



UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA QUÍMICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA



**SISTEMAS INTELIGENTES BASEADOS EM REDES NEURAIS
ARTIFICIAIS APLICADOS AO PROCESSAMENTO DE IMAGENS NO
CÁLCULO DA DISTRIBUIÇÃO DO TAMANHO DAS PARTÍCULAS**

Maria Gabriela Marques Vidal Pereira

Uberlândia – MG

2021



UNIVERSIDADE FEDERAL DE UBERLÂNDIA
FACULDADE DE ENGENHARIA QUÍMICA
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA



**SISTEMAS INTELIGENTES BASEADOS EM REDES NEURAIIS
ARTIFICIAIS APLICADOS AO PROCESSAMENTO DE IMAGENS NO
CÁLCULO DA DISTRIBUIÇÃO DO TAMANHO DAS PARTÍCULAS**

Maria Gabriela Marques Vidal Pereira

Tese de Doutorado apresentada ao Programa de Pós-Graduação em Engenharia Química da Universidade Federal de Uberlândia como parte dos requisitos necessários à obtenção do título de Doutora em Engenharia Química.

Uberlândia – MG

2021

TESE DE DOUTORADO SUBMETIDA AO PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA QUÍMICA DA UNIVERSIDADE FEDERAL DE UBERLÂNDIA COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA OBTENÇÃO DO TÍTULO DE DOUTOR EM ENGENHARIA QUÍMICA, EM 17 DE DEZEMBRO DE 2021.

Dedico esta tese à minha família por estarem sempre presentes me dando o suporte e apoio necessário e ao meu orientador não só pelas orientações, mas pelas palavras amigas, pelos incentivos e pela amizade.

| | |
|---|-----------|
| LISTA DE FIGURAS | 7 |
| LISTA DE TABELAS | 8 |
| NOMENCLATURA | 9 |
| RESUMO | 10 |
| ABSTRACT | 11 |
| CAPÍTULO 1 - INTRODUÇÃO | 13 |
| 1. OBJETIVOS | 15 |
| CAPÍTULO 2 - REVISÃO BIBLIOGRÁFICA | 18 |
| 1. MATERIAIS PARTICULADOS | 18 |
| 2. USO DO PROCESSAMENTO DE IMAGENS NO ESTUDO DOS MATERIAIS PARTICULADOS | 19 |
| 3. USO DE REDES NEURAIS NO PROCESSAMENTO DE IMAGENS | 24 |
| 4. REDES NEURAIS DENSAS (DEEP LEARNING) | 32 |
| CAPÍTULO 3 - METODOLOGIA | 37 |
| 1. REPRESENTAÇÃO ESQUEMÁTICA DA AMOSTRAGEM <i>IN-LINE</i> PARA O CÁLCULO DO PSD | 37 |
| 2. LINGUAGEM DE PROGRAMAÇÃO E MATERIAIS | 38 |
| 3. AQUISIÇÃO DAS IMAGENS | 38 |
| 4. CONFIGURAÇÃO DA REDE NEURAL ARTIFICIAL | 40 |
| 5. DETERMINAÇÃO DO PSD | 42 |
| CAPÍTULO 4 – RESULTADOS E DISCUSSÕES | 46 |
| 1. BASE DE DADOS | 46 |
| 2. SEGMENTAÇÃO AUTOMÁTICA | 48 |
| 3. ANÁLISE DO DESEMPENHO DAS RNAs AVALIADAS | 50 |
| 4. CAPACIDADE DE OBTENÇÃO DO PSD ASSOCIADO AS RNAs SELECIONADAS | 51 |

| | |
|--|-----------|
| 5. ANÁLISE DO DESEMPENHO FINAL DO SISTEMA DE MICROSCOPIO COM PDS PROPOSTO | 52 |
| 6. AVALIAÇÃO COMPARATIVA DO PSD ENTRE O SISTEMA EM TEMPO REAL PROPOSTO E O PROCESSAMENTO DE IMAGEM POR SOFTWARES EM LABORATORIO | 56 |
| CAPÍTULO 5 – CONCLUSÕES | 58 |
| REFERÊNCIAS BIBLIOGRÁFICAS | 59 |
| ANEXO I | 61 |
| ANEXO II | 63 |

LISTA DE FIGURAS

| | |
|--|----|
| FIGURA 1: DIVISÕES DA INTELIGÊNCIA ARTIFICIAL – FONTE: (MENEZES, 2018) | 13 |
| FIGURA 2: REPRESENTAÇÃO NUMÉRICA DE UMA IMAGEM 10X10 PIXELS COM 255 TONS DE CINZA (FONTE: IGATHINATHANE ET AL. 2009) | 20 |
| FIGURA 3: TÉCNICAS DE ANÁLISE DA IMAGEM (FONTE: (VERONEZI ET AL. 2011) | 21 |
| FIGURA 4: ILUSTRAÇÃO DE UM NEURÔNIO BIOLÓGICO (FONTE: JAIN, MAO, AND MOHIUDDIN 1996) | 25 |
| FIGURA 5: ILUSTRAÇÃO DE UM NEURÔNIO ARTIFICIAL. (FONTE: JAIN, MAO, AND MOHIUDDIN 1996) | 26 |
| FIGURA 6: SINAIS DE ENTRADA E DE SAÍDA EM UMA CADEIA DE NEURÔNIOS ARTIFICIAIS INTERLIGADOS (FONTE: JAIN, MAO, AND MOHIUDDIN 1996) | 27 |
| FIGURA 7: MATRIZ CONFUSÃO (FONTE: AUTORA) | 28 |
| FIGURA 8: FUNÇÃO DE ATIVAÇÃO SIGMOIDE | 30 |
| FIGURA 9: FUNÇÃO DE ATIVAÇÃO TANGENTE HIPERBÓLICA | 30 |
| FIGURA 10: FUNÇÃO DE ATIVAÇÃO LINEAR RETIFICADA (RELU) | 31 |
| FIGURA 11: AJUSTE DOS PESOS SINÁPTICOS DA RNA POR BACKPROPAGATION (FONTE: JAIN, MAO, AND MOHIUDDIN 1996) | 32 |
| FIGURA 12: ESQUEMA ILUSTRATIVO PARA DESCREVER A FORMA COMO O CÓRTEX VISUAL RECEBE A INFORMAÇÃO DE UMA IMAGEM (FONTE: ALBAWI, MOHAMMED, AND ALZAWI 2017) | 33 |
| FIGURA 13: EXEMPLO DE UMA CNN (FONTE: ALBAWI, MOHAMMED, AND ALZAWI 2017) | 34 |
| FIGURA 14: AMOSTRAGEM DO AÇÚCAR IN-LINE E ETAPAS PARA A OBTENÇÃO DO PSD (FONTE: AUTORA) | 38 |
| FIGURA 15: MODELO ESQUEMÁTICO DA CONFIGURAÇÃO EXPERIMENTAL PARA O SISTEMA DE AQUISIÇÃO DE IMAGENS (FONTE: AUTORA) | 39 |
| FIGURA 16: FLUXOGRAMA PROPOSTO PARA A DETERMINAÇÃO DO PSD (FONTE: AUTORA) | 43 |
| FIGURA 17: FLUXOGRAMA PARA O CÁLCULO DO PSD (FONTE: AUTORA) | 44 |
| FIGURA 18: BASE DE DADOS DAS IMAGENS UTILIZADAS NESTE TRABALHO (FONTE: AUTORA) | 47 |
| FIGURA 19: IMAGENS DE GRÃOS E NÃO-GRÃOS SEGMENTADAS AUTOMATICAMENTE PELO CÓDIGO DESENVOLVIDO (FONTE: AUTORA) | 49 |
| FIGURA 20: DISTRIBUIÇÃO ACUMULADA DA DISTRIBUIÇÃO DO TAMANHO DE PARTÍCULAS DA AMOSTRA 3. CURVA COR LARANJA DADOS CNN E CURVA AZUL DADOS IMAGE J. | 54 |
| FIGURA 21: DISTRIBUIÇÃO ACUMULADA DA DISTRIBUIÇÃO DO TAMANHO DE PARTÍCULAS DA AMOSTRA 7. CURVA COR LARANJA DADOS CNN E CURVA AZUL DADOS IMAGE J. | 55 |
| FIGURA 22: DISTRIBUIÇÃO ACUMULADA DA DISTRIBUIÇÃO DO TAMANHO DE PARTÍCULAS DA AMOSTRA 8. CURVA COR LARANJA DADOS CNN E CURVA AZUL DADOS IMAGE J. | 56 |
| FIGURA 23: HISTOGRAMAS DO PSD: A) AMOSTRA 3. $R^2=0,919$, B) AMOSTRA 7. $R^2=0,874$ C) AMOSTRA 8. $R^2=0,856$ E D) AMOSTRA 3. $R^2=0,92$ | 57 |

LISTA DE TABELAS

| | |
|---|----|
| TABELA 1: REDES NEURAI AVALIADAS 1 | 41 |
| TABELA 2: COMPARAÇÃO DOS CUSTOS PARA A REALIZAÇÃO DA ANÁLISE GRANULOMÉTRICA DE FORMA MANUAL E AUTOMÁTICA CONSIDERANDO COMO REFERÊNCIA UM ANO DE OPERAÇÃO | 48 |
| TABELA 3: MÉDIA DOS RESULTADOS DAS REDES NEURAI AVALIADAS | 51 |
| TABELA 4: MÉTRICAS AVALIADAS DAS TRÊS REDES NEURAI COM MELHOR DESEMPENHO PARA TRÊS AMOSTRAS | 52 |
| TABELA 5: MATRIZ CONFUSÃO PARA A AMOSTRA 3 | 52 |
| TABELA 6: MATRIZ CONFUSÃO PARA A AMOSTRA 7 | 53 |
| TABELA 7: MATRIZ CONFUSÃO PARA A AMOSTRA 8 | 53 |
| TABELA 8: (TABELA 3 COMPLETA) MÉDIA DOS RESULTADOS DAS REDES NEURAI AVALIADAS | 61 |

NOMENCLATURA

| Sigla | Definição |
|--------|--|
| USB | <i>Universal Serial Bus</i> |
| PSD | Distribuição do Tamanho de Partículas |
| IA | Inteligência Artificial |
| RNA | Rede Neural Artificial |
| ML | <i>Machine Learning</i> |
| DL | <i>Deep Learning</i> |
| VHP | <i>Very High Polarization</i> |
| DIP | Processamento Digital de Imagens |
| MS | Peneiramento Mecânico |
| DNN | <i>Deep Neural Network</i> |
| CNN | <i>Convolutional Neural Network</i> |
| CCD | <i>Charged Coupled dDevice</i> |
| RP | Reconhecimento de Padrões |
| OCR | <i>Optical Character Recognizer</i> |
| tanh | Função Tangente Hiperbólica |
| ReLU | Função Linear Retificada |
| LIMS | <i>Laboratory Information Management Systems</i> |
| PIMS | <i>Plant Information Management Systems</i> |
| ICUMSA | Comissão Internacional de Métodos Uniformes de Análise do Açúcar |

RESUMO

Os materiais na forma de pó e os particulados ocupam a segunda posição em escala de uso nas indústrias químicas, imediatamente atrás da água, por serem utilizados na produção de energia, produção de alimentos, agricultura, petroquímica, materiais de construção entre outros setores. Nesse sentido, tecnologias que realizam a caracterização e análise de partículas, por exemplo, a determinação da distribuição do tamanho de partículas (PSD), são fundamentais para a avaliação das propriedades e comportamento desses materiais. Na presente tese, o controle de qualidade de cristais de açúcar por imagem em tempo real foi avaliado como exemplo de estudo de materiais particulados tendo como objetivo a obtenção do PSD utilizando um dispositivo de baixo custo que utiliza imagens de cristais aglomerados obtidas através de um computador seguido do processamento das mesmas com o uso de aprendizado de máquina. Foi constatado que um microscópio portátil com saída USB de baixo custo é adequado para gerar imagens com resolução e detalhes suficientes para que as amostras possam ser avaliadas na forma de imagens e as dimensões dos cristais possam ser extraídas destas imagens de forma satisfatória. Por outro lado, em relação ao uso de aprendizagem de máquina, foi constatado que redes neurais artificiais do tipo *feedforward* e do tipo de convolução se mostraram capazes de identificar os cristais com acurácia superior a 90%, sendo que nos melhores resultados obtidos, redes de convolução, foram capazes de identificar os cristais nas imagens com acurácia de 96,9%. Como conclusão final pode-se dizer que o uso integrado de microscopia de baixo custo seguido da identificação dos cristais em imagens segmentadas e posterior avaliação das dimensões de cristais, representa uma forma promissora de gerar um sistema em tempo real de análise da distribuição granulométrica de materiais aglomerados.

Palavras-chave: processamento de imagens, rede neural artificial, materiais particulados.

ABSTRACT

Powder and granular materials can be considered the second most important priority, after water, on scale of use in industrial and other fields of human activities including energy production, nanomaterials technology, food production, agriculture, petrochemical, geology, mining, metallurgy, pharmaceutical industries. In this context, technologies that carry out the characterization and analysis of particles, for example, the determination of particle size distribution (PSD), are fundamental for evaluating the properties and behavior of these materials. In the present thesis, the quality control of sugar crystals by real-time was evaluated as an example of a study of granular materials with the objective of obtaining the PSD using a low-cost device that uses images of agglomerated crystals obtained through a computer. followed by processing them using machine learning. It was found that a portable microscope with a low-cost USB output is able for generating images with good resolution and detail, so that the samples can be evaluated like images and the dimensions of the crystals can be satisfactorily extracted from these images. On the other hand, in relation to the use of machine learning, it was found that feedforward and convolution type artificial neural networks were able to identify crystals with an accuracy greater than 90%, and in the best results obtained, networks of convolution, were able to identify the crystals in the images with an accuracy of 96.9%. As a final conclusion, it can be said that the integrated use of low-cost microscopy followed by the identification of crystals in segmented images and subsequent evaluation of crystal dimensions, represents a promising way to generate a real-time system for analyzing the particle size distribution of agglomerated materials.

Keywords: Image processing, artificial neural network, powder and granular materials.

CAPÍTULO 1 - INTRODUÇÃO

CAPÍTULO 1 - INTRODUÇÃO

A Inteligência Artificial (IA) é uma área de pesquisa responsável por simular a inteligência e o comportamento humano através do desenvolvimento de softwares, indo além da programação de ordens específicas para tomar decisões de forma autônoma a fim de perceber e decidir quais caminhos seguir, de forma racional. Essa importante área da computação tem ganhado grande destaque no meio científico graças à sua capacidade de resolver problemas complexos, como, classificação de padrões, aproximação de funções, previsão de séries temporais e otimização.

A Figura 1 mostra as divisões da IA que serão abordadas neste trabalho de pesquisa.

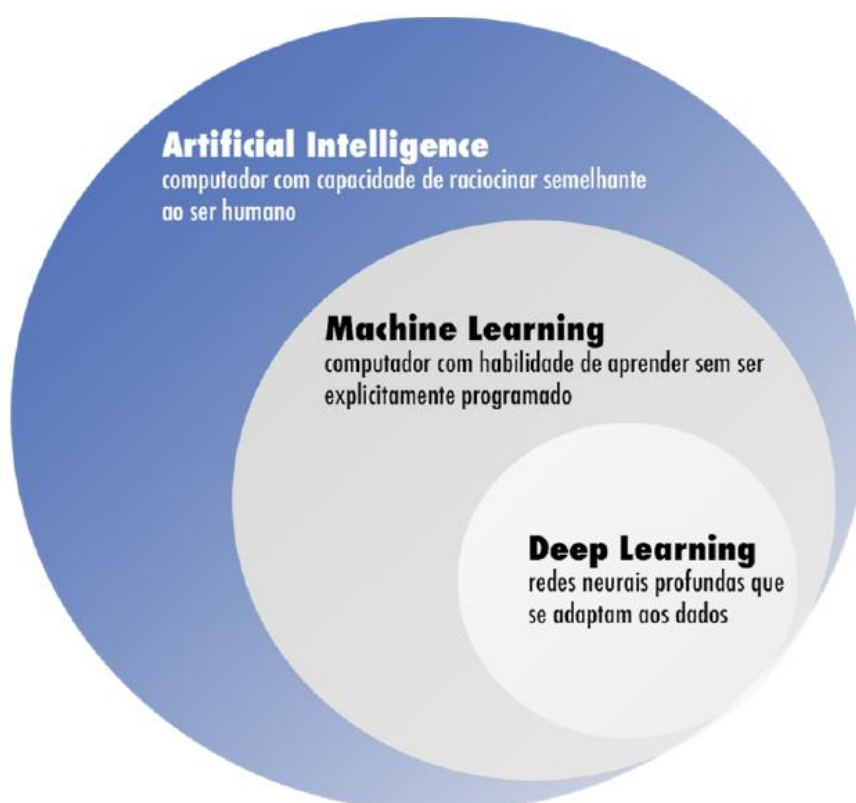


Figura 1: Divisões da Inteligência Artificial – Fonte: (Menezes, 2018)

Machine Learning (ML), o que chamamos em português de Aprendizado de Máquina, é uma ramificação da IA que apresenta um conjunto de estratégias expressas em códigos computacionais e conferem aos computadores, celulares e outras máquinas, a capacidade de aprender com os dados, de forma que possam identificar padrões, fazer previsões e executar tarefas sem serem explicitamente para isso programados (Simon 2013). Entretanto, foi o

desenvolvimento das técnicas que imitam o comportamento do cérebro humano, as chamadas Redes Neurais Artificiais (RNA), subárea do ML, conhecida como Deep Learning (DL), que trouxeram a estratégia de dar cognição e inteligência às máquinas. As RNAs têm a capacidade de se adaptarem e de aprenderem a realizar uma certa tarefa a partir de um conjunto de dados e foram inspiradas no funcionamento das redes de neurônios biológicos do cérebro.

Os materiais na forma de pó e os particulados desempenham um papel muito importante dentro da indústria química, pois ocupam a segunda posição em escala de uso, imediatamente atrás da água. São bastante utilizados na produção de energia, produção de alimentos, agricultura, petroquímica, materiais de construção, geologia, mineração, metalurgia, indústrias farmacêutica (Boschetto and Giordano 2012; Frei and Kruis 2018a; Maiti et al. 2017; Xie et al. 2020). Nesse sentido, tecnologias que realizam a caracterização e análise de partículas, por exemplo, a determinação da distribuição do tamanho de partículas (PSD), são fundamentais para a avaliação das propriedades e comportamento desses materiais (Sudarsan et al. 2018; Oktay and Gurses 2019; Bai et al. 2021).

O controle dos parâmetros e das propriedades desses materiais é monitorado durante o processo produtivo em várias indústrias de alimentos, incluindo a produção de sacarose a partir da cana-de-açúcar que é uma cultura semiperene e representa um dos maiores *commodities* em todo o mundo (Fang et al. 2019; Shendryk, Davy, and Thorburn 2021). As características do cristal de açúcar, incluindo a distribuição do tamanho das partículas, são importantes parâmetros do controle de qualidade das indústrias de sacarose, incluindo açúcar refinado, VHP e açúcar de beterraba, cuja qualidade dos cristais é avaliada utilizando o PSD entre outros parâmetros (J. Chen et al. 2015; Crestani et al. 2021; Shendryk, Davy, and Thorburn 2021)

O monitoramento do PSD é comumente obtido por meio do peneiramento mecânico (MS) e do processamento digital de imagens (DIP). Embora, o peneiramento seja um método clássico, o DIP apresenta várias vantagens incluindo baixo custo, resultados rápidos, a não necessidade de instalações laboratoriais, a capacidade de fornecer informações adicionais sobre o formato da partícula, adequado para o caso quando os materiais estão aglomerados, materiais que reagem quimicamente com a peneira e materiais que apresentam frações de partículas finas. Os computadores também podem oferecer uma ampla gama de análise de dados, são capazes de armazenar uma grande quantidade de dados, automação do processo e redução do custo operacional (Boschetto and Giordano 2012; Drugan and Talbi 2014; Maiti et al. 2017; Cardona et al. 2018; Yang, Ren, and Tan 2020; Oh et al. 2021)

Poucos estudos foram realizados utilizando microscopia na avaliação do PSD (*Particle size distribution*) em processo industrial. Alguns estudos utilizam a microscopia como uma ferramenta importante na análise do cimento, na avaliação da microestrutura e na determinação do PDS (Ferrari et al., 2012; Frei & Kruis, 2018; Gunning & Morris, 2018), mas utilizam equipamentos mais sofisticados, adequados para análises laboratoriais em escala. No DIP, as características do microscópio USB portátil, incluindo a qualidade das imagens e o baixo custo, são fatores importantes para fornecer uma chance de fabricação de dispositivos de baixo custo para a avaliação do PSD.

1. OBJETIVOS

Neste trabalho é proposto um sistema de controle de qualidade de material particulado por meio da análise da distribuição granulométrica automática *in-line* a partir da análise de imagens utilizando como referência para o estudo o particulado açúcar VHP. Como objetivos específicos podem ser destacados.

1. Montagem de sistema de microscopia de baixo custo conectado a um computador que permite que as imagens de cristais de açúcar armazenadas no computador possam ser processadas em tempo real para análise da distribuição granulométrica das mesmas.
2. Desenvolvimento e treinamento de uma rede neural que possibilita identificar o material particulado nas imagens de forma a favorecer a determinação das dimensões do material particulado sem a necessidade de entregar a amostra no laboratório.
3. Integração da rede neural em um algoritmo que possibilita gerar a distribuição granulométrica a partir das imagens armazenadas em tempo real.

Esta tese está organizada da seguinte maneira: no Capítulo 2, apresenta-se uma revisão bibliográfica que reúne os fundamentos necessários para o desenvolvimento do modelo de

análise proposto para o cálculo do PSD. No Capítulo 3, detalha-se a metodologia utilizada na determinação do PSD para geração de dados de treinamento e teste da rede neural. No Capítulo 4, são apresentados os resultados e a eficiência do modelo no cálculo do PSD. Finalmente, no Capítulo 5, são apresentadas as conclusões e as sugestões para trabalhos futuros.

CAPÍTULO 2 – REVISÃO BIBLIOGRÁFICA

CAPÍTULO 2 - REVISÃO BIBLIOGRÁFICA

O objetivo deste capítulo consiste em apresentar os fundamentos teóricos relacionados ao tema desta tese, iniciando-se com uma breve introdução sobre os materiais particulados bem como a utilização do processamento de imagens no estudo desses materiais, a importância da análise da Distribuição do Tamanho de Partículas (PSD) na caracterização dos materiais particulados, seguida pela maneira escolhida para avaliar essa análise, que foi o processamento de imagens. Em seguida, será abordado os fundamentos teóricos das redes neurais artificiais, e, por fim, uma explicação sobre os dois tipos de redes neurais testadas a saber: *Deep Neural Network* (DNN) e a *Convolutional Neural Network* (CNN).

1. MATERIAIS PARTICULADOS

Materiais particulados são todos aqueles formados por partículas: nessa denominação incluem desde os materiais granulares (sementes, cereais, areias), pós (alimentícios, farmacêuticos, minérios), poeiras (poluentes, contaminantes) e até nano partículas. Cerca de 60% dos produtos na indústria química são fabricados na forma particulada, as aplicações industriais envolvendo esses produtos são muitas, e em setores muito diversos. Baseado na tecnologia de fabricação, o controle de qualidade tende a ser mais ou menos sofisticado e rigoroso. Disponível em: <<http://www.ppgeq.ufscar.br/ppgeq/areas-de-pesquisa-1/sistemas-particulados-1/sistemas-particulados>>. Acesso em: 21 de Setembro de 2021.

Segundo Boschetto e Giordano (2012), o primeiro parâmetro geométrico atribuído aos particulados foi o tamanho da partícula. Objetos quase esféricos, podem ser definidos por seu diâmetro, no entanto, as partículas são normalmente prováveis de ter forma irregular e não esférica, então um único grão tem um número infinito de dimensões lineares e cada uma quantifica o tamanho da partícula em apenas uma direção.

Como existe uma grande variedade de tamanho de partículas nos campos da geologia, construção, mineração, metalurgia, agricultura, etc, é essencial conhecer a forma e o tamanho dos materiais particulados utilizados em diversos processos industriais para o aprimoramento, caracterização e controle industrial. Um dos métodos mais simples e popular para a estimativa do tamanho das partículas é o peneiramento mecânico ou análise granulométrica. No entanto, de acordo com Abhik et al. (2017), o peneiramento convencional apresenta certas limitações.

A análise da granulometria utilizando peneiras, consome muito tempo e não dá nenhuma ideia sobre a forma das partículas. Duas partículas com características de forma completamente diferentes podem ter o mesmo tamanho ao passarem pela abertura da peneira. A peneira mecânica também é inadequada no caso de materiais pegajosos ou materiais que podem reagir quimicamente com a tela de separação da peneira, além de demandar custos operacionais e instalações laboratoriais

Segundo Boschetto e Giordano (2012), a microscopia é a única técnica de classificação de partículas amplamente utilizada em que partículas individuais são diretamente observadas e medidas. Por muitos anos, a microscopia manual foi o único método prático na determinação da forma das partículas. Esse método é altamente trabalhoso e depende da habilidade do operador. Medir um número estatisticamente significativo de partículas manualmente é quase impossível e a determinação do tamanho e da forma das partículas que estão aglomeradas é bastante complexa e demorada.

Há uma grande demanda por métodos automatizados de análise no controle do processo produtivo dentro da indústria para melhorar a velocidade de resposta, uniformizar a análise, diminuir custo operacional, entre outras.

O açúcar é um material particulado produzido em mais de 100 países, tendo o Brasil como maior produtor e exportador mundial. No Brasil, o mesmo representa a 3ª cultura temporária, estando atrás somente da soja e o do milho. Açúcar bruto cuja polarização é maior que 99,2°S é chamado de VHP (*very high polarization*), ou seja, polarização muito alta. Para atender as demandas, o controle dos parâmetros e das propriedades do açúcar é monitorado durante o processo produtivo. Esse controle de qualidade deve ser rigoroso de forma a atender as especificações por norma e uma das análises especificadas por norma é a distribuição do tamanho de partículas (PSD).

2. USO DO PROCESSAMENTO DE IMAGENS NO ESTUDO DOS MATERIAIS PARTICULADOS

Segundo Gonzales e Woods (1992), uma imagem pode ser definida como uma função $f(x,y)$, onde o valor nas coordenadas espaciais x e y corresponde ao brilho (intensidade) da

imagem nessa coordenada. A menor unidade de uma imagem digital é denominada pixel. Um pixel é a representação numérica da luminosidade de um ponto da imagem.

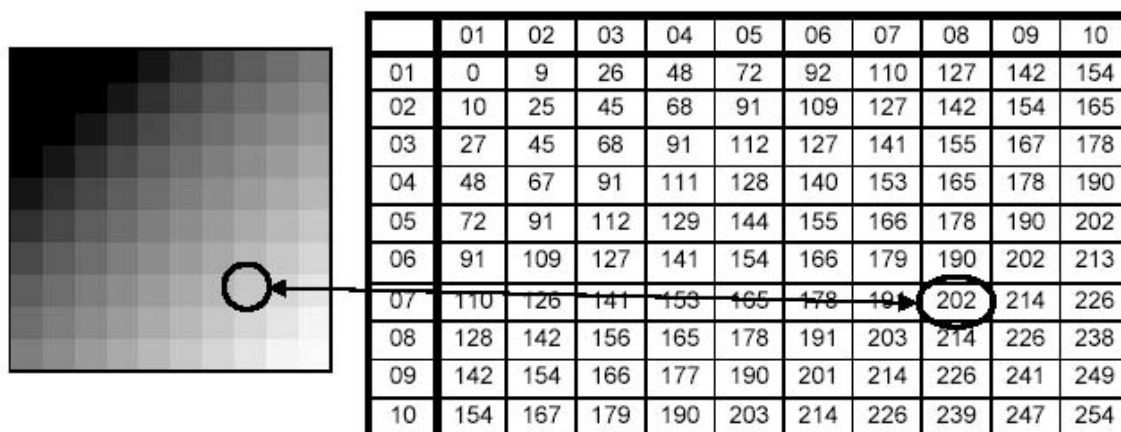


Figura 2: Representação numérica de uma imagem 10x10 pixels com 255 tons de cinza (Fonte: Igathinathane et al. 2009)

De acordo com Holst e Lomheim (2011), a imagem nas câmeras digitais e nos microscópios, são capturadas por um dispositivo eletrônico conhecido como CCD (*charged coupled device*). Esse hardware consiste de uma matriz de sensores de luz que recebe a informação luminosa através das lentes, convertendo-a em uma matriz de valores digitais com linhas e colunas assim como demonstrado na Figura 2. Cada posição nessa matriz contém um valor discreto que é proporcional a intensidade luminosa medida nessa posição, correspondendo a informação mais básica da composição da imagem, o pixel. Essa luminosidade é convertida em cor e varia para tons de cinza, entre 0 e 255, sendo o valor 0 referente a cor preta e o 255 correspondendo ao branco.

A área de processamento de imagens há muito tempo tem sido objeto de interesse crescente por permitir viabilizar grande quantidade de aplicações, principalmente no que diz respeito ao aprimoramento de informações para interpretação humana e análise automática por computador de informações extraídas de uma imagem.

Existem 3 técnicas para se processar uma imagem: (1) processamento de baixo nível, com funções que podem ser vistas como reações automáticas, ou seja, reações que não requerem comportamento inteligente; (2) processamento de nível intermediário, com processos de

extração e caracterização de componentes em uma imagem e (3) processamento de alto nível, que envolve os processos de reconhecimento e interpretação. Os três tipos de análise estão apresentados na Figura 3 [Gonzales e Woods (1992)].

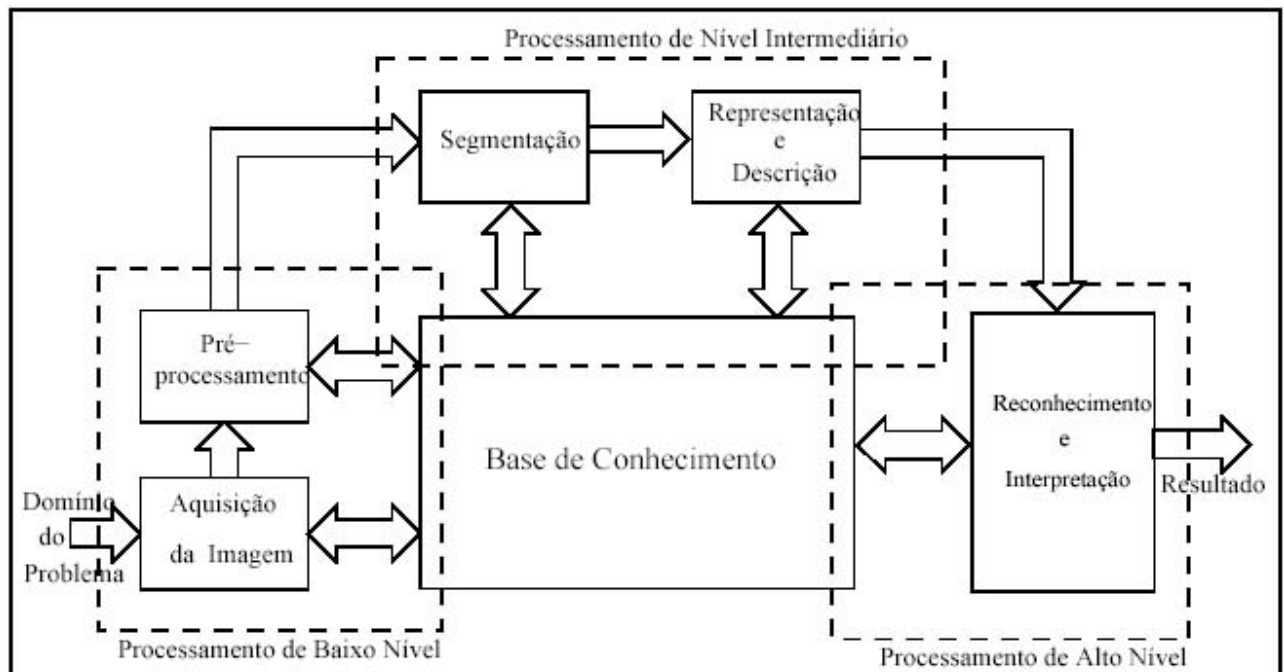


Figura 3: Técnicas de análise da imagem (Fonte: (Veronezi et al. 2011))

Segundo Gonzales e Woods (1992), dentro do processamento digital de imagens de alto nível, 6 passos são percorridos até o reconhecimento e interpretação dessas imagens, a saber:

- Aquisição de imagem

O primeiro passo do processo requer apenas um sensor de imagens (câmera digital ou microscópio) com a capacidade de digitalizar, isto é, converter o sinal elétrico capturado, na sua forma digital.

- Pré-processamento

A função chave do pré-processamento é melhorar a imagem, com o objetivo de aumentar as chances de sucesso dos próximos passos [Gonzales e Woods (1992)]. Nesta etapa, são utilizadas técnicas para aumento de contraste, remoção de ruídos, realce e normalização.

- Segmentação

De um modo geral, a segmentação subdivide uma imagem de entrada em partes constituintes ou objetos. Cada uma destas partes pode ser utilizada no reconhecimento de padrões da próxima etapa ou a segmentação pode ser utilizada para eliminar partes da imagem que não serão avaliadas.

- Representação e descrição

Geralmente, a saída do estágio da segmentação é com dados brutos de pixel. Neste caso pode ser necessário converter os dados para uma forma conveniente, possibilitando o processamento por computador. A descrição, também chamada de seleção de característica, lida com a extração de características que resultam em alguma classe de objetos.

- Reconhecimento

Após feita a separação de classe de objetos com características semelhantes, passamos a uma fase na qual o interesse é identificar o que cada uma dessas classes representa, e assim identificá-las com um respectivo valor. O reconhecimento de padrões (RP) é importante devido às ocorrências na vida humana tomarem forma de padrões. A formação da linguagem, o modo de falar, o desenho das figuras, o entendimento das imagens, tudo envolve padrões. O RP é uma tarefa complexa, em que o homem busca, sempre, avaliar as situações em termos de padrões das circunstâncias que as constituem, descobrir relações existentes no meio, para melhor entendê-lo e adaptar-se. A técnica da abordagem neural no RP, como o próprio nome diz, o reconhecimento é realizado utilizando-se Redes Neurais Artificiais.

- Interpretação

Para Gonzales e Woods (1992), interpretação envolve a fixação de significado a um grupo de objetos reconhecidos. Em outras palavras, estamos interessados em dar significados a imagem. Interpretação de imagens computadorizadas é um processo extremamente complexo. As dificuldades aparecem tanto pela quantidade de dados a serem processadas como também pela falta de ferramentas de processamento fundamental para receber os dados iniciais para gerar os resultados desejados.

O processamento de imagens de materiais particulados tem gerado um grande número de trabalhos de pesquisa junto à comunidade científica. Boschetto e Giordano (2012) desenvolveram uma metodologia *offline* para caracterizar as propriedades do pó de aço e

alumina com base na análise de imagens digitais, capaz de fornecer informações estatisticamente significativas de tamanho e forma em uma única medição. Sudarsan et al. (2018) estudaram as proporções relativas das partículas do solo, chamada textura, através do processamento de imagens digitais. Sabe-se que é de suma importância conhecer as propriedades físicas do solo para os processos agrícolas, ambiental e de engenharia. No entanto, a caracterização do tamanho das partículas em laboratório apresenta uma série de desafios em termos de tempo, trabalho, dificuldade e/ou custo envolvido na análise. Seu estudo traz o desenvolvimento de um algoritmo de visão computacional projetado para processar as imagens em alta resolução coletadas por meio de um sistema portátil de aquisição de imagens (tanto no laboratório quanto no campo) para caracterizar o tamanho das partículas do solo.

Chen et al. (2019) utilizaram como objeto de pesquisa o arroz vermelho e desenvolveram um sistema de visão de máquina para inspecionar grãos de arroz defeituosos que estavam quebrados, farináceos, danificados ou manchados. Imagens de infravermelho próximo das amostras do arroz foram coletadas. Um classificador de máquina de vetor suporte foi aplicado para identificar os grãos de arroz danificados nas imagens. Os resultados mostram que a precisão de reconhecimento para grãos quebrados, danificados e manchados atingiu 99,3%, 96,3% e 93,6%, respectivamente. Além disso, o tempo médio de execução do método proposto foi de 0,15 segundos, com quatro tipos de defeitos detectados de uma só vez, se tratando assim, de um método com potencial significativo para ser aplicado na detecção rápida e precisa da qualidade do arroz.

Segundo Maiti et al. (2017) enquanto a análise granulométrica gasta de 30 a 45 min, o processamento de imagem digital (DIP) pode gerar o mesmo resultado em um tempo muito menor (de 1 a 4 min). Como câmeras digitais e computadores são muito comuns atualmente, o método baseado em DIP é mais conveniente para o uso geral e é relativamente barato. O método de peneiramento convencional fornece apenas a distribuição do peso das partículas. No processamento de imagem, também podemos medir vários outros parâmetros, como área, diâmetro do eixo principal, diâmetro do eixo menor, centro de gravidade e diâmetro equivalente.

A avaliação do PSD é comumente obtida por meio de peneiramento mecânico (MS) e processamento digital de imagem (DIP). Embora o MS seja um método clássico, o DIP apresenta várias vantagens incluindo baixo custo, resultados rápidos, a não necessidade de instalações laboratoriais, bem como fornecer informações adicionais sobre o formato da

partícula, é adequado no caso de materiais pegajosos e materiais que podem reagir quimicamente com a peneira, adequado também para materiais que apresentam frações finas. Os computadores podem oferecer uma ampla gama de análise de dados, podem armazenar uma grande quantidade de dados, executar a automação do processo e contribuir para a redução de custo operacional (Boschetto & Giordano, 2012; Cardona et al., 2018; Drugan & Talbi, 2014; Maiti et al., 2017; Oh et al., 2021; Yang et al., 2020).

Relativamente poucos estudos foram realizados utilizando a microscopia para o cálculo do PSD. Na verdade, a maioria dos estudos com microscopia a colocam como uma ferramenta importante na análise do cimento, avaliação da microestrutura e poucos na determinação do PSD (Ferrari et al., 2012; Frei & Kruis, 2018; Gunning & Morris, 2018). Entretanto, os trabalhos que utilizam a microscopia no cálculo do PSD requerem equipamentos mais sofisticados, mais adequados para serem utilizados dentro do laboratório. No DIP, as características de microscopia USB portátil apresentado neste presente trabalho, incluindo a qualidade das imagens e o baixo custo, são fatores importantes para viabilizar o emprego de dispositivos de baixo custo na análise do PSD.

3. USO DE REDES NEURAIIS NO PROCESSAMENTO DE IMAGENS

A todo momento nosso cérebro trabalha para o reconhecimento de padrões, por exemplo, ao ler um texto, para reconhecer o rosto de uma pessoa, distinguir um cachorro de um gato, ler as diversas caligrafias e até mesmo quando um médico interpreta um exame de eletrocardiograma.

Dentre as técnicas de reconhecimento de padrões mais utilizadas, destacam-se a abordagem estatística e a abordagem conexionista, esta utiliza redes neurais artificiais (RNAs). No caso da abordagem conexionista, os classificadores são mais complexos, capazes de se adaptarem a qualquer tipo de distribuição de dados. São baseados no funcionamento das estruturas neurais inteligentes que aprendem por meio de exemplos. A representação de conhecimentos nas redes conexionistas, como o próprio nome diz, é fortemente ligada a noção de conexão entre os neurônios (elementos processadores de informação) que interagem uns com os outros através dessas ligações. O modelo conexionista é uma simplificação da estrutura do

nosso cérebro, onde se simulam elementos que são como os neurônios e suas conexões, chamadas de sinapses.

A construção das RNAs tem inspiração nos neurônios biológicos e nos sistemas nervosos. Uma RNA está codificada em estrutura de rede, onde se destacam as conexões (sinapses) entre as unidades (neurônios) que a compõem.

Nas redes artificiais obtidas por simulação em computadores, um peso sináptico é associado a cada conexão (valor numérico) que caracteriza a força da conexão entre dois neurônios. O aprendizado em uma RNA é realizado por um processo de adaptação dos seus pesos sinápticos. Na Figura 4 é mostrada a imagem de um neurônio biológico e na Figura 5 a sua representação por meio de um modelo, denominado neurônio artificial ou neurônio matemático.

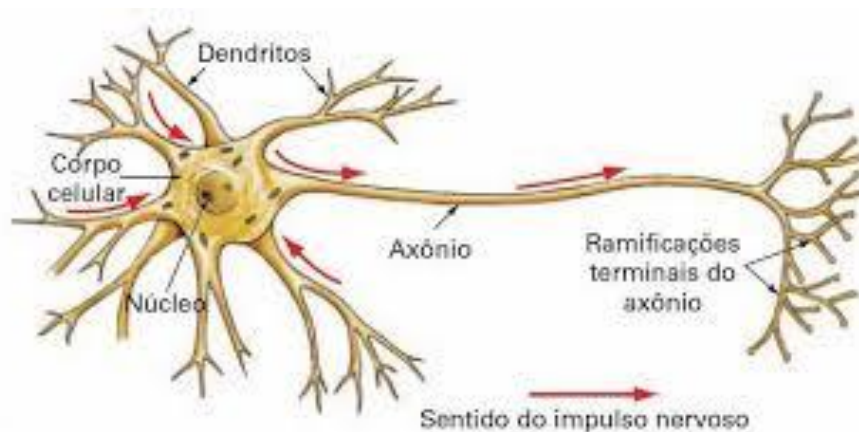


Figura 4: Ilustração de um neurônio biológico (Fonte: Jain, Mao, and Mohiuddin 1996)

De acordo com a Figura 4, o neurônio biológico, é dividido em 3 partes: dendritos, soma ou corpo celular e axônio, seguindo o sentido do sinal nervoso. Simplificadamente, os dendritos captam os sinais elétricos de outros neurônios ou do meio externo, o corpo celular processa essa informação criando um potencial de ativação que poderá ou não ser enviado ao axônio.

De forma análoga ao neurônio biológico, no neurônio artificial ou matemático (Figura 5), os sinais de entrada (x_1, x_2, \dots, x_n) correspondem aos sinais que chegam aos dendritos do neurônio biológico. Os pesos sinápticos (w_1, w_2, \dots, w_n) são associados aos sinais de entrada, o resultado é a geração de um pulso representado na Figura 5 pelo somatório. Esse pulso passa por um limiar de ativação, porque dependendo da força do pulso, pode ser que não ative a

função de ativação em seguida. Finalmente, após a ação da ativação, o neurônio envia essa informação para outro neurônio ou para a saída.

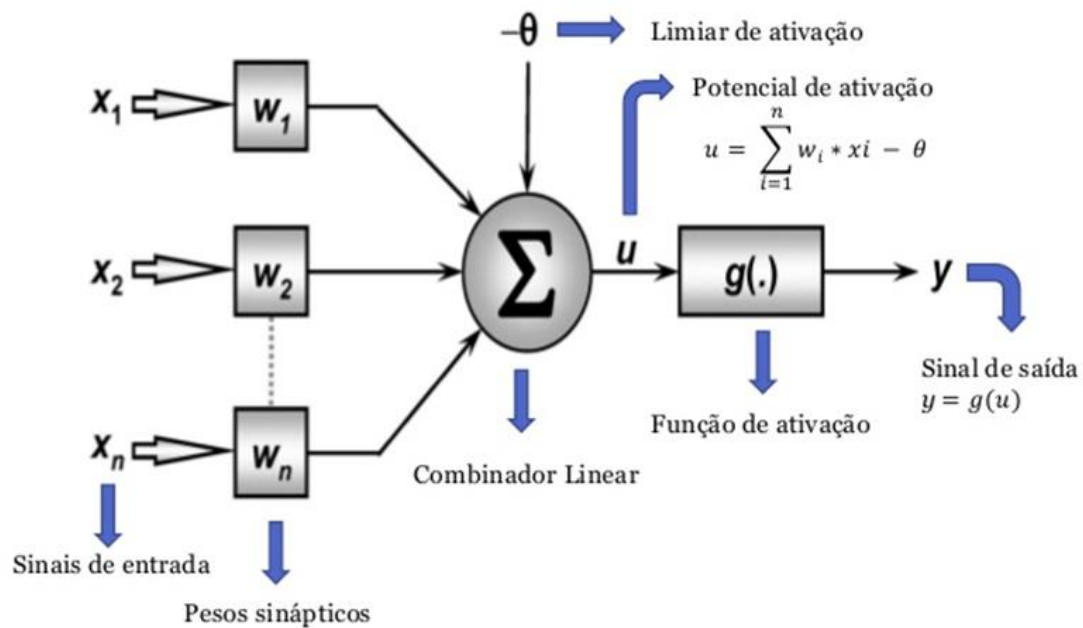


Figura 5: Ilustração de um neurônio artificial. (Fonte: Jain, Mao, and Mohiuddin 1996)

Informalmente, uma RNA é um sistema composto pela combinação de vários neurônios em uma ou mais camadas, que podem conter um ou mais neurônios que são interligados através das sinapses. Os neurônios que recebem excitações do exterior são chamados neurônios de entrada e os que, de alguma forma, tem suas respostas utilizadas para alterar o mundo exterior são chamados de neurônios de saída. Os neurônios que não são nem entrada nem saída são conhecidos como neurônios internos. Esses neurônios internos à rede têm grande importância e são denominados como *hidden*, isto é, ocultos (Figura 6).

As redes neurais artificiais são utilizadas para resolver problemas de inteligência artificial e podem ser aplicadas em diversas áreas do conhecimento. Algumas das principais aplicações destas redes neurais junto ao Processamento de Imagens são:

- Reconhecimento de padrões: sistemas OCR (Optical Character Recognizer), reconhecimento de assinaturas e de impressões digitais;
- Classificação de padrões: controle de qualidade, classificação e agrupamento de cenas/imagens com propriedades similares;

- Tratamento de imagens: filtros inteligentes usados na melhoria da qualidade visual das imagens (ressaltar elementos, detectar contornos, corrigir o foco/distorções);
- Visão artificial: tarefas de alto nível, como o reconhecimento de pessoas, de expressões faciais, de gestos, de elementos e de objetos presentes em um ambiente, etc.

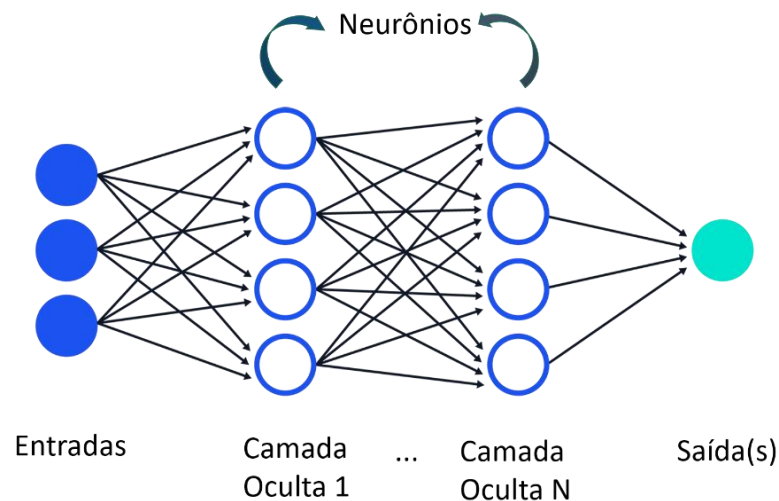


Figura 6: Sinais de entrada e de saída em uma cadeia de neurônios artificiais interligados (Fonte: Jain, Mao, and Mohiuddin 1996)

A propriedade mais importante de uma rede neural artificial é sua capacidade de aprender a partir do ambiente em que está inserida e então melhorar seu processo através da aprendizagem. A rede aprende sobre o ambiente após um processo iterativo de ajustes, de modo que, idealmente, a rede se torna mais instruída sobre seu ambiente após cada iteração do processo de aprendizagem (Haikin, 2001).

Para que o modelo da rede neural obtenha um bom desempenho, são empregados alguns algoritmos de treinamento que fazem os ajustes dos parâmetros da RNA. Neste trabalho, o algoritmo de aprendizado utilizado foi o supervisionado, também chamado de método de correção dos erros, no qual existe uma pessoa, capaz de fornecer à rede neural uma resposta desejada para um vetor de treinamento inicial. Composto por técnicas que usam exemplos de entradas (variáveis independentes) e saídas (variáveis dependentes) a fim de aprender como fazer previsões.

As métricas de avaliação do desempenho do modelo são baseadas na comparação entre as classes preditas pelo modelo e as classes verdadeiras do problema e objetivam mensurar quão distante da classificação perfeita está o modelo. Uma maneira de representar visualmente a performance de um modelo é através da matriz confusão (Figura 7), que indica quantas ocorrências existem em cada grupo: verdadeiro positivo (VP), verdadeiro negativo (VN), falso positivo (FP), falso negativo (FN).

Além da matriz confusão, a acurácia (A) é uma métrica muito utilizada para medir o desempenho do aprendizado e informa quantas ocorrências foram classificadas corretamente, independente da classe, conforme Equação 1.

$$A = \frac{VP + VN}{VP + VN + FP + FN} \quad (1)$$

| | | Classe Predita | |
|-------------|---|------------------------------|------------------------------|
| | | P | N |
| Classe Real | P | Verdadeiro Positivo VP | Falso Negativo FN |
| | N | Falso Positivo FP | Verdadeiro Negativo TN |

Figura 7: Matriz confusão (Fonte: Autora)

A precisão (P) é definida, conforme Equação 2, pela razão entre a quantidade de ocorrências classificadas corretamente como positivas e o total de ocorrências classificadas como positivas.

$$P = \frac{VP}{VP + FP} \quad (2)$$

A taxa de verdadeiro positivo (R , *Recall*) é definida, conforme Equação 3, pela razão entre a quantidade de ocorrências classificadas corretamente como positivas e a quantidade de ocorrências que são, de fato, positivas. Já a taxa de verdadeiro negativo ou especificidade (S , *Specificity*) é a proporção de ocorrências negativas classificadas corretamente, conforme Equação 4.

$$R = \frac{VP}{VP + FN} \quad (3)$$

$$S = \frac{VN}{VN + FP} \quad (4)$$

O F1 score (F1) é definido, conforme Equação 5, pela média harmônica entre a precisão (P) e a taxa de verdadeiro positivo (R). Essa métrica é especialmente útil para medir o desempenho global de problemas com classes não balanceadas.

$$F1 = \frac{2VP}{2VP + FP + FN} \quad (5)$$

Outra característica básica das redes neurais artificiais é a função de ativação, cuja função é restringir a amplitude de saída de um neurônio. Tipicamente, algumas funções tem imagem limitada ao intervalo real $[0,1]$, enquanto outras ao intervalo real $[-1,1]$ (Haikin, 2001). O uso de funções de ativação não-lineares torna a rede neural uma ferramenta poderosa. Tratando-se de redes com mais de uma camada de neurônios, vê-se que o problema do ajuste dos pesos sinápticos será resolvido pela utilização da técnica do gradiente descendente na busca de um erro mínimo global. Se a função de ativação fosse linear, a derivada seria uma constante e não apresentaria nenhuma informação na busca pelo melhor valor para os pesos sinápticos (Ludwing e Costa, 2007). Dentre as funções de ativação utilizadas, pode-se destacar, a função Sigmoid cujo gráfico está representado pela Figura 8 e sua equação representada pela Equação 6, a função tangente hiperbólica (\tanh) (Figura 9 e Equação 7) e a linear retificada (ReLU) (Figura 10 e Equação 8).

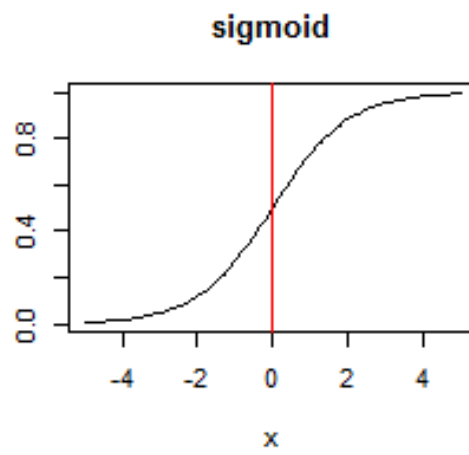


Figura 8: Função de ativação Sigmoide

$$\varphi(x) = \frac{1}{1 + e^{-x}} \quad (6)$$

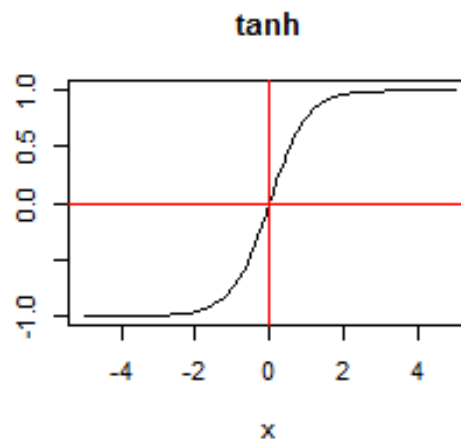


Figura 9: Função de ativação tangente hiperbólica

$$\varphi(x) = \tanh(x) \quad (7)$$

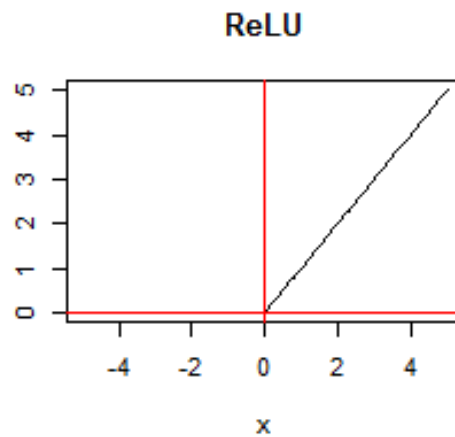


Figura 10: Função de ativação linear retificada (ReLU)

$$\varphi(x) = \begin{cases} 0 & \text{para } x < 0 \\ x & \text{para } x \geq 0 \end{cases} \quad (8)$$

A maneira pela qual os neurônios de uma rede neural estão estruturados, está intimamente ligado com o algoritmo de aprendizagem. Em geral, podemos identificar três classes de arquiteturas de rede fundamentalmente diferentes (Haikin, 2001).

- Redes Alimentadas Adiante com Camada Única - rede neural artificial acíclica que possui uma camada de nós de entrada e uma camada de nós computacionais de saída.
- Redes Alimentadas Diretamente com Múltiplas Camadas - rede neural artificial acíclica que possui uma camada de nós de entrada, uma ou mais camadas de nós computacionais ocultos e uma camada de nós computacionais de saída.
- Redes Recorrentes ou *Feedforward*- rede neural artificial que possui pelo menos um laço de realimentação e pelo menos uma camada de nós.

Durante o processo de ajuste dos pesos sinápticos, uma das estratégias mais utilizadas é o método chamado gradiente descendente, em um processo de retropropagação ou *backproagation* dos erros. O *backproagation* é realizado através da diferença (erro) entre a resposta final esperada e a resposta calculada pelo modelo, seguido do cálculo do gradiente descendente para cada peso que compõe a entrada de cada camada. Esse processo é repetido sequencialmente até que a camada de entrada seja corrigida, conforme mostrado na Figura 11.

Nesta figura, o processo de *backpropagation* é representado por setas vermelhas para todas as respostas finais, bem como para a primeira linha de dados, de modo a representar a correção dos pesos das outras camadas, que ocorre de traz para frente a partir da resposta.

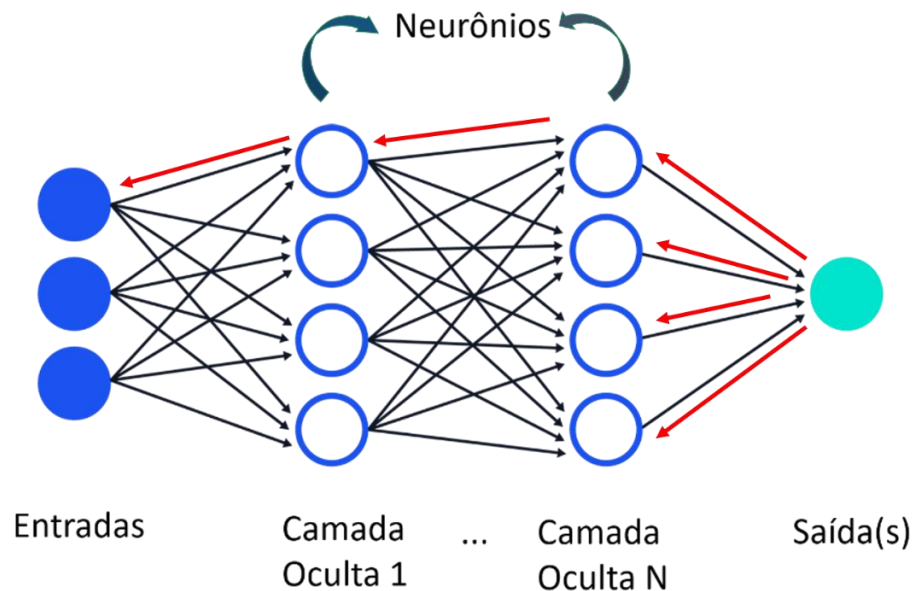


Figura 11: Ajuste dos pesos sinápticos da RNA por backpropagation (Fonte: Jain, Mao, and Mohiuddin 1996)

Ele é dividido basicamente em duas etapas: a propagação e a retro propagação. A propagação consiste na aquisição dos dados pela camada de entrada e sua propagação por toda rede, produzindo um conjunto de saída. Esse conjunto é comparado com valores desejados. A partir daí começa o processo de retro propagação que consiste nos ajustes dos pesos segundo a diferença produzida, para fazer com que a resposta da rede se aproxime do desejado.

4. REDES NEURAIS DENSAS (DEEP LEARNING)

Há vários tipos de RNAs. Algumas possuem uma arquitetura mais simples, na qual há apenas duas camadas, sendo uma camada de entrada e outra de saída. Por outro lado, existem arquiteturas mais complexas, com a existência de três ou mais camadas. As camadas adicionais são conceituadas como intermediárias ou ocultas como citado anteriormente. Nessas redes, as

camadas de entrada têm a função do recebimento dos sinais advindos dos padrões de treinamento. A associação das características, bem como a separação das classes são feitas pelos neurônios das camadas ocultas. Fica a cargo da camada de saída a apresentação dos resultados finais da rede.

1.1. Rede Neural Artificial (DNN *Feedforward*)

Em particular, uma rede neural artificial com n camadas ocultas, conforme mostrado na Figura 11, é classificada como *deep learning*, ou seja, densa. Será uma rede *feedforward* quando a informação da RNA tiver somente uma direção (entrada para a saída).

1.2. Rede Neural de Convolução (CNN)

As redes de convolução (CNN) foram inspiradas na estrutura dos sistemas visuais orientados ao reconhecimento de objetos. Hubel e Weizel estudaram o córtex visual e ganharam o prêmio Nobel de Medicina no ano de 1981 com esse trabalho. No córtex visual, a informação não se propaga tão densa e seus neurônios tem um pequeno campo receptivo.

A CNN vai buscar construir a imagem como um todo por partes. Cada camada do córtex visual vai receber essas informações locais e essa imagem vai se formando como um todo à medida que essa informação vai chegando na camada mais profunda do córtex visual conforme mostrado na Figura 12.

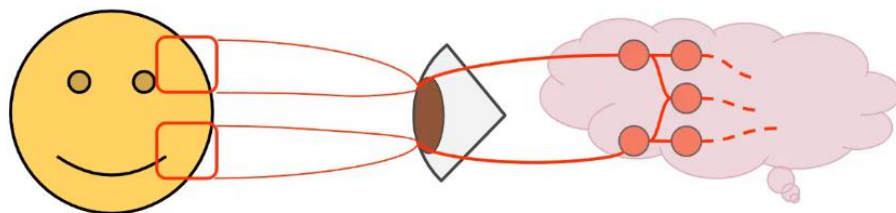


Figura 12: Esquema ilustrativo para descrever a forma como o córtex visual recebe a informação de uma imagem (Fonte: Albawi, Mohammed, and Alzawi 2017)

Como a CNN vai buscar construir essa imagem como um todo, por partes, ela inicia através de padrões mais simples e vai avançando até obter a imagem como um todo. O cérebro

também lê uma imagem dessa forma, porque reduz a quantidade de neurônios utilizados na tarefa. Isso significa que o aprendizado se baseia em conceitos complexos, através da decomposição dos mesmos elementos mais simples, cujo treinamento é realizado pela busca gradual de um grupo de filtros de convolução, que são determinados de acordo com a base de dados de treinamento e a estrutura geral da rede.

A CNN tem uma estrutura bem diferente das RNAs. Nas camadas de convolução a informação passa por vários filtros com a função de acentuar padrões locais e ao mesmo tempo reduzir a dimensão dos dados originais como mostrado na Figura 13. Na parte mais profunda das convoluções, espera-se que os dados num espaço dimensional reduzido conttenham as informações suficientes sobre esses padrões locais para atribuir um valor semântico ao dado original. Depois da convolução, uma função de ativação é utilizada para fazer a operação não-linear citada anteriormente na seção das funções de ativação. Nesse exemplo de CNN, temos a camada de entrada, a camada de saída, duas camadas de convolução e duas camadas com *max pooling*.

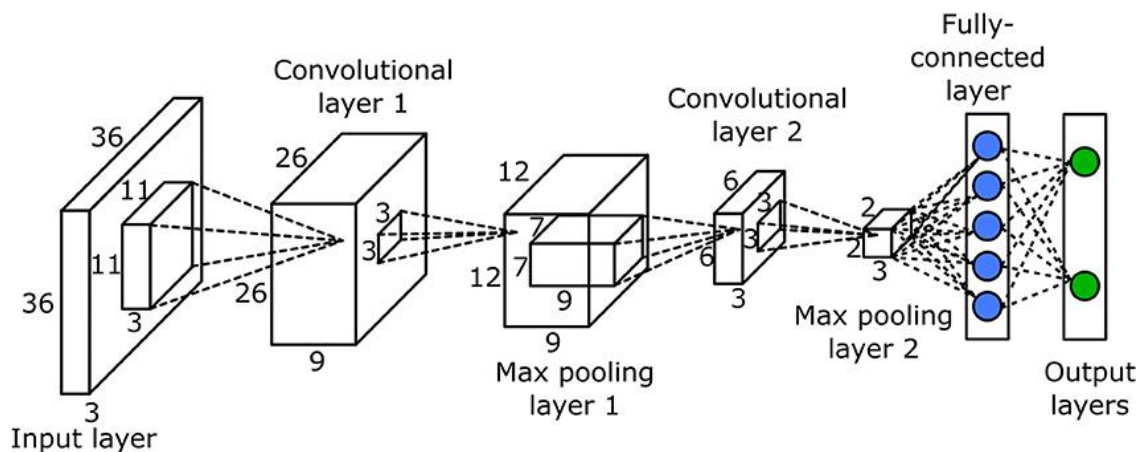


Figura 13: Exemplo de uma CNN (Fonte: Albawi, Mohammed, and Alzawi 2017)

Na construção de códigos computacionais de CNN, as bibliotecas Keras e Pytorch se destacam entre as mais utilizadas interfaces de programação de aplicação (*Application Programming Interface* ou API em inglês) por serem códigos abertos de fácil uso, suportarem diferentes arquiteturas de redes neurais e possibilitarem o uso em diferentes tipos de projetos incluindo plataformas móveis como celulares, internet, uso nas nuvens e computação de alto

desempenho que utiliza processamento paralelo e unidades gráficas de processamento (GPU em inglês).

A biblioteca Keras, que foi a interface de programação de aplicação escolhida para a implementação da CNN desta tese, pode ser utilizada em conjunto com diferentes plataformas de aprendizado de máquina. Um exemplo de plataforma é o TensorFlow que facilita o uso do Keras em processamento de alto desempenho em fácil implementação como mostra as linhas de código abaixo que descrevem a definição da rede de convolução no TensorFlow utilizando convolução com filtros kernel 3x3 e total de imagens de 32, 64 e 128.

```
model = Sequential()
model.add(Conv2D(input_shape=(Img_Size, Img_Size, 3), filters=32, kernel_size=(3, 3)))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D())

model.add(Conv2D(filters=64, kernel_size=(3, 3)))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D())

model.add(Conv2D(filters=128, kernel_size=(3, 3)))
model.add(BatchNormalization())
model.add(Activation('relu'))
model.add(MaxPooling2D())

model.add(Flatten())
model.add(Dense(units=80))
model.add(Activation('relu'))
model.add(Dropout(0.2))
model.add(Dense(units=K))
model.add(Activation('softmax'))
```

CAPÍTULO 3 - METODOLOGIA

CAPÍTULO 3 - METODOLOGIA

Este capítulo apresenta a metodologia utilizada no desenvolvimento deste trabalho de pesquisa, validando de forma prática a utilização de redes neurais no processamento de imagens para o cálculo da distribuição do tamanho de partículas de açúcar VHP.

O açúcar é um material particulado produzido em mais de 100 países, tendo o Brasil o maior produtor e exportador mundial (CONAB- Companhia Nacional de Abastecimento). Seu processo produtivo, passa por um rigoroso controle de qualidade a fim de atender as especificações por norma. Sendo a distribuição do tamanho de partículas (PSD) uma das análises exigidas pela norma ICUMSA GS2/9-37 para esse controle, este trabalho de pesquisa traz o desenvolvimento de uma metodologia para o cálculo do PSD *in-line* utilizando dispositivos de baixo custo.

1. REPRESENTAÇÃO ESQUEMÁTICA DA AMOSTRAGEM *IN-LINE* PARA O CÁLCULO DO PSD

Este presente trabalho propõe a determinação da distribuição do tamanho de partículas (PSD) do açúcar VHP através de um sistema de microscopia compacto e de baixo custo, a fim de avaliar sua qualidade. Esse sistema pode ser instalado no ambiente industrial logo após o processo de secagem, conforme mostrado na Figura 14, como também após a etapa de cristalização ou mesmo antes da secagem, sendo que em cada uma destes diferentes pontos de amostragem tem-se a possibilidade de favorecer o processo de forma distinta que contempla avaliação da qualidade do produto final (instalação após o secador), controle do processo de cristalização (instalação após o cristalizador) ou avaliação de possíveis problemas como quantidade excessiva de mel nos cristais (avaliação antes da secagem).

Na Figura 14 é detalhada o controle dos cristais na amostragem do produto seco que representa o que foi desenvolvido nesta tese. Após a parte da amostragem (Fig. 14b) é possível observar a representação da avaliação do PSD (Figura 14a) e detalhes do processo que consiste no açúcar primeiramente passar pela etapa de cristalização (Figura 14c), em seguida ser desidratado (Figura 14b), finalmente, ser avaliado na forma seca no que se refere a geração do PSD (Figura 14a)

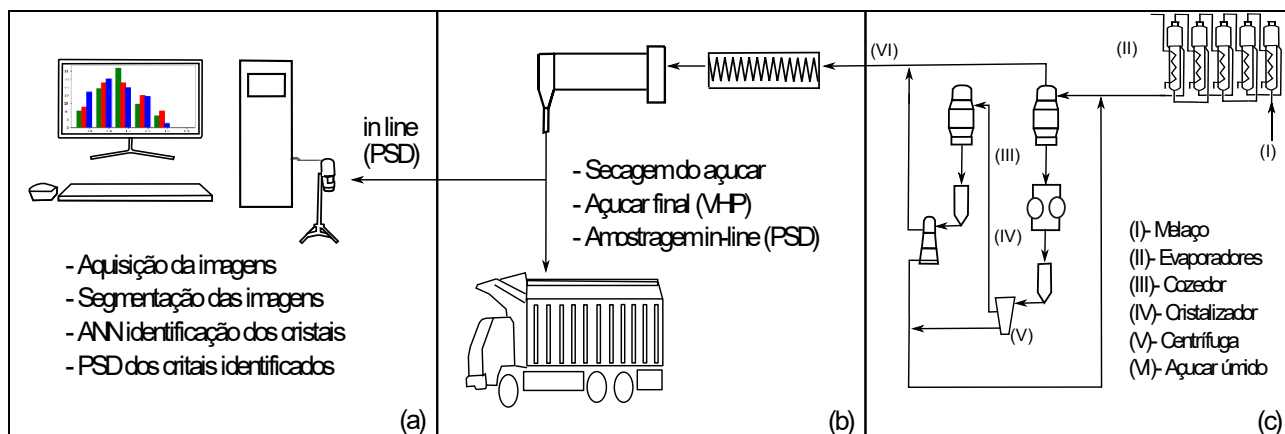


Figura 14: Amostragem do açúcar in-line e etapas para a obtenção do PSD (Fonte: Autora)

2. LINGUAGEM DE PROGRAMAÇÃO E MATERIAIS

A linguagem de programação escolhida para execução do projeto foi o Python. Dentre as principais características que motivaram tal escolha estão: apresenta uma sintaxe intuitiva, apresenta uma plataforma de programação de código aberto, além de ter disponível para consulta diversas fontes de informação e uma grande comunidade ativa.

Nove amostras de açúcar VHP foram adquiridas de duas unidades industriais da região do Triângulo Mineiro para o desenvolvimento deste trabalho. Imediatamente após o recebimento da amostra no laboratório, as mesmas foram armazenadas em sacos plásticos lacrados para não receberem umidade e colocados dentro de uma caixa para que os cristais não sofressem quebras.

3. AQUISIÇÃO DAS IMAGENS

Para cada amostra de açúcar VHP, 5 imagens foram produzidas, totalizando 45 imagens de cristais aglomerados.

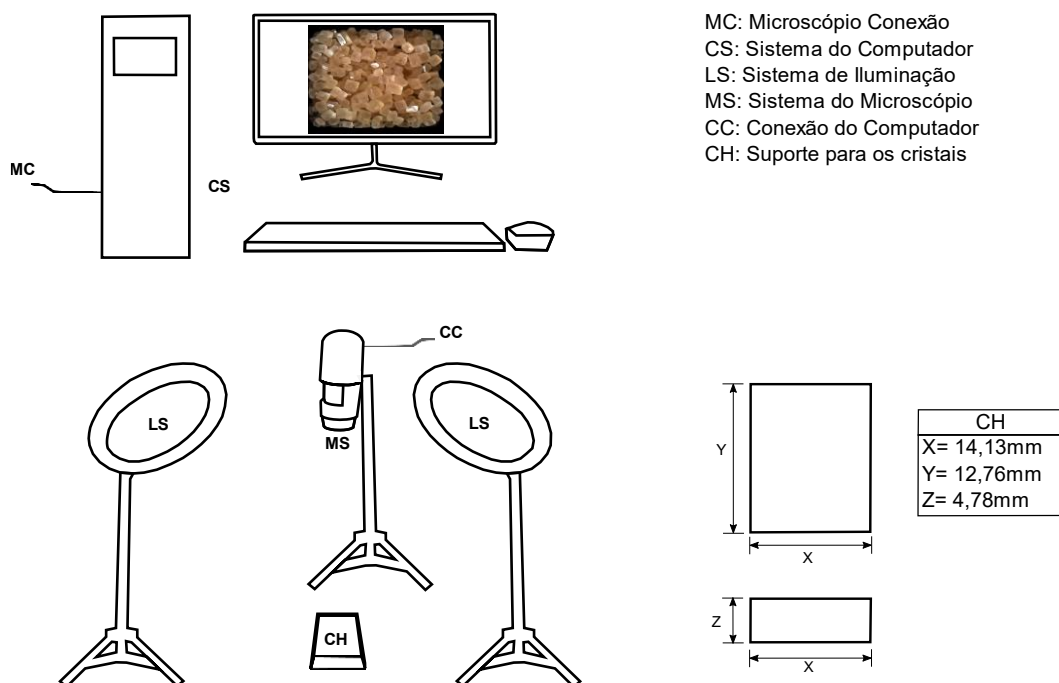


Figura 15: Modelo esquemático da configuração experimental para o sistema de aquisição de imagens (Fonte: Autora)

Vários são os trabalhos na literatura que processam imagens de materiais particulados (Boschetto and Giordano 2012; Frei and Kruis 2018b; Maiti et al. 2017; Cardona et al. 2018). Entretanto, dificilmente os trabalhos processam as imagens com as partículas aglomeradas, eles separam as partículas para posterior processamento das imagens. A partir dessas imagens, foi formado uma base de dados com 3530 imagens de açúcar VHP.

As imagens obtidas dos cristais aglomerados têm resolução 480 x 480 pixels e com o objetivo de desenvolver um sistema de aquisição de imagens de baixo custo que seja prático e de fácil manipulação, foi utilizado um microscópio digital USB (sensor CMOS e faixa de ampliação de 200 a 1600 vezes). As imagens foram processadas online usando um computador (CS, Intel i7-6500 CUP 3,20 GHz RAM 8,00 GB).

Para a obtenção das imagens utilizadas a fim de treinar a RNA e torna-la capaz de identificar os cristais de açúcar, um sistema de iluminação de LED experimental (LS), foi especialmente projetado e acoplado ao sistema de microscopia (MS) para controlar os efeitos do ambiente, posição para garantir que as imagens não sejam afetadas nem pela iluminação e nem pelas sombras (Figura15). Vários suportes com formatos e tamanhos diferentes foram testados previamente até a escolha do formato e tamanho CH. As amostras de cristais são

colocadas dentro do suporte (CH) para a obtenção das imagens uniformes de cristais aglomerados.

O sistema de aquisição de imagens de baixo custo é capaz de produzir imagens de alta qualidade e detectar também a variação de cor do açúcar VHP industrial. Por norma, a cor do açúcar precisa estar numa faixa entre 600 a 1.200 cor ICUMSA, pois é também um parâmetro de qualidade. Apesar deste sistema ser capaz de identificar a cor dos cristais, neste trabalho, foi avaliado somente o parâmetro de qualidade distribuição do tamanho de partículas PSD. A primeira etapa após a aquisição das imagens, é a calibração do microscópio e do sistema de iluminação, aquele ajustado no foco e esse na altura para formar um ângulo de 45° controlando os efeitos de sombra dos cristais.

4. CONFIGURAÇÃO DA REDE NEURAL ARTIFICIAL

O modelo de Rede Neural Artificial (RNA) utilizado para identificar cristais de açúcar em imagens de cristais aglomerados foi selecionado dentre 20 modelos diferentes de RNA testados que estão descritos conforme as siglas CNN-x-D-y-w e DNN-D-y-w.

O TensorFlow é uma plataforma completa de código aberto para *machine learning* (ML). Ele tem um ecossistema abrangente e flexível de ferramentas, bibliotecas e recursos da comunidade que permite aos pesquisadores levar adiante ML de última geração e aos desenvolvedores criar e implantar aplicativos com tecnologia de ML. Os modelos testados utilizaram TensorFlow e nas RNAs, o (x) representa a sequência do número total de neurônios em cada uma das camadas de convoluções (três ou cinco camadas dependendo do modelo), (y) e (w) o número de neurônios nas camadas ocultas e os nomes CNN e DNN descrevem o tipo de RNA, rede *neural convolucional* e rede neural profunda com duas camadas ocultas, respectivamente. Uma breve descrição das RNAs avaliadas está descrita na Tabela 1.

Tabela 1: Redes neurais avaliadas 1

| RNAs | Neurônios (camadas da CNN) | Neurônios (camadas ocultas) |
|----------------------------------|----------------------------|-----------------------------|
| CNN-32-64-128-D-200-10 | 32, 64, 128 | 200, 10 |
| CNN-32-64-128-D-100-100 | 32, 64, 128 | 100, 100 |
| CNN-32-64-128-D-200-200 | 32, 64, 128 | 200, 200 |
| CNN3-32-64-128-D-50-50 | 32, 64, 128 | 50, 50 |
| CNN-32-64-128-D-20-20 | 32, 64, 128 | 20, 20 |
| CNN-64-128-256-D-200-10 | 64, 128, 256 | 200, 10 |
| CNN-64-128-256-D-100-100 | 64, 128, 256 | 100, 100 |
| CNN-64-128-256-D-200-200 | 64, 128, 256 | 200, 200 |
| CNN-64-128-256-D-50-50 | 64, 128, 256 | 50, 50 |
| CNN-64-128-256-D-20-20 | 64, 128, 256 | 20, 20 |
| CNN-32-64-128-256-512-D-200-10 | 32, 64, 128, 256, 512 | 200, 10 |
| CNN-32-64-128-256-512-D-100-100 | 32, 64, 128, 256, 512 | 100, 100 |
| CNN-32-64-128-256-512--D-200-200 | 32, 64, 128, 256, 512 | 200, 200 |
| CNN-32-64-128-256-512-D-50-50 | 32, 64, 128, 256, 512 | 50, 50 |
| CNN-32-64-128-256-512--D-20-20 | 32, 64, 128, 256, 512 | 20, 20 |
| DNN-D-200-10 | 0 (sem convolução) | 200, 10 |
| DNN-D-100-100 | 0 (sem convolução) | 100, 100 |
| DNN-D-200-200 | 0 (sem convolução) | 200, 200 |
| DNN-D-50-50 | 0 (sem convolução) | 50, 50 |
| DNN-D-20-20 | 0 (sem convolução) | 20, 20 |

A CNN seguiu a sequência de convolução das operações acompanhada pela normalização do lote, *polling* e duas camadas profundas com *dropout* de 0,2. Outros detalhes do TensorFlow e da ANN incluem o uso do otimizador Adam, entropia cruzada para treinamento da RNA (Equação 9), taxa de aprendizado de 0,001, kernels convolucionais de 3 * 3 com o pooling utilizando o máximo valor como referência, função de ativação ReLU em todas as camadas, exceto nas camadas de saída aonde foram utilizadas ativação do Softmax.

$$CE = - \sum_{i=1}^C t_i \log(s_i) = -t_1 \cdot \log(s_1) - (1 - t_1) \cdot \log(1 - s_1) \quad (9)$$

Onde t_i e s_i representam as informações fornecidas por observação direta e a pontuação CNN para cada classe i .

Conforme dito anteriormente, 20 configurações diferentes de redes neurais foram avaliadas para uma amostra em quintuplicata (Tabela 1). Logo após, 3 configurações que

obtiveram o melhor desempenho foram escolhidas para serem replicadas para outras três amostras. Para a avaliação do desempenho da rede, a acurácia (A) do teste (Equação 1) foi calculada em todas as simulações e ao final de cada configuração, a média simples de cada quintuplicata foi calculada e considerada como sendo a métrica que avalia o desempenho.

As três melhores configurações de redes neurais foram utilizadas na sequência dos experimentos na qual foi ampliado o número de amostras para três e o algoritmo foi rodado novamente 20 vezes para cada uma das três configurações e para cada uma dessas três amostras. Nesta fase foi considerado como métrica o valor do R^2 da distribuição acumulada expresso em termos da média simples e o desvio padrão (S) para cada configuração.

5. DETERMINAÇÃO DO PSD

O sistema proposto para calcular o PSD (distribuição de tamanho de partícula) é composto por quatro principais etapas conforme ilustrado na Figura 16. Primeiramente, foram selecionadas imagens de grãos obtidas através de uma segmentação manual e imagens de não-grãos obtidas através da simulação de um código que realiza a segmentação automática desenvolvido neste trabalho. Na segunda etapa, com as imagens de grãos e não-grãos, as três melhores configurações da rede neural artificial são treinadas. O teste da rede neural ocorre na terceira etapa onde as imagens dos cristais aglomerados são segmentadas e classificadas em grãos e não-grãos pela RNA até obter pelo menos 150 imagens de cristais de açúcar. Na quarta etapa, os PSD são obtidos a partir das imagens classificadas como grãos.

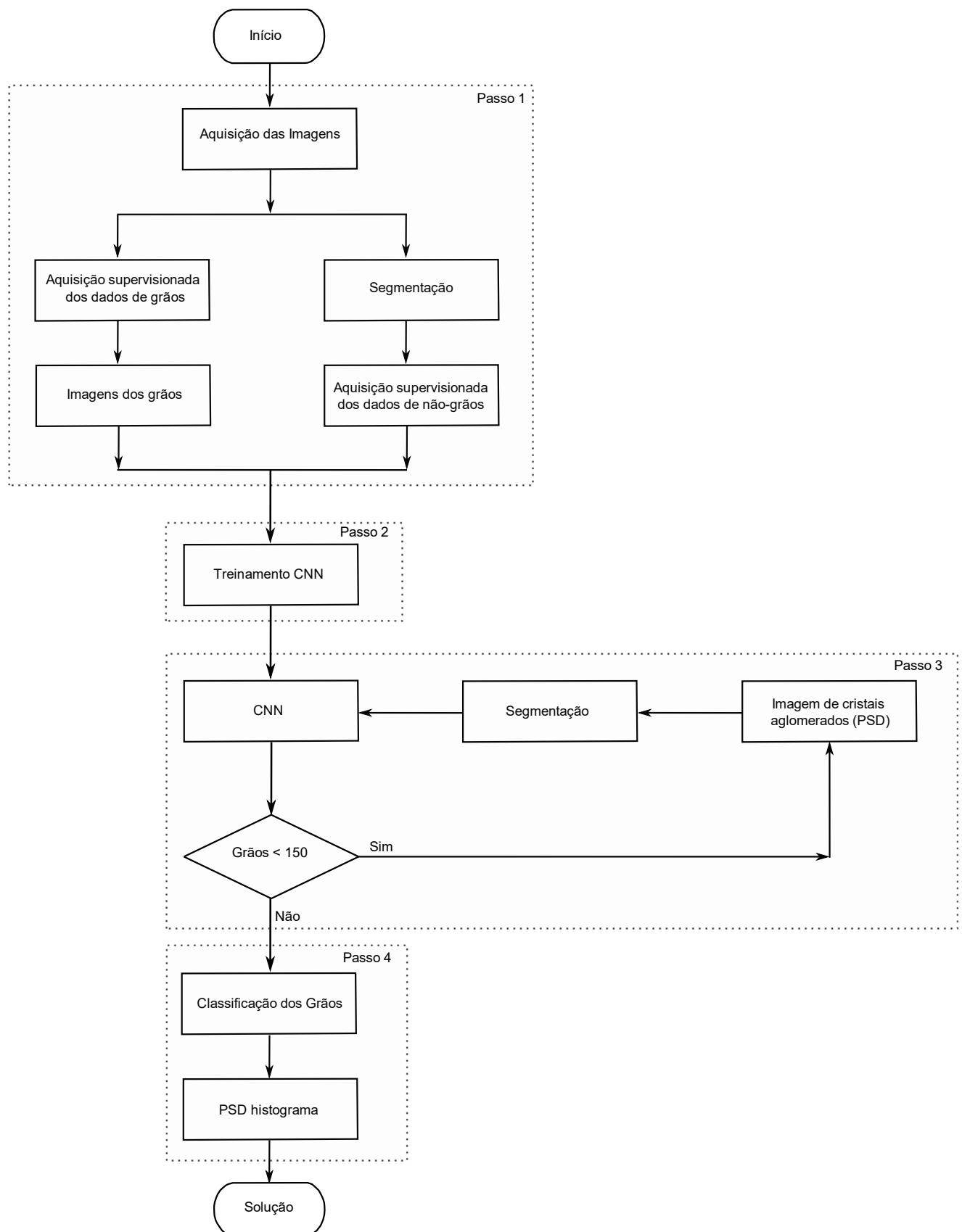


Figura 16: Fluxograma proposto para a determinação do PSD (Fonte: Autora)

Uma discussão mais detalhada da etapa de segmentação descrita na Figura 16 (passo 3) é apresentada na forma de algoritmo na Figura 17. A mesma descreve a forma como a segmentação para o processamento da imagem e o cálculo do PSD foram realizadas empregando-se seis etapas sequenciais. Da primeira à terceira etapa, a imagem dos cristais aglomerados é segmentada usando a posição e o tamanho aleatoriamente do quadrado do pixel com o tamanho dos quadrados menor que $1/12$ e $1/6$ do tamanho do suporte de cristal (CH). Na Figura 17 também pode ser observado a seleção dos grãos que ocorre automaticamente pelo algoritmo e gera as imagens de grãos segmentados (SGI) especificadas na quarta etapa para serem utilizadas no cálculo do PSD a partir da área das partículas.

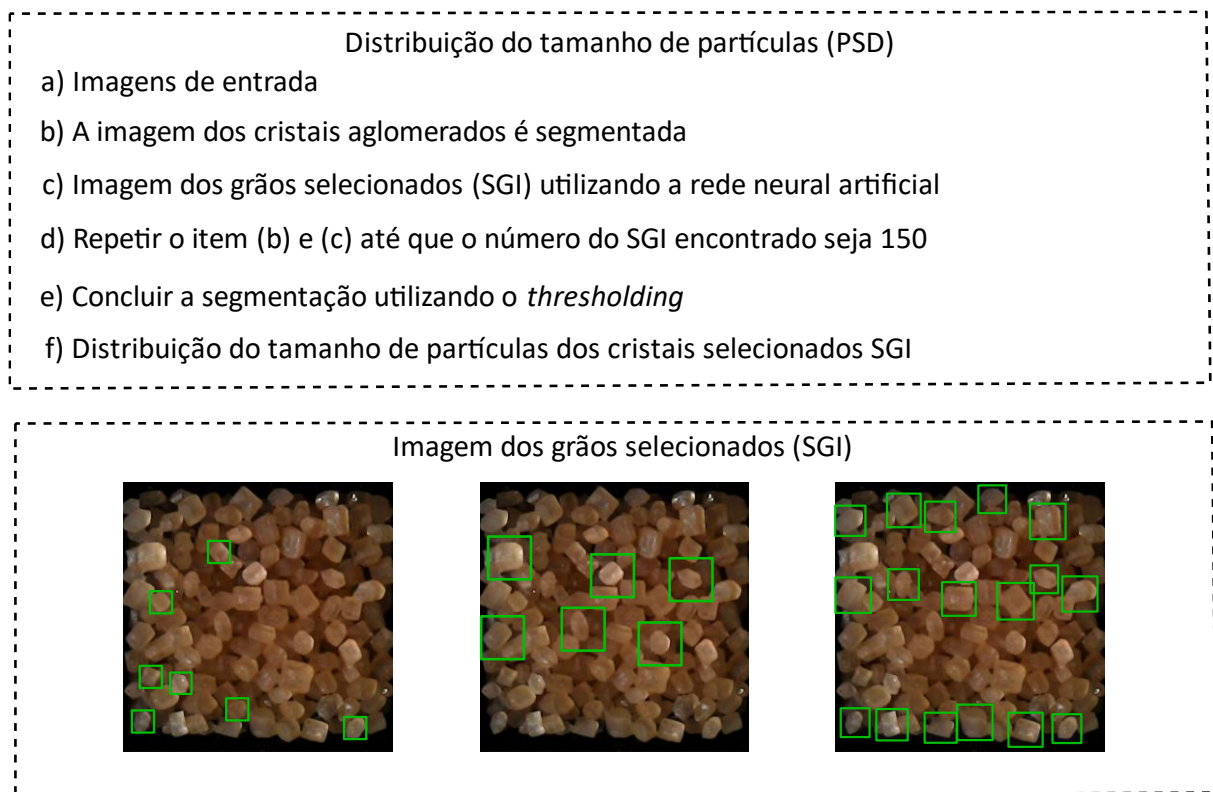


Figura 17: Fluxograma para o cálculo do PSD (Fonte: Autora)

CAPÍTULO 4 – RESULTADOS E DISCUSSÕES

CAPÍTULO 4 – RESULTADOS E DISCUSSÕES

As especificações de tamanho das partículas do açúcar afetam diretamente suas características e qualidade. Tradicionalmente, os fabricantes de açúcar utilizam a granulometria para avaliar o tamanho dos cristais devido ao seu baixo custo inicial e por ser aparentemente um procedimento simples.

Entretanto, a técnica de peneiramento apresenta desvantagens inerentes como, baixa reprodutibilidade, diferença de peneira para peneira, longo tempo de medição, manutenção e limpeza, baixo desempenho para partículas finas - normalmente para partículas $<44\ \mu\text{m}$ (325mesh), pouca possibilidade de automatização integrada ao controle automático do processo em tempo real, maior custo operacional e necessidade de instalações laboratoriais, perda de material (retido na peneira ou quebras com a vibração) e alguns materiais podem reagir quimicamente com a tela de separação da peneira.

Dessa forma, este capítulo apresenta os resultados obtidos com a metodologia de análise da distribuição do tamanho de partículas do açúcar através do processamento de imagens em tempo real apresentada no capítulo 3 e as discussões relativos ao desenvolvimento deste projeto.

1. BASE DE DADOS

Esta seção tem por objeto mostrar o banco de dados de imagens de cristais geradas durante o estudo as quais foram utilizadas para treinar as RNAs e a base de dados que pode ser obtida em condições de linha de produção industrial na qual a segmentação automática é realizada para testar as RNAs.

Na Figura 18 são mostradas trinta imagens que representam exemplos do conjunto de dados das imagens, conjunto esse formado por uma base de dados com 3530 imagens de açúcar VHP. Verifica-se que as imagens dos cristais aglomerados (Figuras 18a-18c) e imagens segmentadas manualmente geradas por um especialista (Figuras 18d-18f) são imagens de qualidade. Portanto, conforme mostrado, o dispositivo microscópico de baixo custo proposto é capaz de produzir imagens de alta qualidade e detectar a variação de cor do açúcar VHP industrial. O padrão de cor determinado por norma (ICUMSA *method* GS9/1/2/3-8) deve estar dentro da faixa 600 a 1200 cor ICUMSA e esse parâmetro é usado como um parâmetro de

qualidade industrial. As Figuras 18d-18f mostram o tipo característico de imagens esperadas durante a segmentação, a Figura 18d é um exemplo de uma imagem de grãos, buracos entre os grãos (Figura 18e) e fragmentos de cristais classificados como indefinidos (Figura 18f). As imagens do açúcar classificadas como buracos e indefinidos, foram agrupadas numa mesma classe chamada não-grãos para o treinamento da rede neural.

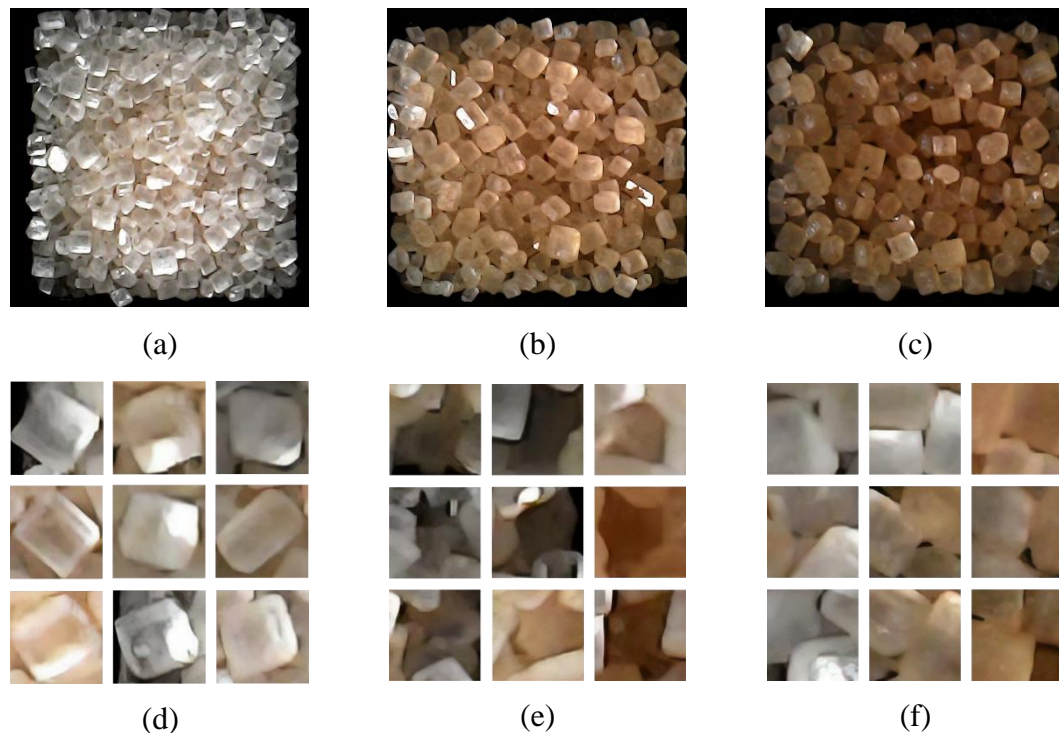


Figura 18: Base de dados das imagens utilizadas neste trabalho (Fonte: Autora)

O uso integrado de microscopia com microscópio portátil associado a um sistema de aquisição de dados como um computador de placa única que possui todos seus componentes no mesmo circuito impresso representa uma forma promissora de gerar resultados de análise da distribuição granulométrica em tempo real de materiais aglomerados. Na Tabela 2 é apresentado a discussão relativa ao investimento do sistema proposto em relação aos equipamentos e operação na análise da distribuição do tamanho de partículas manual realizada em um laboratório tendo como base o primeiro ano que contempla a aquisição dos equipamentos além do gasto com mão de obra durante todo ano. Nesta condição tem-se que o sistema proposto representa um investimento 47 vezes menor que o valor gasto no primeiro ano de funcionamento. Também pode-se obter valores semelhantes quando se efetua outros tipos de análise a exemplo da avaliação do gasto anual

considerando a depreciação total dos equipamentos após 10 anos de uso. Outro aspecto importante a ser ressaltado refere-se ao fato que a metodologia proposta pode ser utilizada na gerência de processos em planta (PIMS) e escala laboratorial (LIMS) ao passo que a abordagem com peneiras contempla somente LIMS.

Tabela 2: Comparação dos custos para a realização da análise granulométrica de forma manual e automática considerando como referência um ano de operação

| PSD | Jogo de peneiras | Agitador de peneiras eletromagnético | Custo unitário da mão de obra | 2 Luminárias LED | Microscópio USB portátil | Raspberry pi | Total ^a |
|------------|------------------|--------------------------------------|-------------------------------|------------------|--------------------------|--------------|--------------------|
| Manual | R\$ 1.848,00 | R\$ 9.916,84 | R\$ 4.267,45 | | | | R\$ 28.834,64 |
| Automática | | | | R\$ 110,00 | R\$ 76,91 | R\$ 419,00 | R\$ 605,91 |

Nota: a- o valor total refere-se ao investimento no primeiro ano que contempla a aquisição dos equipamentos acrescido do custo unitário da mão de obra para situação que um terço do dia do trabalhador é dedicado a esta análise.

2. SEGMENTAÇÃO AUTOMÁTICA

Na Figura 19 é mostrada a representação de 25 imagens segmentadas obtidas aleatoriamente pelo código desenvolvido que segmenta as imagens partindo de imagens de cristais aglomerados. A análise dessas imagens sugere que não-grãos representa o tipo de imagem que aparece com mais frequência quando se utiliza o algoritmo na geração de imagens segmentadas de grãos. A quantificação estatística desta segmentação com intervenção humana confirma este fato, pois na análise manual de 750 imagens segmentadas a partir das imagens de cristais aglomerados foram encontrados dezoito grãos e 732 imagens distintas de grãos. O código gerou 750 imagens segmentadas de 50 em 50 e o especialista contou quantos grãos foram gerados e identificou 18 grãos dentre as 750 imagens geradas. Com isso, pode-se concluir a geração de mais imagens de não-grãos que de grãos.

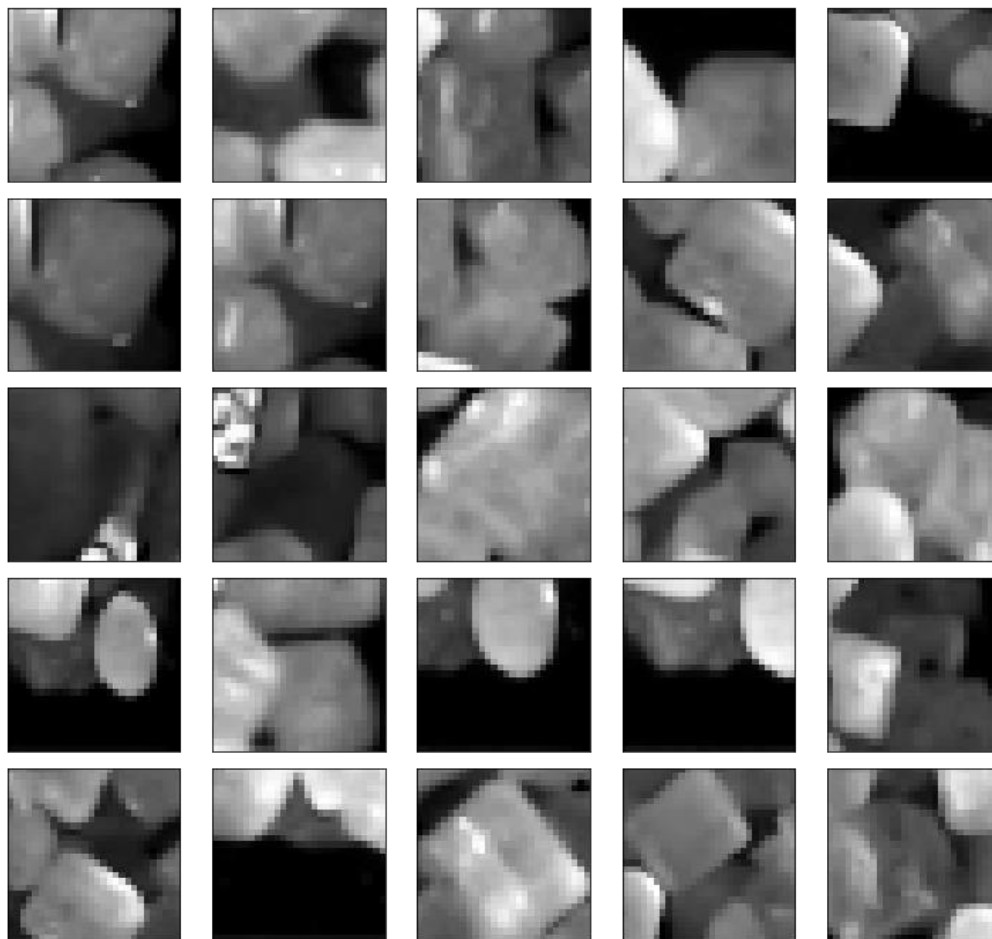


Figura 19: Imagens de grãos e não-grãos segmentadas automaticamente pelo código desenvolvido (Fonte: Autora)

Cabe à rede neural a rejeição das imagens segmentadas que correspondem a não-grãos e essa ação da rede neural retrata a etapa inicial que antecede o cálculo da distribuição do tamanho de partículas (PSD). A definição da rede neural envolve a definição do modelo e da arquitetura da rede que está ligada diretamente a dois aspectos principais: o tipo de problema a ser resolvido e o tipo de aprendizado mais apropriado para a solução deste problema. O próximo item descreve os resultados da avaliação das diferentes redes neurais artificiais (RNAs), assim como a avaliação comparativa destas RNAs nas escolhas do melhor sistema de classificação de imagens em grãos e não-grãos.

3. ANÁLISE DO DESEMPENHO DAS RNAS AVALIADAS

Em cada linha da Tabela 3 é apresentada o resultado de 5 rodagens do código para cada configuração de rede neural. Foram testadas 20 diferentes configurações de redes neurais a fim de classificar os cristais de açúcar. Nos resultados apresentados, pelo menos 50% das configurações apresentam acurácia (A) do teste maior que 0,8 validando a metodologia desenvolvida neste trabalho. Dessa forma, o sistema proposto é capaz de identificar e classificar bem os grãos e não-grãos a partir de imagens de cristais de açúcar aglomerados. Os três melhores resultados mostrados na Tabela 3 estão destacados em negrito e serão utilizados na próxima etapa de avaliar a melhor configuração de rede neural para o cálculo do PSD. A descrição completa com todos os resultados das RNAs para a classificação contendo todos os valores de todas as réplicas das configurações avaliadas encontra-se no Anexo I.

Pode-se observar que 50% das RNAs da Tabela 3 (10 RNAs das 20 testadas) são capazes de identificar e classificar grãos e não-grãos a partir do processamento de imagens de cristais aglomerados com acurácia superior 0,8, sendo que tanto as RNAs do tipo DNN quanto as de convolução (utilizando 3 e 5 camadas de convolução) apresentam acurácia (A) superior a 0,9 como mostra os resultados destacados em negrito e descrevem os melhores resultados de cada um dos tipos (DNN e convolução com três e cinco camadas). A primeira configuração é a rede neural de convolução (CNN) com três camadas de convolução cujas camadas apresentam, 32, 64 e 128 neurônios e duas camadas de neurônios ocultas com 200 e 10 neurônios, respectivamente. Sua sigla é representada por CNN-32-64-128-D-200-10 e obteve uma acurácia de 0,91.

A outra configuração que obteve o melhor valor de acurácia do teste foi também a rede de convolução (CNN) com três camadas de convolução com 64, 128 e 256 neurônios e 2 camadas de neurônios ocultas com 20 e 20 neurônios que apresentou uma acurácia de 0,947. Por fim, a rede neural que obteve melhor desempenho avaliado através da métrica da acurácia (A) do teste foi a rede neural *Deep Learning* (DNN) que apresenta duas camadas de neurônios ocultas com 200 e 10 neurônios e tem a sigla DNN-D-200-10. Essa rede apresentou uma acurácia de 0,969.

Tabela 3: Média dos resultados das redes neurais avaliadas

| Ensaio | RNAs | N1 | N2 | (A) treino | (A) teste |
|-----------|----------------------------------|------------|-----------|---------------|--------------|
| 1 | CNN-32-64-128-D-200-10 | 200 | 10 | 1 | 0,91 |
| 2 | CNN-32-64-128-D-100-100 | 100 | 100 | 1 | 0,78 |
| 3 | CNN-32-64-128-D-200-200 | 200 | 200 | 1 | 0,882 |
| 4 | CNN3-32-64-128-D-50-50 | 50 | 50 | 0,994 | 0,865 |
| 5 | CNN-32-64-128-D-20-20 | 20 | 20 | 0,997 | 0,729 |
| 6 | CNN-64-128-256-D-200-10 | 200 | 10 | 1 | 0,803 |
| 7 | CNN-64-128-256-D-100-100 | 100 | 100 | 1 | 0,724 |
| 8 | CNN-64-128-256-D-200-200 | 200 | 200 | 1 | 0,698 |
| 9 | CNN-64-128-256-D-50-50 | 50 | 50 | 1 | 0,71 |
| 10 | CNN-64-128-256-D-20-20 | 20 | 20 | 1 | 0,947 |
| 11 | CNN-32-64-128-256-512-D-200-10 | 200 | 10 | 0,603 | 0,58 |
| 12 | CNN-32-64-128-256-512-D-100-100 | 100 | 100 | 0,999 | 0,736 |
| 13 | CNN-32-64-128-256-512--D-200-200 | 200 | 200 | 0,999 | 0,691 |
| 14 | CNN-32-64-128-256-512-D-50-50 | 50 | 50 | 0,998 | 0,595 |
| 15 | CNN-32-64-128-256-512--D-20-20 | 20 | 20 | 0,998 | 0,512 |
| 16 | DNN-D-200-10 | 200 | 10 | 0,972 | 0,969 |
| 17 | DNN-D-100-100 | 100 | 100 | 0,99 | 0,959 |
| 18 | DNN-D-200-200 | 200 | 200 | 0,997 | 0,952 |
| 19 | DNN-D-50-50 | 50 | 50 | 0,974 | 0,961 |
| 20 | DNN-D-20-20 | 20 | 20 | 0,948 | 0,963 |

Essas três configurações de redes neurais foram testadas para outras 3 amostras a fim de ampliar os testes de avaliação e validação da metodologia apresentada.

4. CAPACIDADE DE OBTENÇÃO DO PSD ASSOCIADO AS RNAS SELECIONADAS

As três configurações de rede neural que apresentaram os melhores resultados para as métricas avaliadas foram testadas para a avaliação do código cujo objetivo é obter o PSD. Os resultados de 20 execuções do código para cada configuração e três amostras distintas, estão apresentados na Tabela 4 com a média das métricas avaliadas das réplicas, na qual pode-se observar que os valores do R^2 estão acima de 0,83 e o desvio padrão inferior a 0,1 indicando que os métodos são satisfatórios com um grau de dispersão do conjunto de dados próximos de zero, ou seja, os resultados estão homogêneos. Detalhes de todas as réplicas podem ser conferidos no Anexo II.

De acordo com os resultados apresentados na Tabela 4, a configuração da rede neural que obteve melhor desempenho para as amostras avaliadas foi a rede de convolução (CNN) com três camadas de convolução 64, 128 e 256 neurônios em cada respectivamente e com duas camadas ocultas com vinte neurônios em cada (CNN-64-128-256-D-20-20) e assim foi selecionada como a melhor rede a ser utilizada no software de obtenção de PSD.

Tabela 4: Métricas avaliadas das três redes neurais com melhor desempenho para três amostras

| | AM-3 | | AM-7 | | AM-8 | | R ² (Média) |
|-------------------------------|----------------|--------------|----------------|--------------|----------------|--------------|---------------------------|
| | R ² | SD | R ² | SD | R ² | SD | |
| CNN-32-64-128-D-200-10 | 0,959 | 0,016 | 0,850 | 0,037 | 0,831 | 0,056 | 0,880 |
| CNN-64-128-256-D-20-20 | 0,957 | 0,059 | 0,892 | 0,058 | 0,920 | 0,015 | 0,923 |
| DNN-D-200-10 | 0,956 | 0,050 | 0,838 | 0,078 | 0,910 | 0,054 | 0,901 |

5. ANÁLISE DO DESEMPENHO FINAL DO SISTEMA DE MICROSCOPIO COM PDS PROPOSTO

Na Tabela 5 é mostrado os resultados da matriz confusão da amostra 3, na Tabela 6 é mostrado os resultados da matriz confusão da amostra 7 e na Tabela 7 é mostrado os resultados da matriz confusão da amostra 8 considerando a RNA com arquitetura CNN-64-128-256-D-20-20. Pode-se observar que o código apresentado neste trabalho consegue a partir de uma foto de cristais aglomerados, segmentar e classificar como grãos e não-grãos, além de calcular a distribuição do tamanho de partículas (PSD) das amostras de açúcar.

Tabela 5: Matriz confusão para a amostra 3

| Amostra 3 | VP | VN | FP | FN | A (%) | P (%) | R (%) | S (%) | F1 (%) |
|-----------|-----|-----|----|----|-------|-------|-------|-------|--------|
| Treino | 173 | 170 | 0 | 0 | 100 | 100 | 100 | 100 | 100 |
| Validação | 71 | 71 | 4 | 1 | 96,6 | 94,7 | 98,6 | 94,7 | 96,6 |

Tabela 6: Matriz confusão para a amostra 7

| Amostra 7 | VP | VN | FP | FN | A (%) | P (%) | R (%) | S (%) | F1 (%) |
|-----------|-----|-----|----|----|-------|-------|-------|-------|--------|
| Treino | 175 | 168 | 0 | 0 | 100 | 100 | 100 | 100 | 100 |
| Validação | 70 | 73 | 2 | 2 | 97,3 | 97,2 | 97,2 | 97,3 | 97,2 |

Tabela 7: Matriz confusão para a amostra 8

| Amostra 8 | VP | VN | FP | FN | A (%) | P (%) | R (%) | S (%) | F1 (%) |
|-----------|-----|-----|----|----|-------|-------|-------|-------|--------|
| Treino | 175 | 168 | 0 | 0 | 100 | 100 | 100 | 100 | 100 |
| Validação | 71 | 72 | 3 | 1 | 97,3 | 96 | 98,6 | 96 | 97,3 |

Ao todo, 490 imagens foram utilizadas para treinar e testar a CNN-64-128-256-D-20-20 para cada uma das três amostras. Para a amostra 3, a rede classificou 1 grão e 4 não-grãos erroneamente, obtendo uma acurácia de 96,6%. Avaliando a amostra 7, 2 grãos e 2 não-grãos foram avaliados de forma errada e a acurácia teve um resultado de 97,3% e para a amostra 8, a rede errou 1 na classificação de 1 grão e 3 não-grãos, obtendo uma acurácia de 97,3%. Os resultados são bastante realísticos quando comparados com outros autores. Por exemplo, Frei e Kruis (2018) analisaram o tamanho das partículas aglomeradas em microscopia eletrônica de transmissão e encontraram 95,9% de acurácia. Outro exemplo foi o trabalho de Octay e Gurses (2019) que utilizaram microscopia de alta resolução para detectar e segmentar nanopartículas e encontraram respectivamente, 98,2% e 96,6% de acurácia.

Como o principal objetivo do presente trabalho é avaliar a distribuição do tamanho de partículas (PSD), o código calculou a área dos grãos que foram classificados pela rede neural e por fim os diâmetros. Nas Figuras 20, 21 e 22 são mostradas a distribuição acumulada do tamanho das partículas das amostras 3, 7 e 8 respectivamente com valores do PSD acumulado.

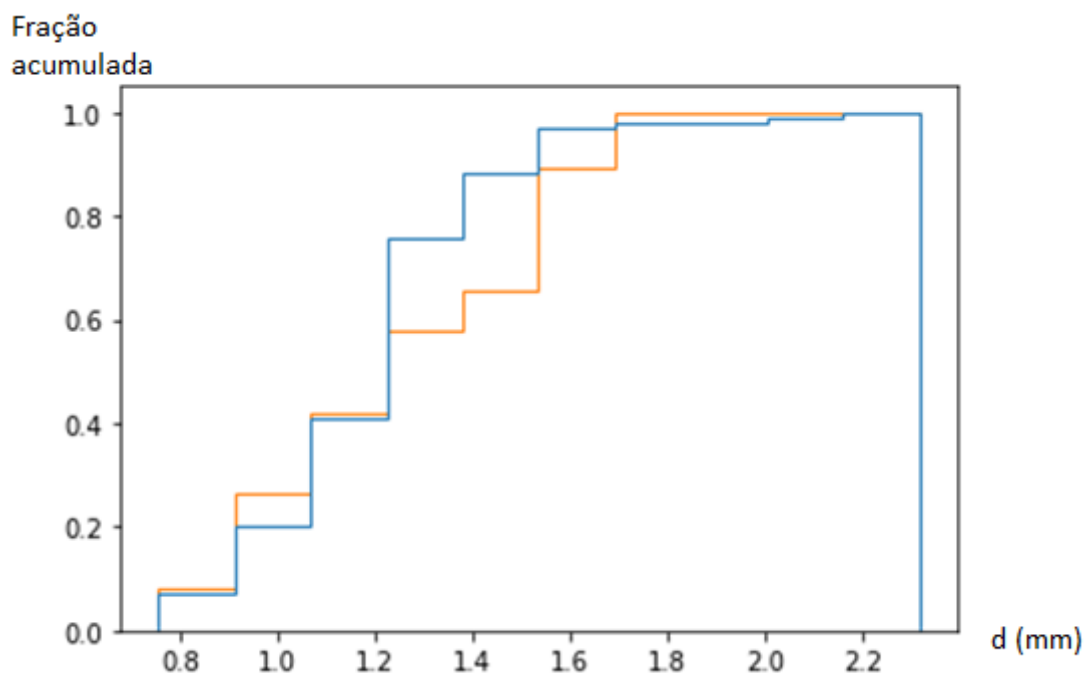


Figura 20: Distribuição acumulada da distribuição do tamanho de partículas da amostra 3. Curva cor laranja dados CNN e curva azul dados Image J.

Para que a produção do açúcar atenda a norma de qualidade, a distribuição do tamanho de partículas (PSD) deve ser calculada e as peneiras utilizadas na análise são: 0,8mm, 1,0mm, 1,2mm, 1,4mm, 1,6mm e 1,8mm. De acordo com a Figura 20, verifica-se que entre os diâmetros 1,3 e 1,7mm há uma diferença de aproximadamente 20% entre os dados da CNN e os experimentais do ImageJ. Embora exista essa diferença, a acurácia é 96,6% (Tabela 5).

A Figura 21 mostra os resultados da distribuição acumulada da amostra 7. Pode-se observar que apesar de existir uma diferença entre os dados experimentais e os calculados através da rede neural, a acurácia também 97,3%.

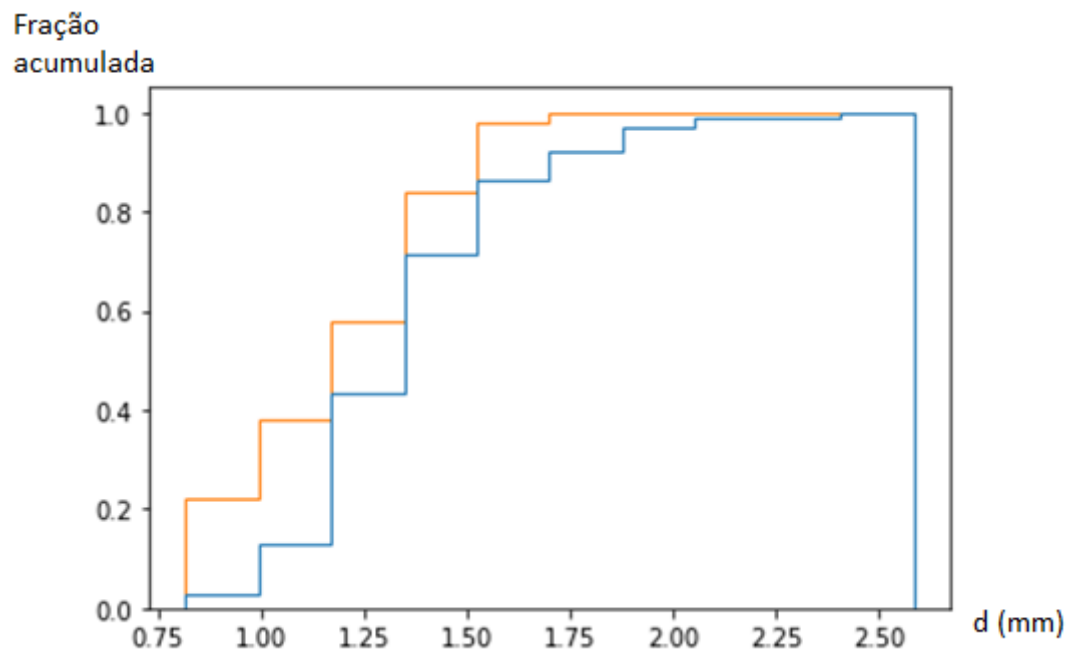


Figura 21: Distribuição acumulada da distribuição do tamanho de partículas da amostra 7.
Curva cor laranja dados CNN e curva azul dados Image J.

Os resultados da amostra 8 estão apresentados na Figura 22 e também trazem alguns pontos com diferença entre os dados experimentais obtidos através do ImageJ e os dados obtidos através da rede neural. Entretanto, a acurácia foi de 97,3%.

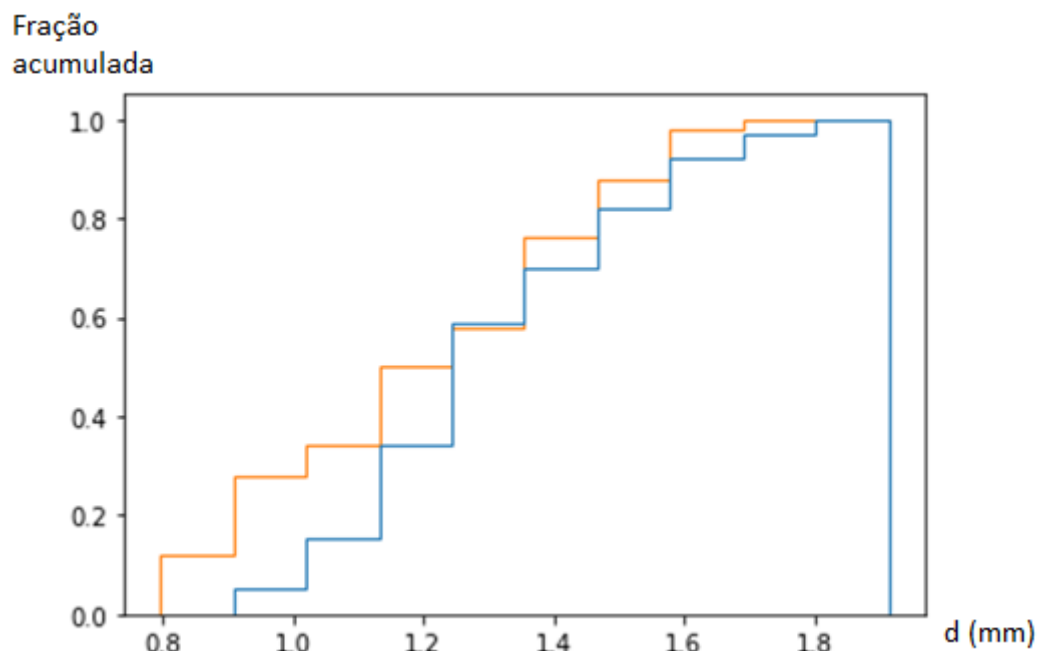


Figura 22: Distribuição acumulada da distribuição do tamanho de partículas da amostra 8. Curva cor laranja dados CNN e curva azul dados Image J.

6. AVALIAÇÃO COMPARATIVA DO PSD ENTRE O SISTEMA EM TEMPO REAL PROPOSTO E O PROCESSAMENTO DE IMAGEM POR SOFTWARES EM LABORATORIO

Na Figura 23 são apresentados os resultados da avaliação da distribuição granulométrica associada ao sistema de aquisição e processamento de imagens com RNAs para avaliação de diferentes amostras (Fig 23 a-c) e uma mesma amostra com obtenção dos resultados em réplica (Fig. 23 d). Pode-se observar que o software proposto e tem a distribuição granulométrica satisfatória em termos de fração mássica acumulada é capaz de representar as diferentes classes de diâmetros sendo que os mesmos ainda necessitam de ajustes finais para que se possa quantificar todas classes com a mesmo nível de acerto.

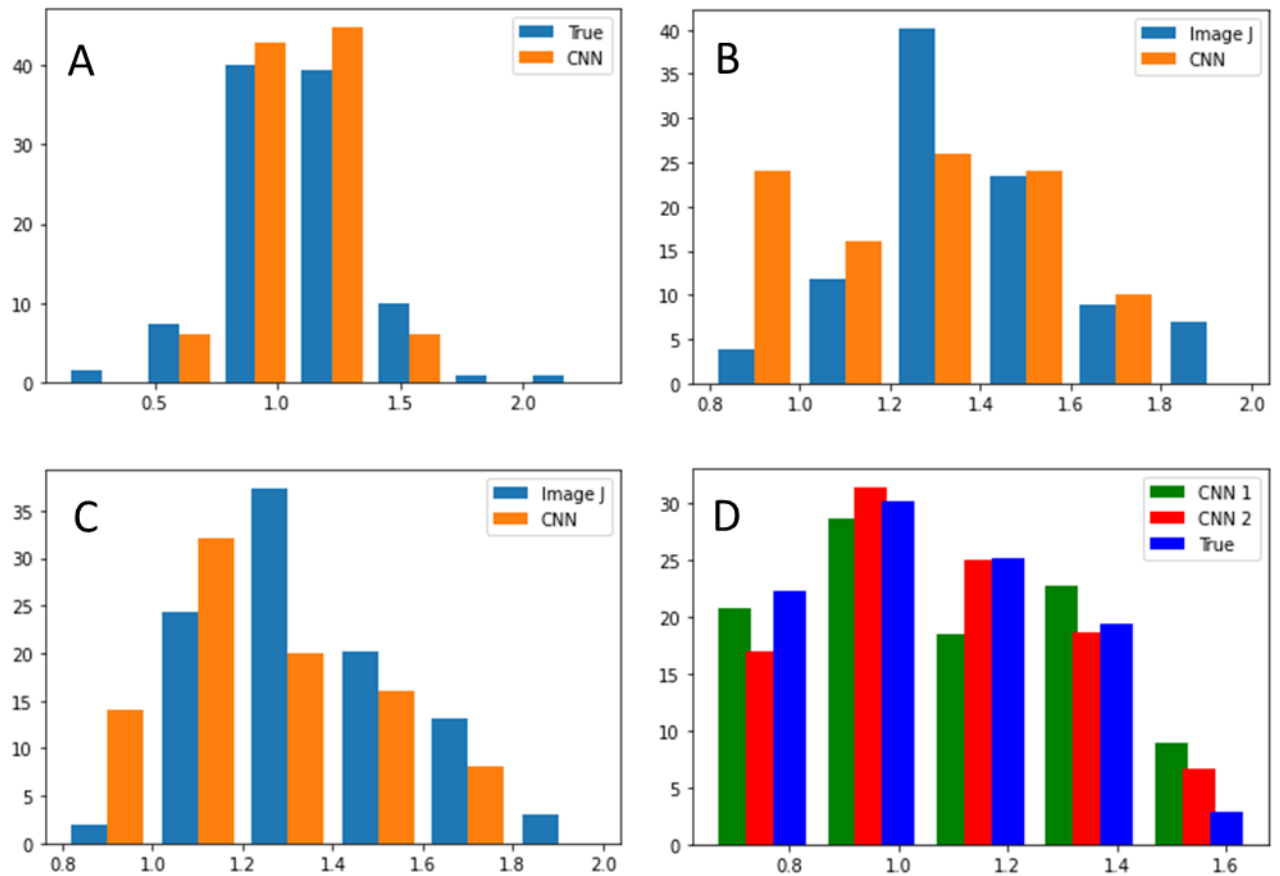


Figura 23: Histogramas do PSD: A) amostra 3. $R^2=0,93$, B) amostra 7. $R^2=0,874$ C) amostra 8. $R^2=0,856$ e D) amostra 3. $R^2=0,92$

Pode-se observar que existem pouca diferença entre os valores experimentais do ImageJ e os calculados através do código desenvolvido no presente trabalho, mas essas diferenças não influenciaram na validação da metodologia empregada, pois os valores das métricas avaliadas R^2 e acurácia foram acima de 90%. Na Figura23-D, a rede CNN-1 utilizou a média dos valores das dez primeiras simulações e a CNN-2 utilizou a média dos valores das próximas dez simulações. Observa-se bons resultados comparados com os valores experimentais do ImageJ apresentados pela cor azul.

CAPÍTULO 5 – CONCLUSÕES

O controle de qualidade de cristais de açúcar pode ser realizado por imagem em tempo real utilizando um dispositivo de baixo custo com obtenção das imagens de aglomerados de cristais por um computador seguido do processamento das mesmas com o uso de aprendizado de máquina para identificar os cristais nas imagens de aglomerados de cristais.

Referente a etapa de obtenção das imagens tem-se que um microscópio portátil com saída USB de baixo custo é adequado para gerar imagens com resolução e detalhes suficientes para que as amostras possam ser avaliadas na forma de imagens e as dimensões dos cristais possam ser extraídos destas imagens de forma satisfatória.

Por outro lado, em relação ao uso de aprendizagem de máquina no processamento das imagens foi constatado que redes neurais artificiais do tipo *feedforward* e do tipo de convolução se mostraram capazes de identificar os cristais com acurácia superior a 90%, sendo que nos melhores resultados obtidos, redes de convolução, foram capazes de identificar os cristais nas imagens com acurácia de 96,9%.

Como conclusão final pode-se dizer que o uso integrado de microscopia de baixo custo seguido da identificação dos cristais em imagens segmentadas e posterior avaliação das dimensões de cristais representa uma forma promissora de gerar um sistema em tempo real de análise distribuição granulométrica de materiais aglomerados.

REFERÊNCIAS BIBLIOGRÁFICAS

- Albawi, Saad, Tareq Abed Mohammed Mohammed, and Saad Alzawi. 2017. "Layers of a Convolutional Neural Network." *Ieee*, 16.
- Bai, Feiyan, Minqiang Fan, Hongli Yang, and Lianping Dong. 2021. "Image Segmentation Method for Coal Particle Size Distribution Analysis." *Particuology* 56: 163–70. <https://doi.org/10.1016/j.partic.2020.10.002>.
- Boschetto, Alberto, and Veronica Giordano. 2012. "Powder Sampling and Characterization by Digital Image Analysis." *Measurement: Journal of the International Measurement Confederation* 45 (5): 1023–38. <https://doi.org/10.1016/j.measurement.2012.01.041>.
- Cardona, Javier, Carla Ferreira, John McGinty, Andrew Hamilton, Okpeafoh S. Agimelen, Alison Cleary, Robert Atkinson, et al. 2018. "Image Analysis Framework with Focus Evaluation for in Situ Characterisation of Particle Size and Shape Attributes." *Chemical Engineering Science* 191: 208–31. <https://doi.org/10.1016/j.ces.2018.06.067>.
- Chen, Jiahui, Christine Nowakowski, Dan Green, and Richard W Hartel. 2015. "State Behavior and Crystal Growth Kinetics of Sucrose and Corn Syrup Mixtures." *Journal of Food Engineering* 161: 1–7. <https://doi.org/https://doi.org/10.1016/j.jfoodeng.2015.03.032>.
- Chen, Shumian, Juntao Xiong, Wentao Guo, Rongbin Bu, Zhenhui Zheng, Yunqi Chen, Zhengang Yang, and Rui Lin. 2019. "Colored Rice Quality Inspection System Using Machine Vision." *Journal of Cereal Science* 88 (May): 87–95. <https://doi.org/10.1016/j.jcs.2019.05.010>.
- Crestani, C. E., A. Bernardo, C. B.B. Costa, and M. Giulietti. 2021. "An Artificial Neural Network Model Applied to Convert Sucrose Chord Length Distributions into Particle Size Distributions." *Powder Technology* 384: 186–94. <https://doi.org/10.1016/j.powtec.2021.01.075>.
- Drugan, M M, and E.-G. Talbi. 2014. *Adaptive Multi-Operator Metaheuristics for Quadratic Assignment Problems. Advances in Intelligent Systems and Computing*. Vol. 288. https://doi.org/10.1007/978-3-319-07494-8_11.
- Fang, Yida, Anderson Ellis, Minori Uchimiya, and Timothy J. Strathmann. 2019. "Selective Oxidation of Colour-Inducing Constituents in Raw Sugar Cane Juice with Potassium Permanganate." *Food Chemistry* 298 (February): 125036. <https://doi.org/10.1016/j.foodchem.2019.125036>.
- Frei, Max, and Frank Einar Kruis. 2018a. "Fully Automated Primary Particle Size Analysis of Agglomerates on Transmission Electron Microscopy Images via Artificial Neural Networks." *Powder Technology* 332: 120–30. <https://doi.org/10.1016/j.powtec.2018.03.032>.
- . 2018b. "Fully Automated Primary Particle Size Analysis of Agglomerates on Transmission Electron Microscopy Images via Artificial Neural Networks." *Powder Technology* 332: 120–30. <https://doi.org/10.1016/j.powtec.2018.03.032>.
- Igathinathane, C., L. O. Pordesimo, E. P. Columbus, W. D. Batchelor, and S. Sokhansanj. 2009. "Sievelless Particle Size Distribution Analysis of Particulate Materials through Computer Vision." *Computers and Electronics in Agriculture* 66 (2): 147–58. <https://doi.org/10.1016/j.compag.2009.01.005>.

- Jain, Anil K., Jianchang Mao, and K. M. Mohiuddin. 1996. "Artificial Neural Networks: A Tutorial." *Computer* 29 (3): 31–44. <https://doi.org/10.1109/2.485891>.
- Maiti, Abhik, Debashish Chakravarty, Kousik Biswas, and Arpan Halder. 2017. "Development of a Mass Model in Estimating Weight-Wise Particle Size Distribution Using Digital Image Processing." *International Journal of Mining Science and Technology* 27 (3): 435–43. <https://doi.org/10.1016/j.ijmst.2017.03.015>.
- Oh, Sung-Mook, Jin Park, Jinsun Yang, Young-Gyun Oh, and Kyung-Woo Yi. 2021. "Image Processing for Analysis of Carbon Black Pellet Size Distribution during Pelletizing: Carbon Black PSD (PSD: Pellet Size Distribution) by Image Processing." *Measurement* 174: 108963. <https://doi.org/https://doi.org/10.1016/j.measurement.2021.108963>.
- Oktaý, Ayse Betul, and Anil Gurses. 2019. "Automatic Detection, Localization and Segmentation of Nano-Particles with Deep Learning in Microscopy Images." *Micron* 120 (October 2018): 113–19. <https://doi.org/10.1016/j.micron.2019.02.009>.
- Shendryk, Yuri, Robert Davy, and Peter Thorburn. 2021. "Integrating Satellite Imagery and Environmental Data to Predict Field-Level Cane and Sugar Yields in Australia Using Machine Learning." *Field Crops Research* 260 (August 2020): 107984. <https://doi.org/10.1016/j.fcr.2020.107984>.
- Simon, Phil. Wiley and. 2013. *Too Big to Ignore : The Business Case for Big Data*. Boston: Wiley. <https://doi.org/DOI:10.1002/9781119204039>.
- Sudarsan, Bharath, Wenjun Ji, Viacheslav Adamchuk, and Asim Biswas. 2018. "Characterizing Soil Particle Sizes Using Wavelet Analysis of Microscope Images." *Computers and Electronics in Agriculture* 148 (August 2017): 217–25. <https://doi.org/10.1016/j.compag.2018.03.019>.
- Veronezi, Carlos Cassiano Denipotti, Priscyla Waleska Targino De Azevedo Simões, Robson Luiz Dos Santos, Edroaldo Lummertz Da Rocha, Suelen Melão, Merisandra Côrtes De Mattos, and Cristian Cechinel. 2011. "Análise Computacional Para Auxílio Ao Diagnósticos de Osteoartrite de Coluna Lombar Baseado Em Redes Neurais Artificial." *Revista Brasileira de Ortopedia* 46 (2): 195–99. <https://doi.org/10.1590/S0102-36162011000200014>.
- Xie, Wei-Qiang, Xiao-Ping Zhang, Xin-Mei Yang, Quan-Sheng Liu, Shao-Hui Tang, and Xin-Bin Tu. 2020. "3D Size and Shape Characterization of Natural Sand Particles Using 2D Image Analysis." *Engineering Geology* 279: 105915. <https://doi.org/https://doi.org/10.1016/j.enggeo.2020.105915>.
- Yang, Xiaohan, Ting Ren, and Lihai Tan. 2020. "Size Distribution Measurement of Coal Fragments Using Digital Imaging Processing." *Measurement: Journal of the International Measurement Confederation* 160: 107867. <https://doi.org/10.1016/j.measurement.2020.107867>.

ANEXO I

A Tabela 3 completa com os resultados de todas as simulações para a avaliação da melhor configuração da rede neural está apresentada abaixo na Tabela 8.

Tabela 8: (Tabela 3 completa) Média dos resultados das redes neurais avaliadas

| N1 | N2 | R ² | (A) treino | (A) teste | Detalhes |
|-----|-----|----------------|------------|-----------|---------------------------------------|
| 200 | 10 | 0,98 | 1,00 | 0,88 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 10 | 0,95 | 1,00 | 0,77 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 10 | 0,87 | 1,00 | 0,98 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 10 | 0,96 | 1,00 | 0,95 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 10 | 0,97 | 1,00 | 0,97 | 3 camadas de convolução: 32, 64, 128 |
| 100 | 100 | 0,98 | 1,00 | 0,94 | 3 camadas de convolução: 32, 64, 128 |
| 100 | 100 | 0,96 | 1,00 | 0,51 | 3 camadas de convolução: 32, 64, 128 |
| 100 | 100 | 0,98 | 1,00 | 0,98 | 3 camadas de convolução: 32, 64, 128 |
| 100 | 100 | 0,97 | 1,00 | 0,96 | 3 camadas de convolução: 32, 64, 128 |
| 100 | 100 | 0,97 | 1,00 | 0,51 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 200 | 0,98 | 1,00 | 0,97 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 200 | 0,98 | 1,00 | 0,97 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 200 | 0,94 | 1,00 | 0,97 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 200 | 0,93 | 1,00 | 0,53 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 200 | 0,98 | 1,00 | 0,98 | 3 camadas de convolução: 32, 64, 128 |
| 50 | 50 | 0,88 | 1,00 | 0,96 | 3 camadas de convolução: 32, 64, 128 |
| 50 | 50 | 0,90 | 1,00 | 0,98 | 3 camadas de convolução: 32, 64, 128 |
| 50 | 50 | 0,98 | 1,00 | 0,93 | 3 camadas de convolução: 32, 64, 128 |
| 50 | 50 | 0,96 | 0,97 | 0,49 | 3 camadas de convolução: 32, 64, 128 |
| 50 | 50 | 0,93 | 1,00 | 0,97 | 3 camadas de convolução: 32, 64, 128 |
| 20 | 20 | 0,95 | 1,00 | 0,51 | 3 camadas de convolução: 32, 64, 128 |
| 20 | 20 | 0,87 | 0,98 | 0,51 | 3 camadas de convolução: 32, 64, 128 |
| 20 | 20 | 0,93 | 1,00 | 0,74 | 3 camadas de convolução: 32, 64, 128 |
| 20 | 20 | 0,97 | 1,00 | 0,95 | 3 camadas de convolução: 32, 64, 128 |
| 20 | 20 | 0,95 | 1,00 | 0,93 | 3 camadas de convolução: 32, 64, 128 |
| 200 | 10 | 0,91 | 1,00 | 0,82 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 10 | 0,93 | 1,00 | 0,56 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 10 | 0,99 | 1,00 | 0,84 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 10 | 0,95 | 1,00 | 0,84 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 10 | 0,93 | 1,00 | 0,95 | 3 camadas de convolução: 64, 128, 256 |
| 100 | 100 | 0,79 | 1,00 | 0,95 | 3 camadas de convolução: 64, 128, 256 |
| 100 | 100 | 0,94 | 1,00 | 0,51 | 3 camadas de convolução: 64, 128, 256 |
| 100 | 100 | 0,94 | 1,00 | 0,69 | 3 camadas de convolução: 64, 128, 256 |
| 100 | 100 | 0,94 | 1,00 | 0,96 | 3 camadas de convolução: 64, 128, 256 |

| | | | | | |
|-----|-----|------|------|------|--|
| 100 | 100 | 0,90 | 1,00 | 0,51 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 200 | 0,96 | 1,00 | 0,97 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 200 | 0,97 | 1,00 | 0,51 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 200 | 0,98 | 1,00 | 0,97 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 200 | 0,96 | 1,00 | 0,53 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 200 | 0,89 | 1,00 | 0,51 | 3 camadas de convolução: 64, 128, 256 |
| 50 | 50 | 0,92 | 1,00 | 0,51 | 3 camadas de convolução: 64, 128, 256 |
| 50 | 50 | 0,92 | 1,00 | 0,97 | 3 camadas de convolução: 64, 128, 256 |
| 50 | 50 | 0,98 | 1,00 | 0,55 | 3 camadas de convolução: 64, 128, 256 |
| 50 | 50 | 0,97 | 1,00 | 0,56 | 3 camadas de convolução: 64, 128, 256 |
| 50 | 50 | 0,88 | 1,00 | 0,97 | 3 camadas de convolução: 64, 128, 256 |
| 20 | 20 | 0,97 | 1,00 | 0,99 | 3 camadas de convolução: 64, 128, 256 |
| 20 | 20 | 0,96 | 1,00 | 0,86 | 3 camadas de convolução: 64, 128, 256 |
| 20 | 20 | 0,99 | 1,00 | 0,97 | 3 camadas de convolução: 64, 128, 256 |
| 20 | 20 | 0,98 | 1,00 | 0,97 | 3 camadas de convolução: 64, 128, 256 |
| 20 | 20 | 0,95 | 1,00 | 0,95 | 3 camadas de convolução: 64, 128, 256 |
| 200 | 10 | 0,97 | 0,50 | 0,49 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 10 | 0,97 | 1,00 | 0,94 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 10 | 0,98 | 0,50 | 0,49 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 10 | 0,96 | 0,50 | 0,49 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 10 | 0,99 | 0,50 | 0,49 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 100 | 100 | 0,93 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 100 | 100 | 0,96 | 1,00 | 0,73 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 100 | 100 | 0,96 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 100 | 100 | 0,99 | 1,00 | 0,99 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 100 | 100 | 0,99 | 1,00 | 0,95 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 200 | 0,94 | 1,00 | 0,95 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 200 | 0,95 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 200 | 0,96 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 200 | 0,96 | 1,00 | 0,97 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 200 | 0,93 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 50 | 50 | 0,95 | 1,00 | 0,49 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 50 | 50 | 0,98 | 0,99 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 50 | 50 | 0,90 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 50 | 50 | 0,95 | 1,00 | 0,95 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 50 | 50 | 0,69 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 20 | 20 | 0,95 | 0,99 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 20 | 20 | 0,98 | 1,00 | 0,52 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 20 | 20 | 0,98 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 20 | 20 | 0,87 | 0,99 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 20 | 20 | 0,97 | 1,00 | 0,51 | 5 camadas de convolução: 32, 64, 128, 256, 512 |
| 200 | 10 | 0,97 | 0,98 | 0,97 | DNN sem convolução |
| 200 | 10 | 0,98 | 0,98 | 0,97 | DNN sem convolução |

| | | | | | |
|-----|-----|------|------|------|--------------------|
| 200 | 10 | 0,99 | 0,96 | 0,97 | DNN sem convolução |
| 200 | 10 | 0,99 | 0,97 | 0,97 | DNN sem convolução |
| 200 | 10 | 0,94 | 0,97 | 0,97 | DNN sem convolução |
| 100 | 100 | 0,97 | 1,00 | 0,97 | DNN sem convolução |
| 100 | 100 | 0,95 | 0,99 | 0,95 | DNN sem convolução |
| 100 | 100 | 0,93 | 0,98 | 0,95 | DNN sem convolução |
| 100 | 100 | 0,94 | 0,99 | 0,95 | DNN sem convolução |
| 100 | 100 | 0,97 | 0,99 | 0,97 | DNN sem convolução |
| 200 | 200 | 0,95 | 0,99 | 0,97 | DNN sem convolução |
| 200 | 200 | 0,97 | 0,99 | 0,95 | DNN sem convolução |
| 200 | 200 | 0,97 | 1,00 | 0,95 | DNN sem convolução |
| 200 | 200 | 0,95 | 1,00 | 0,95 | DNN sem convolução |
| 200 | 200 | 0,97 | 1,00 | 0,95 | DNN sem convolução |
| 50 | 50 | 0,90 | 0,97 | 0,97 | DNN sem convolução |
| 50 | 50 | 0,96 | 0,97 | 0,97 | DNN sem convolução |
| 50 | 50 | 0,98 | 0,97 | 0,95 | DNN sem convolução |
| 50 | 50 | 0,94 | 0,97 | 0,95 | DNN sem convolução |
| 50 | 50 | 0,99 | 0,98 | 0,96 | DNN sem convolução |
| 20 | 20 | 0,99 | 0,96 | 0,96 | DNN sem convolução |
| 20 | 20 | 0,90 | 0,96 | 0,96 | DNN sem convolução |
| 20 | 20 | 0,99 | 0,94 | 0,97 | DNN sem convolução |
| 20 | 20 | 0,94 | 0,94 | 0,97 | DNN sem convolução |
| 20 | 20 | 0,86 | 0,95 | 0,97 | DNN sem convolução |

ANEXO II

Tabela 4 (completa) da Amostra 3, 7 e 8 seguem apresentadas abaixo nesta sequência.

| N1 | N2 | R^2 | acc train | acc test | Details | 0.8 | 1.0 | 1.2 | 1.4 | 1.6 | 1.8 |
|-----|----|-------|-----------|----------|---------------------------------------|--------|--------|--------|--------|--------|-------|
| 200 | 10 | 0,936 | 1,000 | 0,497 | 3 layers of Convolution: 32, 64, 128 | 11,429 | 22,857 | 28,571 | 20,000 | 17,143 | 0,000 |
| 200 | 10 | 0,939 | 1,000 | 0,980 | 3 layers of Convolution: 32, 64, 128 | 20,000 | 20,000 | 16,667 | 33,333 | 10,000 | 0,000 |
| 200 | 10 | 0,945 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 4,000 | 48,000 | 32,000 | 8,000 | 4,000 | 0,000 |
| 200 | 10 | 0,975 | 1,000 | 0,973 | 3 layers of Convolution: 32, 64, 128 | 17,073 | 24,390 | 24,390 | 19,512 | 14,634 | 0,000 |
| 200 | 10 | 0,958 | 1,000 | 0,735 | 3 layers of Convolution: 32, 64, 128 | 0,000 | 27,907 | 39,535 | 23,256 | 9,302 | 0,000 |
| 200 | 10 | 0,989 | 1,000 | 0,986 | 3 layers of Convolution: 32, 64, 128 | 10,345 | 24,138 | 37,931 | 17,241 | 10,345 | 0,000 |
| 200 | 10 | 0,964 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 18,519 | 33,333 | 22,222 | 18,519 | 7,407 | 0,000 |
| 200 | 10 | 0,910 | 1,000 | 0,973 | 3 layers of Convolution: 32, 64, 128 | 16,000 | 24,000 | 16,000 | 24,000 | 20,000 | 0,000 |
| 200 | 10 | 0,977 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 13,793 | 17,241 | 37,931 | 27,586 | 3,448 | 0,000 |
| 200 | 10 | 0,907 | 1,000 | 0,973 | 3 layers of Convolution: 32, 64, 128 | 17,857 | 39,286 | 21,429 | 10,714 | 7,143 | 0,000 |
| 200 | 10 | 0,956 | 1,000 | 0,973 | 3 layers of Convolution: 32, 64, 128 | 16,667 | 16,667 | 26,667 | 36,667 | 3,333 | 0,000 |
| 200 | 10 | 0,872 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 28,000 | 40,000 | 16,000 | 16,000 | 0,000 | 0,000 |
| 200 | 10 | 0,942 | 1,000 | 0,952 | 3 layers of Convolution: 32, 64, 128 | 28,000 | 24,000 | 28,000 | 12,000 | 4,000 | 0,000 |
| 200 | 10 | 0,993 | 1,000 | 0,959 | 3 layers of Convolution: 32, 64, 128 | 23,077 | 15,385 | 34,615 | 23,077 | 3,846 | 0,000 |
| 200 | 10 | 0,930 | 1,000 | 0,755 | 3 layers of Convolution: 32, 64, 128 | 30,769 | 23,077 | 15,385 | 26,923 | 3,846 | 0,000 |
| 200 | 10 | 0,973 | 1,000 | 0,939 | 3 layers of Convolution: 32, 64, 128 | 11,111 | 29,630 | 29,630 | 25,926 | 3,704 | 0,000 |
| 200 | 10 | 0,950 | 1,000 | 0,946 | 3 layers of Convolution: 32, 64, 128 | 15,625 | 31,250 | 18,750 | 25,000 | 9,375 | 0,000 |
| 200 | 10 | 0,867 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 24,000 | 44,000 | 24,000 | 8,000 | 0,000 | 0,000 |
| 200 | 10 | 0,893 | 1,000 | 0,925 | 3 layers of Convolution: 32, 64, 128 | 20,000 | 40,000 | 28,000 | 12,000 | 0,000 | 0,000 |
| 200 | 10 | 0,895 | 1,000 | 0,952 | 3 layers of Convolution: 32, 64, 128 | 25,000 | 37,500 | 30,000 | 7,500 | 0,000 | 0,000 |
| 20 | 20 | 0,946 | 1,000 | 0,973 | 3 layers of Convolution: 64, 128, 256 | 17,647 | 32,353 | 11,765 | 32,353 | 5,882 | 0,000 |
| 20 | 20 | 0,830 | 1,000 | 0,966 | 3 layers of Convolution: 64, 128, 256 | 32,000 | 36,000 | 8,000 | 16,000 | 8,000 | 0,000 |
| 20 | 20 | 0,965 | 1,000 | 0,980 | 3 layers of Convolution: 64, 128, 256 | 17,143 | 34,286 | 17,143 | 25,714 | 5,714 | 0,000 |
| 20 | 20 | 0,976 | 1,000 | 0,912 | 3 layers of Convolution: 64, 128, 256 | 12,500 | 28,125 | 25,000 | 25,000 | 9,375 | 0,000 |
| 20 | 20 | 0,905 | 1,000 | 0,986 | 3 layers of Convolution: 64, 128, 256 | 33,333 | 13,333 | 10,000 | 23,333 | 20,000 | 0,000 |
| 20 | 20 | 0,968 | 1,000 | 0,619 | 3 layers of Convolution: 64, 128, 256 | 17,241 | 34,483 | 24,138 | 17,241 | 6,897 | 0,000 |
| 20 | 20 | 0,970 | 1,000 | 0,952 | 3 layers of Convolution: 64, 128, 256 | 15,385 | 35,897 | 20,513 | 12,821 | 15,385 | 0,000 |
| 20 | 20 | 0,982 | 1,000 | 0,932 | 3 layers of Convolution: 64, 128, 256 | 19,231 | 26,923 | 26,923 | 23,077 | 3,846 | 0,000 |
| 20 | 20 | 0,982 | 1,000 | 0,898 | 3 layers of Convolution: 64, 128, 256 | 12,821 | 25,641 | 23,077 | 25,641 | 10,256 | 0,000 |
| 20 | 20 | 0,929 | 1,000 | 0,946 | 3 layers of Convolution: 64, 128, 256 | 29,630 | 18,519 | 18,519 | 25,926 | 3,704 | 0,000 |
| 20 | 20 | 0,950 | 1,000 | 0,782 | 3 layers of Convolution: 64, 128, 256 | 16,071 | 32,143 | 10,714 | 30,357 | 8,929 | 0,000 |
| 20 | 20 | 0,946 | 1,000 | 0,966 | 3 layers of Convolution: 64, 128, 256 | 27,273 | 22,727 | 18,182 | 25,000 | 6,818 | 0,000 |
| 20 | 20 | 0,940 | 1,000 | 0,939 | 3 layers of Convolution: 64, 128, 256 | 8,000 | 36,000 | 12,000 | 28,000 | 12,000 | 4,000 |
| 20 | 20 | 0,970 | 1,000 | 0,592 | 3 layers of Convolution: 64, 128, 256 | 11,111 | 24,444 | 31,111 | 22,222 | 11,111 | 0,000 |
| 20 | 20 | 0,878 | 0,971 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 20,000 | 28,000 | 36,000 | 8,000 | 0,000 | 0,000 |
| 20 | 20 | 0,959 | 0,997 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 25,000 | 21,429 | 21,429 | 28,571 | 0,000 | 0,000 |
| 20 | 20 | 0,936 | 1,000 | 0,524 | 3 layers of Convolution: 64, 128, 256 | 28,571 | 28,571 | 28,571 | 14,286 | 0,000 | 0,000 |
| 20 | 20 | 0,980 | 1,000 | 0,959 | 3 layers of Convolution: 64, 128, 256 | 10,638 | 31,915 | 27,660 | 14,894 | 12,766 | 0,000 |
| 20 | 20 | 0,972 | 1,000 | 0,803 | 3 layers of Convolution: 64, 128, 256 | 10,714 | 28,571 | 35,714 | 14,286 | 10,714 | 0,000 |
| 20 | 20 | 0,824 | 1,000 | 0,966 | 3 layers of Convolution: 64, 128, 256 | 12,000 | 60,000 | 28,000 | 0,000 | 0,000 | 0,000 |
| 200 | 10 | 0,870 | 0,968 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,981 | 0,974 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,963 | 0,974 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,976 | 0,965 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,992 | 0,980 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,979 | 0,971 | 0,959 | ANN without convolution | | | | | | |
| 200 | 10 | 0,959 | 0,974 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,939 | 0,977 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,865 | 0,977 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,983 | 0,974 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,879 | 0,974 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,956 | 0,959 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,953 | 0,977 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,967 | 0,956 | 0,946 | ANN without convolution | | | | | | |
| 200 | 10 | 0,983 | 0,968 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,977 | 0,977 | 0,946 | ANN without convolution | | | | | | |
| 200 | 10 | 0,955 | 0,968 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,938 | 0,971 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,970 | 0,968 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,978 | 0,974 | 0,973 | ANN without convolution | | | | | | |

| N1 | N2 | R ² | acc train | acc test | Details | 0.8 | 1.0 | 1.2 | 1.4 | 1.6 | 1.8 |
|-----|----|----------------|-----------|----------|---------------------------------------|--------|--------|--------|--------|--------|-------|
| 200 | 10 | 0,858 | 1,000 | 0,776 | 3 layers of Convolution: 32, 64, 128 | 26,667 | 16,667 | 26,667 | 23,333 | 6,667 | 0,000 |
| 200 | 10 | 0,791 | 1,000 | 0,932 | 3 layers of Convolution: 32, 64, 128 | 25,806 | 22,581 | 29,032 | 19,355 | 3,226 | 0,000 |
| 200 | 10 | 0,880 | 1,000 | 0,891 | 3 layers of Convolution: 32, 64, 128 | 6,667 | 33,333 | 31,111 | 24,444 | 4,444 | 0,000 |
| 200 | 10 | 0,877 | 1,000 | 0,510 | 3 layers of Convolution: 32, 64, 128 | 10,638 | 31,915 | 25,532 | 17,021 | 8,511 | 2,128 |
| 200 | 10 | 0,832 | 1,000 | 0,755 | 3 layers of Convolution: 32, 64, 128 | 20,000 | 24,000 | 24,000 | 20,000 | 0,000 | 4,000 |
| 200 | 10 | 0,889 | 0,991 | 0,510 | 3 layers of Convolution: 32, 64, 128 | 20,000 | 18,000 | 26,000 | 24,000 | 10,000 | 0,000 |
| 200 | 10 | 0,768 | 1,000 | 0,558 | 3 layers of Convolution: 32, 64, 128 | 26,471 | 20,588 | 35,294 | 8,824 | 2,941 | 2,941 |
| 200 | 10 | 0,895 | 1,000 | 0,959 | 3 layers of Convolution: 32, 64, 128 | 21,212 | 18,182 | 27,273 | 21,212 | 9,091 | 3,030 |
| 200 | 10 | 0,865 | 1,000 | 0,952 | 3 layers of Convolution: 32, 64, 128 | 21,429 | 14,286 | 28,571 | 21,429 | 10,714 | 0,000 |
| 200 | 10 | 0,839 | 1,000 | 0,946 | 3 layers of Convolution: 32, 64, 128 | 9,091 | 30,303 | 36,364 | 18,182 | 3,030 | 0,000 |
| 200 | 10 | 0,884 | 1,000 | 0,510 | 3 layers of Convolution: 32, 64, 128 | 7,692 | 19,231 | 26,923 | 23,077 | 7,692 | 0,000 |
| 200 | 10 | 0,820 | 1,000 | 0,510 | 3 layers of Convolution: 32, 64, 128 | 10,714 | 28,571 | 21,429 | 28,571 | 0,000 | 3,571 |
| 200 | 10 | 0,804 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 25,926 | 14,815 | 29,630 | 3,704 | 11,111 | 3,704 |
| 200 | 10 | 0,808 | 1,000 | 0,578 | 3 layers of Convolution: 32, 64, 128 | 15,385 | 34,615 | 19,231 | 23,077 | 3,846 | 0,000 |
| 200 | 10 | 0,784 | 1,000 | 0,959 | 3 layers of Convolution: 32, 64, 128 | 21,951 | 36,585 | 7,317 | 19,512 | 14,634 | 0,000 |
| 200 | 10 | 0,882 | 1,000 | 0,973 | 3 layers of Convolution: 32, 64, 128 | 21,277 | 21,277 | 17,021 | 27,660 | 8,511 | 0,000 |
| 200 | 10 | 0,875 | 1,000 | 0,959 | 3 layers of Convolution: 32, 64, 128 | 22,500 | 17,500 | 32,500 | 17,500 | 7,500 | 2,500 |
| 200 | 10 | 0,929 | 1,000 | 0,939 | 3 layers of Convolution: 32, 64, 128 | 12,121 | 18,182 | 27,273 | 24,242 | 12,121 | 0,000 |
| 200 | 10 | 0,789 | 1,000 | 0,952 | 3 layers of Convolution: 32, 64, 128 | 14,815 | 29,630 | 25,926 | 18,519 | 7,407 | 0,000 |
| 200 | 10 | 0,880 | 1,000 | 0,782 | 3 layers of Convolution: 32, 64, 128 | 15,217 | 23,913 | 15,217 | 32,609 | 6,522 | 0,000 |
| 20 | 20 | 0,953 | 1,000 | 0,912 | 3 layers of Convolution: 64, 128, 256 | 4,000 | 22,000 | 34,000 | 24,000 | 12,000 | 0,000 |
| 20 | 20 | 0,898 | 1,000 | 0,932 | 3 layers of Convolution: 64, 128, 256 | 16,000 | 22,000 | 22,000 | 20,000 | 16,000 | 0,000 |
| 20 | 20 | 0,941 | 1,000 | 0,517 | 3 layers of Convolution: 64, 128, 256 | 6,000 | 28,000 | 30,000 | 14,000 | 18,000 | 2,000 |
| 20 | 20 | 0,848 | 1,000 | 0,531 | 3 layers of Convolution: 64, 128, 256 | 20,000 | 20,000 | 34,000 | 14,000 | 6,000 | 2,000 |
| 20 | 20 | 0,818 | 1,000 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 18,000 | 24,000 | 26,000 | 20,000 | 8,000 | 0,000 |
| 20 | 20 | 0,915 | 1,000 | 0,966 | 3 layers of Convolution: 64, 128, 256 | 16,000 | 22,000 | 22,000 | 22,000 | 12,000 | 4,000 |
| 20 | 20 | 0,934 | 1,000 | 0,707 | 3 layers of Convolution: 64, 128, 256 | 10,000 | 24,000 | 20,000 | 34,000 | 10,000 | 0,000 |
| 20 | 20 | 0,845 | 1,000 | 0,878 | 3 layers of Convolution: 64, 128, 256 | 22,000 | 14,000 | 34,000 | 16,000 | 12,000 | 0,000 |
| 20 | 20 | 0,895 | 1,000 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 18,000 | 16,000 | 30,000 | 24,000 | 10,000 | 2,000 |
| 20 | 20 | 0,837 | 1,000 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 14,000 | 26,000 | 24,000 | 22,000 | 8,000 | 0,000 |
| 20 | 20 | 0,919 | 1,000 | 0,850 | 3 layers of Convolution: 64, 128, 256 | 18,000 | 14,000 | 36,000 | 26,000 | 6,000 | 0,000 |
| 20 | 20 | 0,921 | 1,000 | 0,959 | 3 layers of Convolution: 64, 128, 256 | 16,000 | 20,000 | 34,000 | 18,000 | 12,000 | 0,000 |
| 20 | 20 | 0,845 | 1,000 | 0,973 | 3 layers of Convolution: 64, 128, 256 | 12,000 | 30,000 | 34,000 | 16,000 | 4,000 | 0,000 |
| 20 | 20 | 0,874 | 1,000 | 0,986 | 3 layers of Convolution: 64, 128, 256 | 14,000 | 18,000 | 28,000 | 26,000 | 8,000 | 0,000 |
| 20 | 20 | 0,860 | 1,000 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 14,000 | 28,000 | 34,000 | 16,000 | 6,000 | 0,000 |
| 20 | 20 | 0,859 | 1,000 | 0,905 | 3 layers of Convolution: 64, 128, 256 | 20,000 | 24,000 | 24,000 | 16,000 | 14,000 | 0,000 |
| 20 | 20 | 0,825 | 1,000 | 0,973 | 3 layers of Convolution: 64, 128, 256 | 28,000 | 14,000 | 24,000 | 20,000 | 10,000 | 0,000 |
| 20 | 20 | 0,885 | 1,000 | 0,946 | 3 layers of Convolution: 64, 128, 256 | 12,000 | 34,000 | 20,000 | 24,000 | 8,000 | 0,000 |
| 20 | 20 | 0,889 | 1,000 | 0,844 | 3 layers of Convolution: 64, 128, 256 | 12,000 | 24,000 | 28,000 | 22,000 | 10,000 | 0,000 |
| 20 | 20 | 0,854 | 1,000 | 0,517 | 3 layers of Convolution: 64, 128, 256 | 18,000 | 16,000 | 34,000 | 24,000 | 2,000 | 2,000 |
| 200 | 10 | 0,839 | 0,971 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,950 | 0,965 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,730 | 0,971 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,847 | 0,953 | 0,959 | ANN without convolution | | | | | | |
| 200 | 10 | 0,822 | 0,971 | 0,959 | ANN without convolution | | | | | | |
| 200 | 10 | 0,850 | 0,950 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,852 | 0,971 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,875 | 0,968 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,756 | 0,965 | 0,980 | ANN without convolution | | | | | | |
| 200 | 10 | 0,908 | 0,965 | 0,980 | ANN without convolution | | | | | | |
| 200 | 10 | 0,846 | 0,974 | 0,946 | ANN without convolution | | | | | | |
| 200 | 10 | 0,767 | 0,983 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,907 | 0,980 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,874 | 0,959 | 0,959 | ANN without convolution | | | | | | |
| 200 | 10 | 0,763 | 0,971 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,807 | 0,950 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,813 | 0,977 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,884 | 0,965 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,897 | 0,985 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,876 | 0,959 | 0,966 | ANN without convolution | | | | | | |

| N1 | N2 | R ² | acc train | acc test | Details | 0.8 | 1.0 | 1.2 | 1.4 | 1.6 | 1.8 |
|-----|----|----------------|-----------|----------|---------------------------------------|--------|--------|--------|--------|--------|-------|
| 200 | 10 | 0,890 | 1,000 | 0,864 | 3 layers of Convolution: 32, 64, 128 | 13,889 | 22,222 | 19,444 | 27,778 | 8,333 | 0,000 |
| 200 | 10 | 0,831 | 1,000 | 0,510 | 3 layers of Convolution: 32, 64, 128 | 25,641 | 20,513 | 28,205 | 10,256 | 12,821 | 0,000 |
| 200 | 10 | 0,755 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 25,926 | 18,519 | 25,926 | 14,815 | 3,704 | 3,704 |
| 200 | 10 | 0,879 | 1,000 | 0,578 | 3 layers of Convolution: 32, 64, 128 | 22,857 | 17,143 | 20,000 | 31,429 | 5,714 | 0,000 |
| 200 | 10 | 0,799 | 1,000 | 0,946 | 3 layers of Convolution: 32, 64, 128 | 15,385 | 46,154 | 3,846 | 11,538 | 19,231 | 0,000 |
| 200 | 10 | 0,810 | 1,000 | 0,973 | 3 layers of Convolution: 32, 64, 128 | 33,333 | 7,407 | 18,519 | 29,630 | 0,000 | 3,704 |
| 200 | 10 | 0,710 | 1,000 | 0,973 | 3 layers of Convolution: 32, 64, 128 | 25,000 | 25,000 | 28,571 | 10,714 | 7,143 | 0,000 |
| 200 | 10 | 0,857 | 0,991 | 0,510 | 3 layers of Convolution: 32, 64, 128 | 16,000 | 32,000 | 20,000 | 20,000 | 8,000 | 0,000 |
| 200 | 10 | 0,905 | 1,000 | 0,980 | 3 layers of Convolution: 32, 64, 128 | 14,706 | 20,588 | 17,647 | 23,529 | 14,706 | 0,000 |
| 200 | 10 | 0,718 | 1,000 | 0,884 | 3 layers of Convolution: 32, 64, 128 | 26,667 | 23,333 | 20,000 | 20,000 | 3,333 | 0,000 |
| 200 | 10 | 0,808 | 1,000 | 0,973 | 3 layers of Convolution: 32, 64, 128 | 4,000 | 56,000 | 12,000 | 20,000 | 8,000 | 0,000 |
| 200 | 10 | 0,788 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 19,512 | 26,829 | 26,829 | 14,634 | 9,756 | 0,000 |
| 200 | 10 | 0,846 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 19,231 | 30,769 | 15,385 | 19,231 | 7,692 | 0,000 |
| 200 | 10 | 0,879 | 1,000 | 0,701 | 3 layers of Convolution: 32, 64, 128 | 10,256 | 20,513 | 25,641 | 28,205 | 5,128 | 0,000 |
| 200 | 10 | 0,803 | 1,000 | 0,694 | 3 layers of Convolution: 32, 64, 128 | 18,519 | 11,111 | 29,630 | 18,519 | 7,407 | 0,000 |
| 200 | 10 | 0,774 | 1,000 | 0,517 | 3 layers of Convolution: 32, 64, 128 | 12,000 | 36,000 | 28,000 | 8,000 | 8,000 | 4,000 |
| 200 | 10 | 0,800 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 20,000 | 17,500 | 22,500 | 17,500 | 12,500 | 0,000 |
| 200 | 10 | 0,763 | 1,000 | 0,510 | 3 layers of Convolution: 32, 64, 128 | 12,000 | 24,000 | 28,000 | 16,000 | 4,000 | 0,000 |
| 200 | 10 | 0,777 | 1,000 | 0,864 | 3 layers of Convolution: 32, 64, 128 | 20,000 | 32,000 | 24,000 | 16,000 | 4,000 | 4,000 |
| 200 | 10 | 0,744 | 1,000 | 0,966 | 3 layers of Convolution: 32, 64, 128 | 21,622 | 35,135 | 18,919 | 18,919 | 5,405 | 0,000 |
| 20 | 20 | 0,924 | 1,000 | 0,612 | 3 layers of Convolution: 64, 128, 256 | 16,000 | 22,000 | 20,000 | 22,000 | 14,000 | 0,000 |
| 20 | 20 | 0,909 | 1,000 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 14,000 | 24,000 | 24,000 | 24,000 | 8,000 | 0,000 |
| 20 | 20 | 0,943 | 0,980 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 14,000 | 22,000 | 24,000 | 20,000 | 12,000 | 0,000 |
| 20 | 20 | 0,920 | 1,000 | 0,946 | 3 layers of Convolution: 64, 128, 256 | 8,000 | 26,000 | 16,000 | 38,000 | 2,000 | 0,000 |
| 20 | 20 | 0,905 | 1,000 | 0,905 | 3 layers of Convolution: 64, 128, 256 | 16,000 | 30,000 | 18,000 | 20,000 | 12,000 | 0,000 |
| 20 | 20 | 0,846 | 1,000 | 0,973 | 3 layers of Convolution: 64, 128, 256 | 22,000 | 16,000 | 26,000 | 18,000 | 12,000 | 0,000 |
| 20 | 20 | 0,976 | 1,000 | 0,939 | 3 layers of Convolution: 64, 128, 256 | 10,000 | 20,000 | 32,000 | 18,000 | 14,000 | 2,000 |
| 20 | 20 | 0,894 | 1,000 | 0,857 | 3 layers of Convolution: 64, 128, 256 | 18,000 | 24,000 | 26,000 | 24,000 | 4,000 | 0,000 |
| 20 | 20 | 0,948 | 1,000 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 14,000 | 22,000 | 18,000 | 24,000 | 20,000 | 0,000 |
| 20 | 20 | 0,918 | 1,000 | 0,755 | 3 layers of Convolution: 64, 128, 256 | 18,000 | 16,000 | 22,000 | 22,000 | 16,000 | 2,000 |
| 20 | 20 | 0,932 | 1,000 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 16,000 | 18,000 | 12,000 | 36,000 | 18,000 | 0,000 |
| 20 | 20 | 0,966 | 1,000 | 0,939 | 3 layers of Convolution: 64, 128, 256 | 8,000 | 26,000 | 24,000 | 22,000 | 16,000 | 0,000 |
| 20 | 20 | 0,804 | 1,000 | 0,932 | 3 layers of Convolution: 64, 128, 256 | 28,000 | 22,000 | 16,000 | 10,000 | 20,000 | 0,000 |
| 20 | 20 | 0,961 | 1,000 | 0,925 | 3 layers of Convolution: 64, 128, 256 | 10,000 | 26,000 | 26,000 | 20,000 | 12,000 | 0,000 |
| 20 | 20 | 0,939 | 1,000 | 0,952 | 3 layers of Convolution: 64, 128, 256 | 10,000 | 32,000 | 26,000 | 22,000 | 6,000 | 0,000 |
| 20 | 20 | 0,962 | 1,000 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 12,000 | 14,000 | 24,000 | 26,000 | 20,000 | 0,000 |
| 20 | 20 | 0,898 | 1,000 | 0,973 | 3 layers of Convolution: 64, 128, 256 | 10,000 | 36,000 | 24,000 | 18,000 | 6,000 | 0,000 |
| 20 | 20 | 0,923 | 1,000 | 0,966 | 3 layers of Convolution: 64, 128, 256 | 12,000 | 26,000 | 20,000 | 24,000 | 10,000 | 0,000 |
| 20 | 20 | 0,811 | 1,000 | 0,952 | 3 layers of Convolution: 64, 128, 256 | 20,000 | 22,000 | 18,000 | 12,000 | 16,000 | 0,000 |
| 20 | 20 | 0,868 | 0,997 | 0,510 | 3 layers of Convolution: 64, 128, 256 | 20,000 | 24,000 | 22,000 | 16,000 | 16,000 | 2,000 |
| 200 | 10 | 0,939 | 0,974 | 0,959 | ANN without convolution | | | | | | |
| 200 | 10 | 0,900 | 0,968 | 0,959 | ANN without convolution | | | | | | |
| 200 | 10 | 0,945 | 0,974 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,946 | 0,980 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,819 | 0,977 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,814 | 0,965 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,735 | 0,977 | 0,959 | ANN without convolution | | | | | | |
| 200 | 10 | 0,778 | 0,959 | 0,973 | ANN without convolution | | | | | | |
| 200 | 10 | 0,924 | 0,968 | 0,946 | ANN without convolution | | | | | | |
| 200 | 10 | 0,798 | 0,959 | 0,946 | ANN without convolution | | | | | | |
| 200 | 10 | 0,930 | 0,968 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,727 | 0,962 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,864 | 0,956 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,910 | 0,971 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,925 | 0,968 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,934 | 0,956 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,935 | 0,965 | 0,952 | ANN without convolution | | | | | | |
| 200 | 10 | 0,933 | 0,953 | 0,939 | ANN without convolution | | | | | | |
| 200 | 10 | 0,909 | 0,974 | 0,966 | ANN without convolution | | | | | | |
| 200 | 10 | 0,909 | 0,974 | 0,966 | ANN without convolution | | | | | | |