FoodLens AI: Um Sistema de Reconhecimento de Produtos com Visão Computacional, OCR e Enriquecimento de Dados em Tempo Real

Stheffany Santos[1], **Pedro Duran**[1],
Davi Rocha dos Santos[1], **Pedro Abreu**[1],
Victor Flávio Araujo[2]

[1]Universidade Tiradentes (UNIT)
CEP 49032-490 – Aracaju – SE – Brazil

stheffany.cruz@souunit.com.br, pedroav.duran@gmail.com

davi.cs@live.com, pedroarturmabreu@gmail.com

[2]Pontifícia Universidade Católica do Rio Grande do Sul (PUC-RS)

victor.araujo@edu.pucrs.br

Abstract. *This paper presents FoodLens AI, a mobile system for real-time product recognition using Computer Vision and Optical Character Recognition (OCR). The system identifies packaged products and extracts textual information from labels. It integrates object detection with YOLOv8, image preprocessing techniques, and OCR engines to ensure accurate data extraction in real-world environments. The extracted data is enriched using external sources and Large Language Models (LLMs). Unlike traditional approaches focused only on food analysis, the system supports broader product understanding. It provides features such as ingredient interpretation, product comparison, and market insights. These functionalities aim to support more informed consumer decision-making.*

Resumo. *Este artigo apresenta o FoodLens AI, um sistema mobile para reconhecimento de produtos em tempo real utilizando Visão Computacional e OCR. O sistema identifica produtos embalados e extrai informações textuais dos rótulos. A arquitetura integra detecção de objetos com YOLOv8 (avaliada empiricamente em cenários reais de varejo), e prevê etapas de pré-processamento de imagem e OCR para extração textual dos rótulos. Os dados extraídos são enriquecidos com fontes externas e LLMs. Diferente de abordagens focadas apenas em alimentos, o sistema amplia a compreensão de produtos. Além da análise nutricional, oferece interpretação de ingredientes, comparação entre produtos e informações de mercado, auxiliando a tomada de decisão do consumidor.*

1. Introdução

O reconhecimento automático de produtos em ambientes físicos representa um desafio relevante na interseção entre Visão Computacional e sistemas inteligentes. Em contextos de supermercados e pontos de venda, consumidores precisam interpretar rapidamente informações distribuídas em embalagens, como composição, marca, características nutricionais e preço, o que dificulta a tomada de decisão.

Apesar do avanço de aplicações baseadas em reconhecimento de alimentos, a maioria das soluções atuais é limitada a domínios específicos, como pratos preparados, não abordando de forma abrangente o reconhecimento de produtos embalados e o enriquecimento das informações associadas. Além disso, a leitura manual de rótulos continua sendo um processo pouco eficiente devido à complexidade das informações, à baixa legibilidade e ao uso de terminologia técnica.

Nesse contexto, o FoodLens AI propõe transformar o smartphone em um scanner inteligente capaz de reconhecer produtos, extrair informações textuais automaticamente e fornecer análises contextualizadas em tempo real. Dentre as principais contribuições deste trabalho, estão: a condução de um estudo comparativo entre o modelo YOLOv8n, baseado em redes convolucionais, e o RT-DETR-L, baseado em arquitetura Transformer, para fundamentar empiricamente a escolha do modelo de detecção utilizado no sistema. Além da proposição de um sistema integrado para reconhecimento de produtos, o desenvolvimento de um pipeline híbrido que combina detecção de objetos, OCR e modelos de linguagem, a introdução de uma camada de enriquecimento de dados e a avaliação baseada em métricas consolidadas na literatura.

2. Trabalhos Relacionados

Diversos estudos exploram o uso de Visão Computacional para reconhecimento de alimentos e análise nutricional, com destaque para sistemas voltados à identificação de pratos preparados a partir de imagens, frequentemente utilizados para estimativa calórica e monitoramento alimentar [Saputra et al. 2024]. Abordagens baseadas em redes profundas, como o NutriNet [Mezgec and Koroušić Seljak 2017] e o DeepFood [Liu et al. 2016], demonstraram resultados promissores para classificação de refeições em cenários controlados. Contudo, essas soluções apresentam limitações significativas quando aplicadas a produtos industrializados, pois não consideram a diversidade de embalagens, marcas e layouts visuais presentes em ambientes reais de varejo. No contexto da detecção de objetos em tempo real, a eficiência computacional do YOLOv8, decorrente do módulo Cross-Stage Partial (CSP) [Wang et al. 2020] e de otimizações na arquitetura de *backbone*, torna-o particularmente adequado para execução em dispositivos com recursos limitados, como smartphones [Jocher et al. 2023].

A comparação de modelos proposta por este artigo se dá porque avanços recentes evidenciam limitações inerentes às arquiteturas puramente convolucionais. Conforme apontado por [Arkin et al. 2023] em uma revisão abrangente dos métodos de detecção de objetos, desde 2012, a partir do trabalho seminal de [Krizhevsky et al. 2012], as CNNs dominaram as tarefas de Visão Computacional. Contudo, a crescente integração entre o mecanismo de atenção, originário do Processamento de Linguagem Natural, e as redes convolucionais tem tornado o uso de Transformers cada vez mais predominante, uma vez que algumas características dessa arquitetura compensam diretamente limitações das CNNs [Arkin et al. 2023].

Nesse contexto, [Carion et al. 2020] propuseram o DETR (Detection Transformer), o primeiro detector de objetos *end-to-end* baseado em Transformers. Posteriormente, [Lv et al. 2023] propuseram o RT-DETR (Real-Time Detection Transformer), uma variante otimizada para inferência em tempo real. O modelo preserva a capacidade de atenção global do DETR e consolida-se como alternativa competitiva às arquiteturas da

família YOLO, especialmente em cenários que demandam simultaneamente velocidade e robustez na generalização para instâncias visuais não observadas durante o treinamento.

Paralelamente, soluções comerciais frequentemente adotam tecnologias como códigos de barras e RFID para identificação de produtos, o que, embora eficiente, exige infraestrutura específica e não resolve o problema da interpretação semântica das informações contidas nos rótulos. Nesse contexto, o uso de OCR tem sido amplamente investigado como alternativa para extração textual automática. A combinação de Redes Neurais Convolucionais com Redes Neurais Recorrentes (CRNN), incorporando células LSTM para capturar dependências sequenciais entre caracteres, representa o estado da arte em reconhecimento óptico de texto em embalagens [Seitaj and Elangovan 2024]. Estudos específicos sobre rótulos de produtos alimentícios demonstram que a integração do motor Tesseract com modelos CRNN melhora consideravelmente a cobertura de atributos extraídos, embora erros persistam diante de caracteres especiais e ruídos de captura [Seitaj and Elangovan 2024].

A qualidade do pré-processamento de imagens exerce influência direta nos resultados do OCR em ambientes reais. Pesquisas recentes documentam pipelines sequenciais que combinam conversão para escala de cinza, suavização, equalização adaptativa de histograma (CLAHE) e limiarização, reportando ganhos acumulados de precisão superiores a 25 pontos percentuais em relação ao OCR aplicado sem nenhum pré-processamento [jou 2025]. Em particular, a aplicação de CLAHE tem se mostrado eficaz para destacar regiões textuais mesmo em condições adversas de iluminação, uma vez que opera sobre o contraste local em vez do contraste global da imagem [Domiciano 2025, Le et al. 2025].

No que se refere ao enriquecimento de dados a partir de fontes abertas, a base Open Food Facts destaca-se como repositório colaborativo com mais de 3,5 milhões de produtos cadastrados, fornecendo informações sobre ingredientes, tabelas nutricionais e classificações como o Nutri-Score [Open Food Facts 2024]. Sua API aberta tem sido amplamente utilizada em aplicações móveis e sistemas de recomendação para complementar dados extraídos de imagens com informações estruturadas e validadas por uma comunidade global de colaboradores.

Diferentemente dessas abordagens segmentadas, o FoodLens AI propõe uma solução integrada que combina detecção de objetos, extração textual com pré-processamento robusto e interpretação semântica via LLMs, permitindo não apenas reconhecer produtos, mas também transformar dados brutos em informações compreensíveis e contextualizadas para o consumidor final [Amugongo et al. 2023].

3. Metodologia

O FoodLens AI utiliza um pipeline composto por quatro etapas principais: detecção de objetos, pré-processamento de imagem, extração de texto e enriquecimento de dados, estruturadas de forma a garantir robustez e eficiência em cenários reais de uso. A Figura 1 ilustra a arquitetura geral do sistema.

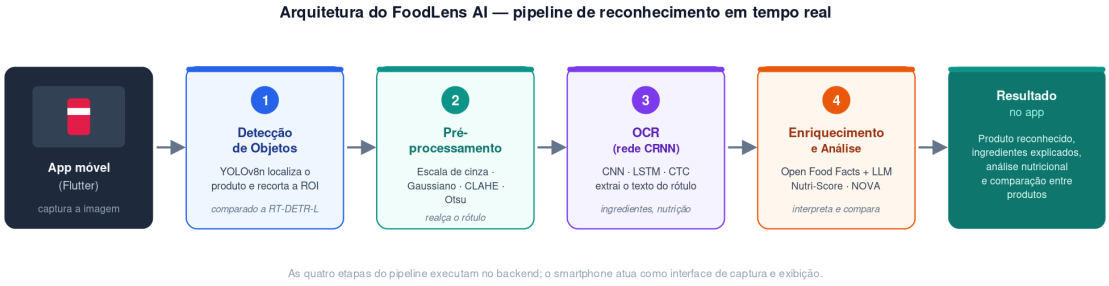


Figura 1. Arquitetura geral do FoodLens AI, composta por quatro etapas: detecção de objetos, pré-processamento de imagem, reconhecimento óptico de caracteres (OCR) e enriquecimento de dados. O smartphone atua como interface de captura e exibição, enquanto o pipeline é executado no backend.

3.1. Detecção de Objetos

A fim de fundamentar empiricamente a escolha do modelo de detecção utilizado no sistema, foram avaliados dois modelos representativos de famílias arquiteturais distintas: o YOLOv8n, menor variante da família YOLO com capacidade de detecção em tempo real (aproximadamente 3 milhões de parâmetros), e o RT-DETR-L, menor representante disponível da família RT-DETR (cerca de 32 milhões de parâmetros). Ambos constituem os pontos de entrada de suas respectivas arquiteturas para aplicações em cenários com restrições computacionais, tornando a comparação representativa do *trade-off* entre eficiência e capacidade de generalização em cada abordagem.

O YOLOv8 [Jocher et al. 2023] adota uma arquitetura de detecção em estágio único que reformula o problema de detecção de objetos como uma tarefa de regressão direta. Diferentemente de métodos tradicionais baseados em múltiplas etapas, o YOLO processa a imagem em uma única passagem pela rede, dividindo-a em uma grade e prevendo simultaneamente caixas delimitadoras, probabilidades de classe e níveis de confiança. A extração hierárquica de características é conduzida por Redes Neurais Convolucionais (CNNs), que aplicam operadores de convolução por meio de filtros (*kernels*) para gerar mapas de características (*feature maps*) progressivamente mais complexos ao longo das camadas, partindo de bordas e texturas até representações de instâncias completas. No YOLOv8, essa extração é implementada por meio de módulos Cross-Stage Partial (CSP), que dividem o fluxo de informações em caminhos distintos, reduzindo redundâncias computacionais e melhorando a eficiência do aprendizado.

O RT-DETR-L [Lv et al. 2023] adota uma arquitetura híbrida composta por três componentes principais [Arkin et al. 2023]: uma rede *backbone* CNN responsável pela extração inicial de *feature maps* a partir da imagem de entrada; um *encoder-decoder* Transformer que converte esses mapas em uma sequência de *embeddings* de comprimento fixo, na qual cada um representa um possível objeto na cena, processando-os por meio de atenção cruzada (*cross-attention*) entre *encoder* e *decoder*; e redes *feedforward* que predizem as coordenadas normalizadas das caixas delimitadoras e as categorias via SoftMax. Diferentemente do YOLOv8, que realiza predições de forma local e independente por célula da grade, o RT-DETR-L consulta a imagem de maneira global a cada inferência: cada *query* de objeto pode combinar informações de qualquer região da ima-

gem, conferindo ao modelo maior capacidade de capturar relações semânticas entre características visuais distribuídas, como cor, logotipo e tipografia, independentemente da forma ou disposição da embalagem.

Para garantir a validade científica do comparativo, ambos os modelos foram treinados sob protocolo idêntico: mesmo conjunto de dados e mesma divisão treino-validação-teste, mesmo número de épocas (50), mesmo tamanho de imagem de entrada (640×640 pixels), mesmo tamanho de *batch* (16) e mesma plataforma de treinamento (Ultralytics, GPU T4 via Google Colab). Os dois modelos partiram de pesos pré-treinados no conjunto COCO, submetendo-se ao mesmo processo de *fine-tuning* no *dataset* customizado descrito na Seção 3.2. Esse protocolo segue a recomendação de [Arkin et al. 2023], segundo a qual comparações entre famílias arquiteturais distintas são cientificamente válidas desde que o pipeline de treinamento seja igualado, de modo que as diferenças observadas nos resultados sejam atribuíveis exclusivamente às escolhas arquiteturais.

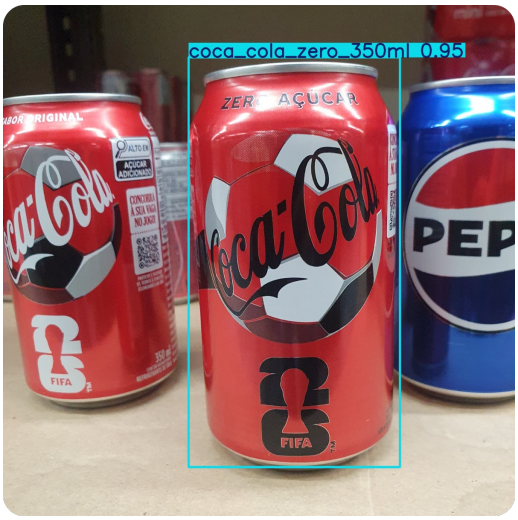


Figura 2. Exemplo de detecção de produto utilizando YOLOv8.

3.2. Base de Dados Utilizada

Para o treinamento modelos, foi construído um *dataset* utilizando a plataforma Roboflow, contendo imagens de produtos capturadas em ambientes que simulam a realidade do varejo. O conjunto de dados é composto por cinco classes distintas: *Coca-Cola 350ml*, *Coca-Cola Zero 350ml*, *Pepsi 350ml*, *Doritos* e *Ruffles*. Ao todo, o *dataset* contém 175 imagens anotadas manualmente, contemplando diferentes condições de iluminação, perspectiva e posicionamento dos produtos nas prateleiras.

A divisão do conjunto de dados seguiu a estratégia 70-20-10, sendo 70% das imagens destinadas ao treinamento, 20% para validação e 10% para testes. Essa distribuição permite avaliar a capacidade de generalização dos modelos em cenários distintos, reduzindo riscos de *overfitting* e contribuindo para uma análise mais robusta do desempenho do sistema.

As anotações foram realizadas no formato de caixas delimitadoras (*bounding boxes*), compatíveis com ambos os modelos para detecção multiclasse.

A Figura 3 apresenta exemplos de imagens do *dataset* contendo diferentes produtos detectados pelo modelo.



(a) Coca-Cola (b) Coca-Cola Zero (c) Pepsi

Figura 3. Exemplos de imagens de refrigerantes (350ml) contidas no *dataset* utilizado no treinamento do YOLOv8.

3.3. Pré-processamento de Imagem

Com o objetivo de maximizar a qualidade da extração textual, são aplicadas técnicas clássicas de processamento de imagem que atuam na melhoria da legibilidade dos rótulos. O pipeline de pré-processamento é executado de forma sequencial, conforme descrito a seguir.

Após a localização do produto pelo modelo de detecção, a região de interesse identificada é extraída e submetida ao pipeline de pré-processamento antes de ser encaminhada ao motor de OCR. Inicialmente, a imagem é convertida para escala de cinza, reduzindo a dimensionalidade dos dados e eliminando ruídos cromáticos que podem interferir na segmentação de regiões textuais. Em seguida, aplica-se o filtro Gaussiano para suavização, que atenua ruídos de alta frequência sem comprometer significativamente as bordas do texto. A equalização de histograma adaptativa por contraste limitado (CLAHE) é então utilizada para aprimorar o contraste local, operando em sub-regiões da imagem de forma independente, o que permite destacar regiões textuais mesmo em condições de iluminação heterogênea ou adversa [Domiciano 2025]. Estudos documentam que a aplicação sequencial dessas técnicas pode elevar a acurácia do OCR em mais de 25 pontos percentuais em relação ao reconhecimento sem pré-processamento [jou 2025].

Por fim, a limiarização de Otsu é empregada para binarização automática da imagem, separando regiões de interesse (texto) do fundo de maneira adaptativa, sem necessidade de parâmetros definidos manualmente. Essa característica é especialmente relevante em cenários reais, onde a diversidade visual das embalagens inviabiliza o uso de limiares fixos [Guru et al. 2024]. Trabalhos recentes aplicados ao reconhecimento de rótulos de embalagens confirmam que pipelines de pré-processamento baseados em CLAHE combinado com técnicas de limiarização consistentemente superam abordagens sem pré-processamento na tarefa de extração de texto [Le et al. 2025].

3.4. Reconhecimento Óptico de Caracteres

A extração textual é realizada por OCR baseado em redes neurais, adotando a arquitetura CRNN (*Convolutional Recurrent Neural Network*) — que combina camadas convolucionais para extração de características visuais com células LSTM (*Long Short-Term Memory*) para modelar dependências sequenciais entre caracteres, evitando a dissipação do gradiente típica de RNNs tradicionais [Seitaj and Elangovan 2024]. A saída é decodificada pelo algoritmo CTC (*Connectionist Temporal Classification*), que alinha a sequência predita ao texto original sem exigir segmentação prévia dos caracteres. Mecanismos de atenção podem ser integrados à arquitetura para melhorar o reconhecimento em condições de baixa qualidade de imagem, mostrando-se eficazes diante de variações de fonte, orientação e impressão comuns em rótulos de produtos industrializados [Label Your Data 2026].

3.5. Enriquecimento de Dados e Análise

Após a extração, os dados são submetidos a uma etapa de enriquecimento proveniente de bases externas e modelos de linguagem. A Open Food Facts, base colaborativa e aberta com mais de 3,5 milhões de produtos cadastrados ao redor do mundo [Open Food Facts 2024], é consultada para validar os dados obtidos diretamente das imagens. Além de ingredientes e tabelas nutricionais, a base provê classificações como o Nutri-Score e a classificação NOVA de grau de processamento, enriquecendo o contexto informacional disponível ao usuário.

Em seguida, Modelos de Linguagem de Grande Porte (LLMs) são utilizados para interpretar semanticamente as informações extraídas, transformando conteúdos técnicos em explicações acessíveis ao usuário leigo. Estudos recentes demonstram que LLMs aplicados a rótulos alimentícios são capazes de extrair elementos nutricionais estruturados e gerar descrições compreensíveis sobre composição e impacto à saúde dos produtos, com técnicas de pós-processamento — como reestruturação textual e normalização de numerais — contribuindo para aprimorar a acurácia das saídas geradas [Mohbat et al. 2025]. A integração de sistemas de OCR com LLMs para análise de rótulos tem se mostrado uma abordagem promissora para entregar avaliações nutricionais confiáveis e acionáveis ao consumidor [hea 2024].

Essa etapa possibilita a geração de *insights* como interpretação de ingredientes em linguagem natural, identificação de componentes críticos à saúde (como aditivos, alérgenos e açúcares adicionados), comparação entre produtos similares e geração de resumos explicativos contextualizados, agregando valor ao sistema e ampliando sua utilidade prática no processo de tomada de decisão do consumidor.

4. Resultados e Avaliação

A avaliação do sistema é conduzida considerando a qualidade da detecção de objetos, adotando métricas amplamente consolidadas na literatura: *Mean Average Precision* com limiar de IoU de 50% (mAP@0.5) e com limiares de 50% a 95% (mAP@0.5:0.95), Precision, Recall, latência de inferência (ms), taxa de quadros por segundo (FPS) e consumo de memória de vídeo GPU (MB). Para a extração textual, utilizam-se a Word Error Rate (WER) e a Character Error Rate (CER), que avaliam respectivamente erros em nível de palavra e de caractere [Guru et al. 2024].

4.1. Comparativo de Detecção de Objetos

A Tabela 1 apresenta os resultados obtidos para YOLOv8n e RT-DETR-L após treinamento sob protocolo idêntico, conforme descrito na Seção 3.1.

Tabela 1. Comparativo de desempenho entre YOLOv8n e RT-DETR-L, medido sob hardware idêntico (GPU T4, Google Colab) para garantir comparabilidade entre arquiteturas. Os valores não representam a latência do backend de produção, cuja infraestrutura está fora do escopo deste trabalho. Os valores em negrito indicam o modelo vencedor em cada métrica.

Métrica	YOLOv8n	RT-DETR-L	Vantagem
mAP@0.5	99,50%	99,50%	Empate
mAP@0.5:0.95	93,88%	**95,67%**	RT-DETR-L
Precision	98,17%	**98,22%**	RT-DETR-L
Recall	100%	100%	Empate
Latência (ms)	**10,89**	62,41	YOLOv8n
FPS	**91,81**	16,02	YOLOv8n
GPU (MB)	**787,70**	967,58	YOLOv8n

Ambos os modelos atingiram mAP@0.5 de 99,5% e Recall de 100% sobre o conjunto de teste, evidenciando que, para as classes presentes no *dataset*, as duas arquiteturas alcançam desempenho equivalente em condições controladas. A diferença mais relevante nas métricas de acurácia ocorre no mAP@0.5:0.95, métrica mais rigorosa que exige sobreposição precisa entre caixa predita e anotação real. Nela, o RT-DETR-L supera o YOLOv8n em 1,79 pontos percentuais (95,67% vs. 93,88%), indicando maior precisão na localização das caixas delimitadoras. Em contrapartida, o YOLOv8n demonstra vantagem expressiva em eficiência computacional: latência de 10,89 ms frente a 62,41 ms do RT-DETR-L, resultando em 91,81 FPS contra 16,02 FPS (uma diferença de 5,7 vezes em favor do modelo convolucional). O consumo de memória de vídeo também é menor no YOLOv8n (787,70 MB vs. 967,58 MB).

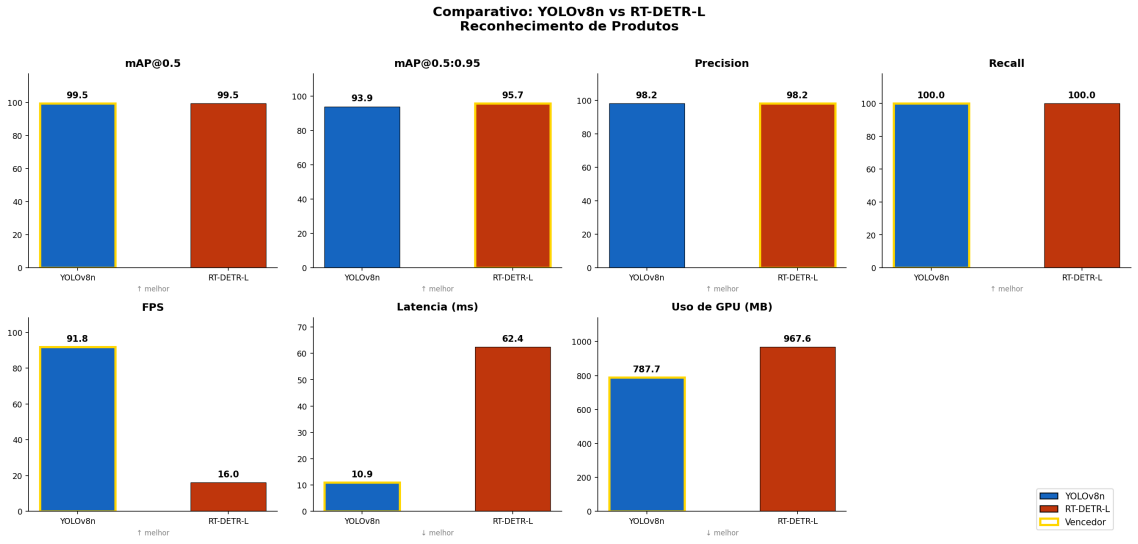


Figura 4. Comparativo de desempenho entre YOLOv8n e RT-DETR-L nas métricas de acurácia e eficiência computacional. O contorno amarelo indica o modelo vencedor em cada métrica.

5. Conclusão

O FoodLens AI propõe uma abordagem integrada para reconhecimento de produtos em tempo real, combinando Visão Computacional, OCR e modelos de linguagem em um único pipeline. A contribuição central deste trabalho é a comparação empírica entre YOLOv8n e RT-DETR-L como base para a escolha do modelo de detecção do sistema, complementada por uma arquitetura proposta de pré-processamento, OCR e enriquecimento de dados voltada a reduzir a dependência da leitura manual de rótulos e ampliar o acesso à informação sobre composição, características nutricionais e impactos à saúde dos produtos.

A escolha do YOLOv8n como modelo de detecção definitivo merece justificativa: a Tabela 1 mostra vantagem do RT-DETR-L nas métricas mais rigorosas, como mAP@0.5:0.95 (95,67% vs. 93,88%) e Precision (98,22% vs. 98,17%), mas essa vantagem é marginal (inferior a 2 pontos percentuais) e não se sustenta nas métricas menos rigorosas, nas quais os dois modelos empatam (mAP@0.5 de 99,50%, Recall de 100%). Já a diferença em eficiência computacional é substancial: 91,81 FPS e 10,89 ms de latência para o YOLOv8n, contra 16,02 FPS e 62,41 ms do RT-DETR-L — mais de 5 vezes de diferença. Para uma aplicação de reconhecimento em tempo real, esse ganho de eficiência supera o benefício marginal de precisão do modelo Transformer: a escolha do YOLOv8n foi deliberada, baseada no critério de eficiência, e não na ausência de alternativas mais precisas.

Como trabalhos futuros, destaca-se a avaliação *end-to-end* do sistema em condições reais de uso, indo além do desempenho isolado da etapa de detecção – já consolidada pela seleção do YOLOv8n. Nesse sentido, prioriza-se a mensuração da assertividade do OCR sobre rótulos capturados em campo, por meio de métricas como *Word Error Rate* (WER) e *Character Error Rate* (CER), e a análise da confiabilidade das saídas dos LLMs, avaliando a ocorrência de alucinações e a precisão das interpretações nutricionais geradas no contexto do sistema *mobile*. Complementarmente, destacam-se a otimização do sistema para execução em dispositivos móveis e a integração com mecanismos de recomendação personalizados, visando ampliar a relevância e o impacto da solução proposta.

Referências

(2024). Ocr and llm-based analysis of packaged food labels. *Frontiers in Health Informatics*, 13(3).

(2025). Optimization of tesseract ocr for automatic text extraction. *IJSECS*.

Amugongo, L. M. et al. (2023). Mobile computer vision-based applications for food recognition. *IEEE Access*.

Arkin, E., Yadikar, N., Xu, X., Aysa, A., and Ubul, K. (2023). A survey: object detection methods from CNN to transformer. *Multimedia Tools and Applications*, 82(14):21353–21383.

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., and Zagoruyko, S. (2020). End-to-end object detection with transformers. In *Computer Vision – ECCV 2020*, volume 12346 of *Lecture Notes in Computer Science*, pages 213–229. Springer.

Domiciano, M. (2025). Image pre-processing techniques for ocr. https://medium.com/@TechforHumans/image-pre-processing-techniques-for-ocr.

Guru, S. et al. (2024). Implementation of health impact assessment of packaged foods through nutritional label recognition using ocr. *Journal of Food Informatics*.

Jocher, G., Chaurasia, A., and Qiu, J. (2023). Ultralytics YOLO. https://github.com/ultralytics/ultralytics.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). ImageNet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*, volume 25.

Label Your Data (2026). Ocr deep learning: How do you do it in 2026. https://labelyourdata.com/articles/ocr-with-deep-learning.

Le, T.-T.-H. et al. (2025). A robust framework for coffee bean package label recognition. *Sensors*.

Liu, C., Cao, Y., Luo, Y., Chen, G., Vokkarane, V., and Ma, Y. (2016). DeepFood: Deep learning-based food image recognition for computer-aided dietary assessment. In *Inclusive Smart Cities and Digital Health*, volume 9677 of *Lecture Notes in Computer Science*, pages 37–48. Springer.

Lv, W., Zhao, Y., Xu, S., Wei, J., Wang, G., Cui, C., Du, Y., Dang, Q., and Liu, Y. (2023). DETRs beat YOLOs on real-time object detection.

Mezgec, S. and Koroušić Seljak, B. (2017). NutriNet: A deep learning food and drink image recognition system for dietary assessment. *Nutrients*, 9(7):657.

Mohbat, F., Zaki, M. J., et al. (2025). Extract nutritional information from bilingual food labels using large language models. *Journal of Imaging*, 11(8).

Open Food Facts (2024). Collective achievements 2024. https://blog.openfoodfacts.org.

Saputra, M. E., Susanto, A., and Carmelita, B. J. (2024). Implementation of tesseract ocr and bounding box for text extraction on food nutrition labels. *International Journal of Advanced Computer Science and Applications*.

Seitaj, H. and Elangovan, V. (2024). Information extraction from product labels. *International Journal of Artificial Intelligence and Applications (IJAIA)*, 15(2).

Wang, C.-Y., Liao, H.-Y. M., Wu, Y.-H., Chen, P.-Y., Hsieh, J.-W., and Yeh, I.-H. (2020). CSPNet: A new backbone that can enhance learning capability of CNN. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 390–391.