INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DO PIAUÍ
PRÓ-REITORIA DE PESQUISA, PÓS-GRADUAÇÃO E INOVAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE MATERIAIS
DISSERTAÇÃO DE MESTRADO

OSMAR LEITE FERREIRA NETO

ZEASCAN: ferramenta computacional para classificação e tipificação de grãos de milho utilizando redes neurais convolucionais

TERESINA

2025

OSMAR LEITE FERREIRA NETO

ZEASCAN: ferramenta computacional para classificação e tipificação de grãos de milho utilizando redes neurais convolucionais

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia de Materiais do Instituto Federal do Piauí, como parte integrante dos requisitos para a obtenção do título de Mestre em Engenharia de Materiais.
Área de concentração: Computação Aplicada

Orientador: Prof. Dr. Otilio Paulo da Silva Neto

TERESINA-PI

2025

Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD

Ferreira Neto, Osmar Leite

F383z Zeascan : ferramenta computacional para classificação e tipificação de grãos de milho utilizando redes neurais convolucionais / Osmar Leite Ferreira Neto. - 2025.

123 f.: il. color.

Dissertação (Mestrado em Engenharia de Materiais) - Instituto Federal de Educação, Ciência e Tecnologia do Piauí, Campus Teresina Central, 2025.

Orientador : Prof Dr. Otílio Paulo da Silva Neto.

1. Visão computacional . 2. Automação . 3. Imagem. I.Título.

CDD - 620.1

Elaborado por Maria Rosismar Farias CRB 3/631

OSMAR LEITE FERREIRA NETO

ZEASCAN: ferramenta computacional para classificação e tipificação de grãos de milho utilizando redes neurais convolucionais

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Engenharia de Materiais do Instituto Federal do Piauí, como parte integrante dos requisitos para a obtenção do título de Mestre em Engenharia de Materiais.

Aprovada em: 22/08/2025

BANCA EXAMINADORA

_____

Prof. Dr. Otilio Paulo da Silva Neto (Orientador)
Instituto Federal do Piauí (IFPI)

_____

Prof. Dr. Rodolpho Carvalho Leite
Instituto Federal do Piauí (IFPI)

_____

Prof. Dr. Airton de Sá Brandim
Instituto Federal do Piauí (IFPI)

_____

Prof. Dr. Francisco Adelton Alves Ribeiro
Instituto Federal do Maranhão (IFMA)

AGRADECIMENTOS

Este trabalho só se tornou possível graças ao apoio, à paciência e à confiança de pessoas muito especiais que estiveram ao meu lado ao longo desta jornada.

À minha esposa, Tereza Cristina Sales Silva, meu amor e eterna gratidão. Seu incentivo constante, sua compreensão inabalável e sua presença firme em cada desafio e em cada pequena conquista foram minha maior fonte de força e motivação.

Ao meu pai, Anastacio Ferreira da Costa Filho, e à minha mãe, Deijames de Jesus Matos Leite Ferreira, agradeço por serem meu alicerce, meus exemplos e meu porto seguro. Sem o amor, os ensinamentos e os sacrifícios de vocês, esta conquista não seria possível. À minha irmã, Thaisa Nara Matos Leite Ferreira, obrigado por sua parceria incondicional, por sempre me inspirar a seguir em frente e por me lembrar, nos momentos difíceis, do valor da perseverança.

Ao meu orientador, Professor Otilio Paulo da Silva Neto, expresso meu sincero reconhecimento pela orientação atenta, pela paciência e pelo apoio constante. Sua dedicação e generosidade intelectual foram fundamentais para o desenvolvimento deste trabalho e para o meu crescimento acadêmico.

Aos amigos que caminharam comigo, nos momentos de descontração e nos de maior esforço, deixo meu profundo agradecimento. A amizade, os conselhos e a presença de vocês tornaram essa trajetória mais leve e significativa.

Ao Programa de Pós-Graduação em Engenharia de Materiais e Processos Industriais do Instituto Federal de Educação, Ciência e Tecnologia do Piauí (PPGEM-IFPI), agradeço pela oportunidade de crescimento acadêmico e profissional, pelo ambiente de aprendizado e pela estrutura oferecida ao longo do curso. Estendo esse agradecimento ao próprio IFPI, instituição que tem contribuído de forma significativa para o desenvolvimento da ciência, da tecnologia e da educação no nosso estado.

Por fim, agradeço a todos que, de alguma forma, contribuíram para essa realização. Cada gesto de apoio e cada palavra de incentivo tiveram papel essencial nessa caminhada. Sou imensamente grato a todos vocês.

“Só se pode alcançar um grande êxito quando
nos mantemos fiéis a nós mesmos."
Friedrich Nietzsche

RESUMO

O milho, cultura essencial para a segurança alimentar global e para a indústria agroalimentar, demanda métodos eficazes de controle de qualidade que minimizem perdas e assegurem conformidade com padrões técnicos. Nesse cenário, este trabalho apresenta o ZeaScan, um aplicativo móvel baseado em redes neurais convolucionais (CNNs), desenvolvido para automatizar a classificação de grãos de milho nas categorias sadios, ardidos e carunchados. A metodologia empregada envolveu o pré-processamento de imagens para o realce de características morfológicas e cromáticas, a aplicação de técnicas de aumento de dados para ampliar a capacidade de generalização do modelo e o treinamento de uma arquitetura de CNN otimizada para execução eficiente em dispositivos móveis. Os resultados demonstraram alto desempenho do sistema. Na classificação em três classes, foram obtidos valores globais de acurácia, precisão, recall e F1-score de 97,36%. Já na configuração binária, que diferencia grãos sadios de defeituosos, o modelo alcançou 97,50% de acurácia, 97,62% de precisão, 97,50% de recall e 97,53% de F1-score. Esses achados evidenciam a robustez do ZeaScan, capaz de reduzir a subjetividade dos métodos manuais e oferecer uma solução prática, inovadora e eficiente para o controle de qualidade do milho. O estudo representa um avanço no emprego da visão computacional aplicada à cadeia produtiva, disponibilizando uma ferramenta que alia rigor técnico e praticidade operacional, com potencial impacto positivo na sustentabilidade e na eficiência do agronegócio.

Palavras-chave: visão computacional; automação; imagem.

ABSTRACT

Maize, an essential crop for global food security and the agri-food industry, requires effective quality control methods to minimize losses and ensure compliance with technical standards. In this context, this study introduces ZeaScan, a mobile application based on convolutional neural networks (CNNs), designed to automate the classification of maize kernels into healthy, mold-damaged, and weevil-damaged categories. The methodology involved image preprocessing to enhance morphological and chromatic features, the application of data augmentation techniques to improve the model's generalization capacity, and the training of a CNN architecture optimized for efficient execution on mobile devices. The results demonstrated high system performance. In the three-class classification, global accuracy, precision, recall, and F1-score values of 97.36% were achieved. In the binary configuration, which differentiates healthy from defective kernels, the model reached 97.50% accuracy, 97.62% precision, 97.50% recall, and 97.53% F1-score. These findings highlight the robustness of ZeaScan, capable of reducing the subjectivity of manual methods and providing a practical, innovative, and efficient solution for maize quality control. This study represents a significant advancement in the application of computer vision to the production chain, offering a tool that combines technical rigor with operational practicality, with potential positive impacts on the sustainability and efficiency of agribusiness.

Keywords: computer vision; automation; image processing.

LISTA DE FIGURAS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

BP	*Back-Propagation*
CNN	*Convolutional Neural Networks*
DNN	*Deep Neural Networks*
Embrapa	Empresa Brasileira de Pesquisa Agropecuária
et al.	*et alii*
FN	Falso Negativo
FP	Falso Positivo
GA	*Genetic Algorithms*
GLCM	*Gray-Level Co-occurrence Matrix*
GS	*Grid Search*
HSV	*Hue, Saturation, Value*
HTTP	*Hypertext Transfer Protocol*
IA	Inteligência Artificial
IN	Instrução Normativa
JSON	*JavaScript Object Notation*
KNN	*K-Nearest Neighbors*
L.	*Linnaeus*
MAPA	Ministério da Agricultura, Pecuária e Abastecimento
MEI	Materiais Estranhos e Impurezas
MVC	*Model-View-Controller*
PCA	*Principal Component Analysis*
PDI	Processamento Digital de Imagens
PSO	*Particle Swarm Optimization*
ReLu	*Rectified Linear Unit*
RF	*Random Forest*
RGB	*Red, Green, Blue*
RN	Redes Neurais

RNA	Redes Neurais Artificiais
ROC	*Receiver Operating Characteristic*
ROI	*Region of Interest*
SCRI	Secretaria de Comércio e Relações Internacionais
SENAR	Serviço Nacional de Aprendizagem Rural
SIG	Sistemas de Informação Geográfica
SVM	*Support Vector Machine*
VANT	Veículo Aéreo Não Tripulado
VC	Visão Computacional
VGG	*Visual Geometry Group*
VN	Verdadeiro Negativo
VP	Verdadeiro Positivo
vs.	*versus*
WSGI	*Web Server Gateway Interface*

SUMÁRIO

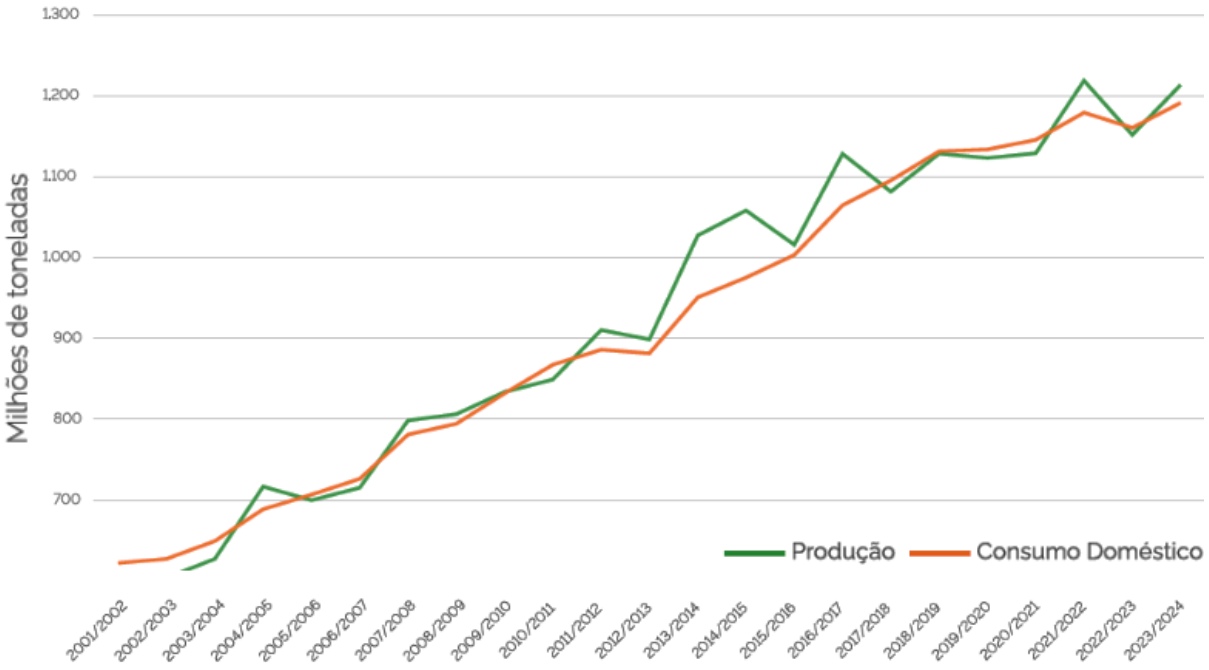
1 INTRODUÇÃO

O milho (*Zea mays Linnaeus*) figura entre as culturas agrícolas mais relevantes do mundo, sustentando tanto a alimentação humana direta — por meio de farinha, óleo e grãos integrais — quanto a produção de insumos para a pecuária. Essa multifuncionalidade consolida o milho como elemento-chave para a segurança alimentar mundial e para a promoção de sistemas econômicos mais sustentáveis (Mauri *et al.*, 2022).

Segundo o Ministério da Agricultura, Pecuária e Abastecimento (MAPA, 2024), por meio da Secretaria de Comércio e Relações Internacionais (SCRI), aproximadamente 83% do milho produzido globalmente é destinado à formulação de rações, situando o cereal como pilar da cadeia de produção de proteínas animais. Além disso, seu uso crescente na indústria de biocombustíveis contribui para diversificar a matriz energética e mitigar as emissões de gases de efeito estufa.

A Figura 1 apresenta a evolução paralela da oferta e da demanda mundial de milho entre as safras de 2001/02 e a projeção de 2023/24.

Figura 1 – Oferta e consumo mundial de milho



Fonte: Adaptado de Brasil. Ministério da Agricultura e Pecuária, 2024.

No intervalo de tempo apresentado na Figura 1, a produção mundial de milho passou de aproximadamente 620 milhões para cerca de 1,22 bilhão de toneladas na safra 2021/22, enquanto o consumo global acompanhou esse crescimento de forma quase uniforme, atingindo 1,18 bilhão de toneladas. A estreita distância entre as curvas de produção e consumo revela um mercado altamente ajustado. Para a safra 2022/23, as projeções indicam uma produção de 1,15 bilhão de toneladas frente a um consumo estimado de 1,16 bilhão, sugerindo possíveis déficits em decorrência de eventos climáticos adversos ou tensões geopolíticas (MAPA, 2024).

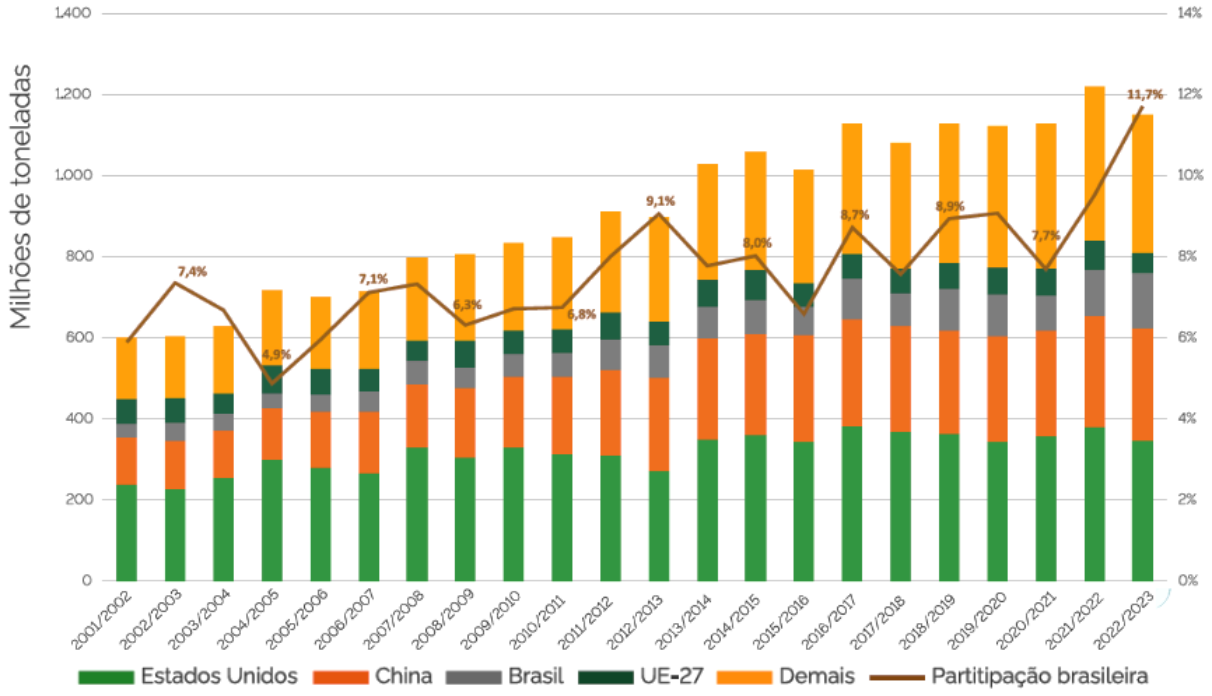
Esse comportamento reflete a expansão contínua da demanda, impulsionada pelo crescimento da cadeia de proteína animal e pelo uso crescente do milho na produção de biocombustíveis. A persistente convergência entre oferta e consumo evidencia a necessidade de políticas públicas e de sistemas de monitoramento capazes de mitigar riscos de escassez e preservar o equilíbrio do mercado global de milho no médio e longo prazos (Mauri *et al.*, 2022).

A Figura 2 mostra uma comparação da produção de milho nos principais polos mundiais — Estados Unidos, China, Brasil, União Europeia (UE-27) e demais produtores — entre as safras de 2001/02 e 2022/23. Nota-se que Estados Unidos e China, juntos, mantiveram trajetória ascendente contínua, passando de aproximadamente 380 milhões de toneladas na safra 2001/02 para 625,95 milhões de toneladas na última safra, o que corresponde a uma parcela significativa da produção mundial.

O Brasil, por sua vez, apresentou um avanço expressivo, passando de aproximadamente 40 milhões para 134,55 milhões de toneladas no mesmo intervalo. Já a UE-27 manteve-se relativamente estável, oscilando entre 60 e 70 milhões de toneladas, enquanto o grupo classificado como “demais produtores” cresceu de forma gradual, refletindo a expansão de fronteiras agrícolas em outras regiões (MAPA, 2024).

Entre as safras de 2001/02 e 2022/23, a participação do Brasil na produção mundial de milho passou de 6,1% para 11,7%. Esse avanço resulta da combinação de políticas de incentivo ao setor, dos progressos em biotecnologia e da adoção de sistemas de cultivo sucessivos. Embora Estados Unidos e China ainda respondam por mais de 50% da oferta global, o fortalecimento do agronegócio brasileiro diversifica as fontes de suprimento e aumenta a resiliência do mercado diante de oscilações climáticas e geopolíticas (MAPA, 2024).

Figura 2 – Produção de milho no mundo e participação brasileira



Fonte: Adaptado de Brasil. Ministério da Agricultura e Pecuária, 2024.

O milho possui relevância econômica expressiva devido aos seus múltiplos destinos, que abrangem desde a formulação de rações animais até aplicações em indústrias de alta tecnologia. Para assegurar a oferta de grãos de elevada qualidade e isentos de patógenos, os produtores adotam sistemas rigorosos de classificação e padronização como ferramentas de controle de qualidade. Os relatórios gerados por esses processos são requisitados pelos consumidores finais, pois atestam a conformidade do produto com padrões de segurança alimentar e de desempenho (Mauri *et al.*, 2022).

Os grãos de milho são analisados e classificados para obtenção de informações quanto ao tipo, variedade, qualidade e rendimento. Segundo Mauri *et al.* (2022), sementes limpas e livres de pragas são consideradas sementes de alta qualidade. Os processos de medição da qualidade são normalmente realizados por pessoas qualificadas que devem seguir os padrões e procedimentos das normas vigentes (Oliveira, 2021).

A classificação dos grãos de milho requer a análise de cada grão para verificar se há defeitos. Os técnicos analisam a amostra e separam os grãos com avarias dos grãos sadios. Além dos tipos de danos que precisam ser determinados, a amostra pode conter materiais estranhos e impurezas (Oliveira, 2021).

Embora os padrões e processos de classificação sejam concebidos para melhorar a qualidade dos grãos, ainda existem lacunas que podem levar a uma classificação imprecisa dos lotes de milho (Devechio *et al.*, 2023). Apresentar a mesma amostra a revisores diferentes pode produzir resultados diferentes. A variação nas avaliações se deve à necessidade de diferentes observadores para classificar o milho com precisão, além de extenso treinamento em habilidades técnicas para padronizar a classificação (Matos *et al.*, 2023). Isso levou ao desenvolvimento de técnicas automatizadas para a classificação de grãos de milho, como as Redes Neurais Convolucionais (*Convolutional Neural Networks* – CNNs) .

A classificação manual é subjetiva porque leva em consideração a visão e interpretação do classificador. Portanto, independentemente da qualificação do técnico, ainda pode ocorrer erro humano durante o procedimento, levando a relatórios imprecisos e, consequentemente, perdas financeiras e danos à saúde do consumidor (Oliveira, 2021).

Nesse contexto, a visão computacional poderá se tornar uma parte importante das operações automatizadas de processamento de alimentos. Algoritmos desse tipo podem ter desempenho satisfatório, permitindo captura e processamento de imagens de baixo custo com a vantagem da portabilidade. Assim, a visão computacional mostra-se uma alternativa de inspeção objetiva, rápida, econômica e consistente para classificar grãos como o milho (Chen *et al.*, 2020; Gayatri *et al.*, 2021).

1.1 OBJETIVOS

1.1.1 Geral

Desenvolver um aplicativo móvel, ZeaScan, concebido como ferramenta computacional para a classificação automática de grãos de milho em três categorias, utilizando um modelo de visão computacional baseado em redes neurais convolucionais.

1.1.2 Específicos

Para atingir o objetivo geral deste trabalho, propõem-se os seguintes objetivos específicos:

- Preparar e estruturar o conjunto de imagens de grãos de milho, incluindo seleção, rotulagem por classe (defeitos, qualidade), pré-processamento e aplicação de técnicas de aumento de dados (*data augmentation*) para garantir representatividade.
- Projetar e otimizar a arquitetura de uma CNN, definindo camadas, filtros, funções de ativação e ajustando hiperparâmetros críticos (taxa de aprendizagem, *batch size*, otimizadores) para eficiência do modelo.
- Treinar e validar o modelo de CNN, utilizando o banco de imagens preparado e técnicas como validação cruzada (*cross-validation*) para evitar *overfitting*.
- Avaliar o desempenho do modelo por meio de métricas como acurácia, precisão, *recall*, *F1-score* e análise da matriz de confusão, garantindo robustez na classificação de defeitos.
- Implementar algoritmos de pós-processamento, incluindo segmentação de grãos individuais, cálculo da área ocupada por defeitos e tipificação da amostra em categorias de qualidade (Tipo 1, 2, 3 ou Fora de Tipo).
- Desenvolver um aplicativo móvel com interface intuitiva para captura de imagens em tempo real, integração do modelo treinado e compatibilidade com sistemas Android, garantindo eficiência computacional.

2 REFERENCIAL TEÓRICO

2.1 GRÃOS DE MILHO

Os grãos de milho, representam as sementes da planta do milho (*Zea Mays L.*) e desempenham diversas funções em diferentes indústrias. Esses grãos são compostos principalmente de endosperma amiláceo e são essenciais para a produção de alimentos como fubá, óleo de milho, xarope de milho, cereais para consumo humano e uma variedade de outros produtos alimentícios que fazem parte do nosso dia a dia (Bazame *et al.*, 2023).

Segundo Pinheiro *et al.* (2021), o milho ocupa posição de destaque no cenário agrícola global, consolidando-se como o terceiro alimento mais produzido mundialmente. Tal estatística evidencia sua relevância econômica e funcional, refletindo o papel estratégico que este cereal versátil desempenha em múltiplos setores, desde a alimentação humana e animal até a produção de biocombustíveis e insumos industriais.

O milho destaca-se por sua notável versatilidade, sendo amplamente empregado na alimentação animal e na indústria de alta tecnologia. Além de constituir uma fonte nutricional relevante para a produção pecuária, é matéria-prima estratégica na fabricação de bioplásticos, biocombustíveis e materiais de construção sustentáveis. Sua aplicação em setores industriais de vanguarda reforça sua importância na matriz econômica global e evidencia seu potencial como insumo para uma economia mais sustentável (De Matos *et al.*, 2024).

No entanto, para manter a qualidade e a segurança do cultivo do milho, os produtores enfrentam diversos desafios. Eles precisam garantir que os grãos produzidos sejam de alta qualidade, livres de agentes patogênicos e adequados para o consumo humano e animal. Isso envolve a implementação de práticas agrícolas sustentáveis e o uso responsável de fertilizantes e defensivos agrícolas (Souza; Queiroz, 2023).

É fundamental investir em pesquisa e desenvolvimento de novas tecnologias e variedades de milho. Isso inclui a criação de cultivos geneticamente modificados resistentes a pragas e doenças, bem como o aprimoramento de técnicas inovadoras de manejo e colheita (Souza, *et al.*, 2020). Tais medidas são essenciais para garantir a sustentabilidade e viabilidade da produção de milho a longo prazo, à medida que a demanda global continua a crescer.

2.1.1 Defeitos em Grãos de Milho

A Instrução Normativa (IN) nº 60, de 8 de novembro de 2011, do Ministério da Agricultura, Pecuária e Abastecimento (MAPA) estabelece o Regulamento Técnico do Milho (Brasil, 2011). Essa norma identifica sete tipos de defeitos que podem ser encontrados nos grãos de milho: ardidos, chochos ou imaturos, fermentados, germinados, gessados, mofados e carunchados. Na Figura 3 é possível observar o aspecto característico de cada uma dessas avarias nos grãos.

Figura 3 – Tipos de avarias em grãos de milho

Fonte: Adaptado do Serviço Nacional de Aprendizagem Rural (SENAR), 2017.

Esses defeitos podem comprometer a qualidade dos grãos de milho, reduzindo sua utilidade para consumo ou processamento industrial. A identificação adequada dessas avarias é fundamental durante a compra ou o processamento dos grãos, a fim de assegurar a obtenção de um produto final com padrão de qualidade elevado (Silva *et al.*, 2021).

Grãos ardidos são aqueles que sofreram danos devido ao calor excessivo durante a colheita ou armazenamento. Grãos chochos ou imaturos são aqueles que não atingiram seu pleno desenvolvimento, apresentando uma aparência enrugada ou subdesenvolvida. Grãos fermentados resultam da exposição a condições de umidade excessiva, levando à fermentação e ao desenvolvimento de odores e sabores indesejáveis. Grãos germinados são aqueles que iniciaram o processo de brotamento, o que pode indicar condições inadequadas de armazenamento. Grãos

gessados possuem um revestimento de gesso, adicionado durante o processamento para melhorar a aparência, mas que pode comprometer a qualidade do grão. Finalmente, grãos mofados ou carunchados estão contaminados com fungos ou insetos, tornando-os inadequados para o consumo humano ou animal (Brasil, 2011; SENAR, 2017; Silva *et al.*, 2021). A identificação e a prevenção desses defeitos são essenciais para assegurar a segurança alimentar e a qualidade do produto final (Matos *et al.*, 2023).

2.1.2 Classificação dos Grãos de Milho

A classificação dos grãos de milho no Brasil é regulamentada por um conjunto de leis sob a supervisão do Ministério da Agricultura, Pecuária e Abastecimento. A Lei Ordinária nº 10.711/2003 (Brasil, 2003) estabelece os padrões para essa classificação, que é fundamental para garantir a qualidade dos grãos e atender aos requisitos dos mercados nacional e internacional. A classificação é obrigatória quando os produtos se destinam diretamente ao consumo humano, em operações de comércio governamental e durante processos de importação realizados em portos, aeroportos e postos de fronteira. O MAPA é o órgão responsável pela organização, fiscalização e controle desse processo.

A classificação dos grãos pode ser realizada por estados, Distrito Federal, cooperativas agrícolas, empresas, bolsas de mercadorias, universidades e institutos de pesquisa que sejam devidamente credenciados junto ao MAPA. As pessoas físicas e jurídicas envolvidas nesse processo deverão ser registradas no cadastro geral de classificação. O descumprimento das normas pode resultar em advertências como advertências, multas, apreensão de produtos, interdição do estabelecimento e até suspensão ou cancelamento do credenciamento (Matos *et al.*, 2023).

A Lei nº 9.972, de 25 de maio de 2000, institui o regime geral de classificação de produtos vegetais, subprodutos e resíduos de valor econômico, estabelecendo as bases para a uniformização e a transparência na comercialização desses itens (Brasil, 2000). No caso específico do milho (*Zea mays L.*), a Instrução Normativa nº 60/2011, define critérios de qualidade — como teor de umidade, impurezas, matérias estranhas, percentuais de grãos ardidos e carunchados —, as classes comerciais (Tipos 1, 2 e 3), os procedimentos de amostragem e análise laboratorial, bem como

a metodologia para o cálculo do fator de redução em lotes fora de especificação (Brasil, 2011).

O processo de classificação de grãos compreende sete etapas sequenciais: (i) amostragem; (ii) homogeneização; (iii) quarteamento; (iv) avaliação de Matérias Estranhas e Impurezas (MEI); (v) determinação do teor de umidade; (vi) identificação do grupo, da classe e do tipo; e, por fim, (vii) emissão do laudo de classificação (Matos *et al.*, 2023).

De acordo com a Instrução Normativa nº 60/2011 do Ministério da Agricultura, Pecuária e Abastecimento (Brasil, 2011), a classificação oficial dos grãos de milho baseia-se em três critérios complementares: (i) Grupo, determinado pela consistência e pelo formato dos grãos; (ii) Classe, definida pela predominância da cor; e (iii) Tipo, estabelecido com base em parâmetros específicos de qualidade. Conforme o § 1º do art. 5º da referida Instrução Normativa, os grãos devem ser classificados nos seguintes Grupos, segundo sua consistência e morfologia:

> I - **duro:** quando apresentar o mínimo de 85% em peso de grãos com as características de duro, ou seja, apresentando endosperma predominantemente córneo, exibindo aspecto vítreo; quanto ao formato, considera-se duro o grão que se apresentar predominantemente ovalado e com a coroa convexa e lisa;
>
> II - **dentado**: quando apresentar o mínimo de 85% em peso de grãos com as características de dentado, ou seja, com consistência parcial ou totalmente farinácea; quanto ao formato, considera-se dentado o grão que se apresentar predominantemente dentado com a coroa apresentando uma reentrância acentuada;
>
> III - **semiduro:** quando apresentar o mínimo de 85% em peso de grãos com consistência e formato intermediários entre duro e dentado; e
>
> IV - **misturado:** quando não estiver compreendido nos grupos anteriores, especificando-se no documento de classificação as percentagens da mistura de outros grupos.

No § 2º do art. 5º da Instrução Normativa nº 60/2011 do Ministério da Agricultura, Pecuária e Abastecimento (Brasil, 2011), que dispõe sobre a classificação do milho de acordo com a coloração do grão, será classificado nas seguintes Classes:

I - **amarela:** constituída de milho que contenha no mínimo 95% (noventa e cinco por cento), em peso, de grãos amarelos, amarelo pálido ou amarelo alaranjado; o grão de milho amarelo com ligeira coloração vermelha ou rósea no pericarpo será considerado da classe amarela;

II - **branca:** constituída de milho que contenha no mínimo 95% (noventa e cinco por cento), em peso, de grãos brancos; o grão de milho com coloração marfim ou palha será considerado da classe branca;

III - **cores:** constituída de milho que contenha no mínimo 95% (noventa e cinco por cento), em peso, de grãos de coloração uniforme, mas diferentes das classes amarela e branca; o grão de milho com ligeira variação na coloração do pericarpo será considerado da cor predominante; e

IV - **misturada:** constituída de milho que não se enquadra em nenhuma das classes anteriores.

Segundo o § 3º do art. 5º da Instrução Normativa nº 60/2011 do Ministério da Agricultura, Pecuária e Abastecimento (Brasil, 2011), o milho é classificado em três tipos, definidos pela qualidade do lote e pelos limites máximos de tolerância estipulados na norma. Quando esses limites são excedidos, o lote passa a ser enquadrado como Fora de Tipo ou Desclassificado. A Tabela 1 apresenta, de forma detalhada, os limites e requisitos de qualidade que regem essa classificação.

Tabela 1 – Limites máximos de tolerâncias para o milho expressos em % do peso

Enquadramento	Grãos Avariados		Grãos quebrados	Matérias Estranhas e Impurezas	Carunchados
	Ardidos	Total			
Tipo 1	1,0	6,0	3,0	1,0	2,0
Tipo 2	2,0	10,0	4,0	1,5	3,0
Tipo 3	3,0	15,0	5,0	2,0	4,0
Fora de tipo	5,0	20,0	> 5,0	> 2,0	8,0

Fonte: Adaptado de Brasil. Ministério da Agricultura e Pecuária, 2011.

Na tipificação por qualidade, Tabela 1, os grãos de milho são classificados como Tipo 1, Tipo 2, Tipo 3 ou Fora de Tipo — esta última categoria aplica-se quando a amostra não satisfaz os requisitos mínimos dos três primeiros tipos. O enquadramento baseia-se na proporção de grãos avariados em relação aos sadios:

à medida que o percentual de defeitos aumenta, a amostra é reclassificada para um tipo inferior (Brasil, 2011).

Essa padronização é fundamental para garantir a qualidade, a segurança alimentar e a conformidade com normas nacionais e internacionais. Além disso, fortalece a credibilidade do agronegócio brasileiro, contribuindo para sua competitividade no mercado global e assegurando que os produtos atendam às exigências de consumidores e parceiros comerciais em todo o mundo (Matos *et al.*, 2023).

2.1.3 Tolerância e qualidade

Os critérios de qualidade do milho concentram-se no formato e na coloração dos grãos, aliados a limites de tolerância previamente definidos. Nesse sistema, o cereal é classificado em três níveis hierárquicos: Grupos, Classes e Tipos, sendo que a inclusão em um grupo específico depende predominantemente do formato do grão (Oliveira *et al.*, 2021).

O § 3º do art. 5º da IN nº 60/2011 do MAPA (Brasil, 2011) define os critérios de classificação do milho, especificando os limites percentuais de grãos ardidos e carunchados que determinam o enquadramento nos Tipos 1, 2, 3 ou “Fora de Tipo”. Para o Tipo 1, os parâmetros são rigorosos: admitem-se até 1 % de grãos ardidos e 2 % de carunchados, refletindo a exigência por lotes de alta qualidade. Nos Tipos 2 e 3, esses percentuais aumentam gradualmente — 2 % e 3 % para ardidos; 3 % e 4 % para carunchados, respectivamente —, preservando, contudo, padrões mínimos de integridade que equilibram flexibilidade comercial e controle sanitário. Quando os valores ultrapassam os limites do Tipo 3 — por exemplo, 5% de ardidos ou 8 % de carunchados —, o lote é classificado como “Fora de Tipo”, ficando impedido de ser comercializado diretamente e exigindo rebeneficiamento ou descarte.

Essa hierarquização de tolerâncias não só padroniza a produção, como também reforça a segurança alimentar, evitando que grãos com deterioração avançada ou infestação afetem a confiabilidade do setor agrícola. Dessa forma, a norma garante que apenas produtos conformes aos requisitos técnicos e sanitários cheguem ao mercado, preservando a qualidade e a reputação da cadeia produtiva nacional (Oliveira *et al.*, 2021; Matos *et al.*, 2023).

A classificação do milho deve obedecer aos limites de tolerância da Tabela 1

para cada Tipo. Quando o lote é enquadrado como Fora de Tipo por excesso de grãos ardidos, avariados ou carunchados, ele pode ser vendido tal como está, desde que identificado como Fora de Tipo, ou após rebeneficiamento. Se a descaracterização decorrer de grãos quebrados ou impurezas, a comercialização só é permitida depois do rebeneficiamento que ajuste o produto a um tipo específico. Já lotes com insetos vivos ou outras pragas não podem ser vendidos antes de passarem por expurgo obrigatório (Brasil, 2011; Pinheiro *et al.*, 2021; Matos *et al.*, 2023).

Quando o milho apresenta condições de mau estado — mofo, fermentação, sementes tratadas ou tóxicas, odor impróprio ou quaisquer valores que excedam os limites da Tabela 1 —, ele é desclassificado e sua comercialização fica proibida. Nesses casos, a unidade de classificação deve emitir o respectivo Laudo de Classificação, registrando o produto como “Desclassificado”. Assim, enquanto os Tipos refletem níveis crescentes de integridade, as categorias Fora de Tipo e Desclassificado impõem restrições ou impedimentos de venda, garantindo a preservação da saúde pública e a transparência do mercado agrícola (Brasil, 2011; Pinheiro *et al.*, 2021).

A Superintendência Federal de Agricultura pode adotar medidas adicionais sobre qualquer lote de milho, e o MAPA reserva-se o direito de realizar novas análises, independentemente do laudo inicial. O grão deve estar fisiologicamente maduro, sadio, limpo, seco e dentro dos limites de tolerância — inclusive umidade máxima de 14 %. Se esse teor for superado, a venda só é permitida quando a condição estiver expressa no documento de classificação (Brasil, 2011; Pinheiro *et al.*, 2021; Silva *et al.*, 2021).

2.2 INTELIGÊNCIA ARTIFICIAL

2.2.1 Visão Computacional

A Visão Computacional (VC) é um subcampo da Inteligência Artificial (IA) que desenvolve algoritmos com o objetivo de replicar, de forma automatizada, as capacidades do sistema visual humano. Inspirada na complexidade do córtex visual (que possui mais de 140 milhões de neurônios), essa área busca permitir que

máquinas adquiram, processem, analisem e interpretem informações visuais, reproduzindo habilidades como segmentação, integração de dados da retina e reconhecimento rápido de objetos e padrões (Zhou *et al.*, 2023a).

Para tornar os problemas visuais mais tratáveis do ponto de vista computacional, os métodos de VC geralmente se concentram em tarefas específicas, como detecção e classificação de objetos, análise de movimento e compreensão espacial, aproximando-se de funções-chave desempenhadas pelo sistema visual humano (Chadebecq *et al.*, 2020).

Do ponto de vista da produção agrícola e da alimentação humana, essa tecnologia viabiliza a rastreabilidade de todo o sistema produtivo, desde o produtor até o consumidor final. Esse monitoramento integral proporciona maior segurança alimentar e pode se configurar como um diferencial no mercado, representando uma vantagem competitiva significativa (Patel *et al.*, 2022).

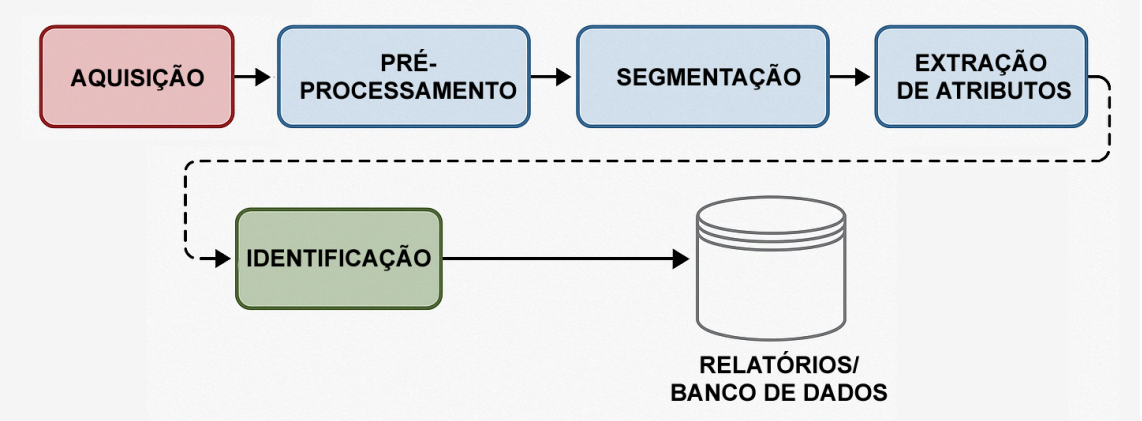
A visão computacional teve grandes avanços nas últimas duas décadas, impulsionados por inovações em poder computacional, sensores de imagem, modelagem matemática e, sobretudo, pelo desenvolvimento de técnicas de aprendizado profundo (*deep learning*). Esses avanços tecnológicos permitiram que sistemas de visão se tornassem mais precisos, eficientes e aplicáveis a uma ampla gama de contextos, desde aplicações industriais até a agricultura de precisão (Chadebecq *et al.*, 2020).

Diferentemente dos sistemas tradicionais, que dependiam fortemente da intervenção humana para projetar e selecionar manualmente as características relevantes das imagens, os modelos baseados em aprendizado profundo são capazes de aprender autonomamente representações discriminantes diretamente dos dados, desde que em quantidade suficiente e devidamente anotados. Essa evolução reduziu a necessidade de conhecimento prévio específico do domínio e ampliou significativamente a aplicabilidade da visão computacional em tarefas complexas, como classificação de objetos, segmentação semântica e detecção de padrões em imagens de diferentes contextos (Chadebecq *et al.*, 2020).

A visão computacional consiste na construção de descrições compreensíveis e significativas de objetos físicos a partir de imagens digitais. Entre suas principais vantagens estão a geração de dados descritivos e precisos, a criação de registros permanentes que possibilitam análises mais aprofundadas e, ainda, a automatização de etapas tradicionalmente realizadas por operadores humanos (Ward *et al.*, 2021).

A Figura 4 apresenta as etapas sequenciais de um sistema de Visão Computacional, abrangendo: aquisição da imagem, processamento (ajustes e normalização), pré-segmentação (isolamento de regiões de interesse), extração de atributos (quantificação de características relevantes), identificação (classificação dos objetos) e, por fim, armazenamento dos dados em relatórios ou bancos de dados (Cavalcanti Neto *et al.*, 2015).

Figura 4 – Etapas de um sistema de visão computacional típico



Fonte: Adaptado de Cavalcanti Neto *et al.*, 2015.

O fluxo operacional ilustrado na Figura 4 converte dados brutos extraídos de imagens em informações estruturadas, possibilitando sua aplicação em diversos domínios, como automação, diagnóstico médico e monitoramento industrial. Contudo, a execução de todas as etapas não é obrigatória em todos os casos, uma vez que sua utilização depende do contexto específico e dos objetivos da análise (Cavalcanti Neto *et al.*, 2015).

2.2.1.1 Aquisição de Imagens

O processo atual de obtenção de imagens é geralmente realizado com o uso de câmeras fotográficas digitais, que podem ser adquiridas como dispositivos independentes ou integradas a outros aparelhos eletrônicos, como microcomputadores, notebooks, *tablets* e *smartphones*. A escolha do equipamento deve considerar tanto o tipo de objeto a ser fotografado quanto o orçamento disponível, avaliando-se a relação custo-benefício, uma vez que há, atualmente,

uma ampla variedade de câmeras, marcas e faixas de preço no mercado (Silva, 2020).

O primeiro estágio do Processamento Digital de Imagens (PDI) envolve a captura, geração e digitalização da imagem. Uma imagem digital pode ser compreendida como uma matriz bidimensional, na qual os índices das linhas e colunas representam os pontos (ou *pixels*) da imagem (Marques Junior; Ulson, 2021).

Cada elemento desta matriz corresponde a um valor de cor ou intensidade de *pixel*. Quando o ambiente de captura é controlado, o processo pode incluir um sistema de iluminação para garantir uniformidade e qualidade visual (Vo *et al.*, 2020).

Após a captura, as imagens passam por um estágio de pré-processamento, que pode envolver conversão de cor e de formato, filtragem para redução de ruído e normalização de intensidade. Além disso, aplicam-se as técnicas de amostragem e quantização, com o objetivo de ajustar, respectivamente, a resolução da imagem e o número de cores ou níveis de cinza, conforme os requisitos da aplicação (Wang *et al.*, 2020).

A qualidade da imagem adquirida influencia diretamente as etapas seguintes do processamento digital, como segmentação e classificação. Por isso, é essencial garantir boas condições de iluminação, foco e estabilidade durante a captura. Em aplicações específicas, como inspeção de grãos ou exames médicos, pode ser necessário o uso de câmeras com alta resolução ou sensores especiais, como os hiperespectrais. Além disso, a padronização dos parâmetros de captura contribui para a reprodutibilidade e a confiabilidade dos resultados (Silva, 2020).

2.2.1.2 Pré-Processamento

O pré-processamento está intimamente relacionado aos conceitos de recuperação e expansão. O primeiro método visa compensar erros que ocorrem durante a captura da imagem, enquanto o segundo método visa corrigir o máximo da imagem, mesmo que detalhes sejam perdidos (Marques Junior; Ulson, 2021).

A qualidade da imagem é essencial, pois influencia diretamente a precisão da análise computacional (Silva, 2020). Por essa razão, técnicas de pré-processamento são amplamente aplicadas. Métodos como a filtragem de ruído e a equalização de

histograma aumentam o contraste, melhoram a nitidez e reduzem artefatos, otimizando a análise nas etapas subsequentes (Tavares *et al.*, 2024).

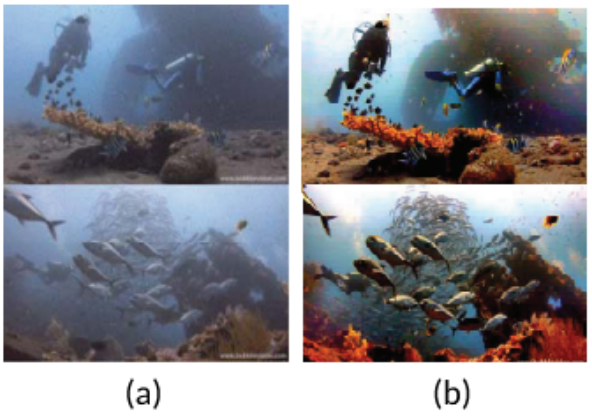
A seleção adequada do espaço de cores é fundamental para otimizar operações de segmentação e realce. O modelo RGB (*Red, Green and Blue*) representa as imagens através da combinação aditiva desses três canais, sendo amplamente utilizado em dispositivos de captura (Fu *et al.*, 2021; Archana; Jeevaraj, 2024)

Contudo, para tarefas que demandam invariância a variações luminosas, o espaço HSV (Matiz, Saturação, Valor) oferece vantagens significativas ao desacoplar a informação cromática (matiz) da intensidade luminosa (valor) e pureza da cor (saturação). Essa separação facilita a identificação robusta de objetos sob condições de iluminação heterogêneas, como ocorre em ambientes agrícolas (Manavalan, 2020).

Um grande número de problemas pode ocorrer em uma única imagem e há inúmeras maneiras de resolver o problema. As soluções encontradas nesta etapa são específicas para o problema em questão, portanto as estratégias utilizadas podem não ser adequadas para a resolução de outros tipos de problemas (Neethirajan, 2021; Zheng, 2023).

A Figura 5 ilustra o pré-processamento aplicado para ajuste de cor e contraste, destacando as diferenças visuais entre a imagem original e a imagem aprimorada.

Figura 5 – Exemplo de pré-processamento para ajuste de cor e contraste: (a) imagem original e (b) imagem após aprimoramento



Fonte: Adaptado de Wang *et al.*, 2017.

As estratégias finais utilizadas incluem redução de ruído, aprimoramento de bordas, correção de iluminação, etc. O realce de bordas é um grande aliado para definir o formato dos objetos em uma foto. A maioria dos métodos funciona identificando gradientes ou mudanças abruptas na intensidade dos *pixels* dentro de uma região de uma imagem (Neethirajan, 2021).

Entre as técnicas de realce morfológico, o detector de bordas de Canny destaca-se pela capacidade de identificar transições sutis de intensidade com baixa susceptibilidade a ruídos. Seu funcionamento envolve múltiplos estágios: suavização gaussiana, detecção de gradientes, supressão não máxima e limiarização por histerese, produzindo contornos finos e conectados. Essas propriedades são críticas para segmentar grãos de milho, onde bordas precisas delimitam regiões de interesse como furos de insetos ou áreas ardidas (Fu *et al.*, 2021).

Complementarmente, o envoltório convexo (*convex hull*) simplifica contornos irregulares ao gerar o menor polígono convexo que engloba todos os pontos de um objeto. Essa aproximação geométrica elimina concavidades indesejadas, facilitando a quantificação de áreas e formas em grãos fragmentados ou sobrepostos (Zheng, 2023).

O pré-processamento desempenha um papel fundamental na preparação das imagens para análises mais avançadas, pois melhora significativamente a qualidade dos dados fornecidos aos algoritmos de visão computacional (Tavares *et al.*, 2024). Ao minimizar distorções, eliminar ruídos e corrigir variações de iluminação, essa etapa garante que as características relevantes da imagem sejam preservadas e destacadas. Isso é especialmente importante em tarefas como segmentação, detecção de padrões e classificação, nas quais a precisão dos resultados depende diretamente da clareza e consistência dos dados de entrada (Eli-Chukwu, 2019).

O sucesso das etapas subsequentes do PDI depende diretamente da eficácia do pré-processamento. Investir tempo na escolha adequada das técnicas e na definição dos parâmetros dessa fase contribui para aumentar a robustez dos modelos computacionais, reduzindo erros e melhorando a capacidade de generalização dos sistemas de visão artificial. Isso se reflete em melhores resultados em aplicações diversas, como na agricultura, na medicina, na segurança e na indústria (Eli-Chukwu, 2019).

2.2.1.3 Segmentação

O objetivo da segmentação é dividir uma região da imagem em suas diversas partes constituintes, diferenciando-as entre si e extraindo atributos para análises e cálculos posteriores (Fu *et al.*, 2021). Essa etapa é considerada a mais importante do processamento, pois erros ou imprecisões na delimitação da região de interesse podem se propagar às etapas subsequentes, comprometendo a eficiência e a acurácia de todo o processo (Vo *et al.*, 2020).

A segmentação de imagens, um dos conceitos mais fundamentais em visão computacional, é amplamente utilizada para isolar regiões de interesse, como órgãos ou lesões. Em particular, métodos baseados em aprendizado profundo, como a arquitetura U-Net, são desenvolvidos especificamente para aplicações médicas, proporcionando alta precisão na delimitação dessas regiões (Tavares *et al.*, 2024).

A evolução das arquiteturas modernas em visão computacional trouxe novos paradigmas que integram teorias geométricas e modelos de aprendizado profundo, ampliando significativamente as aplicações em diagnóstico e reconstrução 3D. Essas inovações têm permitido enfrentar desafios antes considerados intratáveis, como a segmentação de tecidos em imagens de baixa resolução, resultando em avanços expressivos na precisão diagnóstica (Ward *et al.*, 2020).

Em operações de convolução, o preenchimento (*padding*) é uma técnica fundamental para manter as dimensões espaciais das imagens ao longo do processamento em redes neurais. Esse método consiste em adicionar bordas artificiais ao redor da imagem original, geralmente compostas por *pixels* com valor zero. Com isso, o *padding* impede a redução progressiva do tamanho dos mapas de características em arquiteturas profundas (Fu *et al.*, 2021).

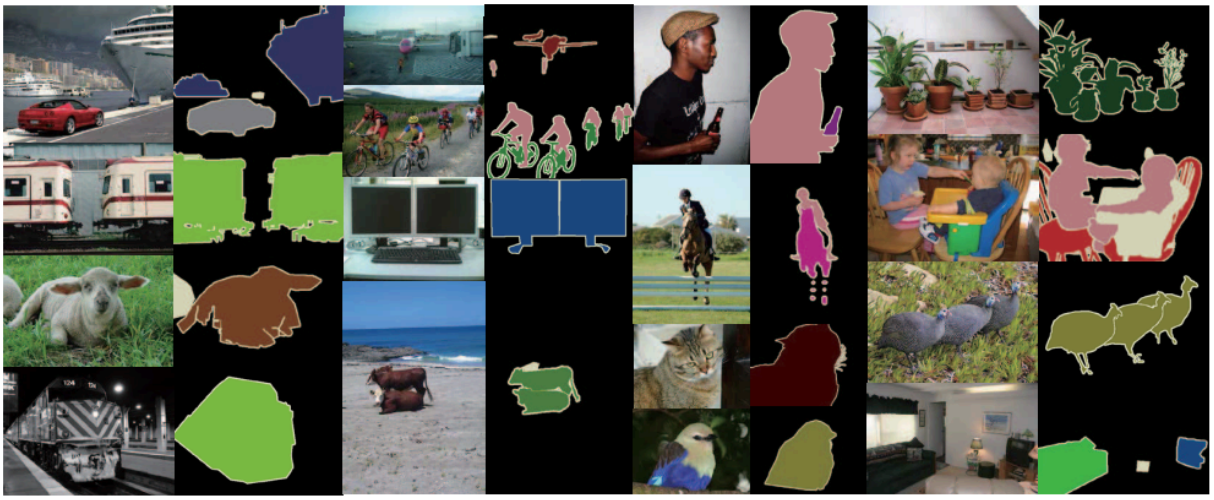
A preservação das dimensões da imagem desempenha um papel crucial em tarefas como segmentação, onde a manutenção da resolução é necessária para capturar detalhes finos, incluindo pequenos orifícios causados por insetos em grãos ou contornos irregulares de folhas. Além disso, ao assegurar que as regiões periféricas da imagem recebam o mesmo nível de processamento que as áreas centrais, o *padding* ajuda a preservar a integridade geométrica dos objetos segmentados (Zheng, 2023).

A segmentação é crucial porque atua como uma ponte entre a aquisição de imagens e a interpretação semântica dos dados visuais. Ao isolar com precisão as

regiões de interesse, ela viabiliza a extração de características significativas, como forma, textura e cor, que são essenciais para etapas posteriores, como classificação, detecção e reconhecimento de padrões (Fu *et al.*, 2021).

A Figura 6 apresenta o processo de seleção e delimitação de um subconjunto por segmentação em imagem colorida, destacando as diferenças entre a imagem original e a região delimitada pelo processamento.

Figura 6 – Exemplos de seleção e delimitação de subconjunto de segmentação



Fonte: Adaptado de Geng, Zhou e Cao, 2018.

Embora existam diversos métodos de segmentação, torna-se essencial adaptá-los ou conceber novas abordagens que levem em consideração as características específicas de cada tipo de imagem e atendam aos objetivos particulares da aplicação (Tavares *et al.*, 2024).

Os algoritmos aplicados nessas técnicas geralmente fundamentam-se em descontinuidades nos *pixels*, recorrendo a métodos como detecção de pontos, linhas e bordas por análise de similaridade entre *pixels*, aplicação de limiares de saliência, algoritmos de crescimento de regiões e máscaras convolucionais baseadas em gradiente (Pantazi; Moshou; Bochtis, 2020).

Em aplicações industriais, por exemplo, uma segmentação precisa pode automatizar processos de inspeção de qualidade; já na agricultura digital, permite a análise individual de folhas, frutos ou grãos com alta acurácia (Pantazi; Moshou; Bochtis, 2020).

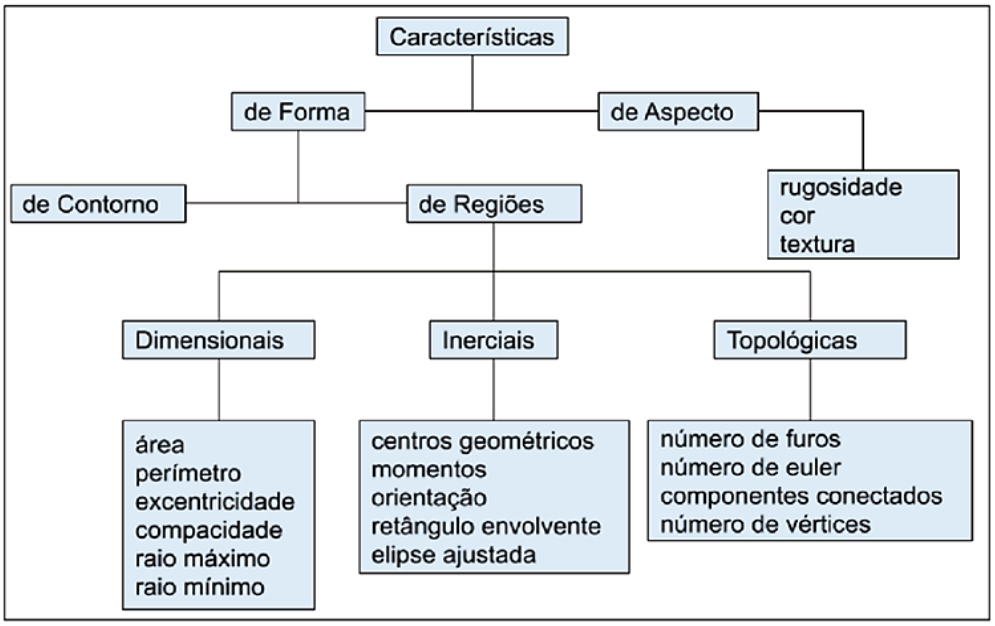
Quando bem aplicada, a segmentação reduz significativamente o volume de dados processados, direcionando os recursos computacionais apenas às áreas relevantes da imagem e, assim, otimizando o desempenho geral do sistema. Investir em técnicas robustas de segmentação é essencial para assegurar a precisão, a escalabilidade e a aplicabilidade dos sistemas visuais em contextos reais (Santos *et al.*, 2020).

2.2.1.4 Extração de Características

A extração de características representa a primeira etapa para interpretar uma imagem. Nesse processo, diferentes abordagens buscam extrair significado a partir de aspectos específicos da imagem, ampliando a compreensão de suas informações visuais (Marques Junior; Ulson, 2021).

De acordo com o diagrama da Figura 7, os recursos podem ser classificados em dois grandes ramos: forma e aspecto. O ramo relacionado à forma destaca linhas de contorno e recursos de regiões, os quais são subdivididos em propriedades dimensionais, inerciais e topológicas, cada uma com características específicas (Azevedo; Conci; Leta, 2022).

Figura 7 – Árvore de extração de características



Fonte: Adaptado de Azevedo, Conci e Leta, 2022.

As propriedades dimensionais incluem características como área, perímetro, excentricidade, compacidade, raio máximo e raio mínimo, que descrevem aspectos quantitativos da geometria da região. As propriedades inerciais, por sua vez, abrangem elementos relacionados à distribuição de massa e orientação, como centros geométricos, momentos, orientação, retângulo envolvente e elipse ajustada. Já as propriedades topológicas referem-se à conectividade e estrutura da região, englobando o número de furos, o número de Euler, componentes conectados e o número de vértices (Azevedo; Conci; Leta, 2022).

O ramo relacionado ao aspecto foca nas informações visuais do objeto, considerando características como rugosidade, cor e textura. Esses elementos fornecem uma análise complementar que, juntamente com as propriedades da forma, contribuem para uma descrição mais completa dos recursos avaliados (Azevedo; Conci; Leta, 2022).

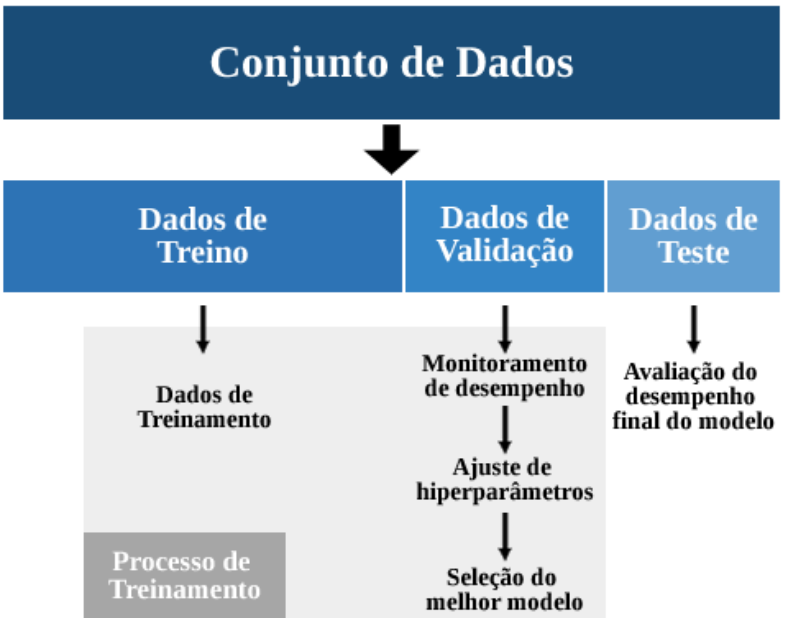
2.2.1.5 Métrica de Validação

Para avaliar o desempenho dos modelos de Inteligência Artificial (IA), é importante estabelecer métricas válidas. A escolha das métricas corretas está intimamente relacionada ao tipo específico de tarefa executada pelo modelo (Mariano, 2021).

Além disso, a escolha do método de validação pode depender da complexidade da tarefa e da disponibilidade dos dados. Os métodos comuns incluem validação cruzada, que divide o conjunto de dados em subconjuntos e avalia o modelo em diferentes conjuntos de treinamento e teste, e validação *holdout*, que reserva partes dos dados para teste para treinamento (Myllyaho *et al.*, 2021).

A Figura 8 ilustra a segmentação do conjunto de dados em três partes essenciais: treinamento, validação e teste. Os dados de treinamento são utilizados para o aprendizado dos padrões pelo modelo, enquanto o conjunto de validação permite acompanhar o desempenho durante o treinamento, ajustar hiperparâmetros e escolher a configuração mais eficaz. Por fim, os dados de teste são reservados exclusivamente para a avaliação final, fornecendo uma estimativa imparcial da capacidade de generalização do modelo. Essa divisão é fundamental para garantir uma modelagem mais robusta, confiável e eficaz (Yamashita *et al.*, 2018).

Figura 8 – Distribuição dos dados nos conjuntos de treinamento, validação e teste para a avaliação do modelo



Fonte: Adaptado de Yamashita *et al.*, 2018.

A aplicação de técnicas de validação apropriadas é fundamental para garantir que o modelo de inteligência artificial se generalize bem para novos dados, além de auxiliar na identificação de problemas como sobreajuste (*overfitting*) ou subajuste (*underfitting*). Dessa forma, o uso de estratégias de validação robustas torna-se essencial no processo de desenvolvimento e aprimoramento contínuo do desempenho dos modelos de IA (Myllyaho *et al.*, 2021).

Com base nessa estrutura de validação, o passo inicial na implementação prática consiste no treinamento dos modelos visando minimizar os erros de predição e evitar o sobreajuste. Para isso, a amostra de dados deve ser criteriosamente dividida, garantindo que o modelo seja treinado em um subconjunto e validado em outro, de modo a verificar sua capacidade de generalização. Esse procedimento permite estimar o desempenho real do modelo utilizando dados que não foram vistos durante o treinamento (Barros; Freitas Junior, 2023).

A adoção de técnicas de amostragem adequadas é essencial nesse processo, sendo comum a alocação de 80% dos dados para o treinamento e 20% para o teste. Tal abordagem contribui para uma avaliação mais precisa e imparcial, reduzindo o viés e fortalecendo a confiabilidade dos resultados obtidos (Barros; Freitas Junior, 2023).

2.2.1.6 Classificação

A classificação é a etapa final de um sistema de visão computacional e representa um dos principais objetivos da análise de imagens: interpretar e rotular automaticamente o conteúdo visual de forma análoga à percepção humana. Essa fase transforma os dados extraídos ao longo do processamento em informações quantitativas, possibilitando a identificação de padrões e categorias relevantes, fundamentais para a tomada de decisões em diversas aplicações (Chadebecq *et al.*, 2020).

O processo de classificação depende da construção de modelos, que são descrições quantitativas ou representações estruturais dos objetos presentes nas imagens, formadas a partir dos descritores extraídos nas etapas de pré-processamento e segmentação. Cada classe corresponde a um conjunto de objetos com características semelhantes, e o reconhecimento de padrões tem como objetivo associar corretamente um objeto observado à sua respectiva classe (Ward *et al.*, 2020). Em contextos como a agricultura de precisão, por exemplo, essa etapa permite identificar frutos, pragas ou anomalias em plantas com rapidez e elevada acurácia (Massruhá *et al.*, 2020).

Com o avanço do aprendizado de máquina e do aprendizado profundo, os sistemas de classificação tornaram-se mais eficientes e capazes de lidar com problemas complexos. Modelos baseados em CNNs são capazes de realizar classificações diretamente a partir dos dados brutos das imagens, eliminando a necessidade de extratores manuais de características (Zvarikova; Horak; Bradley, 2022)

Além disso, arquiteturas modernas, como o *PointRend*, foram desenvolvidas para otimizar a precisão da segmentação e, consequentemente, melhorar os resultados da classificação ao refinar os limites entre os objetos Dessa forma, a integração entre segmentação e classificação, aliada aos avanços da inteligência artificial, tem ampliado significativamente a aplicabilidade da visão computacional em áreas como saúde, segurança, indústria e agricultura (Santos *et al.*, 2020).

2.2.2 **Aplicações da Visão Computacional na Agricultura**

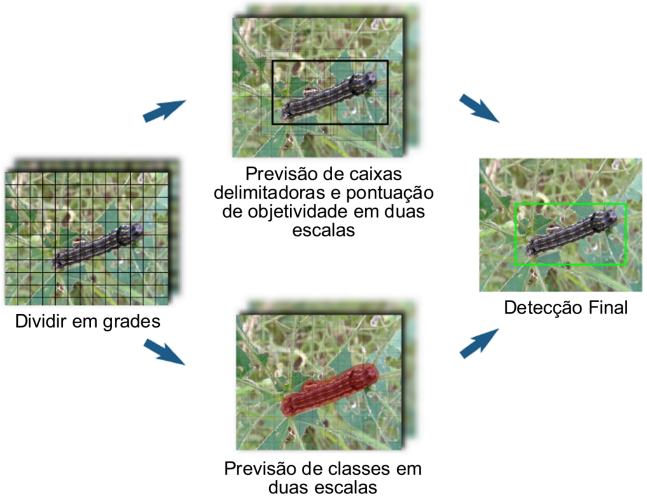
A visão computacional consolidou-se como um componente fundamental na agricultura moderna, proporcionando ganhos expressivos em eficiência operacional e sustentabilidade dos sistemas produtivos. Essa tecnologia possibilita desde o monitoramento contínuo das culturas até a automação inteligente de processos, como a colheita, favorecendo uma agricultura mais precisa e integrada (Da Silva; Santos, 2024).

Por meio do emprego de redes neurais convolucionais, tais aplicações viabilizam a classificação e o diagnóstico de grãos diretamente no campo, promovendo uma monitorização mais detalhada e embasada das lavouras (Carvalho; Rodrigues, 2023).

Algoritmos de análise de imagens digitais permitem avaliar o estado fitossanitário das culturas e detectar infestações com precisão, racionalizando o emprego de recursos hídricos e agroquímicos. Sistemas desenvolvidos para VANTs (Veículos Aéreos Não Tripulados) equipados com sensores imageadores realizam monitoramento sistemático, identificando sinais de estresse hídrico e surtos biológicos. A detecção precoce de patologias viabiliza intervenções direcionadas, minimizando impactos produtivos. Tais sistemas correlacionam simultaneamente parâmetros de solo e clima, otimizando estratégias de manejo agrícola (Souza *et al.*, 2024).

A visão computacional tem transformado práticas agrícolas através de monitoramento em alta resolução de parâmetros edáficos, regimes hídricos e ciclos fenológicos. Sensores ópticos especializados quantificam parâmetros edáficos em tempo real, permitindo ajustes hidráulicos dinâmicos, enquanto sistemas de imageamento hiperespectral detectam estágios fenológicos de frutificação, elevando a seletividade operacional e reduzindo perdas pós-colheita (Oliveira *et al.*, 2024). A Figura 9 apresenta um exemplo de aplicação de visão computacional para a detecção de pragas em lavouras.

Figura 9 – Exemplo de um processo de detecção de pragas em lavouras utilizando visão computacional



Fonte: Adaptado de Khalid *et al.*, 2023.

O processo da Figura 9 demonstra como a imagem de entrada é inicialmente submetida a uma camada de *pooling*, que reduz sua dimensionalidade, seguida por uma camada de convolução responsável pela extração de características relevantes. Em seguida, a imagem é dividida em células, para as quais são previstas coordenadas das caixas delimitadoras, índices de confiança e classes associadas em múltiplas escalas. Por fim, as detecções com menor confiabilidade são eliminadas, gerando o resultado final que destaca as pragas identificadas no campo (Khalid *et al.*, 2023).

A integração de VANTs e câmeras de alta resolução com algoritmos de processamento de imagem permite o monitoramento em larga escala de lavouras, detectando sinais precoces de estresse hídrico e variações na clorofila, o que possibilita ajustes pontuais no manejo e minimiza falhas decorrentes de inspeção manual (Souza *et al.*, 2024).

A integração com Sistemas de Informação Geográfica (SIG) otimiza a produtividade agrícola e a identificação precoce de infestações, viabilizando a aplicação localizada de defensivos. Além disso, algoritmos de aprendizado de máquina aplicados à colheita automatizada realizam classificação morfológica de cultivos mediante análise de padrões de cores e formas, promovendo estratégias sustentáveis e minimizando a dependência de operações manuais (Helfer *et al.*, 2020).

A tecnologia de visão computacional enfrenta desafios operacionais como variações ambientais, custos elevados e barreiras tecnológicas. Condições climáticas variáveis, como temperatura, umidade e luminosidade, comprometem a precisão das análises, enquanto a variedade das culturas dificulta a padronização de algoritmos utilizados para processamento e interpretação de imagens (Adewuyi *et al.*, 2024).

Embora os avanços sejam significativos, os custos permanecem como barreira crítica para agricultores de pequeno e médio porte, demandando infraestrutura especializada e capacitação técnica. Iniciativas colaborativas entre instituições são essenciais para democratizar o acesso e otimizar a aplicação dessas tecnologias no setor agrícola (Viana *et al.*, 2024). Contudo, a visão computacional viabiliza gestão operacional precisa mediante monitoramento contínuo das condições agronômicas, reduzindo o uso de defensivos químicos e melhorando a qualidade produtiva (Da Silva; Santos, 2024).

2.2.3 **Redes Neurais**

Redes Neurais Artificiais (RNAs), ou simplesmente Redes Neurais (RNs), constituem um conjunto de técnicas computacionais inspiradas no funcionamento do cérebro humano, em especial em sua notável capacidade de aprendizagem (Alomar; Aysel; Cai, 2023). A estrutura cerebral é composta por bilhões de neurônios interconectados por meio de sinapses. Quando um estímulo é recebido por um dendrito e sua voltagem supera o limiar necessário, ocorre a polarização do axônio, permitindo que o impulso elétrico seja transmitido de um neurônio a outro (Dong; Catbas, 2021).

As Redes Neurais Artificiais (RNAs) são compostas por um grande número de nós computacionais interconectados, chamados neurônios. Cada neurônio realiza a soma ponderada das entradas recebidas, e esse valor é processado por uma função de ativação (ou função de transferência), que determina a saída correspondente do neurônio. Esse mecanismo permite que a rede execute tarefas complexas de reconhecimento, classificação e predição (Marques Junior; Covolan, 2018; Yousef; Hussain; Mohammed, 2020).

As redes neurais artificiais (RNAs) são amplamente empregadas na classificação de eventos relacionados a distúrbios na qualidade da energia elétrica.

Esse tipo de rede é capaz de lidar com dados não lineares e apresenta robustez na análise de séries temporais, o que é fundamental para o monitoramento contínuo da qualidade da energia. Estudos recentes demonstram que as RNAs têm se mostrado eficazes na classificação automática de eventos, como variações de tensão e distorções harmônicas, com base em padrões extraídos por técnicas de análise de sinal. Essa capacidade de aprendizado contínuo permite que as redes se adaptem conforme novos dados são incorporados, aumentando a precisão na detecção de anomalias (Nicolau; Coelho, 2024).

O aprendizado profundo insere-se em um contexto mais amplo dentro da inteligência artificial. Em comparação com as abordagens tradicionais de aprendizado de máquina, o aprendizado profundo proporciona uma capacidade significativamente superior de extração de atributos relevantes dos dados, o que o torna particularmente eficaz em tarefas complexas (Saeed *et al.*, 2023).

Essa melhoria se dá por meio da construção de estruturas hierárquicas de extração de informação. Nesses modelos, os dados são processados em múltiplas camadas, cada uma responsável por capturar diferentes níveis de abstração. Essa arquitetura em camadas torna os modelos de aprendizado profundo altamente flexíveis e adaptáveis a uma ampla variedade de problemas e domínios (Saeed *et al.*, 2023).

No campo do aprendizado profundo, destacam-se as Redes Neurais Artificiais (RNAs), que são sistemas de processamento computacional inspirados no funcionamento dos sistemas nervosos biológicos, como o cérebro humano. Essas redes buscam reproduzir, de forma simplificada, a capacidade de aprendizado e processamento das redes neurais naturais (Baduge *et al.*, 2022).

Com o avanço da capacidade computacional, as redes neurais artificiais possibilitaram significativos progressos no processamento e na classificação de imagens. Esse avanço culminou no desenvolvimento das Redes Neurais Profundas (*Deep Neural Networks* – DNN) e, especialmente, das Redes Neurais Convolucionais (*Convolutional Neural Networks* – CNN), que se destacam por sua eficácia em tarefas complexas de visão computacional (Marques Junior; Covolan, 2018).

2.2.4 **Redes Neurais Convolucionais**

As CNNs constituem uma classe especializada de redes neurais artificiais, amplamente reconhecida por sua elevada eficácia em tarefas de visão computacional (Alomar; Aysel; Cai, 2023). Esse tipo de modelo, amplamente utilizado no contexto de aprendizagem profunda, é inspirado na estrutura do córtex visual humano, responsável pelo processamento de imagens, o que permite às CNNs extrair e aprender automaticamente características relevantes a partir de dados visuais (Zhou *et al.*, 2023a).

A estrutura de uma CNN é composta por diferentes tipos de camadas, sendo as principais: convolucionais, de *pooling* e totalmente conectadas. As camadas convolucionais aplicam filtros (ou *kernels*) que extraem características espaciais relevantes das imagens, como bordas, texturas e contornos, desempenhando papel fundamental na identificação de padrões visuais básicos (Voulodimos *et al.*, 2018).

As camadas de *pooling*, por sua vez, têm como função reduzir a dimensionalidade das representações, mantendo as informações mais significativas e contribuindo para a diminuição do número de parâmetros, o que melhora a eficiência computacional do modelo. Por fim, as camadas totalmente conectadas integram as informações extraídas nas etapas anteriores e realizam a predição final, transformando os dados em uma decisão classificatória (Li *et al.*, 2022).

As CNNs se destacam como ferramentas poderosas que podem ser usadas com sucesso para uma variedade de tarefas de visão computacional, desde classificação de imagens até detecção de objetos e segmentação de imagens. Vale ressaltar que essas redes neurais não são apenas ferramentas analíticas, mas também desempenham um papel importante no processo produtivo (Li *et al.*, 2022).

Esse tipo de rede neural apresenta a capacidade de aprender hierarquias de características, que vão desde padrões simples, como arestas, até representações complexas e abstratas. Essa propriedade as torna particularmente eficazes em tarefas como reconhecimento de objetos, detecção facial e segmentação de imagens. Um de seus principais diferenciais é a habilidade de extrair automaticamente atributos relevantes diretamente das imagens brutas, eliminando a necessidade de etapas manuais de engenharia de características (Alomar; Aysel; Cai, 2023).

A principal diferença entre as redes neurais tradicionais e as redes neurais convolucionais está na forma como processam os dados e extraem características. As redes tradicionais, também conhecidas como redes densamente conectadas ou do tipo *feedforward*, realizam conexões globais, nas quais cada neurônio de uma camada está interligado a todos os neurônios da camada seguinte, sem considerar explicitamente a estrutura espacial dos dados (Voulodimos *et al.*, 2018).

As redes tradicionais funcionam bem com dados organizados, mas não conseguem captar bem as relações espaciais em imagens. Já as Redes Neurais Convolucionais foram projetadas para processar dados visuais, usando camadas que aplicam filtros deslizantes sobre a imagem. Isso permite identificar padrões locais, como bordas e texturas, de forma eficiente (Voulodimos *et al.*, 2018).

Esse método permite detectar padrões em diferentes níveis (como formas simples e combinações complexas), tornando as CNNs mais eficientes e precisas em análises visuais. Além disso, sua estrutura reduz o número de parâmetros necessários, acelerando o treinamento e diminuindo o risco de sobreajuste em comparação com redes neurais comuns (He *et al.*, 2022).

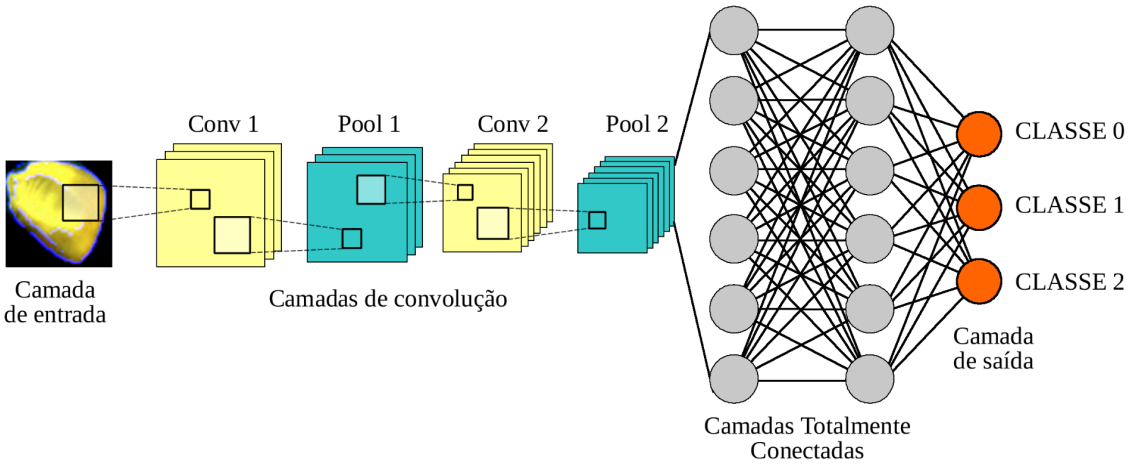
2.2.4.1 Arquitetura

A arquitetura das Redes Neurais Convolucionais foi criada para emular, de forma eficiente, o processo de percepção visual humana. Consideradas uma evolução do *Perceptron* Multicamadas, as CNNs distinguem-se principalmente pela disposição estruturada de suas conexões neuronais, inspirada na organização funcional do córtex visual de animais, o que permite a detecção progressiva e hierárquica de padrões visuais (Alomar; Aysel; Cai, 2023).

Nas Redes Neurais Convolucionais, cada neurônio processa informações de uma pequena região da imagem, chamada de área específica (campo receptivo). Esse funcionamento é semelhante ao sistema visual biológico, em que neurônios biológicos respondem apenas a estímulos em áreas delimitadas. Essas redes são formadas por três camadas principais: convolucionais (que identificam padrões), *pooling* (que simplificam os dados) e totalmente conectadas (que realizam a classificação final). Essa arquitetura possibilita à CNN aprender representações hierárquicas e eficazes, promovendo maior desempenho em diferentes níveis de abstração.(Alomar; Aysel; Cai, 2023).

A Figura 10 ilustra a arquitetura padrão de uma Rede Neural Convolucional, destacando a estrutura e a função de suas principais camadas.

Figura 10 – Arquitetura de uma Rede Neural Convolucional



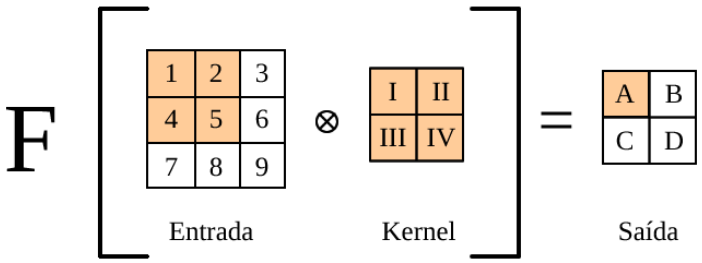
Fonte: Adaptado de Wang, Khalil e Firdi, 2022.

O processo da Figura 10 tem início na camada de entrada, que recebe a imagem a ser analisada, seguida por camadas convolucionais e de *pooling*, responsáveis por extrair e reduzir as características mais relevantes. Essas camadas são organizadas de forma repetitiva, permitindo a captação de padrões cada vez mais complexos. Em seguida, as informações processadas são encaminhadas para as camadas totalmente conectadas, que integram os atributos extraídos e conduzem à camada de saída, encarregada da classificação final (Wang; Khalil; Firdi, 2022).

As camadas convolucionais aplicam filtros (ou *kernels*) ajustados durante o treinamento para detectar padrões relevantes em imagens — como bordas, texturas e formas geométricas. Esses filtros geram mapas de ativação, também chamados de mapas de características, que ressaltam detalhes específicos e permitem à rede reconhecer e classificar objetos ou entidades distintas (Frota *et al.*, 2025).

Na Figura 11, observa-se um exemplo simplificado da operação de convolução, na qual uma matriz de entrada (representando uma imagem ou parte dela) é combinada com um *kernel*, resultando em uma matriz de saída que representa o mapa de características extraído (Gabriel Filho, 2023).

Figura 11 – Exemplo de aplicação de uma camada de convolução



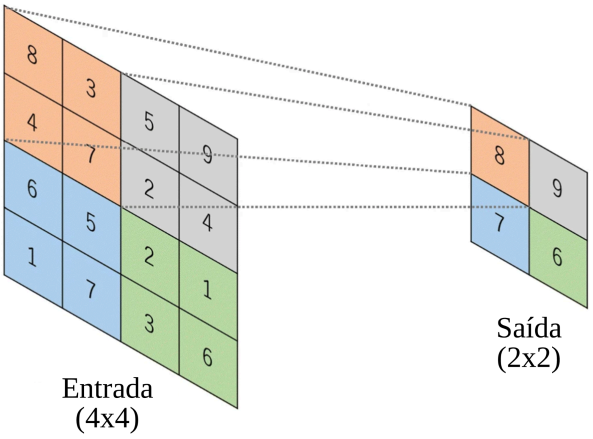
Fonte: Adaptado de Gabriel Filho, 2023.

A Figura 11 ilustra o processo que consiste em deslocar o *kernel* sobre a matriz de entrada, realizando multiplicações elemento a elemento seguidas de somas, produzindo valores agregados que compõem a saída. Cada elemento da matriz resultante (A, B, C, D) é obtido a partir da sobreposição do *kernel* em diferentes regiões da entrada, permitindo que padrões locais sejam capturados e representados de forma compacta. Essa operação é fundamental para que redes convolucionais detectem características espaciais relevantes e, assim, construam representações hierárquicas da informação visual (Gabriel Filho, 2023).

Para o processo de *pooling*, comumente realizado por meio de operações de *max pooling* (para sinalizar o maior valor) ou *average pooling* (para sinalizar o valor médio), é aplicado imediatamente após as camadas convolucionais com o objetivo de reduzir a dimensionalidade dos mapas de ativação, preservando, contudo, as características mais relevantes. Essa etapa fortalece a robustez do modelo ao conferir maior invariância a pequenas variações espaciais nas imagens, aspecto crucial para a extração eficaz de atributos em tarefas de classificação, como a identificação de grãos de milho (Frota *et al.*, 2025).

A Figura 12 mostra um exemplo simplificado de *max pooling* aplicado a uma matriz de entrada 4x4. Usando uma janela de análise 2x2, sem adição de bordas (*padding*) e com passo (*stride*) de 2, o método divide a matriz em blocos 2x2 sem sobreposição. Em cada bloco, apenas o maior valor é mantido, gerando uma matriz de saída 2x2. Essa redução mantém as características mais importantes, aumenta a tolerância a variações de posição e diminui a complexidade computacional das próximas etapas (Yamashita *et al.*, 2018).

Figura 12 – Exemplo de uma operação de *max pooling* em uma camada de *pooling*



Fonte: Adaptado de Yamashita *et al.*, 2018.

As camadas totalmente conectadas (*Fully Connected*), também conhecidas como camadas densas, são posicionadas após as etapas de convolução e *pooling* e têm a função de combinar as características extraídas para gerar as predições finais do modelo. Nessa etapa, cada neurônio se conecta a todos os neurônios da camada anterior, permitindo a síntese dos padrões locais, como bordas e texturas, em decisões globais, viabilizando tarefas de classificação ou regressão de forma precisa e eficiente (Azeem *et al.*, 2024).

Além disso, as camadas convolucionais incorporam funções de ativação aplicadas aos valores resultantes da operação de convolução, desempenhando um papel fundamental ao introduzir não linearidade ao modelo. Essa característica é indispensável para que a rede neural convolucional vá além de combinações lineares simples e consiga aprender padrões complexos presentes nos dados (Azeem *et al.*, 2024).

2.2.4.2 Funções de Ativação

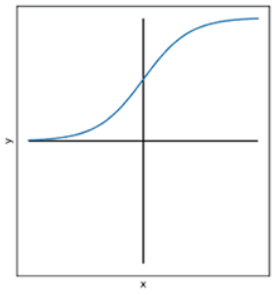
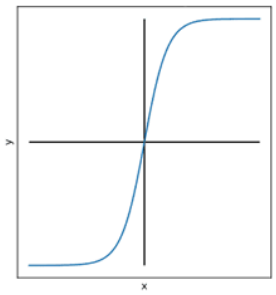
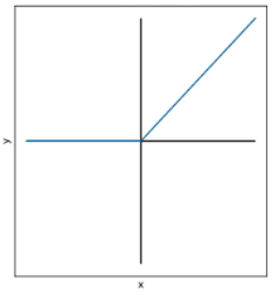
As funções de ativação exercem um papel essencial nas Redes Neurais Convolucionais, ao introduzirem não-linearidades que possibilitam ao modelo representar relações complexas e não triviais nos dados (Alomar; Aysel; Cai, 2023). Essas funções definem como cada neurônio transforma as informações recebidas em sua saída, influenciando diretamente a capacidade de aprendizagem da rede. Dentre as funções mais empregadas em CNNs, destaca-se a *Rectified Linear Unit*

(ReLU), amplamente utilizada por sua simplicidade computacional e elevada eficiência, além de contribuir para a mitigação do problema do desaparecimento do gradiente e facilitar o treinamento de redes profundas (Narin; Kaya; Pamuk, 2021).

Além da função ReLU, funções como Sigmoid e Tangente Hiperbólica (Tanh) também são comuns. A função Sigmoid limita suas saídas ao intervalo entre 0 e 1, sendo particularmente apropriada para camadas de saída em tarefas de classificação binária. A função Tanh produz saídas no intervalo de -1 a 1, o que pode ser vantajoso por promover a centralização dos dados em torno de zero, facilitando o aprendizado. Porém, ambas são menos usadas em camadas intermediárias, pois podem causar a extinção do gradiente em redes com muitas camadas, dificultando o treinamento. (Alomar; Aysel; Cai, 2023).

A ReLU é mais usada como função de ativação do que Sigmoid e Tangente Hiperbólica por não envolver cálculos exponenciais complexos, o que reduz o custo computacional e ajuda a evitar o desaparecimento do gradiente. Essa vantagem é ilustrada na Figura 13, que compara sua simplicidade de cálculo com outras funções (Yang; Yang, 2018).

Figura 13 – Funções de Ativação Sigmóide, Tangente Hiperbólica e ReLu

Sigmóide	Tangente Hiperbólica	ReLu
		
$\sigma(x) = \frac{1}{1+e^{-x}}$	$tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$	$ReLu(x) = max(0,x)$

Fonte: Adaptado de Yang; Yang, 2018.

Em aplicações de classificação multiclasse, a camada de saída costuma empregar a função Softmax para converter os valores brutos produzidos pelas etapas anteriores em uma distribuição de probabilidades normalizada, cujos valores

somam 100%, facilitando a interpretação das previsões como probabilidades de cada classe (Alomar; Aysel; Cai, 2023). A adequação da função de ativação escolhida impacta diretamente o desempenho do modelo e deve ser definida conforme as características dos dados e os objetivos da tarefa em análise (He *et al.*, 2022).

Além das funções de ativação, as CNNs incorporam camadas de normalização, como a *Batch Normalization*, que padronizam a distribuição dos dados entre as camadas, acelerando o treinamento e promovendo maior estabilidade do modelo. Complementarmente, técnicas como o *dropout*, que desativam aleatoriamente uma fração dos neurônios durante o aprendizado, contribuem para o aumento da robustez e a prevenção do *overfitting* (Partel; Kakarla; Ampatzidis, 2019).

A flexibilidade arquitetural das CNNs permite ajustar tanto o número quanto a combinação de camadas conforme os requisitos específicos de cada aplicação, adaptando-se, por exemplo, à classificação de grãos de milho com diferentes texturas e dimensões. Com os avanços contínuos em aprendizado profundo, novas variações de CNNs seguem sendo desenvolvidas, visando aprimorar a eficiência e a precisão na detecção de padrões em imagens (Partel; Kakarla; Ampatzidis, 2019).

2.2.4.3 Técnicas de Regularização e Otimização

As redes de aprendizado profundo fazem uso de uma ampla gama de técnicas de regularização e otimização, essenciais para o ajuste eficiente de seus parâmetros. O treinamento dessas redes, por sua natureza complexa, exige inevitavelmente a configuração de um grande número de parâmetros, além da experimentação com diferentes arquiteturas, a fim de enfrentar desafios como o sobreajuste (*overfitting*) e a lentidão na convergência do modelo (Jain; Rao; Sharma, 2020; WU; XIA; WANG, 2020).

Segundo Jain, Rao e Sharma (2020), esses dois mecanismos, regularização e otimização, funcionam como verdadeiros “motores cognitivos” no processo de aprendizado, exercendo papel decisivo na obtenção de um desempenho satisfatório. A otimização refere-se aos métodos e algoritmos utilizados durante o treinamento para ajustar os pesos da rede, de modo a aprender corretamente a relação entre os dados de entrada e saída. O objetivo é encontrar o conjunto ideal de parâmetros que

minimize o erro cometido sobre os dados de treinamento.

No entanto, a regularização tem como principal função impedir que o modelo se ajuste demasiadamente aos dados de treino, o que comprometeria sua capacidade de generalização. Isso é alcançado pela adição de penalidades ou restrições à função de perda, incorporando conhecimentos prévios sobre o problema ou favorecendo modelos com características desejáveis, como menor complexidade ou maior capacidade de generalização (Jain; Rao; Sharma, 2020; Xiao *et al.*, 2022).

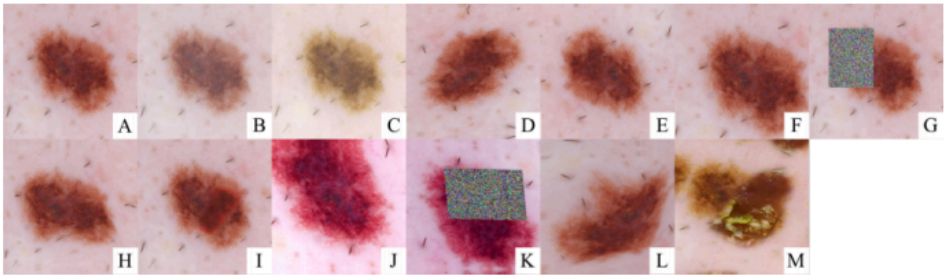
2.2.4.3.1. Data Augmentation

Uma das estratégias para combater o *overfitting* consiste em ampliar o conjunto de dados de treinamento. Contudo, a obtenção de novos dados úteis, aliada à etapa de rotulagem, pode ser uma tarefa trabalhosa, custosa e, em muitos casos, inviável na prática (Marin; Skelin; Grujic, 2020).

Nesse contexto, o *data augmentation* surge como uma técnica de regularização amplamente adotada, que tem como objetivo expandir artificialmente o conjunto de treinamento. Isso é feito por meio da aplicação de diversas transformações — como redimensionamento, escalonamento, recorte aleatório (*random cropping*), rotações, espelhamentos, ruído, deslocamento ou mudanças de brilho — sobre os dados existentes, gerando novas amostras sintéticas que enriquecem o aprendizado do modelo sem a necessidade de coleta adicional (Marin; Skelin; Grujic, 2020).

A Figura 14 ilustra exemplos práticos de *data augmentation* aplicados ao contexto médico, especificamente na análise de lesões cutâneas. Nessa figura, a imagem identificada como “A” corresponde à versão original, enquanto as demais resultam da aplicação de técnicas analíticas que simulam variações da imagem inicial (Perez; Vasconcelos, 2018).

Figura 14 – Exemplos de *data augmentation*



Fonte: Adaptado de Perez e Vasconcelos, 2018.

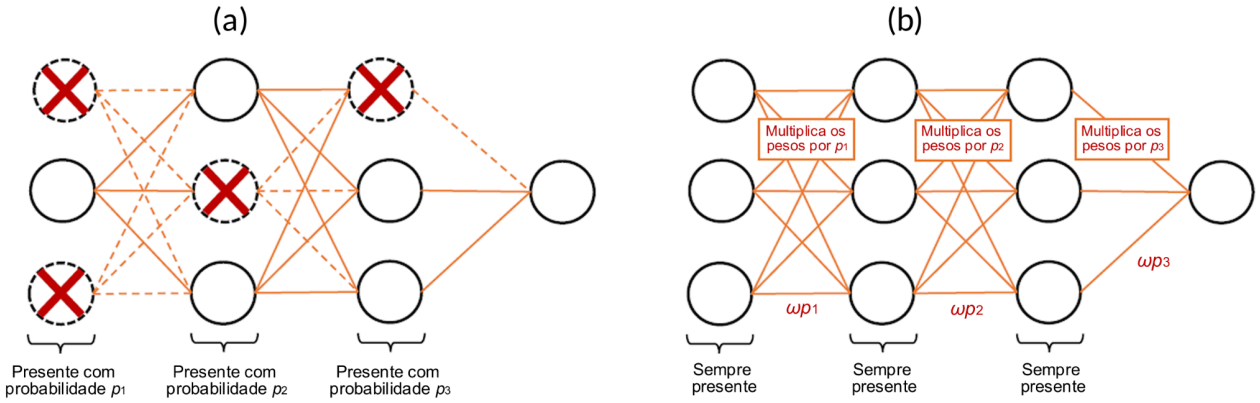
O *data augmentation* pode ser implementado previamente ao treinamento ou de forma dinâmica, durante o próprio processo de treinamento. Porém, ao trabalhar com dados rotulados, é fundamental adotar critérios cuidadosos na escolha das transformações aplicadas, a fim de evitar alterações que possam comprometer a integridade do rótulo original (Perez; Vasconcelos, 2018).

2.2.4.3.2. Dropout

Para reduzir o *overfitting*, ou sobreajuste, situação em que o modelo aprende relações excessivamente específicas dos dados de treinamento, uma técnica de regularização amplamente utilizada é o *dropout*. Durante o treinamento, essa abordagem consiste em desativar aleatoriamente alguns neurônios da rede em cada iteração, com base em uma probabilidade de descarte previamente definida. Essa taxa pode ser configurada individualmente para cada camada, permitindo níveis distintos de regularização ao longo da arquitetura da rede (Otter; Medina; Kalita, 2021).

Na fase de teste, todas as unidades permanecem ativas, mas seus pesos são ajustados multiplicando-se pela probabilidade de retenção p_{keep} (ou 1−p), compensando a ausência de desativações nessa etapa e mantendo a consistência da escala das ativações (Srivastava, 2014). A Figura 15 ilustra o comportamento do *dropout* nas fases de treino e teste.

Figura 15 – Exemplo do *dropout* nas etapas de (a) treino) e (b) teste



Fonte: Adaptado de Srivastava, 2014.

Do ponto de vista teórico, o *dropout* pode ser interpretado de duas formas complementares: como a adição de ruído de Bernoulli nas ativações, simulando perda estocástica de informações; ou como uma média aproximada das saídas de um grande número de sub-redes, formadas a partir de diferentes combinações de neurônios ativos. Essas sub-redes compartilham os mesmos pesos da rede original, permitindo que o modelo atue como um comitê de redes menores, promovendo generalização e robustez (Otter; Medina; Kalita, 2021).

2.2.4.3.3. Batch Normalization

Embora não seja, em essência, um algoritmo de otimização, a *Batch Normalization* (normalização por lote) tornou-se uma das técnicas mais relevantes no aprimoramento do treinamento de redes profundas. Sua principal contribuição está em estabilizar o processo de aprendizado, minimizando as variações indesejadas na distribuição das ativações internas, fenômeno causado pelas constantes atualizações dos parâmetros das camadas anteriores (Xiao *et al.*, 2022).

O método atua por meio da inserção de camadas de normalização ao longo da arquitetura da rede. Essas camadas ajustam dinamicamente a distribuição das entradas de cada camada durante o treinamento, promovendo estabilidade estatística e acelerando a convergência do modelo (Xiao *et al.*, 2022).

A entrada de uma determinada unidade é normalizada com base em cada mini-lote D. Seja z_i a entrada correspondente ao i-ésimo exemplo dentro de um mini-lote de tamanho m; a normalização de z_i é realizada conforme descrito na Equação 1, ajustada por meio dos parâmetros definidos nas Equações 2 e 3, que padronizam as entradas e as ajustam por meio de parâmetros treináveis, promovendo maior estabilidade e eficiência durante o treinamento do modelo (Yang *et al.*, 2020).

$$z_i^{(norm)} = \frac{z_i - \mu_D}{\sqrt{\sigma_D^2 + \varepsilon}}, \; i = 1, \ldots, m \tag{1}$$

$$\mu_D = \frac{1}{m} \sum_{i=1}^{m} z_i \tag{2}$$

$$\sigma_D^2 = \frac{1}{m} \sum_{i=1}^{m} (z_i - \mu_D)^2 \tag{3}$$

Nessa normalização, são utilizadas a média e a variância estimadas a partir do mini-lote D, enquanto ε representa uma pequena constante positiva adicionada para garantir a estabilidade numérica, evitando divisões por zero. Após essa padronização, aplica-se uma transformação linear adicional com parâmetros treináveis de escala e deslocamento, conforme Equação 4, com o objetivo de preservar o poder de representação das unidades ocultas e permitir que a rede aprenda a melhor forma de reescalonar as ativações (Yang et al., 2020).

$$\overline{z_i} = \gamma z_i^{(norm)} + \beta, \; i = 1, \ldots, m \tag{4}$$

Os novos parâmetros γ e β, também aprendidos durante o treinamento (inicializados com γ=1 e β=0) permitem que os dados normalizados sejam reescalonados para qualquer média e variância desejadas (Yang *et al.*, 2020). Já na fase de teste, em vez das estatísticas do mini-lote, utiliza-se uma média móvel exponencial das médias e variâncias estimadas durante o treinamento, garantindo consistência na normalização mesmo sem a presença de um lote de exemplos (Xiao *et al.*, 2022).

2.2.4.3.4. Otimizadores de Parâmetros

De acordo com Yang et al. (2020), a essência dos métodos iterativos clássicos de otimização reside em ajustar os parâmetros do modelo no sentido contrário ao gradiente $-\nabla_\theta J(\theta; D_t)$ direção em que a função custo decresce mais rapidamente. A iteração, os parâmetros são atualizados de acordo com a Equação 5.

$$\Delta\theta = -\eta\nabla_\theta J(\theta; D_t) \quad (5)$$

Nessa expressão, η (um hiperparâmetro estritamente positivo conhecido como taxa de aprendizagem) define a intensidade do ajuste realizado. Convenciona-se que θ_t representa o estado dos parâmetros na iteração t, sendo θ_0 a configuração inicial, que são tipicamente valores aleatórios de baixa magnitude extraídos de distribuições de probabilidade centradas em zero (como normal ou uniforme) (Yang et al., 2020).

Para o Gradiente Descendente Estocástico (*Stochastic Gradient Descent* – SGD), a cada iteração t, calcula-se uma aproximação do gradiente usando um mini-lote ß $\subset$ D_t (Yang *et al.*, 2020). Esse gradiente é então empregado para atualizar os parâmetros da iteração anterior t−1 seguindo a regra de atualização da Equação 6 .

$$\theta_t = \theta_{t-1} - \eta\nabla_\theta J(\theta_{t-1}; D_t) \quad (6)$$

O SGD é frequentemente empregado para designar a variante do algoritmo em que apenas um único exemplo de treinamento é utilizado para estimar o gradiente (Yang *et al.*, 2020). No entanto, quando essa estimativa é realizada com base em um mini-lote de exemplos, a abordagem é denominada *mini-batch gradient descent* (Zhou *et al.*, 2023b).

A escolha da taxa de aprendizado exerce influência decisiva na convergência do SGD. Taxas muito baixas tornam o treinamento excessivamente lento, enquanto taxas elevadas podem resultar em divergência dos parâmetros. Próximo de um ótimo local, o SGD tende a provocar oscilações nos valores dos parâmetros e pode apresentar lentidão ao atravessar regiões planas, típicas em torno de mínimos

locais, onde os gradientes se aproximam de zero (Zhou *et al.*, 2023b).

Essas limitações motivaram o desenvolvimento de algoritmos de otimização mais avançados, que incorporam mecanismos como o termo de *momentum*, visando acelerar a convergência e melhorar a estabilidade do treinamento (Zhang; Ramaswamy; Agarwal, 2020).

A adição de um *momentum* ao SGD contribui para acelerar o aprendizado nas direções mais relevantes, ao mesmo tempo em que amortece oscilações indesejadas durante o treinamento. Esse mecanismo atua desacelerando o avanço nas dimensões em que o gradiente apresenta instabilidade, ou seja, quando há variações frequentes no sinal do gradiente, favorecendo trajetórias de otimização mais suaves e eficientes (Zhang; Ramaswamy; Agarwal, 2020).

2.2.4.3.5. Otimizadores com Taxa de Aprendizado Adaptativa

Enquanto os algoritmos de otimização utilizam uma taxa de aprendizado única para atualizar todos os parâmetros do modelo, abordagens mais recentes propõem melhorias sobre o comportamento original do SGD (Zhang; Ramaswamy; Agarwal, 2020). Esses novos algoritmos introduzem mecanismos que permitem ajustar adaptativamente a taxa de aprendizado para cada parâmetro, tornando o processo de treinamento mais eficiente e sensível às características locais da função de perda (Zhang *et al.*, 2020).

O algoritmo Adagrad foi um marco ao introduzir uma estratégia de ajuste adaptativo da taxa de aprendizagem para cada parâmetro do modelo, baseada no acúmulo histórico dos gradientes ao quadrado. Essa abordagem favorece a atualização de parâmetros relacionados a atributos esparsos, ao mesmo tempo em que reduz gradualmente o passo de atualização para parâmetros frequentemente ativados. No entanto, seu principal limitador é o crescimento indefinido desse acumulador, o que pode causar uma queda prematura na taxa de aprendizagem, comprometendo a eficiência do treinamento em fases posteriores (Otter; Medina; Kalita, 2021).

Para contornar essa limitação, foi proposto o RMSProp, que substitui o acumulador fixo por uma média móvel exponencialmente ponderada dos gradientes ao quadrado. Isso permite dar mais relevância aos gradientes recentes, mantendo a adaptação da taxa de aprendizagem sem que ela se torne excessivamente pequena.

Como resultado, o RMSProp se mostra mais eficaz em cenários com dados esparsos ou funções de custo não estacionárias, oferecendo estabilidade e melhor capacidade de convergência ao longo do treinamento (Zhang *et al.*, 2020; Otter; Medina; Kalita, 2021).

Avançando essa ideia, o Adadelta elimina a necessidade de configurar manualmente a taxa de aprendizagem, ajustando-a dinamicamente com base nas atualizações recentes dos próprios parâmetros. Por meio de uma janela deslizante, ele calcula a magnitude média das mudanças anteriores, permitindo escalonamentos mais equilibrados ao longo do tempo. Essa característica torna o Adadelta especialmente útil em redes profundas e contextos com grande sensibilidade a hiperparâmetros, proporcionando um treinamento mais estável e adaptativo (Zhang *et al.*, 2020).

2.2.4.4 Funções de Perda (Loss Functions)

As funções de perda quantificam a discrepância entre as predições do modelo e os rótulos reais, servindo como bússola para orientar o processo de aprendizagem das redes neurais. No contexto de classificação multiclasse emprega-se frequentemente a entropia cruzada categórica (*categorical_crossentropy*) (Zhou *et al.*, 2023b). Esta função penaliza progressivamente desvios probabilísticos entre as previsões da camada softmax e a distribuição real dos rótulos, sendo particularmente eficaz quando as classes são mutuamente exclusivas (Mao; Mohri; Zhong, 2023).

Para problemas de regressão, o Erro Quadrático Médio (*Mean Squared Error*) destaca-se ao mensurar a média das diferenças quadráticas entre valores previstos e observados, favorecendo a convergência suave em espaços contínuos. A minimização da perda logística (sinônimo de *crossentropy* na prática) consolidou-se como substituta preferencial para tarefas de classificação, especialmente pela inviabilidade computacional da minimização direta da perda zero-um (Zhou *et al.*, 2023b).

Em redes neurais com saída softmax, essa função equivale à perda logística multinomial, que, sob condições ideais e com espaço de hipóteses suficientemente expressivo, minimiza eficazmente a perda de classificação. Contudo, em cenários práticos, os modelos operam sob restrições arquiteturais que limitam sua

capacidade de representação (Shaukat *et al.*, 2020; Zhou *et al.*, 2023b).

Nessas condições, não há garantias formais em regime não-assintótico sobre o desempenho da *crossentropy*, especialmente diante de minimizadores aproximados. Embora consolidada empiricamente, suas bases teóricas permanecem parcialmente circunscritas ao cenário assintótico idealizado (Jadon, 2020; Zhou *et al.*, 2023b).

Historicamente, funções alternativas para classificação multiclasse, como *sum-losses* e perdas com restrições, já possuíam garantias não-assintóticas estabelecidas. Curiosamente, tais fundamentos formais só recentemente foram estendidos à *crossentropy*, apesar de sua ubiquidade em aplicações de *deep learning*. Essa lacuna entre prática consolidada e teoria em desenvolvimento reforça a importância da seleção criteriosa de funções de perda alinhadas às particularidades do problema e arquitetura (Jadon, 2020; Pauli *et al.*, 2022).

2.2.5 Métricas de avaliação de classificação

Na avaliação de modelos de classificação de imagens de grãos de milho com Redes Neurais Convolucionais, adotam-se métricas estatísticas robustas e complementares — como acurácia, precisão, *recall* e *F1-score* — para mensurar tanto o desempenho global quanto o equilíbrio entre as previsões corretas por classe (Kearns; Roth, 2020).

Em cenários caracterizados por desequilíbrio na distribuição das classes, é comum o uso das médias macro, micro e ponderada das métricas de avaliação. Esses métodos visam proporcionar uma análise mais equitativa do desempenho do modelo, assegurando que as classes minoritárias não sejam negligenciadas em relação às majoritárias (Kearns; Roth, 2020).

Para uma análise mais detalhada do desempenho do modelo, a Matriz de Confusão oferece uma representação visual dos verdadeiros positivos, falsos positivos, verdadeiros negativos e falsos negativos. Essa ferramenta permite identificar padrões específicos de erro por classe, subsidiando ajustes mais precisos e direcionados na etapa de refinamento do modelo (Vilela Junior *et al.*, 2022).

2.2.5.1 Acurácia, precisão, *recall* e F1-score

Em contextos críticos, como a identificação de grãos de milho danificados ou doentes, erros em classes minoritárias podem acarretar prejuízos econômicos e operacionais. Assim, uma análise detalhada que avalie a relação entre as métricas e revele fragilidades do modelo em dados desbalanceados é essencial. Essa abordagem viabiliza ajustes precisos, alinhando o desempenho técnico às demandas práticas do setor agrícola (Chicco; Jurman, 2020).

Para quantificar o desempenho do modelo, as predições são categorizadas em quatro grupos: verdadeiros positivos (VP), que representam instâncias positivas corretamente identificadas; verdadeiros negativos (VN), referentes a instâncias negativas corretamente reconhecidas; falsos positivos (FP), que são casos negativos incorretamente classificados como positivos; e falsos negativos (FN), correspondentes a instâncias positivas que o modelo deixou de identificar. Essa categorização é fundamental para a interpretação precisa de métricas como acurácia, precisão, *recall* e *F1-score*, além de fornecer a base para a análise da matriz de confusão (Dinga *et al.*, 2019).

Por outro lado, a acurácia, definida como a proporção entre o número de predições corretas e o total de predições realizadas, oferece uma visão geral do desempenho do modelo. No entanto, sua interpretação pode ser enganosa em conjuntos de dados desbalanceados, pois a predominância de uma classe majoritária pode inflar artificialmente esse valor, ocultando deficiências significativas na classificação de classes minoritárias (Chicco; Jurman, 2020). A fórmula da acurácia é apresentada na Equação (7).

$$acurácia = \frac{VP + VN}{VP + FP + VN + FN} \tag{7}$$

Além da acurácia, métricas como precisão e *recall* são essenciais para uma avaliação detalhada. A precisão mede quantas das previsões positivas estão corretas, sendo crucial em situações em que erros ao classificar algo como positivo são problemáticos. Por exemplo, identificar erroneamente grãos de milho como saudáveis, quando não o são, pode levar a decisões inadequadas no manejo agrícola, afetando a produtividade (Manavalan, 2020). A fórmula da precisão é apresentada na Equação (8).

$$precisão = \frac{VP}{VP + FP} \quad (8)$$

O *recall*, também denominado sensibilidade ou taxa de detecção, quantifica a capacidade do modelo em identificar corretamente todas as instâncias realmente positivas. Essa métrica desempenha um papel crítico em aplicações nas quais a omissão de casos positivos pode acarretar consequências significativas, uma vez que um valor reduzido de *recall* indica a presença de falsos negativos, comprometendo diretamente a eficácia do sistema (Alkanan; Gulzar, 2023). A fórmula do *recall* é apresentada na Equação (9).

$$recall = \frac{VP}{VP + FN} \quad (9)$$

O *F1-score*, definido como a média harmônica entre precisão e *recall*, oferece uma métrica unificada que concilia a exatidão das previsões com a capacidade do modelo de identificar todas as instâncias relevantes, sendo particularmente útil quando se busca equilibrar falsos positivos e falsos negativos. Em aplicações agrícolas, como a classificação de grãos de milho, essa medida torna-se essencial, pois decisões equivocadas podem gerar prejuízos financeiros substanciais ao negligenciar exemplares comprometidos ou ao classificar erroneamente grãos saudáveis como defeituosos (Li *et al.*, 2019). A fórmula do *F1-score* é apresentada na Equação (10).

$$F1\text{-}score = \frac{2x\ Precisão\ x\ Recall}{Precisão + Recall} \quad (10)$$

A combinação de métricas oferece uma avaliação mais completa e realista do desempenho de Redes Neurais Convolucionais na classificação de imagens. Enquanto a acurácia fornece uma visão geral do acerto global, precisão e *recall* permitem diferenciar falsos positivos e falsos negativos, e o *F1-score* equilibra essas duas perspectivas em um único indicador. Essa abordagem integrada facilita a interpretação dos resultados e orienta ajustes específicos para o aprimoramento contínuo dos sistemas de classificação automática, assegurando maior confiabilidade prática (Kearns; Roth, 2020).

2.2.5.2 Matriz de confusão

Em algoritmos de inteligência artificial, frequentemente é exigido, por motivos de segurança, que todas as métricas utilizadas para avaliar sua eficiência apresentem resultados excelentes, uma vez que, em situações críticas, uma classificação incorreta pode gerar prejuízos significativos e, em cenários mais graves, colocar em risco a vida humana, de animais e de ecossistemas naturais (Vilela Junior *et al.*, 2022).

A maioria das métricas utilizadas para avaliar a qualidade de um classificador no contexto do aprendizado de máquina (*machine learning*) é derivada da matriz de confusão, também conhecida como tabela de contingência (Vilela Junior *et al.*, 2022). A matriz de confusão organiza as predições realizadas pelo modelo em quatro categorias principais: verdadeiros positivos, verdadeiros negativos, falsos positivos e falsos negativos. No caso de dados binários, em que há apenas duas classes possíveis, a estrutura da matriz torna-se bastante simples, como ilustrado no Quadro 1 (Mishra; Sachan; Rajpal, 2020).

Quadro 1 – Estrutura de uma Matriz de Confusão para modelos de classificação

	Classe Predita: Positiva	**Classe Predita: Negativa**
Classe Real: Positiva	Verdadeiro Positivo (VP)	Falso Negativo (FN)
Classe Real: Negativa	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Fonte: Adaptado de Mishra; Sachan e Rajpal, 2020.

No Quadro 1, cada célula da diagonal principal (VP e VN) representa os acertos por classe, enquanto as células fora da diagonal indicam erros de classificação, distinguindo falsos positivos (instâncias classificadas incorretamente como positivas) e falsos negativos (casos positivos não detectados pelo modelo) (Mishra; Sachan; Rajpal, 2020).

A análise da matriz de confusão exige atenção não apenas aos valores absolutos, mas também às proporções relativas nas linhas e colunas, permitindo identificar possíveis vieses do modelo e dificuldades na distinção entre classes específicas. Essa abordagem é fundamental para orientar ações corretivas, como o reequilíbrio dos dados ou o ajuste nos limiares de decisão, com o objetivo de aprimorar a performance do sistema (Vilela Junior *et al.*, 2022).

É importante destacar que, tanto no contexto real, em que diagnósticos são realizados por profissionais treinados, quanto no ambiente computacional, em que algoritmos inteligentes assumem essa função, as taxas de falsos positivos e falsos negativos podem ser significativas. Minimizar esses erros — ou idealmente reduzi-los a zero — é o cenário almejado, seja para um médico que busca maior precisão em seus diagnósticos, seja para um classificador automatizado que visa maior confiabilidade em suas predições (Vilela Junior *et al.*, 2022).

A Matriz de Confusão é ferramenta analítica que não se limita ao cálculo de métricas derivadas, como precisão e *recall*, mas também atua como um instrumento diagnóstico, ao evidenciar de forma direta os padrões de erro do modelo. A análise dos valores posicionados na diagonal principal da matriz indica a acurácia por classe, enquanto os elementos fora da diagonal revelam os tipos específicos de erro cometidos, permitindo identificar possíveis vieses do modelo ou dificuldades na distinção entre determinadas categorias (Mishra; Sachan; Rajpal, 2020).

Amplamente utilizada em tarefas de classificação supervisionada, a matriz de confusão deve ser interpretada à luz da distribuição das classes e dos custos associados a cada tipo de erro, servindo como base para ajustes estratégicos na arquitetura do modelo, na definição de limiares de decisão ou no balanceamento dos dados, com o objetivo de aprimorar o desempenho geral do sistema (Mishra; Sachan; Rajpal, 2020).

2.2.6 Redes neurais convolucionais para classificação de imagens

As CNNs consolidaram-se como ferramentas essenciais na classificação de imagens, graças à sua capacidade de extrair automaticamente características hierárquicas diretamente dos dados visuais. Estruturadas em múltiplas camadas que operam em estágios sucessivos, essas redes empregam filtros convolucionais para detectar padrões em diferentes escalas e contextos, permitindo uma representação eficaz de formas, texturas e cores (Li *et al.*, 2023).

A principal característica distintiva das CNNs é sua capacidade de aprender automaticamente representações relevantes a partir dos próprios dados de entrada, eliminando a necessidade de uma engenharia manual intensiva para extração de características. Essa automação é particularmente vantajosa em tarefas como a classificação de grãos de milho, nas quais as variações entre as classes podem ser

sutis e complexas, exigindo abordagens mais sofisticadas para garantir uma diferenciação precisa (Guimarães *et al.*, 2025).

O treinamento de uma CNN para tarefas de classificação de imagens baseia-se na apresentação de um conjunto de dados rotulados, utilizado para otimizar os pesos das conexões sinápticas da rede (Kattenborn *et al.*, 2021). Por meio do algoritmo de retropropagação do erro, a rede ajusta iterativamente seus parâmetros com o objetivo de minimizar a discrepância entre as previsões geradas e os rótulos reais. Após o treinamento, a CNN é capaz de generalizar o conhecimento adquirido para classificar corretamente novas imagens, proporcionando resultados rápidos e de alta precisão (Ludemir, 2021).

Além disso, a adoção de técnicas como regularização e aumento de dados (*data augmentation*) contribui significativamente para a prevenção do *overfitting*, um dos principais desafios enfrentados no desenvolvimento de modelos de *machine learning* (Ludemir, 2021). No contexto específico da classificação de imagens de grãos de milho, as CNNs demonstram notável capacidade de lidar com variáveis visuais complexas, como padrões de cor, textura e morfologia. A aplicação de diferentes configurações arquiteturais — incluindo camadas de *pooling*, *dropout* e normalização — permite aos pesquisadores aprimorar a robustez e a eficácia dos modelos, mesmo em cenários de alta variabilidade (Kattenborn *et al.*, 2021).

A evolução das redes neurais convolucionais (CNNs) é evidenciada pelos avanços em arquiteturas como AlexNet, VGG e ResNet, que têm apresentado desempenho superior em diversas competições e benchmarks de visão computacional. A aplicação dessas tecnologias transcende o setor agrícola, estendendo-se a áreas como segurança alimentar e controle de qualidade, o que evidencia o potencial transformador das CNNs na modernização das práticas agrícolas e na garantia de produtos mais seguros e padronizados (Kamilaris; Prenafeta-Boldú, 2018).

2.2.7 Arquiteturas Distribuídas para Sistemas de Visão Computacional

A crescente complexidade das aplicações modernas de visão computacional tem impulsionado a adoção de estratégias inovadoras de processamento distribuído,

capazes de integrar de forma sinérgica dispositivos heterogêneos em ecossistemas computacionais híbridos. Essas arquiteturas articulam inteligentemente recursos de borda (*edge computing*), nuvem (*cloud computing*) e dispositivos móveis, a fim de contornar restrições associadas à latência, à largura de banda e à limitação de processamento local. Essa abordagem viabiliza análises em tempo real e favorece respostas rápidas em ambientes dinâmicos (Liyanage *et al.*, 2021).

Nesse contexto, a integração entre aplicações móveis e as tecnologias de *edge* e *cloud computing* tem revolucionado a captura e o processamento de informações no âmbito da visão computacional aplicada à agricultura. A crescente necessidade de uma agricultura mais eficiente, sustentável e orientada por dados impõe a adoção de soluções que explorem, simultaneamente, a conectividade pervasiva dos dispositivos móveis e o poder computacional escalável dos servidores remotos, possibilitando diagnósticos precisos e intervenções ágeis no manejo das culturas (Liyanage *et al.*, 2021).

2.2.7.1 Aplicações Móveis e *Edge Computing*

O uso da visão computacional em dispositivos móveis impõe desafios associados a limitações de *hardware*, como capacidade restrita de processamento, memória e consumo energético. Tais restrições exigem otimizações algorítmicas específicas para viabilizar inferência com baixa latência e desempenho adequado em tempo quase real (Liu *et al.*, 2020).

Nesse contexto, destaca-se a importância do desenvolvimento de modelos compactos e da aplicação de técnicas de aceleração que integrem *hardware* e *software*, especialmente em aplicações críticas, como o monitoramento industrial e sistemas de segurança, que demandam respostas imediatas (Lu *et al.*, 2023).

A *edge computing* surge como uma solução complementar, ao deslocar parte do processamento para dispositivos próximos à fonte de geração dos dados. Essa descentralização permite a execução da inferência diretamente no ponto de captura, oferecendo vantagens como operações *offline* em cenários com conectividade limitada e redução substancial da latência, fatores essenciais para decisões em tempo crítico (Liu *et al.*, 2020; Lu *et al.*, 2023).

Dispositivos *edge* com aceleradores dedicados processam fluxos contínuos de imagem sem depender de servidores remotos, o que viabiliza respostas

instantâneas em ambientes como linhas de produção ou espaços públicos. Além disso, as aplicações móveis continuam exercendo um papel estratégico nos ecossistemas computacionais híbridos, aliando portabilidade e interfaces intuitivas a um processamento inicial eficiente (Lu *et al.*, 2023; Dai *et al.*, 2024).

No campo agrícola, por exemplo, soluções mobile permitem que produtores capturem imagens diretamente no local de cultivo, utilizando *smartphones* ou *tablets*. A análise dessas imagens com algoritmos de aprendizado de máquina possibilita a detecção precoce de doenças, pragas ou deficiências nutricionais, promovendo intervenções ágeis e direcionadas (Xu *et al.*, 2022).

Em áreas rurais com infraestrutura limitada, o uso da computação em borda garante decisões fundamentadas em tempo real, mesmo sem conexão contínua com a nuvem, consolidando uma arquitetura híbrida robusta que combina responsividade local e escalabilidade remota (Xu *et al.*, 2022).

2.2.7.2 *Cloud Computing* e Integração Híbrida

A computação em nuvem fornece uma infraestrutura escalável e de alto desempenho, essencial para o treinamento de modelos complexos de visão computacional. Utilizando recursos computacionais intensivos, como *clusters* com GPUs ou TPUs, a nuvem viabiliza operações exigentes sobre grandes volumes de dados heterogêneos (Hong *et al.*, 2019; Alouffi *et al.*, 2021).

Essa capacidade permite o refinamento contínuo de arquiteturas profundas aplicadas a conjuntos extensos de imagens e dados provenientes de sensores, promovendo ganhos significativos tanto na precisão quanto na capacidade de generalização dos sistemas preditivos (Alouffi *et al.*, 2021).

A comunicação entre o *frontend* (dispositivos de captura) e o *backend* (nuvem) é geralmente realizada por meio de APIs (*Application Programming Interfaces*, ou Interfaces de Programação de Aplicações) do tipo RESTful (*Representational State Transfer*, ou Transferência de Estado Representacional) ou gRPC (*Remote Procedure Call*, ou Chamada de Procedimento Remoto). Essas interfaces garantem interoperabilidade, segurança e transferência assíncrona dos dados para processamento em lote, alinhando eficiência e escalabilidade às

demandas crescentes de sistemas distribuídos (Hong *et al.*, 2019; Rapôso, 2025).

A sinergia entre *edge computing* e *cloud computing* forma ecossistemas computacionais híbridos e eficientes. Nessa arquitetura, os dispositivos de borda são responsáveis pela inferência local em tempo real, enquanto a nuvem atua na centralização e no retreinamento periódico dos modelos com base em novos dados adquiridos (Alouffi *et al.*, 2021).

Essa dinâmica garante que os parâmetros atualizados possam ser redistribuídos de forma seletiva para os dispositivos *edge*, conforme as exigências de cada cenário. Além de assegurar continuidade operacional sem interrupções, essa integração permite a análise de fluxos contínuos e o processamento de grandes *datasets* históricos, promovendo o aperfeiçoamento constante dos algoritmos (Kattenborn *et al.*, 2021).

Ao desempenhar também um papel central no armazenamento e integração de dados multiorigem, a nuvem potencializa o uso de técnicas avançadas de aprendizado de máquina, elevando o desempenho preditivo dos modelos aplicados à agricultura. Essa abordagem torna possível a atualização adaptativa dos modelos em tempo real, com base em dados coletados diretamente no campo (Hong *et al.*, 2019).

Como destacam Kattenborn *et al.* (2021), essa capacidade de atualização contínua fortalece a precisão das análises, enquanto Rapôso (2025) evidencia que tais avanços promovem não apenas a inovação no monitoramento agrícola, mas também contribuem diretamente para a sustentabilidade e o aumento da produtividade das culturas, consolidando a visão computacional como um elemento-chave na agricultura moderna.

2.2.8 Tecnologias para Aplicações Móveis

2.2.8.1 *Frameworks Frontend*

No desenvolvimento de *software*, o *frontend* corresponde à camada responsável pelos aspectos visuais e interativos apresentados ao usuário final. É tudo aquilo que o usuário vê e com o que interage, incluindo elementos gráficos, interfaces e parte da lógica que rege a experiência no site ou aplicativo (Novac *et al.*, 2024). Essa camada envolve o uso de linguagens como HTML (*HyperText Markup*

Language), CSS (*Cascading Style Sheets*) e JavaScript, além de abarcar todo o *software* e *hardware* relacionados à interface de usuário (Alessandria, 2020).

Tanto usuários humanos quanto sistemas digitais interagem diretamente com os componentes do *frontend*, como campos de entrada, botões e demais elementos da interface. Esses componentes são projetados para assegurar acessibilidade, usabilidade e uma experiência consistente. Além disso, é responsabilidade do desenvolvedor *frontend* criar interfaces compatíveis com as diversas plataformas, garantindo o funcionamento adequado em ambientes web e mobile (Karić; Durmić, 2024; Sangucho *et al.*, 2024).

No contexto do desenvolvimento *frontend* para sistemas móveis, o foco se volta para a criação de interfaces responsivas e eficientes, ajustadas às limitações computacionais dos dispositivos portáteis. Nesse cenário, o *Flutter* destaca-se como um SDK (*Software Development Kit*) de código aberto criado pelo Google, baseado na linguagem Dart, que permite desenvolver aplicativos multiplataforma para dispositivos móveis, *web* e *desktop* (Alessandria, 2020).

O *Flutter* possibilita desenvolvimento rápido, criação de interfaces acessíveis e amigáveis, além de fornecer código nativo para as diferentes plataformas. Seu fluxo de desenvolvimento organiza-se em torno de widgets, componentes visuais como botões, textos e imagens, que formam uma estrutura hierárquica na aplicação. Essa arquitetura facilita a criação de interfaces adaptáveis, consistentes e de alto desempenho (Oliveira; Lima; Caridade, 2022; Novac *et al.*, 2024).

Entre os *frameworks* que ampliam as capacidades do *Flutter*, o *MaterialApp* é fundamental por adotar os princípios do Material Design do Google, garantindo coerência visual e padronização de componentes. O *FlutterModular*, por sua vez, introduz recursos para gerenciamento de estado e navegação modular, facilitando a organização do código em módulos independentes e promovendo escalabilidade e manutenção simplificada, mesmo em dispositivos com recursos limitados (Sangucho *et al.*, 2024).

A integração entre *MaterialApp* e *FlutterModular* torna possível o desenvolvimento ágil de aplicações, aliando sofisticação visual e desempenho, elementos fundamentais nos atuais ecossistemas móveis (Karić; Durmić, 2024).

2.2.8.2 *Frameworks Backend* e Protocolos

Na área das aplicações móveis, a camada backend constitui a base para o processamento distribuído e a gestão de dados, demandando soluções leves e eficientes devido às limitações de recursos típicas desses ambientes. Nesse cenário, o *microframework* Flask se destaca por seu caráter minimalista e flexível, permitindo o desenvolvimento de serviços *web* robustos com baixa sobrecarga computacional (Zeng et al., 2025; Liu et al., 2023).

Essa arquitetura modular favorece a criação de APIs enxutas, capazes de operar até mesmo em dispositivos de borda com recursos restritos, como gateways IoT ou controladores industriais, nos quais a otimização do uso de memória e do poder de processamento é um requisito essencial (Costa; Barbosa; Cunha, 2023).

A comunicação entre o *frontend* e o *backend* em aplicações móveis ocorre, em geral, por meio do formato JSON (*JavaScript Object Notation*), cuja estrutura leve e legível facilita a serialização e o transporte eficiente de dados complexos. Essa escolha minimiza o *overhead* de transmissão e permite uma integração mais simples entre linguagens e plataformas distintas. Trata-se de um requisito crucial quando dispositivos móveis ou sistemas periféricos enviam dados a servidores Flask, que os processam e os disponibilizam para consumo em interfaces de usuário (Liu et al., 2023).

Para reforçar a segurança desse fluxo, o protocolo de autenticação JWT (*JSON Web Token*) introduz uma camada descentralizada de proteção, baseada em *tokens* assinados digitalmente, que dispensam o armazenamento de estado no servidor e garantem a validação de identidades mesmo em cenários de conectividade instável (Choma; Chwaleba; Dzieńkowsk, 2023).

A combinação entre Flask, JSON e JWT constitui uma base sólida e coerente para o desenvolvimento de aplicações móveis: o Flask fornece a infraestrutura leve de serviços web, o JSON padroniza a troca de informações e o JWT assegura a integridade e a autenticidade das transações. Esse arranjo possibilita soluções como o monitoramento em tempo real de ativos industriais e a coleta segura de métricas ambientais, sempre com foco na eficiência computacional, aspecto essencial em dispositivos com restrições de energia e recursos (Choma; Chwaleba; Dzieńkowsk, 2023).

2.2.8.3 Padrões de Projeto

Os padrões de projeto fornecem soluções arquitetônicas reutilizáveis para problemas recorrentes no desenvolvimento de aplicações móveis, contribuindo para a melhor organização do código e para o uso eficiente dos recursos em ambientes com restrições computacionais. Entre esses padrões, destaca-se o MVC (*Model-View-Controller*), que promove a separação da lógica da aplicação em três componentes interdependentes. O *Model* é responsável pelo gerenciamento dos dados e das regras de negócio, a *View* é encarregada da apresentação visual e da interface com o usuário, e o *Controller* atua como intermediário entre o *Model* e a *View*, coordenando suas interações (Mufid et al., 2019).

Essa divisão modular oferecida pelo padrão MVC é especialmente vantajosa em aplicações móveis de maior complexidade, pois permite que alterações na camada de interface sejam realizadas de forma isolada, sem impactar a lógica principal do sistema. Isso se mostra particularmente útil em cenários onde a aplicação precisa ser executada em diferentes dispositivos e versões do sistema operacional, nos quais a interface gráfica pode variar de acordo com o tipo de aparelho utilizado, enquanto o núcleo responsável pelo processamento e pelas regras de negócio permanece estável e consistente (Mufid et al., 2019).

Complementarmente, a injeção de dependência é uma técnica que promove o desacoplamento entre componentes, permitindo que objetos externos sejam fornecidos às classes durante sua inicialização, ao invés de serem criados internamente por elas mesmas. Essa abordagem facilita a substituição de módulos durante fases de teste ou adaptações para diferentes *hardwares*, favorecendo a criação de sistemas mais flexíveis, modulares e menos suscetíveis a falhas (Paolone *et al.*, 2020; Ahmad; Rana; Maqbool, 2022).

Paralelamente, o padrão Singleton assegura que uma determinada classe possua apenas uma única instância global durante toda a execução do aplicativo, oferecendo um ponto de acesso unificado a recursos compartilhados, como *pools* de conexões de rede ou configurações globais. Essa abordagem é especialmente importante em dispositivos móveis, que frequentemente enfrentam restrições severas de memória, pois evita a duplicação desnecessária de objetos e possibilita um controle centralizado e eficiente sobre serviços críticos do aplicativo (Paolone et al., 2020).

A combinação desses padrões fortalece a robustez dos aplicativos móveis: enquanto o MVC organiza a estrutura geral da aplicação, o Singleton gerencia

recursos críticos de forma eficiente, e a injeção de dependência proporciona flexibilidade na integração dos subsistemas (Ahmad; Rana; Maqbool, 2022).

Em dispositivos industriais de Internet das Coisas (IoT), por exemplo, o padrão MVC pode estruturar o fluxo de dados dos sensores, a apresentação visual nos painéis e a lógica de controle. O padrão Singleton gerencia o acesso concorrente ao barramento *Controller Area Network* (CAN), enquanto a injeção de dependência facilita a troca entre protocolos de comunicação, como Message *Queuing Telemetry Transport* (MQTT) e *Constrained Application Protocol* (CoAP), de acordo com a disponibilidade da rede (Ahmad; Rana; Maqbool, 2022).

3 TRABALHOS RELACIONADOS

3.1 Trabalho de Rosin (2019b)

O trabalho desenvolvido por Rosin (2019b), intitulado *"Visão Computacional Aplicada na Classificação de Grãos de Milho"*, explorou abordagens tradicionais de aprendizado de máquina para realizar a classificação de grãos de milho em três categorias distintas: ardidos (classe 0), carunchados (classe 1) e sadios (classe 2).

Para tanto, a autora utilizou os algoritmos *Random Forest*, K-*Nearest Neighbors* (KNN) e *Support Vector Machine* (SVM), combinando-os com técnicas de extração manual de características, com destaque para a Matriz de Coocorrência de Níveis de Cinza (GLCM) e a Análise de Componentes Principais (PCA).

A Figura 16 apresenta um exemplo das imagens capturadas por Rosin (2019b), utilizadas na construção do banco de dados que originou o *dataset* posteriormente publicado por Rosin (2019a) no repositório GitLab, o qual foi empregado no presente trabalho.

Figura 16 – Imagem capturada por Rosin (2019b) para a composição do *dataset*

Fonte: Adaptado de Rosin, 2019b.

Apesar de alcançar uma acurácia global de 83,0%, os resultados obtidos revelaram limitações significativas, especialmente na identificação da classe

“carunchado”. Para essa categoria, todas as demais métricas de desempenho (precisão, *recall* e *F1-score*) apresentaram valor nulo, evidenciando a total ausência de acertos por parte dos modelos utilizados.

Essa deficiência pode ser atribuída à dependência de atributos manuais, como as texturas extraídas via GLCM, que não foram suficientemente discriminativas para diferenciar os grãos carunchados das demais classes, além da provável influência do desbalanceamento entre as categorias presentes no conjunto de dados.

Por outro lado, os modelos demonstraram desempenho mais consistente nas classes “ardido” e “sadio”, cujos *F1-scores* oscilaram entre 76% e 78%, e 89% e 90%, respectivamente. Ainda assim, tais resultados ficaram aquém da acurácia observada em análises visuais conduzidas por classificadores humanos experientes. O Quadro 2 apresenta um resumo consolidado dos principais resultados reportados por Rosin (2019b).

Quadro 2 - Resumo dos resultados obtidos por Rosin (2019b)

Método	Extração de Características	Classe	Acurácia	Precisão	*Recall*	*F1-score*
RANDOM FOREST	GLCM	0	83,0%	83,0%	74,0%	78,0%
		1		0,0%	0,0%	0,0%
		2		83,0%	97,0%	90,0%
KNN	GLCM + PCA	0	83,0%	94,0%	64,0%	76,0%
		1		0,0%	0,0%	0,0%
		2		81,0%	99,0%	89,0%
SVM	GLCM + PCA	0	83,0%	80,0%	74,0%	77,0%
		1		0,0%	0,0%	0,0%
		2		84,0%	97,0%	90,0%

Fonte: Adaptado de Rosin, 2019b.

3.2 Trabalho de Wu *et al.* (2018)

O estudo de Wu *et al.* (2018), intitulado "*Classification of corn kernels grades using image analysis and support vector machine*", propôs um sistema de classificação de grãos de milho utilizando Máquinas de Vetores de Suporte (SVM) e Redes Neurais do tipo *Back-Propagation* (BP).

A abordagem foi desenvolvida com base na norma chinesa GB/T 17890-2008, que categoriza os grãos em três níveis de qualidade: Grau A, composto por grãos de alta qualidade, com diâmetro superior a 7,2 mm, baixa umidade e ausência de defeitos; Grau B, que inclui grãos com diâmetro intermediário e até 2% de impurezas; e Grau C, formado por grãos com diâmetro inferior a 6,8 mm e alto índice de defeitos.

Em termos de desempenho, Wu *et al.* (2018) reportaram uma acurácia global de 97,44% na classificação de grãos em três categorias comerciais, utilizando algoritmos genéticos (GA) e *Particle Swarm Optimization* (PSO) para otimizar os hiperparâmetros do SVM, a partir de um conjunto amostral relativamente reduzido (129 amostras). Os principais aspectos metodológicos, contribuições e resultados do estudo conduzido por Wu *et al.* (2018) encontram-se no Quadro 3, apresentado abaixo.

Quadro 3 - Resumo da metodologia e resultados de Wu *et al.* (2018)

Classificação	3 classes: Grau A, B, C (baseado em diâmetro, umidade e impurezas)
Critério de Defeito	Atributos físico-composicionais (tamanho, umidade, impurezas)
Tamanho do *Dataset*	129 amostras (43 por classe)
Método de Classificação	SVM com otimização (GA/PSO/GS) e Redes Neurais BP
Acurácia Global	97,44% (SVM-GA/SVM-PSO) 94,87% (SVM-GS) 92,31% (SVM/BP sem otimização)
Extração de Características	Manual (área, circularidade, complexidade geométrica)
Otimização de Parâmetros	GA, PSO e GS para ajuste de hiperparâmetros de SVM

Fonte: Adaptado de Wu *et al.*, 2018.

3.3 Trabalho de Velesaca *et al.* (2020)

O estudo de Velesaca *et al.* (2020), intitulado *"Deep Learning based Corn Kernel Classification"*, propôs dois modelos distintos para a classificação de grãos de milho com base em redes neurais profundas. O primeiro modelo utilizou uma abordagem binária, diferenciando os grãos entre defeituosos — agrupando ardidos, quebrados e impurezas — e sadios. Já o segundo modelo adotou uma classificação mais específica, distribuindo os grãos em três categorias: defeituosos, sadios e impurezas, esta última tratada como uma classe independente.

Ambos os modelos foram implementados por meio de um *pipeline* que integrava segmentação e classificação, utilizando a arquitetura *Mask* R-CNN para detectar e segmentar individualmente cada grão, e a CK-CNN (*Corn Kernel Convolutional Neural Network*), uma rede convolucional especializada desenvolvida pelos autores para a tarefa de reconhecimento e categorização dos grãos.

Conforme apresentado no Quadro 4, os resultados indicaram um desempenho superior do modelo triclasse em relação ao modelo binário. A abordagem com três categorias alcançou uma acurácia global de 95,60%, superando os 94,50% obtidos pelo modelo binário. Na avaliação por classe, o modelo triclasse obteve precisão de 97,90% para grãos sadios, 97,30% para impurezas e 90,00% para grãos defeituosos.

Quadro 4 - Resumo da metodologia e resultados de Velesaca *et al.* (2020)

Categoria	**Modelo com 2 classes**	**Modelo com 3 classes**
Classes	Defeituosos e Sadios	Defeituosos, Sadios e Impurezas
Arquitetura	*Mask* R-CNN (segmentação) + CK-CNN (classificação)	*Mask* R-CNN (segmentação) + CK-CNN (classificação)
Acurácia Global	94,50%	95,60%
Acurácia por Classe	Defeituosos: 93,30% Sadios: 95,60%	Defeituosos: 90,00% Sadios: 97,90% Impurezas: 97,30%
Tamanho do *Dataset*	3300 imagens	4500 imagens

Fonte: Adaptado de Velesaca *et al.*, 2020.

Apesar de o modelo binário ter atingido uma acurácia ligeiramente maior para a classe de grãos defeituosos (93,30%), sua capacidade de discriminação foi comprometida pela agregação de categorias distintas (como ardidos, quebrados e impurezas) sob uma mesma classificação, reduzindo a especificidade da análise.

3.4 Trabalho de Suárez *et al.* (2023)

No estudo de Suárez *et al.* (2023), intitulado *"Corn kernel classification from few training samples"*, foram desenvolvidos dois modelos distintos para a classificação de grãos de milho em contato físico. O primeiro adotou uma abordagem binária, distinguindo grãos em duas categorias: sadios e defeituosos. Já o segundo modelo utilizou uma abordagem multiclasse, segmentando os grãos em três categorias: sadios, defeituosos e impurezas como classe independente.

Ambas as abordagens foram implementadas por meio de um *pipeline* que integrou segmentação baseada na arquitetura *Mask* R-CNN (treinada com dados sintéticos para evitar a necessidade de anotações manuais) e classificação realizada pela CK-CNNLW (*Corn Kernel Convolutional Neural Network LightWeight*), uma rede convolucional ultra leve desenvolvida com foco em alta eficiência computacional.

Conforme apresentado no Quadro 5, o modelo triclasse demonstrou desempenho global ligeiramente superior, alcançando uma acurácia de 96,70%, em comparação aos 96,50% obtidos pelo modelo binário. Na avaliação por categoria, o modelo de três classes obteve 99,00% de acurácia para grãos sadios, 98,60% para impurezas e 93,00% para grãos defeituosos. Já o modelo binário apresentou 98,30% de acurácia para a classe defeituosa e 98,00% para grãos sadios.

Um destaque relevante do estudo foi a eficiência da arquitetura CK-CNNLW, que conta com apenas 548 mil parâmetros, tornando-se aproximadamente seis vezes mais leve que versões anteriores desenvolvidas pelo mesmo grupo e cerca de 40 vezes mais compacta que modelos convencionais como a VGG16, sem comprometer o desempenho classificatório.

Quadro 5 - Resumo da metodologia e resultados de Suárez *et al.* (2023)

Categoria	**Modelo com 2 classes**	**Modelo com 3 classes**
Classes	Defeituosos e Sadios	Defeituosos, Sadios e Impurezas
Arquitetura	CK-CNNLW	CK-CNNLW
Acurácia Global	96,50%	96,70%
Acurácia por Classe	Defeituosos: 98,30% Sadios: 98,00%	Defeituosos: 93,00% Sadios: 99,00% Impurezas: 98,60%
Tamanho do *Dataset*	3710 imagens	10660 imagens

Fonte: Adaptado de Suárez *et al.*, 2023.

4 MATERIAIS E MÉTODOS

Para alcançar o objetivo de desenvolver um aplicativo móvel capaz de classificar amostras de grãos de milho em três classes distintas, por meio de um modelo de visão computacional baseado em redes neurais convolucionais, foi necessário estruturar o trabalho em duas etapas principais.

Na primeira delas, iniciou-se pela aquisição das imagens, seguida pelo pré-processamento e pela divisão dos dados nos conjuntos de treino, validação e teste. Em seguida, procedeu-se ao treinamento do modelo, ao ajuste dos hiperparâmetros da rede neural e, por fim, à coleta e à análise dos resultados obtidos.

Na segunda etapa, com o modelo devidamente treinado, realizou-se o desenvolvimento do aplicativo móvel destinado à captura e ao envio das imagens das amostras para processamento em nuvem. Nesse ambiente, as imagens são submetidas à segmentação e ao pré-processamento, seguidos pela predição do modelo, pelo cálculo da área correspondente a cada classe de grão e, finalmente, pela classificação das amostras de acordo com o tipo identificado.

4.1 AQUISIÇÃO DO BANCO DE IMAGENS

Para a composição do banco de imagens utilizado neste trabalho, foi empregado o *dataset* disponibilizado por Rosin (2019a), acessível no repositório GitLab. Esse conjunto de dados é composto por um total de 4.339 (quatro mil, trezentas e trinta e nove) imagens de grãos de milho, obtidas a partir de 9 (nove) espigas distintas. Cada imagem representa um único grão e possui três canais de cor no formato RGB, com dimensões fixas de 48 × 48 *pixels*.

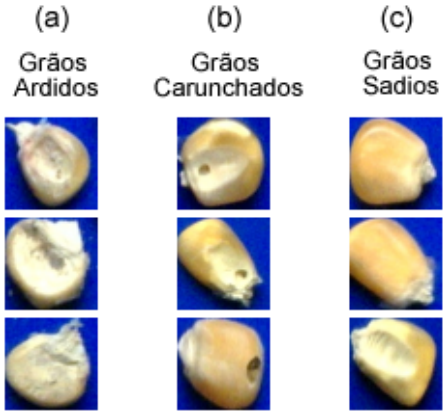
As imagens foram classificadas em três categorias: ardido, carunchado e sadio. Grãos ardidos apresentam um aspecto deteriorado devido à umidade ou condições adversas; grãos carunchados correspondem àqueles atacados por insetos, apresentando danos visíveis; e grãos sadios não apresentam defeitos, estando em perfeitas condições.

Segundo Rosin (2019b) as imagens foram capturadas utilizando equipamentos de alta resolução para garantir clareza e precisão na identificação das características dos grãos. Cada imagem foi cuidadosamente analisada e classificada

em uma das três categorias mencionadas, assegurando a consistência e a precisão dos dados. Para facilitar a manipulação e o processamento subsequente, as imagens foram organizadas em pastas separadas de acordo com sua classificação.

A Figura 17 ilustra exemplos representativos das três categorias presentes no banco de dados de Rosin (2019a): grãos ardidos, carunchados e sadios.

Figura 17 – Exemplos de grãos de acordo com a classe: (a) ardidos, (b) carunchados e (c) sadios



Fonte: Elaboração do autor.

Este banco de imagens é fundamental para a realização de estudos sobre a qualidade dos grãos de milho, permitindo a análise de diferentes defeitos e a comparação com grãos em perfeitas condições. A classificação precisa das imagens é essencial para o desenvolvimento de algoritmos de detecção e classificação automática de defeitos em grãos de milho. Todo o processo de aquisição e utilização das imagens foi realizado em conformidade com as normas éticas e de consentimento, respeitando os direitos autorais e a propriedade intelectual da autora do *dataset*.

Com base nesse banco foi desenvolvido o código capaz de gerar o modelo de predição para a classificação dos grãos. Esse modelo foi treinado e testado pelo banco de imagens de maneira supervisionada para aprimorar sua capacidade de discernir e categorizar os diferentes tipos de grãos.

Ao realizar análise dessas imagens, constatou-se que alguns grãos não estavam corretamente classificados, além de que em alguns grãos apresentava uma incerteza em sua identificação exata. Para solucionar esses pontos, foi realizada

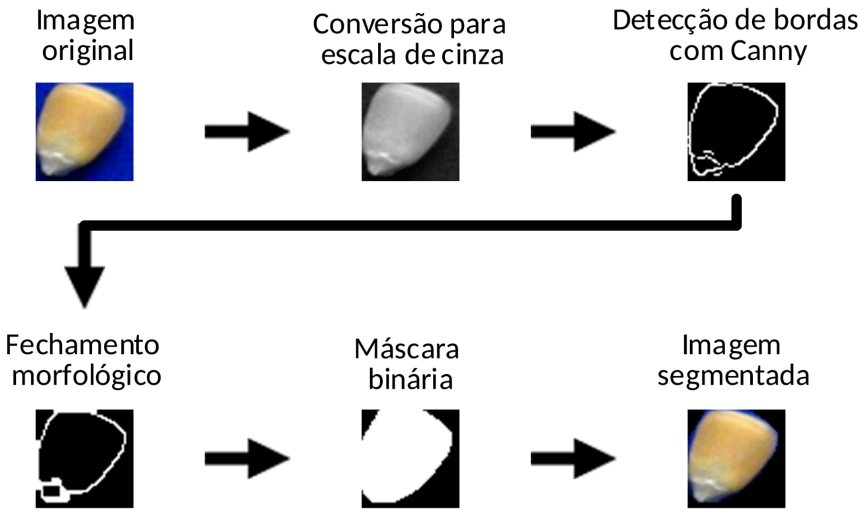
uma limpeza e reclassificação dos grãos, de modo que ao final restaram 87 imagens na categoria carunchados, 575 da categoria ardidos e 2918 na categoria sadios, totalizando em 3580 imagens.

4.2 PRÉ-PROCESSAMENTO DAS IMAGENS

Para o treinamento do modelo, as imagens de grãos de milho foram submetidas a um pré-processamento sequencial composto por duas etapas fundamentais. A primeira etapa concentrou-se no isolamento preciso do grão em relação ao fundo da imagem. Inicialmente, a imagem original foi convertida para escala de cinza, e em seguida aplicou-se o algoritmo de detecção de bordas de Canny, com o objetivo de identificar os contornos externos do grão.

Em seguida, foi realizado um fechamento morfológico, o que possibilitou a consolidação das bordas detectadas e a identificação do maior contorno externo, correspondente ao grão completo. Esse contorno foi então aproximado por meio de seu envoltório convexo (*convex hull*), gerando-se uma máscara binária que preserva exclusivamente a região do grão, enquanto suprime completamente o fundo por meio do preenchimento com *pixels* pretos. O resultado é uma imagem segmentada, na qual apenas o grão permanece visível. A Figura 18 ilustra detalhadamente as etapas envolvidas neste primeiro estágio do pré-processamento.

Figura 18 – Primeira etapa do pré-processamento das imagens



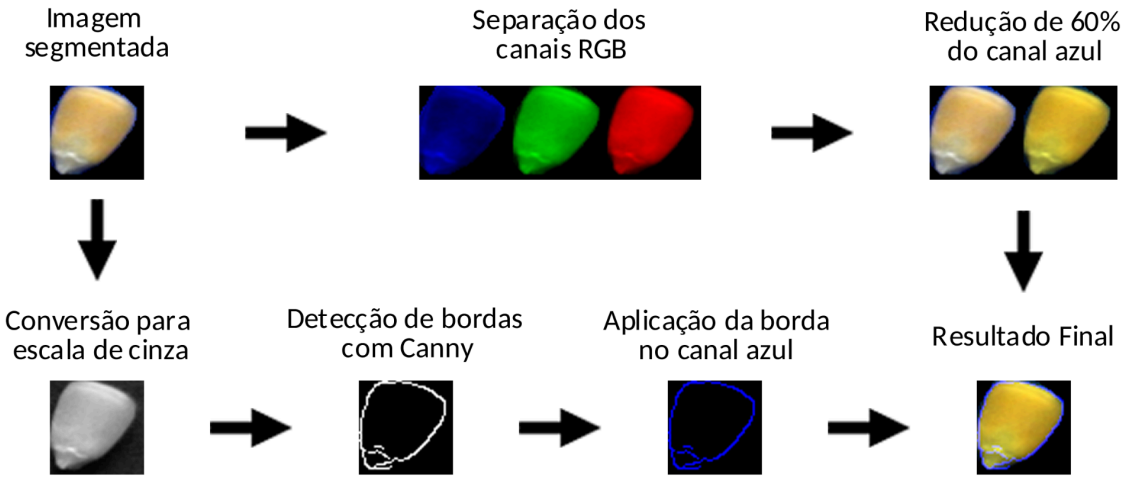
Fonte: Elaboração do autor.

Na segunda etapa do pré-processamento, ilustrada na Figura 19, foi realizado o realce estratégico de atributos cromáticos e morfológicos da imagem segmentada. Inicialmente, os canais RGB foram separados, e aplicou-se uma redução proposital de 60% na intensidade do canal azul, com o intuito de acentuar as tonalidades amareladas, típicas dos grãos de milho sadios.

Em paralelo, o algoritmo de detecção de bordas de Canny foi reaplicado para gerar um mapa refinado de contornos. Esse mapa foi então incorporado ao canal azul atenuado, de forma a evidenciar as bordas estruturais sem comprometer o equilíbrio cromático da imagem.

Por fim, os canais modificados foram recombinados, resultando em imagens padronizadas com três características essenciais: remoção completa do fundo, valorização das cores amareladas e realce das bordas no espectro azul. Essa configuração otimiza a qualidade visual das amostras, favorecendo a etapa subsequente de treinamento do modelo de classificação.

Figura 19 – Segunda etapa do pré-processamento das imagens



Fonte: Elaboração do autor.

4.3 COMPOSIÇÃO DAS LISTAS DE TREINO, VALIDAÇÃO E TESTE

Nessa etapa foram definidas as listas de treino, validação e teste para a concepção do modelo de classificação. Para cada classe foi reservado 64% das imagens para treino, 16% para validação e 20% para teste, conforme detalhado na Tabela 2.

Tabela 2 - Distribuição das imagens nas listas de treino, validação e teste antes do *data augmentation*

Classe	0	1	2	**Total**	**Distribuição**
Classificação	**Ardido**	**Carunchado**	**Sadio**		
Treino	368	55	1.867	2.290	64%
Validação	92	13	466	571	16%
Teste	115	19	585	719	20%
Total	575	87	2.918	3.580	100%

Fonte: Elaboração do autor.

A fim de expandir as listas de treino e validação, foi utilizada a técnica de *data augmentation* (aumento de dados), que gera dados adicionais ao rotacionar e espelhar as imagens já existentes com o objetivo de melhorar a generalização do modelo.

No final desse processo, obteve-se 3669 imagens na lista de treino, 912 na lista de validação e a quantidade de imagens da lista de teste não sofreu variação, permanecendo com 719 imagens, totalizando 5300 imagens para treino, validação e teste do algoritmo, como é detalhado na Tabela 3.

Tabela 3 - Distribuição das imagens nas listas de treino, validação e teste depois do *data augmentation*

Classe	0	1	2	**Total**
Classificação	**Ardido**	**Carunchado**	**Sadio**	
Treino	1.472	330	1.867	3.669
Validação	368	78	466	912
Teste	115	19	585	719
Total	1.955	427	2.918	5.300

Fonte: Elaboração do autor.

4.4 PARAMETRIZAÇÃO DA REDE NEURAL

O algoritmo foi desenvolvido para treinar um modelo de redes neurais convolucionais (CNN) capaz de classificar grãos de milho em três categorias: ardido (classe 0), carunchado (classe 1) e sadio (classe 2). O modelo utiliza um conjunto de 5300 imagens, distribuídas em listas de treino, validação e teste, com dimensões de 48x48 *pixels* e três canais de cor (RGB). O processo de treinamento foi realizado em 150 épocas com um tamanho de lote (*Batch size*) de 512, utilizando o otimizador Adam e a função de perda *categorical_crossentropy*.

A arquitetura da rede neural convolucional consiste em oito camadas convolucionais, cada uma delas com 32 filtros, sendo seis delas com *kernel* de tamanho 3x3 e as duas últimas com *kernel* de tamanho 2x2. Todas as camadas convolucionais utilizam ativação ReLU (*Rectified Linear Unit*), que é amplamente empregada em redes neurais devido à sua eficiência na resolução de problemas de classificação complexos. Em todas as camadas convolucionais foram aplicadas a técnica de preenchimento *padding* do tipo "*same*" e utilizando um *stride* igual a 1, garantindo que a dimensionalidade das imagens seja preservada ao longo das operações.

O modelo também faz uso de camadas de *MaxPooling* após cada camada convolucional, com diferentes tamanhos de *Pool*, variando de (2, 2) a (1, 1). Essas camadas são essenciais para a redução da dimensionalidade e para a extração das principais características das imagens, mantendo as informações mais relevantes e reduzindo o custo computacional.

Além disso, duas camadas de *Dropout* com taxas de 25% foram incluídas — uma após a terceira e outra após a sexta camada convolucional — para prevenir o sobreajuste, o que é particularmente importante em modelos complexos como esse. Em seguida, a rede passa por um *Flatten* e segue para uma camada densa (*Fully Connected*) com 32 neurônios ativados por ReLU. A camada final de saída contém três neurônios (correspondentes às três classes) e utiliza a função de ativação Softmax para gerar as probabilidades de classificação.

O treinamento do modelo foi realizado em um conjunto de dados de treino composto por 3669 imagens, com 912 imagens reservadas para validação e 719 para teste. A performance do modelo foi avaliada por meio de diversas métricas, incluindo a acurácia e métricas mais robustas como precisão, *recall* e *F1-score*,

proporcionando uma visão mais abrangente do desempenho do classificador. Essa abordagem permite uma avaliação minuciosa da capacidade do modelo em distinguir entre as diferentes classes de grãos de milho, contribuindo para a automatização e eficiência dos processos de triagem e controle de qualidade na indústria agrícola. As informações utilizadas para a parametrização estão resumidas no Quadro 6.

Quadro 6 - Resumo Estrutural e Paramétrico da Rede Neural Convolucional

Dataset	Total de imagens: 5.300 imagens Dimensões: 48x48 *pixels*, 3 canais (RGB).
Divisão dos Dados	Treinamento: 3.669 imagens Validação: 912 imagens Teste: 719 imagens.
Arquitetura da CNN	8 camadas convolucionais, cada uma com: - 32 filtros - *Kernel* 3x3 e 2x2 - *Padding*: "*same*" - *Stride*: 1 - Funções de ativação: ReLU
Pooling	*MaxPooling* após cada camada convolucional: - Tamanhos: (2,2) ou (1,1).
Camadas Densas (*Fully Connected*)	32 neurônios (ReLu) Camada final: 3 neurônios (softmax).
Regularização	*Dropout*: 2 camadas, taxa de 25%.
Hiperparâmetros	Épocas: 150 *Batch size*: 512 Otimização: Adam Função de perda: *Categorical Crossentropy*
Avaliação do Modelo	Métricas: Acurácia, Precisão, *Recall*, *F1-score*

Fonte: Elaboração do autor.

4.5 SEGMENTAÇÃO E PREDIÇÃO DO MODELO

O algoritmo para segmentação e classificação de grãos de milho baseia-se em um *pipeline* organizado em quatro etapas principais: pré-processamento, realce de contornos, segmentação por cor e predição do modelo treinado.

Considerando a variabilidade de resolução entre as imagens capturadas por dispositivos móveis, aplica-se, inicialmente, uma função de redimensionamento que padroniza todas as entradas para uma dimensão máxima de 700×700 *pixels*, preservando a proporção original dos eixos. Essa etapa consiste no cálculo da menor razão entre as dimensões-alvo e as dimensões originais, seguido do redimensionamento por interpolação de área e da adição de preenchimento (*padding*) centralizado com *pixels* pretos nas bordas. Dessa forma, assegura-se a uniformidade dimensional das imagens sem comprometer a integridade visual do conteúdo capturado.

Em seguida, o algoritmo incorpora duas estratégias complementares de realce de contornos, replicando o mesmo procedimento utilizado no pré-processamento do conjunto de treino, validação e teste do modelo. A primeira função aplica o detector de bordas de Canny sobre a versão em escala de cinza, multiplica seletivamente os canais de cor — preservando os canais vermelho e verde e atenuando o canal azul — e, finalmente, insere o mapa de contornos no canal azul, reforçando visualmente as arestas dos grãos.

A segunda função converte a imagem para escala de cinza, detecta bordas com Canny, preenche eventuais lacunas por meio de fechamento morfológico e identifica o contorno externo de maior área. A aplicação de *convex hull* sobre esse contorno gera uma máscara binária que, quando combinada *bitwise* com a imagem original, isola completamente cada grão, eliminando ruídos periféricos e garantindo que o fundo da imagem não interfira na predição.

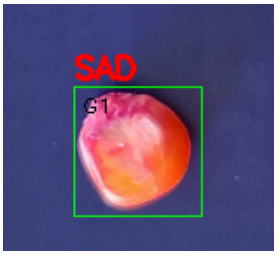
A segmentação baseia-se no espaço de cor HSV para separar o fundo, predominantemente azul, da amostra dos grãos. Define-se um intervalo de matizes que corresponde ao tom do fundo, gera-se a máscara de fundo e, em seguida, inverte-se para obter a máscara dos grãos.

Essa máscara binária passa por operações de abertura e fechamento morfológico para remover pequenos fragmentos e preencher falhas, resultando em

regiões mais coesas. Contornos externos são então extraídos e filtrados com base em dois critérios: área mínima para descartar pequenos ruídos (menos de 300 *pixels*) e solidez (razão entre a área do contorno e a área de sua caixa delimitadora), no qual valores inferiores a 0,6 indicam fragmentação ou segmentação incompleta, sendo descartados.

Cada contorno válido origina uma ROI (*Region of Interest* – Região de Interesse), expandida por uma pequena margem para assegurar a integridade do grão segmentado. Essas ROIs são então redimensionadas para 48 × 48 *pixels*, adequando-se à dimensão de entrada exigida pela rede neural convolucional previamente treinada. A Figura 20 ilustra um exemplo de ROI, destacando com um retângulo verde a região correspondente a um grão de milho em uma amostra.

Figura 20 – Exemplo de uma ROI em uma amostra de grão de milho



Fonte: Elaboração do autor.

Antes da inferência, as imagens passam novamente pelas funções de realce de contorno, amplificando bordas e texturas cruciais para a diferenciação de classes. Em seguida, normalizam-se os valores de *pixel* para o intervalo [0,1] e acrescenta-se a dimensão de *batch*, preparando o tensor para o método de predição do modelo, que produz probabilidades associadas às classes ‘ARD’, ‘CAR’ e ‘SAD’. A classe de maior probabilidade é selecionada como predição final. Por fim, o algoritmo anota a imagem original redimensionada, desenhando caixas delimitadoras em volta de cada grão e incluindo o rótulo predito, o que facilita a validação visual em campo.

4.6 CÁLCULO DA ÁREA E TIPIFICAÇÃO DA AMOSTRA

Após as etapas de segmentação e predição individual dos grãos de milho, baseadas nas imagens segmentadas e pré-processadas, o algoritmo realiza a quantificação da área correspondente a cada classe identificada — ardido (ARD), carunchado (CAR) e sadio (SAD).

Essa análise visa estimar a proporção relativa de cada tipo de defeito presente na amostra, permitindo a posterior tipificação conforme os critérios da Instrução Normativa nº 60/2011 do Ministério da Agricultura, Pecuária e Abastecimento (MAPA). Esta etapa é fundamental, uma vez que a referida norma estabelece limites percentuais máximos para defeitos, os quais determinam a classificação da amostra em Tipo 1, Tipo 2, Tipo 3 ou 'Fora de Tipo'.

Embora a Instrução Normativa nº 60/2011 estabeleça que a tipificação da amostra deve ser realizada com base na proporção das massas de cada classe de grão, neste trabalho optou-se pela utilização da proporção das áreas ocupadas por cada classe na imagem como medida substitutiva. Essa abordagem foi adotada devido à inviabilidade técnica de estimar com precisão o volume individual dos grãos a partir de imagens bidimensionais.

Para atender à exigência normativa de que a classificação da amostra seja realizada com base na proporção em massa dos grãos, adotou-se como premissa que todos os grãos possuem densidade semelhante. Considerando ainda que as variações de volume entre os grãos se refletem proporcionalmente na área projetada visível nas imagens, a massa estimada de cada classe de grãos (m_i) pode ser aproximada pela respectiva área segmentada (A_i).

A massa do grão pode ser calculada como o produto entre a densidade (ρ) e o volume do grão (V_i), conforme expressa a Equação 11:

$$m_i = \rho . V_i \tag{11}$$

Assumindo que o volume do grão (V_i) é proporcional à área ocupada projetada na imagem (A_i), e considerando que a densidade dos grãos é aproximadamente constante (ρ ≈ constante), chega-se na Equação 12:

$$\frac{m_i}{\Sigma m} \approx \frac{V_i}{\Sigma V} \approx \frac{A_i}{\Sigma A} \tag{12}$$

Dessa forma, a proporção da massa de cada classe de grãos em relação à massa total da amostra pode ser estimada pela razão entre a área visível ocupada por essa classe e a área total ocupada pelos grãos na imagem segmentada. Ou

seja, se A_i representa a área segmentada de uma determinada classe, e ΣA representa a somatória de todas as áreas ocupadas pelos grãos da amostra, então a razão entre as áreas pode ser utilizada como uma estimativa da relação m_i e Σm onde m_i e Σm são, respectivamente, a massa da classe analisada e a massa total da amostra.

Sob a hipótese de densidade semelhante entre os grãos e considerando que suas formas e espessuras médias são comparáveis, a área visível projetada em imagem bidimensional pode ser considerada proporcional ao volume, e, consequentemente, à massa. Assim, um valor como 3% da área ocupada por grãos ardidos em relação à área total da amostra constitui uma estimativa razoável para representar 3% da massa de grãos ardidos em relação à massa total.

Essa equivalência percentual justifica o uso da proporção de área segmentada como critério técnico para a classificação automatizada das amostras, em conformidade com a Instrução Normativa nº 60/2011 do MAPA, que exige que os percentuais de defeitos sejam expressos com base na massa da amostra analisada.

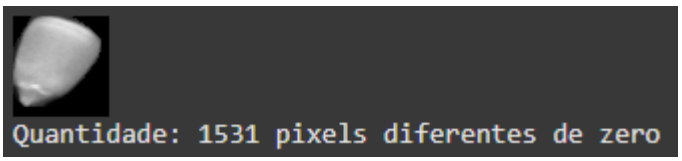
Para efetuar a quantificação da área, o algoritmo percorre individualmente cada grão previamente identificado, extraindo sua respectiva imagem segmentada e calculando a área ocupada com base na contagem de *pixels* distintos do fundo.

Essa análise considera exclusivamente os *pixels* com valores diferentes de zero na imagem em escala de cinza correspondente à máscara binária gerada na etapa de segmentação. A área de cada grão, expressa em número de *pixels*, é registrada e associada à classe predita, permitindo a posterior estimativa da distribuição percentual das classes na amostra.

A Figura 21 apresenta um exemplo do processo de cálculo da área visível de um grão de milho segmentado. Nesse procedimento, o algoritmo percorre a imagem segmentada e contabiliza os *pixels* com valor diferente de zero, ou seja, distintos do fundo preto.

A imagem do grão analisado possui dimensões de 48 × 48 *pixels*, totalizando 2.304 *pixels*. Desse total, 1.531 *pixels* correspondem à área ocupada pelo grão na imagem. Esse valor é posteriormente utilizado para estimar, de forma proporcional, a massa relativa da classe correspondente, contribuindo para a classificação da amostra segundo os critérios estabelecidos na norma.

Figura 21 – Exemplo de uma quantificação de área de uma amostra de grão de milho segmentado



Fonte: Elaboração do autor.

Na etapa seguinte à quantificação das áreas, o algoritmo agrupa os grãos conforme a classe predita — ardido, carunchado ou sadio — e realiza a soma das áreas individuais pertencentes a cada categoria, conforme descrito na Equação 13. Essa operação fornece os valores totais de área segmentada por classe, os quais são utilizados para calcular a proporção relativa de cada tipo de grão em relação à área total da amostra.

$$\sum A_{total} = \sum A_{ard} + \sum A_{car} + \sum A_{sad} \tag{13}$$

Em seguida, calcula-se a proporção percentual (p) de cada classe em relação à área total da amostra, conforme demonstrado nas Equações 14, 15 e 16 — correspondentes, respectivamente, às classes de grãos ardidos, carunchados e sadios. Esse processo fornece uma estimativa da representatividade espacial de cada tipo de grão na imagem analisada.

$$p_{ard\ (\%)} = \frac{\sum A_{ard}}{\sum A_{total}} \tag{14}$$

$$p_{car\ (\%)} = \frac{\sum A_{car}}{\sum A_{total}} \tag{15}$$

$$p_{sad\ (\%)} = \frac{\sum A_{sad}}{\sum A_{total}} \tag{16}$$

Com base nos percentuais de área calculados para cada classe de grão, o algoritmo realiza a classificação da amostra quanto ao tipo comercial do milho, aplicando os critérios estabelecidos pela legislação brasileira, que define faixas percentuais específicas para enquadrar a amostra nos Tipos 1, 2 ou 3, ou classificá-la como “Fora de Tipo”; essa decisão considera os limites máximos

permitidos para os teores individuais de grãos ardidos e carunchados, bem como a soma desses dois defeitos.

Por exemplo, para que a amostra seja considerada Tipo 1, os grãos ardidos devem estar abaixo de 1%, os carunchados abaixo de 2%, e a soma de ambos não pode ultrapassar 6%. Caso algum desses limites seja excedido, a amostra é reclassificada para um tipo inferior ou, em situações mais críticas, desclassificada.

4.7 MÉTRICAS DE AVALIAÇÃO

A avaliação do desempenho do modelo de rede neural convolucional (CNN) proposto foi conduzida por meio de métricas amplamente reconhecidas na literatura de aprendizado de máquina: acurácia, precisão, *recall* e *F1-score*. Cada uma dessas métricas fornece uma perspectiva complementar sobre a eficácia do modelo na tarefa de classificação, permitindo uma análise robusta e detalhada de seu comportamento frente ao conjunto de dados de teste.

A acurácia (Equação 7) representa a proporção de predições corretas em relação ao total de amostras avaliadas, oferecendo uma visão geral do desempenho do modelo. Contudo, essa métrica pode ser influenciada pelo balanceamento das classes, tornando-se menos informativa em contextos com desequilíbrio de dados, situação comum em problemas de classificação multiclasse.

Por sua vez, a precisão (Equação 8) mensura a proporção de amostras corretamente classificadas em uma determinada classe em relação ao total de predições atribuídas a essa classe. Essa métrica reflete o grau de confiança do modelo ao indicar que uma amostra pertence a uma categoria específica e é especialmente importante para minimizar falsos positivos, garantindo que as classificações positivas sejam realmente válidas.

O *recall* (Equação 9) avalia a capacidade do modelo em identificar corretamente todas as amostras que realmente pertencem a uma determinada classe. Ao calcular a proporção de verdadeiros positivos em relação ao total de amostras relevantes, essa métrica é essencial em cenários que demandam a minimização de falsos negativos, evitando que elementos de interesse sejam omitidos na classificação.

Por fim, o *F1-score* (Equação 10) representa a média harmônica entre precisão e *recall*, combinando essas duas métricas em um único índice. Essa

medida possibilita avaliar o equilíbrio entre a exatidão das predições e a capacidade do modelo de recuperar todas as amostras relevantes, sendo especialmente valiosa em contextos com desequilíbrio entre classes ou quando é necessário controlar com igual rigor tanto os falsos positivos quanto os falsos negativos.

A utilização combinada dessas quatro métricas permite uma avaliação detalhada e multifacetada do desempenho do modelo de CNN, facilitando a identificação de seus pontos fortes e limitações em tarefas de classificação automática. Dessa maneira, a análise integrada de acurácia, precisão, *recall* e *F1-score* oferece uma base técnica robusta para avaliar a viabilidade do modelo em aplicações práticas.

4.8 APLICATIVO DE CLASSIFICAÇÃO E TIPIFICAÇÃO DOS GRÃOS

4.8.1 Desenvolvimento e Estrutura

O desenvolvimento do aplicativo ZeaScan, voltado para a classificação e tipificação de grãos de milho, foi baseado em uma arquitetura descentralizada, que assegura a independência entre as camadas de interface e de processamento. Essa abordagem facilita manutenções, ampliações e futuras integrações com outras plataformas, promovendo maior agilidade e escalabilidade na evolução do sistema.

Para isso, optou-se pela utilização do *framework Flutter* na implementação do *frontend*, uma vez que ele oferece uma base robusta para o desenvolvimento de interfaces gráficas responsivas e multiplataforma. Essa escolha assegura consistência visual, desempenho eficiente e compatibilidade especialmente otimizada para dispositivos Android.

Dentro desse contexto, a estruturação do aplicativo segue o padrão do *MaterialApp*, elemento central do *Flutter* que fornece temas, navegação e gerenciamento de estados básicos, enquanto a adoção da biblioteca *FlutterModular* oferece um mecanismo de modularização de rotas e injeção de dependências, facilitando a separação de responsabilidades e a manutenção do código. Dessa maneira, cada módulo do aplicativo — seja a tela de captura de imagem, o componente de pré-visualização ou o painel de exibição dos resultados — é encapsulado, o que confere maior robustez e escalabilidade à base de código.

A separação entre as camadas de *frontend* e *backend* permite que

dispositivos móveis com diferentes versões do sistema Android executem a aplicação de forma homogênea, sem a necessidade de recompilação ou reconfiguração do modelo de *machine learning*, uma vez que o processamento intensivo é delegado integralmente ao servidor.

No servidor, a API foi implementada em Python 3, utilizando o *microframework Flask* para o roteamento das requisições e o gerenciamento das interações via protocolo HTTP. Para operação em ambiente de produção, emprega-se o servidor WSGI Gunicorn, que assegura alta disponibilidade e escalabilidade por meio do gerenciamento eficiente de múltiplos *workers* simultâneos.

A API adota a arquitetura MVC (*Model-View-Controller*), adaptada ao contexto de um serviço web. Nesse arranjo, o *Controller* é responsável por receber a imagem transmitida pelo aplicativo, executar etapas adicionais de pré-processamento e acionar a camada de *Model*, que encapsula o carregamento e a execução do modelo de rede neural convolucional (CNN) otimizado para inferência. Após o processamento, o *Controller* gera uma resposta estruturada em formato JSON. A *View*, nesse cenário, corresponde aos *endpoints* que retornam essas respostas já formatadas, fornecendo ao aplicativo *Flutter* todas as informações necessárias para a apresentação dos resultados ao usuário final.

A comunicação entre o *frontend* e a API é feita por meio de chamadas HTTP assíncronas, utilizando endpoints seguros que seguem o padrão RESTful. Quando o usuário envia uma imagem por meio do formato multipart/form-data, o aplicativo recebe como resposta um JSON contendo a tipificação da amostra analisada. Com base nessa resposta, o ZeaScan exibe de forma imediata uma interface gráfica interativa, sobrepondo à própria fotografia da amostra a classificação final — *Tipo 1, Tipo 2, Tipo 3* ou *Fora de Tipo* — conforme os critérios técnicos estabelecidos pela Instrução Normativa nº 60/2011 do MAPA. Esse retorno em tempo real é essencial para a atuação ágil dos operadores em campo, permitindo decisões rápidas quanto ao descarte de lotes, separação de amostras ou acionamento de processos de correção, como a limpeza e a secagem dos grãos.

A escolha pelo *Flutter* se deu pela necessidade de desenvolver um aplicativo leve, com desempenho fluido, capaz de operar em dispositivos com limitações de *hardware*, como é comum em contextos de uso no campo. Além disso, o *Flutter* oferece portabilidade nativa entre plataformas, o que viabiliza futuras expansões para iOS e até mesmo aplicações web ou desktop com uma base de código única.

A adoção da biblioteca *FlutterModular* reforça essa arquitetura escalável e sustentável ao proporcionar uma separação clara de responsabilidades. A modularização do projeto permite que cada funcionalidade seja desenvolvida e testada de forma independente. Além disso, o *FlutterModular* oferece um mecanismo robusto de injeção de dependências, essencial para gerenciar serviços como clientes HTTP, controladores de estado e provedores de autenticação, garantindo a manutenção eficiente da comunicação entre o *frontend* e o *backend*.

No servidor, a arquitetura MVC assegura a separação de responsabilidades, permitindo que modificações no modelo de *machine learning* sejam realizadas exclusivamente na camada *Model*, sem afetar o roteamento (*Controller*) ou a apresentação das respostas (*View*), o que facilita a manutenção e o versionamento do sistema. Além disso, a utilização do Gunicorn como servidor WSGI possibilita a execução concorrente de múltiplos processos, garantindo escalabilidade horizontal, baixa latência e alta disponibilidade, mesmo sob demandas elevadas, assegurando a responsividade e a confiabilidade do serviço ZeaScan.

Além da otimização de desempenho, as bibliotecas adotadas no *backend* contemplam a implementação de logs centralizados e configuráveis, possibilitando o monitoramento detalhado do fluxo das requisições e a identificação de eventuais gargalos durante o processo de inferência. A arquitetura MVC contribui igualmente para a organização dos testes, facilitando a aplicação de testes unitários e de integração. Dessa forma, os módulos da camada *Model* podem ser avaliados isoladamente por meio de imagens de referência, enquanto os *Controllers* são submetidos a testes automatizados que simulam requisições HTTP, garantindo a robustez e a confiabilidade do sistema como um todo.

No aplicativo, a execução de testes de *widget* e de integração assegura a estabilidade das interfaces, especialmente nas telas de captura de imagem, bem como a fluidez e consistência da experiência do usuário durante a navegação e interação com a aplicação.

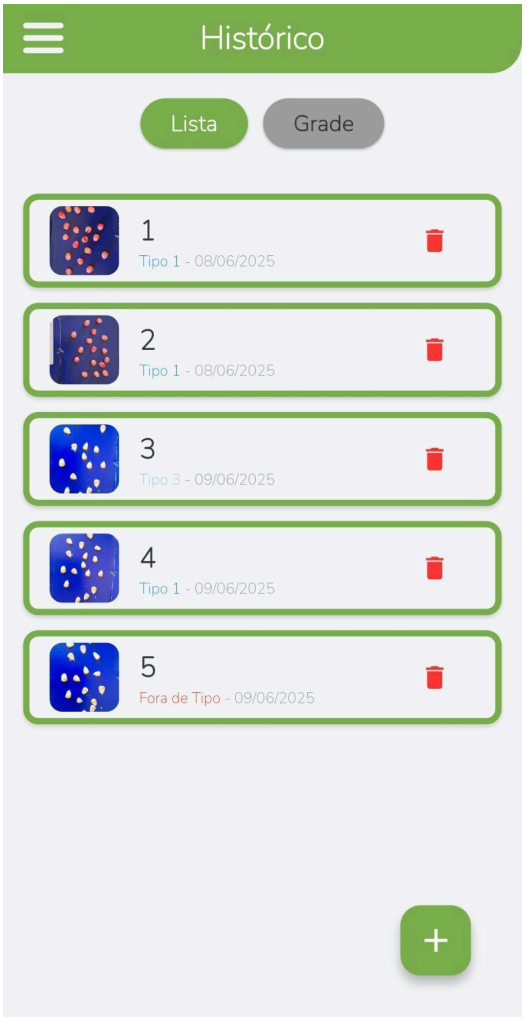
4.8.2 *Design* da Interface e Funcionalidades

O aplicativo ZeaScan foi desenvolvido com foco na simplicidade e na funcionalidade, visando oferecer uma experiência prática e eficiente para a tipificação automatizada de amostras de grãos de milho.

Seu design intuitivo apresenta, logo na tela inicial, o histórico das análises já realizadas. As amostras são exibidas em formato de lista ou grade, contendo informações como o nome atribuído, a data da análise e o respectivo resultado da tipificação — Tipo 1, Tipo 2, Tipo 3 ou Fora de Tipo.

A Figura 22 ilustra a tela inicial do ZeaScan, evidenciando o histórico das amostras tipificadas e o botão flutuante que possibilita iniciar uma nova análise de forma rápida e direta.

Figura 22 – Tela inicial do aplicativo ZEASCAN



Fonte: Elaboração do autor.

Essa estrutura facilita o rastreamento e o gerenciamento das análises realizadas, atendendo tanto usuários com perfil técnico quanto produtores e profissionais da cadeia agrícola. A flexibilidade da interface permite o uso eficiente em diferentes contextos, desde o campo até ambientes laboratoriais.

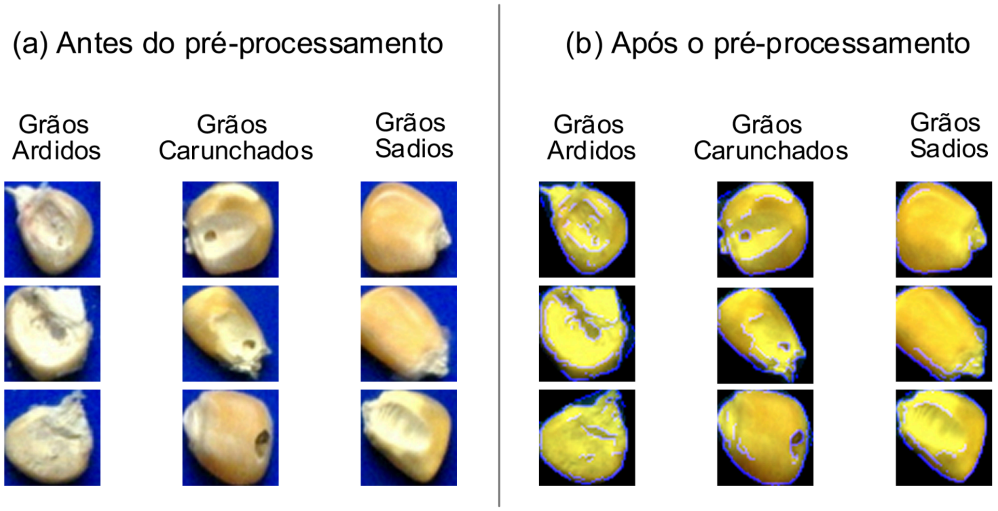
No canto inferior direito da tela inicial, um botão flutuante com o ícone “+” permite iniciar uma nova análise. Ao ser acionado, o aplicativo oferece duas opções ao usuário: selecionar uma imagem já armazenada no dispositivo ou capturar uma nova foto diretamente pela câmera do *smartphone*.

5 RESULTADOS

5.1 PRÉ-PROCESSAMENTO DO *DATASET*

Como apresentado no capítulo anterior, realizou-se um pré-processamento das imagens visando realçar a morfologia e as características cromáticas dos grãos. Inicialmente, definiu-se o contorno das sementes por meio de detecção de bordas e segmentação do fundo, seguido da aplicação de máscara binária que elimina completamente o plano de fundo, destacando assim o objeto de interesse. Em seguida, procedeu-se à atenuação seletiva do canal azul e ao reforço do traçado de contorno e das texturas superficiais do grão, de modo a evidenciar possíveis defeitos. A Figura 23 ilustra, para cada classe, três exemplos antes e após o processamento aplicado pelo algoritmo.

Figura 23 – (a) Amostras das imagens antes do pré-processamento e (b) Amostras das imagens após do pré-processamento



Fonte: Elaboração do autor.

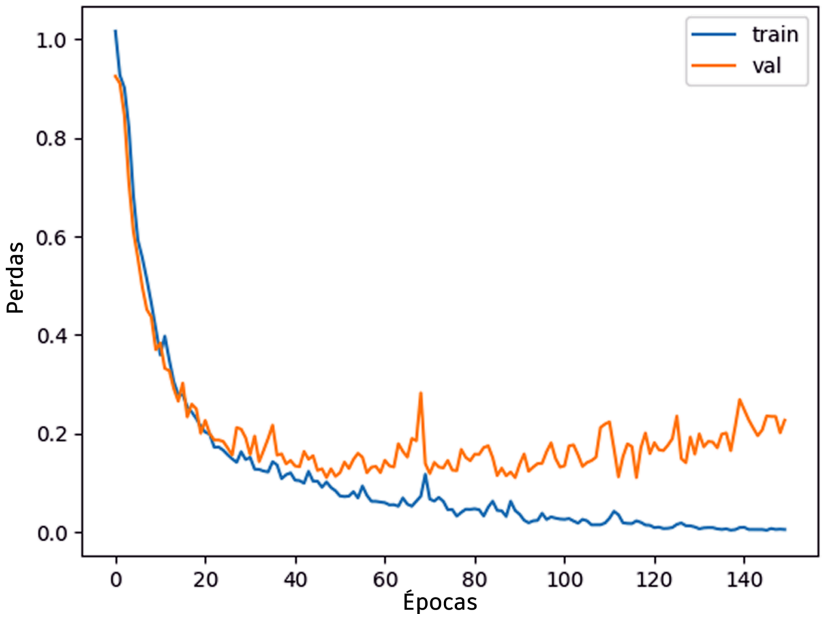
Como evidenciado na Figura 23, o pré-processamento resulta em imagens com o fundo escurecido, contornos dos grãos realçados em azul e realce das cores da superfície em comparação com as imagens originais. Dessa forma, as características de forma, contorno e textura tornam-se mais nítidas, facilitando a extração de atributos pela rede neural convolucional.

Ainda na Figura 23, observa-se que os grãos ardidos tendem a exibir contornos superficiais irregulares, resultado de seu aspecto rugoso. Nos grãos carunchados, nota-se a presença de um contorno interno menor dentro do contorno externo do grão, evidenciando o orifício causado por insetos. Por sua vez, os grãos sadios geralmente apresentam apenas o contorno externo, devido à superfície mais lisa, quando comparada às demais classes.

5.2 TREINAMENTO DO MODELO

Após o pré-processamento, o modelo foi desenvolvido utilizando conjuntos de imagens para treino e validação, com um total de 150 épocas durante o processo de treinamento. A Figura 24 apresenta o gráfico de redução da função de perda ao longo das épocas, evidenciando a progressiva otimização do desempenho do modelo.

Figura 24 – Perdas por época do treino e validação do modelo



Fonte: Elaboração do autor.

O gráfico da Figura 24 apresenta a evolução das perdas do conjunto de treino e validação ao longo das épocas. Inicialmente, as perdas para ambos os conjuntos

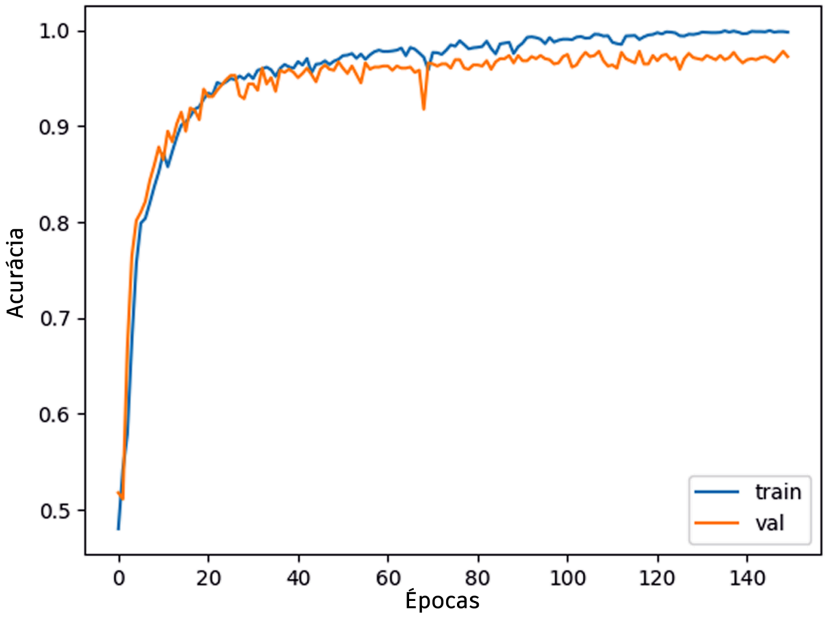
iniciam em valores elevados, indicando que o modelo ainda não foi suficientemente ajustado aos dados. À medida que o treinamento avança, observa-se uma queda acentuada nas perdas, especialmente nas primeiras 20 a 30 épocas, o que reflete a eficiência do algoritmo de otimização em reduzir o erro.

Após essa fase inicial, as perdas do conjunto de treino continuam diminuindo de forma consistente até se estabilizarem em torno de valores muito baixos, abaixo de 0,01, evidenciando que o modelo conseguiu ajustar-se bem aos dados de treino. Por outro lado, as perdas do conjunto de validação apresentam maior oscilação, com uma estabilização menos evidente. Isso indica que, embora o modelo esteja generalizando bem, há certa sensibilidade aos dados de validação, possivelmente causada por variações no conjunto ou por leve sobreajuste (*overfitting*) nas últimas épocas.

Um ponto importante a ser destacado é que, mesmo com as oscilações na validação, o modelo não apresenta uma divergência significativa entre as curvas de treino e validação, o que sugere que o sobreajuste foi moderado e o treinamento foi conduzido de maneira eficiente.

A Figura 25 apresenta o gráfico da evolução da acurácia ao longo das épocas de treinamento do modelo.

Figura 25 – Acurácia por época do treino e validação do modelo



Fonte: Elaboração do autor.

O gráfico da Figura 25 complementa a análise do desempenho exibida na Figura 24, ao ilustrar a progressão da acurácia nos conjuntos de treino e validação. Inicialmente, ambos os conjuntos apresentam acurácias baixas, o que é esperado, já que o modelo ainda está em fase inicial de aprendizagem. No entanto, entre as épocas 20 e 30, observa-se um aumento acentuado na acurácia, acompanhando a redução da função de perda, o que indica que o modelo passou a identificar padrões relevantes nas imagens.

A partir da metade do treinamento (em torno de 70 a 80 épocas), a acurácia do conjunto de treino aproxima-se de 1,0, indicando que o modelo conseguiu aprender praticamente todos os padrões presentes nos dados de treino. Já a acurácia do conjunto de validação estabiliza-se em torno de 0,95, com pequenas oscilações. Essa diferença entre os dois conjuntos é esperada e reflete a tendência natural de o modelo apresentar desempenho ligeiramente superior nos dados de treino, que são vistos repetidamente durante o treinamento.

Outro ponto relevante é que a curva de validação não apresenta quedas acentuadas ou instabilidades significativas, o que reforça a robustez do modelo e sua capacidade de generalização. A diferença entre as acurácias de treino e validação é relativamente pequena, indicando que o modelo não sofreu de sobreajuste.

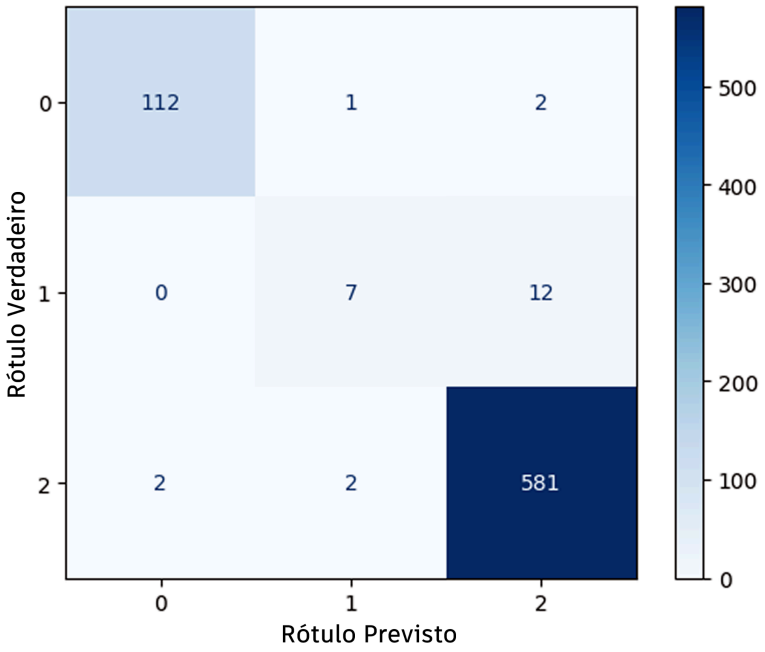
5.3 TESTE DO MODELO

O modelo desenvolvido com base em Rede Neural Convolucional foi treinado ao longo de 150 épocas e, nos testes realizados, apresentou uma acurácia global de 97,36%. As métricas de precisão, *recall* e *F1-score* também alcançaram o mesmo valor (97,36%), evidenciando um desempenho equilibrado e robusto.

O teste do modelo foi conduzido com um total de 719 imagens de grãos de milho, das quais 700 foram corretamente classificadas, resultando na acurácia mencionada. Esses resultados demonstram a elevada eficácia do modelo na tarefa de classificação, reforçando sua adequação para aplicações práticas na avaliação automatizada de grãos.

Para uma melhor visualização dos resultados, foi gerada a matriz de confusão com o objetivo de avaliar a correspondência entre as predições do modelo e a classificação real dos grãos de milho. A Figura 26 apresenta essa matriz, contemplando as três classes consideradas: “0” para grãos ardidos, “1” para grãos carunchados e “2” para grãos sadios.

Figura 26 – Matriz de confusão do teste do modelo



Fonte: Elaboração do autor.

O modelo apresentou desempenho excepcional na classificação dos grãos ardidos, com 112 acertos em um total de 115 amostras pertencentes a essa classe. O número de verdadeiros negativos (602) evidencia que, em 602 ocasiões, o modelo classificou corretamente grãos carunchados ou sadios como não ardidos, demonstrando baixa taxa de confusão com outras classes. Os indicadores quantitativos reforçam a robustez do modelo: acurácia de 99,30%, precisão de 98,25%, *recall* de 97,39% e *F1-score* de 97,82%. Esses resultados revelam elevada capacidade de identificação da classe “Ardido”, com equilíbrio entre sensibilidade e especificidade, sendo adequada para aplicações práticas que exigem alta confiabilidade na detecção desse tipo de defeito.

A classificação dos grãos carunchados representou o maior desafio para o modelo, possivelmente em decorrência da alta similaridade visual com os grãos

sadios. Das 19 amostras carunchadas, apenas 7 foram corretamente identificadas, enquanto 12 foram erroneamente classificadas como sadias. Apesar disso, o valor de verdadeiros negativos (697) indica que o modelo foi eficaz ao evitar a atribuição incorreta da classe “Carunchado” a amostras ardidas ou sadias. Ainda que a acurácia para essa classe tenha alcançado 97,91%, os indicadores de precisão (70,00%), *recall* (36,84%) e *F1-score* (48,28%) revelam uma limitação significativa na sensibilidade do modelo para detectar corretamente os grãos carunchados. Esse desempenho inferior evidencia a necessidade de aprimoramentos específicos, como aumento do número de amostras dessa classe ou técnicas de balanceamento de dados, a fim de elevar a capacidade discriminativa do sistema.

Em relação à classe de grãos sadios, o modelo apresentou desempenho altamente satisfatório, com 581 das 585 amostras classificadas corretamente. Apenas quatro grãos foram incorretamente identificados, sendo dois rotulados como “ardidos” e dois como “carunchados”. O valor de verdadeiros negativos (120) indica que o modelo também foi eficiente ao reconhecer como não sadias as amostras pertencentes às demais classes. Os resultados obtidos — acurácia de 97,50%, precisão de 97,65%, *recall* de 99,32% e *F1-score* de 98,47% — demonstram a robustez do modelo na identificação de grãos sadios, evidenciando sua elevada capacidade de generalização e precisão nesta categoria.

Na análise global do sistema, registraram-se 700 verdadeiros positivos, indicando acertos na classificação das amostras, enquanto 19 falsos positivos e 19 falsos negativos representaram, respectivamente, atribuições incorretas e falhas na identificação da classe correta.

O valor nulo para os verdadeiros negativos decorre do fato de todas as 719 amostras pertencerem a uma das três classes avaliadas, inexistindo exemplos de “não pertencentes” ao conjunto de interesse. Assim, não há situações em que o modelo pudesse reconhecer corretamente algo como “fora das classes”. Esses resultados, traduzidos em acurácia, precisão, recall e F1-score globais de 97,36%, evidenciam a elevada consistência e equilíbrio do modelo no desempenho geral, mesmo considerando as diferenças entre as classes individuais.

A Tabela 4 apresenta um resumo dos dados extraídos da matriz de confusão (Figura 26), incluindo os valores de acurácia, precisão, *recall*, *F1-score* e a quantidade de verdadeiros negativos para cada uma das classes analisadas.

Tabela 4 - Acurácia, precisão, *recall* e *F1-score* por classe

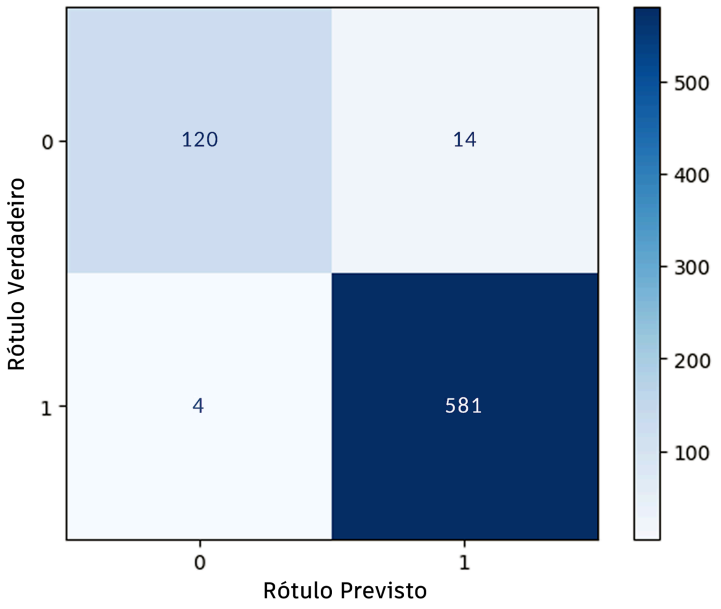
Classe	Verdadeiro Positivo (VP)	Verdadeiro Negativo (VN)	Falso Positivo (FP)	Falso Negativo (FN)	Acurácia	Precisão	*Recall*	*F1 score*
Ardido	112	602	2	3	99,30%	98,25%	97,39%	97,82%
Carunchado	7	697	3	12	97,91%	70,00%	36,84%	48,28%
Sadio	581	120	14	4	97,50%	97,65%	99,32%	98,47%
Total	**700**	**-**	**19**	**19**	**97,36%**	**97,36%**	**97,36%**	**97,36%**

Fonte: Elaboração do autor.

Com o intuito de verificar a capacidade do modelo em distinguir grãos sadios daqueles com defeitos (ardidos ou carunchados), realizou-se uma avaliação em formato binário. Essa abordagem simula um cenário prático de triagem, no qual o objetivo principal é separar os grãos comercializáveis dos que não atendem aos critérios de conformidade.

A Figura 27 apresenta a matriz de confusão resultante dessa avaliação, na qual a classe “0” representa os grãos defeituosos (ardidos ou carunchados) e a classe “1” corresponde aos grãos sadios.

Figura 27 – Matriz de confusão para o modelo binário



Fonte: Elaboração do autor.

Os resultados demonstram a eficácia do modelo em distinguir grãos conformes (sadio) e não conformes (ardido e carunchado), evidenciando seu potencial de aplicação em sistemas automatizados de controle de qualidade.

A Tabela 5 apresenta um resumo das métricas de desempenho obtidas na classificação binária, confirmando a robustez do modelo frente à tarefa proposta e reforçando sua aplicabilidade prática em ambientes industriais e agronômicos.

Tabela 5 - Acurácia, precisão, *recall* e *F1-score* considerando duas classes: “Defeituoso” e “Sadio”

Classe	Verdadeiro Positivo (VP)	Verdadeiro Negativo (VN)	Falso Positivo (FP)	Falso Negativo (FN)	Acurácia	Precisão	*Recall*	*F1 score*
Defeituoso (ardidos + carunhcados)	120	581	4	14	97,50%	96,77%	89,55%	93,02%
Sadio	581	120	14	4	97,50%	97,65%	99,32%	98,47%
Total	**701**	**-**	**18**	**18**	**97,50%**	**97,50%**	**97,50%**	**97,50%**

Fonte: Elaboração do autor.

De forma semelhante aos dados obtidos na Tabela 4, o valor nulo para os verdadeiros negativos na Tabela 5 ocorre porque a análise foi realizada considerando apenas duas classes mutuamente exclusivas: “Defeituoso” e “Sadio”. Nesse cenário binário, toda amostra obrigatoriamente pertence a uma dessas classes, não havendo, portanto, espaço para contabilizar verdadeiros negativos fora desse conjunto.

Os resultados obtidos confirmam o excelente desempenho do modelo nessa configuração binária. A acurácia global alcançou 97,50%. O modelo classificou corretamente 120 das 134 amostras defeituosas (89,55% de *recall*) e 581 das 585 amostras sadias (99,32% de *recall*), totalizando 701 acertos em 719 imagens analisadas.

As métricas por classe destacam-se pela alta precisão na detecção de itens sadios (97,65% de precisão) e robustez na identificação de defeituosos (96,77% de precisão e 93,02% de *F1-score*), embora o *recall* para 'Defeituoso' (89,55%) aponte espaço para melhorias.

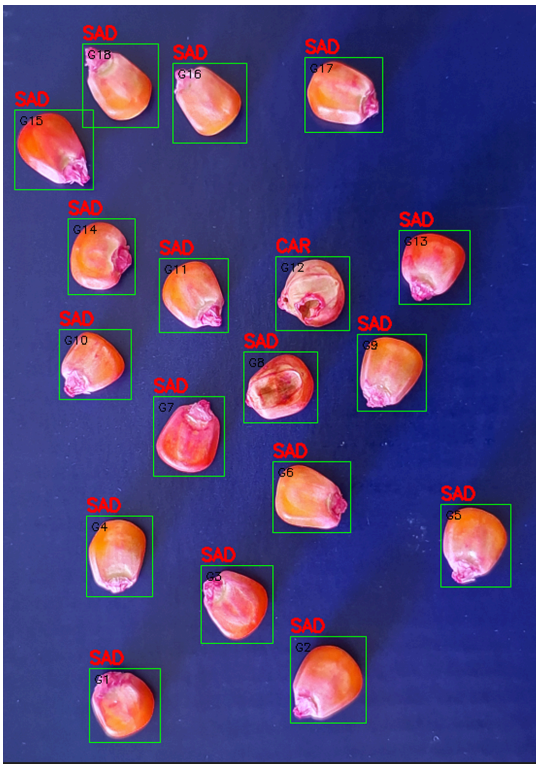
5.4 SEGMENTAÇÃO DOS GRÃOS

O resultado da aplicação do *pipeline* de segmentação e classificação de grãos de milho é apresentado na Figura 28, evidenciando sua eficácia em condições controladas. O processo incluiu o redimensionamento da imagem original para uma largura fixa de 700 *pixels*, mantendo a proporção da altura para garantir uniformidade na entrada do sistema. A segmentação foi conduzida com base em critérios de cor no espaço HSV, seguida de um refinamento morfológico com filtragem por área mínima e solidez, o que permitiu o isolamento preciso dos grãos sobre o fundo azul.

A etapa de realce de contornos combinou múltiplas técnicas: detecção de bordas com o operador de Canny, reforço seletivo nos canais de cor e aplicação do *convex hull* (envoltória convexa). Essas estratégias resultaram em regiões de interesse (ROIs) com alta qualidade visual e estrutural, mantendo texturas e arestas relevantes para o processo de classificação. As ROIs foram, então, encaminhadas ao modelo de rede neural convolucional (CNN), que realizou automaticamente a classificação dos grãos em três categorias: ardido (ARD), carunchado (CAR) e sadio (SAD).

A Figura 28 exemplifica a análise ao ilustrar a robustez do *pipeline* frente a variações na coloração dos grãos. Nesse teste com 18 grãos, sendo 17 sadios e 1 carunchado, o sistema classificou todos corretamente, demonstrando precisão tanto na segmentação quanto na inferência. A amostra apresentou grãos com tonalidade rosada, resultado do uso de corantes em tratamentos com fungicidas e inseticidas — uma característica típica de sementes destinadas ao plantio e distinta da coloração amarela dos grãos usados no treinamento.

Figura 28 – Teste de segmentação, classificação e tipificação de amostra de grãos de milho



Fonte: Elaboração do autor.

O resultado das etapas de segmentação, pré-processamento, classificação e cálculo de área indicou que o somatório de *pixels* correspondentes a cada classe foi de 0 para grãos ardidos, 1.659 para grãos carunchados e 27.294 para grãos sadios, conforme apresentado na Tabela 6. A partir desses valores, foram obtidos o seguintes percentuais de composição da amostra: $p_{ARD}(\%)=0\%$, $p_{CAR}(\%)=5{,}73\%$ e $p_{SAD}(\%)=94{,}27\%$.Esses percentuais refletem a proporção de área visível ocupada por cada classe na imagem analisada.

Tabela 6 – Áreas por classe resultante da segmentação e classificação

Classe	**Área (*pixels*)**	**Percentual**
Ardidos	0	0,00%
Carunchados	1.659	5,73%
Sadios	27.294	94,27%
Total	28.953	100,00%

Fonte: Elaboração do autor.

De acordo com os critérios de tipificação definidos pela Instrução Normativa nº 60/2011 (Tabela 1), esses valores classificam a amostra como “Fora de Tipo”.

Essa variação cromática, que se afasta dos tons convencionais de amarelo e alaranjado, impõe um desafio adicional aos algoritmos de segmentação baseados em cor, particularmente àqueles que operam no espaço HSV. Ainda assim, o sistema demonstrou excelente desempenho, adaptando-se com precisão às tonalidades atípicas sem comprometer a qualidade da segmentação ou a acurácia da classificação.

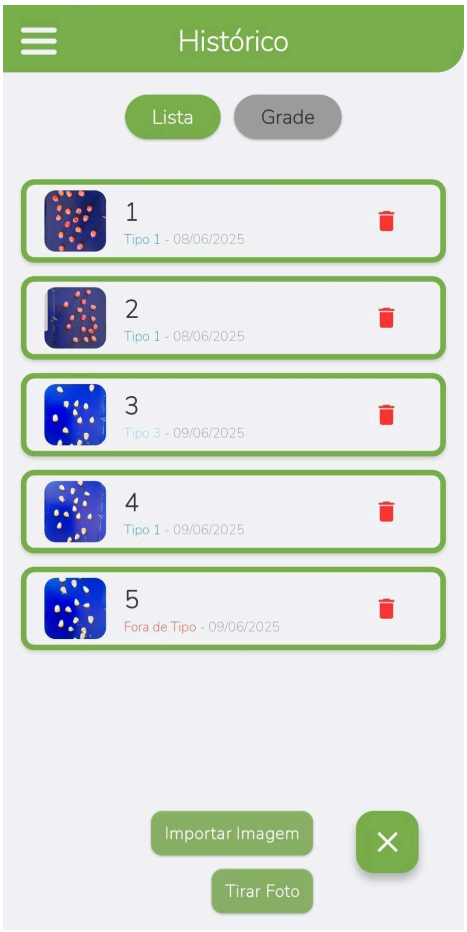
Esse resultado evidencia a capacidade de generalização do *pipeline* diante de variações cromáticas, aspecto essencial para aplicações em ambientes reais e não controlados. A habilidade de identificar corretamente grãos tratados, mesmo com coloração artificial, demonstra que o modelo não se apoia exclusivamente em tonalidades específicas, mas sim em um conjunto integrado de características visuais para a tomada de decisão. Dessa forma, o teste reforça a eficácia da abordagem proposta, mesmo em amostras com alta densidade de grãos e presença de variações de cor significativas.

5.5 TESTE DO APLICATIVO

Com o aplicativo finalizado e devidamente integrado ao *pipeline* de processamento em nuvem, foram realizados testes com o objetivo de avaliar sua funcionalidade, usabilidade e desempenho geral. Essa etapa buscou verificar a estabilidade da aplicação, a fluidez na navegação, o tempo de resposta na entrega dos resultados e a consistência das análises realizadas. A seguir, são descritas as principais etapas da interação do usuário com o ZeaScan, bem como os resultados observados durante os testes práticos.

O processo se inicia na tela principal do aplicativo. Ao selecionar o botão “+”, localizado no canto inferior direito da interface — conforme ilustrado na Figura 29 — o usuário tem acesso a duas opções: importar uma imagem armazenada no dispositivo (ou em serviço de nuvem vinculado) ou capturar uma nova fotografia utilizando a câmera do *smartphone*.

Figura 29 – Tela inicial do aplicativo com acesso às opções de captura ou importação de imagem para nova análise



Fonte: Elaboração do autor.

Após a seleção da imagem, o sistema direciona o usuário para uma tela de confirmação, permitindo que ele aceite ou recuse a imagem antes de prosseguir com a análise. Essa etapa intermediária visa garantir a qualidade da imagem submetida ao processamento, promovendo maior controle sobre os dados de entrada. A Figura 30 ilustra essa interface de confirmação, na qual o usuário pode optar por validar ou substituir a imagem capturada.

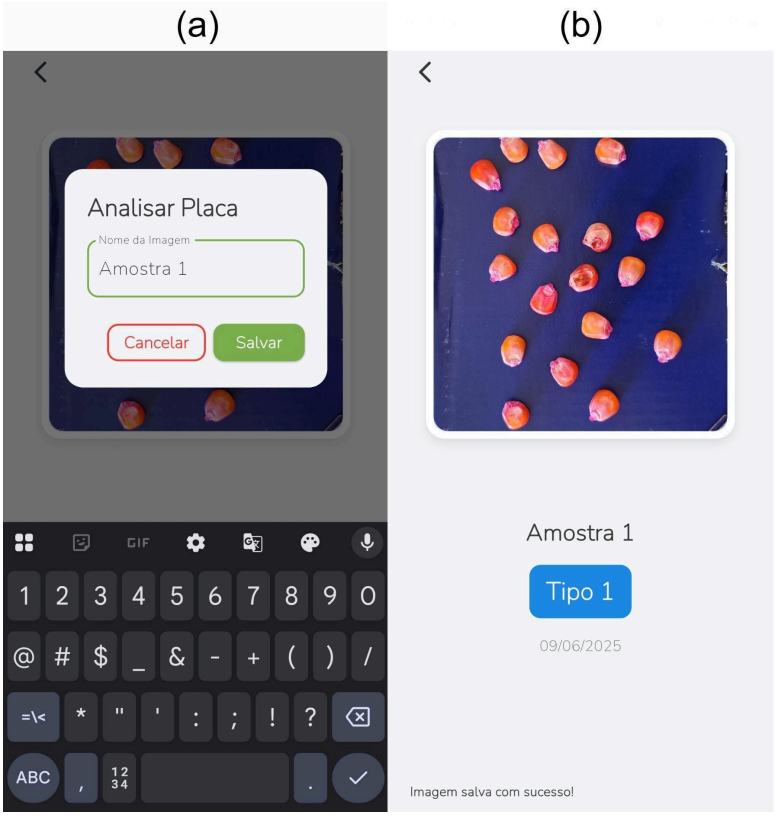
Figura 30 – Tela de confirmação da imagem selecionada ou capturada para análise

Fonte: Elaboração do autor.

Após a confirmação da imagem, o usuário é solicitado a inserir um nome identificador para a amostra. Em seguida, o aplicativo realiza o envio da imagem para processamento em nuvem, utilizando o *pipeline* previamente desenvolvido para segmentação e classificação. Em poucos segundos, o resultado da tipificação é retornado e exibido de forma clara na tela, sendo também automaticamente registrado no histórico inicial para consultas futuras.

A Figura 31 apresenta essas duas etapas finais do processo: a inserção do nome da amostra e a exibição do resultado da tipificação na interface do aplicativo.

Figura 31 – Etapas de (a) nomeação da amostra e (b) exibição do resultado da tipificação



Fonte: Elaboração do autor.

Durante os testes de uso, o ZeaScan apresentou desempenho eficiente e alinhado aos objetivos propostos pelo projeto. O aplicativo manteve estabilidade nas interações, forneceu retorno ágil dos resultados e demonstrou elevada consistência entre os dados processados localmente e as classificações obtidas via nuvem.

A clareza das informações, a fluidez na navegação e a precisão das análises confirmaram a eficácia da solução como ferramenta prática para a tipificação automatizada de grãos de milho, consolidando seu potencial como apoio à tomada de decisão em processos de controle de qualidade e classificação de lotes.

5.6 COMPARAÇÃO COM TRABALHOS RELACIONADOS

A síntese comparativa entre o método proposto, denominado ZeaScan, e os trabalhos relacionados é apresentada no Quadro 7. Nela destacam-se a técnica de classificação utilizada, o número de classes consideradas, as categorias analisadas, a acurácia global obtida e o tamanho dos conjuntos de dados empregados para treinamento e teste.

Quadro 7 - Síntese comparativa entre os trabalhos relacionados

Estudo	Abordagem	Número de Classes	Classes	Acurácia Global	*Dataset* de treinamento	*Dataset* de teste
Rosin (2019b)	RF, KNN e SVM	3	Ardidos, Carunchados e Sadios	83,00%	3.037	1.302
Wu *et al.* (2018)	SVM-GA/PSO	3	Grau A, Grau B e Grau C	97,44%	90	39
Velesaca *et al.* (2020)	CK-CNN	2	Defeituosos e Sadios	94,50%	16.500	360
		3	Defeituosos, Sadios e Impurezas	95,60%	22.500	2.100
Suárez *et al.* (2023)	CK-CNNLW	2	Defeituosos e Sadios	96,50%	3.350	360
		3	Defeituosos, Sadios e Impurezas	96,70%	8.600	2.060
Modelo proposto	CNN	3	Ardidos, Carunchados e Sadios	97,36%	4.581	719
	CNN	2	Defeituosos e Sadios	97,50%	4.581	719

Fonte: Elaboração do autor.

Conforme apresentado no Quadro 7, este trabalho (ZeaScan) destaca-se de maneira expressiva em relação aos demais trabalhos da área. Na tarefa de classificação triclasse (ardidos, carunchados e sadios), a abordagem baseada em CNN atingiu uma acurácia global de 97,36%, superando em 14,36% o resultado obtido por Rosin *et al.* (83,0%) e apresentando robustez estatística superior à de Wu *et al.* (97,44%), cujo conjunto de dados reduzido (129 amostras) limita a confiabilidade e abrangência dos achados.

Enquanto o método de Wu *et al.* prioriza atributos físico-composicionais, como tamanho e umidade, o modelo proposto concentra-se em defeitos biológicos de alta relevância para a agricultura tropical, como grãos ardidos (indicadores de contaminações fúngicas) e carunchados (danos provocados por insetos).

Na avaliação binária (defeituosos *versus* sadios), o ZeaScan manteve uma performance destacada, com acurácia de 97,50%, superando Velesaca *et al.* (94,50%) e Suárez *et al.* (96,50%). Ressalta-se que, enquanto esses trabalhos tendem a agregar todos os defeitos em uma única classe ou requerem *pipelines* complexos com dados sintéticos para alcançar uma segmentação eficaz, o modelo

proposto opera de maneira nativa com três categorias distintas, ampliando a granularidade e a capacidade de distinguir diferentes tipos de danos.

Tal capacidade reflete-se particularmente no *F1-score* obtido para grãos carunchados (48,28%), uma classe para a qual Rosin *et al.* não apresentaram sucesso (*F1-score* = 0%), destacando a vantagem do método proposto ao realizar extração automática de características visuais com CNN em comparação às abordagens tradicionais, baseadas em atributos manuais.

O modelo proposto não oferece apenas uma classificação mais precisa, mas contribui para uma identificação mais detalhada e confiável dos danos em grãos de milho, ampliando suas possibilidades de aplicação prática no controle de qualidade e seleção de amostras.

5.7 CONTRIBUIÇÕES DO TRABALHO

A principal contribuição deste trabalho foi o desenvolvimento do aplicativo ZeaScan, que incorpora um modelo de classificação automática de grãos de milho utilizando Redes Neurais Convolucionais. O sistema foi projetado para atender às diretrizes da Instrução Normativa nº 60/2011 do Ministério da Agricultura, Pecuária e Abastecimento (MAPA).

Outra contribuição relevante foi o registro oficial do *software* ZeaScan, assegurando a proteção da propriedade intelectual da solução proposta e estabelecendo as bases para sua aplicação em cenários de inspeção e controle de qualidade em processos produtivos.

Além disso, destaca-se a submissão do artigo científico “*Classification of Corn Kernels Using Convolutional Neural Networks*” à revista *Biocatalysis and Agricultural Biotechnology*, que promove a divulgação dos resultados desta pesquisa à comunidade científica internacional e insere o estudo no contexto das aplicações de aprendizado profundo voltadas à agricultura de precisão.

6 CONCLUSÃO

O desenvolvimento da ferramenta computacional ZeaScan, baseada em redes neurais convolucionais (CNNs), revelou-se uma solução eficaz para a classificação automatizada de grãos de milho, com alto potencial de impacto nos processos de controle de qualidade do setor agrícola. Os resultados obtidos demonstraram a robustez do modelo na distinção entre grãos sadios e defeituosos, sobretudo em contextos que exigem triagens rápidas e confiáveis. A aplicação de técnicas de pré-processamento mostrou-se fundamental para evidenciar características morfológicas e superficiais, otimizando a capacidade da rede neural em extrair padrões discriminativos relevantes.

Embora o desempenho tenha sido satisfatório na maioria das categorias, foram identificados desafios na detecção precisa de grãos com danos específicos, como os provocados por insetos. Tais dificuldades indicam a influência de fatores como o desbalanceamento entre classes e a similaridade visual entre determinados tipos de defeitos. Essas limitações reforçam a necessidade de estratégias complementares, como a ampliação do conjunto de dados e a aplicação de técnicas avançadas de aumento de amostras, para promover maior capacidade de generalização do modelo. Ainda assim, nos testes práticos de segmentação e classificação binária, o sistema demonstrou consistência nos resultados, validando sua aplicabilidade em ambientes industriais que demandam agilidade e menor subjetividade nas decisões.

A integração do modelo em uma plataforma móvel, equipada com funcionalidades de processamento em tempo real e interface amigável, evidenciou seu potencial de escalabilidade e aplicação direta em ambientes de campo. A segmentação precisa dos grãos, combinada à classificação automatizada conforme os critérios estabelecidos por normativas técnicas, favorece a padronização dos procedimentos e mitiga o risco de inconsistências, elevando a confiabilidade nas avaliações de qualidade. Apesar da necessidade de ajustes pontuais para aprimorar a detecção em situações-limite, os testes realizados confirmaram a viabilidade e o caráter promissor da abordagem proposta.

O modelo fundamentado em redes neurais convolucionais demonstrou superioridade em relação a abordagens anteriores ao superar limitações críticas, como a baixa acurácia na detecção de grãos danificados por pragas e a

dependência de procedimentos manuais ou *pipelines* excessivamente complexos. Ainda que sejam necessários ajustes contínuos para aprimorar a identificação de certos tipos de defeitos, a solução se sobressai pela capacidade de generalização robusta e pela aplicabilidade prática em cenários que exigem automação, simplicidade operacional e agilidade nos processos agrícolas.

Em síntese, este trabalho avançou na aplicação de técnicas de visão computacional para a otimização da cadeia produtiva do milho, ao propor uma solução inovadora que combina elevada precisão técnica com praticidade operacional, favorecendo sua adoção em ambientes agrícolas reais.

REFERÊNCIAS

ADEWUYI, A. Y.; ANYIBAMA, B.; ADEBAYO, K. B.; KALINZI, J. M.; ADENIYI, S. A.; WADA, I. Precision agriculture: leveraging data science for sustainable farming. *International Journal of Science and Research Archive*, [S. l.], v. 12, n. 2, p. 1122-1129, 2024. DOI: https://doi.org/10.30574/ijsra.2024.12.2.1371.

AHMAD, S. I.; RANA, T.; MAQBOOL, A. A model-driven *framework* for the development of MVC-based (web) application. *Arabian Journal for Science and Engineering*, v. 47, p. 1733–1747, 2022. DOI: https://doi.org/10.1007/s13369-021-06087-4.

ALESSANDRIA, S. *Flutter* projects: a practical, project-based guide to building real-world cross-platform mobile applications and games. [S. l.]: *Packt Publishing*, 2020. 490 p. ISBN 978-1838647773.

ALKANAN, M.; GULZAR, Y. Enhanced corn seed disease classification: leveraging MobileNetV2 with feature augmentation and transfer learning. *Frontiers in Applied Mathematics and Statistics*, Lausanne, v. 9, 2023. DOI: https://doi.org/10.3389/fams.2023.1320177.

ALOMAR, K.; AYSEL, H. I.; CAI, X. Data augmentation in classification and segmentation: a survey and new strategies. *Journal of Imaging*, v. 9, n. 2, p. 46, 2023. DOI: https://doi.org/10.3390/jimaging9020046.

ALOUFFI, B.; HASNAIN, M.; ALHARBI, A.; ALOSAIMI, W.; ALYAMI, H.; AYAZ, M. A systematic literature review on *cloud computing* security: threats and mitigation strategies. *IEEE Access*, v. 9, p. 57792–57807, 2021. Disponível em: https://doi.org/10.1109/ACCESS.2021.3073203.

ARCHANA, R.; JEEVARAJ, P. S. E. Deep learning models for digital image processing: a review. *Artificial Intelligence Review*, v. 57, n. 1, p. 11, 2024. DOI: https://doi.org/10.1007/s10462-023-10631-z.

AZEEM, M.; KIANI, K.; MANSOURI, T.; TOPPING, N. SkinLesNet: classification of skin lesions and detection of melanoma cancer using a novel multi-layer deep convolutional neural network. *Cancers*, [S. l.], v. 16, n. 1, p. 108, 2024. DOI: https://doi.org/10.3390/cancers16010108.

AZEVEDO, E.; CONCI, A.; LETA, F. Computação gráfica: teoria e prática: geração de imagens (Volume 2). [S. l.]: Alta Books, 2022. 352 p. ISBN 978-6555208160.

BADUGE, S. K.; THILAKARATHNA, S.; PERERA, J. S.; JAYAWICKRAMA, B. A.; GOSLING, J. P. Artificial intelligence and smart vision for building and construction 4.0: Machine and *deep learning* methods and applications. *Automation in Construction*, v. 141, p. 104440, 2022. DOI: http://doi.org/10.1016/j.autcon.2022.104440

BARROS, P. H. B. de; FREITAS JUNIOR, A. M. de. Combinando inteligência artificial e imagens de satélite para a previsão de sinistros agrícolas: uma nota. *Revista Brasileira de Economia*, v. 77, e012023, 2023. DOI: https://doi.org/10.5935/0034-7140.20230001.

BAZAME, H. C.; MOLIN, J. P.; ALTHOFF, D.; MARTELLO, M. Detection of coffee fruits on tree branches using computer vision. *Scientia Agricola*, v. 80, e20220064, 2023. DOI: https://doi.org/10.1590/1678-992X-2022-0064.

BRASIL. Lei nº 9.972, de 25 de maio de 2000. Institui o Sistema de Classificação de Produtos Vegetais, seus subprodutos e resíduos de valor econômico, e dá outras providências. *Diário Oficial da União*: seção 1, Brasília, DF, 26 maio 2000. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/L9972.htm. Acesso em: 19 jun. 2024.

BRASIL. Lei nº 10.711, de 5 de agosto de 2003. Dispõe sobre o Sistema Nacional de Sementes e Mudas e dá outras providências. *Diário Oficial da União*: seção 1, Brasília, DF, 6 ago. 2003. Disponível em: https://www.planalto.gov.br/ccivil_03/leis/2003/L10.711.htm. Acesso em: 19 jun. 2024.

BRASIL. Ministério da Agricultura, Pecuária e Abastecimento. Instrução Normativa nº 60, de 22 de dezembro de 2011. Estabelece o Regulamento Técnico do Milho. *Diário Oficial da União*: seção 1, Brasília, DF, 23 dez. 2011. Disponível em: https://sistemasweb.agricultura.gov.br/sislegis/action/detalhaAto.do?method=visualizarAtoPortalMapa&chave=1739574738. Acesso em: 19 jun. 2024.

BRASIL. Ministério da Agricultura, Pecuária e Abastecimento (MAPA). Secretaria de Comércio e Relações Internacionais. Exportações Brasileiras – MILHO. Brasília, DF, 2024. Disponível em: https://www.gov.br/agricultura/pt-br/assuntos/relacoes-internacionais/documentos/Milho.pdf. Acesso em: 17 fev. 2025.

CARVALHO, F.; RODRIGUES, L. A. Um sistema de informação para monitoramento de qualidade e estimativa de perdas em instalações de armazenamento de grãos usando dados de sensores de IoT e outros mecanismos. *Revista Brasileira de Computação Aplicada*, v. 15, n. 1, p. 12-21, 2023. DOI: https://doi.org/10.5335/rbca.v15i1.13778.

CAVALCANTI NETO, E.; REBOUÇAS, E. S.; MORAES, J. L.; GOMES, S. L.; REBOUÇAS FILHO, P. P. Development control parking access using techniques digital image processing and applied computational intelligence. *IEEE Latin America Transactions*, v. 13, n. 1, p. 1-7, jan. 2015. DOI: https://doi.org/10.1109/TLA.2015.7040658.

CHADEBECQ, F.; VASCONCELOS, F.; MAZOMENOS, E.; STOYANOV, D. Computer vision in the surgical operating room. *Visceral Medicine*, v. 36, n. 6, p. 456-460, dez. 2020. DOI: https://doi.org/10.1159/000511934.

CHEN, J.; CHEN, J.; ZHANG, D.; SUN, Y.; NANEHKARAN, Y. A. Using deep transfer learning for image-based plant disease identification. *Computers and Electronics in Agriculture*, v. 173, p. 105393, 2020. DOI: https://doi.org/10.1016/j.compag.2020.105393.

CHICCO, D.; JURMAN, G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, v. 21, n. 6, 2020. DOI: https://doi.org/10.1186/s12864-019-6413-7

CHOMA, D.; CHWALEBA, K.; DZIEŃKOWSKI, M. The efficiency and reliability of *backend* technologies: Express, Django, and Spring Boot. *Informatyka, Automatyka, Pomiary w Gospodarce i Ochronie Środowiska*, [S. l.], v. 13, n. 4, p. 73–78, 2023. DOI: https://ph.pollub.pl/index.php/iapgos/article/view/4279.

COSTA, L. S.; BARBOSA, S. B.; CUNHA, J. P. A *backend* platform for supporting the reproducibility of computational experiments. *arXiv, v. 2308.00703*, 2023. DOI: https://doi.org/10.48550/arXiv.2308.00703.

DA SILVA, H.; SANTOS, L. L. A contribuição da Tecnologia da Informação na Agricultura para o Desenvolvimento Sustentável. *Ciência & Trópico*, [S. l.], v. 48, n. 2, 2024. DOI: https://doi.org/10.33148/CETROPv48n2(2024)2313.

DAI, X.; XIAO, Z.; JIANG, H.; LUI, J. C. S. UAV-assisted task offloading in vehicular *edge computing* networks. *IEEE Transactions on Mobile Computing*, v. 23, n. 4, p. 2520–2534, 2024. Disponível em: https://doi.org/10.1109/TMC.2023.3259394.

DE MATOS, N. A. V.; SANTOS, L. R.; MALLMANN, C. A.; OLIVEIRA, C. A. F.; CORASSA, L. B. A survey on free and hidden fumonisins in Brazilian corn and corn-based products. Food Control, v. 156, p. 110135, 2024. DOI: https://doi.org/10.1016/j.foodcont.2023.110135.

DEVECHIO, F. F. S.; LUZ, P. H. C.; ROMUALDO, L. M.; BAESSO, M. M.; SILVA, T. L.; HERLING, V. R. Análise de imagens digitais para identificação do estado nutricional de nitrogênio em híbridos de milho cultivados no campo: uso de análise de imagens para detectar deficiências nutricionais em milho cultivado em campo submetido à omissão de nitrogênio. Brazilian Journal of Animal and Environmental Research, Curitiba, v. 6, n. 1, p. 177-188, jan./mar. 2023. DOI: https://doi.org/10.34188/bjaerv6n1-016.

DINGA, R.; PENNINX, B. W. J. H.; VELTMAN, D. J.; SCHMAAL, L.; MARQUAND, A. F. Beyond accuracy: measures for assessing *machine learning* models, pitfalls and guidelines. *bioRxiv*, [s. l.], 2019. Preprint. DOI: https://doi.org/10.1101/743138.

DONG, C.-Z.; CATBAS, F. N. A review of computer vision-based structural health monitoring at local and global levels. *Structural Health Monitoring*, v. 20, n. 2, p. 692-743, 2021. DOI: https://doi.org/10.1177/1475921720935585.

ELI-CHUKWU, N. C. Applications of artificial intelligence in agriculture: a review. *Engineering, Technology & Applied Science Research*, Grécia, v. 9, n. 4, p. 4377-4383, ago. 2019. DOI: https://doi.org/10.48084/etasr.2756.

FROTA, R. P.; SOUZA, F. C.; FERREIRA, D. S.; RIPARDO, V. E.; ANDRADE, F. J. E. T.; CÉSAR, L. T.; MOREIRA, R. A. F.; FARIAS, M. D. P. Aplicação de redes neurais convolucionais para treinamento de inteligência artificial na avaliação de embutidos de pescados. *Revista Observatorio de la Economía Latinoamericana*, [S. l.], v. 23, n. 1, 2025. DOI: https://doi.org/10.55905/oelv23n1-045.

FU, X.; QI, Q.; ZHA, Z.-J.; ZHU, Y.; DING, X. Rain streak removal via dual graph convolutional network. *Proceedings of the AAAI Conference on Artificial Intelligence*, v. 35, n. 2, p. 1352–1360, 2021. DOI: https://doi.org/10.1609/aaai.v35i2.16224.

GABRIEL FILHO, O. Inteligência artificial e aprendizagem de máquina: aspectos teóricos e aplicações. São Paulo: Blucher, 2023. 462 p.

GAYATRI, K.; KANTI, R. D.; RAYAVARAPU, V. S. R.; SRIDHAR, B.; BOBBILI, V. R. G. Image processing and pattern recognition based plant leaf diseases identification and classification. In: *JOURNAL OF PHYSICS: CONFERENCE SERIES*, v. 1804, 2021. *Anais* [...]. [S. l.]: IOP Publishing, 2021. p. 012160. DOI: https://doi.org/10.1088/1742-6596/1804/1/012160.

GENG, Q.; ZHOU, Z.; CAO, X. Survey of recent progress in semantic image segmentation with CNNs. *Science China Information Sciences*, v. 61, p. 051101, 2018. DOI: https://doi.org/10.1007/s11432-017-9189-6.

GUIMARÃES, A. J. S.; BARBOSA, D. S.; SOUSA, G. da S.; SANTOS, A. J. de S. Ação de redes neurais convolucionais na classificação de imagens: abordagens e resultados. *Revista Contemporânea*, [S. l.], v. 5, n. 1, p. 1-19, 2025. DOI: https://doi.org/10.56083/RCV5N1-012.

HE, M.; ZHAO, Q.; GAO, H.; ZHANG, X.; ZHAO, Q. Image segmentation of a sewer based on deep learning. *Sustainability*, [S. l.], v. 14, n. 11, p. 6634, 2022. DOI: https://doi.org/10.3390/su14116634.

HELFER, G. A.; BARBOSA, J.; SANTOS, R. B.; COSTA, A. A computational model for soil fertility prediction in ubiquitous agriculture. *Computers and Electronics in Agriculture*, [S. l.], v. 175, n. 4, p. 105602, ago. 2020. DOI: https://doi.org/10.1016/j.compag.2020.105602.

HONG, J. B.; NHLABATSI, A.; KIM, D. S.; HUSSEIN, A.; FETAIS, N.; KHAN, K. M. Systematic identification of threats in the *cloud*: a survey. Computer Networks, v. 150, p. 46–69, 26 fev. 2019. DOI: https://doi.org/10.1016/j.comnet.2018.12.009.

JADON, S. A survey of loss functions for semantic segmentation. In: *Ieee Conference on Computational Intelligence in Bioinformatics and Computational Biology* (CIBCB), 2020. *Anais...* [S. l.]: IEEE, 2020. p. 1–7. Disponível em: https://doi.org/10.1109/CIBCB48159.2020.9277638.

JAIN, A. K.; RAO, P. P. R. D.; SHARMA, K. V. A perspective analysis of regularization and optimization techniques in machine learning. In: *COMPUTATIONAL analysis and deep learning for medical care*. [S. l.]: [s. n.], 2021. p. 393–427. DOI: https://doi.org/10.1002/9781119785750.ch16.

KAMILARIS, A.; PRENAFETA-BOLDÚ, F. X. Deep learning in agriculture: a survey. *Computers and Electronics in Agriculture*, [S. l.], v. 147, p. 70-90, abr. 2018. DOI: https://doi.org/10.1016/j.compag.2018.02.016.

KARIĆ, A.; DURMIĆ, N. Comparison of JavaScript *frontend frameworks* – Angular, React, and Vue. *International Journal of Innovative Science and Research Technology*, v. 9, n. 6, 2024. DOI: https://doi.org/10.38124/ijisrt/IJISRT24JUN600.

KATTENBORN, T.; LEITLOFF, J.; SCHIEFER, F.; HINZ, S. Review on Convolutional Neural Networks (CNN) in vegetation remote sensing. *ISPRS Journal of Photogrammetry and Remote Sensing*, [S. l.], v. 173, p. 24-49, mar. 2021. DOI: https://doi.org/10.1016/j.isprsjprs.2020.12.010.

KEARNS, M.; ROTH, A. The ethical algorithm: the science of socially aware algorithm design. New York: Oxford University Press, 2020.

KHALID, S.; OQAIBI, H. M.; AQIB, M.; HAFEEZ, Y. Small pests detection in field crops using *deep learning* object detection. *Sustainability*, v. 15, n. 8, p. 6815, 2023. DOI: https://doi.org/10.3390/su15086815.

LI, C.; CHEN, Z.; JING, W.; WU, X.; ZHAO, Y. A lightweight method for maize seed defects identification based on convolutional block attention module. *Frontiers in Plant Science*, Lausanne, v. 14, 2023. DOI: https://doi.org/10.3389/fpls.2023.1153226.

LI, X.; DAI, B.; SUN, H.; LI, W. Corn classification system based on computer vision. *Symmetry*, Basel, v. 11, n. 4, p. 591, abr. 2019. DOI: https://doi.org/10.3390/sym11040591.

LI, Z.; LIU, F.; YANG, W.; PENG, S.; ZHOU, J. A survey of convolutional neural networks: analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, [S. l.], v. 33, n. 12, p. 6999-7019, dez. 2022. DOI: https://doi.org/10.1109/TNNLS.2021.3084827.

LIU, Y.; PENG, M.; SHOU, G.; CHEN, Y.; CHEN, S. Toward *edge* intelligence: multiaccess *edge computing* for 5G and Internet of Things. *IEEE Internet of Things Journal*, v. 7, n. 8, p. 6722–6747, 2020. DOI: https://doi.org/10.1109/JIOT.2020.3004500.

LIU, Y.; YANG, C.; ZHAO, T.; HAN, H.; ZHANG, S.; WU, J.; ZHOU, G.; HUANG, H.; WANG, H.; SHI, C. GammaGL: a multi-*backend* library for graph neural networks. In: *International ACM SIGIR Conference on Research And Development In Information Retrieval*, 46., 2023, Taipei, Taiwan. Proceedings... New York, NY, USA: Association for Computing Machinery, 2023. p. 2861–2870. DOI: https://doi.org/10.1145/3539618.3591891.

LIYANAGE, M.; PORAMBAGE, P.; DING, A. Y.; KALLA, A. Driving forces for multi-access *edge computing* (MEC) IoT integration in 5G. *ICT Express*, [S. l.], v. 7, n. 2, p. 127-137, jun. 2021. DOI: https://doi.org/10.1016/j.icte.2021.05.007.

LU, S.; LU, J.; AN, K.; WANG, X.; HE, Q. Edge computing on IoT for machine signal processing and fault diagnosis: a review. *IEEE Internet of Things Journal*, v. 10, n. 13, p. 11093–11116, 2023. Disponível em: https://doi.org/10.1109/JIOT.2023.3239944.

LUDEMIR, T. B. Inteligência Artificial e aprendizado de máquina: estado atual e tendências. *Estudos Avançados*, v. 35, n. 101, p. 85-94, 2021. DOI: https://doi.org/10.1590/s0103-4014.2021.35101.007.

MANAVALAN, R. Automatic identification of diseases in grain crops using computational approaches: a review. *Computers and Electronics in Agriculture*, [S. l.], v. 178, p. 105802, nov. 2020. DOI: https://doi.org/10.1016/j.compag.2020.105802.

MAO, A.; MOHRI, M.; ZHONG, Y. Cross-Entropy Loss Functions: Theoretical Analysis and Applications. *Proceedings of the 40th International Conference on Machine Learning*. [S. l.]: PMLR, 2023. DOI: https://doi.org/10.48550/arXiv.2304.07288.

MARIANO, D. Métricas de avaliação em *machine learning*: acurácia, sensibilidade, precisão, especificidade e F-score. *BIOINFO – Revista Brasileira de Bioinformática*, [S. l.], n. 1, jul. 2021. DOI: https://doi.org/10.51780/978-6-599-275326-15.

MARIN, I.; SKELIN, A. K.; GRUJIC, T. Empirical evaluation of the effect of optimization and regularization techniques on the generalization performance of deep convolutional neural network. *Applied Sciences*, Basel, v. 10, n. 21, p. 1–30, 4 nov. 2020. DOI: https://doi.org/10.3390/app10217817.

MARQUES JUNIOR, L. C.; COVOLAN, J. A. U. Aplicação de redes neurais profundas para detecção e classificação de plantas daninhas: seu estado da arte. *REGRAD*, Marília-SP, v. 11, n. 1, p. 391-403, ago. 2018. Disponível em: https://revista.univem.edu.br/REGRAD/article/view/2638/743. Acesso em: 23 abr. 2025.

MARQUES JUNIOR, L. C.; ULSON, J. A. C. Real time weed detection using computer vision and deep learning. In: *IEEE International Conference on Industry Applications* (INDUSCON), 14., 2021. Anais [...]. [S. l.], 2021. p. 1131-1137. DOI: https://doi.org/10.1109/INDUSCON51756.2021.9529761.

MASSRUHÁ, S. M. F. S.; LEITE, M. A. de A.; OLIVEIRA, S. R. de M.; MEIRA, C. A. A.; LUCHIARI JUNIOR, A.; BOLFE, E. L. (Eds.). Agricultura digital: pesquisa, desenvolvimento e inovação nas cadeias produtivas. Brasília, DF: Embrapa, 2020. 406 p. ISBN 978-65-86056-37-2.

MATOS, M. V. R.; MAMEDES, A. de S.; SILVA, T. T. da; CAMPANHA, M. M.; MATRANGOLO, W. J. R.; PIMENTEL, M. A. G. Qualidade de grãos de milho em sistema orgânico de produção na região central de Minas Gerais. In: CONFERÊNCIA BRASILEIRA DE PÓS-COLHEITA, 8., 2023, Rio Verde, GO. *Anais*... Londrina: Associação Brasileira e Pós-Colheita, 2023. p. 122-128. Disponível em: http://www.alice.cnptia.embrapa.br/alice/handle/doc/1158043. Acesso em: 22 jan. 2025.

MAURI, G. S.; GERLACH, G. A. X.; CATALANI, G. C.; CRUSCIOL, G. C. D. Produtividade da cultura do milho em função da adubação de cobertura. *Revista Científica Unilago*, [S. l.], v. 1, n. 1, 2022. Disponível em: https://revistas.unilago.edu.br/index.php/revista-cientifica/article/view/796. Acesso em: 20 fev. 2025.

MISHRA, S.; SACHAN, R.; RAJPAL, D. Deep convolutional neural network based detection system for real-time corn plant disease recognition. *Procedia Computer Science*, v. 167, p. 2003-2010, 2020. DOI: https://doi.org/10.1016/j.procs.2020.03.236.

MUFID, M. R.; BASOFI, A.; AL RASYID, M. U. H.; ROCHIMANSYAH, I. F.; ROKHIM, A. Design an MVC model using Python for Flask *framework* development. In: *International Electronics Symposium (IES)*, p. 214–219, 2019. DOI: 10.1109/ELECSYM.2019.8901656.

MYLLYAHO, L.; RAATIKAINEN, M.; MÄNNISTÖ, T.; MIKKONEN, T.; NURMINEN, J. K. Systematic literature review of validation methods for AI systems. *Journal of Systems and Software*, v. 181, p. 111050, nov. 2021. DOI: https://doi.org/10.1016/j.jss.2021.111050.

NARIN, A.; KAYA, C.; PAMUK, Z. Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks. *Pattern Analysis and Applications*, v. 24, n. 3, p. 1207-1220, 2021. DOI: https://doi.org/10.1007/s10044-021-00984-y.

NEETHIRAJAN, S. The use of artificial intelligence in assessing affective states in livestock. *Frontiers in Veterinary Science*, v. 8, p. 715261, 2 ago. 2021. DOI: https://doi.org/10.3389/fvets.2021.715261

NICOLAU, I. da S.; COELHO, S. de S. Inteligência artificial no setor elétrico: monitoramento da qualidade da energia por meio de redes neurais. *Revista FT*, v. 29, n. 140, 2024. DOI: https://doi.org/10.69849/revistaft/ra10202411142331.

NOVAC, O. C.; NOVAC, M. C.; OPROESCU, M.; MLINARCIC, A. C.; GORDAN, C. E.; DINDELEGAN, C. M. Employing comparative study between *frontend frameworks*: React vs Ember vs Svelte. In: *International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, p. 1–5., 2024. DOI: https://doi.org/10.1109/ECAI61503.2024.10607455.

OLIVEIRA, A. K. S.; LIMA, E. C. S.; CARIDADE, E. R. S. O uso dos framework Flutter e Nest.js para desenvolvimento de um frontend e um back-end de uma aplicação de help desk. *Revista Ibero-Americana de Humanidades, Ciências e Educação*, São Paulo, v. 8, n. 11, p. 1766–1786, nov. 2022. Disponível em: https://doi.org/10.51891/rease.v8i11.7771.

OLIVEIRA, D. P.; ATAIDES, G. M.; BARBOSA, U. C.; BERGLAND, A. C. R. O.; OLIVEIRA, D. C.; OLIVEIRA, D. E. C.; FURQUIM, M. G. D.; SOUSA JÚNIOR, J. C. Análise e modelagem gradual de um aplicativo para classificação de grãos de soja, milho, feijão e sorgo. *Research, Society and Development*, [S. l.], v. 10, n. 5, e34310515040, 2021. DOI: https://doi.org/10.33448/rsd-v10i5.15040.

OTTER, D. W.; MEDINA, J. R.; KALITA, J. K. A survey of the usages of *deep learning* for natural language processing. *IEEE Transactions on Neural Networks and Learning Systems*, v. 32, n. 2, p. 604–624, 2021. Disponível em: https://doi.org/10.1109/TNNLS.2020.2979670.

PANTAZI, X. E.; MOSHOU, D.; BOCHTIS, D. Sensors in agriculture. In: PANTAZI, X. E.; MOSHOU, D.; BOCHTIS, D. (Ed.). *Intelligent data mining and fusion systems in agriculture*. 1. ed. San Diego: Academic Press, 2019. p. 1-15. ISBN 978-0-12-814391-9.

PAOLONE, G.; MARINELLI, M.; PAESANI, R.; DI FELICE, P. Automatic code generation of MVC web applications. *Computers*, v. 9, n. 3, p. 56, 2020. DOI: https://doi.org/10.3390/computers9030056.

PARTEL, V.; KAKARLA, S. C.; AMPATZIDIS, Y. Development and evaluation of a low-cost and smart technology for precision weed management utilizing artificial intelligence. *Computers and Electronics in Agriculture*, v. 157, p. 339-350, fev. 2019. DOI: https://doi.org/10.1016/j.compag.2018.12.048.

PATEL, H.; SAMAD, A.; HAMZA, M.; MUAZZAM, A.; HARAHAP, M. K. Role of artificial intelligence in livestock and poultry farming. *Sinkron: Jurnal dan Penelitian Teknik Informatika*, [S. l.], v. 7, n. 4, p. 2425-2429, out. 2022. DOI: https://doi.org/10.33395/sinkron.v7i4.11837.

PAULI, P.; KOCH, A.; BERBERICH, J.; KOHLER, P.; ALLGÖWER, F. Training robust neural networks using Lipschitz bounds. *IEEE Control Systems Letters*, v. 6, p. 121–126, 2022. DOI: https://doi.org/10.1109/LCSYS.2021.3050444.

PEREZ, F.; VASCONCELOS, C. *et al.* Data augmentation for skin lesion analysis. In: OR 2.0 Context-aware Operating Theaters, Computer Assisted Robotic Endoscopy, Clinical Image-Based Procedures, and Skin Image Analysis. Cham: *Springer International Publishing*, 2018. p. 303–311. (Lecture Notes in Computer Science, ISSN 1611-3349). DOI: https://doi.org/10.1007/978-3-030-01201-4_33.

PINHEIRO, L. S.; GATTI, V. C. M.; OLIVEIRA, J. T.; SILVA, J. N.; SILVA, V. F. A.; SILVA, P. A. Características agro econômicas do milho: uma revisão. *Natural Resources*, [S. l.], v. 11, n. 2, p. 13-21, 2021. DOI: 10.6008/CBPC2237-9290.2021.002.0003.

RAPÔSO, C. F. L. Desafios e conformidade de *cloud computing* com a LGPD: uma revisão sistemática. *Revista Tópicos*, 2025. DOI: https://doi.org/10.5281/zenodo.14888954.

ROSIN, J. I. F. Imagens do conjunto de dados. [S. l.], 2019a. Disponível em: https://gitlab.com/julianaRosin/*dataset*_image. Acesso em: 7 jan. 2024.

ROSIN, J. I. F. Visão computacional aplicada na classificação de grãos de milho. 2019. 64 f. Trabalho de Conclusão de Curso (Bacharelado em Ciência da Computação) – Universidade Federal da Fronteira Sul, Chapecó, SC, 2019b.

SAEED, F.; ALDERA, S.; ALKHATIB, M.; AL-SHAMMA'A, A. A.; FARH, H. M. H. A data-driven convolutional neural network approach for power quality disturbance signal classification (DeepPQDS-FKTNet). *Mathematics*, v. 11, p. 4726, 2023. DOI: https://doi.org/10.3390/math11234726.

SANGUCHO, J. P. T.; CARRASCO, J. L. O.; RODRÍGUEZ, J. P. G.; GRANJA, D. S. B.; MOYA, S. J. Análisis de *frameworks frontend* para aplicar UX/UI en el desarrollo web: una revisión sistemática. *Ciencia Latina Revista Científica Multidisciplinar*, v. 8, n. 3, p. 841–862, 3 jun. 2024. DOI: https://doi.org/10.37811/cl_rcm.v8i3.11290.

SANTOS, T. T.; SOUZA, L. L. de; SANTOS, A. A. dos; AVILA, S. Grape detection, segmentation, and tracking using deep neural networks and three-dimensional association. *Computers and Electronics in Agriculture*, v. 170, p. 105247, mar. 2020. DOI: https://doi.org/10.1016/j.compag.2020.105247.

SERVIÇO NACIONAL DE APRENDIZAGEM RURAL (SENAR). Grãos: classificação de soja e milho. Brasília: SENAR, 2017. 152 p. *Coleção SENAR*. ISBN 978-85-7664-150-6. Disponível em: https://www.cnabrasil.org.br/assets/arquivos/178-GRÃOS.pdf. Acesso em: 19 jun. 2024.

SHAUKAT, K.; LUO, S.; VARADHARAJAN, V.; HAMEED, I. A.; XU, M. A survey on *machine learning* techniques for cyber security in the last decade. *IEEE Access*, v. 8, p. 222310–222354, 2020. Disponível em: https://doi.org/10.1109/ACCESS.2020.3041951.

SILVA, A. J. J. Metodologia de ajuste de parâmetros para aquisição de imagens de boa qualidade utilizando uma câmera fotográfica semiprofissional comercial. *Revista Sítio Novo*, Palmas, v. 4, n. 4, p. 202-216, 2020. DOI: https://doi.org/10.47236/2594-7036.2020.v4.i4.202-216p.

SILVA, Aline Oliveira da; SILVA, Alasse Oliveira da; GOMES, Jhonatah Albuquerque; OLIVEIRA, Raylan Costa de; SILVA, Diocléa Almeida Seabra; VIÉGAS, Ismael de Jesus Matos. Armazenamento de grãos na agricultura familiar: principais problemáticas e formas de armazenamento na região nordeste paraense. *Research, Society and Development*, [S. l.], v. 10, n. 1, p. e36610111835, 17 jan. 2021. DOI: https://doi.org/10.33448/rsd-v10i1.11835.

SOUZA, C. A. C.; ALMEIDA, M. C.; OLIVEIRA, R. M.; SANTOS, M. M.; CUNHA, C. G.; MORONI, G. C. F.; CALIARI, H. A. O.; KANEKIYO, J. A. Automação robótica: soluções sustentáveis e inclusivas: Tecnoagro. São João da Boa Vista: Centro Universitário de Ensino Octávio Bastos (UNIFEOB), 2024. Disponível em: http://localhost:8080/handle/prefix/7088. Acesso em: 25 jun. 2024.

SOUZA, J. C. S.; QUEIROZ, E. S. Análise dos efeitos econômicos e ambientais da produção de etanol de milho. Engineering Sciences, v. 11, n. 1, p. 18-27, 2023. DOI: https://doi.org/10.6008/CBPC2318-3055.2023.001.0003.

SOUZA, W. C. L.; SILVA, L. G.; SILVA, L. E. B.; SILVA, R. L. V.; LIMA, L. L. C.; BRITO, D. R. Aspectos comparativos entre milho (Zea mays L.) e sorgo (Sorghum bicolor L. Moench): diferenças e semelhanças. *Diversitas Journal*, [S. l.], v. 5, n. 4, p. 2337-2357, 2020. DOI: https://doi.org/10.17648/diversitas-journal-v5i4-891

SRIVASTAVA, N.; HINTON, G.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, v. 15, n. 56, p. 1929–1958, 2014.

SUÁREZ, P. L.; VELESACA, H. O.; CARPIO, D.; SAPPA, A. D. Corn kernel classification from few training samples. *Artificial Intelligence in Agriculture*, v. 9, p. 89-99, 2023. DOI: https://doi.org/10.1016/j.aiia.2023.08.006.

TAVARES, A. R.; RIBEIRO, H. A.; CAMPOS, H. B.; OLIVEIRA, C. S.; SILVA, R. R.; BISSACO, M. A. S.; MARTINI, S. C.; MENEGIDIO, F. B. Visão computacional na saúde: revisão de métodos e desafios educacionais para integração multidisciplinar. *Cuadernos de Educación y Desarrollo*, v. 16, n. 13, 2024. DOI: https://doi.org/10.55905/cuadv16n13-169.

VELESACA, H. O.; MIRA, R.; SUÁREZ, P. L.; LARREA, C. X.; SAPPA, A. D. Deep learning based corn kernel classification. In: *IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION WORKSHOPS* (CVPRW), 2020, Seattle, EUA. *Anais [...]*. [S. l.]: IEEE, 2020. p. 294-302. DOI: https://doi.org/10.1109/CVPRW50498.2020.00041.

VIANA, N. C. R. T.; RIBEIRO, M. M. L. O.; TAVARES, T. D. R.; TAVARES, D. H. R.; CARRER, C. C. Inovação na indústria agrícola brasileira: tecnologias de agricultura de precisão. *International Journal of Scientific Management and Tourism*, Curitiba, v. 10, n. 6, p. 1-19, 2024. DOI: https://doi.org/10.55905/ijsmtv10n6-013.

VILELA JUNIOR, G. B.; LIMA, B. N.; PEREIRA, A. A.; RODRIGUES, M. F.; OLIVEIRA, J. R. L.; SILIO, L. F.; CARVALHO, A. S.; FERREIRA, H. R.; PASSOS, R. P. Determinação das métricas usuais a partir da matriz de confusão de classificadores multiclasses em algoritmos inteligentes nas ciências do movimento humano. *Revista CPAQV - Centro de Pesquisas Avançadas em Qualidade de Vida*, [S. l.], v. 14, n. 2, 2022. DOI: https://doi.org/10.36692/v14n2-01.

VO, H. T.; YU, G.; DANG, T. V.; KIM, J. Late fusion of multimodal deep neural networks for weeds classification. *Computers and Electronics in Agriculture*, v. 175, p. 105506, 2020. DOI: https://doi.org/10.1016/j.compag.2020.105506.

VOULODIMOS, A.; DOULAMIS, N.; DOULAMIS, A.; PROTOPAPADAKIS, E. Deep learning for computer vision: a brief review. *Computational Intelligence and Neuroscience*, v. 2018, Artigo ID 7068349, p. 1-13, 2018. DOI: https://doi.org/10.1155/2018/7068349.

WANG, C.-Y.; LIAO, H.-Y. M.; YEH, I.-H.; WU, Y.-H.; CHEN, P.-Y.; HSIEH, J.-W. CSPNet: A new backbone that can enhance learning capability of CNN. In: *IEEE/CVF CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION* (CVPR), 2020, Washington, DC. Anais [...]. [S. l.]: IEEE, 2020. p. 1571–1580. DOI: https://doi.org/10.48550/arXiv.1911.11929.

WANG, C.-W.; KHALIL, M.-A.; FIRDI, N. P. A survey on *deep learning* for precision oncology. *Diagnostics* (Basel), v. 12, n. 6, p. 1489, 17 jun. 2022. DOI: https://doi.org/10.3390/diagnostics12061489.

WANG, Y.; ZHANG, J.; CAO, Y.; WANG, Z. A deep CNN method for underwater image enhancement. In: *IEEE International Conference on Image Processing (ICIP)*, 2017. *Proceedings…* [S. l.]: IEEE, 2017. p. 1382–1386. DOI: https://doi.org/10.1109/ICIP.2017.8296508.

WARD, A.; CHEN, Y.; JIA, Z.; TAN, K. Explainability in computer vision applied to surgery: bridging the gap between AI and clinical practice. *Artificial Intelligence in Medicine*, v. 58, n. 9, p. 89–104, 2020.

WARD, T. M.; MASCAGNI, P.; BAN, Y.; ROSMAN, G.; PADOY, N.; MEIRELES, O.; HASHIMOTO, D. A. Computer vision in surgery. *Surgery*, [S. l.], v. 169, n. 5, p. 1253-1256, maio 2021. DOI: https://doi.org/10.1016/j.surg.2020.10.039.

WU, A.; ZHU, J.; YANG, Y.; LIU, X.; WANG, X.; WANG, L.; ZHANG, H.; CHEN, J. Classification of corn *kernels* grades using image analysis and support vector machine. *Advances in Mechanical Engineering*, v. 10, n. 12, p. 1-9, 2018. DOI: https://doi.org/10.1177/1687814018817642.

WU, D.; XIA, S.; WANG, Y. Adversarial weight perturbation helps robust generalization. In: CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS (NeurIPS), 34., 2020. *Advances in Neural Information Processing Systems*, v. 33. [S. l.]: Curran Associates, Inc., 2020. DOI: https://doi.org/10.48550/arXiv.2004.05884.

XIAO, J.; FAN, Y.; SUN, R.; WANG, J.; LUO, Z. Stability analysis and generalization bounds of adversarial training. In: CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS (NeurIPS), 36., 2022. *Advances in Neural Information Processing Systems*, v. 35. [S. l.]: Curran Associates, Inc., 2022. DOI: https://doi.org/10.48550/arXiv.2210.00960.

XU, P.; TAN, Q.; ZHANG, Y.; ZHA, X.; YANG, S.; YANG, R. Research on maize seed classification and recognition based on machine vision and deep learning. *Agriculture*, [S. l.], v. 12, n. 2, p. 232, 2022. DOI: https://doi.org/10.3390/agriculture12020232.

YAMASHITA, R.; NISHIO, M.; DO, R. K. G.; TOGASHI, K. Convolutional neural networks: an overview and application in radiology. *Insights into Imaging*, [S. l.], v. 9, n. 4, p. 611-629, 2018. DOI: https://doi.org/10.1007/s13244-018-0639-9.

YANG, H.; ZHANG, J.; DONG, H.; INKAWHICH, N.; GARDNER, A.; TOUCHET, A.; WILKES, W.; BERRY, H.; LI, H. DVERGE: diversifying vulnerabilities for enhanced robust generation of ensembles. In: CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS (NeurIPS), 34., 2020. *Advances in Neural Information Processing Systems*, v. 33. [S. l.]: Curran Associates, Inc., 2020. DOI: https://doi.org/10.48550/arXiv.2009.14720.

YANG, J.; YANG, G. Modified convolutional neural network based on *dropout* and the stochastic gradient descent optimizer. *Algorithms*, [S. l.], v. 11, n. 3, p. 28, 2018. DOI: https://doi.org/10.3390/a11030028.

YOUSEF, M.; HUSSAIN, K. F.; MOHAMMED, U. S. Accurate, data-efficient, unconstrained text recognition with convolutional neural networks. *Pattern Recognition*, v. 108, p. 107482, 2020. DOI: http://doi.org/10.1016/j.patcog.2020.107482.

ZENG, J.; ZHANG, D.; PENG, A.; ZHANG, X.; HE, S.; WANG, Y.; LIU, X.; BI, H.; LI, Y.; CAI, C.; ZHANG, C.; DU, Y.; ZHU, J.-X.; MO, P.; HUANG, Z.; ZENG, Q.; SHI, S.; QIN, X.; YU, Z.; LUO, C.; DING, Y.; LIU, Y.-P.; SHI, R.; WANG, Z.; BORE, S. L.; CHANG, J.; DENG, Z.; DING, Z.; HAN, S.; JIANG, W.; KE, G.; LIU, Z.; LU, D.; MURAOKA, K.; OLIAEI, H.; SINGH, A. K.; QUE, H.; XU, W.; XU, B.; ZHUANG, Y.-B.; DAI, J.; GIESE, T. J.; JIA, W.; XU, B.; YORK, D. M.; ZHANG, L.; WANG, H. DeePMD-kit v3: A multiple-*backend framework* for *machine learning* potentials. *Journal of Chemical Theory and Computation*, v. 21, n. 9, p. 4375–4385, 2025. DOI: https://doi.org/10.1021/acs.jctc.5c00340.

ZHANG, M.; RAMASWAMY, H. G.; AGARWAL, S. Convex calibrated surrogates for the multi-label F-measure. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING (ICML), 37., 2020. *Proceedings of the 37th International Conference on Machine Learning*. [S. l.]: PMLR, 2020. Disponível em: https://doi.org/10.48550/arXiv.2009.07801.

ZHANG, X.; YAO, L.; WANG, X.; MONAGHAN, J.; MCALPINE, D.; ZHANG, Y. A survey on *deep learning*-based non-invasive brain signals: recent advances and new frontiers. [S. l.]: arXiv, V.5, 2020. DOI: https://doi.org/10.48550/arXiv.1905.04149.

ZHENG, W. Current technologies and applications of digital image processing. *Journal of Biomedical and Sustainable Healthcare Applications*, v. 3, n. 1, 2023. DOI: https://doi.org/10.53759/0088/JBSHA202303002.

ZHOU, S.; CHEN, B.; FU, E. S.; YAN, H. Computer vision meets microfluidics: a label-free method for high-throughput cell analysis. *Microsystems & Nanoengineering*, [S. l.], v. 9, art. 116, 2023a. DOI: https://doi.org/10.1038/s41378-023-00562-8.

ZHOU, X.; LIU, X.; ZHAI, D.; JIANG, J.; JI, X. Asymmetric loss functions for noise-tolerant learning: theory and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 45, n. 7, p. 8094–8109, jul. 2023b. Disponível em: https://doi.org/10.1109/TPAMI.2023.3236459.

ZVARIKOVA, K.; HORAK, J.; BRADLEY, P. Machine and *deep learning* algorithms, computer vision technologies, and internet of things-based healthcare monitoring systems in COVID-19 prevention, testing, detection, and treatment. *American Journal of Medical Research*, v. 9, n. 1, p. 145-160, 2022. DOI: https://doi.org/10.22381/ajmr91202210.