

Infraestrutura para sistema de Apoio à Decisão em Hangares de Manutenção com Visão Computacional

Rafaella Carmo dos Santos Tonaco¹, Henrique Costa Marques²

¹Centro de Estudos e Projetos de Engenharia da Aeronáutica (CEPE), SP, Brasil

²Instituto Tecnológico de Aeronáutica (ITA), São José dos Campos/SP, Brasil

Resumo—Este trabalho propõe uma arquitetura computacional para apoio à decisão em hangares de manutenção aeronáutica, integrando visão computacional como uma infraestrutura para identificação de tarefas críticas e emissão de alertas. Foi desenvolvida uma simulação de hangar tridimensional no Unity, com agentes animados e câmeras virtuais, empregando a arquitetura YOLOv8 com *OpenCV* para detecção de objetos e atividades. A partir dos dados visuais, o sistema realiza inferência contextual de risco, permitindo maior consciência situacional e prevenção de falhas operacionais. Os testes com modelos personalizados atingiram precisão média de 92,8% e mostraram viabilidade da aplicação em tempo quase real. O painel gerencial desenvolvido exibe indicador de alerta, sinalizando situações críticas pré-determinadas, confirmando a viabilidade de execução de tal sistema.

Palavras-Chave—Visão Computacional, manutenção aeronáutica, tomada de decisão.

I. INTRODUÇÃO

A criticidade inerente à atividade de manutenção aeronáutica para a segurança das operações de voo resulta em uma exigência de alta precisão técnica, um controle rigoroso do espaço de trabalho e uma alta coordenação dos mecânicos e engenheiros envolvidos nessa atividade. Em especial, os hangares de manutenção concentram uma variedade de tarefas simultâneas envolvendo operadores, ferramentas, veículos motorizados e procedimentos específicos e pré-determinados, porém sem previsibilidade e em ordenamento (hangares de manutenção tendem a estar em conformações distintas em espaços temporais distintos). Somando-se a isso, as tarefas citadas, quando mal executadas, representam risco elevado à integridade da aeronave e à saúde dos trabalhadores envolvidos nas atividades de manutenção, comprometendo diretamente a operacionalidade aérea, uma vez que constituem um dos maiores custos da companhia [1].

Segundo Tsagkas, cada hora de voo exige cerca de 12 horas de manutenção, e as falhas provenientes de atividades mantenedoras estão relacionadas a mais de 15% dos acidentes aeronáuticos. Não só isso: falhas ligadas à manutenção são a segunda maior causa de acidentes fatais em aeronaves. No contexto brasileiro, o cenário é semelhante: de acordo com o painel SIPAER, que trata das ocorrências aeronáuticas na Aviação Civil Brasileira, mais de 10% dos acidentes têm causa atribuída a falhas de manutenção [1].

R. C. S. Tonaco, rafaella.tonaco@ita.br; H. C. Marques, henrique.marques@ita.br.



Fig. 1. Painel SIPAER, contendo as estatísticas dos acidentes aeronáuticos brasileiros dos últimos 10 anos.

Diante do cenário exposto, tornam-se necessárias soluções capazes de aumentar a consciência situacional dentro de tais ambientes. Mais do que isso: nota-se uma urgência em adequar os hangares de manutenção para o recebimento de tecnologias e processos capazes de melhorar a gestão de risco em seu interior, otimizando processos e diminuindo erros.

Diante do advento de novas tecnologias relacionadas ao monitoramento de ambientes industriais, tornou-se possível o rastreamento de objetos, ações e tarefas e a análise de contexto em tempo próximo ao tempo real. Entretanto, no contexto aeronáutico o uso dessas tecnologias é limitado, principalmente no que tange à identificação de tarefas específicas da manutenção e à sua associação com riscos operacionais potenciais. Isso se deve à existência de tarefas específicas e objetos específicos, aliado à dinamicidade dos hangares. Destaca-se, ainda, que iniciativas como os Hangares 4.0 (*Smart Hangars*) avançam no uso de sensores e de gêmeos digitais, porém, ainda carecem de mecanismos interpretativos capazes de transformar dados brutos em decisões acionáveis.

Dessa forma, o presente trabalho propõe uma simulação de um hangar de manutenção no *Unity*, contendo algoritmos de detecção baseados no YOLOv8, um dos modelos YOLO para detecção de objetos em tempo real, de forma a realizar o mapeamento de ações, objetos e tarefas e, em sequência, realiza a interpretação de tais tarefas com o apoio de heurísticas computacionais para a inferência de risco, emissão de alertas e apoio à tomada de decisão. Propõe-se uma arquitetura que integra as etapas de percepção, interpretação e ação, estabelecendo uma ferramenta que seja capaz de monitorar o hangar de maneira automatizada, informando padrões operacionais e riscos potenciais.

O escopo do modelo consiste na simulação de um hangar de manutenção aeronáutica tridimensional no *Unity*, com agentes, objetos e rotinas modeladas conforme rotinas de manutenção, com a inclusão de câmeras virtuais fixas e móveis, estrategicamente posicionadas para capturar imagens do ambiente. Utiliza-se o YOLOv8 para reconhecimento de técnicos em diferentes posturas e atividades, além de elementos específicos, tais como: carrinhos, escadas, guindastes e tanques de combustível. Por fim, objetiva-se a geração de

alertas com base nas inferências do sistema e exibição de painel gerencial.

O modelo proposto pretende ser uma análise de viabilidade de sistema, apresentando uma metodologia e um fluxograma capaz de ser replicado em um ambiente real. Dessa forma, foi modelado em um ambiente virtual que, embora permita o controle total das variáveis e reprodutibilidade dos testes, não incorpora a variabilidade, ruídos e imprevisibilidades típicas de ambientes reais. Além disso, sombras, reflexos, oclusões e condições adversas de iluminação, comuns em hangares reais, embora tenham sido usados no treinamento do modelo, não são capazes de ser completamente simulados. Ainda, os datasets utilizados foram gerados manualmente e anotados por meio de uma simulação, tendo a quantidade de imagens e diversidade de contextos restritas.

A definição de risco é feita por meio de regras pré-determinadas, reduzindo sua capacidade de adaptação diante de cenários imprevistos. Por fim, a análise aqui abordada se concentra exclusivamente em riscos operacionais que são visíveis por câmeras.

Para cumprir ao proposto neste artigo, faz-se a seguinte divisão: a seção 1, aqui descrita, apresenta a motivação, o contexto de estudo, o problema de pesquisa e a proposta de solução. Divide-se a seção 2 em duas partes: a primeira apresenta a fundamentação teórica necessária para a compreensão do trabalho, de forma a embasar a metodologia. A segunda parte consiste na revisão da literatura, objetivando-se a análise de como outros autores abordam problemas semelhantes. A seção 3 descreve a metodologia do trabalho e as ferramentas utilizadas e apresenta a sequência de ações necessárias para a replicação do trabalho. A seção 4 apresenta os resultados obtidos e a discussão e, por fim, a seção 5 mostra as considerações finais do estudo.

II. FUNDAMENTAÇÃO TEÓRICA E REVISÃO DA LITERATURA

A. Fundamentação teórica

1) *Hangares de manutenção*: Os hangares de manutenção aeronáutica são locais que abrigam aeronaves durante períodos de permanência e de manutenção, sendo classificados da seguinte forma: hangares privados (são propriedades privadas que acomodam aeronaves particulares), hangares de aviação geral (responsáveis por acomodar aviões de menor porte, geralmente clubes de aviação e aeroportos de baixa demanda), hangares militares (abrigam as aeronaves de defesa) e hangares destinados às linhas aéreas.

As atividades realizadas em um hangar dependem do escopo da manutenção (preventiva, corretiva ou programada) e demandam a atuação simultânea de diferentes operadores. Além disso, a rotina em um hangar envolve uma alta circulação de pessoas, veículos e aeronaves, aumentando a complexidade do ambiente. Por fim, a dinâmica operacional dentro de um hangar gera desafios à supervisão e ao gerenciamento de riscos, principalmente devido à simultaneidade das tarefas.

2) *Indústria 4.0 e Operador 4.0*: A Indústria 4.0, também conhecida como manufatura inteligente, representa a quarta revolução industrial, sucedendo marcos anteriores como a mecanização (Indústria 1.0), a produção em massa (Indústria 2.0) e a automação digital (Indústria 3.0) [2]. Essa nova fase é

caracterizada pela integração entre sistemas físicos e digitais, inteligência artificial, *Big data*, internet das coisas (IoT) e robótica avançada, com foco na tomada de decisão autônoma e na otimização adaptativa dos processos. Nesse cenário, emerge o conceito de Operador 4.0, um trabalhador cognitivo que atua de forma colaborativa com sistemas inteligentes, utilizando tecnologias como realidade aumentada, gêmeos digitais, sensores fisiológicos e inteligência artificial para aumentar sua capacidade de percepção, resposta e análise [3]. Esse operador pode ser classificado conforme sua interação com a tecnologia: operador de superforça (com exoesqueletos) [4], aumentado (com realidade aumentada) [2], virtual (com realidade virtual e gêmeos digitais) [5], de bem-estar amplificado (com sensores biométricos) [6], inteligente (interagindo com IA) [7], colaborativo (com robôs) [8], social (conectado em redes móveis) [9] e analítico (com uso de *Big Data*) [10].

Diferentemente da concepção reducionista que associa a Indústria 4.0 à substituição de pessoas por máquinas, o modelo contemporâneo reconhece o ser humano como núcleo do sistema produtivo, sendo a tecnologia um suporte para sua atuação mais eficaz, segura e informada. Nesse contexto, surge o conceito de hangar 4.0, entendido como um ambiente de manutenção aeronáutica que incorpora os mesmos princípios da Indústria 4.0 — automação inteligente, integração entre humano e máquina e análise preditiva — aplicados ao contexto técnico-operacional da aviação. Os técnicos que atuam nesses hangares, portanto, podem ser caracterizados como operadores cognitivos, exigindo não apenas habilidades técnicas tradicionais, mas também competências digitais, capacidade analítica e interação fluida com sistemas de apoio à decisão em tempo real.

3) *Visão computacional*: Define-se visão computacional como uma subárea da inteligência artificial capaz de interpretar dados por meio de imagens, possibilitando que máquinas interpretem cenários visuais. Para tal finalidade, as imagens são tratadas como matrizes discretas de valores (geralmente com pixels em escala de cinza ou RGB), sobre as quais são aplicadas operações computacionais e matemáticas. Os principais campos da visão computacional são: segmentação de imagem, rastreamento de movimento, reconhecimento de objetos e reconstrução tridimensional.

Atualmente, o estado da arte em detecção de objetos é a arquitetura YOLO (You Only Look Once), capaz de realizar a identificação e a classificação de múltiplos elementos visuais em tempo quase real. No modelo YOLOv8, usado ao longo deste trabalho de pesquisa, são incorporadas funcionalidades avançadas, como segmentação semântica, ampliando seu escopo de aplicação. No contexto industrial, essa arquitetura tem se destacado em sistemas de inspeção visual automatizada, contribuindo para a redução de falhas operacionais, o aumento da eficiência nas linhas de produção e a melhoria da confiabilidade dos processos.

Para a detecção de objetos, as redes neurais convolucionais (CNNs) são destacadas como uma abordagem moderna, sendo capaz de fazer a identificação automática de padrões visuais relevantes em imagens. Evoluções dessas arquiteturas originaram algoritmos tais como o R-CNN, Fast R-CNN e SDD, que aprimoraram a precisão e a velocidade da detecção por meio de estratégias como regiões de interesse, redes de proposta de regiões (RPNs) e multiescalas [11].

Tais algoritmos podem ser complementados com ferramen-

tas como o OpenCV e o Barracuda. No presente trabalho, essa integração é explorada para o reconhecimento de tarefas e objetos em um hangar simulado, formando a base para a inferência de riscos e o apoio à decisão.

4) *Unity*: A *Unity Engine* é uma plataforma de desenvolvimento 3D em tempo real amplamente utilizada na criação de jogos, simulações e aplicações interativas. Ela oferece uma interface gráfica intuitiva, o *Unity Editor*, além de suporte à programação por meio de scripts em C#, permitindo desde a construção visual até o controle avançado da lógica. Os projetos na *Unity* são organizados em *Assets*, *Scenes*, *GameObjects* e *Components*, que podem incluir câmeras (*Cameras*), malhas (*Meshes*) e corpos rígidos (*Rigid Bodies*) [12].

Devido à sua flexibilidade, a engine é particularmente adequada para a criação de ambientes simulados e gêmeos digitais, viabilizando o mapeamento de tarefas, a modelagem de interações físicas e a geração de dados sintéticos com controle total das variáveis. Essas características a tornam uma ferramenta estratégica para aplicações de visão computacional em contextos complexos.

5) *Apoio à Tomada de Decisão com Base em Visão Computacional*: Um dos principais desafios no desenvolvimento de sistemas inteligentes autônomos é a conversão de dados coletados por meio de câmera em interpretações e decisões com base em inferências confiáveis. Os modelos clássicos de apoio à tomada de decisão se baseiam em abortagens de sistemas especialistas, lógica *fuzzy*, redes bayesianas e algoritmos supervisionados, como árvores de decisão e máquinas de vetores de suporte [13]. Os modelos citados fazem a utilização de variáveis observadas para a inferência de estados latentes e sugestão de medidas corretivas.

A definição da arquitetura de tomada de decisão depende do volume de dados disponíveis, da necessidade de explicabilidade e da criticidade de domínio. Neste trabalho, a combinação de regras pré-definidas com a lógica *fuzzy* se apresenta como uma abordagem adequada, permitindo lidar com variações de contexto, incertezas na detecção e combinações ambíguas entre agentes e objetos. Pode-se, por exemplo, inferir o risco elevado por meio da presença simultânea entre um técnico e uma escada, embora não se tenha certeza absoluta da tarefa em execução. Faz-se o uso de pesos *fuzzy* e funções de pertinência para graduar o risco, ao invés de fazer inferências de forma binária.

Apesar da existência de estudos capazes de integrar percepção visual com heurísticas de inferência, poucos fazem uso de arquiteturas completas capazes de conectar captura de imagens, interpretação semântica, modelagem de risco e emissão de alertas. Dessa forma, este trabalho propõe a construção de um sistema unificados que parte de dados visuais (captura e identificações de dados de imageamento por meio do YOLOv8) para alimentar uma camada intermediária de interpretação contextual baseada em regras e lógica *fuzzy*, possibilitando a tomada de decisão e geração de alertas em uma interface gerencial.

B. Revisão da Literatura

Segundo Qiu et al. (2017) [14], mundos virtuais realistas são úteis para pesquisas (especialmente de visão computacional), podendo ser usados em muitas tarefas que o mundo real não consegue realizar (possibilitando o teste com

parâmetros difíceis, realizando o teste em situações de riscos e possibilitando que várias situações sejam testadas em um curto espaço de tempo). Osigbesan et al. (2022) [15] testa câmeras e visão computacional para fazer a detecção de quedas em hangares de manutenção, por meio da utilização de uma fusão não intrusiva de gravações de vídeo e análise de movimento com visão computacional para a detecção de quedas. O autor propõe o uso de câmeras com lente grande angular instaladas na posição diagonal no teto do hangar para a captação de dados visuais, fazendo a utilização de um modelo de detecção de quedas permitindo o treinamento em lote e compreensão da distribuição temporal dos dados de entrada. Destaca-se que o autor sugere um mínimo de 3 câmeras para lidar razoavelmente com oclusões (porém ressalta que esse número pode ser aumentado dependendo das configurações do hangar).

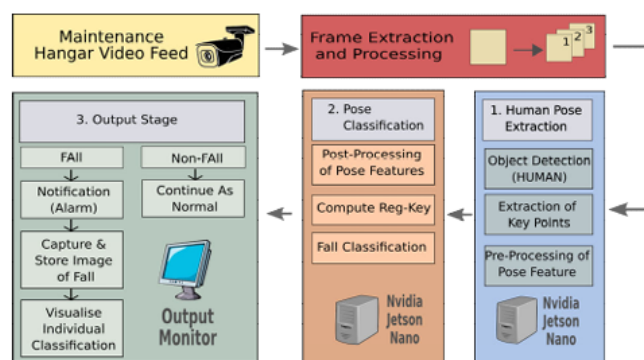


Fig. 2. Proposta da arquitetura do sistema de detecção de quedas, retirado de [15]

Estudos recentes demonstram o potencial da visão computacional para aplicação em inspeções técnicas e manutenção aeronáutica. Romeral [16] analisa tendências emergentes como detecção de objetos e segmentação de imagens, apresentando exemplos práticos adaptáveis a inspeções em hangares, com ganhos em precisão e redução nos tempos de manutenção. Xu [17] abordou o uso dessas técnicas no setor da construção, com foco em falhas estruturais, sugerindo que abordagens similares podem ser aplicadas à manutenção de aeronaves em tempo real. Complementarmente, Zhao [18] destacou como o processamento de imagens pode ser adaptado ao contexto aeronáutico, com aumento na confiabilidade das detecções automatizadas. Além disso, tecnologias como realidade aumentada e virtual têm sido empregadas com sucesso em tarefas de manutenção. Segundo Safi et al. [19], a realidade aumentada melhora a precisão em procedimentos complexos e reduz custos operacionais, enquanto De Sá et al. [20] demonstraram ganhos de até 40% na eficiência de processos com o uso de realidade virtual, especialmente na simulação de cenários de risco. Em todos esses casos, destaca-se que a visão computacional constitui a base essencial para o funcionamento e a integração dessas tecnologias avançadas no ambiente dos hangares.

O autor Wang et al. [21] defende que a lógica de decisão baseada em regras usando como parâmetro *outputs* gerados por algoritmos de visão computacional resulta em um menor tempo de inatividade e aumento da segurança quando aplicados na indústria manufatureira. Em um estudo semelhante,

Jarrahi [22] explora a colaboração entre inteligência artificial e heurísticas humanas como um fator preponderante em decisões mais explicáveis e adaptáveis em sistemas complexos. Ainda em ambientes de manutenção industrial, Zhang et al. [23] apresenta um modelo híbrido combinando algoritmos fuzzy com redes neurais para avaliação de falhas, mostrando robustez frente à variabilidade dos dados operacionais. Por fim, Chen et al. [24] exploram sistemas especialistas para análise de múltiplos fatores de risco em tempo real, por meio da utilização de dados sensoriais integrados, inclusive visuais, para a emissão de alertas e orientação de intervenções corretivas.

Os trabalhos citados confirmam a viabilidade de integração de arquitetura em hangares de manutenção da forma que aqui será desenvolvido o estudo.

III. METODOLOGIA

A. Materiais e Métodos

Este trabalho propõe uma arquitetura computacional capaz de identificar tarefas, agentes e objetos em um hangar de manutenção aeronáutica por meio de visão computacional aplicada a um ambiente virtual simulado. O sistema foi desenvolvido em etapas, englobando modelagem tridimensional do ambiente, integração de sensores virtuais (câmeras), aplicação de algoritmos de detecção de objetos, treinamento de classes personalizadas, controle de agentes autônomos e construção de uma base preliminar de apoio à decisão com inferência de risco.

1) *Modelagem do Ambiente de Hangar*: Utilizou-se o motor gráfico *Unity*, integrando um cenário realista de hangar obtido a partir da *Unreal Engine* com o pacote *Aircraft Maintenance Hangar*. A conversão foi realizada por meio do plugin *Exporter for Unreal to Unity 2024*, que permite a migração de malhas, animações e texturas entre os ambientes. O hangar virtual contempla uma aeronave, operadores, carrinhos de ferramentas, plataformas elevatórias, guindastes, rebocadores e outros elementos típicos do ambiente de manutenção.

Foram testadas diferentes configurações de *render pipeline* (Built-In, URP e HDRP) com o objetivo de otimizar o desempenho da simulação em um notebook gamer (GTX 1660Ti, Intel i7, 16 GB RAM). A versão final utiliza *Universal Render Pipeline* para balancear qualidade visual e tempo de resposta.

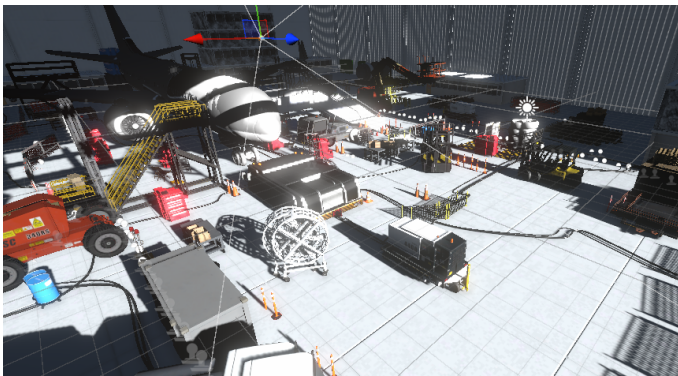


Fig. 3. Visão do hangar obtida pelo Unity, configurações do Unity

2) *Controle e Animação de Agentes*: Para simular operadores de manutenção, foram utilizados agentes do pacote *Convai*, disponível no Unity Assets Store, capazes de executar

movimentos, interações contextuais e responder a comandos. Cada agente teve sua trajetória planejada com base em pontos de controle no ambiente (*waypoints*) e deslocamento programado via *NavMesh* (uma malha de navegação típica para cenas do Unity para permitir que agentes interajam com o ambiente e saibam por onde podem trafegar durante a execução do cenário). Ações específicas como pintura de componentes ou movimentação de cargas foram programadas utilizando os softwares *IClone* e *Character Creator* (ambos pertencentes à Reallusion), com exportação das animações no formato *.fbx*.

Após a importação para o *Unity*, os movimentos foram vinculados aos personagens por meio de um *Animator Controller*, possibilitando simulação realista de rotinas operacionais típicas de um hangar.

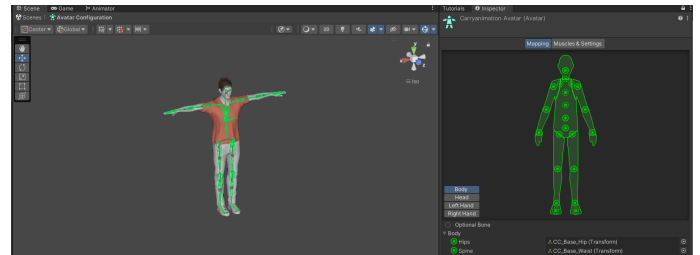


Fig. 4. Configuração de avatar

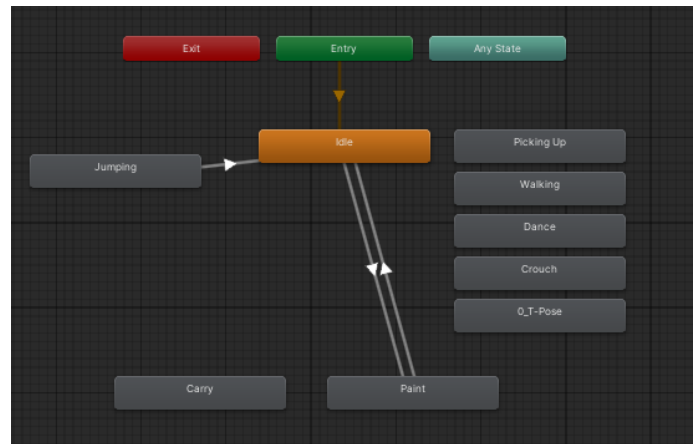


Fig. 5. Exemplificação de *Animator Controller* implementado com a função “pintar”.

3) *Configuração de Câmeras Virtuais*: A captura visual do ambiente foi realizada com câmeras virtuais fixas posicionadas estrategicamente para cobertura total da área de interesse. Foram definidos critérios para evitar oclusões, garantir visibilidade dos objetos críticos (como tanques de combustível e empilhadeiras) e minimizar sobreposição entre campos de visão. Três câmeras foram consideradas suficientes para essa primeira etapa, balanceando qualidade da percepção e carga computacional para o processamento em tempo real.

4) *Integração do Sistema de Visão Computacional*: O processamento das imagens capturadas foi feito com o pacote *OpenCV for Unity*, que integra a biblioteca *OpenCV* diretamente ao ambiente de simulação. A conversão de espaço de cor e aplicação de filtros foram utilizadas para otimizar a segmentação de bordas e contraste de objetos. A biblioteca foi essencial para pré-processamento dos frames antes da inferência pelos modelos de detecção.

5) *Deteção de Objetos com YOLOv8*: A arquitetura YOLOv8, por suas capacidades de segmentação e velocidade, foi integrada por meio do pacote *YOLOv8WithOpenCVForUnityExample*, permitindo inferência local a partir de modelos treinados no formato .onnx. Inicialmente, foram usados modelos baseados no dataset COCO, que identificam classes genéricas como pessoas e aeronaves.

No entanto, diversos objetos típicos de hangares não estavam disponíveis no dataset original, exigindo o treinamento de um modelo personalizado. Foram capturadas imagens diretamente do ambiente simulado e anotadas manualmente na plataforma *Roboflow*, com cerca de 130 imagens anotadas para classes como empilhadeiras, guindastes, rebocadores e carrinhos de ferramentas.

O treinamento foi realizado em *Google Colab*, gerando pesos no formato .pt, posteriormente convertidos para .onnx. As classes foram especificadas em arquivos .names. O modelo final foi capaz de detectar todos os elementos de interesse com acurácia satisfatória e baixa latência.

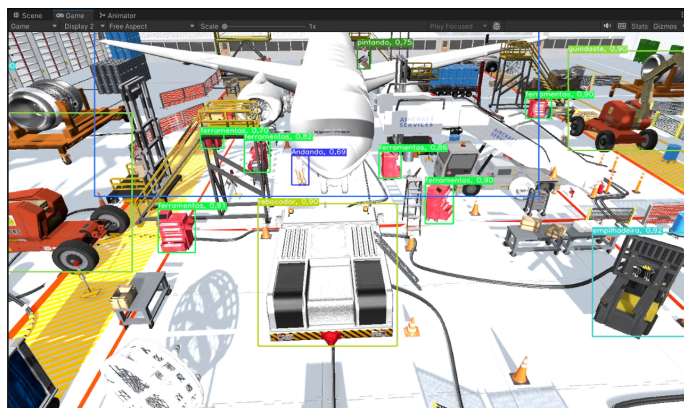


Fig. 6. Execução do modelo treinado no Unity.

6) *Inferência de Tarefas e Riscos*: A partir das detecções de objetos e agentes, o sistema constrói um mapeamento da cena em tempo real. O objetivo é reconhecer padrões de coocorrência para inferir tarefas em andamento e riscos associados. O alerta de risco encontra-se em desenvolvimento, utiliza lógica fuzzy e regras pré-definidas para representar relações causais entre configurações da cena e níveis de risco.

IV. ESTUDO DE CASO, RESULTADOS E DISCUSSÕES

O estudo de caso foi conduzido com o objetivo de avaliar a viabilidade da utilização de visão computacional em ambientes de manutenção aeronáutica simulados, com vistas ao mapeamento de tarefas, identificação de objetos críticos e apoio à tomada de decisão em tempo quase real. O cenário adotado foi um hangar tridimensional desenvolvido originalmente no Unreal Engine, posteriormente exportado para o Unity com uso do pacote *Exporter for Unreal to Unity*, disponível no Unity Assets Store. O ambiente inclui aeronaves, plataformas elevatórias, veículos de abastecimento, operadores e demais elementos típicos de hangares operacionais.

Para viabilizar a movimentação lógica de operadores no cenário, foi utilizada a biblioteca *Convai* integrada ao Unity, com suporte à navegação via *NavMeshSurface*. Cinco agentes foram inseridos na cena, com rotas distintas e aleatoriedade parcial, além de dois agentes animados executando ações

específicas: pintura e carregamento de peso. Os modelos foram animados externamente via *iClone* e *Character Creator*, conforme descrito na metodologia.

Inicialmente, a detecção foi limitada às classes disponíveis no dataset COCO, com sucesso na identificação de pessoas e aeronaves. Para ampliar a abrangência, foi realizado um treinamento personalizado com imagens extraídas do ambiente simulado. A base inicial consistiu em 288 imagens capturadas a partir de vídeos das três câmeras, segmentadas manualmente e anotadas na plataforma *Roboflow*. Após aumentos por rotação, brilho e espelhamento, o banco foi expandido para 1.050 imagens.

A. Classes Treinadas e Métricas Obtidas

O modelo foi treinado para identificar 10 classes, sendo 7 objetos e 3 tarefas. A Tabela I apresenta os resultados obtidos na validação (com 57 imagens) para cada classe.

Classe	Precisão (P)	mAP50	mAP50-95
Aeronave	0.956	0.995	0.811
Empilhadeira	1.000	0.995	0.933
Ferramentas	1.000	0.995	0.911
Guindaste	1.000	0.995	0.961
Combustível	1.000	0.995	0.911
Gerador	1.000	0.995	0.973
Rebocador	1.000	0.995	0.949
Pintando	0.947	0.963	0.815
Carregando peso	0.880	0.952	0.431
Andando	0.455	0.647	0.226
Média geral	0.928	0.953	0.807

TABELA I

RESULTADOS DE VALIDAÇÃO POR CLASSE PARA MODELO TREINADO COM YOLOv8

O mAP mede a precisão média dos modelos de detecção de objetos. O mAP50 considera corretas as predições com sobreposição mínima de 50% entre a caixa predita e a real, sendo mais permissivo. Já o mAP50-95 avalia a média em limiares de 0,50 a 0,95 de IoU (*Intersection over Union*), tornando-se mais rigoroso e representando melhor a qualidade geral da detecção.

Como se observa, os objetos apresentaram alto desempenho ($mAP50 \approx 0.99$), enquanto tarefas complexas como “Carregando peso” e “Andando” tiveram acurácias inferiores, especialmente no mAP50-95. Isso pode ser atribuído à menor variabilidade nas poses e à semelhança entre ações humanas nas imagens, o que exige bases maiores e dados temporais para generalização mais robusta.

B. Comparação com Roboflow

Para fins de comparação, um segundo modelo foi treinado na própria plataforma *Roboflow*, utilizando o algoritmo “Roboflow 3.0 Object Detection (Accurate)”. Os resultados gerais foram: mAP = 93,1%, precisão = 93,5% e recall = 93,5%. Apesar de levemente inferiores ao modelo do Colab, os resultados reforçam a consistência da base de dados criada. A matriz de confusão e as curvas de mAP mostraram coerência na identificação da maioria das classes, com menor desempenho também observado nas tarefas dinâmicas.

C. Aplicação Prática na Simulação

A integração do modelo ao ambiente Unity permitiu a visualização das detecções em tempo quase real. Foram utilizados quatro painéis com câmeras independentes processando entradas distintas. O sistema conseguiu identificar objetos críticos, operadores e comportamentos com razoável fluidez, mesmo com as limitações do hardware (GTX 1660Ti). O tempo de inferência médio foi de 25,6 ms por imagem, com pós-processamento adicional de 14,7 ms, totalizando cerca de 40 ms por quadro, ou 25 quadros por segundo.

Foi também implementado um sistema de carro de ferramentas autônomo, movendo-se pelo hangar com uso de NavMesh e detecção de obstáculos. O agente foi monitorado pelo YOLOv8 e demonstrou comportamento coerente com o esperado, reforçando a aplicabilidade do sistema para gerenciamento de elementos móveis em hangares reais.

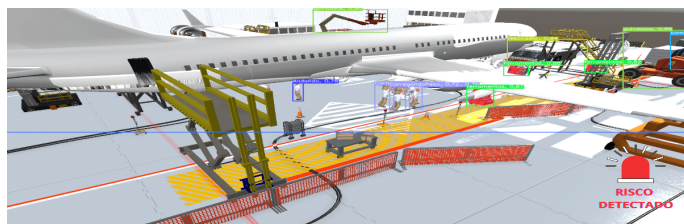


Fig. 7. Sistema de alerta para detecção de carregamento de peso por tempo superior a 30 segundos

V. CONCLUSÃO

O uso de visão computacional para apoio à decisão em hangares de manutenção aeronáutica mostrou-se viável em ambiente simulado e controlado. O modelo treinado com dados sintéticos atingiu níveis de acurácia compatíveis com aplicações industriais, sendo capaz de identificar tanto objetos quanto tarefas relevantes. Embora existam desafios para operação em tempo real com alta complexidade, o sistema demonstra maturidade para ser utilizado em ambientes semi-controlados e servir de base para aplicações futuras mais avançadas, como monitoramento preditivo, sistemas de alerta em tempo real e análise de produtividade.

O treinamento com imagens sintéticas permitiu a criação de um sistema funcional, com métricas que superam 90% de acurácia em classes críticas. Como estudos futuros, sugere-se a expansão do modelo para a abrangência de maior quantidade de tarefas detectadas e aumento das métricas indicadoras de riscos, para que, por fim, seja feita a aplicação em ambientes de manutenção reais.

REFERÊNCIAS

- [1] V. Tsagkas, V. Tsagkas, D. Nathanael, D. Nathanael, N. Marmaras, and N. Marmaras, "A pragmatic mapping of factors behind deviating acts in aircraft maintenance," *Reliability Engineering System Safety*, 2014.
- [2] Jeevan S. Devagiri, P. Sidike, S. Paheding, Quamar Niyaz, Xiaoli Yang, S. Smith, and Samantha Smith, "Augmented reality and artificial intelligence in industry," *Expert Systems With Applications*, pp. 118 002–118 002, Jul. 2022, mAG ID: 4283802212 S2ID: 4a8e83e01d83f9f3c796fda7d07588995ed65e19.
- [3] D. Diaz, D. R. Díaz, J. Stahre, J. Stahre, T. Wuest, O. Noran, O. S. Noran, P. Bernus, P. Bernus, Åsa Fast-Berglund, Åsa Fast-Berglund, D. Gorecky, and D. Gorecky, "Towards an operator 4.0 typology: A human-centric perspective on the fourth industrial revolution technologies," *Computers in Entertainment*, 2016.
- [4] N. Sylla, V. Bonnet, F. Colledani, and P. Fraisse, "Ergonomic contribution of able exoskeleton in automotive industry," *International Journal of Industrial Ergonomics*, 2014.

- [5] K. Gkagiannis and A. Komninos, "Spatial multimodal alert cues for virtual reality control environments in industry 4.0," *International Conference of the ACM Greek SIGCHI Chapter*, 2023.
- [6] S. Sun, X. Zheng, B. Gong, J. Paredes, and J. Ordieres-Meré, "Healthy operator 4.0: A human cyber-physical system architecture for smart workplaces," *Sensors*, 2020.
- [7] K. Myers, P. M. Berry, J. Blythe, K. Conley, M. Gervasio, D. L. McGuinness, D. N. Morley, A. Pfeffer, M. E. Pollack, and M. Tambe, "An intelligent personal assistant for task and time management," *Ai Magazine*, 2007.
- [8] N. E. Nilsson and J. Jacobsson, "Operator - robot collaboration - creation and evaluation of a collaborative operator - robot workstation," *null*, 2017.
- [9] T. Wuest, K. Hribernik, and K. Thoben, "Can a product have a facebook? a new perspective on product avatars in product lifecycle management," *Product Lifecycle Management*, 2012.
- [10] T. Wuest, D. Weimer, C. Irgens, and K. Thoben, "Machine learning in manufacturing: advantages, challenges, and applications," *Production and Manufacturing Research: An Open Access Journal*, 2016.
- [11] S. Agarwal, S. Agarwal, S. Agarwal, J. O. du Terrail, J. O. D. Terrail, F. Jurie, and F. Jurie, "Recent advances in object detection in the age of deep convolutional neural networks," *arXiv: Computer Vision and Pattern Recognition*, 2018.
- [12] A. Juliani, A. Juliani, V.-P. Berges, V.-P. Berges, E. Vckay, E. Vckay, Y. Gao, Y. Gao, H. Henry, H. Henry, M. Mattar, M. Mattar, D. B. Lange, D. Lange, and D. Lange, "Unity: A general platform for intelligent agents," *arXiv: Learning*, 2018.
- [13] S. jin Rong, W. Guo, and D. qing Zhang, "A study on group decision making problem based on fuzzy reasoning and bayesian networks," *arXiv.org*, 2025.
- [14] W. Qiu, F. Zhong, Y. Zhang, S. Qiao, Z. Xiao, T. S. Kim, and Y. Yu, "Unrealcv: Virtual worlds for computer vision," *ACM Multimedia*, 2017.
- [15] A. Osigbesan, S. Barrat, H. Singh, D. Xia, S. Singh, Y. Xing, W. Guo, and A. Tsourdos, "Vision-based fall detection in aircraft maintenance environment with pose estimation," *International Conference on Multisensor Fusion and Integration for Intelligent Systems*, 2022.
- [16] P. A. D. A. F. Romeral, E. de Senzi Zancul, and P. T. de Souza Nascimento, "Visão computacional na indústria: tendências e exemplos práticos," *Anais do Encontro Nacional de Engenharia de Produção*, 2023.
- [17] S. Xu, S. Xu, D. Wang, J. Wang, X. Wang, X. Wang, W. Shou, and W. Shou, "Computer vision techniques in construction, operation and maintenance phases of civil assets: A critical review," *Proceedings of the 36th International Symposium on Automation and Robotics in Construction (ISARC)*, 2019.
- [18] R. Zhao and Y. Tian, "Application of computer vision and image processing technology in bridge condition monitoring system," *2023 IEEE International Conference on Image Processing and Computer Applications (ICIPCA)*, 2023.
- [19] M. Safi, J. Chung, and P. Pradhan, "Review of augmented reality in aerospace industry," *Aircraft Engineering and Aerospace Technology*, vol. 91, no. 9, pp. 1187–1194, Oct. 2019, mAG ID: 2958645313 S2ID: 6b31b3fde662d85567f7c530ab5b7537f410129f.
- [20] A. G. de Sá and G. Zachmann, "Virtual reality as a tool for verification of assembly and maintenance processes," *Computers & Graphics*, vol. 23, no. 3, pp. 389–403, Jun. 1999, mAG ID: 1982954541 S2ID: 62567b885a9a936bd57a9e753c231f5046ac68d8.
- [21] L. Wang, H. Zhang, and C. Yang, "Intelligent decision-making in manufacturing systems: a review," *Journal of Manufacturing Systems*, vol. 62, pp. 760–776, 2022.
- [22] M. H. Jarrahi, "Artificial intelligence and the future of work: Human-ai symbiosis in organizational decision making," *Business Horizons*, vol. 61, no. 4, pp. 577–586, 2018.
- [23] L. Zhang, Z. He, and Y. Wang, "A hybrid decision-making model for fault diagnosis in industrial systems using fuzzy logic and neural networks," *Expert Systems with Applications*, vol. 168, p. 114281, 2021.
- [24] X. Chen, B. Li, and C. Xu, "Multi-source data fusion for intelligent decision-making in smart logistics systems," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 12, pp. 7642–7651, 2020.