Universidade de Brasília
Faculdade de Tecnologia

Detecção e Reconhecimento de Resíduos Sólidos em Ambientes Não-Estruturados Utilizando Modelos de Aprendizado Profundo

Luís Humberto Chaves Senno
David Fanchic Chatelard

TRABALHO DE GRADUAÇÃO
ENGENHARIA DE CONTROLE E AUTOMAÇÃO

Brasília
2023

Universidade de Brasília
Faculdade de Tecnologia

Detecção e Reconhecimento de Resíduos Sólidos em Ambientes Não-Estruturados Utilizando Modelos de Aprendizado Profundo

Luís Humberto Chaves Senno
David Fanchic Chatelard

Trabalho de Graduação submetido como requisito parcial para obtenção do grau de Engenheiro de Controle e Automação

Orientador: Prof. Dr. Flávio de Barros Vidal

Brasília
2023

Chaves Senno, Luís Humberto.
C512d Detecção e Reconhecimento de Resíduos Sólidos em Ambientes Não-Estruturados Utilizando Modelos de Aprendizado Profundo / Luís Humberto Chaves Senno; David Fanchic Chatelard; orientador Flávio de Barros Vidal. -- Brasília, 2023.
55 p.

Trabalho de Graduação (Engenharia de Controle e Automação) -- Universidade de Brasília, 2023.

1. YOLO. 2. Classificação de Resíduos Sólidos. 3. TACO. I. Fanchic Chatelard, David. II. Barros Vidal, Flávio de, orient. III. Título

Universidade de Brasília
Faculdade de Tecnologia

Detecção e Reconhecimento de Resíduos Sólidos em Ambientes Não-Estruturados Utilizando Modelos de Aprendizado Profundo

Luís Humberto Chaves Senno
David Fanchic Chatelard

Trabalho de Graduação submetido como requisito parcial para obtenção do grau de Engenheiro de Controle e Automação

Trabalho aprovado. Brasília, 04 de Julho de 2023:

Prof. Dr. Flávio de Barros Vidal,
CIC/IE/UnB
Orientador

Prof. Dr. Marcelo Ladeira,
CIC/IE/UnB
Examinador interno

Prof. Dr. Marcus Vinícius Lamar,
CIC/IE/UnB
Examinador interno

Brasília
2023

Este trabalho é dedicado à minha família que me trouxe até aqui e é a razão de eu ser tudo que sou.

Luís Humberto Chaves Senno

Dedico este trabalho à minha mãe, namorada e amigos que sempre estiveram comigo nos momentos que mais precisei.

David Fanchic Chatelard

Agradecimentos

Gostaria de agradecer a minha família que sempre me deu todo o apoio e carinho que eu precisei. Principalmente minha mãe, que sempre foi além pra me ajudar em tudo.

Queria agradecer também aos meus colegas de curso que entraram junto comigo no semestre 01/2018, que deixaram essa jornada mais tranquila e prazerosa. Com isso, gostaria de fazer um agradecimento especial aos macacos do zoológico de brasília que sempre me ajudaram nos piores momentos.

Gostaria de agradecer os outros colegas e companheiros que eu fiz enquanto frequentava a universidade, que mudaram minha vida de diversas maneiras.

Por fim, gostaria de agradecer meus amigos mais próximos tanto aqueles que me conhecem desde tempos imemoriais, com isso agradeço novamente meu irmão o maior deles, quanto aqueles que fiz durante meu ensino médio. Aqueles que eu sei que posso contar com em qualquer momento.

São muitas outras pessoas que merecem o meu agradecimento e infelizmente muito pouco espaço para isso, mas muito obrigado a todos que participaram dessa jornada de alguma maneira.

Luís Humberto Chaves Senno

Primeiramente gostaria de agradecer à minha mãe que sempre me apoiou em todos os aspectos da minha vida e me ensinou a ser uma boa pessoa e a ter valores.

Agradeço à minha namorada que sempre esteve comigo nos momentos bons e ruins, me ensinou tanto sobre a vida e me ajudou a superar todos os momentos difíceis ao longo da graduação.

Agradeço também aos meus amigos que são grandes companheiros e sempre me ajudaram a esquecer dos problemas e me divertir. Por fim, mas não menos importante, agradeço ao meu querido grupo de amigos que fiz na faculdade: Marcos Eduardo, Alexandre Pinto, Emanuel Couto, Gabriel Tambara e Luís Humberto. A companhia de vocês foi essencial para a minha graduação, vocês foram ótimos companheiros, sem vocês não seria possível ter concluído este curso e espero levar a amizade de vocês para o resto da minha vida.

David Fanchic Chatelard

“Não existe homem vivo que
não seja capaz de fazer mais
do que pensa que pode.”
(Henry Ford)

Resumo

A Inteligência Artificial (IA) desempenha um papel cada vez mais crucial na área da sustentabilidade, oferecendo soluções inovadoras e ajudando a enfrentar os desafios ambientais. Nesta linha, este trabalho tem como objetivo apresentar o desenvolvimento de uma metodologia para realizar a detecção e reconhecimento de resíduos sólidos a partir de imagens. Para isso, foram utilizados conjuntos de dados contendo diversas imagens de resíduos sólidos que estão disponíveis publicamente, sendo o principal deles o conjunto de imagens da *Trash Annotations in Context* (TACO) que possui 60 tipos de resíduos sólidos distintos. Além do treinamento no conjunto de imagens, foi realizada a comparação entre os principais modelos de aprendizagem profunda da arquitetura *You Only Look Once* (YOLO) versão 7, aplicando técnicas de otimização (transferência de aprendizagem e aumento de dados manuais e artificiais). Dentre os resultados obtidos, estes comparando com trabalhos do estado-da-arte, alcançaram valores de melhorias de até 119% para o mesmo conjunto de imagens da base TACO.

Palavras-chave: YOLO. Classificação de Resíduos Sólidos. TACO.

Abstract

Artificial Intelligence (AI) plays an increasingly crucial role in the field of sustainability, offering innovative solutions and helping address environmental challenges. Accordingly, this study aims to present the development of a methodology for detecting and recognizing solid waste from images. To achieve this, publicly available datasets containing various images of solid waste were used, with the main one being the Trash Annotations in Context (TACO) image dataset, which includes 60 distinct types of solid waste. In addition to training on the image dataset, a comparison was made among the main deep learning models of the You Only Look Once (YOLO) architecture (version 7), applying optimization techniques such as transfer learning and data augmentation. The results obtained, when compared to state-of-the-art works, achieved improvement values of up to 119% for the same set of images from the TACO dataset.

Keywords: YOLO. Littering Classification. TACO.

Lista de ilustrações

Lista de tabelas

Lista de abreviaturas e siglas

CNN	*Convolutional Neural Network* 17
FPN	*Feature Pyramid Network* 22
FPS	*Frames* Por Segundo 30
IA	Inteligência Artificial 7
IoU	*Intersection over Union* 26
mAP	*Mean Average Precision* 24
MS COCO	*Microsoft Common Objects in Context* 35
TACO	*Trash Annotations in Context* 7
VAE	Autoencoder Variacional 30
YOLO	*You Only Look Once* 7

Sumário

1 Introdução

1.1 Motivação

O descarte inadequado de resíduos sólidos é um problema ambiental urgente, já que além de causar poluição ambiental também diminui a qualidade de vida dos habitantes da região. Além disso, projeções mostram que a geração de resíduos sólidos urbanos nas principais cidades metropolitanas em todo o mundo aumentará de 1,3 bilhões de toneladas em 2012 para 2,2 bilhões em 2025 (HOORNWEG DANIEL; BHADA-TATA, 2012), o que só ira exacerbar a situação.

Figura 1.1 – Acúmulo de resíduo sólido nas cidades.

Fonte: Ambiente Legal (2016)

Os resíduos sólidos, como o lixo, têm efeitos danosos significativos no meio ambiente. O acúmulo descontrolado de resíduos contribui para a poluição do solo, da água e do ar. A decomposição dos resíduos sólidos orgânicos libera gases de efeito estufa, como metano, que contribuem para o aquecimento global. Além disso, muitos materiais descartados, como plásticos, demoram anos, ou até mesmo séculos, para se decompor, resultando em poluição duradoura. Esses resíduos também representam uma ameaça à fauna e à flora, pois animais podem se alimentar de detritos plásticos, causando danos à saúde e, em alguns casos, levando à morte. A contaminação de aquíferos e corpos d'água por produtos químicos tóxicos presentes no lixo também representa um risco para os ecossistemas aquáticos e para a saúde humana.

Como uma das principais contribuições da IA na sustentabilidade é sua capacidade de processar e analisar grandes volumes de dados é possível a identificação de padrões, a previsão de tendências e a ajuda na tomada de decisões baseadas em evidências, consequentemente

melhorando o gerenciamento de resíduos sólidos urbanos e a recuperação, por meio da reciclagem, de materiais de valor (GUNDUPALLI; HAIT; THAKUR, 2017).

Nesta linha, o processo de identificação de resíduos sólidos (ex.: lixo urbano) pode se tornar uma tarefa de grande relevância para aprimoramento da sustentabilidade.

Desta feita, a criação de ferramental utilizando IA para a detecção de resíduos sólidos é de extrema importância. Então, este trabalho tem como objetivo apresentar o desenvolvimento de uma metodologia para realizar a detecção e reconhecimento de resíduos sólidos a partir de imagens.

1.2 Contextualização

A Inteligência Artificial (IA) é um campo de estudo que busca criar sistemas capazes de realizar tarefas abstratas ou intuitivas que, até então, só eram possíveis para seres humanos, seres inteligentes. Por isso, o campo de estudos de IA é variado e extenso e, entre tantos subcampos, o campo de Aprendizado de Máquina tem se destacado e ganhado muito impulso na última década, principalmente devido ao aumento da capacidade de processamento e da possibilidade de uso de grandes quantidades de dados, proporcionados pelo avanço tecnológico (GOODFELLOW; BENGIO; COURVILLE, 2016). Aprendizado de Máquina é uma técnica que permite que as máquinas aprendam a partir de dados sem serem explicitamente programadas, o que tem permitido a criação de sistemas avançados com aplicações em diversos setores, desde saúde e finanças até jogos e recomendações. Apesar de já ser um dos ramos da Inteligência Artificial, Aprendizado de Máquina ainda é um campo extenso e uma área que é particularmente interessante para problemas de classificação, segmentação de imagem e detecção de objetos em imagens é o Aprendizado Profundo.

O Aprendizado Profundo, uma subárea do Aprendizado de Máquina, utiliza redes neurais profundas para modelar e aprender padrões complexos a partir de dados, o que permite a criação de sistemas avançados.

Um dos modelos de detecção de objetos baseados em Aprendizado Profundo mais famosos é o *You Only Look Once* (YOLO) (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022). Ele foi introduzido em 2016 e atualmente, sua versão mais atual é YOLOv7, sendo um modelo que tem se tornado cada vez mais popular devido a sua eficiência e rapidez. A principal vantagem do YOLO é que ele realiza a detecção de objetos em uma única passada pela imagem, ao invés de dividir a imagem em pequenos blocos e analisá-los separadamente como em outros modelos. Isso permite que ele seja muito rápido em comparação com outros modelos, o que o torna adequado para aplicações em tempo real, como condução autônoma, segurança ou análise de vídeos feitos pelo celular para um aplicativo.

Outro fator importante que deve ser levado em consideração em soluções de Apren-

dizado de Máquina e consequentemente de Aprendizado Profundo é o conjunto de imagens ou o banco de dados usado para o treinamento do modelo. Com isso, tendo em mente o problema de detecção de resíduos sólidos, um bom ponto de início será o conjunto de dados *Trash Annotations in Context* (TACO), (PROENÇA; SIMÕES, 2020), que é uma coleção de imagens de resíduo sólido e resíduos, classificadas de acordo com seus tipos. O conjunto de imagens foi criado justamente com o objetivo de ajudar a desenvolver sistemas de visão computacional para a classificação de resíduos. A coleção de imagens é bem documentada e amplamente utilizada para treinar e avaliar modelos de aprendizado de máquina, incluindo o YOLO, para a detecção e classificação de resíduo sólido.

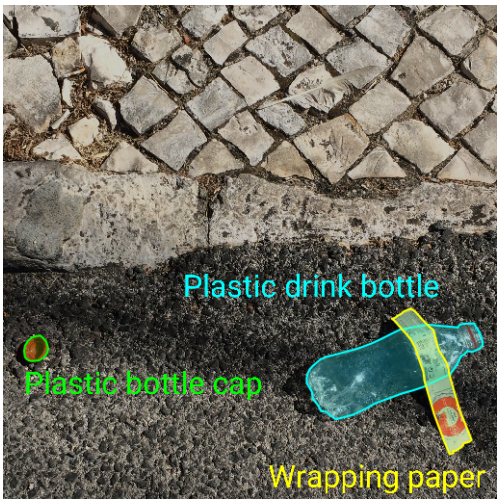

Figura 1.2 – Exemplo de classificação de resíduo sólido da TACO.

Fonte: Proença e Simões (2020)

Em resumo, a combinação da Inteligência Artificial, especificamente o Aprendizado Profundo no modelo YOLO, com o conjunto de imagens de resíduo sólido da TACO possibilita a criação de uma solução poderosa para a detecção e classificação de resíduos. Esse sistema em troca pode ser usado para ajudar na gestão de resíduos e na tomada de decisões.

1.3 Objetivos

Esse trabalho, tem como principal objetivo avaliar e aprimorar o processo de detecção e reconhecimento de imagens de resíduos sólidos a partir de um modelo de aprendizado profundo em diferentes tipos de resíduos sólidos, assim como analisar e comparar com trabalhos de estado-da-arte atualmente desenvolvidos. Para isso, é possível destacar os principais objetivos secundários, que devem ser atingidos, a saber:

- Definição do modelo de arquitetura de Aprendizado Profundo para a detecção de resíduos sólidos em imagens;

- Elaborar o estudo das principais bases públicas de imagens contendo resíduos sólidos;
- Apresentar protocolo de avaliação e comparação dos resultados obtidos pelo modelo escolhido em relação aos principais trabalhos do estado-da-arte na detecção e reconhecimento de resíduos sólidos.

1.4 Organização do Trabalho

Este trabalho é composto de seis capítulos: O Capítulo 2 apresenta conhecimentos pertinentes à área de IA, utilizados no desenvolvimento deste trabalho, como: o uso de Aprendizado Profundo, os modelos usados e como é resolvido um problema de detecção de objetos. Já o Capítulo 3 trata da discussão dos trabalhos relacionados já feitos e quais suas influências sobre o trabalho atual. No Capítulo 4 é apresentada a metodologia utilizada na realização do trabalho, detalhando todas as ideias, processos e ferramentas usadas e o que será realizado no restante do projeto. O Capítulo 5 expõe os resultados obtidos ao longo do desenvolvimento do projeto, apresentando comparações que comprovam a eficiência da pesquisa realizada. Por fim, o Capítulo 6 possui a conclusão deste trabalho, apresentando uma discussão sobre os resultados obtidos e sobre o que pode ser feito no futuro para melhorar ainda mais os resultados finais.

2 Fundamentação Teórica

Conforme mencionado em (JIA et al., 2023), a área de Inteligência Artificial, mais especificamente da visão computacional, tem apresentado grandes avanços ao longo dos últimos anos, em grande parte devido ao desenvolvimento e aprimoramento de técnicas de Aprendizado Profundo. Uma das subáreas em destaque é a detecção de objetos em imagens e vídeos (SALAS; BARROS VIDAL; MARTINEZ-TRINIDAD, 2019; ALMEIDA; BARROS VIDAL, 2021). Um dos modelos de detecção de objetos mais usado atualmente é o *You Only Look Once* (YOLO) (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022), que se destaca pela sua rapidez e principalmente precisão em detectar objetos em aplicações de tempo real. Além de um bom modelo, existem várias técnicas, como por exemplo o aumento de dados, que podem auxiliar em problemas de detecção de objetos e que tem sido usadas não só para esse tipo de problema, mas para melhorar o desempenho de diversos modelos de Aprendizado Profundo, ao gerar conjuntos de dados sintéticos aumentando a quantidade de dados usados no treinamento, como evidenciado em (PELLICER; FERREIRA; COSTA, 2023). Neste capítulo de fundamentação teórica serão abordados os principais conceitos sobre Aprendizado Profundo, detecção de objetos, modelos de Redes Neurais Convolucionais (CNN), o modelo YOLO e a técnica de aumento de dados para que seja estabelecida uma base teórica para o desenvolvimento de um modelo de detecção de objetos em imagens.

2.1 Aprendizado Profundo

Uma das subáreas de Aprendizado de Máquina é o Aprendizado Profundo, que se utiliza de redes neurais profundas com o objetivo de aprender, a partir de dados de entrada, a realizar tarefas como classificação, regressão e segmentação de imagens, (GOODFELLOW; BENGIO; COURVILLE, 2016). Tendo em vista estas tarefas, é possível utilizar o Aprendizado Profundo para: análise de imagens e vídeos, processamento de linguagem natural, reconhecimento de fala, tradução automática, entre outras. O Aprendizado Profundo é baseado em múltiplas camadas de unidades de processamento que se comunicam entre si compartilhando informações. A Figura 2.3 apresenta um fluxograma mostrando como funcionam diferentes sistemas de inteligência artificial.

De acordo com (WANG, Y.; VINOGRADOV, 2023), as unidades de processamento de cada camada recebem um conjunto de valores de entrada, realizam operações matemáticas com esses valores e produzem um conjunto de valores de saída. E assim, as camadas seguintes recebem esses valores de saída como sendo valores de entrada para poderem realizar as próximas operações matemáticas e produzirem novos valores de saída. O processo continua até que seja produzida a saída final da rede neural.

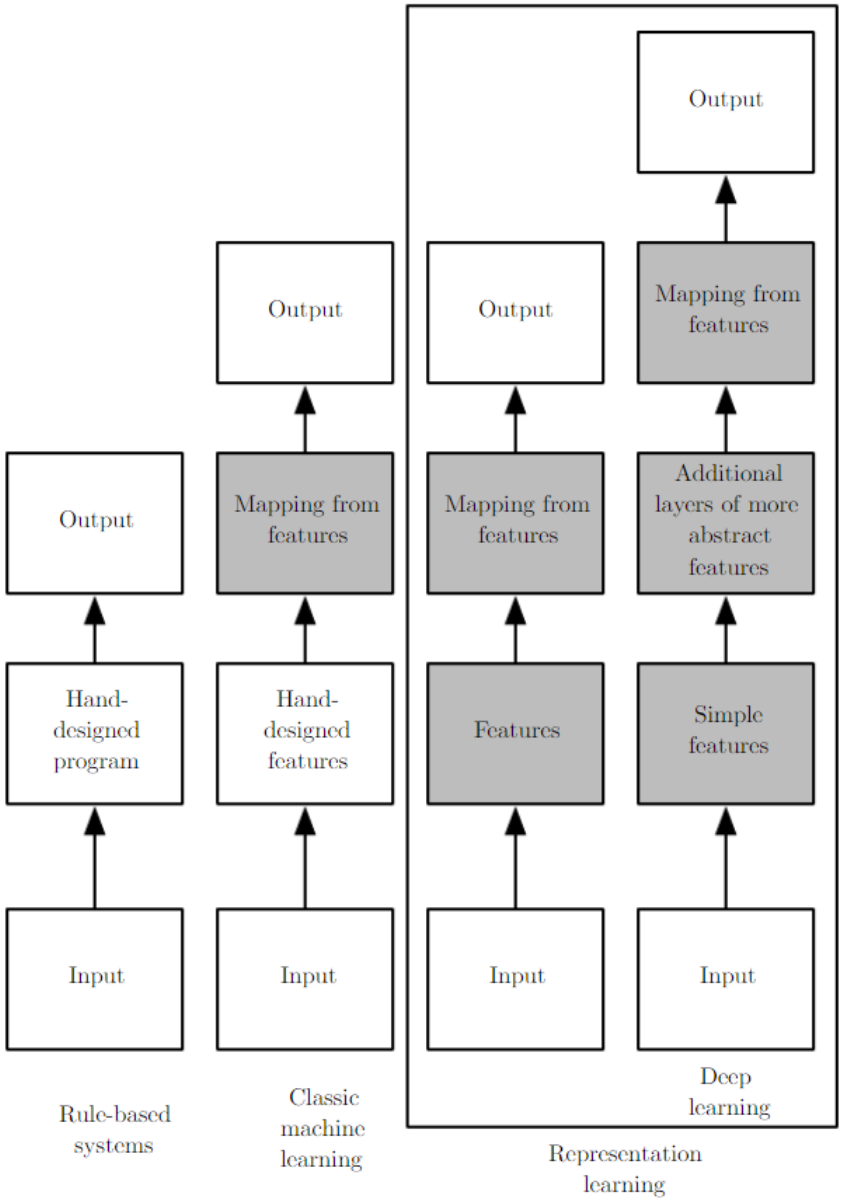

Figura 2.3 – Fluxogramas mostrando como as diferentes partes de um sistema de IA se relacionam entre si em diferentes áreas de IA. As caixas sombreadas indicam componentes que podem aprender com dados.

Fonte: Goodfellow, Bengio e Courville (2016)

Durante o treinamento de uma rede neural, os pesos das conexões entre cada uma das unidades de processamento são ajustados de forma iterativa, usando como base o erro de saída da rede. O treinamento tem como objetivo minimizar este erro para que a rede neural apresente bons resultados independente de quais sejam os dados de entrada, como é apresentado em (GARCÍA-AGUILAR et al., 2023).

Ao se treinar uma rede neural profunda é preciso minimizar a função objetivo (GOODFELLOW; BENGIO; COURVILLE, 2016), que é a função que mede o desempenho da rede em relação aos dados do treinamento. Essa função, também chamada de função de

custo ou de perda, mede a diferença entre as saídas da rede e as saídas que eram esperadas para um conjunto de valores de exemplos de treinamentos. Ao ser treinada, é possível utilizar a rede neural com diferentes dados de entrada.

Uma característica importante do Aprendizado Profundo é a sua capacidade de conseguir aprender de forma não-supervisionada, como demonstrado em (TSENG; JIANG, 2022). Isto é, a rede neural consegue aprender a partir de dados que não possuem etiquetas de identificação. Com isso ela consegue identificar padrões e objetos por conta própria, o que é extremamente útil em casos onde existem muitos dados a serem analisados, porém eles não estão rotulados e o processo de identificação seria muito demorado ou custoso.

A área de IA e do Aprendizado Profundo estão em constante evolução e a partir do desenvolvimento de novas técnicas de treinamento, como por exemplo a regularização e a normalização (GOODFELLOW; BENGIO; COURVILLE, 2016), é possível se construir novos modelos mais eficientes e com desempenho cada vez maior.

2.2 Detecção de Objetos e Modelos de CNN

Um dos problemas mais comuns vistos no campo de visão computacional e aprendizado profundo é a detecção de objetos em imagens. Esse problema é um dos que se enquadra na categoria de saída estruturada (GOODFELLOW; BENGIO; COURVILLE, 2016) e consiste em receber uma entrada, nesse caso uma foto, e emitir uma saída no formato de uma estrutura de dado cuja relação entre elementos distintos é importante, nesse caso a informação sobre a existência ou não de objetos na foto, além da foto com a localização dos objetos detectados. A localização geralmente é dada em forma de caixa delimitadora ou *bounding box* e geralmente podem existir múltiplos objetos de diferentes classes em uma única imagem, assim como nenhum(GOODFELLOW; BENGIO; COURVILLE, 2016).

Dentro do campo de aprendizado profundo, os métodos mais usados para resolver o problema de detecção de objetos e outros problemas envolvendo imagens são modelos chamados de rede neural convolucional (CNN). Esse algoritmo é baseado no padrão de conectividade dos neurônios encontrados no córtex visual de animais (FUKUSHIMA, 1980) (MATSUGU et al., 2003) e são redes neurais especializadas em realizar a operação de convolução.

Como apresentado em (GOODFELLOW; BENGIO; COURVILLE, 2016), uma CNN é composta de várias camadas, onde cada uma possui um conjunto de neurônios que são chamados de filtros (ou *kernels*). Os filtros são convoluídos com as entradas da camada anterior para que seja gerada uma ativação, que será enviada para a camada seguinte. Ao longo do treinamento, os pesos dos filtros são ajustados de forma que a CNN consiga aprender a encontrar características relevantes nos dados de entrada, como por exemplo em imagens. A Figura 2.4 apresenta uma visão abstrata de atuação de um método de classificação de

imagens utilizando CNNs.

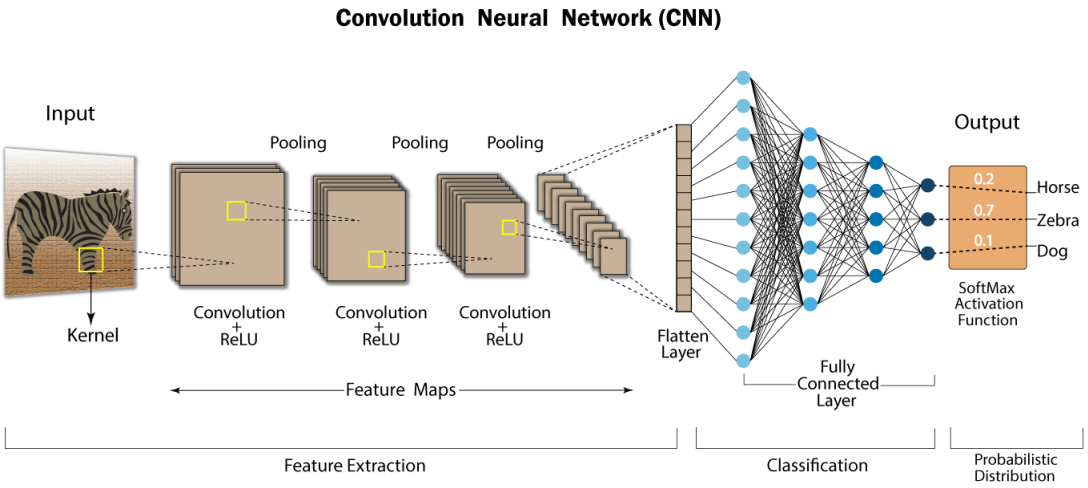

Figura 2.4 – Exemplo de uma rede neural convolucional (CNN).

Fonte: Developers Breach (2020)

Geralmente a primeira camada de uma CNN é a de convolução (WU et al., 2023), que é onde são extraídas as características de baixo nível, como por exemplo: texturas e bordas. É comum utilizar filtros pequenos nessa camada, de tamanho 3x3 ou 5x5. Eles se deslocam pela imagem de entrada para produzir um mapa de características(ou de ativação), que tem como objetivo destacar a presença ou ausência de uma certa característica nas regiões da imagem.

A camada seguinte normalmente é a camada de *pooling* (WU et al., 2023), que é onde o mapa de características tem a sua dimensionalidade diminuída. Para obter essa diminuição são feitas operações com o *max pooling*, que consiste em selecionar o valor máximo em cada região do mapa, assim reduzindo a sua resolução.

Depois dessas duas camadas iniciais é comum que se tenha mais camadas de convolução e *pooling*, para que se possa extrair as características de níveis mais altos da imagem. Para finalizar o processo, é comum haver uma camada final de convolução, onde a sua saída é transformada em um vetor unidimensional que será passado para camadas densas, também chamadas de camadas totalmente conectadas, que irão gerar a saída final da rede neural.

Ao longo do treinamento, a CNN faz o ajuste dos pesos dos filtros a partir de um processo que se chama retro propagação, onde se usa o gradiente descendente com o objetivo de minimizar o erro, ou perda, entre a saída da rede e as etiquetas de treinamento (GOODFELLOW; BENGIO; COURVILLE, 2016). Para se realizar esse processo é calculada a derivada da perda em relação a cada peso da rede neural.

2.3 YOLO

Dentre os modelos de CNN, os mais estabelecidos para solução do problema de detecção de objetos são: YOLO (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022), *Mask* R-CNN (HE et al., 2018), R-CNN (GIRSHICK et al., 2014) e algumas outras variações desses modelos. A YOLO se destaca devido ao baixo tempo de processamento já que realiza a detecção em apenas uma passada, sem separar a imagem. Na Figura 2.5 é possível observar a estrutura geral da arquitetura YOLO.

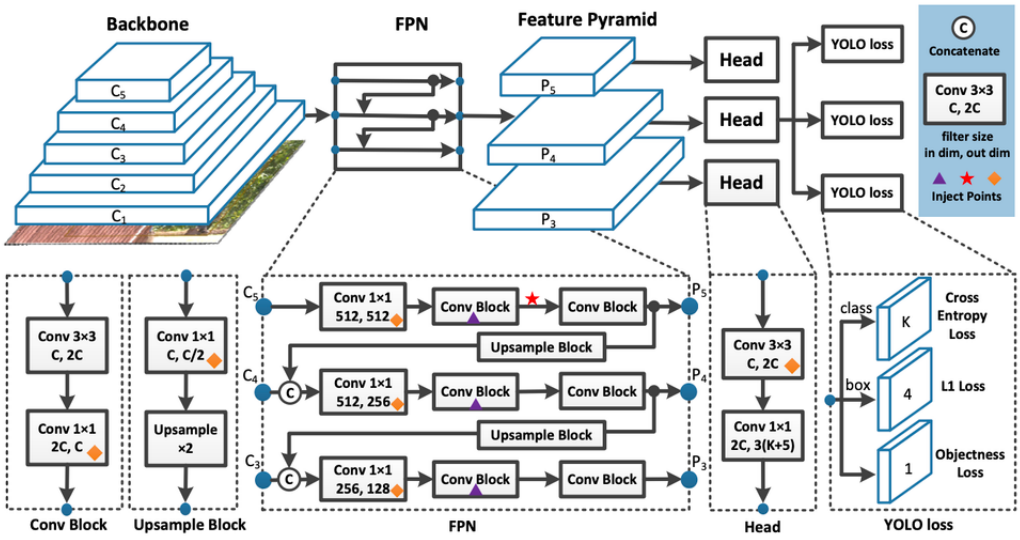

Figura 2.5 – Estrutura da arquitetura YOLOv4.

Fonte: Long et al. (2020)

A estrutura geral pode ser dividida em três partes principais, o *backbone* onde as características dos dados são extraídas, o *neck*, ou pescoço, onde as características extraídas de diferentes camadas são combinadas, denotado na Figura 2.5 por FPN e por fim a cabeça da rede ou, *head*, onde são feitas as predições sobre as caixas de detecção. Após essas três etapas as caixas de detecção passam pela *loss function* que decide quais serão de fato as predições finais do modelo.

No caso da versão YOLOv7 os principais blocos computacionais do *backbone* são os chamados *Efficient Layer Aggregation Networks*(ELAN) (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022), redes que foram desenvolvidas buscando minimizar a deteriorização da convergência que ocorre ao se aumentar a escala do modelo.

A ELAN é composta principalmente por uma combinação de VoVNet (LEE et al., 2019) e CSPNet (WANG, C.-Y.; LIAO; YEH et al., 2019) e funciona analisando os caminhos de propagação do gradiente e tentando evitar que o caminho mais curto de propagação fique mais longo (WANG, C.-Y.; LIAO; YEH, 2022).

Além disso, a YOLO é um modelo baseado em âncoras. Isso significa que a maneira como o modelo faz a detecção de objetos na imagem é desenhando uma grade sobre a imagem e, em seguida, gerando em cada ponto de intersecção(pontos de ancoragem), dessa grade, caixas candidatas(caixas de ancoragem). A mesma quantidade de caixas de ancoragem é repetida para cada ponto de ancoragem, então o que o modelo aprende de fato a fazer é selecionar as caixas mais promissoras e em seguida alterar sua posição e o seu tamanho.

Um problema, porém, é que os objetos a serem detectados podem variar extensamente em tamanho e o modelo deve ser efetivo para todos esses tamanhos. Por esse motivo, normalmente não é utilizada apenas uma grade e conjunto de âncoras e sim múltiplas grades, cada uma com seu conjunto de âncoras, com caixas de ancoragem de tamanhos diferentes baseados na quantidade de pontos de ancoragem e suas distâncias. Esse é o caso da YOLOv7 que utiliza um método chamado *Feature Pyramid Network* (FPN) (LIN et al., 2016), que como mencionado seria o pescoço da rede.

A ideia principal da FPN é se aproveitar da natureza das camadas de convolução que reduzem o tamanho do espaço de análise e aumentam a cobertura de cada característica para obter diferentes candidatos, com diferentes resoluções. Por esse motivo, FPNs geralmente são implementadas com um conjunto de camadas de convolução com algumas conexões adicionais em sentidos opostos, de maneira a garantir que cada camada tenha acesso tanto às informações de camadas anteriores quanto posteriores, como pode ser visto na Figura 2.6.

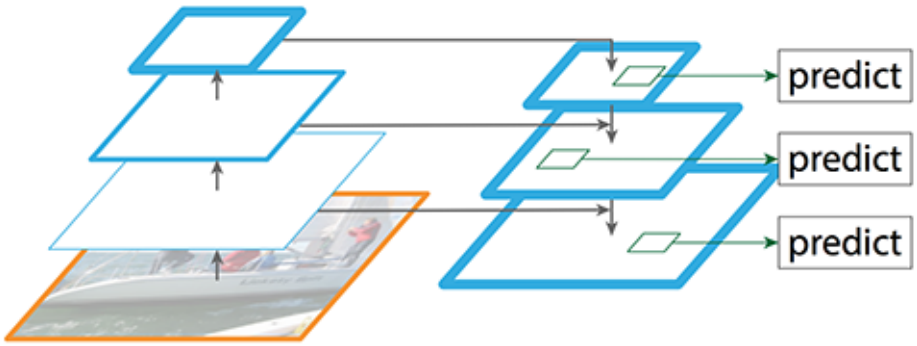

Figura 2.6 – Exemplo das camadas de uma FPN.

Fonte: Retirado de (LIN et al., 2016)

Modelos que utilizam o método de FPN podem ter um número variado de camadas e, consequentemente, de grades para detecção (LIN et al., 2016). Essas grades são comumente chamadas, de acordo com a profundidade da camada de convolução, pela nomenclatura P2, P3 e assim por diante. No caso da YOLOv7 foram disponibilizados tanto modelos que utilizavam âncoras P2 até P5 quanto modelos que incluíam também âncoras P6, usadas para detectar objetos maiores. Um exemplo das diferentes grades pode ser visto na Figura 2.7.

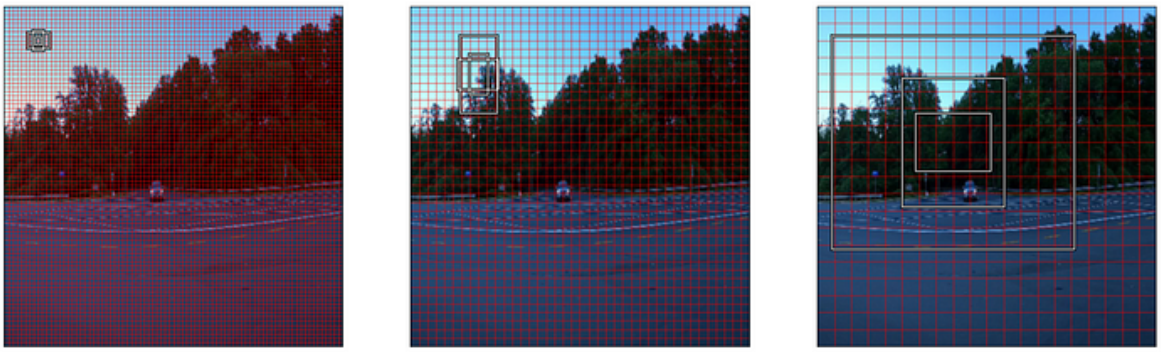

Figura 2.7 – Exemplo de grades e âncoras geradas para detecção de objetos.

Fonte: Towards Data Science (2022)

2.4 Técnicas de Aperfeiçoamento

Ao realizar o treino de qualquer modelo de Aprendizado de Máquina, em certos casos o resultado poderá não ser satisfatório (SMOLA; VISHWANATHAN, 2010). Nesses casos, vários problemas diferentes podem ter ocorrido e um bom diagnóstico é importante. Os exemplos de problemas que podem ocorrer são: um conjunto de dados desbalanceado, ou seja, com quantidades muito desiguais de exemplos de classes diferente, um conjunto de dados insuficiente, ou seja, menor do que o necessário para obtenção dos resultados desejados, ou até mesmo uma divisão inadequada entre os dados de treino e os dados de validação e teste (LI et al., 2022).

Para resolver esses problemas, pode ser feito o emprego de diferentes técnicas. Primeiramente, para auxiliar o diagnóstico do problema, é possível utilizar um método de validação cruzada, mais especificamente o *k-fold cross-validation* (GOODFELLOW; BENGIO; COURVILLE, 2016), para detectar se o problema está na divisão do conjunto de dados ou no próprio conjunto. O método consiste em dividir o conjunto de dados inteiro em "k" subconjuntos que não se sobrepõem, então, um subconjunto é separado para o teste e os outros para o treino. O processo é, então, repetido para cada subconjunto e o erro final é dado pela média dos erros (ABRIHA; SRIVASTAVA; SZABÓ, 2023).

Com a realização do procedimento de validação cruzada, conforme é apresentado pelo autor de (BERRAR, 2018), temos uma noção maior sobre os dados usados. Se, por exemplo, o erro de generalização dado pela média for baixo o suficiente, o problema pode ser na divisão dos dados e uma análise individual do erro de cada etapa pode ajudar a entender como separar melhor os conjuntos de treino e validação. Em outro caso, onde a média de erros ainda está acima do desejado, é provável que o erro esteja no próprio conjunto e seja causado por um tamanho insuficiente ou por um desbalanceamento nos exemplos mas, novamente, vale apena também analisar os resultados de cada etapa do teste (ABRIHA; SRIVASTAVA; SZABÓ, 2023).

Por fim, caso o problema esteja na divisão do banco de dados, a solução é relativamente fácil e consiste em tentar novas divisões, utilizando algum método de divisão estratificada, ou seja, que busque uma proporção adequada para cada classe entre o treino e a validação e analisar novamente os resultados, até que estes sejam satisfatórios (BERRAR, 2018).

Caso o problema seja no conjunto de dados, a situação fica um pouco mais complexa e a solução mais trabalhosa, porém o método mais efetivo seria um aumento manual da quantidade de dados.

Olhando especificamente para o caso de uma quantidade de dados insuficiente, outras técnicas que podem ser usadas são a transferência de aprendizado (ZHANG et al., 2022) e o aumento artificial de dados ou *data augmentation*. O primeiro método consiste em treinar o modelo, inicialmente em um conjunto de dados diferente e em seguida começar o treino no conjunto desejado com o modelo treinado (G.; VUTKUR; P., 2022). Isso faz com que o modelo tenha um ponto de partida melhor e uma taxa de aprendizado maior, já que com o treino anterior ele terá aprendido características em comum entre os conjuntos de dados, como por exemplo detecção de bordas. Já a outra técnica é a geração de dados a partir de manipulações feitas sobre os dados anteriores (PELLICER; FERREIRA; COSTA, 2023). Essas manipulações geram novos exemplos que podem ajudar no treino e incluem, mas não estão limitadas por rotação da imagem, espelhamento, alteração do brilho ou da saturação e adição de ruído.

Em relação ao problema de um conjunto de dados desbalanceado, pode ser feito o aumento artificial de dados em apenas algumas classes, buscando diminuir a diferença entre a quantidade de exemplos de cada classe, ou alterações no modelo, com objetivo de aumentar o peso de aprendizado de classes com menos exemplos (PELLICER; FERREIRA; COSTA, 2023).

2.5 Métricas Relevantes de Avaliação

As áreas de Aprendizado de Máquina e de Aprendizado Profundo possuem diversas métricas para avaliar o desempenho dos modelos utilizados (GOODFELLOW; BENGIO; COURVILLE, 2016). Para avaliar os modelos que serão usados neste trabalho, as principais métricas utilizadas foram a precisão, a revocação e o *Mean Average Precision* (mAP). Para um melhor entendimento sobre os resultados que serão apresentados na Seção 5, as métricas mais relevantes serão explicadas a seguir.

2.5.1 Precisão

Sendo uma das métricas mais comuns para avaliar modelos de Aprendizado Profundo, a precisão é definida como sendo a razão entre os verdadeiros positivos e o total de

classificações de positivos, incluindo os verdadeiros e falsos positivos, conforme é indicado na Equação 2.1

$$Precisão = \frac{Verdadeiros\ Positivos}{Verdadeiros\ Positivos + Falsos\ Positivos}. \tag{2.1}$$

Neste trabalho, os casos considerados como verdadeiros positivos ocorrem quando o modelo identifica um objeto como sendo um tipo de resíduo sólido e o categoriza corretamente. Já um falso positivo ocorre quando o modelo identifica um objeto como sendo um tipo de resíduo sólido, porém o categoriza incorretamente.

Ao se observar a Equação 2.1 é possível interpretar a precisão como sendo a porcentagem de classificações que realmente foram corretas dentre todas as classificações que ocorreram.

2.5.2 Revocação

Outra métrica importante é a revocação, que é definida como sendo a razão entre os verdadeiros positivos e o total de exemplos que realmente são positivos, o que inclui os verdadeiros positivos e os falsos negativos, como é mostrado na Equação 2.2

$$Revocação = \frac{Verdadeiros\ Positivos}{Verdadeiros\ Positivos + Falsos\ Negativos}. \tag{2.2}$$

A classificação de falsos negativos ocorre quando um resíduo sólido não é identificado e nem classificado.

A partir da Equação 2.2, nota-se que a revocação é uma métrica que indica a porcentagem de classificações que foram feitas corretamente dentre todas as possíveis classificações positivas. Ou seja, a revocação mede a proporção de instâncias positivas que foram identificadas corretamente como positivas pelo modelo.

2.5.3 *Mean Average Precision*

O *Mean Average Precision*, que será referido como mAP ao longo deste trabalho, é uma métrica muito utilizada para avaliar o desempenho de modelos de detecção e classificação de objetos, conforme (MAJCHROWSKA et al., 2022). O mAP indica o quão bem posicionada foi a caixa delimitadora em relação à localização real do objeto na imagem e se a classe do objeto identificado foi prevista corretamente.

Para apresentar os resultados obtidos serão utilizados dois tipos de mAP, o mAP@0.5 e o mAP@0.5:0.95, como descritas em (MAJCHROWSKA et al., 2022).

De modo que seja mais compreensível a explicação desses dois tipos de mAP antes é necessário compreender o que é o *Intersection Over Union* (IoU), que é a métrica que avalia

a precisão da caixa delimitadora prevista em relação à caixa delimitadora real do objeto detectado, podendo ser calculada como a razão entre a área da interseção entre a caixa delimitadora prevista pelo modelo e a caixa delimitadora verdadeira do objeto, e a área da união entre essas duas caixas, como é mostrado na Equação

$$IoU = \frac{Área\ de\ Intersecção}{Área\ de\ União}. \tag{2.3}$$

A Área de Intersecção se refere à área da caixa delimitadora prevista que coincide com a área da caixa delimitadora real do objeto, já a Área de União é a área total de ambas as caixas delimitadoras.

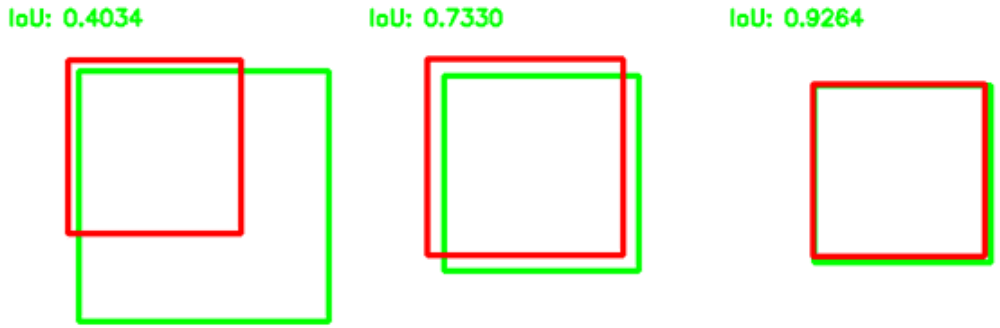

Figura 2.8 – Exemplo demonstrando diferentes níveis de IoU.

Fonte: Py Image Search (2016)

Ao se analisar a Equação 2.3 e a Figura 2.8, pode-se notar que o IoU mede a sobreposição das caixas delimitadoras previstas e reais do objeto em questão, assim fornecendo uma medida que indica o quão bem o modelo está localizando os objetos nas imagens.

Com isso será possível compreender o que são o mAP@0.5 e o mAP@0.5:0.95. O mAP@0.5 se refere ao mAP obtido ao se levar em conta um limiar para o IoU de 0.5 e já o mAP@0.5:0.95 leva em conta uma faixa de valores para o IoU, indo de 0.5 até 0.95, com incrementos de 0.05. Para o calcular seu valor final será feita uma média dos valores obtidos em cada nível de IoU dentro desse intervalo. Ou seja, o mAP@0.5 avalia o modelo em uma faixa de valores mais restritas e o mAP@0.5:0.95 abrange uma faixa de valores maior.

Nesta seção foram realizados estudos para se ter um melhor embasamento teórico ao longo deste trabalho. Já no Capítulo 3, será feita a análise de algumas pesquisas que estão relacionados com o tema desta, como por exemplo artigos que tratam de detecção de resíduos sólidos ou que utilizam conjuntos de dados de resíduos com o objetivo de obter informações sobre quais técnicas ou conjuntos de dados podem ser úteis para o desenvolvimento do projeto.

3 Trabalhos Relacionados

Neste capítulo, será apresentada uma revisão de outros trabalhos relacionados a este projeto. Essa revisão é importante para compreender o estado atual do conhecimento na área de estudo e identificar as lacunas que possam ser preenchidas por esta pesquisa. Neste sentido, busca-se apresentar uma visão geral das abordagens e métodos utilizados por outros autores, bem como seus resultados e conclusões mais relevantes. Além disso, serão destacadas as principais contribuições e limitações dos trabalhos apresentados, e como eles servirão de base para o desenvolvimento deste projeto.

3.1 Detecção de Resíduos Sólidos

De acordo com (YE et al., 2021) os modelos atuais de Aprendizado de Máquina são limitados pela sua velocidade de processamento e pelo tamanho do modelo, o que os torna inviáveis de serem usados em dispositivos portáteis e energeticamente eficientes. Para tentar resolver este problema, o autor sugere o uso de um novo modelo baseado no modelo de detecção de objetos YOLO.

Além de ser baseado no YOLO, o modelo também conta com um Autoencoder Variacional (VAE), para que ele possua maior acurácia, seja mais rápido e possua tamanho menor, para que seja possível utilizá-lo em situações reais de reciclagem de resíduos.

A conclusão e resultados do trabalho de (YE et al., 2021) mostraram que o modelo teve uma taxa de acerto de 69,70% com um número total de parâmetros de 32,1 milhões e uma velocidade de processamento de 60 *Frames* Por Segundo (FPS), o que foi melhor do que alguns modelos existentes, como o YOLOv1 e o *Fast* R-CNN. Apesar desse modelo apresentar bons resultados, ainda existem pontos a serem melhorados; conforme é dito pelo autor, o modelo não apresenta bons resultados quando há resíduos muito próximos entre si e a acurácia pode ser melhorada caso seja usada uma rede de base mais profunda e complexa.

O trabalho apresentado por (CONLEY et al., 2022) busca ajudar a solucionar o problema de monitoramento das grandes quantidades de resíduos sólidos que são produzidos em centros urbanos. O autor diz que as cidades possuem sistemas com informações insuficientes para fazer este monitoramento, o que leva a decisões baseadas em dados incorretos. O objetivo do trabalho é implementar um modelo que consiga monitorar esse grande número de resíduos de maneira mais econômica e com dados corretos para que as decisões tomadas tenham um embasamento estatístico correto.

O trabalho comparou a performance de detecção de resíduos sólidos de três modelos diferentes o *Mask* R-CNN, SOLO e YOLOv6. Os três modelos tiveram boas performances e

cada um se sobressaiu em diferentes parâmetros de resultados. Porém, no geral, o melhor modelo foi o *Mask* R-CNN, que obteve 83% de precisão e 77% de acurácia usando dados coletados de 84 ruas, em duas cidades do estado da Califórnia.

Outro estudo extremamente interessante pode ser encontrado em (CAROLIS; LADOGANA; MACCHIARULO, 2020), onde também é realizado um estudo com modelos YOLO, nesse caso YOLOv3, para detecção de resíduos sólidos em ambientes urbanos. A motivação é similar a este trabalho, sendo o objetivo principal o de facilitar a coleta de resíduos sólidos pelas autoridades. A escolha por modelos YOLO também se deu pelo mesmo motivo, bons resultados e a excelente performance em aplicações de tempo real, como filmagem de câmera. No trabalho foi utilizado um conjunto de dados feito pelos próprios autores com 2265 imagens e foram também utilizadas técnicas de aumento de dados para ampliar a quantidade de exemplos de treino.

Contudo, apesar de resultados razoáveis, com uma precisão de 68%, o trabalho não realizou nenhuma classificação quanto ao material de composição dos resíduos sólidos, optando apenas por detectar a presença deles e o recipiente no qual eles se localizavam, como: em uma lata de lixo, em um saco ou no chão.

Em comparação com o trabalho anterior, o artigo (ZHANG et al., 2022) traz outra abordagem para o problema da classificação de resíduos sólidos. Em (ZHANG et al., 2022) também é utilizado o modelo YOLO, apesar de também serem usados outros modelos para comparação e um conjunto de dados próprio criado para o trabalho. Porém, o autor optou por utilizar a técnica de transferência de aprendizado para melhorar os resultados e poder utilizar um conjunto de dados menor.

Outra diferença foi a classificação, um pouco mais complexa e que levava em conta o material de composição dos resíduos sólidos, em 5 categorias: vidro, tecido, metal, plástico e papel. A conclusão obtida foi que o modelo YOLO teve a melhor performance e a segunda melhor acurácia, porém a diferença de acurácia obtida foi tão pequena que poderia ser desconsiderada. Dessa maneira, mais uma vez é comprovada a utilidade de modelos YOLO para o problema em questão.

Um dos trabalhos mais relevantes para os estudos desse projeto foi (MAJCHROWSKA et al., 2022), no qual foram feitos testes utilizando diferentes modelos em diferentes conjuntos de imagem. Dentre os bancos de imagem usados, o mais relevante para este projeto foi a TACO ou para ser mais específico TACO *extended*, que seria o conjunto de dados oficial acrescido das fotos não oficiais. O trabalho é importante porque pode servir como uma base direta de comparação já que utiliza bancos de dados públicos.

O objetivo do trabalho (MAJCHROWSKA et al., 2022) é exatamente o de tentar criar padrões para a comparação de desempenho de modelos usados na detecção de resíduos sólidos. No final, as autoras optaram por utilizar um modelo do tipo *EfficientDet*, (TAN;

PANG; LE, 2020), e tentaram abordagens tanto de uma etapa quanto de duas etapas, no caso detecção e classificação. Um fato que vale ressaltar é que a autora agrupou as classes de todos os conjuntos de dados em apenas 7 classes: *bio; glass; metals and plastic; non-reciclable; other* e *paper*, se baseando no sistema de reciclagem de Gdansk na Polônia.

Considerando os resultados finais, no caso do classificador, na abordagem de duas etapas, o modelo teve bons resultados com uma precisão em torno de 70%. Esse resultado, porém, não demonstra a precisão real, já que o valor de 70% é apenas para as instâncias de resíduos sólidos detectados na primeira etapa, que teve uma precisão menor com um mAP@0.5, em torno de 62% para a TACO *extended*. Já avaliando o modelo em uma etapa, observamos resultados inferiores com um mAP@0.5 de 16,2%.

Para finalizar, os criadores do conjunto de dados TACO (PROENÇA; SIMÕES, 2020), também treinaram um modelo de classificação. Os autores utilizaram um modelo do tipo *Mask* R-CNN e, utilizando o conjunto de imagens próprio, obtiveram um mAP@0.5 de 15,9%. Porém, a principal motivação do trabalho era a criação e disponibilização do banco de imagens.

3.2 Conjunto de Dados

Outro fator importante a se considerar é o conjunto de dados usado para o treino do modelo, nesse campo vários esforços foram feitos.

Provavelmente o mais interessante dos conjuntos de dados, e o que foi utilizado nesse trabalho, é a TACO (PROENÇA; SIMÕES, 2020), um conjunto de dados aberto que conta atualmente com 1500 imagens e quase 5000 anotações. Por ser um conjunto de imagens aberto, o objetivo é que mais dados sejam adicionados pela comunidade, com as ferramentas fornecidas pelos responsáveis pelo conjunto de dados. Além da oportunidade de crescimento com a ajuda da comunidade, esses dados também contam com uma taxonomia extensa, com mais de 50 classes, categorizando o material que compõe os resíduos sólidos. A taxonomia foi pensada de maneira a facilitar os esforços de reciclagem.

O conjunto de imagens da TACO, como mencionado anteriormente, é um banco de dados aberto, o que significa que qualquer pessoa pode contribuir para o aumento de dados. Com isso várias imagens não oficiais são disponibilizadas pelos autores (PROENÇA; SIMÕES, 2020). Utilizando as imagens oficiais e rotulando, novamente, as imagens não oficiais, de forma a garantir uma melhor qualidade dos rótulos nessas imagens, as autoras (MAJCHROWSKA et al., 2022) disponibilizaram o conjunto TACO *extended*. Apesar de conter uma quantidade muito maior de imagens, a nova rotulação feita no conjunto reduziu o número de classes de 60 para 7. A divisão de rótulos desse conjunto pode ser vista na Tabela 3.1.

Tabela 3.1 – Rótulos por classe.

Conjunto de Imagens	Orgânico	Vidro	Metal e Plástico	Papel	Não Reciclável	Outros	Desconhecido
TACO *Extended*	69	592	6057	601	2802	154	3258

O conjunto de dados PlastOPol (CÓRDOVA et al., 2022) é outro conjunto de imagens interessante, já que é mais extenso do que a TACO (PROENÇA; SIMÕES, 2020), contando com quase 2500 imagens. Apesar da quantidade maior de imagens, esse conjunto possui apenas uma categoria, usada para identificar a presença ou não de resíduos sólidos, o que de certa forma diminui seu valor para esse projeto. Apesar disso posteriormente esse banco de dados pode ser reclassificado, com ajuda do modelo YOLO treinado, o que resultaria em um novo conjunto de dados que ajudaria ainda mais com treinos futuros do modelo.

Além desses, como mencionado anteriormente, alguns dos autores responsáveis por treinar modelos para a detecção e classificação de resíduos sólidos optaram pela criação do próprio conjunto de dados, porém esses dados não se encontram disponíveis. (CAROLIS; LADOGANA; MACCHIARULO, 2020) e (ZHANG et al., 2022) são exemplos. Isso significa que é viável o incremento de bases de dados utilizando imagens próprias.

Após ter sido feita a análise de alguns trabalhos que são relacionados a este, será apresentada a metodologia proposta para o desenvolvimento desta pesquisa. No Capítulo 4 será descrito como foram feitas cada uma das etapas deste trabalho.

4 Metodologia Proposta

Este trabalho foi realizado com base na metodologia proposta no fluxograma da Figura 4.9. As etapas do desenvolvimento desse trabalho são sequenciais, indo do levantamento bibliográfico até a análise dos resultados finais. Ao longo dessas etapas serão feitas diversas avaliações e análises de cada um dos resultados, estes obtidos antes de avançar para a próxima etapa, para assim garantir o sucesso no fim do projeto. Ao longo deste Capítulo o desenvolvimento do trabalho será explicado em mais detalhes.

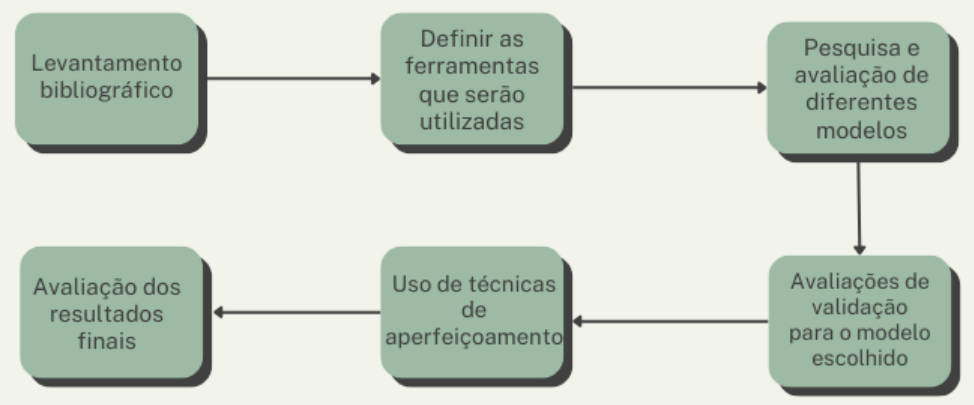

Figura 4.9 – Fluxograma da metodologia.

4.1 Levantamento Bibliográfico

A principal fonte referência bibliográfica utilizada para a realização deste trabalho foi (GOODFELLOW; BENGIO; COURVILLE, 2016), que possui um vasto conteúdo sobre Aprendizado Profundo. A partir da leitura de diversos capítulos foi possível aprender muito sobre a área de Aprendizado Profundo e, com isso, obter um bom embasamento teórico para dar prosseguimento ao trabalho.

Para o desenvolvimento do projeto foram levados em conta diversos modelos de detecção de objetos. Após a leitura de artigos e pesquisas sobre os modelos, a conclusão foi que o modelo YOLO (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022) seria o mais adequado para o projeto pela sua rapidez e eficiência.

Este trabalho obteve inspiração a partir da pesquisa de (CAROLIS; LADOGANA; MACCHIARULO, 2020), que utilizou o modelo YOLO e mostrou que este modelo é uma ótima escolha para problemas de detecção de objetos. O autor de (YE et al., 2021) também utilizou o modelo YOLO na detecção e classificação de resíduos sólidos em imagens e obteve resultados que reforçam a boa precisão e velocidade do modelo. O trabalho de (CONLEY et

al., 2022), comparou vários modelos de Aprendizado Profundo, incluindo o YOLO e também obteve resultados comprovando o bom desempenho do modelo. Outro trabalho utilizado como motivação foi o de (ZHANG et al., 2022), que faz o uso do modelo YOLO e implementa técnicas de transferência de aprendizado para obter melhores resultados.

Uma das pesquisas mais relevantes para este trabalho foi o artigo de (MAJCHROWSKA et al., 2022), em que foram feitas a detecção e classificação de resíduos sólidos em imagens. Apesar de não terem utilizado o modelo YOLO, o artigo continua sendo útil, uma vez que trata da comparação entre os resultados de diferentes modelos de detecção de objetos, além de terem sido utilizadas fotos de conjunto de dados públicos, como o da TACO, o que facilita a comparação entre os resultados obtidos no artigo e nesta pesquisa.

Após a leitura, análise e avaliação dos trabalhos citados anteriormente neste capítulo, foram obtidas informações valiosas sobre quais seriam os melhores modelos para se utilizar na detecção de objetos e quais técnicas podem ser associadas para se obter melhores resultados na detecção de resíduos sólidos em imagens. Além disso foi possível analisar os resultados obtidos em outros trabalhos e por outros modelos de detecção de objetos, assim possibilitando que se tenha uma noção de quais resultados são satisfatórios ou não.

4.2 Ferramentas Utilizadas

As duas principais escolhas que devem ser feitas ao resolver um problema de Aprendizado de Máquina são o modelo e o conjunto de dados a serem usados (GOODFELLOW; BENGIO; COURVILLE, 2016). Inicialmente, foi levada em consideração a escolha do banco de imagens. Dentre as opções analisadas anteriormente o conjunto de imagens TACO foi escolhido, principalmente devido a sua complexa taxonomia, que ajudaria nos esforços de reciclagem e a possibilidade de crescimento com ajuda da comunidade. A escolha também foi feita contando com a possibilidade de expansão do banco de dados futuramente, caso os resultados do estudo fossem bons o suficiente para serem utilizados na aquisição de novos dados.

Além do uso do banco de dados TACO oficial, foi feito também o uso da TACO *extended* para alguns testes. A TACO *extended* se refere às imagens oficiais junto com as imagens não oficiais, com anotações feitas pelas autoras (MAJCHROWSKA et al., 2022), utilizando, entretanto, uma nova taxonomia de classes. Na Figura 4.10 pode-se observar exemplos de imagens do banco de dados oficial da TACO.

Com a seleção do banco de dados e após os estudos dos trabalhos relacionados, foi decidido o uso de um modelo YOLOv7 (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022) para resolver o problema proposto. Essa escolha foi feita, tendo em mente o desempenho e a rapidez do modelo quando comparado a outros modelos utilizados nos estudos (ZHANG et al., 2022).

(a) Exemplo 1

(b) Exemplo 2

(c) Exemplo 3

(d) Exemplo 4

Figura 4.10 – Exemplos de imagens do banco de dados TACO oficial.

Fonte: TACO (2023)

Com a escolha do modelo, veio a necessidade de reformatar os rótulos usados para classificar os dados, já que os dados da TACO vêm no formato COCO JSON, enquanto a YOLOv7 utiliza rótulos no formato YOLO. Para isso foi utilizada a biblioteca em python, *pylabel*, que, além de possibilitar a conversão entre os formatos, também oferece a funcionalidade de divisão do banco de dados em conjuntos de treino e validação, de maneira proporcional.

Além das ferramentas e componentes essenciais mencionados anteriormente, também foi utilizada a plataforma do *Roboflow* para realizar as anotações em fotos novas e o uso da biblioteca *Upscayl* para realizar o aumento da resolução, usando o método *Real*-ESRGAN (WANG, X. et al., 2021), de algumas imagens do subconjunto de treino.

4.3 Avaliação de Modelos

Após a preparação do conjunto de dados e do modelo inicial, foi necessário fazer a divisão do conjunto de imagens em subconjuntos. Em casos normais, a divisão é feita em 3 subconjuntos, sendo eles: treino, validação e teste, porém, para os primeiros testes, com o objetivo de decidir os pesos e avaliar os modelos, a divisão foi feita apenas em treino e validação, com as respectivas proporções de 70% e 30%, já que os dados não são muito extensos. Como muitas das fotos presentes no conjunto de dados possuem múltiplos exemplos, até mesmo de classes diferentes, foi utilizado um *script* próprio para reordenar o conjunto de imagens e tentar obter uma divisão o mais próxima possível das proporções desejadas.

Visto isso, com o conjunto de dados determinado e dividido para os testes, foram elaborados *scripts* em python para automatizar os testes.

Com o modelo inicial selecionado, o conjunto de dados formatado e devidamente dividido e um *script* para os testes pronto, tudo que faltava era a própria realização dos testes. Nesse sentido, o autor do YOLOv7 (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022) fornece vários pesos iniciais, então cada um dos pesos foi utilizado no treino do modelo, com objetivo de descobrir o melhor deles para esse trabalho. Os testes iniciais foram feitos sobre a TACO oficial.

Com esses resultados será possível identificar o melhor modelo para ser usado como ponto de partida para os próximos testes, evitando o uso de empirismos no processo.

4.4 Avaliações Principais de Validação

Após a seleção do modelo com as avaliações preliminares, foram feitas análises, inicialmente, sobre o conjunto de dados da TACO *extended*, em que se utiliza de apenas 7(sete) classes de objetos do tipo de resíduos sólidos. A primeira avaliação visa avaliar uma situação mais simples, já que o banco de dados estava divido em menos classes e era um conjunto mais extenso. Além disso, utilizar esse conjunto de imagens, permitiu realizar uma boa base de comparação com o trabalho de (MAJCHROWSKA et al., 2022).

Após a finalização das primeiras avaliações, foram utilizados alguns métodos que serão discutidos posteriormente, na Seção 4.5, com o objetivo de aperfeiçoar o modelo e com isso realizar mais avaliações para verificar os novos resultados.

Após a conclusão das avaliações sobre o conjunto TACO *extended*, foram realizadas avaliações limitadas ao conjunto da TACO oficial, dessa vez, utilizando todas as 60 classes, com o objetivo de avaliar o modelo em um caso de estudo mais complexo. Assim como na avaliação anterior, foram utilizadas técnicas de aperfeiçoamento em busca de obter melhores resultados.

4.5 Aperfeiçoamento do Modelo

Após a seleção do modelo de partida, foram estudados e então aplicados dois métodos de aperfeiçoamento, a transferência de aprendizado e o aumento de dados, na tentativa de melhorar os resultados.

O primeiro método utilizado foi o uso de transferência de aprendizado. A transferência de aprendizado foi realizada utilizando um modelo treinado com banco de imagens *Microsoft Common Objects in Context* (MS COCO), disponibilizado pelos autores do trabalho (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022). Essa técnica foi utilizada já nos testes

preliminares de escolha de modelo, encima do melhor modelo. A utilização prematura se deve ao fato de que o modelo utilizando transferência de aprendizado seria um modelo próprio.

O segundo método utilizado, foi o aumento do conjunto de imagens utilizadas, tanto de maneira manual, com a busca de novas fotos usando ferramentas de busca e utilizando anotações feitas manualmente, quanto de maneira automática utilizando bibliotecas de aumento de dados (PELLICER; FERREIRA; COSTA, 2023). Esse aumento do banco de imagens foi feito apenas para classes específicas do conjunto, levando em consideração o grande desbalanceamento do conjunto de dados. As classes que tiveram imagens adicionadas manualmente estão na Figura 4.11. Já com o uso da biblioteca foram criadas 12(doze) novas imagens de treino para cada uma das classes *Paper straw; Squeezable tube; Shoe; Scrap metal; Rope & strings; Plastic utensils; Plastic glooves; Other plastic container; Foam food container; Disposable food container; Spread tub; Crisp packet; Polypropylene bag; Garbage bag; Six pack rings; Plastified paper bag; Paper bag; Wrapping paper; Tissues; Magazine paper; Metal lid; Glass jar; Other plastic cup; Glass cup; Foam cup; Pizza box; Meal carton; Corrugated carton; Drink carton; Egg carton; Toilet tube; Aerosol; Food Can; Other plastic bottle; Carded blister pack; Aluminium blister pack; Battery e Aluminium foil*, mudando de forma aleatória o valor de brilho entre 80% e 120%, de saturação entre 80% e 120% e do contraste entre 80% e 120%.

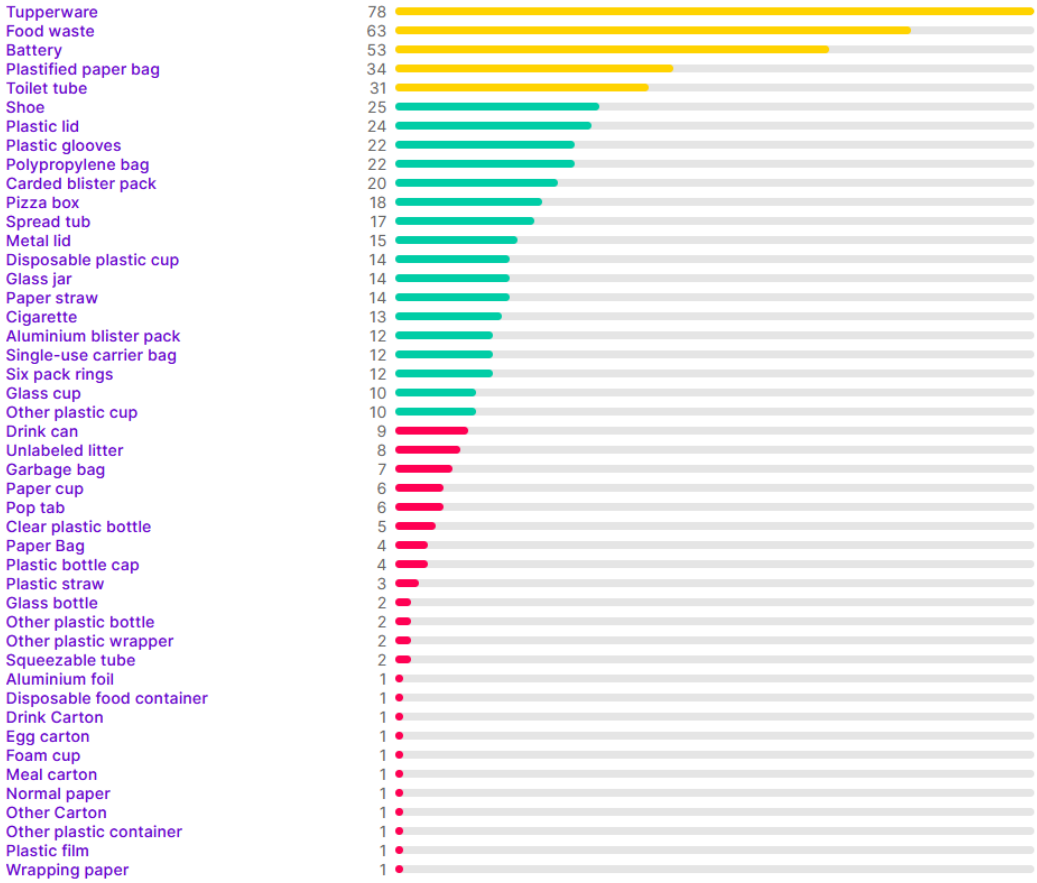

Figura 4.11 – Exemplos adicionados manualmente.

Por fim, apenas para os testes realizados no conjunto de imagens TACO *extended*, foi realizado o aumento da resolução com ajuda de inteligência artificial para algumas imagens com resoluções menores do que as recomendadas. Isso se deve ao fato de que algumas fotos não oficiais eram de baixa resolução. O método utilizado para o aumento da resolução foi o *Real*-ESRGAN (WANG, X. et al., 2021).

Os resultados obtidos ao longo desta etapa serão apresentados no Capítulo 5. No capítulo seguinte serão mostrados gráficos e tabelas comparando os resultados de diferentes modelos e os resultados ao se utilizar diferentes quantidades de classes.

5 Resultados

Neste capítulo serão apresentados os resultados obtidos na detecção e classificação de resíduos sólidos em imagens. Os resultados foram obtidos utilizando o modelo YOLOv7, (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022) e as imagens do conjunto de dados TACO e TACO *extended*, (PROENÇA; SIMÕES, 2020). Será apresentada a comparação dos resultados obtidos dos diversos modelos disponíveis da YOLOv7, assim evidenciando qual modelo apresenta o melhor resultado para este trabalho. Além disso, ao se classificar os diferentes tipos de resíduos sólidos foram considerados dois cenários distintos, onde, em um deles os resíduos sólidos foram divididos em 60 classes e, no outro caso, foram divididos em 7 classes. Ambos os resultados serão mostrados a seguir.

5.1 Parâmetros de Treinamento

Inicialmente o conjunto de imagens da TACO foi dividido conforme o proposto no Capítulo 4, seguindo a proporção de 70% das imagens para treinamento e 30% para validação. Essa divisão pode ser vista na Figura 5.12.

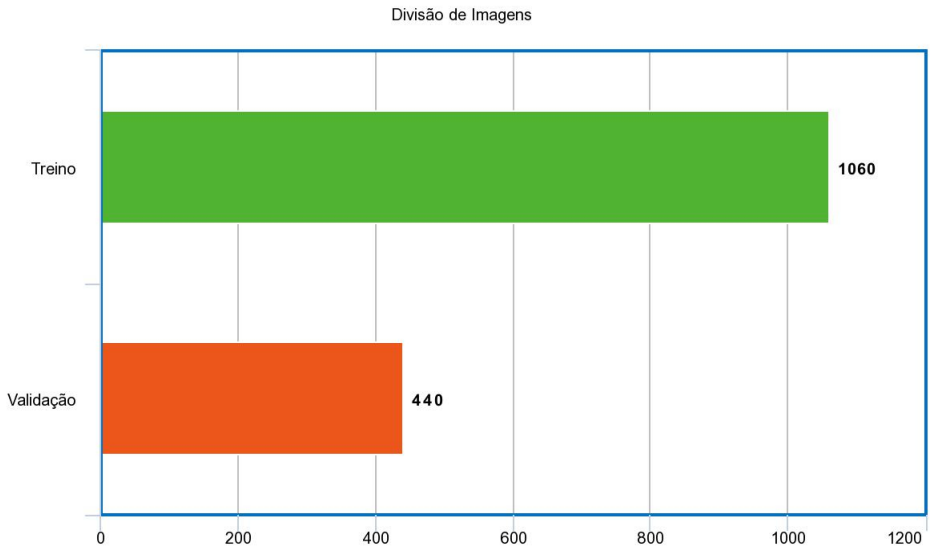

Figura 5.12 – Divisão inicial do conjunto de imagens TACO.

Já o conjunto de imagens TACO *extended* foi divido de maneira idêntica à (MAJCHROWSKA et al., 2022), de maneira que fosse possível realizar a comparação entre os resultados obtidos. Essa divisão foi de 80% das imagens para o treinamento e 20% para a validação, e pode ser vista na Figura 5.13.

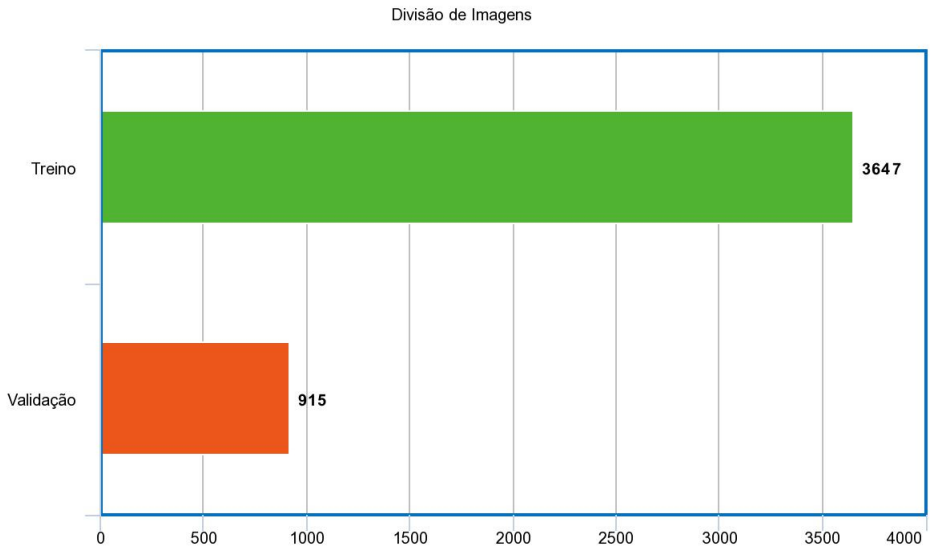

Figura 5.13 – Divisão inicial do conjunto de imagens TACO *extended*.

Após aplicar a técnica de aumento de dados, de maneira manual, conforme descrito no Capítulo 4, os conjuntos de imagens da TACO e TACO *extended* foram divididos novamente e podem ser vistos na Figura 5.14 e na Figura 5.15, respectivamente.

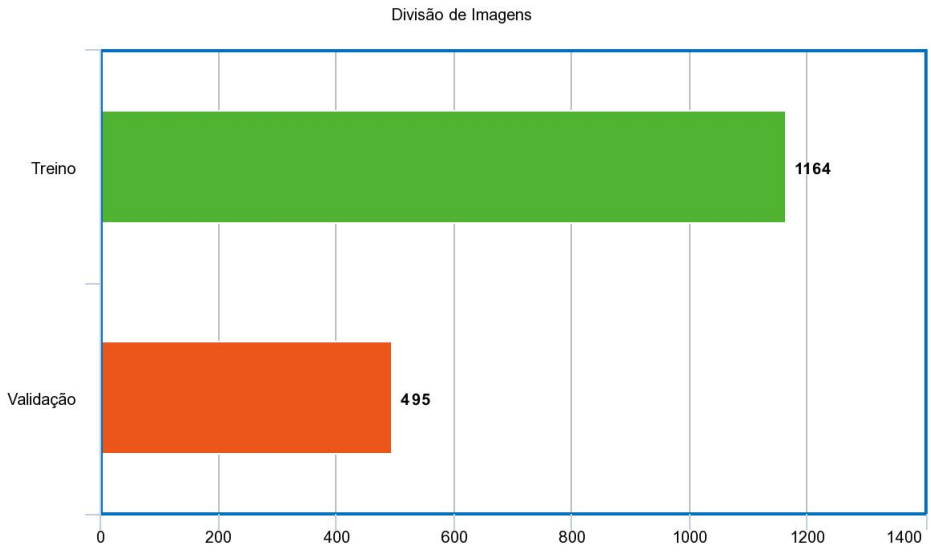

Figura 5.14 – Divisão final do conjunto de imagens TACO.

Então, para os testes realizados nesse trabalho os parâmetros utilizados foram 350 épocas, um tamanho de lote de 6 e um tamanho fixo de imagem de 640×640 para os modelos YOLOv7 e YOLOv7-X, que utilizam âncoras no formato P5, e 1280×1280 para o restante dos modelos, que utilizam também âncoras no formato P6, sendo a adição de bordas pretas o método utilizado para o redimensionamento das imagens.

O valor do parâmetro de tamanho de lote não é tão relevante, nesse caso, já que o

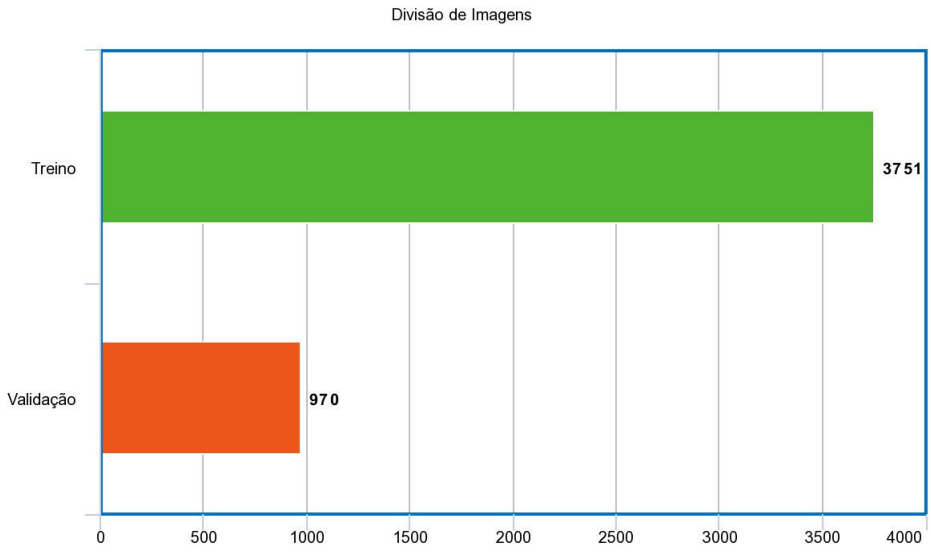

Figura 5.15 – Divisão final do conjunto de imagens TACO *extended*.

modelo YOLOv7 se utiliza do método de acumulação de gradiente, onde o gradiente de múltiplos passos é acumulado para formar um lote de tamanho mínimo antes de atualizar o modelo (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022), o que permite efetivamente o uso de lotes maiores mesmo sem memória suficiente na GPU. Já o número de épocas foi escolhido baseado nos primeiros treinamentos de teste, considerando o ponto onde os valores de mAP@0.5:0.95 começam a se estabilizar e atingem o máximo. Um exemplo pode ser visto na Figura 5.16, onde o mAP@0.5:0.95 se estabilizou por volta da época 200 e em um caso teve seu valor mais alto por volta da época 300.

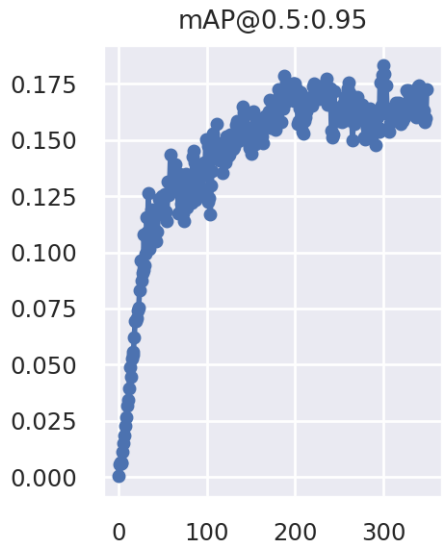

Figura 5.16 – mAP@0.5:0.95 ao longo de 350 épocas no conjunto TACO utilizando o modelo YOLOv7-E6E.

5.2 Comparação de Modelos

A YOLOv7 (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022) possui diversos modelos, então, para saber qual seria o mais adequado para este trabalho foram feitos testes em todos os modelos disponíveis. Os resultados estão representados na Tabela 5.2.

O modelo YOLOv7 padrão serve como base para os outros modelos. O modelo YOLOv7-W6 é apenas o modelo YOLOv7 só que alterado para ser utilizado com GPUs mais potentes, usufruindo de um maior poder computacional, inclusive utilizando imagens de resolução maior. O modelo YOLOv7-X é obtido ao se aplicar, no modelo YOLOv7 padrão, uma abordagem tipo empilhamento piramidal de imagens em múltiplas escalas na etapa anterior ao topo do modelo (i.e. *stack scaling on neck*) e com isso aumenta a profundidade e largura do modelo inteiro. Já a partir do modelo YOLOv7-W6, ao aplicar o mesmo método de escalonamento composto proposto se obtém os modelos YOLOv7-D6 e YOLOv7-E6. Para se obter o modelo YOLOv7-E6E é aplicado o método E-ELAN no YOLOv7-E6. Por fim, o modelo TL-YOLOv7-E6E é similar ao YOLOv7-E6E, porém neste caso se utiliza o método de aperfeiçoamento de transferência de aprendizado, (WANG, C.-Y.; BOCHKOVSKIY; LIAO, 2022).

Tabela 5.2 – Comparação de métricas de avaliação dos modelos utilizados.

Modelo	Melhor Época	Precisão	Revocação	mAP@0.5	mAP@0.5:0.95
YOLOv7	252	0.3426	0.2199	0.156	0.1321
YOLOv7-X	333	0.2642	0.1903	0.1047	0.08182
YOLOv7-W6	209	0.3901	0.2038	0.1939	0.1628
YOLOv7-E6	233	0.3402	0.2318	0.1975	0.1677
YOLOv7-D6	205	0.296	0.2419	0.1912	0.1603
YOLOv7-E6E	190	0.2521	0.2523	0.2047	0.1755
TL-YOLOv7-E6E	264	0.4366	0.2721	0.2337	0.2026

Ao analisar os resultados obtidos na Tabela 5.2 é possível notar que o modelo TL-YOLOv7-E6E obteve o melhor resultado tanto de mAP@0.5:0.95, com 0.2026, quanto de precisão e revocação. Este modelo é similar ao YOLOv7-E6E, porém utilizando um método de aperfeiçoamento com transferência de aprendizado, conforme explicado na Seção 4.5. Este modelo teve um aumento de 15,44% no mAP@0.5:0.95, em comparação com o mesmo modelo sem o aperfeiçoamento e foi o único modelo utilizando a técnica a ser testado, devido a performance superior a dos outros modelos.

Os valores dos hiper parâmetros utilizados nos modelos ao longo deste trabalho estão presentes nas Tabelas 5.3 e 5.4.

Os resultados inicias da comparação de modelo foram promissores. Levando em conta o trabalho original (PROENÇA; SIMÕES, 2020), o modelo inicial, utilizando transferência de aprendizado, obteve resultados superiores, com um mAP@0.5 de 23.37%, em comparação com a medida de 15.9% obtida pelo autor, utilizando um modelo *Mask* R-CNN.

Tabela 5.3 – Hiper parâmetros para modelos com âncora P5

Hiper parâmetros	Valores
lr0: initial learning rate (SGD=1E-2, Adam=1E-3)	0.01
lrf: final OneCycleLR learning rate (lr0 * lrf)	0.1
momentum: SGD momentum/Adam beta1	0.937
weight_decay: optimizer weight decay	0.0005
warmup_epochs: warmup epochs (fractions ok)	3.0
warmup_momentum: warmup initial momentum	0.8
warmup_bias_lr: warmup initial bias lr	0.1
box: box loss gain	0.05
cls: cls loss gain	0.3
cls_pw: cls BCELoss positive_weight	1.0
obj: obj loss gain (scale with pixels)	0.7
obj_pw: obj BCELoss positive_weight	1.0
iou_t: IoU training threshold	0.20
anchor_t: anchor-multiple threshold	4.0
fl_gamma: focal loss gamma (efficientDet default gamma=1.5)	0.0
degrees: image rotation (+/- deg)	0.0
translate: image translation (+/- fraction)	0.2
scale: image scale (+/- gain)	0.9
shear: image shear (+/- deg)	0.0
perspective: image perspective (+/- fraction), range 0-0.001	0.0
flipud: image flip up-down (probability)	0.0
fliplr: image flip left-right (probability)	0.5
mosaic: image mosaic (probability)	1.0
mixup: image mixup (probability)	0.15
copy_paste: image copy paste (probability)	0.0
paste_in: image copy paste (probability), use 0 for faster training	0.15
loss_ota: use ComputeLossOTA, use 0 for faster training	1

5.3 Classificação em 7 classes

Nesta seção serão apresentados os resultados dos testes realizados com o modelo identificando e classificando os tipos de resíduos sólidos em 7 classes. No caso, é feito utilizando as imagens do conjunto de dados da TACO *extended*, conforme é descrito em (MAJCHROWSKA et al., 2022).

Inicialmente, foram feitos os testes sobre o conjunto TACO *extended* sem nenhuma alteração no conjunto, utilizando apenas o modelo TL-YOLOv7-E6E, que foi escolhido por ter apresentado os melhores resultados na etapa anterior. Os resultados podem ser vistos na figura 5.17 e na Tabela 5.5. Esse teste foi feito com o objetivo de comparar o modelo utilizado no trabalho com o *EfficientDet*-D2 usado em (MAJCHROWSKA et al., 2022).

Comparando os resultados obtidos em (MAJCHROWSKA et al., 2022), que podem ser vistos na tabela 5.6, é possível concluir que, quando treinado e validado apenas no banco

Tabela 5.4 – Hiper parâmetros para modelos com âncora P6

Hiper parâmetros	Valores
lr0: initial learning rate (SGD=1E-2, Adam=1E-3)	0.01
lrf: final OneCycleLR learning rate (lr0 * lrf)	0.2
momentum: SGD momentum/Adam beta1	0.937
weight_decay: optimizer weight decay	0.0005
warmup_epochs: warmup epochs (fractions ok)	3.0
warmup_momentum: warmup initial momentum	0.8
warmup_bias_lr: warmup initial bias lr	0.1
box: box loss gain	0.05
cls: cls loss gain	0.3
cls_pw: cls BCELoss positive_weight	1.0
obj: obj loss gain (scale with pixels)	0.7
obj_pw: obj BCELoss positive_weight	1.0
iou_t: IoU training threshold	0.20
anchor_t: anchor-multiple threshold	4.0
fl_gamma: focal loss gamma (efficientDet default gamma=1.5)	0.0
degrees: image rotation (+/- deg)	0.0
translate: image translation (+/- fraction)	0.2
scale: image scale (+/- gain)	0.9
shear: image shear (+/- deg)	0.0
perspective: image perspective (+/- fraction), range 0-0.001	0.0
flipud: image flip up-down (probability)	0.0
fliplr: image flip left-right (probability)	0.5
mosaic: image mosaic (probability)	1.0
mixup: image mixup (probability)	0.15
copy_paste: image copy paste (probability)	0.0
paste_in: image copy paste (probability), use 0 for faster training	0.15
loss_ota: use ComputeLossOTA, use 0 for faster training	1

Tabela 5.5 – Resultado do primeiro treinamento no conjunto TACO *extended*.

Modelo	Precisão	Revocação	mAP@0.5	mAP@0.5:0.95
TL-YOLOv7-E6E	0.3244	0.2576	0.2202	0.1643

Tabela 5.6 – Resultado do classificador em uma etapa no conjunto TACO *extended* retirado de Majchrowska et al. (2022).

Modelo	mAP@0.5	mAP@0.5:0.95
EfficientDet-D2	0.162	0.119

TACO *extended*, o modelo utilizado nesse trabalho é superior.

Em seguida, analisando os resultados na Figura 5.17, podemos observar que os resultados variam significativamente entre classes. Um dos principais fatores para que os valores das classes Metais e Plásticos e Vidros sejam muito melhores do que o de classes como de Biológico e Outros é a variabilidade nos elementos da própria classe, já que vidros ou metais possuem muito mais características em comum do que materiais biológicos. Outro

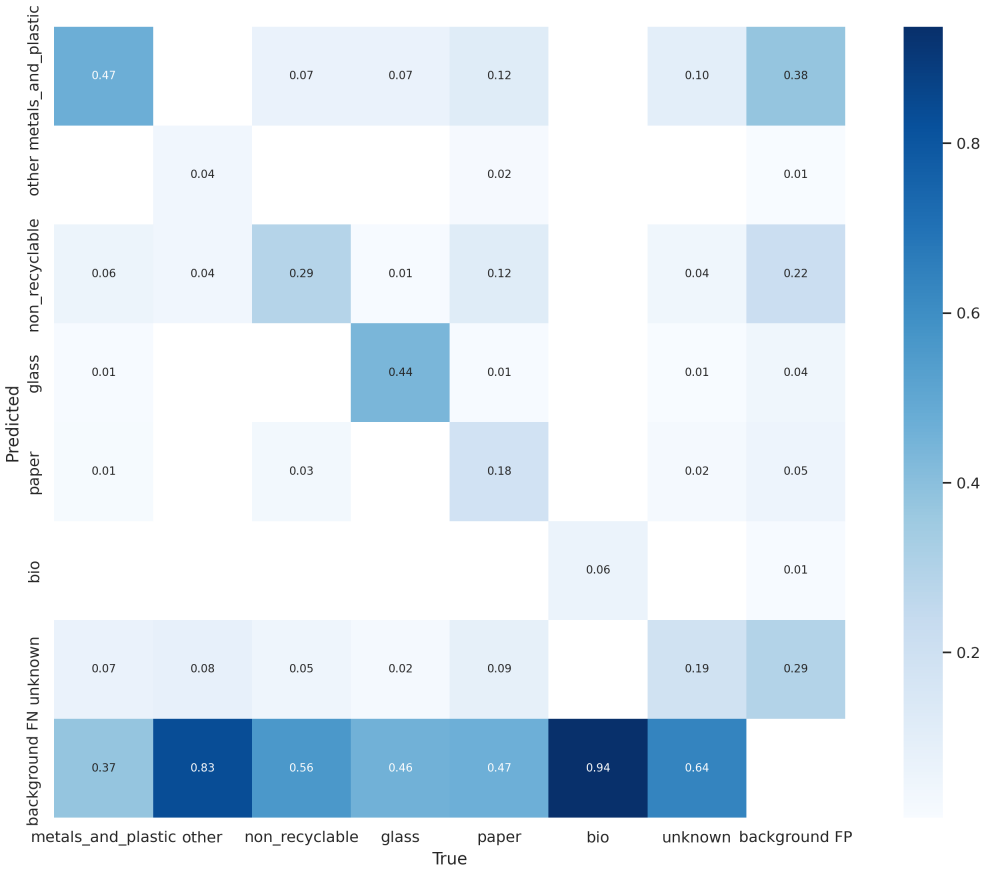

Figura 5.17 – Matriz de confusão do primeiro treinamento no conjunto TACO *extended*.

ponto que vale ser ressaltado é também a quantidade desbalanceada de exemplos de cada classe, como pode ser visto na Tabela 3.1. Além disso, outro fator que pode ter impactado é a baixa resolução de diversas imagens do conjunto de dados.

Outra observação interessante é que não há muitos erros de classificação entre as classes, com a maior parte dos erros do modelo vindo da não detecção de exemplos, de falsos negativos, que estão representados na última linha da matriz de confusão.

Com o objetivo de mitigar os problemas anteriores, os métodos de aperfeiçoamento de adição de dados manual e de aumento artificial de resolução foram utilizados, dando continuidade aos treinamentos do modelo.

Os resultados obtidos no treinamento, utilizando os métodos, podem ser vistos na Figura 5.18 e na Tabela 5.7.

Tabela 5.7 – Resultado do treinamento utilizando técnicas de aperfeiçoamento no conjunto TACO *extended*.

Modelo	Precisão	Revocação	mAP@0.5	mAP@0.5:0.95
TL-YOLOv7-E6E	0.3630	0.3396	0.2748	0.2053

Analisando os dados, é possível concluir que o uso das técnicas apresentou um

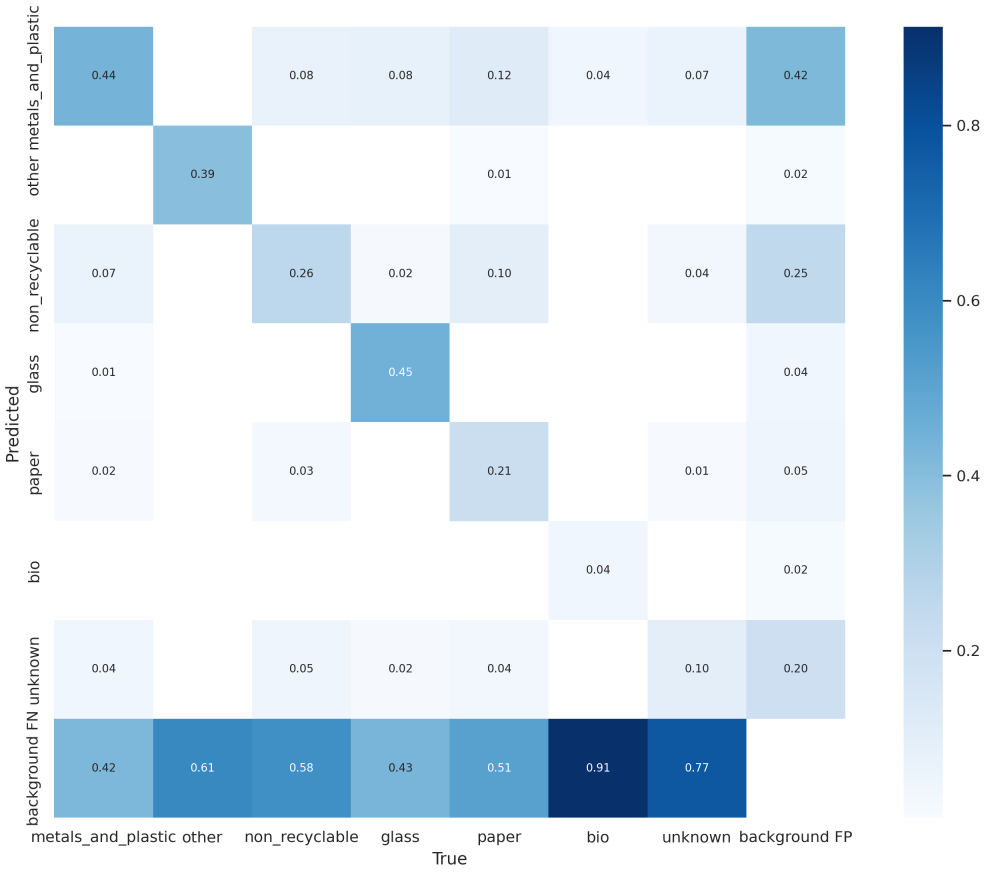

Figura 5.18 – Matriz de confusão do treinamento utilizando técnicas de aperfeiçoamento no conjunto TACO *extended*.

impacto positivo com um aumento de quase 25% no valor do mAP@0.5, que foi de 0.2202 para 0.2748. Podemos observar que grande parte dessa melhora veio de um aumento considerável na detecção de objetos do tipo Outro.

5.4 Classificação em 60 classes

Nesta seção serão apresentados os resultados dos testes realizados com o modelo identificando e classificando os tipos de resíduos sólidos em 60 classes, utilizando as imagens do conjunto de dados oficial da TACO (PROENÇA; SIMÕES, 2020).

Apesar do modelo que utiliza transferência de aprendizado já mostrar resultados superiores aos de outros trabalhos, outras técnicas de aperfeiçoamento, como aumento de dados, foram utilizadas em busca de resultados ainda melhores.

Inicialmente, o aumento de dados foi realizado de maneira manual com a coleta de exemplos na internet e obtidos manualmente e em uma etapa posterior foi utilizada a criação artificial de exemplos.

O resultado do teste do modelo TL-YOLOv7-E6E apenas com a adição das imagens

obtidas manualmente pode ser visto na Tabela 5.8 e na Figura 5.19.

Tabela 5.8 – Resultado do treinamento utilizando aumento de dados manual no conjunto TACO.

Modelo	Precisão	Revocação	mAP@0.5	mAP@0.5:0.95
TL-YOLOv7-E6E	0.4426	0.3691	0.3495	0.3073

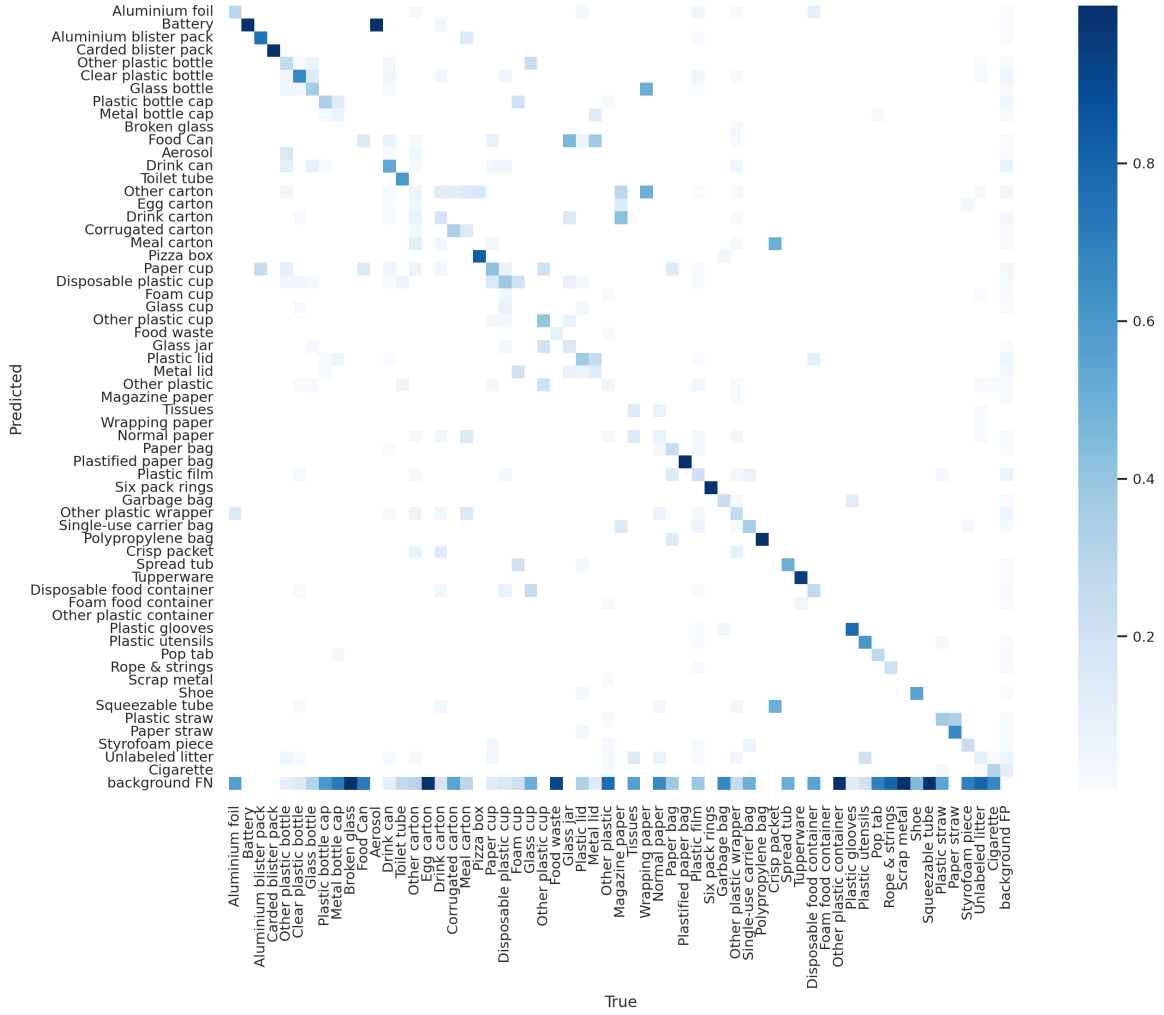

Figura 5.19 – Matriz de confusão do treinamento utilizando aumento de dados manual no conjunto TACO.

O resultado observado representa uma melhora de 51.67% no valor do mAP@0.5:0.95 quando comparado com o valor sem o uso da técnica, que pode ser visualizado na Tabela 5.2. Esse aumento pode ser explicado por uma melhora no balanceamento do banco de imagens, já que as imagens coletadas buscavam aumentar a quantidade de exemplos de classes sub representadas.

Em seguida foi realizado o treinamento utilizando uma técnica de aumento de dados artificial, utilizando uma biblioteca de geração de imagens, a partir de transformações realizadas em outros exemplos do próprio banco de imagens.

Como pode ser visto na Figura 5.20 e na Tabela 5.9, os resultados acabaram sendo inferiores aos resultados obtidos no teste anterior. É possível que a piora no resultado tenha sido causada por exemplos artificiais com características destoantes do resto dos exemplos da classe. Apesar de resultados gerais inferiores, analisando as duas matrizes de confusão é possível notar que houve uma diminuição de classificações erradas entre classes e que a piora veio na detecção dos objetos.

Os resultados obtidos nos treinamentos dos modelos apresentados ao longo deste trabalho, assim como as instruções para se obter os conjuntos de imagens utilizados, estão disponíveis neste repositório do GitHub.

Tabela 5.9 – Resultado do treinamento utilizando aumento de dados manual e artificial no conjunto TACO.

Modelo	Precisão	Revocação	mAP@0.5	mAP@0.5:0.95
TL-YOLOv7-E6E	0.4252	0.3471	0.3413	0.2951

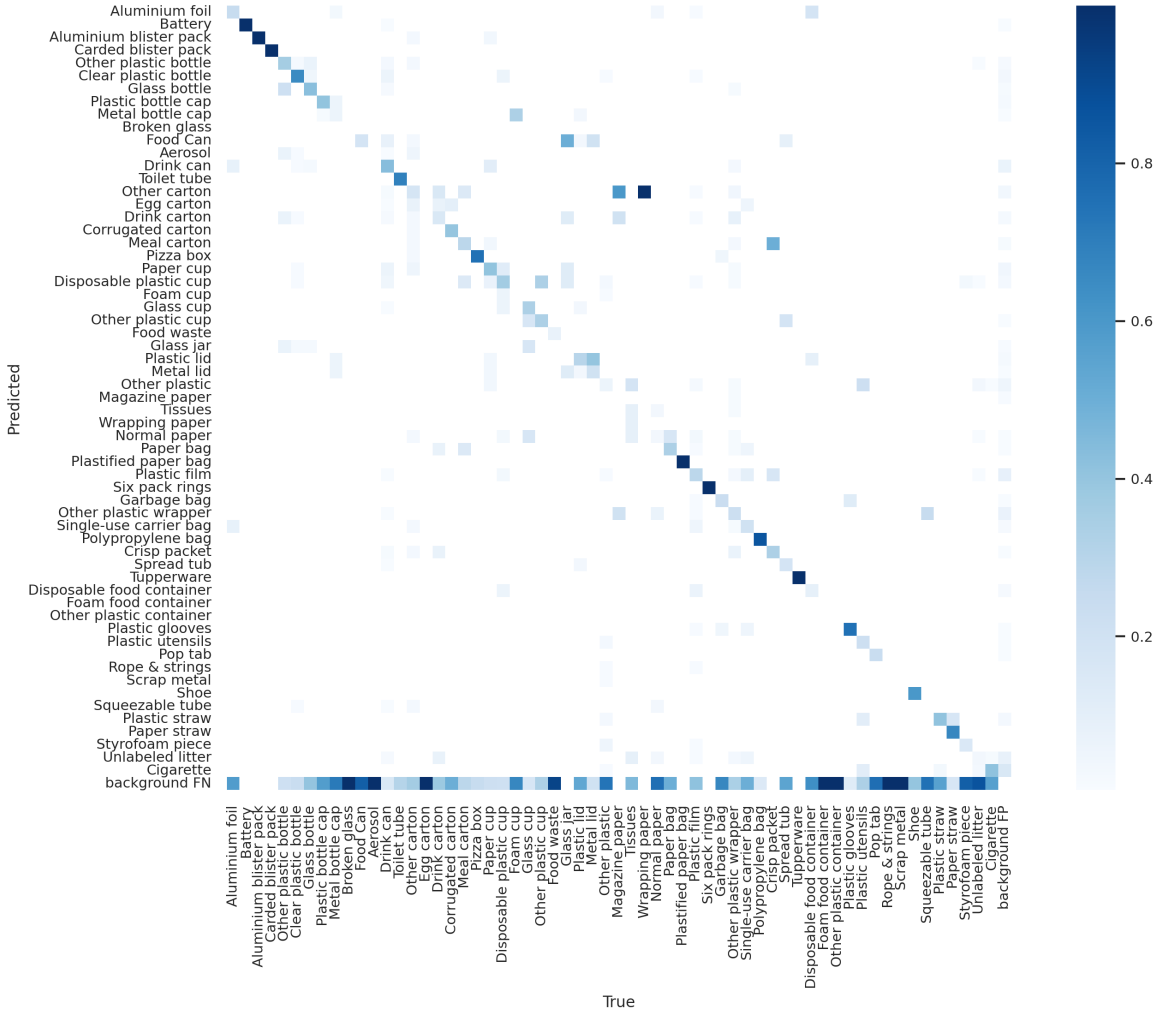

Figura 5.20 – Matriz de confusão do treinamento utilizando aumento de dados manual e artificial no conjunto TACO.

6 Conclusões

A proposta deste trabalho foi obter um modelo de Aprendizado Profundo para a detecção e classificação de resíduos sólidos. Inicialmente foram feitas comparações entre diferentes modelos e, após ter selecionado o melhor modelo, foram utilizadas técnicas de aperfeiçoamento com o objetivo de promover uma melhoria adicional ao seu desempenho.

O modelo escolhido foi o TL-YOLOv7-E6E, que utiliza transferência de aprendizado, visto que este modelo apresentou os melhores resultados nos testes iniciais, tendo um valor de mAP@0.5 14% melhor do que o segundo melhor modelo.

Nos testes realizados com o modelo que classifica os resíduos sólidos em 7 classes, o valor obtido para o mAP@0.5 foi de 0.2202. Com o objetivo de obter resultados ainda melhores, foi utilizada a técnica de aperfeiçoamento de aumento de dados, assim resultando em uma melhora de 24.8% no mAP@0.5, que passou a ser de 0.2748. Com o objetivo de verificar a efetividade do modelo, os resultados foram comparados com o trabalho de (MAJCHROWSKA et al., 2022), já que na pesquisa citada foi realizada a classificação do mesmo conjunto de dados e com as mesmas classificações que neste trabalho. Em comparação com o trabalho citado, o resultado obtido para o mAP@0.5, por esse trabalho, obteve uma melhora de 69.6%.

Além disso, foram feitos testes classificando os resíduos sólidos em 60 classes, de acordo com a divisão oficial da TACO (PROENÇA; SIMÕES, 2020). Neste caso havia várias classes sub representadas, então o primeiro passo foi realizar o aumento de dados manual com fotos obtidas online e tiradas pelos autores do trabalho. Com isso, o resultado para o mAP@0.5 foi de 0.3495, apresentando uma melhora de 49.55% em relação ao mesmo modelo sem o aumento de dados. Com o objetivo de obter resultados melhores foi feito o aumento de dados de forma artificial variando o brilho, a saturação e o contraste das imagens existentes, porém, após realizar esse aumento, houve uma diminuição de 2.3% no valor do mAP@0.5. Essa piora possivelmente ocorreu devido ao fato de que os exemplos gerados artificialmente possuem características discrepantes dos demais exemplos.

Em relação a trabalhos futuros envolvendo esse tema, é interessante mencionar a necessidade de um aumento na quantidade de exemplos de algumas classes específicas dos bancos de dados usados, já que eles são muito desbalanceados e isso com certeza afetou o desempenho do modelo.

Além de resolver o problema do desbalanceamento no conjunto de imagens, seria interessante que trabalhos futuros adicionassem mais imagens de ambientes que representassem melhor a realidade no Brasil, já que o conjunto é composto principalmente de fotos tiradas no exterior que, em alguns casos, não reflete a situação local.

Por fim, outro ponto que pode ser explorado em trabalhos futuros é a utilização de modelos ainda mais leves, buscando ser o mais eficiente possível no processamento, sem uma redução no desempenho.

Referências

ABRIHA, D.; SRIVASTAVA, P. K.; SZABÓ, S. Smaller is better? Unduly nice accuracy assessments in roof detection using remote sensing data with machine learning and k-fold cross-validation. **Heliyon**, v. 9, n. 3, e14045, 2023. ISSN 2405-8440. DOI: https://doi.org/10.1016/j.heliyon.2023.e14045. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2405844023012525>. Citado na p. 23.

ALMEIDA, A. P. G. S. de; BARROS VIDAL, F. de. **Turning old models fashion again: Recycling classical CNN networks using the Lattice Transformation**. 2021. arXiv: 2109.13885 [cs.CV]. Citado na p. 17.

AMBIENTE LEGAL. **A crise chega ao lixo**. 2016. Disponível em: <https://www.ambientelegal.com.br/a-crise-chega-ao-lixo/>. Acesso em: 17 fev. 2023. Citado na p. 13.

BERRAR, D. Cross-Validation. In: jan. 2018. ISBN 9780128096338. DOI: 10.1016/B978-0-12-809633-8.20349-X. Citado nas pp. 23, 24.

CAROLIS, B. D.; LADOGANA, F.; MACCHIARULO, N. YOLO TrashNet: Garbage Detection in Video Streams. In: 2020 IEEE Conference on Evolving and Adaptive Intelligent Systems (EAIS). 2020. P. 1–7. DOI: 10.1109/EAIS48028.2020.9122693. Citado nas pp. 28, 30, 31.

CONLEY, G.; ZINN, S. C.; HANSON, T.; MCDONALD, K.; BECK, N.; WEN, H. Using a deep learning model to quantify trash accumulation for cleaner urban stormwater. **Computers, Environment and Urban Systems**, v. 93, p. 101752, 2022. ISSN 0198-9715. DOI: https://doi.org/10.1016/j.compenvurbsys.2021.101752. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0198971521001599>. Citado nas pp. 27, 31.

CÓRDOVA, M.; PINTO, A.; HELLEVIK, C. C.; ALALIYAT, S. A.-A.; HAMEED, I. A.; PEDRINI, H.; S. TORRES, R. da. **PlastOPol: A Dataset for Litter Detection**. Versão 1.0. Zenodo, jan. 2022. DOI: 10.5281/zenodo.5829156. Disponível em: <https://doi.org/10.5281/zenodo.5829156>. Citado na p. 30.

DEVELOPERS BREACH. **Convolutional Neural Network | Deep Learning**. 2020. Disponível em: <https://developersbreach.com/convolution-neural-network-deep-learning/>. Acesso em: 17 fev. 2023. Citado na p. 20.

FUKUSHIMA, K. Neocognitron: A Self-Organizing Neural Network Model for a Mechanism of Pattern Recognition Unaffected by Shift in Position. **Biological Cybernetics**, v. 36, p. 193–202, 1980. Citado na p. 19.

G., V.; VUTKUR, P.; P., V. Food classification using transfer learning technique. **Global Transitions Proceedings**, v. 3, n. 1, p. 225–229, 2022. International Conference on Intelligent Engineering Approach(ICIEA-2022). ISSN 2666-285X. DOI: https://doi.org/10.1016/j.gltp.2022.03.027. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2666285X22000334>. Citado na p. 24.

GARCÍA-AGUILAR, I.; GARCÍA-GONZÁLEZ, J.; LUQUE-BAENA, R. M.; LÓPEZ-RUBIO, E. Automated labeling of training data for improved object detection in traffic videos by fine-tuned deep convolutional neural networks. **Pattern Recognition Letters**, v. 167, p. 45–52, 2023. ISSN 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2023.01.015. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0167865523000223>. Citado na p. 18.

GIRSHICK, R.; DONAHUE, J.; DARRELL, T.; MALIK, J. **Rich feature hierarchies for accurate object detection and semantic segmentation**. 2014. arXiv: 1311.2524 [cs.CV]. Citado na p. 21.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. **Deep Learning**. MIT Press, 2016. http://www.deeplearningbook.org. Citado nas pp. 14, 17–20, 23, 24, 31, 32.

GUNDUPALLI, S. P.; HAIT, S.; THAKUR, A. A review on automated sorting of source-separated municipal solid waste for recycling. **Waste Management**, v. 60, p. 56–74, 2017. Special Thematic Issue: Urban Mining and Circular Economy. ISSN 0956-053X. DOI: https://doi.org/10.1016/j.wasman.2016.09.015. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0956053X16305189>. Citado na p. 14.

HE, K.; GKIOXARI, G.; DOLLÁR, P.; GIRSHICK, R. **Mask R-CNN**. 2018. arXiv: 1703.06870 [cs.CV]. Citado na p. 21.

HOORNWEG DANIEL; BHADA-TATA, P. What a Waste : A Global Review of Solid Waste Management. World Bank, Washington, DC, 2012. http://hdl.handle.net/10986/17388. Citado na p. 13.

JIA, T.; KAPELAN, Z.; DE VRIES, R.; VRIEND, P.; PEEREBOOM, E. C.; OKKERMAN, I.; TAORMINA, R. Deep learning for detecting macroplastic litter in water bodies: A review. **Water Research**, v. 231, p. 119632, 2023. ISSN 0043-1354. DOI: https://doi.org/10.1016/j.watres.2023.119632. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0043135423000672>. Citado na p. 17.

LEE, Y.; HWANG, J.-w.; LEE, S.; BAE, Y.; PARK, J. **An Energy and GPU-Computation Efficient Backbone Network for Real-Time Object Detection**. 2019. arXiv: 1904.09730 [cs.CV]. Citado na p. 21.

LI, Q.; ZHAO, C.; HE, X.; CHEN, K.; WANG, R. The Impact of Partial Balance of Imbalanced Dataset on Classification Performance. **Electronics**, v. 11, n. 9, 2022. ISSN 2079-9292. DOI: 10.3390/electronics11091322. Disponível em: <https://www.mdpi.com/2079-9292/11/9/1322>. Citado na p. 23.

LIN, T.; DOLLÁR, P.; GIRSHICK, R. B.; HE, K.; HARIHARAN, B.; BELONGIE, S. J. Feature Pyramid Networks for Object Detection. **CoRR**, abs/1612.03144, 2016. arXiv: 1612.03144. Disponível em: <http://arxiv.org/abs/1612.03144>. Citado na p. 22.

LONG, X.; DENG, K.; WANG, G.; ZHANG, Y.; DANG, Q.; GAO, Y.; SHEN, H.; REN, J.; HAN, S.; DING, E.; WEN, S. **PP-YOLO: An Effective and Efficient Implementation of Object Detector**. 2020. arXiv: 2007.12099 [cs.CV]. Citado na p. 21.

MAJCHROWSKA, S.; MIKOŁAJCZYK, A.; FERLIN, M.; KLAWIKOWSKA, Z.; PLANTYKOW, M. A.; KWASIGROCH, A.; MAJEK, K. Deep learning-based waste detection in natural and urban environments. **Waste Management**, v. 138, p. 274–284, 2022. ISSN 0956-053X. DOI: https://doi.org/10.1016/j.wasman.2021.12.001. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0956053X21006474>. Citado nas pp. 25, 28, 29, 32, 34, 37, 41, 42, 49.

MATSUGU, M.; MORI, K.; MITARI, Y.; KANEDA, Y. Subject independent facial expression recognition with robust face detection using a convolutional neural network. **Neural Networks**, v. 16, n. 5, p. 555–559, 2003. Advances in Neural Networks Research: IJCNN '03. ISSN 0893-6080. DOI: https://doi.org/10.1016/S0893-6080(03)00115-1. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0893608003001151>. Citado na p. 19.

PELLICER, L. F. A. O.; FERREIRA, T. M.; COSTA, A. H. R. Data augmentation techniques in natural language processing. **Applied Soft Computing**, v. 132, p. 109803, 2023. ISSN 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2022.109803. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1568494622008523>. Citado nas pp. 17, 24, 35.

PROENÇA, P. F.; SIMÕES, P. **TACO: Trash Annotations in Context for Litter Detection**. arXiv, 2020. DOI: 10.48550/ARXIV.2003.06975. Disponível em: <https://arxiv.org/abs/2003.06975>. Citado nas pp. 15, 29, 30, 37, 40, 44, 49.

PY IMAGE SEARCH. **Intersection over Union (IoU) for object detection**. 2016. Disponível em: <https://pyimagesearch.com/2016/11/07/intersection-over-union-iou-for-object-detection/>. Acesso em: 12 mai. 2023. Citado na p. 26.

SALAS, J.; BARROS VIDAL, F. de; MARTINEZ-TRINIDAD, F. Deep Learning: Current State. **IEEE Latin America Transactions**, v. 17, n. 12, p. 1925–1945, 2019. DOI: 10.1109/TLA.2019.9011537. Citado na p. 17.

SMOLA, A.; VISHWANATHAN, S. **Introduction to Machine Learning**. The Press Syndicate of the University of Cambridge, 2010. https://alex.smola.org/drafts/thebook.pdf. Citado na p. 23.

TACO. **TACO Dataset**. 2023. Disponível em: <http://tacodataset.org/>. Acesso em: 30 mai. 2023. Citado na p. 33.

TAN, M.; PANG, R.; LE, Q. V. EfficientDet: Scalable and Efficient Object Detection. In: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). 2020. P. 10778–10787. DOI: 10.1109/CVPR42600.2020.01079. Citado na p. 28.

TOWARDS DATA SCIENCE. **YOLOv7: A Deep Dive into the Current State-of-the-Art for Object Detection**. 2022. Disponível em: <https://towardsdatascience.com/yolov7-a-deep-dive-into-the-current-state-of-the-art-for-object-detection-ce3ffedeeaeb>. Acesso em: 17 mai. 2023. Citado na p. 23.

TSENG, Y.-H.; JIANG, F.-J. A comment on the training of unsupervised neural networks for learning phases. **Results in Physics**, v. 40, p. 105832, 2022. ISSN 2211-3797. DOI: https://doi.org/10.1016/j.rinp.2022.105832. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2211379722004843>. Citado na p. 19.

WANG, C.-Y.; BOCHKOVSKIY, A.; LIAO, H.-Y. M. **YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors**. arXiv, 2022. DOI: 10.48550/ARXIV.2207.02696. Disponível em: <https://arxiv.org/abs/2207.02696>. Citado nas pp. 14, 17, 21, 31, 32, 34, 37, 39, 40.

WANG, C.-Y.; LIAO, H.-Y. M.; YEH, I.-H. **Designing Network Design Strategies Through Gradient Path Analysis**. 2022. arXiv: 2211.04800 [cs.CV]. Citado na p. 21.

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**. 2019. arXiv: 1911.11929 [cs.CV]. Citado na p. 21.

WANG, X.; XIE, L.; DONG, C.; SHAN, Y. **Real-ESRGAN: Training Real-World Blind Super-Resolution with Pure Synthetic Data**. 2021. arXiv: 2107.10833 [eess.IV]. Citado nas pp. 33, 36.

WANG, Y.; VINOGRADOV, A. Simple is good: investigation of history-state ensemble deep neural networks and their validation on rotating machinery fault diagnosis. **Neurocomputing**, p. 126353, 2023. ISSN 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2023.126353. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0925231223004769>. Citado na p. 17.

WU, T.-W.; ZHANG, H.; PENG, W.; LÜ, F.; HE, P.-J. Applications of convolutional neural networks for intelligent waste identification and recycling: A review. **Resources, Conservation and Recycling**, v. 190, p. 106813, 2023. ISSN 0921-3449. DOI: https://doi.org/10.1016/j.resconrec.2022.106813. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921344922006450>. Citado na p. 20.

YE, A.; PANG, B.; JIN, Y.; CUI, J. A YOLO-Based Neural Network with VAE for Intelligent Garbage Detection and Classification. In: 2020 3rd International Conference on Algorithms, Computing and Artificial Intelligence. Sanya, China: Association for Computing Machinery, 2021. (ACAI 2020). ISBN 9781450388115. DOI: 10.1145/3446132.3446400. Disponível em: <https://doi.org/10.1145/3446132.3446400>. Citado nas pp. 27, 31.

ZHANG, Q.; YANG, Q.; ZHANG, X.; WEI, W.; BAO, Q.; SU, J.; LIU, X. A multi-label waste detection model based on transfer learning. **Resources, Conservation and Recycling**, v. 181, p. 106235, 2022. ISSN 0921-3449. DOI: https://doi.org/10.1016/j.resconrec.2022.106235. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0921344922000830>. Citado nas pp. 24, 28, 30, 32.