**São Paulo Tech School**

**Curso de Ciência da Computação**

**CLASSIFICATION SYSTEM FOR RECYCLABLE MATERIALS FOCUSED ON GLASS AND PLASTIC USING ARTIFICIAL INTELLIGENCE AND COMPUTER VISION**

Giovana Sanchez Mainente – 02211022  
 Helder Davidson Rodrigues Alvarenga- 03221027  
 Pedro Henrique Freitas Dias - 02221037

São Paulo  
 2025

**Sumário**

[1. Resumo 2](#_Toc499895965)

[ 1.1 Abstract 3](#_Toc1567162966)

[ 1.2 Palavras Chaves 3](#_Toc232446778)

[2. Introdução 4](#_Toc1073940597)

[3. Contextualização 4](#_Toc1259031645)

[4. Justificativa 5](#_Toc1815686180)

[5. Contextos Técnicos 6](#_Toc1153281310)

[ 5.1 Machine Learning (Aprendizado de Máquina) 7](#_Toc2088576209)

[ 5.2 Deep Learning (Aprendizado Profundo) 7](#_Toc1815421579)

[ 5.3 Classificação vs. Detecção de Objetos 7](#_Toc1829494172)

[ 5.4 Detecção de Objetos (Object Detection) 8](#_Toc75975928)

[ 5.5 Bounding Box (Caixa Delimitadora) 8](#_Toc1030296587)

[ 5.6 Métricas de Avaliação de Desempenho 8](#_Toc192260016)

[ 5.7 Precisão (Precision) e Recall (Sensibilidade) 9](#_Toc382283317)

[ 5.8 Average Precision (AP) e mean Average Precision (mAP) 9](#_Toc1955536290)

[6. Metodologia 10](#_Toc1175499797)

[ 6.1 Estrutura da Base de Dados 11](#_Toc560598995)

[ 6.2 Uso da Plataforma Roboflow 11](#_Toc1061402514)

[7. Arquitetura 11](#_Toc2125776570)

[8. Implementação da solução 14](#_Toc1077856722)

[ 8.1 - 1ª versão do modelo: 15](#_Toc1401630069)

[9. Conclusão 25](#_Toc288520354)

[10. Referências 27](#_Toc494413232)

# 1. Resumo

Esse trabalho desenvolve um sistema automatizado para identificação e classificação de materiais recicláveis, com foco em plástico e vidro, por meio de visão computacional e inteligência artificial. Diante do crescente volume de resíduos sólidos urbanos e da baixa eficiência do processo manual de separação, a solução proposta busca aumentar a precisão, velocidade e segurança da triagem. Para isso, foi utilizada uma base de dados estruturada no formato YOLO e processada com o auxílio da plataforma Roboflow. O modelo YOLOv11 foi treinado em duas versões, permitindo avaliar a evolução do desempenho. Os resultados mostram melhorias significativas, especialmente na classe “vidro”, cujo recall aumentou de 66,67% para 89,66%. A comparação entre as versões evidencia avanços tanto na estabilidade quanto na performance global do modelo. O sistema proposto demonstra potencial para otimizar processos de reciclagem, reduzir custos, melhorar a qualidade dos materiais recuperados e contribuir para práticas sustentáveis alinhadas à economia circular.

## **1.1 Abstract**

This work develops an automated system for identifying and classifying recyclable materials, focusing on plastic and glass, through computer vision and artificial intelligence. Given the growing volume of urban solid waste and the low efficiency of the manual separation process, the proposed solution aims to increase the accuracy, speed, and safety of sorting. For this purpose, a dataset structured in the YOLO format was used and processed with the support of the Roboflow platform. The YOLOv11 model was trained in two versions, allowing an assessment of the performance of evolution. The results show significant improvements, especially in the “glass” class, whose recall increased from 66.67% to 89.66%. The comparison between versions highlights advancements in both stability and overall model performance. The proposed system demonstrates potential to optimize recycling processes, reduce costs, improve the quality of recovered materials, and contribute to sustainable practices aligned with the circular economy.

## **1.2 Palavras Chaves**

Machine Learning, dataset, detecção de objetos, AWS ( Amazon Web Services), RoboFlow, data augmentation, processamento de imagens, YOLOv11, reciclagem, visão computacional, classificação de material reciclável.

# 2. Introdução

O crescente volume de resíduos sólidos urbanos representa um dos maiores desafios ambientais e sociais da atualidade. A gestão inadequada desses resíduos não apenas sobrecarrega aterros sanitários, mas também resulta na perda de materiais com alto potencial de reaproveitamento, intensificando a exploração de recursos naturais (CALDERONI, 2003). Nesse cenário, a reciclagem emerge como uma solução fundamental. Contudo, a eficiência dos processos de reciclagem está diretamente atrelada à qualidade da triagem dos materiais. O método de separação manual, ainda predominante em muitos centros, é caracterizado pela baixa produtividade, altos custos operacionais e riscos à saúde dos trabalhadores (GONÇALVES, 2019). Este projeto propõe o desenvolvimento de um sistema automatizado, fundamentado em Inteligência Artificial e visão computacional, para a identificação e classificação de resíduos recicláveis, visando melhorar o processo de triagem e aumentar a eficiência da cadeia de reciclagem.

# 3. Contextualização

A sociedade contemporânea enfrenta uma crise na gestão de resíduos, impulsionada pelo consumo em massa e pela produção de bens com ciclo de vida reduzido. Segundo dados do Panorama dos Resíduos Sólidos no Brasil, o país gera milhões de toneladas de lixo anualmente, das quais uma parcela significativa poderia ser reintegrada ao ciclo produtivo através da reciclagem (ABRELPE, 2023). A transição para um modelo de economia circular, onde o resíduo é visto como recurso, é imperativa para a sustentabilidade. A tecnologia, especialmente no âmbito da Indústria 4.0, oferece ferramentas poderosas para essa transformação. Sistemas ciberfísicos e Inteligência Artificial (IA) já demonstram potencial para revolucionar setores produtivos, e sua aplicação na gestão de resíduos é um campo promissor e necessário para escalar os processos de reciclagem a um nível compatível com a demanda atual.

# 4. Justificativa

A implementação de um sistema automatizado para a triagem de recicláveis se justifica por três pilares principais: econômico, ambiental e social.

* **Viabilidade Econômica:** A automação do processo de separação eleva a velocidade e a precisão da triagem, resultando em um volume maior de material reciclável de melhor qualidade. Materiais mais puros possuem maior valor de mercado, gerando mais receita para cooperativas e empresas de reciclagem. Adicionalmente, há uma redução de custos associados à mão de obra e à diminuição de erros operacionais (XAVIER; CORRÊA, 2021).
* **Impacto Ambiental:** Um sistema eficiente garante que uma maior quantidade de papel, plástico, vidro e metal seja efetivamente reciclada, reduzindo a necessidade de extração de matérias-primas virgens, economizando energia e água, e diminuindo a emissão de gases de efeito estufa. A consequência direta é a mitigação da poluição e a preservação dos ecossistemas.
* **Benefício Social:** A separação manual de resíduos expõe os trabalhadores a condições insalubres e perigosas, com riscos de contaminação e acidentes. A automação desta tarefa realoca a força de trabalho para funções de supervisão, manutenção e gestão, criando um ambiente de trabalho mais seguro e digno.

# 5. Contextos Técnicos

O projeto de desenvolvimento de um sistema automatizado para a identificação de resíduos recicláveis está fundamentalmente ancorado nos pilares da **Inteligência Artificial (IA)**, mais especificamente no campo da **Visão Computacional** e do **Deep Learning**. Este capítulo detalha os conceitos técnicos que servem de base para a metodologia e o desenvolvimento do sistema proposto.

## **5.1 Aprendizado de Máquina (Machine Learning)**

O Machine Learning (ML) é um subcampo da Inteligência Artificial que se concentra no desenvolvimento de algoritmos e modelos estatísticos que permitem que sistemas de computador "aprendam" a partir de dados, sem serem explicitamente programados para executar uma tarefa específica. Em essência, o ML busca padrões em grandes volumes de dados para tomar decisões ou fazer previsões. Formalmente, um programa de computador aprende com a experiência E em relação a alguma classe de tarefas T e medida de desempenho P, se o seu desempenho em T, medido por P, melhorar com a experiência E (MITCHELL, 1997).

## **5.2 Aprendizado Profundo (Deep Learning)**

O Deep Learning (DL) é um subconjunto do Machine Learning que se destaca pela utilização de **Redes Neurais Artificiais (RNAs) profundas**, ou seja, redes que contêm múltiplas camadas ocultas (*hidden layers*). A profundidade da rede permite que o sistema aprenda representações hierárquicas dos dados, extraindo características cada vez mais complexas. O DL é a base para as Redes Neurais Convolucionais (CNNs), que são o modelo padrão para a tarefa de Visão Computacional, como o modelo **YOLOv11** empregado neste trabalho.

## **5.3 Classificação vs. Detecção de Objetos**

O sistema proposto está inserido no domínio da Visão Computacional, que é a disciplina científica que lida com a forma como os computadores podem obter uma compreensão de alto nível a partir de imagens ou vídeos digitais.

**Classificação de Imagem (Image Classification)**

A classificação de imagem é a tarefa mais básica da visão computacional. O objetivo é receber uma imagem como entrada e retornar um único rótulo de classe (categoria) que descreva o conteúdo principal da imagem. É uma tarefa de identificação global.

## **5.4 Detecção de Objetos (Object Detection)**

A detecção de objetos é uma tarefa mais complexa e crucial para o sistema de triagem. Ela não só identifica a **categoria** do objeto (classificação), mas também determina sua **localização exata** dentro da imagem ou *frame* de vídeo. O sistema deve localizar múltiplas instâncias de diferentes objetos simultaneamente na mesma cena.

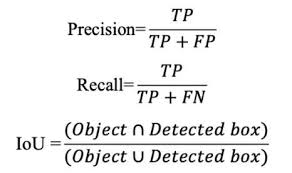
## **5.5 Caixa Delimitadora (Bounding Box)**

Uma **Bounding Box** é um retângulo que circunscreve um objeto de interesse em uma imagem. É tipicamente definida por quatro coordenadas (x mínimo, y mínimo, x máximo, y máximo). Essas coordenadas permitem que o sistema de automação (como um braço robótico) saiba exatamente onde o material a ser separado está localizado, diferenciando a detecção da classificação, que apenas informa a presença do objeto sem sua posição.

## **5.6 Métricas de Avaliação de Desempenho**

A eficácia do modelo YOLOv11 é medida por um conjunto de métricas que avaliam tanto a precisão da localização quanto a correção da classificação.

O **IoU (Interseção sobre União)** é a métrica que avalia a precisão da localização de uma detecção. É calculado dividindo a área de interseção entre a caixa delimitadora prevista pelo modelo (Predicted Box) e a caixa delimitadora real (Ground Truth Box) pela área da união dessas duas caixas.



*Figura 1. Fórmula IoU - Fonte: learnopencv (2022)*

Se o IoU for superior a um limiar pré-definido (geralmente 0.5), a detecção é considerada um **Verdadeiro Positivo**.

## **5.7 Precisão (Precision) e Recall (Sensibilidade)**

Essas métricas são baseadas nos resultados de classificação do modelo, após a definição do limiar de IoU:

* **True Positive (TP):** O modelo detectou e localizou um objeto corretamente (IoU >= limiar).
* **False Positive (FP):** O modelo detectou um objeto que é inexistente (falso alarme) ou a localização está incorreta (IoU < limiar).
* **False Negative (FN):** O modelo falhou em detectar um objeto que estava presente na imagem.

**a) Precisão (P):** Mede a proporção de detecções positivas que foram, de fato, corretas. Um valor alto de Precisão significa poucos Falsos Positivos.

Fórmula da Precisão:

Precisão = TP / (TP + FP)

**b) Recall (R):** Mede a proporção de objetos reais presentes que foram detectados corretamente. Um valor alto de Recall significa poucos Falsos Negativos.

Fórmula do Recall:

Recall = TP / (TP + FN)

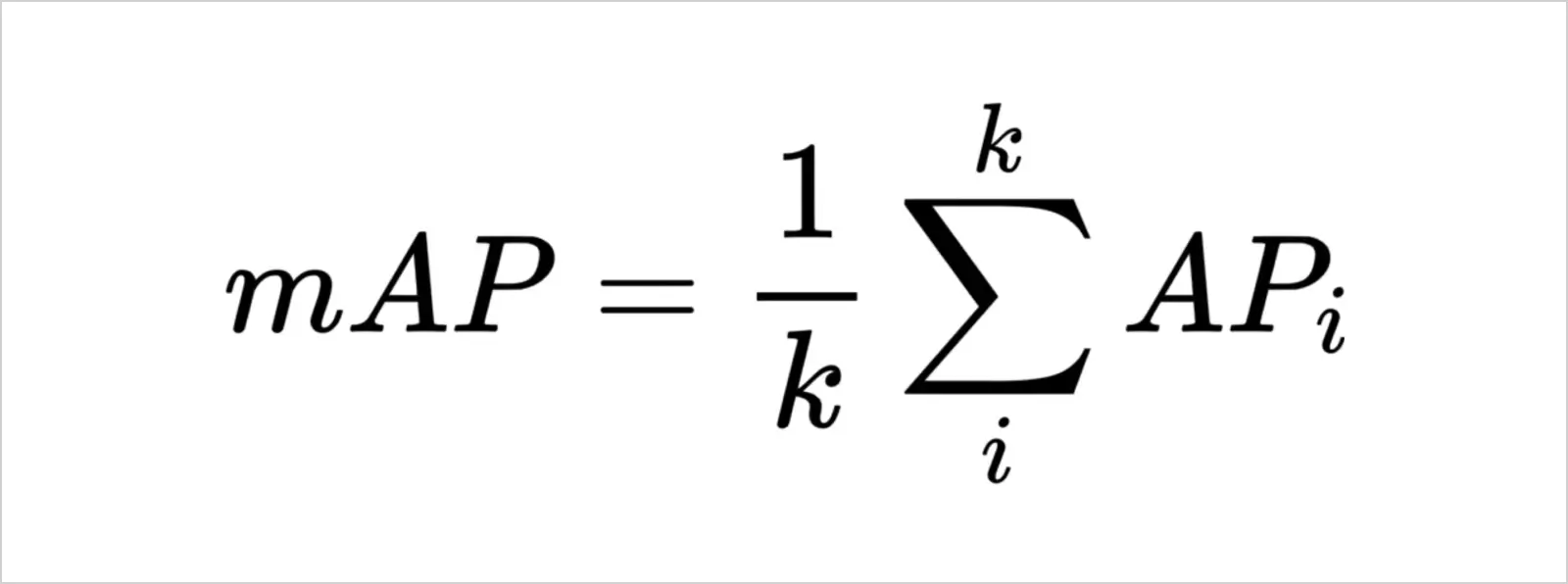
## **5.8 Average Precision (AP) e mean Average Precision (mAP)**

Em sistemas de detecção, o **Average Precision (AP)** é uma métrica que resume a curva de Precisão-Recall (P-R), calculando a área sob esta curva.

Fórmula do AP:

AP = Integral de 0 a 1 de P(R) dR (Área sob a curva de Precisão-Recall)

O **mAP (mean Average Precision)** é a métrica padrão para avaliar modelos multicasses, sendo o valor médio do AP calculado para cada classe individual que o modelo deve detectar. É o indicador mais robusto da qualidade global do modelo.



*Figura 2. Fórmula mAP - Fonte: superannotate(2023)*

Onde N é o número total de classes (neste projeto, N=2: Plástico e Vidro), e AP subscrito i é o Average Precision para a classe i.

# **6**. Metodologia

Para o desenvolvimento deste Trabalho de Conclusão de Curso, especificamente na etapa de visão computacional, utilizamos uma base de dados estruturada no padrão **YOLO (You Only Look Once)**, empregando a versão mais recente do modelo, o **YOLOv11**.

## 6.1 Estrutura da Base de Dados

O formato YOLO é amplamente utilizado e altamente eficiente para tarefas de detecção de objetos, pois armazena as informações de classe e localização por meio de arquivos de texto (.txt) associados a cada imagem. Cada arquivo contém as coordenadas normalizadas das bounding boxes, facilitando o processamento durante o treinamento do modelo.

A adoção do **YOLOv11** proporcionou vantagens como maior precisão, melhor desempenho em tempo real e aprimoramentos na capacidade de generalização em comparação às versões anteriores, tornando-o adequado às necessidades deste projeto.

## 6.2 Uso da Plataforma Roboflow

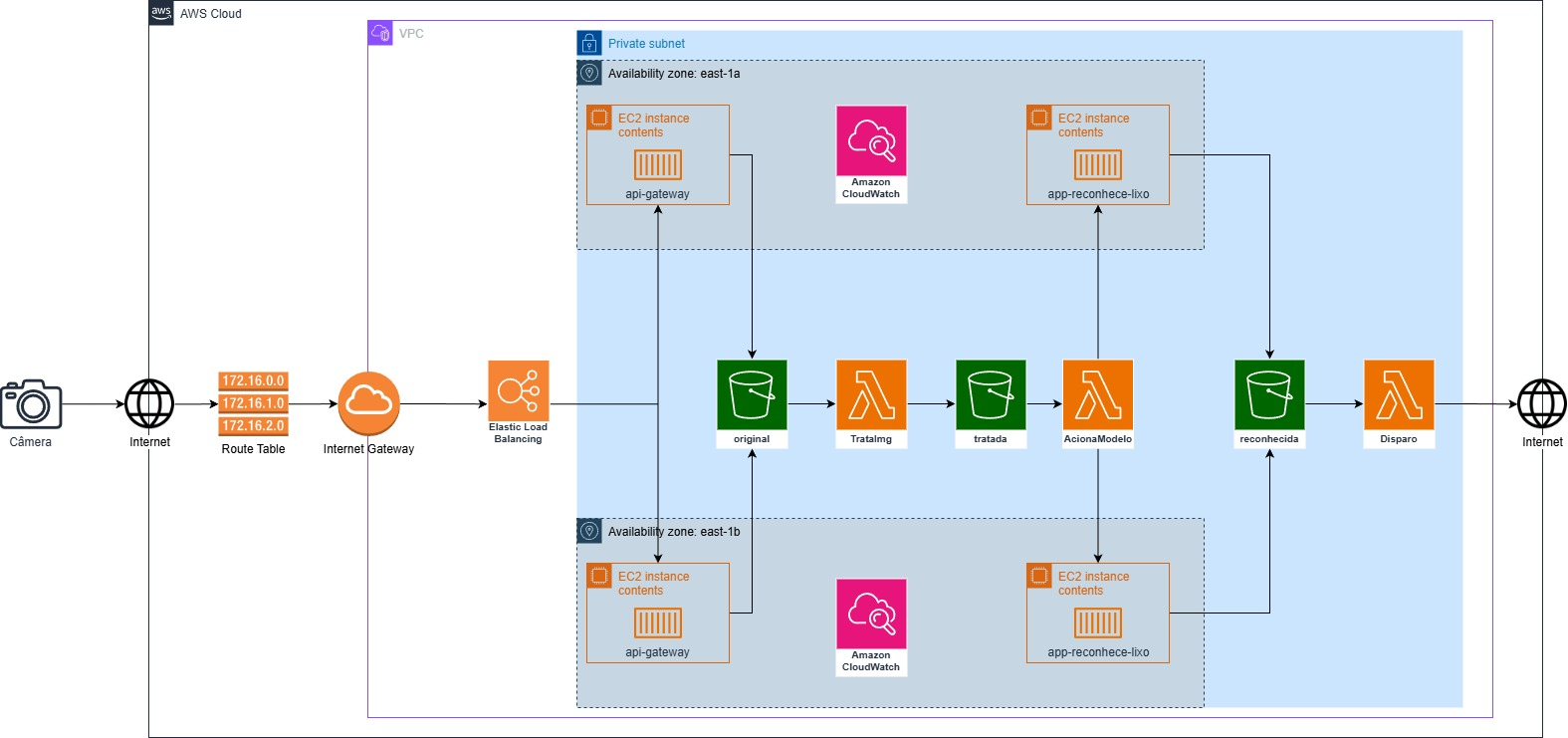
Para a organização, pré-processamento e gerenciamento da base de dados no formato YOLO, utilizamos a plataforma **Roboflow**, que se mostrou essencial na metodologia por permitir:

**• Versionamento e organização:** Centralização do dataset e controle eficiente de suas diferentes versões durante o desenvolvimento.  
 **• Pré-processamento:** Aplicação de transformações como redimensionamento, padronização e correções necessárias para uniformizar o conjunto de dados.  
 **• Aumento de Dados (Data Augmentation):** Geração de amostras sintéticas (como rotações, flips, recortes e ajustes de iluminação) para aumentar a robustez e a capacidade de generalização do modelo treinado.

# 7. Arquitetura

## 7.1 Infraestrutura em Nuvem do Projeto

A infraestrutura do projeto foi desenvolvida utilizando os serviços da AWS (Amazon Web Services), adotando o modelo de *Infrastructure as a Service* (IaaS). A arquitetura foi projetada para operar de forma assíncrona, garantindo escalabilidade, modularidade e alta disponibilidade.



*Figura 3. Arquitetura de infraestrutura do projeto – Fonte: Autoral*

1. **Monitoramento com Amazon CloudWatch**

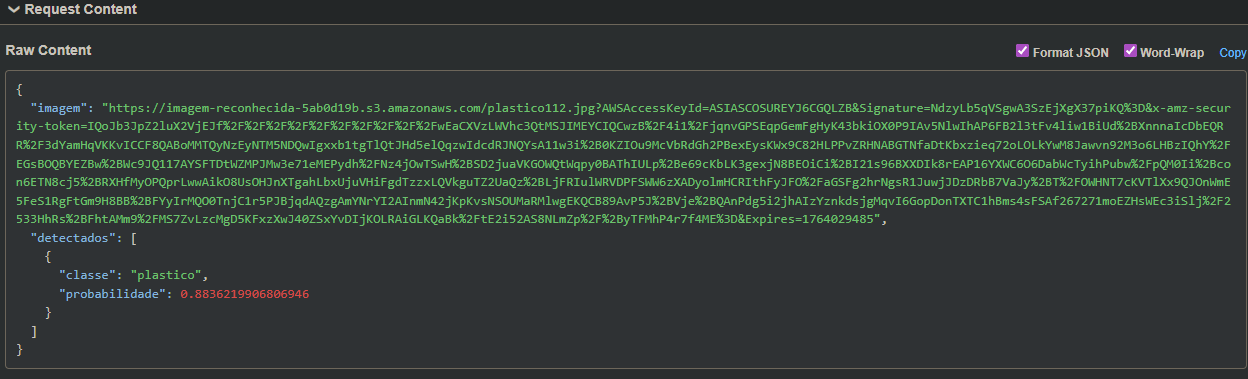
**O serviço Amazon CloudWatch é utilizado para monitorar as instâncias EC2 e as funções Lambda, permitindo o acompanhamento de métricas, logs e alertas em tempo real, o que contribui para a observabilidade e manutenção do sistema.**

1. ****Segurança e Rede (VPC, Subnets e Internet Gateway)****  
   Toda a infraestrutura está organizada em uma **VPC (Virtual Private Cloud)**, dividida em subnets **privadas** em 2 zonas de disponibilidade (east-1a e east-1b), o que aumenta a resiliência do sistema. O acesso à internet é feito por meio de um **Internet Gateway**, enquanto as tabelas de roteamento controlam o tráfego de rede de forma segura.
2. **Envio da Imagem pelo Usuário**

**O processo de identificação inicia-se com o envio de um arquivo de imagem, com tamanho máximo de 5MB, por parte do usuário, e um endereço webhook que receberá o retorno. Esse arquivo é recebido por uma aplicação hospedada em uma instância EC2 (api-gateway) e armazenado em um bucket S3 chamado original, destinado ao armazenamento das imagens sem qualquer tipo de tratamento prévio.**

1. **Tratamento da Imagem**  
   Após o armazenamento inicial, uma função Lambda chamada TrataImg é acionada automaticamente. Essa função realiza o redimensionamento da imagem, com o objetivo de otimizar seu tamanho para os próximos estágios do fluxo de processamento. A imagem tratada é então salva em um segundo bucket S3, denominado tratada.
2. **Acionamento do Modelo de Reconhecimento**  
   Em seguida, a função Lambda AcionaModelo é ativada, sendo responsável por enviar a imagem tratada para um endpoint interno. Nesse endpoint, uma instância EC2 (app-reconhece-lixo) executa o modelo de reconhecimento, que identifica o conteúdo da imagem.
3. **Armazenamento do Resultado Reconhecido**  
   Após o reconhecimento, a imagem tratada, juntamente com os resultados obtidos pelo modelo, é armazenada em um terceiro bucket S3, chamado reconhecida.
4. ****Retorno ao Usuário via Webhook****

**Por fim, a função Lambda Disparo é acionada para enviar ao remetente, via webhook, um JSON contendo o link pré-assinado da imagem tratada e outro JSON com as análises realizadas pelo modelo de reconhecimento.**

*Figura 4. Arquitetura de infraestrutura do projeto – Fonte: Autoral*

## 7.2 Análise Comparativa dos Algoritmos

O código de tratamento implementado na função Lambda *TrataImg* foi projetado para otimizar o custo-benefício do sistema. O objetivo principal é reduzir o tamanho das imagens recebidas, garantindo menor consumo de largura de banda e maior eficiência no processamento subsequente. Essa etapa é crítica, pois influencia diretamente o tempo de resposta do modelo de reconhecimento e o custo de execução em ambiente serverless.

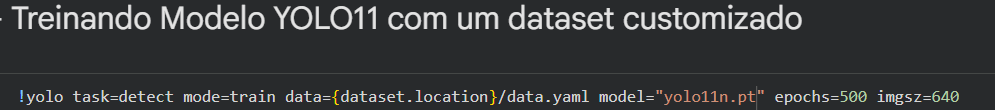
Para essa finalidade, foram desenvolvidos dois algoritmos distintos de tratamento de imagem, denominados **Algoritmo 1 (Byte)** e **Algoritmo 2 (Donwload)**. Ambos realizam operações de redimensionamento e compressão, mas diferem em sua abordagem de otimização de recursos computacionais.

Os testes realizados em ambiente AWS Lambda consideraram métricas como **duração da execução (ms)**, **memória alocada (GB)**, **tamanho da imagem (KB)**, **GB-segundos consumidos** e **custo por execução (USD)**.  
*Fonte: https://aws.amazon.com/pt/lambda/pricing/*

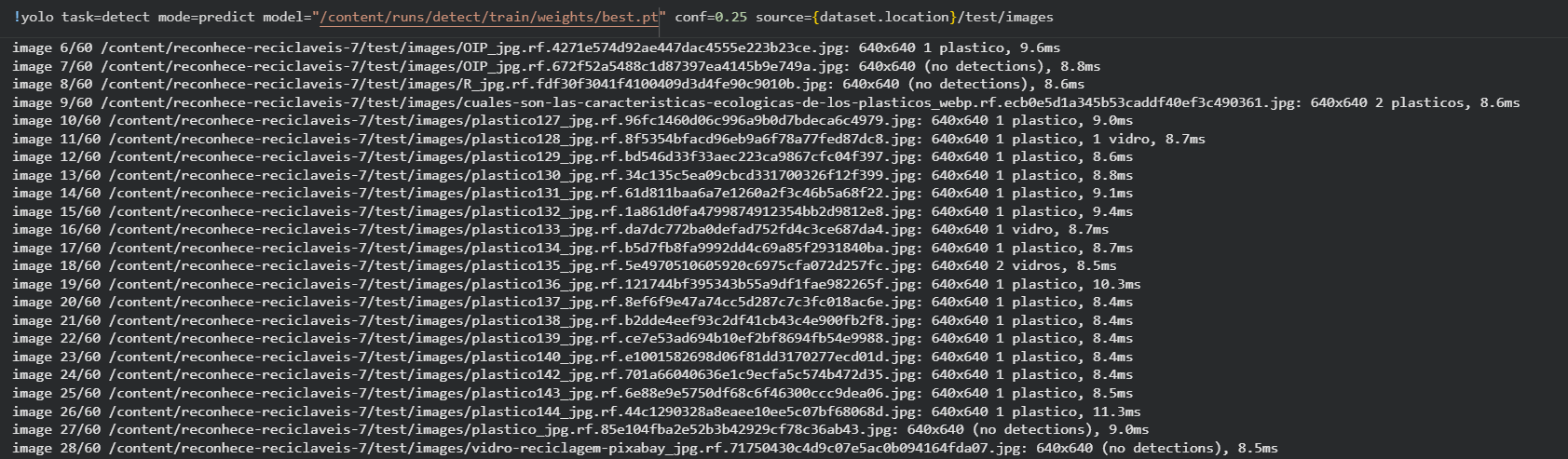
* **Algoritmo 1 (Byte)**
  + Executou em tempos médios entre **512–618 ms** com **0,5 GB de memória**.
  + Apresentou os **menores custos por execução**, variando entre **4,27×10⁻⁶ e 5,15×10⁻⁶ USD**.
  + Demonstrou maior eficiência em cenários com imagens de até ~755 KB, mantendo baixo consumo de recursos.
* **Algoritmo 2 (Download)**
  + Obteve tempos de execução semelhantes, mas com maior variação de memória e imagens maiores (até 7061 KB).
  + Os custos por execução foram superiores, chegando a **9,27×10⁻⁶ USD** em imagens médias e até **5,0×10⁻⁵ USD** em imagens grandes (~22 MB).
  + Apesar de robusto, mostrou-se menos eficiente em termos de custo-benefício.

Com base nos resultados, o **Algoritmo 1 (Byte)** foi considerado o mais adequado para implementação no Lambda *TrataImg*. Ele apresentou o melhor equilíbrio entre **tempo de execução**, **uso de memória** e **custo por execução**, especialmente em cenários de imagens menores, que representam a maioria dos casos de uso do sistema. Essa escolha garante maior escalabilidade e menor impacto financeiro em larga escala, além de manter a qualidade necessária para o processamento posterior no modelo de reconhecimento.

# 8. Implementação da solução

*Figura 5. Código utilizado para treinamento – Fonte: Autoral*

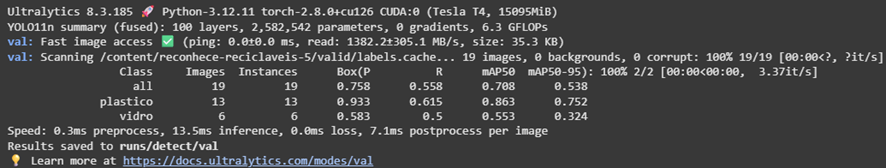
Para ambas as versões do modelo, foi utilizado o comando acima para realizar o treinamento em nosso dataset, tendo padronizado a quantidade de épocas (epochs) para 500 e o tamanho da imagem 640x640 pixels.

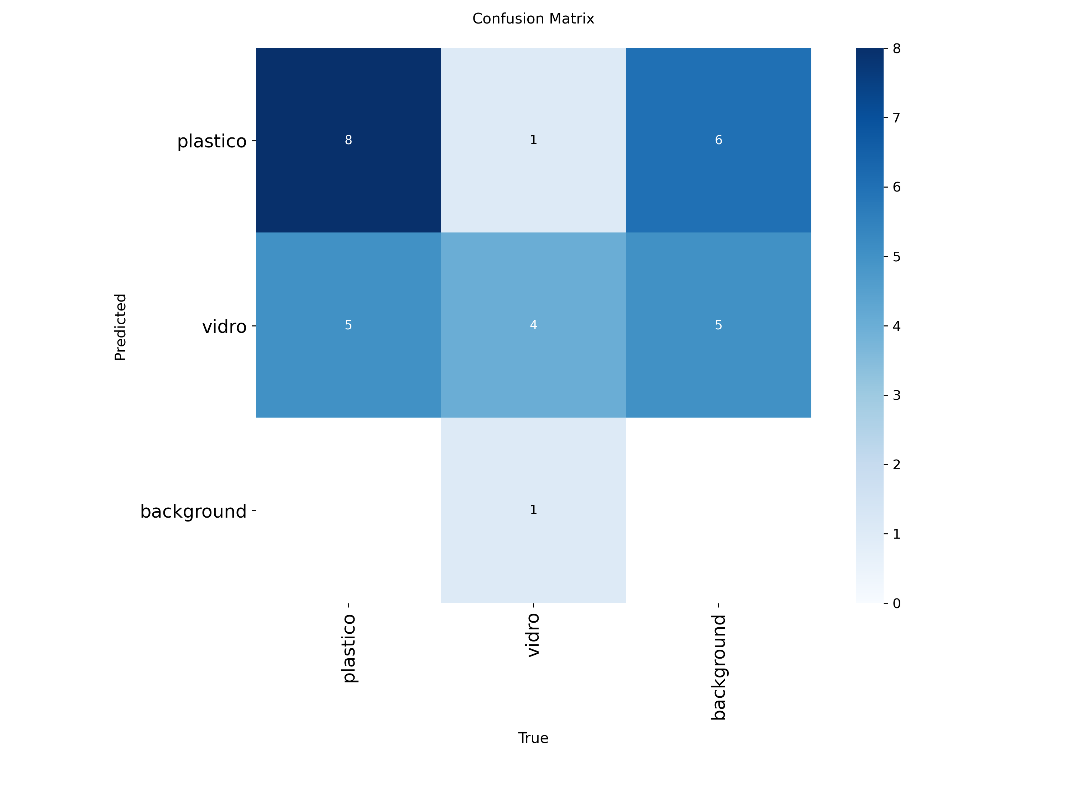
*Figura 6. Código que executa etapa de teste e inferência. Fonte: Autoral*

## **8.1 - 1ª versão do modelo:**

O modelo da Versão 1.0 foi configurado para um total de **500 épocas** de treinamento. Contudo, a execução foi interrompida automaticamente na **época 223**, utilizando-se a técnica de **Parada Antecipada (*Early Stopping*)**. Esta decisão de interrupção prematura baseou-se na análise da curva de **Perda de Validação (*Validation Loss*)**, que demonstrou uma estabilização da performance.

A estabilização precoce, antes da conclusão das 500 épocas, é um forte indicador de **fácil convergência**, sugerindo que o modelo rapidamente atingiu sua capacidade máxima de aprendizado com o *dataset* inicial. O **melhor resultado (*best checkpoint*)** foi, na verdade, registrado na **época 123**, momento em que o modelo apresentou a menor taxa de erro (*Loss*) e a mais alta métrica de **mAP (*mean Average Precision*)** no conjunto de validação. Este checkpoint foi selecionado para evitar um potencial **superajuste (*overfitting*)** que poderia comprometer a capacidade de generalização do modelo em dados não vistos.

*Figura 7. resultados do treinamento da versão 1 do modelo – Fonte: Autoral*

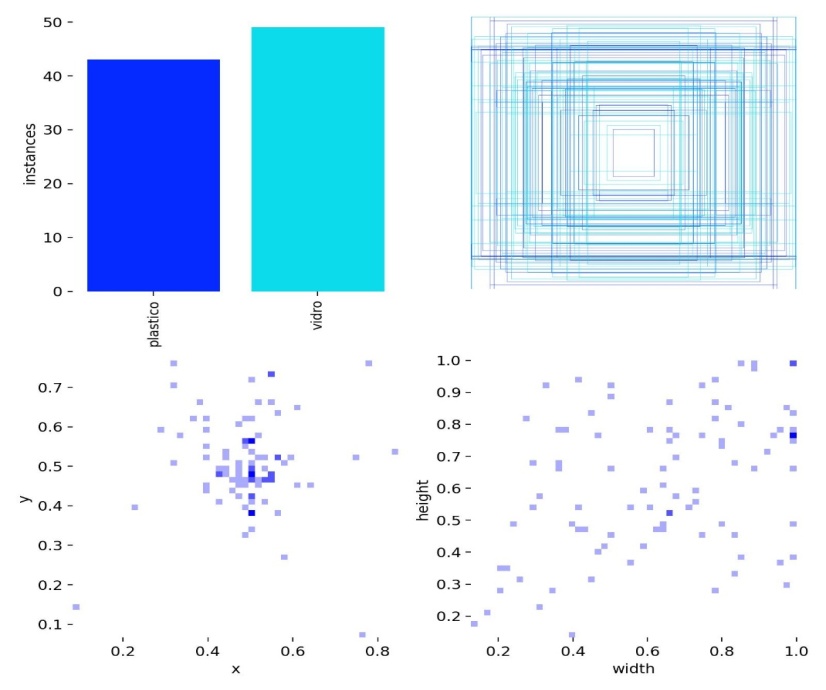


*Figura 8. matriz de confusão gerada pelo treinamento da versão 1 do modelo. - Fonte Autoral*

**Recall:**

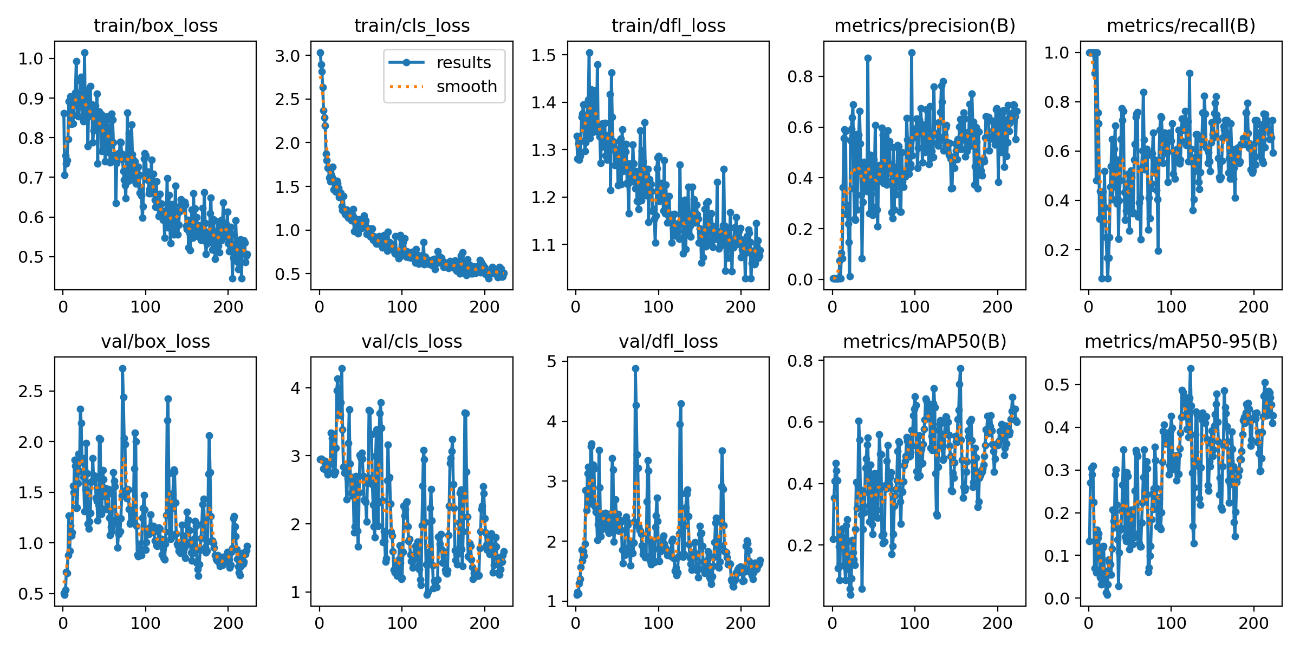
Plástico: 61.54%

Vidro: 66.67%



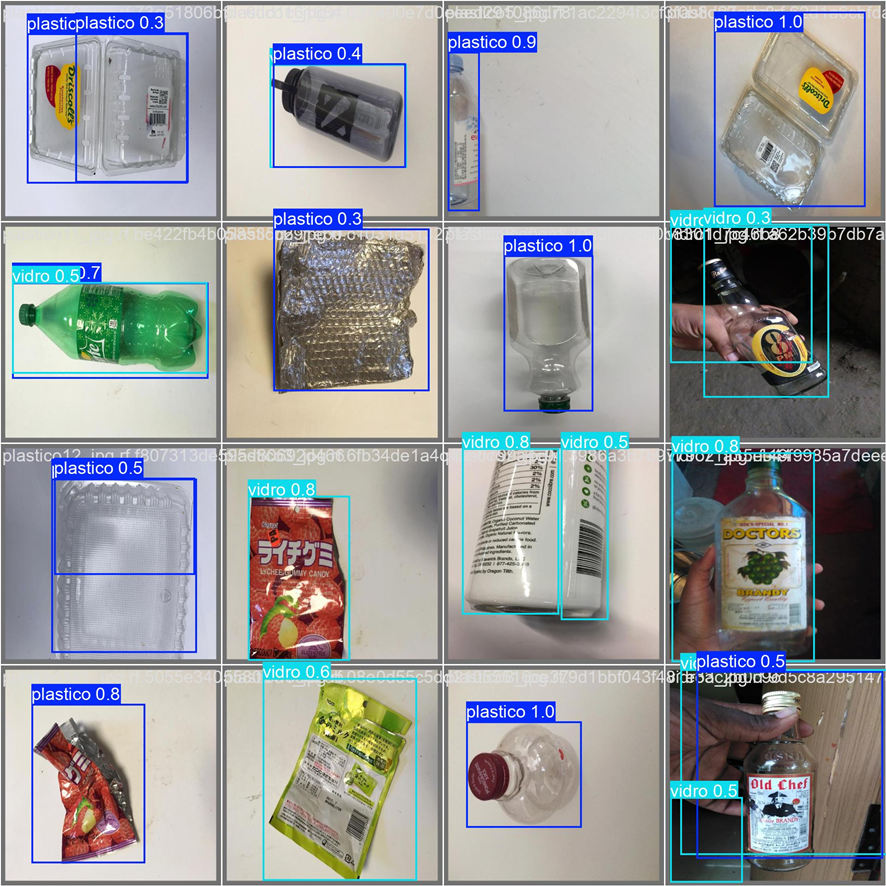
*Figura 9. Imagem contendo 4 gráficos distintos referentes a identificação e posição onde se encontram os objetos do modelo 1 – Fonte: Autoral*

Os gráficos gerados na imagem [9] mostram que mais de 40 instancias foram classificadas como vidro ou plástico. Elas também mostram os bounding boxes detectados das amostras são de maioria no centro da imagem.



*Figura 10. Resultados das métricas de treinamento e validação da versão 1º do modelo – Fonte Autoral*

Na figura [10] temos os resultados do nosso modelo, mostrando acurácia, perda e sensibilidade. Estes tiveram resultados mostram uma tendência constante de queda a medida das iterações, isso fez com que o modelo chegasse ao resultado assertivo mais rápido.



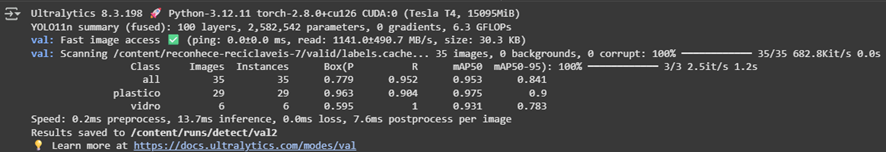
*Figura 11. Resultado das figuras identificadas da 2ª versão do modelo. Fonte: Autoral*

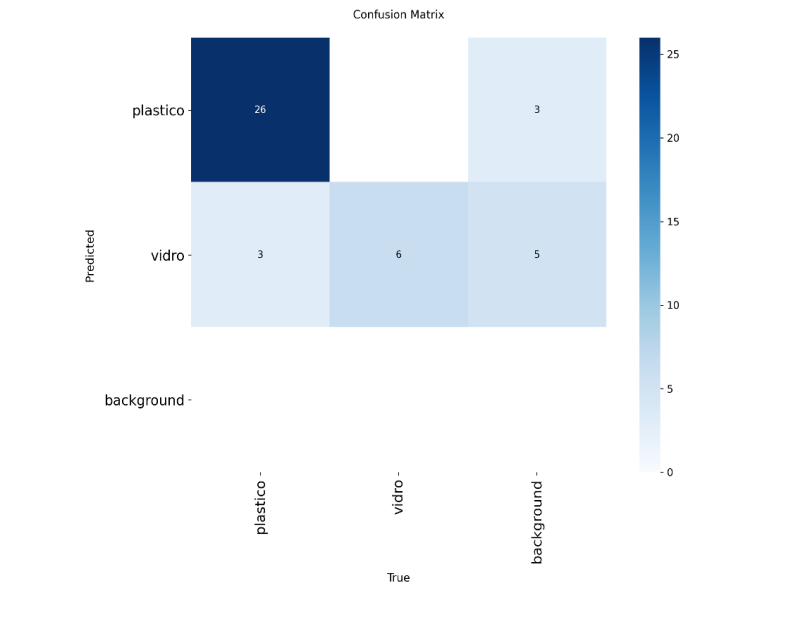
Na figura [11], foi feito um teste da validação e inferência do modelo com seleção de um conjunto de imagens de testes. Com isso a imagem retorna o desempenho do modelo com imagens em diferentes cenários, com diferentes tamanhos e cores.

* **8.2 - 2ª versão do modelo:**

Para a **segunda versão (V2.0)** do modelo, a estratégia de treinamento foi ajustada para incorporar um **volume de dados significativamente maior e uma complexidade aprimorada**, resultante do Data Augmentation e da diversidade do *dataset*. Como esperado em cenários de Deep Learning com conjuntos de dados mais robustos, o modelo exigiu um período de treinamento estendido para absorver a nova complexidade e generalizar o aprendizado, levando a uma convergência mais lenta.

Configurado, assim como a V1.0, para um total de **500 épocas**, a estabilidade do modelo V2.0 foi alcançada somente na **época 440**, refletindo a maior dificuldade em otimizar a função de perda em um espaço de dados mais denso. O **melhor resultado (best checkpoint)**, caracterizado pela combinação ideal entre as métricas de mAP e *Loss* no conjunto de validação, foi registrado na **época 340**. Este desempenho superior demonstrou que o aumento na diversidade e no volume do *dataset* contribuiu para uma curva de aprendizado mais longa, mas que resultou em um modelo final mais **estável, robusto e com maior capacidade de generalização** em comparação com a versão inicial.

*Figura 12. resultados do treinamento da versão 2 do modelo – Fonte: Autoral*

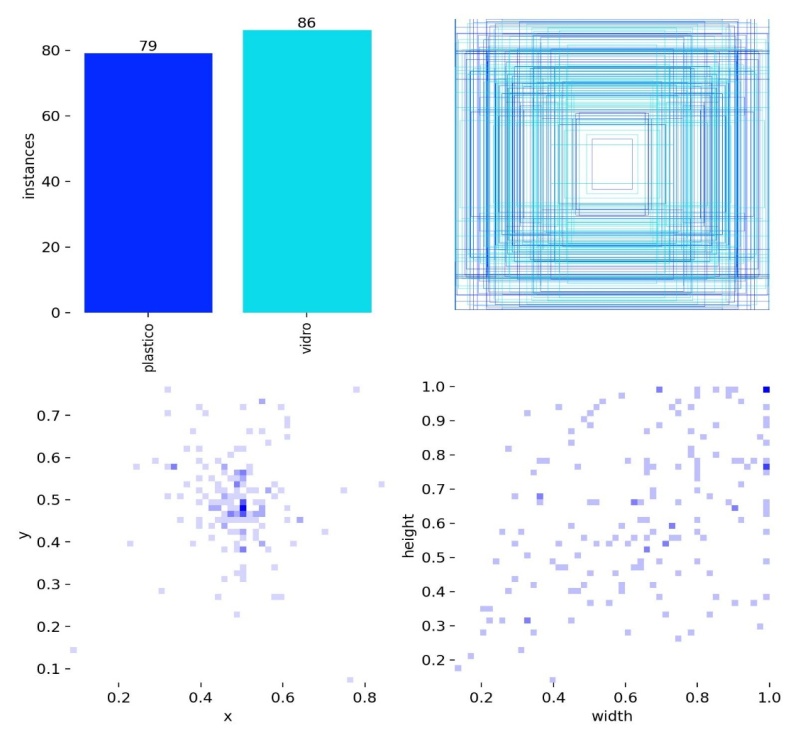


*Figura 13. matriz de confusão gerada pelo treinamento da versão 2 do modelo. - Fonte Autoral*

**Recall:**

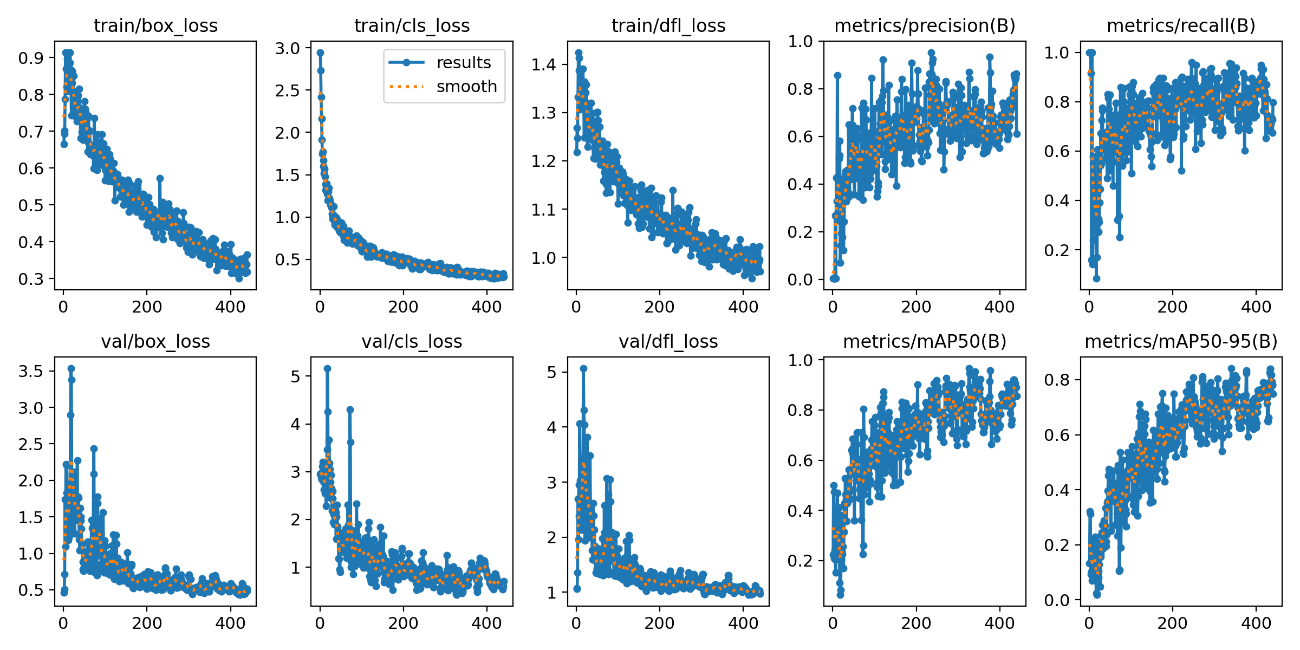
Vidro: 89.66%

Plastico: 66.67%



*Figura 14. Imagem contendo 4 gráficos distintos referentes a identificação e posição onde se encontram os objetos do modelo 2 – Fonte: Autoral*

Os gráficos gerados na imagem [14] mostram que 79 instancias foram classificadas como plastico e 86 como vidro. Elas também mostram que os bounding boxes também apresentam em sua maioria no centro da imagem, isso ocorreu novamente pois foram utilizadas as mesmas amostras da versão anterior do modelo.



*Figura 15. Resultados das métricas de treinamento e validação da versão 2º do modelo – Fonte Autoral*

O modelo teve uma queda maior ainda de erros sendo de 0,3 na hora do treinamento. Além disso, também teve uma melhora no desempenho, aonde no modelo anterior à tendência chegava a 0,5 e no novo modelo ajustado, chegou a bater 0,8.



*Figura 16. Resultado das figuras identificadas da versão 2º do modelo. Fonte: Autoral*

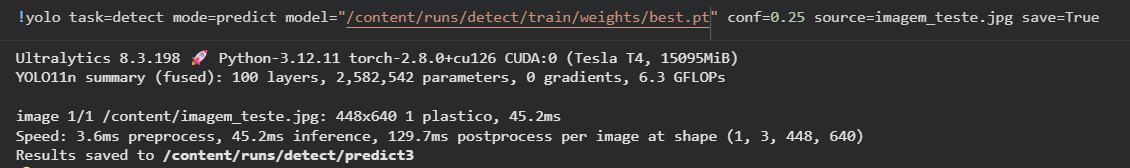
Foi submetido novamente o modelo ao mesmo teste da validação e inferência com a mesma quantidade de dados. Dessa forma a maior parte teve um resultado melhor do que anteriormente como mostrado na figura [15].

* **8.3 - Testando o modelo usando imagem específica:**

Nesta etapa de avaliação, foi conduzido um teste de inferência qualitativa utilizando uma amostra de imagem real não pertencente aos conjuntos de treinamento ou validação. Este procedimento visa simular as condições operacionais encontradas em um centro de reciclagem, permitindo verificar a capacidade de generalização e a robustez do modelo mais atual em um ambiente prático.



*Figura 17. Imagem a ser utilizada. Autor: Volodymyr Shtun*

*Figura 18. Comando utilizado para tentar identificar imagem. Fonte: Autoral*

Na figura [18] o comando está submetendo a imagem [17] ao nosso modelo treinado, ele usa um limite de confiança igual a 0.25, para garantir a sensibilidade máxima na identificação, capturando o maior número possível de Plástico e Vidro para posterior processamento.



*Figura 19. Imagem detectada pelo atual modelo. Fonte: Autoral*

Na imagem [19], vemos que o modelo conseguiu com sucesso encontrar o plástico e classificou ele como 88% de chance de ser realmente um plástico. Após isso foi submetido a mesma figura [17] no modelo anterior para fazermos então a comparação entre ambos e o resultado foi que o modelo retornou que a imagem contém cerca de **82% de confiança de ser um plástico**, resultado contém na imagem [19]. Sendo assim um acréscimo de **6%** entre a versão anterior para a versão atual. Isso pode parecer pouco, mas para um cenário real de uma coletora de lixos onde existem diversas variáveis, um aumento desses é algo que realmente vai fazer uma enorme diferença ao tentar detectar um plástico.



*Figura 20. Imagem detectada pela versão anterior do modelo. Fonte: Autoral*

# 9. Conclusão

O desenvolvimento deste estudo evidenciou o potencial da inteligência artificial e da visão computacional com ferramentas capazes de transformar significativamente os processos de triagem de materiais recicláveis em centros de reciclagem. A solução proposta, construída com base no modelo YOLOv11 e apoiada por uma base de dados, demonstrou resultados expressivos na identificação de plástico e vidro, reforçando a viabilidade da automação nesse contexto.

A comparação entre as versões 1.0 e 2.0 do modelo revelou avanços consistentes na performance. A segunda versão, alimentada por um conjunto de dados mais amplo e variado, apresentou aumento relevante nas métricas de recall, especialmente para a classe vidro, que passou de 66,67% para 89,66%. Essa evolução comprova que o aumento da complexidade e diversidade do dataset contribui diretamente para a capacidade de generalização e precisão do modelo.

Além dos aspectos técnicos, o projeto reafirma a importância da automação como estratégia que beneficia não apenas a eficiência operacional, mas também dimensões sociais e ambientais. A redução de erros, o aumento da velocidade de triagem e a possibilidade de realocação de trabalhadores para funções menos insalubres tornam o sistema proposto uma alternativa sustentável e socialmente responsável.

Dessa forma, conclui-se que a solução de detecção automatizada de materiais recicláveis desenvolvida neste trabalho é tecnicamente viável, apresenta desempenho satisfatório e possui potencial real de aplicação prática em centros de reciclagem. Com aprimoramentos adicionais, como expansão do dataset para outras classes de resíduos e integração com sistemas robóticos de separação física, a proposta pode evoluir para um sistema completo e escalável, contribuindo efetivamente para o avanço da economia circular e a modernização da gestão de resíduos no Brasil.

# 10. Referências

ABRELPE - Associação Brasileira de Empresas de Limpeza Pública e Resíduos Especiais. Panorama dos Resíduos Sólidos no Brasil 2023. São Paulo: ABRELPE, 2023.

CALDERONI, Sabetai. Os bilhões perdidos no lixo. 4. ed. São Paulo: Humanitas, 2003.

GONÇALVES, A. P. Automação no Processo de Triagem de Resíduos Sólidos Urbanos. *In*: CONGRESSO BRASILEIRO DE ENGENHARIA SANITÁRIA E AMBIENTAL, 30., 2019, Natal. Anais [...]. Rio de Janeiro: ABES, 2019.

XAVIER, Y. M. A.; CORRÊA, H. L. S. Análise de Viabilidade Econômica da Automação de Processos em Cooperativas de Reciclagem. Revista de Gestão e Sustentabilidade Ambiental, v. 10, n. 2, p. 210-225, 2021.

MITCHELL, Tom M. *Machine Learning*. New York: McGraw Hill, 1997.

REDMON, Joseph; FARHADI, Ali. YOLOv3: An Incremental Improvement. 2018. ArXiv:1804.02767. (Referência para modelos YOLO)

RUSSAKOVSKY, Olga et al. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision*, 2015. (Referência para métricas de visão computacional).

**ASSINATURAS**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Acadêmico: Nome do Acadêmico**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Orientador: Nome do Orientador**

**São Paulo/2025**