

## APLICAÇÃO DE YOLOv8 E MARCADORES ARUCO PARA RECONHECIMENTO E MEDIÇÃO DE PARAFUSOS.

Williams da Conceição dos Santos <sup>(1)</sup> (2020006669@ifam.edu.br), Michaella Socorro Bruce Fialho <sup>(1)</sup> (michaella.fialho@ifam.edu.br), Nei Junior da Silva Farias (nei.farias@ifam.edu.br) <sup>(1)</sup>

<sup>(1)</sup> Instituto Federal de Educação, Ciência e Tecnologia do Amazonas (IFAM); Departamento de Processos Industriais - DPI

**RESUMO:** *Este estudo aborda a integração do YOLOv8 e Marcadores ArUco para aprimorar a detecção e medição de parafusos. O objetivo principal é desenvolver uma solução que irá permitir a identificação e a medição de parafusos com precisão utilizando visão computacional. A metodologia envolve a utilização de uma webcam para capturar imagens, onde o algoritmo YOLO realiza a detecção dos objetos, e os Marcadores ArUco são empregados para calcular as dimensões dos parafusos. O sistema foi implementado para capturar e salvar imagens dos objetos, além de registrar as medidas em uma planilha. A pesquisa foi realizada em ambiente controlado, com foco na integração dos sistemas. Os resultados demonstraram a eficiência do sistema em identificar e medir parafusos de forma automatizada e precisa mostrando que a combinação de técnicas de visão computacional pode aumentar a capacidade de sistemas robóticos em tarefas de manipulação e inspeção de objetos, sendo viável em processos de inspeção industrial e controle de qualidade.*

**PALAVRAS-CHAVE:** VISÃO COMPUTACIONAL, INTELIGÊNCIA ARTIFICIAL, DETECÇÃO DE OBJETOS, QUALIDADE.

## APPLICATION OF YOLOv8 AND ARUCO MARKERS FOR BOLT RECOGNITION.

**ABSTRACT:** *This study addresses the integration of YOLOv8 and ArUco Markers to improve screw detection and measurement. The main objective is to develop a solution that will allow the identification and measurement of screws accurately using computer vision. The methodology involves using a webcam to capture images, where the YOLO algorithm performs object detection, and ArUco Markers are used to calculate screw dimensions. The system was implemented to capture and save images of objects, in addition to recording measurements in a spreadsheet. The research was carried out in a controlled environment, focusing on systems integration. The results improved the system's efficiency in identifying and measuring screws in an automated and precise way, showing that the combination of computer vision techniques can increase the capacity of robotic systems in object manipulation and inspection tasks, being viable in industrial inspection processes. and quality control.*

**KEYWORDS:** COMPUTER VISION, ARTIFICIAL INTELLIGENCE, OBJECT DETECTION, QUALITY.

## 1. INTRODUÇÃO

A qualidade é um fator essencial em processos industriais, diretamente ligada à satisfação do cliente e à eficiência operacional. Juntamente com o controle de qualidade, se tornam fundamentais em qualquer processos industriais, garantindo que o produto final atenda aos requisitos de precisão, segurança e confiabilidade. A forma com que a qualidade é monitorada e assegurada evoluiu, incorporando novas tecnologias a fim de produzir mais, com menos recursos e mão de obra (Santos, 2015). Hoje, as indústrias contam com ferramentas de visão computacional e inteligência artificial para automatizar e aprimorar processos de inspeção, evitando ou minimizando erros, aumentando a produção, a eficiência e reduzido custos.

A inteligência artificial é a combinação de várias tecnologias que fazem com que máquinas e sistemas computacionais simulem o raciocínio humano, com isso, tomando decisões e automatizando tarefas. A partir de sua implementação pode ser usado em robôs, diagnósticos médicos, reconhecimentos de imagens, entre outros (Silva, 2023). Na indústria, a inteligência artificial é usada para aumentar os resultados do negócio, aumentando a produtividade, reduzindo custos e tempo.

A visão computacional é a ciência que estuda e desenvolve tecnologias que permitem que as máquinas enxerguem o ambiente ao nosso redor, podendo reconhecer e extrair informações do meio do ambiente ou objetos (Barelli, 2018). Essa tecnologia se tornou importante nas empresas para o controle de qualidade, permitindo uma análise mais detalhada dos objetos em tempo real. A implementação da visão computacional permite que às indústrias otimizem o trabalho na detecção de defeitos, melhorando a eficiência produtiva ao identificar problemas que seriam despercebidos em inspeções manuais (Borges, 2024). A visão computacional utiliza da inteligência artificial para simular as capacidades do cérebro humano para reconhecer e classificar objetos, identificando padrões.

Entre as tecnologias que envolve visão computacional, destacam-se o algoritmo YOLO e os Marcadores ArUco, que juntos podem oferecer soluções para aplicações onde requerem alta confiabilidade e precisão, auxiliando o trabalho humano, aumentando a produtividade com menos chances de desperdícios e erros.

Sendo desenvolvido pela Ultralytics, YOLO (*You Only Look Once*) é um algoritmo de detecção de objetos e segmentação de imagens em tempo real que se destaca por sua rapidez permitindo a identificação de múltiplos objetos em uma única imagem. Sendo amplamente usado em aplicações que envolvem desde câmeras de segurança, até veículos autônomos. Sua 8ª versão recebeu novas

funcionalidades e melhorias, suportando várias tarefas de IA de visão computacional, como detecção, segmentação, estimativa de pose, rastreamento e classificação de imagens (Ultralytics, 2023).

Os Marcadores ArUco é uma biblioteca de visão computacional que fornece identificação para marcadores fiduciais quadrados em imagens. Utiliza de marcadores, que são compostos por uma borda externa negra com quadrados brancos em seu interior que codifica um padrão binário único, sendo muito utilizado em realidade aumentada, robótica, rastreamento de movimento, além de serem usado para calibração de câmeras (Santos, 2022) (Munhoz-Salinas, 2012). Apesar de poder gerar seu marcador aruco no código, existe pela internet a disponibilidade de geradores online que podem gerar os marcadores de forma simplificada (Sarcanjo, 2023).

O objeto a ser analisado é o parafuso, por ser amplamente utilizado em diversas estruturas, principalmente em projetos de máquinas. Existem parafusos para fixar objetos, como os parafusos de fixação, e os parafusos que transmitem potência e movimento, como os parafusos de avanço.

Este estudo propõe a combinação dessas duas tecnologias para aprimorar a detecção e medição de parafusos. Através desta integração, o objetivo é desenvolver um sistema eficiente e automatizado que possa ser aplicado em processos de inspeção industrial e controle de qualidade, aumentando a capacidade em tarefas de manipulação e inspeção de objetos.

## 1.1 Objetivos

Este estudo tem como objetivo desenvolver uma solução que permita identificar e medir parafusos com precisão, utilizando visão computacional, de forma que demonstre o benefícios da união das duas tecnologias para auxiliar e melhorar atividades em ambientes industriais que envolvam medição e detecção, reduzindo a porcentagem de falhas que possam ocorrer na seleção de objetos na produção e a levando a uma redução significativa nos custos operacionais e de produção.

## 1.2 Justificativa

O sucesso e ou falha de um projeto pode depender da seleção correta e apropriada de componentes. A necessidade de um sistema identificar e medir parafusos com alta precisão é crucial para as indústrias, especialmente na manufatura e na montagem de produtos, onde o erro humano seria reduzido, evitando impactos negativos para indústria. A visão computacional oferece uma solução promissora para esses desafios, permitindo a automação e a precisão que são difíceis de alcançar manualmente.

A tecnologia YOLOv8, sendo uma das mais avançadas em detecção de objetos, oferece alta precisão e rapidez, tornando-a ideal para aplicações em tempo real. A adição dos marcadores ArUco proporciona uma camada adicional de precisão na medição e no posicionamento.

## 2. MATERIAIS E MÉTODOS

Neste estudo o *hardware* utilizado foi uma *webcam* da marca PC Camera, mas podendo ser utilizada qualquer *webcam* ou câmera que tenha uma conexão USB (Universal Serial Bus) com o computador ou notebook. Foi utilizada a linguagem de programação Python, devido a sintaxe simples, de fácil compreensão e sendo usadas em muitas aplicações, principalmente em inteligência artificial, e oferecendo uma gama de bibliotecas que aumentam a produtividade e reduz número de erros (Menezes, 2014).

Juntamente com o Python, foi usada o algoritmo YOLO, para aplicação de inteligência artificial para reconhecimento de parafusos em imagens em tempo real. Usando a versão 8, que é conhecido por ter mais alta precisão e velocidade em tarefas detecção comparado a seus antecessores. E o marcador ArUco para o desenvolvimento do código do sistema.

O softwares utilizado para este estudo foi o Pycharm, uma IDE (*Integrated Development Environment*, ou Ambiente de Desenvolvimento Integrado) especialmente usado em para linguagem de programação Python em ciência de dados e desenvolvimento web.

Para a medição dos parafusos, foram utilizados um paquímetro digital da marca Starrett, e um paquímetro analógico da marca Mitutoyo, este com tolerância de 0,02 mm.

Foi usado o Roboflow (<https://roboflow.com>), que é uma plataforma online que permite que os usuários criem suas próprias versões de visão computacional para atender sua necessidade com treinamento personalizado (Alexandrova *et.al*, 2015).

### 2.1 Treinamento do Sistema e Desenvolvimento do Algoritmo

Foram coletadas um conjunto de imagens de vários tipos de modelos, tamanhos, formatos e cores de parafusos, em diferentes posições, ângulos, assim como em diferentes intensidade de luminosidade. As imagens podem ser registradas por câmeras de celular. Utilizando o Roboflow, após criar um registro, foi realizado o upload das imagens para fazer a marcação dos parafusos com *bounding box*, caixas de marcação feito pelo próprio usuário. As imagens foram redimensionadas e normalizadas, com as anotações sendo convertidas para o formato adequado para o treinamento.

No fim, deve-se escolher qual a versão que será usado no treinamento, caso YOLOv8, e um arquivo será baixado, necessário para utilização.

Após baixar o dataset, foi modificado os parâmetros na localização do arquivo para realizar o treinamento, também foi instalado o YOLOv8 no sistema pela própria IDE. O modelo foi treinado para reconhecer apenas parafusos, em 100 *epochs* (ou épocas), levando algumas horas, dependendo do hardware disponível. Após o treinamento, o sistema está apto para detectar os parafusos que serão visto pela câmera.

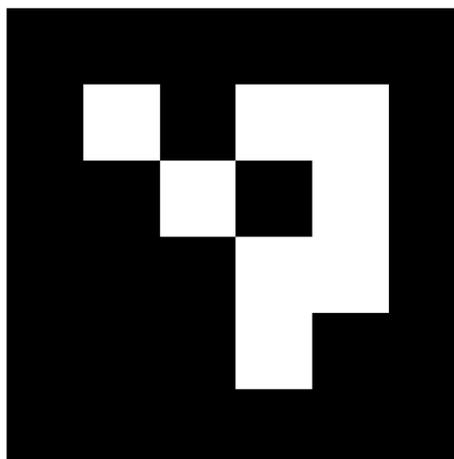
O modelo treinado foi validado com um conjunto de tipos de objetos, incluindo parafusos de diferentes tipos, para avaliar sua detecção para reduzir falsos positivos e negativos.

Em seguida, foi dado início ao desenvolvimento do ArUco, que foram utilizados para medir a posição e orientação dos parafusos, fornecendo um sistema de referência visual que permite identificar o parafuso detectado pelo YOLOv8. Marcador foi gerado e impresso para ser utilizado no projeto, sendo posicionado estrategicamente no campo de visão da câmera para fornecer os pontos de referência e facilitando a calibração do sistema.

Algoritmos de visão computacional foram desenvolvidos para detectar e decodificar os marcadores ArUco nas imagens capturadas pela câmera. A biblioteca OpenCV foi utilizada para facilitar o processo.

Com o marcador devidamente posicionado, a próxima etapa foi para desenvolver um algoritmo capaz de detectá-lo e decodificá-lo nas imagens capturadas câmera. A câmera foi configurada para capturar imagens em uma resolução adequada para que permitisse a detecção clara do marcador, assim como a taxa de quadro foi ajustada para garantir uma detecção em tempo real. O ajuste manual também foi necessário visto que o modelo da câmera tem essa possibilidade, assim como foi necessário ajustar a iluminação para melhor nitidez.

FIGURA 1. Marcador ArUco

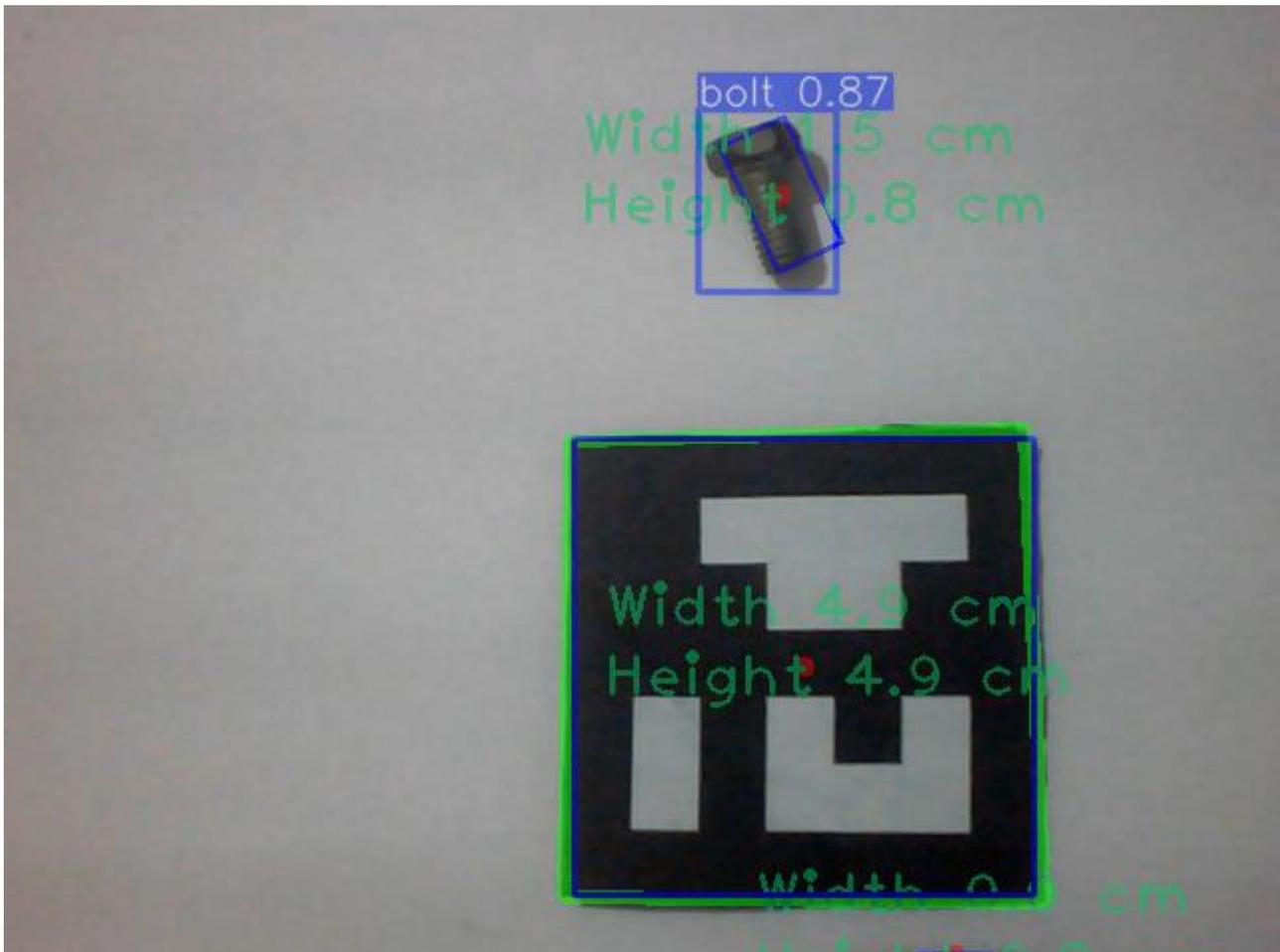


Fonte: Próprio Autor (2024)

A integração do Marcador ArUco com o YOLOv8 foi essencial para fornecer informações de referência adicionais que permitissem a medição precisa dos parafusos. Os dados de detecção dos marcadores foram sincronizados com os dados de detecção do YOLOv8, isso envolveu a fusão das coordenadas dos objetos detectados com as coordenadas do marcador. Utilizando a marcação como referência fixa, foi possível calcular as distâncias relativas entre o parafuso detectado e o marcador.

Quando o sistema detectou o parafuso, com confiança de 80% (0.80), foi programado para que registra-se três imagens, uma na visão do ArUco, outra na visão do YOLOv8 e uma com a união das duas imagens. Enquanto isso, o ArUco realiza as medições em tempo real com uma precisão adequada. Foi necessário a criação de uma barra de ajuste de brilho na janela do ArUco para dar mais nitidez e precisão nas medidas.

Figura 2. Visão do YOLO e Aruco integrados.



Fonte: Próprio Autor (2024)

## 2.2 Testes

Os parafusos foram colocados na região onde a câmera enxerga, próximo do marcador, os teste foram realizados para garantir que as medições dos parafusos detectados fossem precisas e consistentes. Foram capturadas várias imagens da detecção dos parafusos e do marcador ArUco utilizando a câmera configurada, sendo processadas pelo sistema de visão computacional. Além disso, o sistema cria automaticamente um arquivo estilo planilha .xlsx para salvar as medidas de largura, altura, tipo de objeto e o tempo em que a medida ocorreu.

As medições realizadas pelo sistema foram comparadas com as medições manuais com paquímetro digital e paquímetro analógico. Diferenças significativas entre as medições foram analisadas para identificar possíveis erros.

**Figura 3.** Medida com os paquímetros.



Fonte: Próprio Autor (2024)

A iluminação do ambiente foi ajustada de detectar melhor incidência de luz para uma melhor leitura, incluindo luz natural, luz artificial, pouca luz e sombra. A detecção do ArUco e dos parafusos foi avaliada para garantir que o sistema pudesse atuar corretamente em várias condições.

Parafusos foram colocados em diversos locais para identificar se haveria algum lugar que pudesse melhor ser detectado e medido. Foram também posicionados em diferentes ângulos e distâncias em relação à câmera. Isso testou a capacidade do sistema de detectar e medir independente da orientação e posição.

Objetos adicionais foram introduzidos ao campo de visão da câmera para avaliar a capacidade do sistema de detecção contra interferências visuais. O sistema foi monitorado para garantir que pudesse distinguir corretamente parafuso de outros objetos.

Ajuste finais foram feitos com base nos teste realizados com a intenção de deixar o sistema mais confiável. Os algoritmos foram refinados para melhorar a precisão e robustez, a calibração da câmera foi feita para garantir máxima precisão.

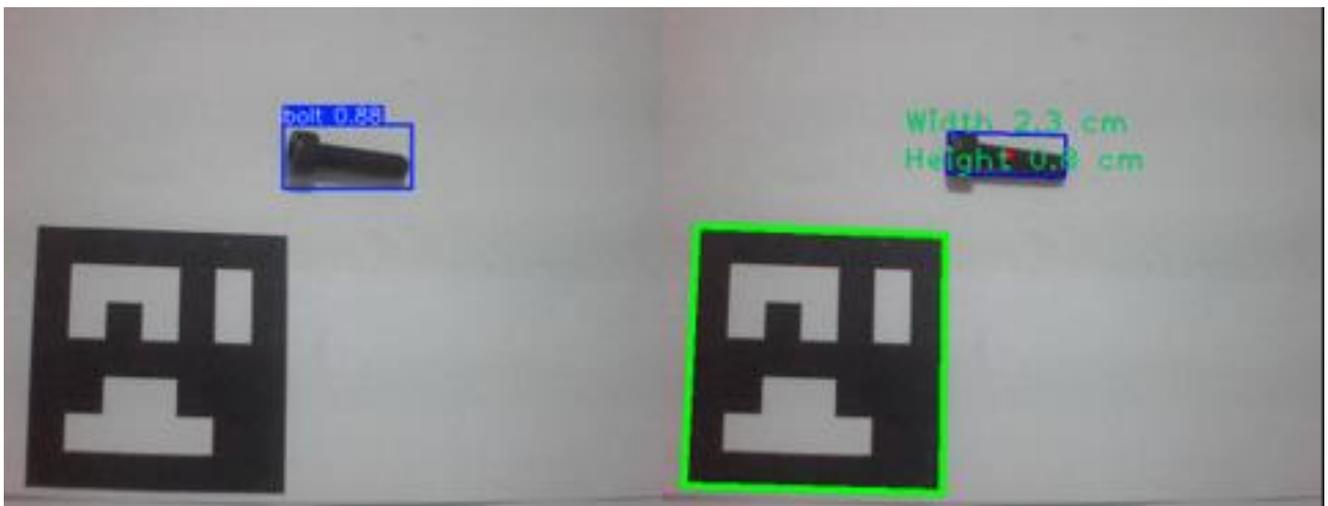
### 3. RESULTADOS

Os resultados obtidos a partir do desenvolvimento e implementação mostram-se promissores em vários aspectos.

O modelo YOLOv8, após o treinamento intensivo com um conjunto diversificado de imagens de parafusos, demonstrou alta precisão na detecção de parafusos em tempo real. Os testes realizados o modelo atingiu uma taxa de detecção acima de 80%, o que é considerado adequado, considerando limitações de hardware.

A integração do marcador ArUco permitiu que as medições das dimensões de largura e comprimento dos parafusos fossem precisas. Utilizando a biblioteca OpenCV, demonstraram ser eficazes em fornecer dados de referências para as medições. Em comparação com medições manuais realizadas por paquímetro digital e paquímetro analógico, as medições pelo ArUco apresentaram uma margem de erro inferior a 2%, indicando precisão e confiabilidade no sistema.

**Figura 5.** Visão do Aruco e YOLO separadas



Fonte: Próprio Autor (2024)

A detecção e medição foi eficaz em cenários com vários tipos de iluminação. O sistema também foi capaz de detectar os parafusos diante de outros objetos.

Devido a limitações de hardware (computador e câmera), o desempenho foi um pouco lento, porém o sistema ainda pode garantir a operação em tempo real, podendo processar as imagens e enviar os comandos para o braço robótico poder tomar as ações.

As imagens salvas pelo sistema mostraram ser bastante visíveis, permitindo ter uma detecção clara dos parafusos e das medições. Ajuste na resolução e de luminosidade contribuíram para qualidade das imagens.

Figura 6. Tabela em .xlsx gerada pelo sistema com as dimensões registradas

Date	Object	Width (cm)	Height (cm)
2024-07-28 02:08:54	Bolt	0,8	2,3
2024-07-28 02:08:58	Bolt	0,8	2,3
2024-07-28 02:09:25	Bolt	0,8	2,3
2024-07-28 02:09:45	Bolt	0,8	2,3
2024-07-28 02:10:00	Bolt	1,6	2,7
2024-07-28 02:10:28	Bolt	1,6	2,7
2024-07-28 02:10:54	Bolt	0,8	1,6
2024-07-28 02:10:54	Bolt	0,8	1,6
2024-07-28 02:11:14	Bolt	0,9	1,2
2024-07-28 02:11:36	Bolt	0,9	1,6
2024-07-28 02:12:05	Bolt	0,9	1,6

Fonte: Próprio Autor (2024)

#### 4. CONCLUSÃO

A integração de visão computacional e robótica neste projeto demonstrou ser uma abordagem eficaz para identificação e medição de parafusos, podendo ser usado tanto em ambientes industriais quanto em ambientes que envolvam automação que seja necessário um tipo de controle dimensional. A utilização do YOLOv8 para detecção e do Marcador Aruco para medição, mostrou ser um sistema robusto e confiável que pode ser implementado onde a manipulação de parafusos seja essencial.

A implementação de visão computacional para automação de tarefas pode proporcionar benefícios significativos, incluindo a redução de erros humanos, aumento da precisão e eficiência e na qualidade do produto final. A utilização dessas tecnologias, aliadas a automação e robótica, mostra-se como uma abordagem promissora para enfrentar desafios da manufatura.

Os testes realizados mostraram que o sistema pode operar eficientemente em diversas condições de iluminação e posicionamento, mantendo uma alta taxa de precisão e confiabilidade.

Isso abre possibilidades para a adoção de tecnologias avançadas em pequenas e médias empresas que buscam melhorar seus processos produtivos sem incorrer em altos custos.

## DECLARAÇÃO DE RESPONSABILIDADE

Os autores são os únicos responsáveis por este trabalho.

## REFERÊNCIAS

ALEXANDROVA, S.; TATLOCK, Z.; CAKMAK, M.; RoboFlow: A flow-based visual programming language for mobile manipulation tasks; IEEE International Conference on Robotics and Automation (ICRA), Seattle, WA, USA, 2015.

BARELLI, F.; Introdução à Visão Computacional: Uma abordagem prática com Python e OpenCV. Editora Casa do Código, 2018.

BORGES, M.; Como aplicar Visão Computacional na indústria. Venturus, 2024. Disponível em <https://www.venturus.org.br/insights/blog/como-aplicar-visao-computacional-na-industria>. Acesso em 05 de ago. 2024.

MENEZES, N. N. C.; Introdução à Programação com Python: Algoritmos e lógica de programação para iniciantes, 2a ed. Novatec, São Paulo, SP, Brasil, p. 24-26, 2014.

MUNOZ-SALINAS, R.. Aruco: a minimal library for augmented reality applications based on opencv. Universidad de Córdoba, v. 386, 2012.

SANTOS, H. I. A.; Sistema de localização de robôs móveis usando marcadores ArUco. 2022. 120 f. Monografia (Graduação em Engenharia Eletrônica) – Universidade de Brasília, Brasília, DF, p. 18, 2022.

SANTOS, A. A.; Gestão da Qualidade e Confiabilidade; Grupo Anima Educação, Belo Horizonte, MG, Brasil, p. 6 – 12, 2014.

SARCANJO, J.; Detecção de marcadores ArUco em streams de vídeo em tempo real com OpenCV. Medium, 2023. Disponível em <https://jonysarcanjo.medium.com/detecção-de-marcadores-aruco-em-streams-de-vídeo-em-tempo-real-com-opencv-9a3d99c667d7>. Acesso em: 25 jul. 2024.

SILVA, W.; Curso de Qualificação Profissional – Introdução a Inteligência Artificial – Fundamentos e Histórico da Inteligência Artificial (IA), CETAM, Manaus, AM, 2023.

ULTRALYTICS. YOLOv8: State-of-the-Art Object Detection. Disponível em: <https://docs.ultralytics.com/yolov8/>. Acesso em: 25 jul. 2024.)

## AGRADECIMENTOS

Este trabalho foi desenvolvido com o apoio do Governo do Estado do Amazonas por meio Fundação de Amparo à Pesquisas do Estado do Amazonas, com a concessão de bolsa de estudo.