e *dataset* externo, obtendo assim, acurácia média de 0,9819, sensibilidade 0,9783 e especificidade 0,9867. A comparação do resultado do modelo WCNN com o estado da arte, resultou que o modelo WCNN ficou entre os três melhores trabalhos.

## 8.11. Conclusão

Diante do exposto, o modelo WCNN apresenta as seguintes vantagens em comparação com os trabalhos anteriores relacionados: o modelo evita processar e redimensionar previamente as imagens, mediante a criação e aplicação de uma camada adicional à CNN, chamada *wave layer*. Esta camada utiliza a transformada *wavelet* para decompor a imagem e extrair suas características, sendo responsável pelo pré-processamento e geração da imagem de saída, a qual será processada pelas demais camadas do modelo.

Além disso, a criação da *Wave layer*, que substitui a camada *Input* da biblioteca do *Keras*, seleciona a região de interesse, normaliza a região através da sua média e desvio padrão e forma uma nova imagem através da decomposição da transformada *wavelet*, usando a família *coiflet 5*. A seleção da região de interesse elimina o plano de fundo da imagem; a normalização elimina as variações causadas por diversos equipamentos e a decomposição da transformada *wavelet* resulta em uma imagem com resolução espacial  $170 \times 170$ , que retêm informações essenciais para a classificação da doença, além de acelerar o processo de treinamento da rede.

Os resultados obtidos indicam que o investimento de tempo, recursos humanos, financeiros e computacionais, na criação do WCNN, é uma abordagem promissora para auxiliar os profissionais no prognóstico do novo coronavírus por meio de imagens de Tomografia Computadorizada de tórax.

## 9. Conclusão

---

O modelo WCNN foi desenvolvido inspirado em arquiteturas CNN bem estabelecidas na literatura (KRIZHEVSKY, SUTSKEVER e HINTON, 2012, ZEILER e FERGUS, 2014, SZEGEDY, LIU, *et al.*, 2015) mediante a realização de revisão bibliográfica, documentada no Capítulo 3 e nas seções Trabalhos Relacionados dos Capítulos 4, 5 e 8.

O modelo WCNN cumpriu seus objetivos com a criação de uma nova camada de entrada que processa as imagens sem redimensioná-las, utilizando a transformada *wavelet* para decomposição das imagens. Além disso, O WCNN foi utilizado em várias bases de imagens radiológicas heterogêneas, tanto públicas quanto privadas, de tamanho, equipamento e perfis diferentes. Ele foi comparado com o estado da arte, obtendo resultados compatíveis com os encontrados na literatura.

A *Wave layer* é uma camada de entrada que substitui a camada *Input* da biblioteca do *Keras*. Ela executa três subprocessos com as imagens: o primeiro passo *i)* é normalizar a imagem mediante sua média e desvio padrão. Com a imagem normalizada, no segundo passo, *ii)* ela é recortada para evidenciar a anatomia do órgão em análise, reduzindo, assim, o impacto do fundo. Feito isso, o terceiro e último passo é *iii)* processar a imagem pela transformada wavelet, em um único nível de decomposição, utilizado a wavelet-mãe *Coiflets 5*. Dos quatro coeficientes gerados (aproximada, horizontal, vertical e diagonal), apenas três foram selecionadas para formar a imagem digital (aproximada, vertical e diagonal). Considerando modelos de espaços de imagens como, *Red, Green e Blue* (RGB), RGB, a decomposição das imagens por wavelet assumiu também um espaço de 3 canais, assim o canal R recebe o coeficiente de aproximação, o canal G recebe o coeficiente vertical e o canal B recebe o coeficiente diagonal, formando a output of decomposition.

Assim, uma das vantagens da *wave layer* em relação a camada *Input* da biblioteca *keras*, é permitir que o pré-processamento seja feito nela, como por exemplo:

- A normalização da entrada de dados;
- A aplicação de filtros para reduzir os ruídos das imagens médicas, a partir da decomposição da imagem com a transformada *wavelet*;
- A aplicação de recortes, a fim de obter um melhor foco do órgão de estudo;

Os experimentos descritos nos capítulos 4, 5 e 8 alcançaram resultados compatíveis com os da literatura, sendo que para as métricas ACC, SEN e SP ficaram no intervalo de [0,97, 1].

A partir das métricas SEN e SP, a curva ROC foi calculada, para cada estudo. Os resultados da curva ROC ficaram no intervalo de [0,97, 1].

O WCNN foi testado nos seguintes experimentos:

- Comparativo de desempenho utilizando a base *ChestXpert*, que contém imagens de radiografia de tórax com 14 patologias pulmonares. Ele foi realizado para verificar o desempenho da WCNN quando comparada a outra CNN. O teste das eficiências resultou em AUC de 85,96% para o modelo WCNN e 80,65% para a rede DenseNet121, o que representa que a WCNN é 6,58% mais eficaz que a DenseNet121.
- Uma aplicação da WCNN como técnica de segmentação, usando a base DRIVE, de imagens de fundo do olho com retinopatia diabética, degeneração macular, glaucoma e hipertensão, a qual levantou alguns pontos positivos do modelo: *i*) pode ser utilizado como técnica de segmentação; *ii*) a Wave layer pode receber imagens médicas de diferentes resoluções espaciais; e, *iii*) foi reforçado o problema de redimensionamento, como mostrado por este experimento, uma vez que o modelo não conseguiu aprender a segmentação dos vasos sanguíneos mais finos. Mesmo com o problema do redimensionamento, a WCNN alcançou acurácia de 0,9570, enquanto a tese (SANTOS, 2022), onde foi utilizada Unet, atingiu acurácia de 0,9713.
- Utilização da WCNN para reconhecimento de padrões em imagens de TC de tórax, de pacientes diagnosticados com COVID-19 e outras infecções pulmonares. WCNN foi aplicada em diversas bases de imagens públicas e privadas, obtidas de equipamentos radiológicos de vários modelos e fabricantes. Os resultados consolidados das métricas Acurácia (ACC), Sensibilidade (Sen) e Especificidade (Sp) retornaram os valores 0,9819, 0,9783 e 0,98, respectivamente; o que indica que a associação de CNNs e transformadas wavelet é promissora para a criação de modelos de classificação.

Em todos os experimentos em que a WCNN foi utilizada, os resultados foram compatíveis com os da literatura, mostrando que o modelo, além de versatilidade na solução de problemas, apresenta bom poder de generalização, algo desejável em classificadores.

Dentre as dificuldades que precisaram ser superadas na condução desta pesquisa, destacam-se:

- 1) O início do trabalho foi marcado pelo fato de apenas serem encontradas bases de imagens, redimensionadas e com profundidade de 8 bits, o que não condiz com a realidade de trabalho dos médicos radiologistas. A solução de contorno adotada foi

a utilização dos bancos de dados de imagens do Hospital de Clínicas da Universidade Federal de Uberlândia e do Hospital São Lucas da Pontifícia Universidade Católica de Porto Alegre, ambos compostos por imagens de pacientes com COVID-19 e outras enfermidades pulmonares.

- 2) A maioria dos trabalhos correlatos usaram a camada Input da biblioteca do Keras para tratar as entradas dos dados. Como não existia referência na literatura sobre a criação de uma camada de entrada personalizada, foi necessário criar totalmente esta camada, usando a biblioteca *TensorFlow*.

Algumas sugestões para trabalhos futuros são:

- 1) Aplicar o modelo em outras áreas que também precisam de ferramentas robustas de reconhecimento de padrões, tais como, metalurgia, agropecuária, processamento de alimentos, entre outras.
- 2) Aplicar o modelo em outras modalidades de imagens médicas e testar o modelo como ferramenta de segmentação em diferentes problemas.

Por fim, conclui-se que os resultados obtidos indicam que o investimento de tempo, recursos humanos, financeiros e computacionais, na criação do WCNN, é uma abordagem promissora para auxiliar os profissionais em sistemas computadorizados de auxílio ao diagnóstico e como demonstrado nas publicações, no prognóstico do novo coronavírus por meio de imagens de Tomografia Computadorizada de tórax.

## Referências

- ABBAS, A.; ABDELSAMEA, M.; GABER, M. Classification of COVID-19 in chest X-ray images using DeTraC deep convolutional neural network, abr. 2020. <https://doi.org/10.1101/2020.03.30.20047456>
- ACHARYA, T.; RAY, A. K. **Image processing: principles and applications**. [S.l.]: John Wiley & Sons, 2005. <https://doi.org/10.1002/0471745790>
- AGGARWAL, C. C.; OTHERS. **Neural networks and deep learning**. [S.l.]: Springer, 2018. <https://doi.org/10.1007/978-3-319-94463-0>
- ASSARI, Z.; MAHLOOJIFAR, A.; AHMADINEJAD, N. A bimodal BI-RADS-guided GoogLeNet-based CAD system for solid breast masses discrimination using transfer learning. **Computers in Biology and Medicine**, v. 142, p. 105160, 2022. ISSN ISSN: 0010-4825. <https://doi.org/10.1016/j.compbiomed.2021.105160>
- ÁVILA-TOMÁS, J. F.; MAYER-PUJADAS, M. A.; QUESADA-VARELA, V. J. La inteligencia artificial y sus aplicaciones en medicina II: importancia actual y aplicaciones prácticas. **Atención Primaria**, v. 53, p. 81, 2020. <https://doi.org/10.1016/j.aprim.2020.04.013>
- BALAS, V. E. et al. **Handbook of deep learning applications**. [S.l.]: Springer, v. 136, 2019. <https://doi.org/10.1007/978-3-030-11479-4>
- BARSTUGAN, M.; OZKAYA, U.; OZTURK, S. Coronavirus (covid-19) classification using ct images by machine learning methods. **arXiv preprint arXiv:2003.09424**, 2020. <https://doi.org/10.48550/arXiv.2003.09424>
- BRUNO, A. et al. A novel solution based on scale invariant feature transform descriptors and deep learning for the detection of suspicious regions in mammogram images. **Journal of Medical Signals and Sensors**, v. 10, p. 158, 2020. [https://doi.org/10.4103/jmss.jmss\\_31\\_19](https://doi.org/10.4103/jmss.jmss_31_19)
- CHAN, Y. T. **Wavelet basics**. [S.l.]: Springer Science & Business Media, 1994.
- CHEN, J. et al. Deep learning-based model for detecting 2019 novel coronavirus pneumonia on high-resolution computed tomography. **Scientific reports**, v. 10, p. 1–11, 2020. <https://doi.org/10.1038/s41598-020-76282-0>
- CHEN, Y.-M. et al. Classifying chest CT images as COVID-19 positive/negative using a convolutional neural network ensemble model and uniform experimental design method. **BMC bioinformatics**, v. 22, p. 1–19, 2021. <https://doi.org/10.1186/s12859-021-04083-x>

CHOLLET, F. Building powerful image classification models using very little data. **Keras Blog**, 2016. Disponível em: <<https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html>>. Acesso em: 01 jan. 2020.

CHOLLET, F. **Xception**: Deep learning with depthwise separable convolutions. Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l.]: [s.n.]. 2017. p. 1251–1258.

COHEN, J. P.; MORRISON, P.; DAO, L. COVID-19 image data collection. **arXiv preprint arXiv:2003.11597**, 2020.

<https://doi.org/10.48550/arXiv.2003.11597>

COSTA, J. R. C. A. et al. Técnicas de Seleção de Atributos para Classificação de Imagens Radiográficas. **XXVII Congresso Brasileiro de Engenharia Biomédica -CBEB2020**, p. 632–637, out. 2020. Disponível em: <<https://www.cbeb.org.br/index.html>>.

DA COSTA JUNIOR, C. A. **Proposta de uma Metodologia para Suavização de Ruído em Imagens Mamográficas de Mamas Densas**. Dissertação (Mestrado em Engenharia Biomédica) – Faculdade de Engenharia Elétrica, Universidade Federal de Uberlândia. Uberlândia, p. 80. 2019.

<http://dx.doi.org/10.14393/ufu.di.2019.2036>

DA COSTA JUNIOR, C. A.; PATROCINIO, A. C. **Performance Evaluation of Denoising Techniques Applied to Mammograms of Dense Breasts**. XXVI Brazilian Congress on Biomedical Engineering. [S.l.]: [s.n.]. 2019. p. 369–374.

[https://doi.org/10.1007/978-981-13-2517-5\\_56](https://doi.org/10.1007/978-981-13-2517-5_56)

DA SILVA, I. N.; SPATTI, D. H.; FLAUZINO, R. A. Redes neurais artificiais para engenharia e ciências aplicadas- fundamentos teóricos e aspectos práticos. **2 ed. São Paulo: Artliber**, 2016.

DAI, W.-C. et al. CT imaging and differential diagnosis of COVID-19. **Canadian Association of Radiologists Journal**, v. 71, p. 195–200, 2020. <https://doi.org/10.1177/0846537120913033>

DASH, S.; SENAPATI, M. R. Enhancing detection of retinal blood vessels by combined approach of DWT, Tyler Coye and Gamma correction. **Biomedical Signal Processing and Control**, v. 57, p. 101740, 2020. ISSN: 1746-8094. <https://doi.org/10.1016/j.bspc.2019.101740>

DE LA IGLESIA VAYÁ, M. et al. BIMCV COVID-19+: a large annotated dataset of RX and CT images from COVID-19 patients. **arXiv preprint arXiv:2006.01174**, 2020. <https://doi.org/10.48550/arXiv.2006.01174>.

DEBNATH, L.; SHAH, F. A. **Lecture Notes on Wavelet Transforms**. [S.l.]: Springer, 2017. <https://doi.org/10.1007/978-3-319-59433-0>

DEMIR, A.; YILMAZ, F.; KOSE, O. **Early detection of skin cancer using deep learning architectures: resnet-101 and inception-v3**. 2019 Medical Technologies Congress (TIPTEKNO). [S.l.]: [s.n.]. 2019. p. 1-4.

<https://doi.org/10.1109/TIPTEKNO47231.2019.8972045>

DOS SRIBEIRO, C. et al. How ownership rights over microorganisms affect infectious disease control and innovation: A root-cause analysis of barriers to data sharing as experienced by key stakeholders. **PLoS One**, v. 13, p. e0195885, 2018.

<https://doi.org/10.1371/journal.pone.0195885>

DOS SANTOS, J. C. M. et al. Fundus image quality enhancement for blood vessel detection via a neural network using CLAHE and Wiener filter. **Research on Biomedical Engineering**, v. 36, p. 107–119, 2020. ISSN ISSN: 2446-4740.

<https://doi.org/10.1007/s42600-020-00046-y>

DUC-JACQUET, P. Big Data & Data Science. **Big Data & Data Science**, 2017. Disponível em: <<https://patducjacquet.wordpress.com/2017/07/04/cnn-convolution-neural-network-une-introduction/>>. Acesso em: 01 set. 2021.

EMARA, T. et al. **A Modified Inception-v4 for Imbalanced Skin Cancer Classification Dataset**. 2019 14th International Conference on Computer Engineering and Systems (ICCES). [S.l.]: [s.n.]. 2019. p. 28-33.

<https://doi.org/10.1109/ICCES48960.2019.9068110>

FAN, X. et al. COVID-19 CT Image Recognition Algorithm Based on Transformer and CNN. **Displays**, p. 102150, 2022.

<https://doi.org/10.1016/j.displa.2022.102150>

FAUSETT, L. V. **Fundamentals of neural networks: architectures, algorithms and applications**. [S.l.]: Pearson Education India, 2006.

GUO, T. et al. **Deep wavelet prediction for image super-resolution**. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. [S.l.]: [s.n.]. 2017. p. 104–113.

HASAN, N. et al. DenseNet Convolutional Neural Networks Application for Predicting COVID-19 Using CT Image. **SN Computer Science**, v. 2, p. 389, 2021. ISSN ISSN: 2661-8907.

<https://doi.org/10.1007/s42979-021-00782-7>

HBORHAN. cs230 project codes and results, 2019. Disponível em: <<https://github.com/hborhan/DL-CNN-CheXpert->

data/blob/master/Model\_CheXpert\_DenseNet121\_R1\_TransferLearning.ipynb>. Acesso em: 01 ago. 2021.

HE, K. et al. **Deep residual learning for image recognition**. Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l.]: [s.n.]. 2016. p. 770–778.

HE, K.; SUN, J. **Convolutional neural networks at constrained time cost**. Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l.]: [s.n.]. 2015. p. 5353–5360.

HERTZ, J.; KROGH, A.; PALMER, R. G. **Introduction to the theory of neural computation**. [S.l.]: CRC Press, 2018.

<https://doi.org/10.1201/9780429499661>

HOOVER, A. D.; KOUZNETSOVA, V.; GOLDBAUM, M. Locating blood vessels in retinal images by piecewise threshold probing of a matched filter response. **IEEE Transactions on Medical Imaging**, v. 19, p. 203-210, 2000.

<https://doi.org/10.1109/42.845178>

HUANG, G. et al. **Densely connected convolutional networks**. Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l.]: [s.n.]. 2017. p. 4700–4708.

IRVIN, J. et al. **CheXpert: A large chest radiograph dataset with uncertainty labels and expert comparison**. Proceedings of the AAAI Conference on Artificial Intelligence. [S.l.]: [s.n.]. 2019. p. 590–597.

<https://doi.org/10.1609/aaai.v33i01.3301590>

ISHITAKI, T.; ODA, T.; BAROLLI, L. **A neural network based user identification for Tor networks: Data analysis using Friedman test**. 2016 30th International Conference on Advanced Information Networking and Applications Workshops (WAINA). IEEE. 2016. p. 7–13.

<https://doi.org/10.1109/WAINA.2016.143>

JANSEN, M. **Noise reduction by wavelet thresholding**. [S.l.]: Springer Science & Business Media, v. 161, 2012.

<https://doi.org/10.1007/978-1-4613-0145-5>

KASSANI, S. H. et al. **Diabetic Retinopathy Classification Using a Modified Xception Architecture**. 2019 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT). [S.l.]: [s.n.]. 2019. p. 1-6.

<https://doi.org/10.1109/ISSPIT47144.2019.9001846>

KHATAMI, A. et al. Medical image analysis using wavelet transform and deep belief networks. **Expert Systems with Applications**, v. 86, p. 190–198, 2017.

<https://doi.org/10.1016/j.eswa.2017.05.073>

[KINGMA, D. P.; BA, J. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014.](https://doi.org/10.48550/arXiv.1412.6980)

<https://doi.org/10.48550/arXiv.1412.6980>

KOGILAVANI, S. V. et al. COVID-19 Detection Based on Lung Ct Scan Using Deep Learning Techniques. **Computational and Mathematical Methods in Medicine**, v. 2022, 2022.

<https://doi.org/10.1155/2022/7672196>

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. **Imagenet classification with deep convolutional neural networks**. Advances in neural information processing systems. [S.l.]: [s.n.]. 2012. p. 1097–1105.

KUMARI, S.; RANI, K. Analysis on Various Feature Extraction Methods for Medical Image Classification. [S.l.]: [s.n.], 2020. p. 19-31. ISBN: 978-3-030-46942-9.

[https://doi.org/10.1007/978-3-030-46943-6\\_3](https://doi.org/10.1007/978-3-030-46943-6_3)

LECUN, Y. et al. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, v. 86, p. 2278-2324, 1998.

<https://doi.org/10.1109/5.726791>

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. **nature**, v. 521, p. 436–444, 2015.

<https://doi.org/10.1038/nature14539>

LEVIN, M. D. Digital technology in endodontic practice. In: **Cohen's Pathways of the Pulp**. [S.l.]: Elsevier, 2011. p. 969–1006.

LIN, C. et al. CIR-Net: Automatic Classification of Human Chromosome based on Inception-ResNet Architecture. **IEEE/ACM Transactions on Computational Biology and Bioinformatics**, p. 1-1, 2020.

<https://doi.org/10.1109/TCBB.2020.3003445>

LUO, X. et al. Health classification of Meibomian gland images using keratography 5M based on AlexNet model. **Computer Methods and Programs in Biomedicine**, v. 219, p. 106742, 2022. ISSN: 0169-2607.

<https://doi.org/10.1016/j.cmpb.2022.106742>

MARTIN, D. R. et al. A deep learning convolutional neural network can recognize common patterns of injury in gastric pathology. **Archives of pathology & laboratory medicine**, v. 144, p. 370–378, 2020.

<https://doi.org/10.5858/arpa.2019-0004-OA>

MERRY, R. J. E. Wavelet theory and applications: a literature study. **DCT rapporten**, v. 2005, 2005.

MEYES, R. et al. Ablation studies in artificial neural networks. **arXiv preprint arXiv:1901.08644**, 2019.

<https://doi.org/10.48550/arXiv.1901.08644>

MOREIRA, I. C. et al. Inbreast: toward a full-field digital mammographic database. **Academic radiology**, v. 19, p. 236–248, 2012.

<https://doi.org/10.1016/j.acra.2011.09.014>

MURALI, A. Quora. **Quora**, 2018. Disponível em: <<https://www.quora.com/What-do-you-think-of-Deep-Learning-2>>. Acesso em: 01 dez. 2021.

NARIN, A.; KAYA, C.; PAMUK, Z. Automatic detection of coronavirus disease (covid-19) using x-ray images and deep convolutional neural networks. *Pattern Anal Applic* 24, 1207–1220 (2021).

<https://doi.org/10.1007/s10044-021-00984-y>

NIEMEIJER, M. et al. **Comparative study of retinal vessel segmentation methods on a new publicly available database**. *Medical imaging 2004: image processing*. [S.l.]: [s.n.], 2004. p. 648–656.

<https://doi.org/10.1117/12.535349>

ÖZKAYA, U.; ÖZTÜRK, Ş.; BARSTUGAN, M. Coronavirus (COVID-19) classification using deep features fusion and ranking technique. In: \_\_\_\_\_ **Big Data Analytics and Artificial Intelligence Against COVID-19: Innovation Vision and Approach**. [S.l.]: Springer, 2020. p. 281–295.

[https://doi.org/10.1007/978-3-030-55258-9\\_17](https://doi.org/10.1007/978-3-030-55258-9_17)

OZTURK, S.; OZKAYA, U.; BARSTUGAN, M. Classification of coronavirus images using shrunken features. **medRxiv**, 2020.

<https://doi.org/10.1101/2020.04.03.20048868>

PEOPLE, P. W. A. D. MACHINE LEARNING AUTOMATIZADO: CONCEITOS E IMPORTÂNCIA. **Path we are data people**, 2021. Disponível em: <<https://path.com.br/noticias/o-que-e-machine-learning/>>. Acesso em: 20 set. 2021.

PINHEIRO, J. I. D. et al. **Estatística Básica: A arte de trabalhar com dados**. Rio de Janeiro–RJ. [S.l.]: Editora Elsevier, 2009.

PONTI, M. A.; DA COSTA, G. B. P. Como funciona o deep learning. **arXiv preprint arXiv:1806.07908**, 2018.

<https://doi.org/10.48550/arXiv.1806.07908>

- PRATES, W. R. Curva ROC e AUC em Machine Learning. **https://cienciaenegocios.com/**, nov. 2020. Disponível em: <[https://cienciaenegocios.com/curva-roc-e-auc-em-machine-learning/#Exemplos\\_de\\_Curva\\_ROC\\_para\\_casos\\_hipoteticos](https://cienciaenegocios.com/curva-roc-e-auc-em-machine-learning/#Exemplos_de_Curva_ROC_para_casos_hipoteticos)>. Acesso em: 09 maio 2021.
- RAFAEL, C. Gonzalez, and Richard E. Woods. **Digital image processing**, 2006.
- RAJPURKAR, P. et al. Chexnet: Radiologist-level pneumonia detection on chest x-rays with deep learning. **arXiv preprint arXiv:1711.05225**, 2017.  
<https://doi.org/10.48550/arXiv.1711.05225>
- RANGARAJAN, A. K.; RAMACHANDRAN, H. K. A fused lightweight CNN model for the diagnosis of COVID-19 using CT scan images. **Automatika**, v. 63, p. 171-184, 2022.  
<https://doi.org/10.1080/00051144.2021.2014037>
- RAVI, D. et al. Deep learning for health informatics. **IEEE journal of biomedical and health informatics**, v. 21, p. 4–21, 2016.  
<https://doi.org/10.1109/JBHI.2016.2636665>
- RIBEIRO, C. D. S.; KOOPMANS, M. P.; HARINGHUIZEN, G. B. Threats to timely sharing of pathogen sequence data. **Science**, v. 362, p. 404–406, 2018.  
<https://doi.org/10.1126/science.aau5229>
- ROBERTS, M. et al. Common pitfalls and recommendations for using machine learning to detect and prognosticate for COVID-19 using chest radiographs and CT scans. **Nature Machine Intelligence**, v. 3, p. 199–217, 2021.  
<https://doi.org/10.1038/s42256-021-00307-0>
- RUUSKA, S. et al. Evaluation of the confusion matrix method in the validation of an automated system for measuring feeding behaviour of cattle. **Behavioural processes**, v. 148, p. 56–62, 2018.  
<https://doi.org/10.1016/j.beproc.2018.01.004>
- SAEZ, D. M.; OTHERS. Avaliação da influência dos formatos DICOM e JPEG na reprodutibilidade de pontos cefalométricos em Telerradiografia digital em Norma Frontal, 2009.
- SALEEM, H. N.; SHEIKH, U. U.; KHALID, S. A. **Classification of Chest Diseases from X-ray Images on the CheXpert Dataset**. Innovations in Electrical and Electronic Engineering. Singapore: Springer Singapore. 2021. p. 837–850.  
[https://doi.org/10.1007/978-981-16-0749-3\\_64](https://doi.org/10.1007/978-981-16-0749-3_64)
- SANTOS, J. C. M. D. **Técnicas de aprendizado de máquinas na segmentação automática de vasos sanguíneos da retina**. Tese (Doutorado em Engenharia Elétrica) - Universidade Federal de Uberlândia. Uberlândia, p. 136. 2022.

<http://doi.org/10.14393/ufu.te.2022.129>

SAVELLI, B. et al. A multi-context CNN ensemble for small lesion detection. **Artificial Intelligence in Medicine**, v. 103, p. 101749, 2020.

<https://doi.org/10.1016/j.artmed.2019.101749>

SETHY, P. K.; BEHERA, S. K. Detection of coronavirus disease (covid-19) based on deep features. **Preprints**, v. 2020030300, p. 2020, 2020.

<https://doi.org/10.20944/preprints202003.0300.v1>

SHERRY, S. T. et al. dbSNP: the NCBI database of genetic variation. **NIH National Library of Medicine**. Disponivel em: <<https://doi.org/10.1093/nar/29.1.308>>. Acesso em: 2020.

SHIRAZI, A. Z.; CHABOK, S. J. S. M.; MOHAMMADI, Z. A novel and reliable computational intelligence system for breast cancer detection. **Medical & biological engineering & computing**, v. 56, p. 721–732, 2018.

<https://doi.org/10.1007/s11517-017-1721-z>

SHUKLA, K. K.; TIWARI, A. K. **Efficient algorithms for discrete wavelet transform: with applications to denoising and fuzzy inference systems**. [S.l.]: Springer Science & Business Media, 2013.

<https://doi.org/10.1007/978-1-4471-4941-5>

SIMON, J. H. M. et al. Managing severe acute respiratory syndrome (SARS) intellectual property rights: the possible role of patent pooling. **Bulletin of the World Health Organization**, v. 83, p. 707–710, 2005.

SIMONYAN, K.; ZISSERMAN, A. Very deep convolutional networks for large-scale image recognition. **arXiv preprint arXiv:1409.1556**, 2014.

<https://doi.org/10.48550/arXiv.1409.1556>

SKANSI, S. **Introduction to Deep Learning: from logical calculus to artificial intelligence**. [S.l.]: Springer, 2018.

<https://doi.org/10.1007/978-3-319-73004-2>

SUMMERS, R. NIH Clinical Center:dataset of 32,000 CT images. **NHI National Institutes of Health**, 2020. Disponivel em: <<https://www.nih.gov/news-events/news-releases/nih-clinical-center-releases-dataset-32000-ct-images>>. Acesso em: 01 jan. 2021.

SZEGEDY, C. et al. **Going deeper with convolutions**. Proceedings of the IEEE conference on computer vision and pattern recognition. [S.l.]: [s.n.]. 2015. p. 1–9.

SZEGEDY, C. et al. **Rethinking the Inception Architecture for Computer Vision**. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). [S.l.]: [s.n.]. jun. 2016.

SZEGEDY, C. et al. **Inception-v4, inception-resnet and the impact of residual connections on learning**. Thirty-first AAAI conference on artificial intelligence. [S.l.]: [s.n.]. 2017.

TER-SARKISOV, A. COVID-CT-Mask-Net: prediction of COVID-19 from CT scans using regional features. **Applied Intelligence**, 2022. ISSN: 1573-7497.

<https://doi.org/10.1007/s10489-021-02731-6>

TRIWIJYOYA, B. K.; ADILA, A. Analysis of medical image resizing using bicubic interpolation algorithm. **J. Ilmu Komput**, v. 14, p. 20–29, 2021.

<https://doi.org/10.24843/JIK.2021.v14.i01.p03>

URBANIAK, I.; WOLTER, M. Quality assessment of compressed and resized medical images based on pattern recognition using a convolutional neural network. **Communications in Nonlinear Science and Numerical Simulation**, v. 95, p. 105582, 2021. ISSN: 1007-5704.

<https://doi.org/10.1016/j.cnsns.2020.105582>

WANG(A), X. et al. A Weakly-supervised Framework for COVID-19 Classification and Lesion Localization from Chest CT. **IEEE Transactions on Medical Imaging**, 2020.

<https://doi.org/10.1109/TMI.2020.2995965>

WANG(C), G. et al. A noise-robust framework for automatic segmentation of COVID-19 pneumonia lesions from CT images. **IEEE Transactions on Medical Imaging**, v. 39, p. 2653–2663, 2020.

<https://doi.org/10.1109/TMI.2020.3000314>

WANG, L.; LIN, Z. Q.; WONG, A. Covid-net: A tailored deep convolutional neural network design for detection of covid-19 cases from chest x-ray images. **Scientific Reports**, v. 10, p. 1–12, 2020.

<https://doi.org/10.1038/s41598-020-76550-z>

WANG, X. et al. ChestX-ray: Hospital-Scale Chest X-ray Database and Benchmarks on Weakly Supervised Classification and Localization of Common Thorax Diseases. In: **Deep Learning and Convolutional Neural Networks for Medical Imaging and Clinical Informatics**. [S.l.]: Springer International Publishing, 2019. p. 369–392.

WANG, X. et al. HER2-ResNet: A HER2 classification method based on deep residual network. **Technology and Health Care**, p. 1–10, 2022.

<https://doi.org/10.3233/THC-228020>

WANI, M. A. et al. **Advances in deep learning**. [S.l.]: Springer, v. 57, 2020.

<https://doi.org/10.1007/978-981-13-6794-6>

WILLIAMS, T.; LI, R. **Advanced image classification using wavelets and convolutional neural networks**. 2016 15th IEEE international conference on machine learning and applications (ICMLA). [S.l.]: [s.n.]. 2016. p. 233–239.

<https://doi.org/10.1109/ICMLA.2016.0046>

WU, J. Institute of Genomics, Chinese Academy of Science, China National Center for Bioinformation & National Genomics Data Center, 2020. Disponível em: <<https://bigd.big.ac.cn/ncov/?lang=en>>. Acesso em: jun. 2020.

WU, X. et al. Deep learning-based multi-view fusion model for screening 2019 novel coronavirus pneumonia: a multicentre study. **European Journal of Radiology**, p. 109041, 2020.

<https://doi.org/10.1016/j.ejrad.2020.109041>

YANG, S. et al. Deep learning for detecting corona virus disease 2019 (COVID-19) on high-resolution computed tomography: a pilot study. **Annals of Translational Medicine**, v. 8, 2020.

<https://doi.org/10.21037/atm.2020.03.132>

ZANIN, M. et al. An early stage researcher's primer on systems medicine terminology. **Network and systems medicine**, v. 4, p. 2–50, 2021.

<https://doi.org/10.1089/nsm.2020.0003>

ZEILER, M. D.; FERGUS, R. **Visualizing and understanding convolutional networks**. European conference on computer vision. [S.l.]: [s.n.]. 2014. p. 818–833.

[https://doi.org/10.1007/978-3-319-10590-1\\_53](https://doi.org/10.1007/978-3-319-10590-1_53)

ZHANG, J. et al. Covid-19 screening on chest x-ray images using deep learning based anomaly detection. **arXiv preprint arXiv:2003.12338**, 2020.

ZIMMERMAN, D. W.; ZUMBO, B. D. Relative power of the Wilcoxon test, the Friedman test, and repeated-measures ANOVA on ranks. **The Journal of Experimental Education**, v. 62, p. 75–86, 1993.

<https://doi.org/10.1080/00220973.1993.9943832>