

**RECONHECIMENTO FACIAL COM INTELIGÊNCIA ARTIFICIAL
UTILIZANDO A PLATAFORMA ROBOFLOW**

**FACIAL RECOGNITION WITH ARTIFICIAL INTELLIGENCE
USING THE ROBOFLOW PLATFORM**

**RECONOCIMIENTO FACIAL CON INTELIGENCIA ARTIFICIAL
MEDIANTE LA PLATAFORMA ROBOFLOW**

Guilherme Rodrigues Matuck¹; Arthur Jackson Abreu Castro²; Lázaro Eduardo da Silva³;
Eduardo Gomes Carvalho⁴.

¹CEFET- MG - guilherme130501@gmail.com

²CEFET- MG - arthurabreulvs@hotmail.com

³CEFET- MG - lazaro@cefetmg.br

⁴CEFET- MG - eduardogomes@cefetmg.br

Resumo: Este artigo tem como objetivo fornecer uma visão geral sobre reconhecimento facial em tempo real e discutir as melhores práticas para treinar modelos de detecção em tempo real usando a plataforma Roboflow. Serão discutidos os diferentes algoritmos de detecção de objetos em tempo real e como selecionar o algoritmo adequado para treinar modelos. Além disso, serão apresentados exemplos de conjuntos de dados de treinamento e modelos de detecção de objetos em tempo real, bem como estudos de caso de aplicativos reais. Por fim, serão discutidas considerações éticas e legais ao implementar sistemas de detecção de objetos em tempo real e como a plataforma Roboflow pode ajudar a garantir a privacidade e a segurança dos dados.

Palavras-chave: Roboflow, Reconhecimento facial, YOLO.

Abstract: This article aims to provide an overview of real-time facial recognition and discuss best practices for training real-time detection models using the Roboflow platform. Different real-time object detection algorithms and how to select the appropriate algorithm for training models will be discussed. Additionally, examples of training datasets and real-time object detection models will be presented, as well as case studies of real-world applications. Finally, ethical and legal considerations when implementing real-time object detection systems and how the Roboflow platform can help ensure data privacy and security will be discussed.

Key words: Roboflow, Facial recognition, YOLO.

Resumen: Este artículo tiene como objetivo proporcionar una descripción general del reconocimiento facial en tiempo real y analizar las mejores prácticas para entrenar modelos de detección en tiempo real utilizando la plataforma Roboflow. Se discutirán diferentes algoritmos de detección de objetos en tiempo real y cómo seleccionar el algoritmo apropiado para los modelos de entrenamiento. Además, se presentarán ejemplos de conjuntos de datos de entrenamiento y modelos de detección de objetos en tiempo real, así como estudios de casos de aplicaciones del mundo real. Finalmente, se discutirán las consideraciones éticas y legales al implementar sistemas de detección de objetos en tiempo real y cómo la plataforma Roboflow puede ayudar a garantizar la privacidad y seguridad de los datos.

Palabras llave: Roboflow, Reconocimiento facial, YOLO.

1. INTRODUÇÃO

A programação é o processo de criar instruções para um computador executar uma tarefa específica. As primeiras linguagens de programação foram desenvolvidas na década de 1950, incluindo a linguagem Assembly, que permitia aos programadores escrever código diretamente em linguagem de máquina. A evolução das linguagens de programação avançou rapidamente, com linguagens como COBOL, FORTRAN, C e C++ se tornando populares nas décadas seguintes (LECUN, 1998, p. 2278-2324).

Com o avanço da tecnologia, surgiram novas necessidades de programação, incluindo a necessidade de reconhecimento de imagens. O reconhecimento de imagens é um campo da inteligência artificial que envolve o uso de algoritmos para identificar objetos, pessoas e outros elementos em imagens e vídeos. Isso requer o uso de linguagens de programação avançadas, como Python e Matlab, e o uso de bibliotecas especializadas, como TensorFlow e OpenCV (GIRSHICK; DONAHUE; DARRELL; MALIK, 2014, p. 580-587).

Nos últimos anos, o reconhecimento de imagens tornou-se uma das aplicações mais populares da inteligência artificial, sendo utilizado em várias áreas, incluindo segurança, saúde, varejo e automação industrial. Com o desenvolvimento de plataformas como o Roboflow, a criação de modelos de reconhecimento de imagem tornou-se mais fácil e acessível, permitindo que empresas de todos os tamanhos possam aproveitar essa tecnologia.

A detecção de objetos em tempo real é uma técnica de processamento de imagens que permite identificar objetos específicos em um fluxo contínuo de imagens em tempo real. Essa técnica é amplamente utilizada em diversas áreas, incluindo vigilância, automação industrial, robótica, reconhecimento de faces e veículos autônomos.

Para realizar a detecção de objetos em tempo real, são usados algoritmos de visão computacional que analisam as características dos objetos em uma imagem, como cor, forma e textura, e as comparam com um banco de dados de objetos conhecidos. Esses algoritmos são capazes de identificar objetos em tempo real, mesmo em ambientes desafiadores, como em condições de iluminação ruins ou com objetos parcialmente ocultos (ZHANG; LI; QIAO, 2016, p. 1499-1503).

Com isso, há o senso comum de que para elaborar um sistema de detecção de objetos em tempo real é necessário um conhecimento aprofundado no tema e que apenas poucos profissionais são capazes de elaborar tal coisa. Com isso, tem-se como objetivo quebrar esse dogma e elaborar uma metodologia acessível e facilmente replicável.

2. REVISÃO BIBLIOGRÁFICA

Neste artigo, o foco é o estudo de uma metodologia acessível de baixa

complexibilidade e baixo processamento, para isso foi necessário reduzir os parâmetros possíveis e reduzir a quantidade de dados necessárias. Acerca de tudo, vale um entendimento mais aprofundado sobre isso para ser possível aprimorar os parâmetros e conseguir definir estes com melhor eficiência.

2.1 Aprendizado de máquina

Aprendizado de máquina, também chamado de machine learning, é um subcampo da ciência da computação onde o objeto de estudo é a elaboração de um algoritmo que visa extrair padrões a partir de grandes volumes de dados (RUBACK,2021). Acerca disso, as máquinas são treinadas para melhorarem seu desempenho ao longo do tempo, conforme realizam a tarefa. Os algoritmos de aprendizado de máquina são aplicados de diversas maneiras, abordando desde reconhecimento de padrões até tomadas de decisões complexas (BURKOV 2020).

2.2 Detalhes sobre a detecção de objetos em tempo real

A detecção de objetos em tempo real é uma técnica de visão computacional que permite que um sistema detecte objetos em tempo real, em vez de apenas em imagens estáticas. Isso é alcançado por meio da aplicação de algoritmos de detecção de objetos em vídeos em tempo real.

Os algoritmos de detecção de objetos em tempo real são capazes de identificar objetos em um fluxo de vídeo em tempo real, o que é essencial para muitas aplicações em áreas como vigilância, robótica e automação industrial.

Vale ressaltar que, de maneira simplificada, a máquina compara o que ela está vendo com o banco de dados que ela possui de seu treinamento (CUNHA, 2019). Este treinamento é semelhante a uma criança aprendendo sobre a diferença de um tomate e uma maçã, por exemplo, onde ambos possuem formatos semelhantes e cores semelhantes.

2.3 Introdução à plataforma Roboflow

Roboflow é uma plataforma de visão computacional que permite aos usuários treinar e implantar modelos de detecção de objetos em tempo real. A plataforma simplifica muitos dos desafios associados à construção de modelos de visão computacional, incluindo pré-processamento de dados, treinamento e implantação.

Com a plataforma Roboflow, os usuários podem treinar modelos de detecção de objetos em tempo real para uma ampla gama de aplicações, incluindo reconhecimento facial, detecção de objetos em vídeos de vigilância e navegação autônoma. A plataforma é

uma solução poderosa para empresas e organizações que desejam usar a detecção de objetos em tempo real para melhorar seus processos e serviços.

A plataforma permite diversas configurações de imagens, com camadas de edição e processamento, como saturação, brilho, inverter e angulação, por exemplo. É uma plataforma paga, mas pode ser gratuita caso o usuário deixe o projeto público.

3. CRIAÇÃO DE UM CONJUNTO DE DADOS DE TREINAMENTO

A criação de um conjunto de dados de treinamento é um dos passos mais importantes no processo de detecção de objetos em tempo real. O conjunto de dados é uma coleção de imagens rotuladas que são usadas para ensinar um modelo de detecção de objetos a reconhecer padrões em novas imagens.

A plataforma Roboflow é uma ferramenta poderosa que pode ajudar a criar rapidamente conjuntos de dados de treinamento para detecção de objetos em tempo real. Usando o Roboflow, é possível importar e rotular imagens, além de ajustar e visualizar as anotações.

Para criar um conjunto de dados de treinamento eficaz, é importante seguir algumas melhores práticas, como garantir que as imagens estejam de alta qualidade, incluir uma ampla variedade de objetos em diferentes condições de iluminação e ângulos de visão, e manter um equilíbrio entre as classes de objetos para evitar tendências ou desequilíbrios no modelo treinado (LECUN; BENGIO; HINTON, 2015, p. 436-444).

Alguns exemplos de conjuntos de dados de treinamento incluem o COCO (Common Objects in Context), o Pascal VOC (Visual Object Classes) e o ImageNet, que contêm milhões de imagens rotuladas em dezenas de classes diferentes. Além disso, muitas empresas e organizações criam seus próprios conjuntos de dados de treinamento personalizados para atender às suas necessidades específicas (EVERINGHAM; VAN GOOL; WILLIAMS; WINN; ZISSERMAN, 2010, p. 303-338).

Com a plataforma Roboflow e seguindo as melhores práticas, é possível criar conjuntos de dados de treinamento de alta qualidade para detecção de objetos em tempo real, permitindo que os modelos sejam treinados com eficácia e precisão.

4. SELEÇÃO DE ALGORITMOS DE DETECÇÃO DE OBJETOS EM TEMPO REAL

Ao selecionar um algoritmo de detecção de objetos em tempo real, é importante considerar fatores como precisão, eficiência computacional e facilidade de uso. Existem vários algoritmos disponíveis no mercado, cada um com suas próprias vantagens e

desvantagens.

Um dos algoritmos mais populares é o YOLO (You Only Look Once), que é conhecido por sua velocidade e precisão. O YOLO divide a imagem em grade e, em seguida, para cada célula, o algoritmo prevê a probabilidade de haver um objeto e suas coordenadas de caixa delimitadora. Outro algoritmo popular é o SSD (Single Shot Detector), que é semelhante ao YOLO, mas é conhecido por ser ainda mais rápido.

Outro algoritmo que tem sido amplamente utilizado é o Faster R-CNN (Region- based Convolutional Neural Network), que é conhecido por sua precisão, mas é mais lento em comparação com o YOLO e o SSD. O Faster R-CNN usa uma rede neural convolucional para extrair recursos da imagem e, em seguida, propõe regiões de interesse onde objetos podem estar presentes. Em seguida, uma caixa delimitadora é gerada em torno de cada região proposta para identificar os objetos (TAIGMAN; RANZATO; WOLF, 2014, p.1701-1708).

Além desses, há vários outros algoritmos, como o RetinaNet, que foi projetado para lidar com objetos pequenos e difíceis de detectar, e o EfficientDet, que é conhecido por ser altamente eficiente em termos de uso de recursos computacionais.

Ao selecionar um algoritmo para treinar modelos de detecção de objetos em tempo real usando a plataforma Roboflow, é importante considerar a natureza dos objetos que você está tentando detectar, a velocidade e a precisão desejadas e os recursos computacionais disponíveis. Exemplos de algoritmos de detecção de objetos em tempo real incluem o YOLOv5, o COCO, o EfficientDet e o RetinaNet, todos disponíveis na plataforma Roboflow.

5. TREINAMENTO DO MODELO DE DETECÇÃO DE OBJETOS EM TEMPO REAL

Treinar um modelo de detecção de objetos em tempo real é uma tarefa crucial para garantir o bom desempenho da tecnologia de reconhecimento facial. Aqui, discutiremos como ajustar os hiperparâmetros do modelo para melhorar sua precisão e desempenho, bem como superar os desafios comuns enfrentados durante o processo de treinamento usando a plataforma Roboflow.

Ajustando os hiperparâmetros do modelo, como a taxa de aprendizado e o tamanho do lote, pode ajudar a melhorar a precisão e o desempenho do modelo. A taxa de aprendizado determina a velocidade com que o modelo ajusta seus pesos com base nos dados de treinamento, enquanto o tamanho do lote afeta a quantidade de dados que o modelo processa de uma vez. Além disso, a seleção do algoritmo de detecção de objetos adequado também pode influenciar o desempenho do modelo.

Ao treinar modelos de detecção de objetos em tempo real, podem surgir desafios comuns, como a falta de dados de treinamento, a variação de iluminação e a complexidade

do objeto a ser detectado. A plataforma Roboflow pode ajudar a superar esses desafios, fornecendo ferramentas para criar e gerenciar conjuntos de dados de treinamento, bem como oferecer opções de pré-processamento de imagem e aumento de dados. Um dos maiores desafios enfrentados ao treinar modelos de detecção de objetos em tempo real é a falta de dados de treinamento suficientes para cobrir todas as possíveis condições de iluminação, ângulos de visão e objetos. A plataforma Roboflow ajuda a superar esse problema, permitindo a criação de grandes conjuntos de dados de treinamento, além de fornecer ferramentas para aumentar a variabilidade dos dados (NELSON, 2020).

Outro desafio comum é a necessidade de ajustar constantemente os hiperparâmetros para maximizar o desempenho do modelo. A plataforma Roboflow ajuda a simplificar esse processo, permitindo que os usuários testem rapidamente diferentes combinações de hiperparâmetros e avaliem a precisão do modelo em tempo real.

Algumas das soluções de detecção de objetos em tempo real mais populares incluem o YOLOv3, que é conhecido por sua velocidade e precisão, e o SSD MobileNet, que é popular por sua eficiência energética. Outros modelos populares incluem o Faster R-CNN, que é conhecido por sua alta precisão e o RetinaNet, que é conhecido por sua capacidade de detectar objetos em escalas diferentes.

Independentemente do modelo escolhido, a plataforma Roboflow pode ajudar a maximizar a precisão e o desempenho do modelo, permitindo que os usuários ajustem facilmente os hiperparâmetros e criem conjuntos de dados de treinamento grandes e diversificados.

6. AVALIAÇÃO DO MODELO DE DETECÇÃO DE OBJETOS EM TEMPO REAL

A avaliação de modelos de detecção de objetos em tempo real é uma parte crucial do processo de desenvolvimento de sistemas de IA. Existem várias métricas que podem ser usadas para avaliar a precisão do modelo, sendo as mais comuns a precisão, o recall e o F1-score.

A precisão mede a proporção de detecções verdadeiramente positivas em relação ao total de detecções positivas. Por exemplo, se o modelo detectou 100 objetos e 80 deles foram verdadeiramente positivos, a precisão seria de 80%. No entanto, a precisão pode ser enganosa, especialmente em casos em que há muitos falsos positivos (ROSS; QUILTER, 2023).

O recall mede a proporção de detecções verdadeiramente positivas em relação ao número total de objetos presentes no conjunto de dados. Por exemplo, se houver 100 objetos no conjunto de dados e o modelo detectar corretamente 80 deles, o recall seria de 80%. O recall é uma métrica importante porque ajuda a evitar que o modelo ignore objetos

importantes.

O F1-score é uma média ponderada da precisão e do recall, que varia de 0 a 1. O F1-score é uma métrica importante porque leva em consideração tanto a precisão quanto o recall.

Existem outras métricas que podem ser usadas para avaliar modelos de detecção de objetos em tempo real, como a curva ROC (Receiver Operating Characteristic) e a matriz de confusão. Cada métrica tem seus prós e contras, e é importante escolher a métrica certa para o problema específico que está sendo resolvido (ZHANG, 2010).

Para avaliar modelos de detecção de objetos em tempo real, é necessário ter um conjunto de dados de teste que contenha exemplos de objetos que o modelo deve detectar. Esse conjunto de dados deve ser diferente do conjunto de dados de treinamento, para que o modelo seja avaliado em dados que não foram usados para treiná-lo.

Além disso, é importante considerar o tempo de execução do modelo, especialmente em aplicações de detecção de objetos em tempo real. O modelo deve ser capaz de executar com rapidez e eficiência em hardware com recursos limitados, como dispositivos móveis ou sistemas embarcados.

Exemplos de avaliação de modelos de detecção de objetos em tempo real incluem a avaliação do modelo YOLOv5 em um conjunto de dados de detecção de objetos em tempo real e a avaliação de um modelo de detecção de pessoas em imagens de vigilância. Em ambos os casos, as métricas de precisão, recall e F1-score foram usadas para avaliar a precisão do modelo.

7. INTEGRAÇÃO COM OUTRAS TECNOLOGIAS

A detecção de objetos em tempo real pode ser integrada com outras tecnologias para criar soluções mais avançadas e inteligentes. Uma das tecnologias que pode ser integrada é o reconhecimento de voz, permitindo que o sistema seja controlado por comandos de voz. Isso pode ser particularmente útil em situações em que as mãos estão ocupadas, como em ambientes de manufatura ou em carros autônomos. Além disso, a detecção de objetos em tempo real também pode ser integrada com o processamento de linguagem natural para criar sistemas que possam interpretar e responder a solicitações de voz mais complexas (REDMON; DIVVALA; GIRSHICK; FARHADI, 2016, p. 779-788).

Um exemplo de integração de tecnologias é a utilização de detecção de objetos em tempo real em conjunto com a tecnologia de reconhecimento facial para identificar indivíduos em tempo real. Por exemplo, essa integração pode ser utilizada para garantir a segurança em eventos públicos ou em aeroportos, onde a detecção de objetos em tempo real é utilizada para identificar objetos perigosos e a tecnologia de reconhecimento facial é

utilizada para identificar pessoas em uma lista de observação. Outro exemplo de integração de tecnologias é a utilização de detecção de objetos em tempo real em conjunto com a tecnologia de processamento de linguagem natural para criar assistentes virtuais inteligentes em casas inteligentes. O sistema pode detectar objetos em tempo real e interpretar comandos de voz para controlar as luzes, o termostato, o sistema de som e outros dispositivos em casa.

8. CONSIDERAÇÕES ÉTICAS E LEGAIS

A detecção de objetos em tempo real é uma tecnologia poderosa que tem o potencial de melhorar muitos aspectos da vida, desde a segurança pública até a automação industrial. No entanto, é importante considerar as implicações éticas e legais ao implementar essa tecnologia.

Uma das principais preocupações é a privacidade das pessoas. O reconhecimento facial em tempo real pode envolver a coleta de imagens e dados pessoais sem o consentimento das pessoas afetadas. Isso pode ser particularmente problemático em locais públicos, onde as pessoas têm a expectativa razoável de privacidade.

Outra preocupação é a precisão do sistema de detecção de objetos em tempo real. Se o sistema estiver sujeito a falsos positivos ou falsos negativos, pode resultar em consequências negativas para as pessoas afetadas. Por exemplo, se um sistema de detecção de objetos em tempo real for usado para identificar suspeitos de crimes, um falso positivo pode levar à prisão injusta de uma pessoa inocente.

Além disso, é importante considerar as implicações legais da detecção de objetos em tempo real. Dependendo do contexto em que a tecnologia é usada, pode haver leis que regulem seu uso. Por exemplo, em alguns países, é ilegal gravar imagens de pessoas sem o seu consentimento.

A plataforma Roboflow oferece várias ferramentas para ajudar a garantir a privacidade e a segurança dos dados. Por exemplo, a plataforma permite a anonimização de imagens e a remoção de metadados pessoais antes do treinamento do modelo. Além disso, a plataforma é compatível com vários provedores de nuvem que possuem medidas de segurança robustas para proteger os dados armazenados

9. METODOLOGIA

Para o desenvolvimento do projeto foi utilizado a metodologia PDCA (Plan-Do-Check-Act ou Planejar-Fazer-Verificar-Agir).

A etapa “Plan” consistiu na definição das metas, os parâmetros a serem utilizados e como coletar os dados.

A etapa “Do” consistiu na execução da coleta de dados, adicionando pré-processamentos e efeitos adicionais (Augmentations) para comparação e treinar a interface.

A etapa “Check” consistiu na comparação dos resultados e das variações destas com os pré-processamentos.

A etapa “Act” consistiu na implementação da melhor versão.

De início foi utilizado o rosto do aluno 1 como modelo principal, onde foram tiradas de 5 a 10 fotos em ângulos de 45°, totalizando 10 posições diferentes. Foi escolhido o ângulo de 45° visando um equilíbrio entre os detalhes pela posição e a quantidade de fotos, visando não ser necessário tantas fotos adicionais. O rosto de Aluno 2 foi utilizado uma quantidade menor de fotos, totalizando 25 apenas, onde também foram tiradas fotos nos mesmos ângulos, conforme a Figura 1 a seguir, que representa os ângulos divididos nas partições descritos anteriormente:

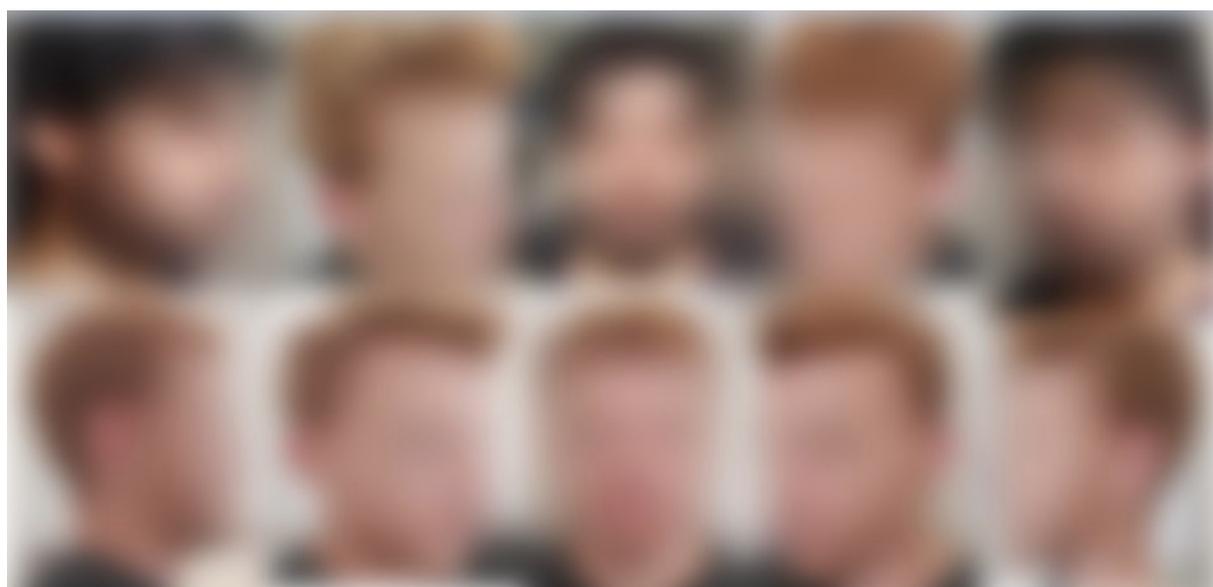


Figura 1: Coleta de dados para treinamentos e ângulos principais

Fonte: Autor

Tendo as fotografias em mãos, foram submetidas na plataforma, onde foram feitas as anotações e o pré-processamento das imagens, onde foi adicionado o ajuste automático de contraste usando equalização de histograma, adicionando também o efeito de contraste em 3 versões adicionais para comparação, sendo estes os de 0%, +-25% e +-50%. Com isso, foi utilizada a proporção aproximada de 70%, 20%, 10% para treinamento, validação e teste. Também foi aplicado o redimensionamento de imagens, definido em 640x640px. A proporção real foi de 71%, 19% e 10%. A proporção 70%, 20% e 10% é uma proporção muito comum em machine learning, tanto que a Roboflow sugere inicialmente esta proporção

(MATERIALS, s.d.). Visando a facilidade de replicação da metodologia, foi utilizada esta proporção.

Para início de testes, não foi utilizado processamento adicional de imagens. O projeto foi treinado utilizando a rede neural YOLO v8, visando uma metodologia mais intuitiva e de fácil reprodução, sendo possível fazer o treinamento no Google Colab disponibilizado pela Roboflow. Vale ressaltar que este treinamento é feito de forma gratuita, não necessitando de quaisquer tipos de pagamento.

Abordando diretamente a plataforma Roboflow, após o upload das imagens e separados os rostos com um quadrado simples, chega-se a esta tela (Figura 2):

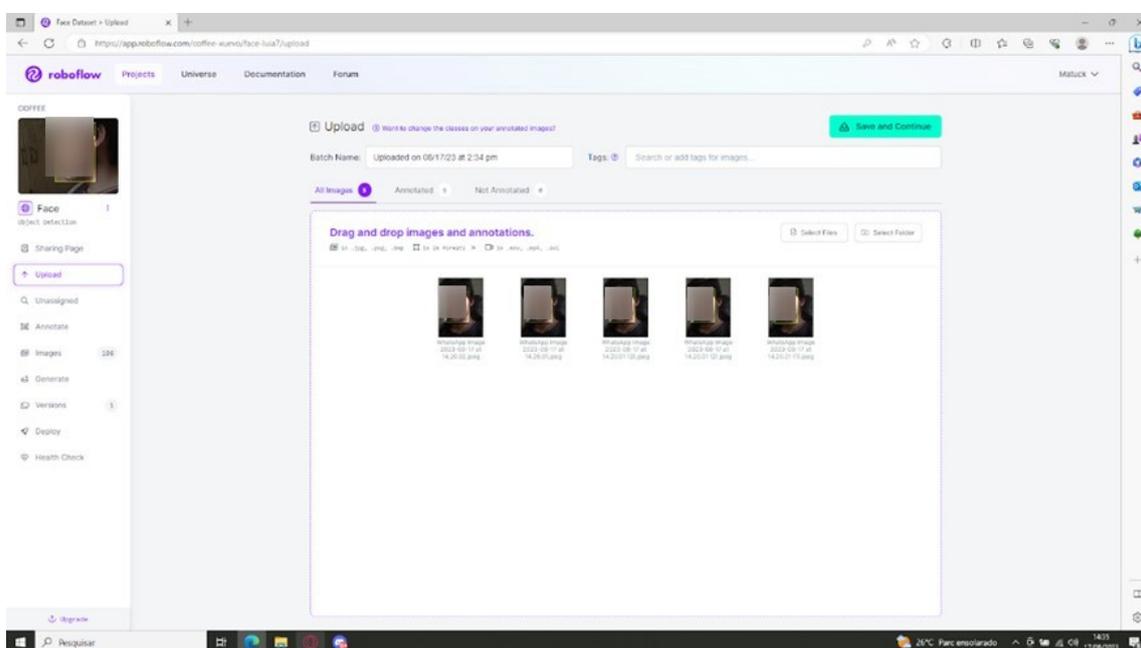


Figura 2: Upload de imagens

Fonte: Autor

No processo de salvar (Figura 3), já se encontra esta tela, onde também se define a % de treino, validação e teste:

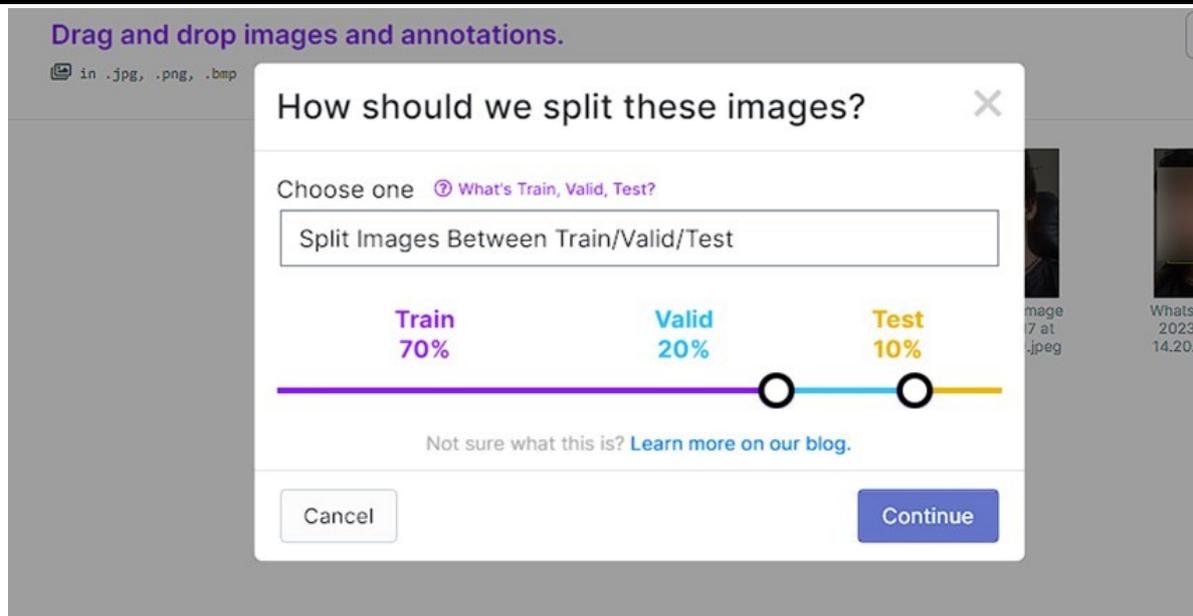


Figura 3: Treinamento, Validação e Teste

Fonte: Autor

Após isso, basta apenas confirmar as etapas, que estão previamente configuradas conforme foi descrito anteriormente. A Figura 4 demonstra a tela onde se aplica os pré-processamentos desejados. Na tela em questão não foram aplicados.

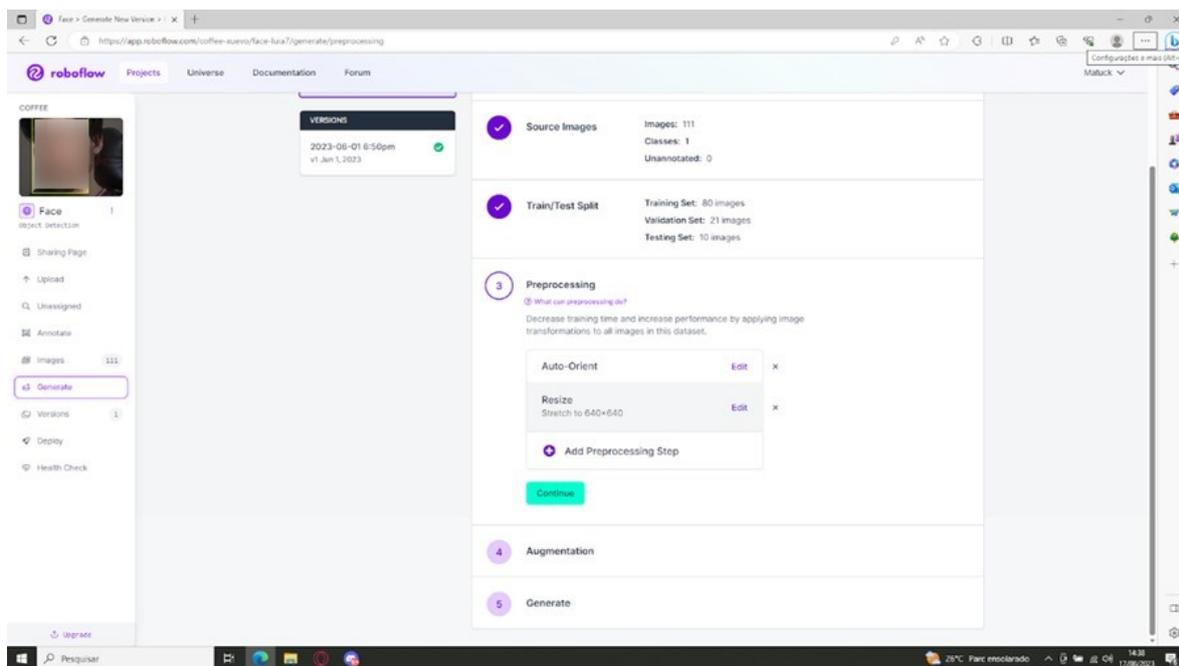


Figura 4: Etapas de pré-processamento e efeitos

Fonte: Autor

Colocando em prática a metodologia PDCA, onde tem-se como objetivo diminuir a quantidade de fotos necessárias, mas mantendo um resultado satisfatório para que seja viável implementar para outros alunos, avaliando que enquanto o rosto de Aluno 2 era reconhecido com somente 25 fotos enquanto o de Aluno 1 possuía 86 fotos, iniciou o processo para redução de fotos. Assim sendo, o processo foi repetido diversas vezes até encontrar um valor satisfatório.

10. RESULTADOS E DISCUSSÕES

Com o treinamento feito, iniciaram-se os testes, primeiramente com o rosto do Aluno 1. Assim sendo, houve um reconhecimento de 91%. Entretanto, sem a barba, o rosto de Aluno 2 teve um índice de reconhecimento de 81%. Já com a barba o reconhecimento foi um pouco inferior, atingindo índices próximos dos 73%, variando com base na iluminação.

Conforme foi dito anteriormente, foram feitas 3 versões adicionais para que fosse possível comparar os resultados e avaliar a interferência das modificações. Um fator negativo do pré-processamento é que ele torna o programa mais pesado, sendo mais demorado de ser treinado e de ser compilado. O projeto em questão possui 111 imagens e na versão natural levou menos de 2 minutos para ser treinado, enquanto na versão com pré-processamento e com ajuste de saturação levou 6 minutos e 9 segundos. É o triplo do tempo, sendo que existem projetos com dezenas de milhares de imagens carregadas para treinamento.

Acerca das versões, teve-se como resultado, em sequência da melhor para a pior: a natural, 50-50%, 0% e 25-25%, conforme as Figuras 7 e 8 a seguir exemplificam. Entretanto, como foi dito anteriormente, o pré-processamento possui uma necessidade maior de tempo de dedicação e de treinamento, sendo mais custoso para ser processado. Por essa razão, a versão natural foi escolhida como a mais adequada, por ter uma maior acuracidade e por ser mais leve no quesito de processamento também.

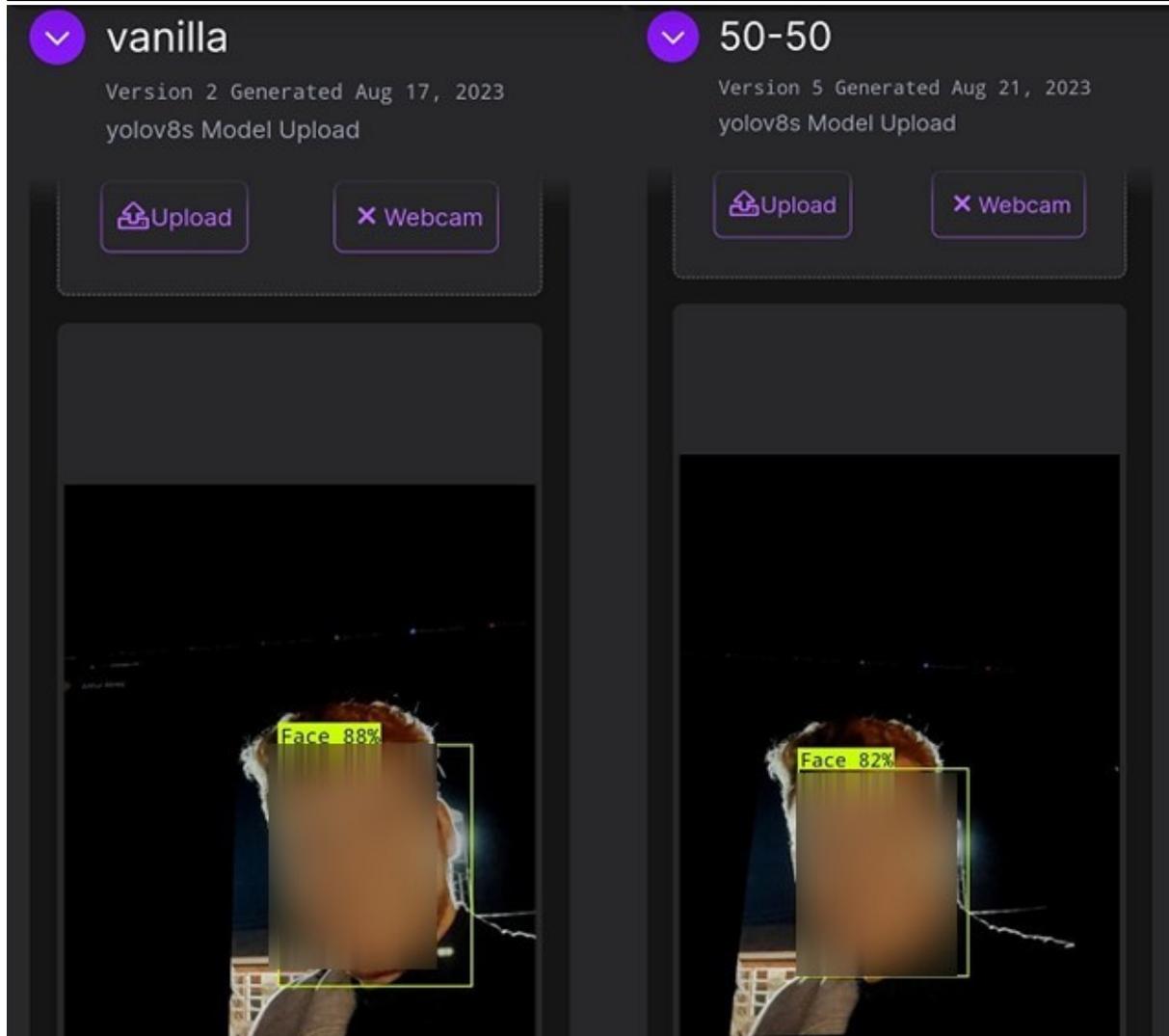


Figura 7: Natural e 50%
Fonte: Autor

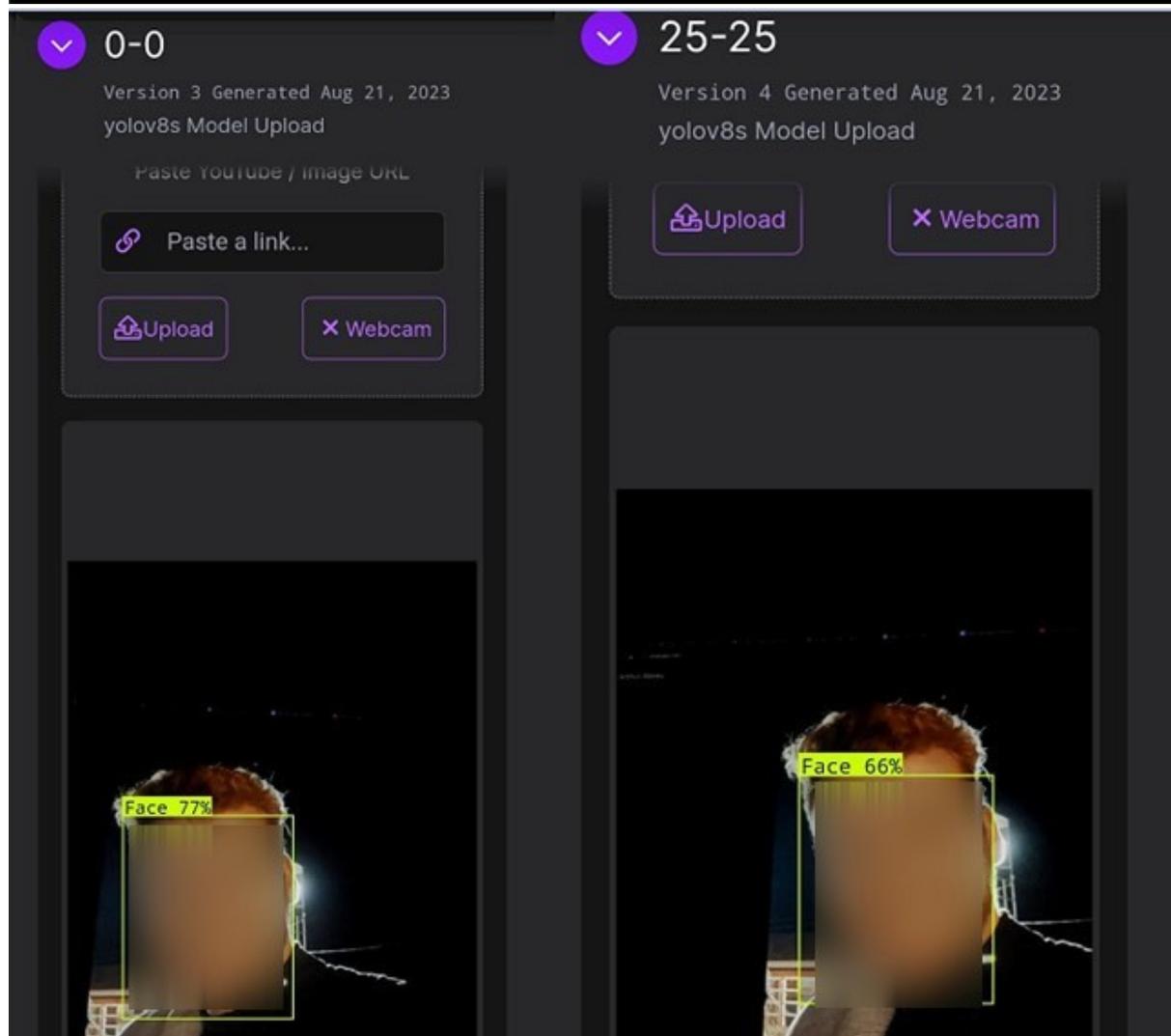


Figura 8: 0% e 25%
Fonte: Autor

Entende-se que os pré-processamentos e os efeitos adicionados auxiliariam em casos em que a câmera não pudesse captar uma imagem tão nítida, mas caso contrário mais atrapalharia do que ajudaria. Vale ressaltar que nas versões com mais efeitos e pré-processamentos houve um maior tempo para a estabilização da imagem e de processamento para que fosse reconhecido o rosto.

Assim sendo, conclui-se que de maneira geral a versão natural foi mais útil principalmente por ser a mais compacta. Acerca dos resultados desta, se mantiveram próximos da versão com 50% de saturação.

Referente ao processo de redução de imagens, vale ressaltar que em menores quantidades de fotos o pré-processamento e os efeitos adicionais atrapalharam, onde houve testes que chegaram ao reconhecimento de 7% como pior valor, enquanto sem os efeitos o pior valor de reconhecimento encontrado foi de 23%. Além disso, em quantidades menores

há uma dificuldade de distinguir os alunos, aonde muitas vezes chegava em uma sobreposição dos reconhecimentos, conforme a Figura 9.

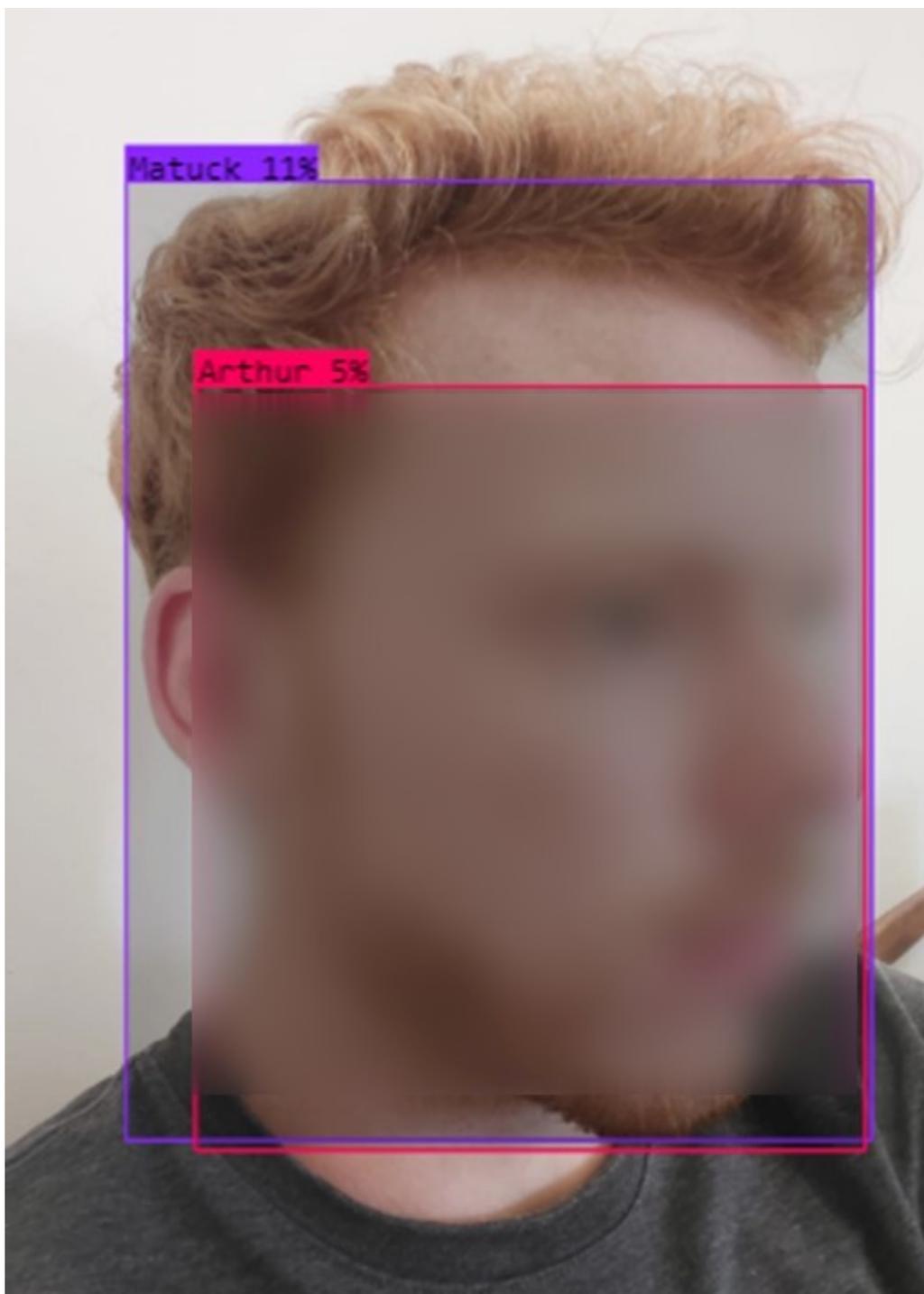


Figura 9: Rosto Aluno 1 com reconhecimento sobreposto

Fonte: Autor

Em um dos ciclos do processo PDCA, no de 29 fotos em que durante os testes foi

percebido que devido a escolha do aluno em manter a barba, havia uma maior dificuldade em ter o reconhecimento constante. Vale lembrar que a questão da iluminação e posição da câmera podem influenciar no reconhecimento, onde no momento o usuário pode acabar não percebendo que estas estão desfavoráveis. Na etapa em questão, havia o reconhecimento de 52% em situações favoráveis.

O último ciclo do processo PDCA chegou na quantidade de 40 fotos para cada aluno, onde atingiu 88% de reconhecimento com o rosto de Aluno 2, enquanto o rosto do Aluno 1 atingiu 90%. Vale ressaltar que em alguns momentos chegaram a atingir 98% de reconhecimento, o que é um valor ótimo.

Com base nisso, pode-se concluir que foi uma metodologia de fácil acesso, permitindo o treinamento para algo com uma complexidade alta de definição, o que abre um leque maior de possibilidades para demais aplicações, como por exemplo identificar patologias em construções, reconhecer objetos.

REFERÊNCIAS BIBLIOGRÁFICAS

BURKOV, Andriy. Machine learning engineering. Montreal, QC, Canada: True Positive Incorporated, 2020.

CUNHA, Kelvin Batista da. Detecção de objetos em 6-DoF em tempo real utilizando técnicas de aprendizagem profunda. 2019. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal de Pernambuco, Recife, 2019.

EVERINGHAM, Mark. VAN GOOL, Luc. WILLIAMS, Christopher. WINN, John; ZISSERMAN, Andrew. The Pascal Visual Object Classes (VOC) Challenge. International Journal of Computer Vision, v. 88, n. 2, p. 303-338, 2010.

GIRSHICK, Ross. DONAHUE, Jeff. DARRELL, Trevor. MALIK, Jitendra. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, p. 580-587, 2014.

LECUN, Yann. BENGIO, Yoshua. HINTON, Geoffrey. Deep learning. Nature, v. 521, n. 7553, p. 436-444, 2015.

LECUN, Yann. et al. Gradient-based learning applied to document recognition. Proceedings of the IEEE, v. 86, n. 11, p. 2278-2324, 1998.

NELSON, Joseph. Roboflow Blog, 28 de Outubro de 2020. Disponível em <<https://blog.roboflow.com/train-test-split-with-roboflow/>>

MATERIALS, Course. (s.d.). Convolutional neural networks. Disponível em: <http://cs231n.github.io/convolutional-networks/>. Acesso em: 22 mai. 2023.

REDMON, Joseph. DIVVALA, Santosh. GIRSHICK, Ross. FARHADI, Ali. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, p. 779-788, 2016.

ROSS, Beverley; QUILTER, Ben. Faces of discrimination: gender, race and facial recognition technology. Disponível em: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3185980. Acesso em: 5 abr. 2023.

RUBACK, Livia; AVILA, Sandra; CANTERO, Lucia. Vieses no aprendizado de máquina e suas implicações sociais: Um estudo de caso no reconhecimento facial. In: Anais do II Workshop sobre as Implicações da Computação na Sociedade. SBC, 2021. p. 90-101.

TAIGMAN, Yaniv. MING, Yang. RANZATO, Marc' Aurelio. WOLF, Lior. DeepFace: Closing the Gap to Human-Level Performance in Face Verification. 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 2014, pp. 1701-1708, doi: 10.1109/CVPR.2014.220.

ZHANG, Cha; ZHANG, Zhengyou. A survey of recent advances in face detection. 2010.

ZHANG, Kaipeng. ZHANG, Zhanpeng. LI, Zhifeng. QIAO, Yu. Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, v. 23, n. 10, p. 1499-1503, 2016.