

Pipeline automatizado para segmentação, rotulação e detecção de outliers em imagens de grãos de soja para geração de datasets

Gesmar de Paula Santo Júnior^{†*}, Alexandre Cardoso[†], Leonardo Garcia Marques^{†*}, Igor S. Peretta[†], Pedro Carvalho[‡], Gabriel Henrique Camargo Silva[†], Bruno Sousa Pereira[†] e Ivan Josef de Oliveira Esposito[†],

[†]Universidade Federal de Uberlândia, 38408-902

Email: {gesmarjunior, alexandre, iperetta}@ufu.br

Email: {gabrielhcs1411, brunosousapereira.az, ivanjoself141}@gmail.com

^{*}Instituto Federal de Goiás, Itumbiara, Goiás, 75524-245

Email: {leonardo.garcia, gesmar.junior}@ifg.edu.br

[‡]Grider, Praça Dr. Duarte, Sala 1, Uberlândia, Minas Gerais 38400-156

Email: pedro.carvalho@grider.com.br

Abstract - This paper presents a robust and automated computational pipeline for preprocessing grain images, performing the steps of segmentation, labeling, and outlier detection. The proposed methodology is flexible, processing both RGB and multispectral images captured in controlled environments as well as from external data sources, which introduce variations in lighting and background. The process begins with the optimization of color channels, such as the conversion from RGB to CMYK for soybean grains, followed by background removal and contour detection to isolate each grain individually. A key contribution of this work is an outlier detection module that uses the Structural Similarity Index (SSIM) to compare each segmented grain against a reference sample, automatically separating non-conforming grains for expert review. The result is a clean, standardized, and reliably labeled dataset, ready to be used in neural networks. The implementation, based on open-source libraries such as OpenCV and Scikit-image, demonstrates that a well-structured acquisition and processing pipeline acts as a quality multiplier, being fundamental to the success of AI-based solutions in agribusiness.

Keywords - Soybean Classification; Computer Vision Image Processing; Dataset Labeling; Machine Learning.

Resumo - Este artigo apresenta um pipeline computacional robusto e automatizado para o pré-processamento de imagens de grãos, realizando as etapas de segmentação, rotulação e detecção de outliers. A metodologia proposta é flexível, tratando imagens RGB e multiespectrais capturadas tanto em ambientes controlados quanto provenientes de fontes de dados externas, que introduzem variações de iluminação e fundo. O processo inicia com a otimização de canais de cor, como a conversão de RGB para CMYK para grãos de soja, seguida pela remoção de fundo e detecção de contornos para isolar cada grão individualmente. Uma contribuição central deste trabalho é um módulo de detecção de outliers que utiliza o Índice de Similaridade Estrutural (SSIM) para comparar cada grão

segmentado a uma amostra de referência, separando automaticamente os grãos não conformes para revisão de um especialista. O resultado é um conjunto de dados limpo, padronizado e confiavelmente rotulado, pronto para ser utilizado em redes neurais. A implementação, baseada em bibliotecas de software livre como OpenCV e Scikit-image, demonstra que um pipeline de aquisição e tratamento bem estruturado atua como um multiplicador de qualidade, sendo fundamental para o sucesso de soluções baseadas em IA no agronegócio.

Palavras-chave - Processamento de Imagens; Visão Computacional; Classificação de Grãos; Preparação de Dataset; Aprendizado de Máquina.

I. INTRODUÇÃO

A integração de tecnologias digitais e Inteligência Artificial (IA) na agricultura, ou Agrotech, tem revolucionado práticas tradicionais, promovendo avanços significativos em áreas como monitoramento de safras, automação de processos e controle de qualidade [1]. Um dos maiores desafios para o desenvolvimento de soluções de IA robustas é a disponibilidade de grandes volumes de dados de alta qualidade, devidamente anotados e padronizados, que sirvam como base para o treinamento de modelos de aprendizado de máquina.

No contexto da classificação de grãos, a qualidade do dataset é um fator determinante para a acurácia dos algoritmos. Imagens brutas, mesmo quando capturadas em ambientes controlados, contêm ruídos, variações de iluminação e informações de fundo que podem prejudicar o desempenho dos modelos. A etapa de segmentação, sendo o processo de isolar objetos de interesse (os grãos) do restante da imagem é, portanto, fundamental para padronizar as entradas da rede neural e garantir que o modelo foque nas características relevantes do grão.

Este trabalho aborda diretamente esse desafio ao propor um pipeline automatizado e completo para a criação de datasets de grãos. O objetivo principal é desenvolver um fluxo de trabalho que transforme imagens brutas (RGB e multispectrais) em um conjunto de dados organizado, onde cada grão é segmentado, salvo individualmente, rotulado de acordo com sua classe de origem e verificado quanto à sua conformidade. Uma inovação chave do nosso pipeline é a inclusão de uma etapa de detecção de outliers, que identifica e separa grãos que desviam significativamente em forma, cor ou tamanho de um padrão pré-definido, garantindo a integridade e a consistência do dataset final. O sistema foi projetado para ser robusto, lidando não apenas com imagens de nosso ambiente controlado, mas também com o "estresse de domínio" introduzido por imagens de terceiros, disponíveis em publicações científicas.

A interface do software desenvolvido é apresentada na Fig.1.

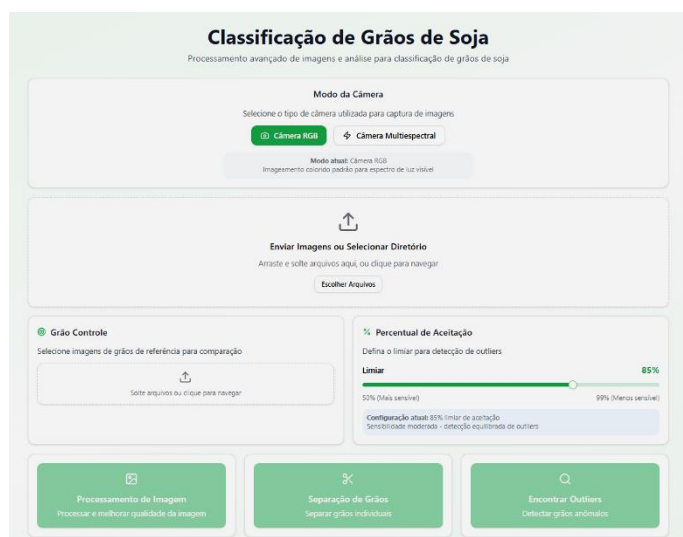


Fig. 1. Interface da ferramenta desenvolvida.

A segmentação de objetos em imagens agrícolas é um campo de pesquisa bem estabelecido. Técnicas clássicas, como limiarização, detecção de bordas e transformações de espaço de cor, são amplamente utilizadas, como demonstrado em nosso trabalho anterior focado na metodologia de aquisição de imagens de soja [1]. Outros pesquisadores também exploraram o uso de imagens de fontes diversas para treinar modelos, enfrentando desafios relacionados à variabilidade dos dados.

Nosso trabalho se diferencia ao integrar essas técnicas em um pipeline coeso que não apenas segmenta, mas também valida a qualidade de cada instância segmentada através de um sistema de detecção de outliers. Para validar a robustez de nossa abordagem, aplicamos nosso pipeline em conjuntos de dados públicos e imagens de artigos científicos, incluindo os disponibilizados em [2], [3], [4] e [5]. A capacidade de processar com sucesso essas imagens de

ambientes não controlados demonstra a generalização e a eficácia das técnicas de pré-processamento adotadas, como o uso combinado das funções morphologyEx e Canny Edge.

II. MATERIAIS E MÉTODOS

O pipeline proposto é estruturado em uma sequência de etapas modulares, projetadas para processar lotes de imagens desde a sua forma bruta até a geração de um dataset organizado e pronto para treinamento. A arquitetura foi concebida para lidar tanto com imagens RGB quanto com dados multispectrais.

A. Pré-processamento e Otimização de Canal de Cor

A primeira etapa consiste em converter a imagem original para um espaço de cor que maximize o contraste entre os grãos e o fundo. A escolha do canal ideal é dependente do objeto de estudo. Para grãos de soja, que possuem tons predominantemente amarelados, a conversão do espaço RGB para CMYK se mostrou altamente eficaz. Especificamente, o canal Amarelo isola os grãos com grande distinção, facilitando as etapas subsequentes, conforme validado em [1]. Este módulo é adaptável para outros tipos de grãos e espectros de imagem.

A Fig.2 apresenta as imagens após o pré-processamento, a esquerda o grão RGB com a banda Y isolada e a direita a imagem Multispectral também já tratada.

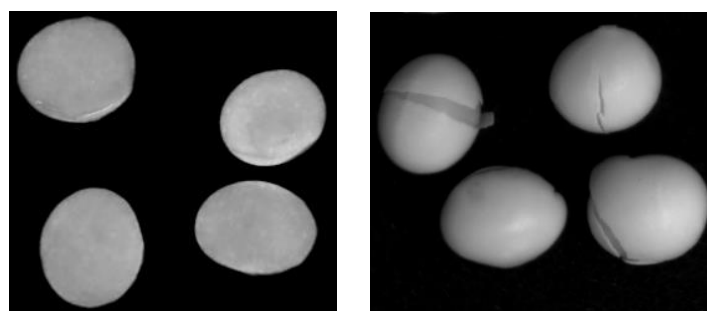


Fig. 2. Imagens pré-processadas RGB (esquerda) e Multispectral (direita).

A função `retirar_fundo` são utilizadas para remover um fundo azul (ou outra cor no caso de fontes externas) das imagens dos grãos no formato RGB.

Função: `retirar_fundo(dir_path, tipo, categoria)`

Parâmetros:

- `dir_path`: Diretório contendo as imagens dos grãos.
- `tipo`: Categoria do grão (ex: "normais").
- `categoria`: Diretório principal onde os resultados serão salvos.

B. Segmentação de Grãos Individuais

Com o canal de cor otimizado, o processo de segmentação é realizado da seguinte forma:

- Remoção de Fundo: Técnicas de limiarização ou mascaramento de cor são aplicadas para criar uma imagem binária onde os grãos são representados como objetos em primeiro plano.
- Detecção de Contornos: O algoritmo findContours do OpenCV é utilizado para identificar os limites de cada grão na imagem binária.
- Isolamento e Salvamento: Para cada contorno detectado, uma bounding box (caixa delimitadora) é calculada. A região correspondente a esta caixa é recortada da imagem original (colorida). O processo gera duas versões de cada grão: uma com o fundo original preservado e outra com o fundo removido (transparente), oferecendo flexibilidade para diferentes arquiteturas de redes neurais.

A Fig.3 mostra os grãos já tratados segmentados, na sequência o RGB sem fundo, RGB com fundo e o Multipectral.



Fig. 3. Grãos segmentados RGB (esquerda) sem fundo, RGB com fundo (meio) e Multipectral (direita).

A função Separar_graos isola e salva cada grão individualmente.

Função: Separar_graos(imagem_tratada_path, imagem_original_path, tipo_grao, output_base_dir, min_area=100)

Parâmetros:

- imagem_tratada_path: Caminho para a imagem que passou pela função de tratamento.
- imagem_original_path: Caminho para a imagem original colorida.
- tipo_grao: Define a subpasta onde os grãos serão salvos (ex: "normal", "multispectral").
- output_base_dir: Diretório base para salvar os grãos separados.
- min_area: Área mínima do contorno para ser considerado um grão.

C. Tratamento de Imagens de Fontes Externas

Um dos desafios ao utilizar imagens de terceiros é a falta de controle sobre a aquisição, resultando em problemas como grãos que se tocam e iluminação irregular. Para contornar isso, o pipeline incorpora operadores morfológicos (morphologyEx, como abertura e fechamento) para separar contornos levemente conectados e o detector de bordas Canny para refinar os limites dos grãos antes da segmentação, aumentando a precisão do processo em cenários não ideais, como mostra a Fig.4.

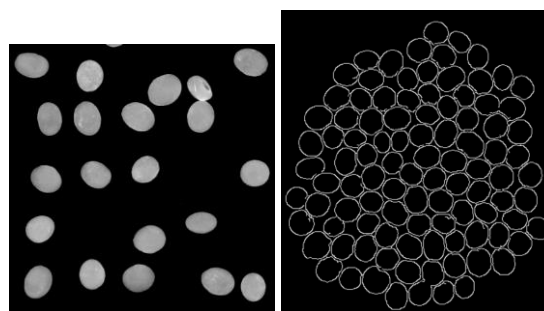


Fig. 4. Tratamento morphologyEx e Canny Edge de imagens de outra fonte [2].

A detecção dos grãos de fontes externas quando o fundo é uniforme e em cor azul é muito precisa, pois apresenta grande contraste com o grão de soja, como verificado por [1]. A Fig.5 mostra a detecção da imagem de [5].

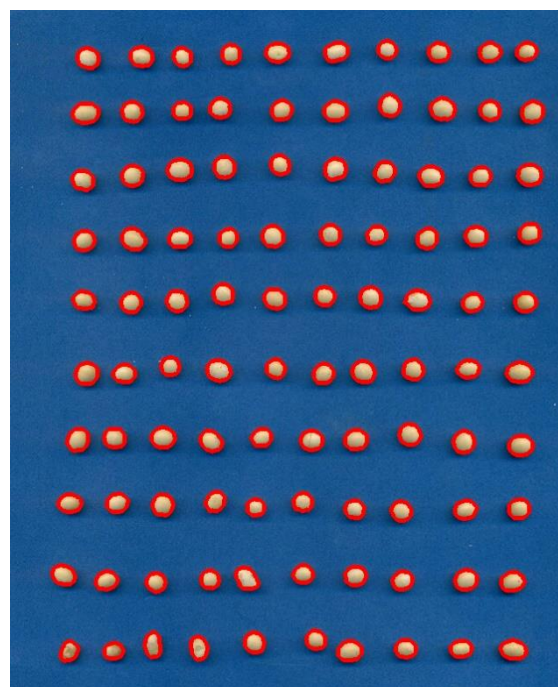


Fig. 5. Detecção de grãos de [5] com fundo azul de dados externos.

D. Detecção de Outliers

Como ferramenta complementar ao OpenCV, adotamos o scikit-image (sob licença BSD) por sua API científica orientada a NumPy, integração nativa com SciPy e scikit-learn, e amplo conjunto de rotinas para filtros, segmentação, morfologia e mensuração. No contexto deste trabalho, o scikit-image foi utilizado para processamento morfológico e extração de propriedades geométricas, facilitando análises morfológicas e auditorias quantitativas dos recortes por grão. A adoção dessa biblioteca reforça a reprodutibilidade do pipeline e sua aderência ao ecossistema de software livre, mantendo transparência nos parâmetros e resultados.

Esta é uma etapa crucial para garantir a qualidade do dataset. O processo é definido por duas regras principais:

- Amostra de Referência: Um dos grãos segmentados da amostra é selecionado manualmente como "grão de controle" ou referência para aquela classe.
- Limiar de Similaridade: Cada grão segmentado é comparado com o grão de referência usando o Índice de Similaridade Estrutural (SSIM), uma métrica de percepção que avalia similaridade em termos de luminância, contraste e estrutura. Um limiar de aceitação (ex: 0.60) é definido para determinar o quão sensível o sistema é a variações.

Grãos cujo score de SSIM com a referência fiquem abaixo do limiar são classificados como outliers. Eles não são descartados, mas movidos para uma pasta separada, permitindo que um especialista humano os reclassifique como uma nova classe, um tipo de defeito, ou os descarte definitivamente. Na Fig.1 é mostrado as opções na interface para detecção de outliers.

E. Rotulação Automatizada

A rotulagem final é baseada na estrutura de pastas de origem. O sistema assume que todas as imagens em um determinado diretório de entrada pertencem à mesma classe (ex: "soja boa", "soja ardida"). O pipeline preserva essa informação e a utiliza para criar uma estrutura de diretórios de saída organizada, seguindo uma taxonomia que considera o tipo de imagem (com/sem fundo), a classe do grão e o status (normal/outlier).

III. DESENVOLVIMENTO

O pipeline foi implementado em Python 3, utilizando principalmente as bibliotecas de código aberto OpenCV para operações de visão computacional, Scikit-image para a métrica SSIM, e NumPy para manipulação de arrays.

Utilizamos a versão 3.10.12 do Python e a Biblioteca do OpenCV versão 4.10.0. Todo o processamento foi feito no Google Colab.

A. Arquitetura do Software

O código é estruturado em funções modulares, cada uma responsável por uma etapa do pipeline. As principais funções são Tratamento_imagem, Separar_graos e achar_outliers.

O algoritmo para tratamento e conversão de canal pode ser resumido no seguinte pseudocódigo:

```
FUNÇÃO tratar_imagem(caminho_imagem):
    imagem_rgb = LER_IMAGEM(caminho_imagem)
    // Conversão para CMYK (simplificada)
    imagem_float = imagem_rgb / 255.0
    canal_K = 1 - MAX(imagem_float, eixo=cor)
    canal_C = (1 - imagem_float.R - canal_K)/(1 - canal_K)
    canal_M = (1 - imagem_float.G - canal_K)/(1 - canal_K)
    canal_Y = (1 - imagem_float.B - canal_K)/(1 - canal_K)
    // Aplica limiarização e detecção de bordas no canal Y
    _, imagem_tratada = THRESHOLD(canal_Y)
    imagem_tratada = CANNY_EDGE(imagem_tratada)
    imagem_tratada = MORPH_CLOSE(imagem_tratada)
    SALVAR_IMAGEM(imagem_tratada)
    RETORNAR sucesso
```

O núcleo da detecção de outliers, a função achar_outliers, é detalhada abaixo. Sua lógica é central para a limpeza do dataset.

```
FUNÇÃO achar_outliers(diretorio_graos,
imagem_controle):
    PARA cada imagem_grao em
    LISTAR_ARQUIVOS(diretorio_graos):
        grao_atual = LER_IMAGEM(imagem_grao)
        // Pula grãos muito pequenos
        SE LARGURA(grao_atual) < 100 OU
        ALTURA(grao_atual) < 100:
            MOVER_PARA_OUTLIERS(imagem_grao)
        CONTINUAR
    // Padroniza o tamanho para comparação
    grao_redimensionado =
    REDIMENSIONAR(grao_atual,
    TAMANHO(imagem_controle))
```

```
// Calcula a similaridade
score, _ = SSIM(imagem_controle,
grao_redimensionado)
// Verifica o limiar
SE score <= 0.60:
    MOVER_PARA_OUTLIERS(imagem_grao)
RETORNAR sucesso
```

B. Integração de Dados Multiespectrais

Embora o foco do código apresentado seja RGB, o pipeline foi projetado para acomodar dados multiespectrais. A aquisição desses dados utiliza LEDs de alta potência nas faixas NIR (Near-Infrared) e SWIR (Short-Wave Infrared), com detectores InGaAs. A seleção de bandas foi guiada por referências na área de imageamento SWIR, como [6], para capturar características não visíveis no espectro RGB, que são relevantes para a detecção de umidade e defeitos internos.

IV. RESULTADOS E DISCUSSÃO

A aplicação do pipeline resultou em um método eficaz e automatizado para a criação de datasets de grãos. O principal resultado é a capacidade do sistema de gerar uma estrutura de pastas organizada, contendo imagens de grãos individuais, devidamente rotuladas e separadas por classe e qualidade (normal vs. outlier).

A abordagem de salvar cada grão com e sem fundo mostrou-se vantajosa, pois permite testes comparativos sobre qual tipo de entrada gera melhores resultados para uma determinada arquitetura de rede neural. A etapa de detecção de outliers foi particularmente impactante. Em testes com lotes de imagens, o sistema foi capaz de identificar e isolar com sucesso não apenas grãos de classes diferentes misturados na amostra, mas também impurezas, grãos quebrados e artefatos de segmentação (fragmentos de contornos).

Um ponto importante de discussão é o trade-off entre o controle na aquisição das imagens e a complexidade do pré-processamento. Conforme observado, um ambiente de aquisição controlado, com iluminação e fundo padronizados, atua como um "multiplicador de qualidade", simplificando a segmentação e produzindo resultados mais consistentes. No entanto, a robustez do nosso pipeline, especialmente com o uso de operadores morfológicos e o detector Canny, permite o aproveitamento de dados de fontes externas, enriquecendo a variabilidade do dataset. A separação de outliers garante que essa variabilidade não se traduza em "sujeira" nos dados de treinamento, mas sim em amostras que podem ser analisadas e, se necessário, formar novas classes.

V. CONCLUSÃO

Este artigo apresentou um pipeline completo e automatizado para a segmentação, rotulação e detecção de outliers em imagens de grãos, visando a criação de datasets de alta qualidade para o treinamento de modelos de IA. A metodologia demonstrou ser robusta e flexível, processando com sucesso imagens RGB e multiespectrais de diferentes fontes.

As principais contribuições do trabalho são a integração de um fluxo de trabalho completo, desde a imagem bruta até o dataset pronto para uso, um método eficaz de detecção de outliers baseado em similaridade estrutural que garante a limpeza e consistência dos dados e a validação da abordagem em cenários com "estresse de domínio", utilizando imagens da literatura.

Este trabalho apresenta o pipeline de forma a ser reproduzível, concebido para facilitar as entradas de redes neurais e reduzir a variabilidade induzida pelo ambiente, iluminação e fundo. A solução utiliza entradas de diferentes formatos e bandas do espectro (RGB e multispectral). Para avaliar robustez, incorporamos imagens de terceiros e mitigamos dificuldades em imagens de diferentes fontes que é uma preocupação constante quando se busca aumentar as fontes de dados.

Como trabalhos futuros, pretendemos integrar este pipeline diretamente a um framework de treinamento de redes neurais, criando um sistema de aprendizado ativo onde os outliers identificados possam ser rapidamente rotulados por um especialista e reintroduzidos no ciclo de treinamento. Adicionalmente, planejamos expandir a aplicação do pipeline para outros tipos de grãos e sementes, ajustando os parâmetros de pré-processamento e validação para cada caso de uso.

AGRADECIMENTOS

Os autores gostariam de expressar sua gratidão à Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG) pelo apoio financeiro concedido por meio do auxílio de pesquisa COMPETE MINAS – LINHA TRÍPLICE HÉLICE APQ-03343-23, essencial para a realização deste trabalho.

Estendemos também nossos agradecimentos ao Grupo de Realidade Virtual e Aumentada (GRVA) da Universidade Federal de Uberlândia, ao Instituto Federal de Goiás e à Grider Automação e Tecnologia LTDA por suas importantes contribuições no fornecimento do suporte tecnológico e estrutural necessário a esta pesquisa.

Além disso, reconhecemos o auxílio de ferramentas de inteligência artificial (IA) na organização e tradução de diversas seções deste artigo. Essa integração de IA ao nosso fluxo de trabalho evidencia o potencial crescente da IA para aprimorar o processo de pesquisa e a produção acadêmica.

REFERÊNCIAS

- [1] SANTOS JÚNIOR, G. DE P. et al. Proposed Approach for Creating Soybean Grain Image Dataset. Anais do XXI Congresso Latino-Americano de Software Livre e Tecnologias Abertas (Latinoware 2024), p. 222–228, 27 nov. 2024. <https://doi.org/10.5753/latinoware.2024.245770>.
- [2] YANG, S. et al. High-throughput soybean seeds phenotyping with convolutional neural networks and transfer learning. Plant Methods, v. 17, n. 1, 5 maio 2021. <https://doi.org/10.1186/s13007-021-00749-y>.
- [3] ZHAO, G. et al. Real-time recognition system of soybean seed full-surface defects based on deep learning. Computers and Electronics in Agriculture, v. 187, p. 106230, ago. 2021. <https://doi.org/10.1016/j.compag.2021.106230>.
- [4] SONAWANE, S.; MOHANTY, B. K. An improved image processing scheme for automatic detection of harvested soybean seeds. Journal of Food Measurement and Characterization, v. 15, n. 6, p. 5607–5621, 30 ago. 2021. <https://doi.org/10.1007/s11694-021-01124-0>.
- [5] Villaverde, J. E., & Cleva, M. (2024). Volume estimation of unbroken soybeans samples using digital image processing techniques. Revista de Investigaciones Agropecuarias (RIA), 50(1), 8–13. <https://doi.org/10.58149/7aj0-5549>.
- [6] HANSEN, M. P.; MALCHOW, D. S. Overview of SWIR detectors, cameras, and applications. Thermosense XXX, 16 mar. 2008. <https://doi.org/10.1117/12.777776>