Industrialização, Digitalização, Desempenho

5º Simpósio Brasileiro de Tecnologia da Informação e Comunicação na Construção e 5º Workshop de Tecnologia de Processos e Sistemas Construtivos

[1]AVALIAÇÃO DAS REDES YOLO PARA DETECÇÃO DE DANOS EM COBERTURAS EM TEMPO REAL VISANDO APOIO A GESTÃO DE MANUTENÇÃO

Evaluation of YOLO Networks for Real-Time Roof Damage Detection to Support Maintenance Management

Pedro Afonso Vieira Fernandes Braga
Universidade Federal da Bahia | Salvador, Bahia | pedrofernandes@ufba.com

Alisson Souza Silva
Universidade Federal da Bahia | Salvador, Bahia | so_alisson@hotmail.com

Walisson Santos Oliveira
Universidade Federal da Bahia | Salvador, Bahia | walissonoliveira@ufba.br

Dayana Bastos Costa
Universidade Federal da Bahia | Salvador, Bahia | dayanabcosta@ufba.br

RESUMO

A detecção automática de defeitos em telhados e coberturas utilizando drones enfrenta desafios relacionados à precisão, eficiência computacional e viabilidade de implementação em sistemas embarcados. Nesse sentido, com o avanço das arquiteturas de Aprendizado Profundo, como o YOLO, tornou-se possível realizar inspeções prediais de forma mais rápida e precisa, reduzindo custos e riscos associados a métodos tradicionais. Este trabalho teve como objetivo avaliar o desempenho de diferentes versões do YOLO (v5 a v11) na detecção de defeitos em telhados, comparando métricas de precisão, mAP50 e tempo de inferência para identificar o modelo mais adequado para aplicações em tempo real. Nesse contexto, foram utilizadas 7.210 imagens capturadas por drones, pré-processadas e divididas em conjuntos de treinamento, validação e teste. Seis modelos YOLO foram treinados e avaliados, com ajustes de hiperparâmetros para garantir comparabilidade. O YOLOv9 destacou-se com a maior precisão (66,6%) e mAP50 (53,9%), enquanto o YOLOv5 apresentou o menor tempo de inferência (6,6 ms). O estudo demonstra que o YOLOv8 é o modelo mais viável para implementação dispositivos embarcados, oferecendo um equilíbrio ideal entre desempenho e eficiência computacional para inspeções prediais em tempo real no apoio a gestão de manutenção.

Palavras-chave: YOLO; Aprendizado profundo; Gestão da manutenção; Inspeção automatizada; Inteligência Artificial

ABSTRACT

Automatic detection of roof defects using drones faces challenges related to accuracy, computational efficiency, and feasibility of implementation in embedded systems. In this sense, with the advancement of Deep Learning architectures, such as YOLO, it has become possible to perform building inspections faster and more accurately, reducing costs and risks associated with traditional methods. This work aimed to evaluate the performance of different versions of YOLO (v5 to v11) in detecting roof defects, comparing accuracy, mAP50, and inference time metrics to identify the most suitable model for real-time applications. In this context, 7,210 images captured by drones were used, preprocessed, and divided into training, validation, and test sets. Six YOLO models were trained and evaluated, with hyperparameter adjustments to ensure comparability. YOLOv9 stood out with the highest accuracy (66.6%) and mAP50 (53.9%), while YOLOv5 presented the fastest inference time (6.6 ms). The study demonstrates that YOLOv8 is the most viable model for embedded device deployment, offering an ideal balance between performance and computational efficiency for real-time building inspections in support of maintenance management.

Keywords: YOLO; Deep Learning; Facility Management; Automated Inspection; Artificial Intelligence

1. INTRODUÇÃO

O avanço tecnológico na área de Inteligência Artificial (IA), impulsionado pelo desenvolvimento do Aprendizado de Máquina (AM) e do Aprendizado Profundo (AP), tem transformado diversos campos de atuação (LeCun et al., 2015). Um exemplo notável dessa revolução é a detecção em tempo real, que utiliza técnicas avançadas de aprendizado e processamento de imagens. Aplicações como o reconhecimento automático de padrões visuais têm gerado impactos significativos em áreas como saúde e segurança pública (Khojaste-Sarakhsi et al., 2022; Shehab et al., 2022), gerenciamento de transporte (Garg et al., 2024), monitoramento ambiental (Dave et al., 2024) e gestão de infraestrutura (Jency, 2024; Haleem, 2023). Além

[1]BRAGA, P. A. V. F., SILVA, A. S.; OLIVEIRA, W. S.; COSTA, D. B. Avaliação das Redes YOLO para Detecção de Danos em Coberturas em tempo real visando Apoio à Gestão de Manutenção. In: 5º SIMPÓSIO BRASILEIRO DE TECNOLOGIA DA INFORMAÇÃO E COMUNICAÇÃO NA CONSTRUÇÃO, 4., 2025, Florianópolis. **Anais [...].** Porto Alegre: ANTAC, 2025.

disso, a integração de algoritmos de IA embarcados em dispositivos portáteis associados à Internet das Coisas (*Internet of Things* - IoT) tem viabilizado análises precisas em tempo real, promovendo maior eficiência operacional e redução de custos (Mehta et al., 2024; Jakkani, 2024).

Na construção civil, a integração dessas tecnologias emergentes está remodelando processos tradicionais (Baduge et al., 2022; Silva et al., 2025). Com a crescente complexidade de projetos e a necessidade de maior eficiência, métodos baseados em visão computacional vêm sendo empregados para identificar defeitos em telhados (Staffa et al., 2023), fachadas (Silva et al., 2024), otimização de recursos e predição de custos (Zhang e Zhang, 2024) e, monitoramento de progresso de obras (Chua e Cheah, 2024). A capacidade de realizar análises em tempo real traz vantagens inegáveis, especialmente no contexto de gestão da manutenção e inspeções, sobretudo no uso de drones (Silva et al., 2024) onde o gestor de operações e manutenção dependem de uma tomada de decisão rápida e eficaz, situação na qual essas ferramentas surgem como apoio (Sanzana et al., 2022). Entretanto, desafios ainda persistem, principalmente no que tange à adaptação dessas ferramentas aos ambientes adversos e dinâmicos que caracterizam o setor (Melo e Costa, 2019).

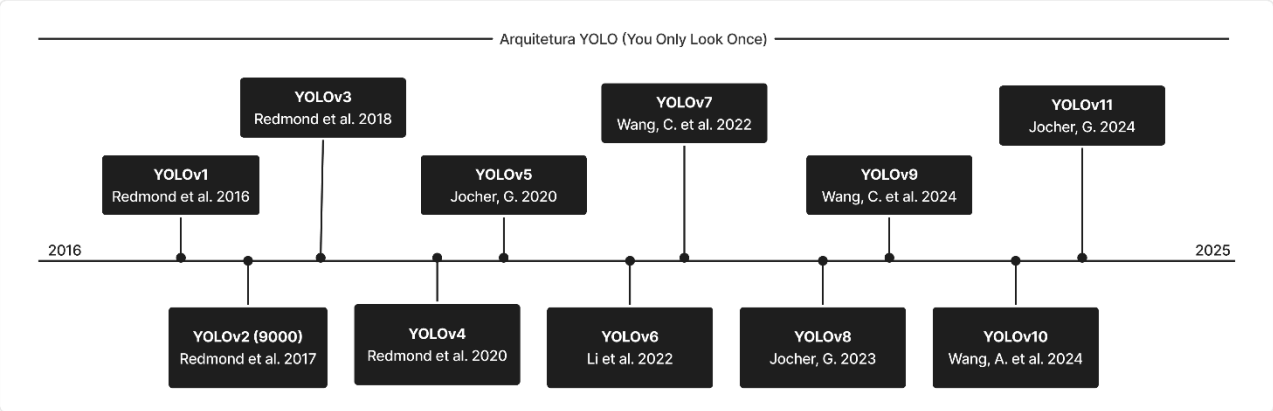
Nesse cenário, a detecção em tempo real é vista como um suporte essencial para a gestão da manutenção de edificações, permitindo intervenções preventivas e corretivas mais ágeis e eficazes (Scaife, 2024). No entanto, a aplicação prática dessas soluções vem sendo abordadas de maneira tímida, apesar do potencial na área de operação e manutenção apesar do avanço da capacidade computacional de sistemas embarcados de baixo custo (AFTAB, 2017). Esses problemas são agravados pela diversidade de padrões de deterioração, no contexto de inspeções de coberturas, como telhas quebradas, fratura em lajes, acúmulo de biofilme, e outros (Silva et al., 2025). Essa diversidade de condições requer modelos robustos e versáteis para diferentes condições de operação. Nesse sentido, a família de algoritmos *You Only Look Once* (YOLO) une velocidade (baixa latência), boa generalização na detecção de objetos, porém, tem dificuldade em localizar com precisão alguns objetos, especialmente os pequenos (Redmon et al. 2016). A YOLO já foi utilizada em aplicações com foco em detecções em tempo real no contexto de tráfego viário (Ćorovic et al., 2018), detecção facial (Yang e Jiachun, 2018).

Com o avanço da computação e da ciência de dados, a YOLO atualmente alcançou sua 11ª versão. No campo das inspeções de telhados e coberturas, até o momento ainda não foi encontrado estudos que comparem essas versões utilizando bases de dados específicas de telhados, dificultando a escolha da versão com o melhor custo-benefício para sistemas embarcados. Portanto, o objetivo deste estudo é avaliar as versões mais recentes da arquitetura YOLO, utilizando uma base de dados própria composta por imagens de telhados e coberturas coletadas por drones em 25 edificações. O foco está nas versões a partir da YOLOv5 até a YOLOv11, pois as versões anteriores à YOLOv5 possuem limitações de resolução (416x416) em comparação às resoluções superiores praticadas nas versões mais recentes (640x640), inviabilizando uma comparação direta. Além disso, a análise dessas versões visa identificar qual delas oferece o melhor custo-benefício para o desenvolvimento de sistemas embarcados e inspeções em tempo real, contribuindo para a geração de informações que apoiem a gestão da manutenção.

2. REVISÃO DA LITERATURA

Na área de visão computacional, a detecção de objetos, conhecida por sua complexidade, passou por avanços notáveis com o surgimento da arquitetura *You Only Look Once* (YOLO) em 2016. Diferentemente das arquiteturas tradicionais de dois estágios para detecção de objetos, o YOLO apresentou uma abordagem unificada, capaz de prever, de forma simultânea, as caixas delimitadoras e as probabilidades de classe (Redmon et al., 2016). Essa inovação atendeu às exigências de processamento em tempo real, marcando um avanço significativo no campo (Koubaa et al., 2021). Segundo Viswanatha (2022), A YOLO se destaca de outras Redes Convolucionais Neurais (RCN), pois, na tarefa de detecção de objetos é mais preciso e rápido do que as demais (Figura 1). Neste trabalho, serão abordados os modelos YOLOv5 a YOLOv11 na configuração S (*small*), considerando sua relevância para aplicações em sistemas embarcados e na construção civil. A configuração S destaca-se pela otimização de parâmetros e operações de *backbone*, reduzindo a complexidade do modelo sem comprometer significativamente a acurácia, o que a torna adequada para cenários com restrições de hardware (Viswanatha, 2022).

Figura 1: Contexto histórico da YOLO

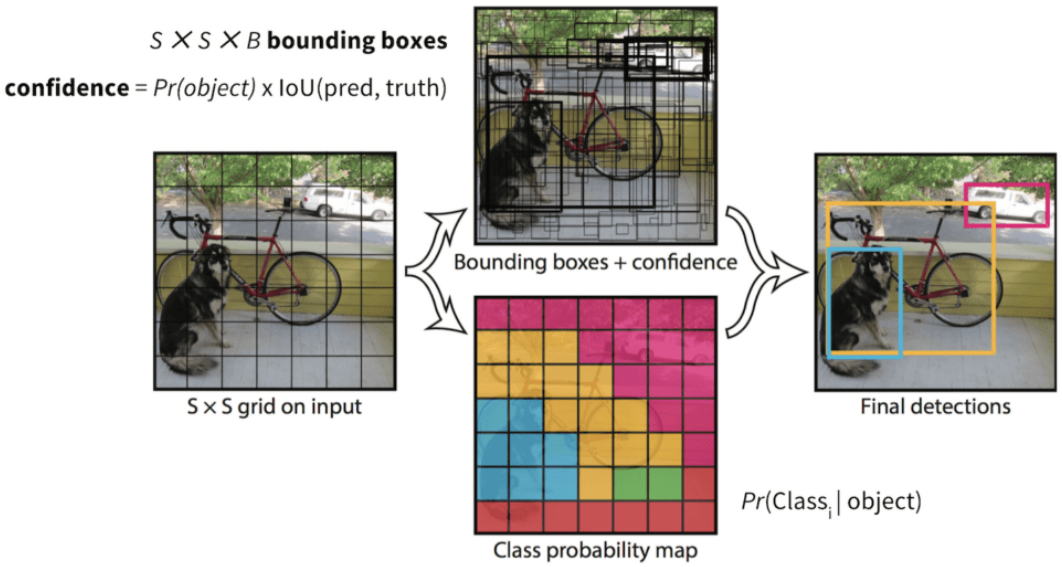


Fonte: Os autores.

Entre os modelos de detectores de estágio único, a YOLOv1 (Redmon et al., 2016) destacou-se por sua robustez e eficiência, sendo um marco no campo da detecção de objetos (Jegham et al., 2024). Este modelo revolucionou a abordagem ao realizar, em uma única análise da imagem completa, a predição simultânea das caixas delimitadoras (bounding boxes) e das probabilidades das classes associadas. Em sequência, a YOLOv2 introduziu melhorias significativas ao ser pré-treinada com imagens de maior resolução (448x448), o que resultou em avanços tanto na precisão quanto na velocidade de detecção, além de inovações na localização das predições de classes (Redmon et al., 2017). Ambas as versões, YOLOv1 e YOLOv2, basearam suas predições em um sistema de grade, como ilustrado na Figura 2.

A partir da terceira versão, houve uma mudança metodológica importante, com a adoção de predições baseadas em caixas delimitadoras, suportadas por um novo framework, o Darknet-53, empregado para extração de características (features). Já a YOLOv4 (Bochkovskiy et al., 2020) introduziu uma nova estrutura conhecida como neck (pescoço), que conecta as características extraídas pelo backbone (espinha dorsal) à head (cabeça) do modelo, mantendo o head da YOLOv3 como referência, melhorando a precisão média em 12%. Essas primeiras iterações da YOLO, no entanto, não foram incluídas na comparação deste estudo devido às limitações impostas pelas resoluções utilizadas (416x416), que poderiam comprometer a avaliação de modelos posteriores, como aqueles que oferecem suporte a resoluções mais altas (640x640).

Figura 2: YOLOv1 bounding boxes



Fonte: Redmond et al. (2016)

Os modelos YOLO analisados nesse trabalho são as versões v5 à v11, em sua distribuição S, a qual oferece melhor compromisso em termos de tempo (latência) de detecção e mAP em *benchmarks* como COCO. Esta distribuição também oferece um balanço equilibrado entre consumo de energia para dispositivos com restrições energéticas, como dispositivos embarcados. A literatura (Han et al., 2015) cita algumas técnicas

de compressão de redes neurais para o uso em dispositivos embarcados. A versão 7 (Wang et al., 2022) não teve um modelo comparável à v6S ou v8S, portanto foi excluída da análise.

A YOLO já vem sendo utilizada em tarefas de detecção em tempo real em diversas indústrias. Em inspeções de fachadas (Jiang et al., 2020) redes neurais foram utilizadas, do tipo *SSD (Single Stage Detector)* embarcado em um drone escalador de paredes, para detecção de rachaduras, apenas. Santos et al. 2023, testou 6 diferentes algoritmos de redes neurais convolucionais (CNNs), no contexto da manutenção de telhados. No entanto, neste trabalho foi avaliada apenas uma classe para comparativo, em um único edifício. Não foram encontrados trabalhos que avaliem a YOLO, no contexto de detecção multiclasse, na inspeção de telhado. Silva et al. 2023 fez uso das redes AlexNet e ResNet no âmbito da detecção multiclasse, mas o fez usando o Microsoft Custom Vision, o que impede a manipulação de hiperparamêtros relevantes e a impossibilidade de *fine tunning*.

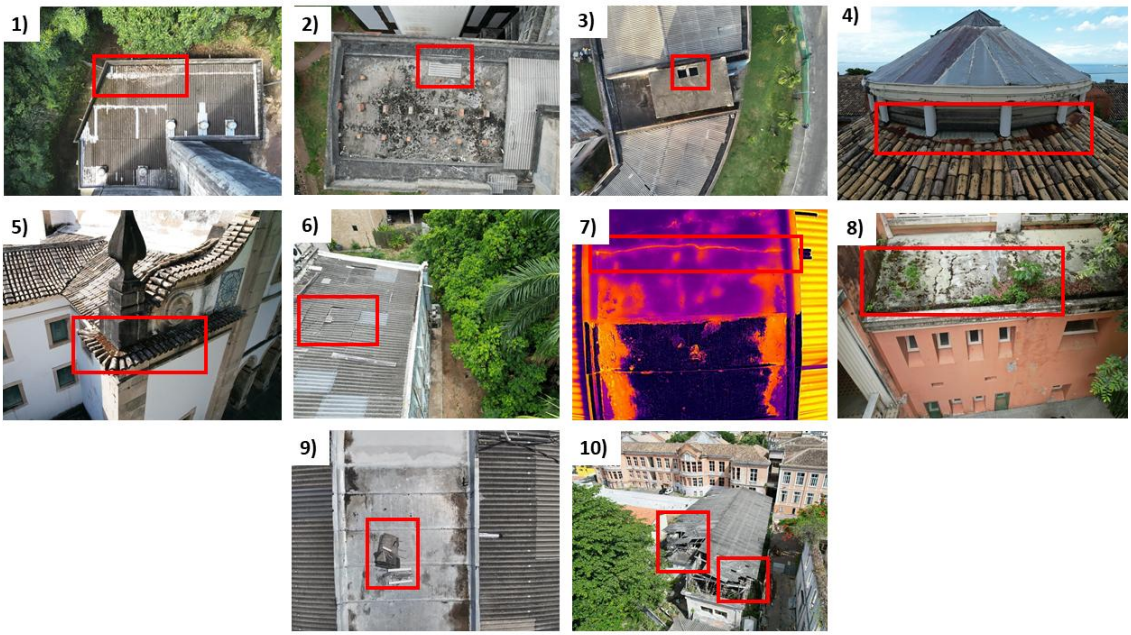
3. MÉTODO DE PESQUISA

Este estudo adotou a estratégia de pesquisa do estudo de caso exploratório, com o objetivo de identificar procedimentos a serem utilizados em pesquisas futuras (Yin, 2018). Assim, este estudo foi desenvolvido nas seguintes etapas: (a) revisão da literatura sobre a arquitetura YOLO; (b) preparação e refinamento da base de dados; (c) desenvolvimento dos modelos YOLO; (d) avaliação e comparação dos modelos.

3.1 Preparação do conjunto de dados

Nesta etapa, foi realizada a organização da base de dados utilizada no estudo. As imagens foram obtidas ao longo de 25 inspeções, abrangendo uma área total superior a 42.000 m² de telhados analisados, resultando na coleta de 7.937 imagens. Para garantir a relevância dos dados, foi conduzido um processo criterioso de refinamento, no qual foram mantidas apenas imagens contendo falhas ou anomalias em telhados e coberturas das edificações inspecionadas. Após essa filtragem, o conjunto final foi composto por 7.210 imagens, disponíveis em dois formatos: RGB e térmico. Além disso, foram definidas 10 categorias de defeitos identificados nos telhados e coberturas (Figura 3), a saber: (1) presença de sujeira; (2) telhas extras; (3) alçapão abertos; (4) oxidação; (5) presença de biofilme; (6) deslocamento de telhas; (7) fissuras na laje; (8) presença de vegetação; (9) resíduos acumulados; e (10) danos às telhas.

Figura 3: Classes presentes do conjunto de dados



Fonte: Os autores.

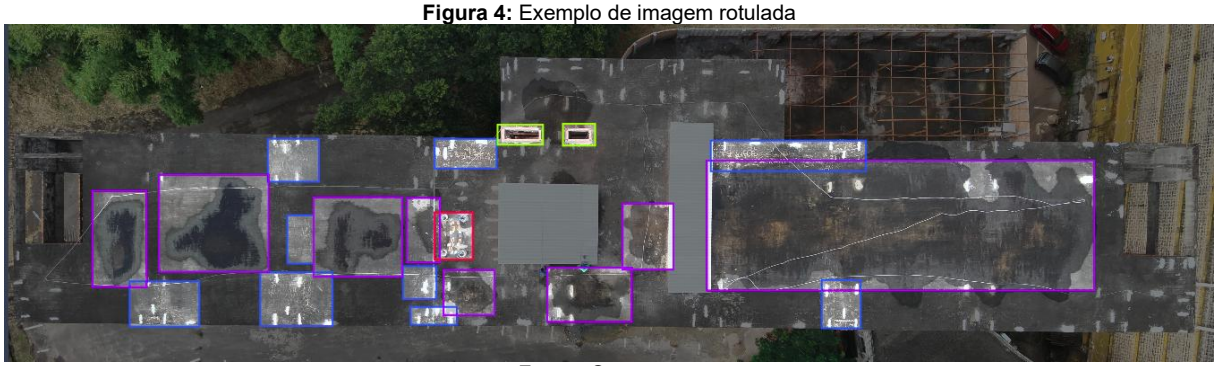
O conjunto de dados utilizada neste estudo foi composta por imagens de telhados e coberturas coletadas por meio de um drone do tipo quadricóptero modelo DJI Mavic Enterprise 3T. As imagens foram obtidas em diferentes horários e ângulos de captura para garantir a diversidade do conjunto de dados e sua aplicabilidade

a cenários reais. Após a aquisição de dados, os dados foram separados conforme os critérios apresentados na Tabela 1. A Figura 4 apresenta como as imagens foram rotuladas e a frequência de instancias de cada classe presente no conjunto de dados pode ser observada na Figura 5.

Tabela 1: Padrão de imagens utilizadas

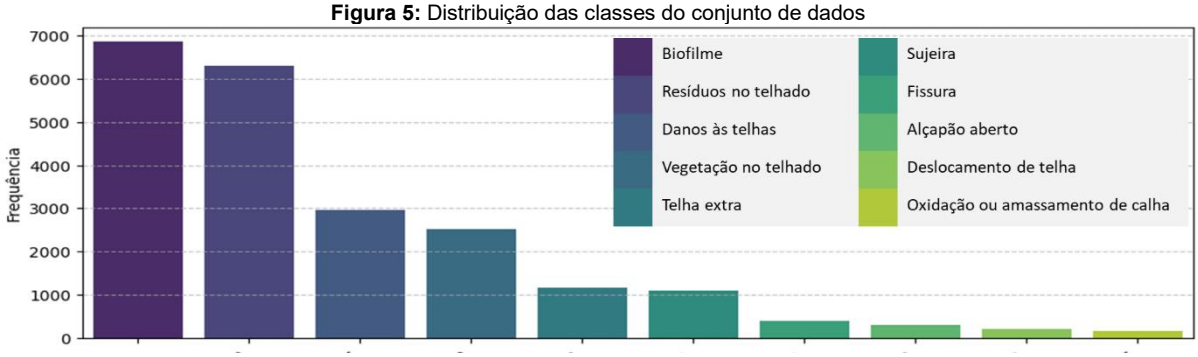
Resolução	5472x3648 pixels
Condições de Iluminação	Captadas em dias ensolarados
Variabilidade das coberturas	Telhados cerâmicos, metálicos, lajes, telhas de fibrocimento
Presença de danos	Presença de sujeira; Telha extra; Alçapão aberto; oxidação; Biofilme; Deslocamento de telhas; Fissura na laje; Presença de vegetação; Resíduos no telhado; Danos às telhas.
Tipo de imagem	RGB e Termais

Figura 4: Exemplo de imagem rotulada



Fonte: Os autores.

Figura 5: Distribuição das classes do conjunto de dados

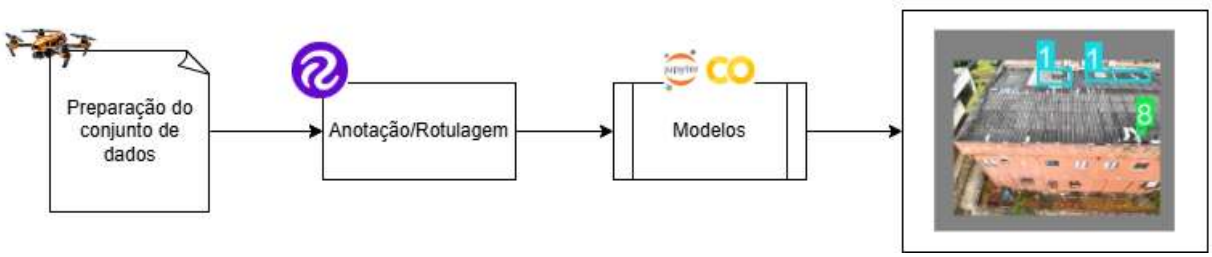


Fonte: Os autores.

3.2 Desenvolvimento dos modelos YOLO

O desenvolvimento dos modelos seguiu um fluxo metodológico estruturado em quatro etapas principais: (1) preparação do conjunto de dados, (2) anotação e rotulagem, (3) treinamento e ajuste dos modelos, e (4) avaliação comparativa, conforme Figura 6. Os modelos utilizados são encontrados na Tabela 2.

Figura 6: Processo de desenvolvimento do modelo



Fonte: Os autores.

a) Preparação do Conjunto de Dados

O conjunto de dados foi composto por 7.210 imagens capturadas por drone, selecionadas para representar a diversidade de defeitos em coberturas e telhados. As imagens passaram por pré-processamento padronizado, incluindo:

- Conversão para o formato JPEG para compatibilidade com frameworks de *deep learning*;
- Redimensionamento para resolução de 640×640 pixels, alinhado às especificações máximas das configurações small (S) das arquiteturas YOLO (Jocher et al., 2021);
- Divisão estratificada em subconjuntos de treinamento (70%), validação (20%) e teste (10%), preservando a distribuição balanceada de classes.

Optou-se por não aplicar técnicas de aumento de dados (*data augmentation*) para evitar introdução de viés artificial, mantendo a fidelidade das condições reais de captura.

b) Anotação e Rotulagem

As imagens foram anotadas utilizando a plataforma Roboflow, garantindo a demarcação precisa de bounding boxes para nove classes de defeitos (e.g., oxidação, fissuras, biofilme). O processo de rotulagem seguiu protocolos de qualidade, com validação cruzada por três especialistas em inspeção predial, reduzindo inconsistências e *outliers*.

c) Treinamento e Ajuste dos Modelos

Os modelos YOLOv5 a YOLOv11 (configuração S) foram implementados em ambiente de desenvolvimento baseado em *PyTorch*, com hardware dedicado (Tesla T4, NVIDIA A100). Para garantir comparabilidade equitativa:

- Hiperparâmetros críticos (*learning rate, batch size*, otimizador) foram fixados em valores padrão da documentação oficial;
- O número de épocas foi limitado a 300, com early stopping baseado no mAP50 do conjunto de validação.

d) Avaliação Comparativa:

A performance dos modelos foi quantificada por meio de métricas consolidadas na literatura:

- Precisão (*precision*) e Revocação (*Recall*), para análise da confiabilidade das detecções;
- MAP@50 (*mean Average Precision*), considerando *Intersection over Union* ≥ 0.5;
- Tempo de Inferência, medido em milissegundos (ms), para avaliação da eficiência computacional;

As predições foram avaliadas em cenário realísticos utilizando imagens inéditas do conjunto de teste.

Tabela 2: Modelos desenvolvidos e arquiteturas utilizadas

Modelo	Arquitetura (Fonte)	Número de parâmetros (mi)	Número de camadas (*layers*)	Tempo de inferência T4 (ms)	MAP@50-95 (*COCO Dataset*)
Modelo 1	YOLOv5S (*Ultralytics*)	~9.1	193	1,92	37,4
Modelo 2	YOLOv6S (*Meituan*)	~16.3	142	2,66	45
Modelo 3	YOLOv8S (*Ultralytics*)	~11.2	168	2,66	44,9
Modelo 4	YOLOv9S (*Ultralytics*)	~7,2	486	3,54	46,5
Modelo 5	YOLOv10S (*Ultralytics*)	~8	293	2,66	46,7
Modelo 6	YOLOv11S (*Ultralytics*)	~9.4	238	2,63	47

3.3 Avaliação e comparação

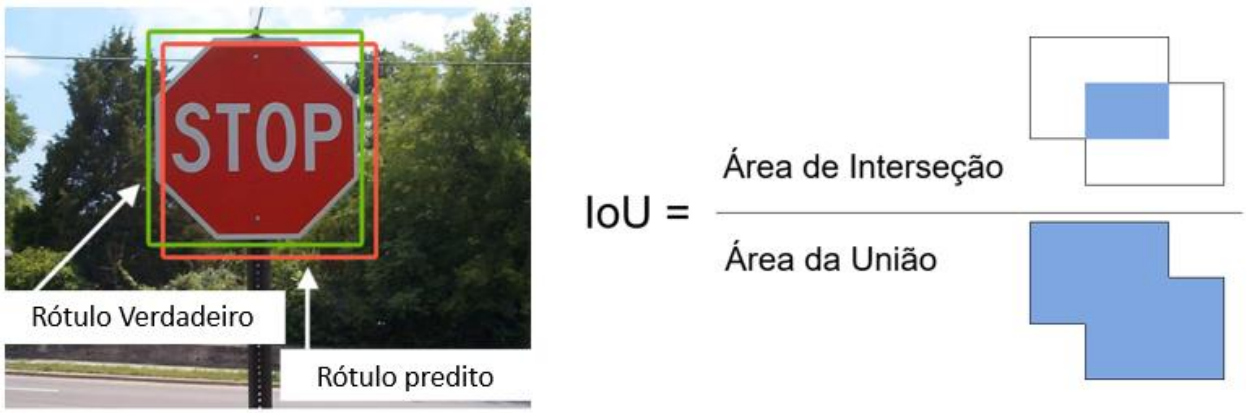
A avaliação dos modelos desenvolvidos foi realizada considerando métricas amplamente utilizadas na literatura para tarefas de detecção de objetos. Os principais indicadores analisados foram a precisão (*Precision*), a média das precisões médias (mAP@50) e o tempo de inferência, selecionados para quantificar a eficiência preditiva quanto a viabilidade computacional.

A precisão (Equação 1) reflete a proporção de detecções corretas em relação ao total de detecções realizadas, sendo uma métrica essencial para entender a confiabilidade do modelo ao identificar defeitos nos telhados, onde falsos positivos podem gerar custos operacionais desnecessários. Já o mAP@50 (*Mean Average Precision at IoU* 0.50), com a metodologia demonstrada na Figura 7, mede a qualidade geral das detecções considerando a sobreposição entre a predição e a anotação real, sendo um dos principais indicadores de desempenho em *benchmarks* como o COCO. Valores elevados de mAP@50 indicam robustez tanto na classificação quanto na localização de defeitos.

Além disso, o tempo de inferência foi avaliado para verificar a eficiência computacional dos modelos, mensurado em milissegundos (ms), reflete a eficiência computacional dos modelos, parâmetro essencial para implantação em sistemas embarcados ou dispositivos com restrições de hardware.

$$Precisão = \frac{Verdadeiros\ Positivos\ (VP)}{Verdadeiros\ Positivos\ (VP) + Falsos\ Positivos\ (FP)} \quad \text{(Eq.1)}$$

Figura 7: Esquema do cálculo do IoU (*Intersection over Union*) para o mAP



Fonte: Adaptado de Shah (2022)

Os resultados obtidos foram organizados em tabelas para facilitar a análise comparativa entre os diferentes modelos testados. Essa abordagem permite visualizar o desempenho relativo de cada arquitetura, auxiliando na escolha do modelo mais adequado para a tarefa proposta. A análise priorizou não apenas o ranking absoluto das métricas, mas também a consistência do desempenho entre classes de defeitos, garantindo que os modelos selecionados atendam a requisitos de generalização e confiabilidade operacional.

4. RESULTADOS

Esta seção apresenta os resultados obtidos nesta pesquisa.

4.1 Desempenho dos modelos

São apresentados, na tabela 3, os 6 modelos desenvolvidos. Foram tabulados os valores de Precisão e mAP@50 para cada classe. Os melhores resultados foram destacados. Os resultados gerais de cada modelo serão discutidos a seguir.

Tabela 3: Precisão e mAP50 para cada classe, por modelo

Modelos	Métricas	C1	C2	C3	C4	C5	C6	C7	C8	C9	C10
Modelo 1	Precisão	62,3%	68,7%	76,0%	54,4%	68,1%	66,0%	67,9%	**66,3%**	**70,8%**	**61,2%**
	mAP50	50,1%	59,4%	**66,4%**	41,3%	56,8%	32,6%	59,0%	42,7%	48,5%	48,1%
Modelo 2	Precisão	55,4%	61,1%	63,4%	32,0%	63,4%	51,0%	59,6%	55,5%	54,6%	49,8%
	mAP50	41,6%	49,0%	54,7%	24,6%	53,0%	27,1%	52,3%	36,3%	32,2%	38,2%
Modelo 3	Precisão	62,1%	67,1%	**84,0%**	58,6%	67,2%	70,6%	60,6%	58,3%	66,2%	58,0%
	mAP50	54,5%	61,7%	74,8%	42,3%	55,5%	39,2%	58,7%	40,5%	48,1%	46,7%
Modelo 4	Precisão	**65,6%**	69,2%	81,7%	**61,2%**	67,5%	**73,1%**	55,6%	64,3%	67,9%	59,6%
	mAP50	**54,4%**	**65,6%**	74,8%	42,8%	57,7%	**43,3%**	58,0%	**45,0%**	**48,5%**	49,2%
Modelo 5	Precisão	55,0%	**70,2%**	73,8%	60,5%	67,7%	63,3%	59,0%	57,0%	62,5%	58,7%
	mAP50	48,3%	62,6%	71,2%	**48,0%**	**58,5%**	42,1%	**60,0%**	40,9%	45,8%	**49,8%**
Modelo 6	Precisão	55,1%	67,5%	74,5%	57,3%	**70,1%**	64,0%	**74,4%**	65,8%	63,0%	61,0%
	mAP50	44,0%	58,1%	68,0%	35,4%	55,3%	38,0%	59,0%	43,2%	42,5%	46,0%

Nota: Classes: C1: Presença de sujeira; C2: Telha extra; C3: Alçapão aberto; C4: Oxidação; C5: Biofilme; C6: Deslocamento de telhas; C7: Fissura na laje; C8: Presença de vegetação; C9: Resíduos no telhado; C10: Danos às telhas.

Fonte: os autores

A partir dos resultados destacados na Tabela 3, notou-se que o modelo 2 apresenta desempenho inferior aos demais em todas as classes. Este resultado também pôde ser observado nas figuras 7 e 8, em que o desempenho em cada classe é avaliado. Em termos de precisão, pode-se destacar as seguintes classes para cada modelo:

i) Modelo 1: “Presença de vegetação”, “Resíduos no telhado” e “Danos às telhas”.

ii) Modelo 3: “Alçapão aberto”.

iii) Modelo 4: “Presença de sujeira”, “Presença de Oxidação” e “Deslocamento de telhas”.

iv) Modelo 5: “Telha extra”.

v) Modelo 6: “Biofilme” e “Fissura na laje”

Em termos de mAP@50, destacam-se as seguintes classes para cada modelo:

i) Modelo 1: “Resíduos no telhado”.

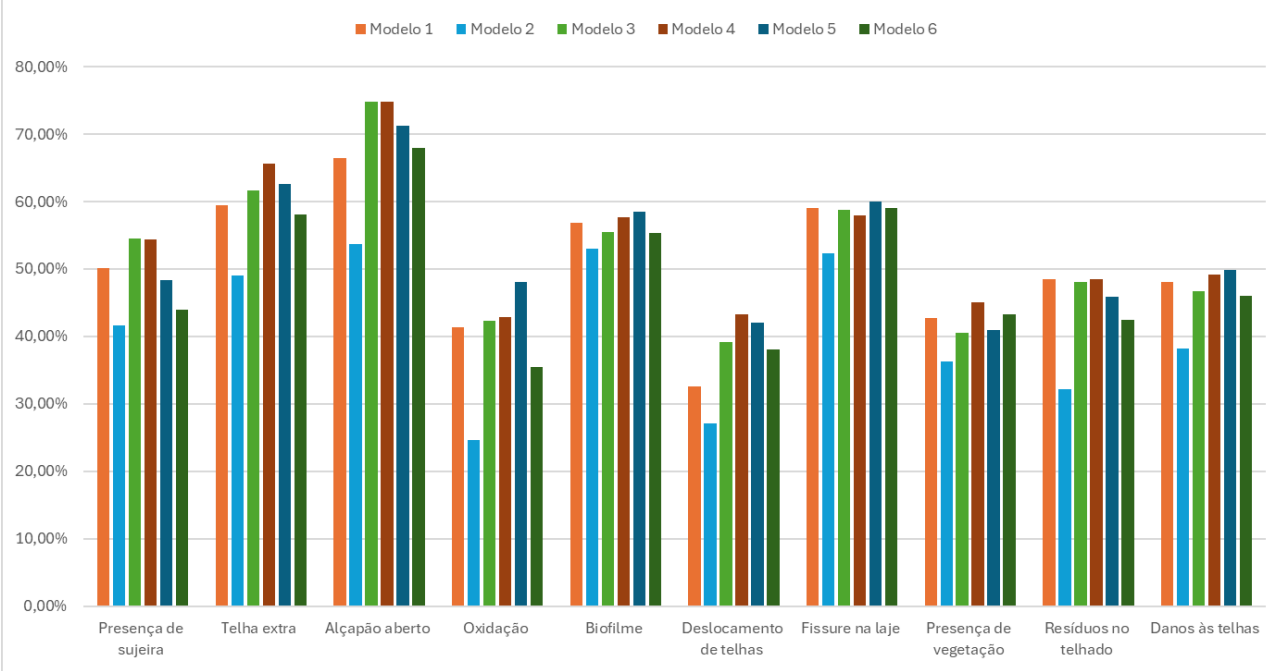
ii) Modelo 3: “Presença de sujeira” e “Alçapão aberto”

iii) Modelo 4: “Telha extra”, “Alçapão aberto”, “Deslocamento de telha”, “Presença de vegetação” e “Resíduos no telhado”.

iv) Modelo 5: “Oxidação”, “Biofilme”, “Fissura na laje” e “Danos as telhas”.

Observa-se que os modelos 2 e 6 não se destacaram em nenhuma das classes, em desempenho de mAP50. No entanto, no contexto de detecção em tempo real, maior precisão ou maior mAP não necessariamente refletem o melhor modelo, já que se procura um modelo que tenha equilíbrio entre essas métricas e o tempo de inferência. Nesta situação, é preferível o modelo com menor tempo de inferência, a qual será discutida adiante. Na figura 8 são avaliados o desempenho de mAP50 em cada classe, por modelo. Na figura 9, tem-se a precisão de cada modelo em sua respectiva classe.

Figura 8: mAP50 em cada classe por modelo

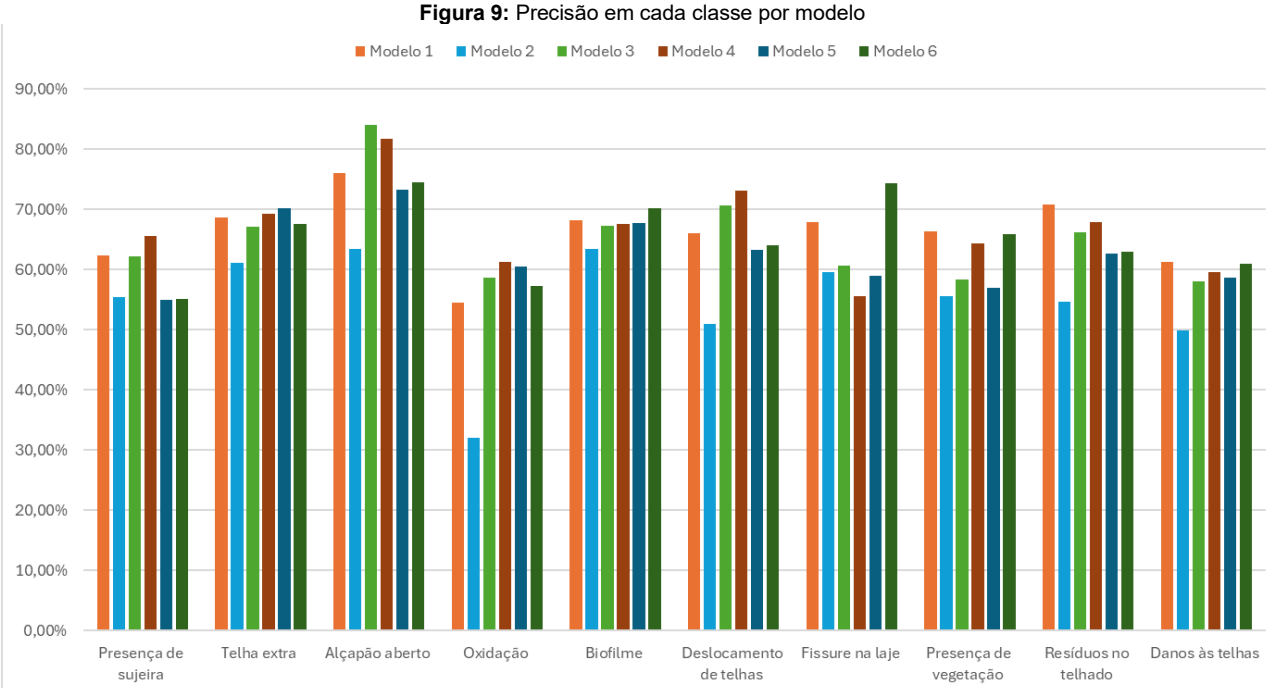


Fonte:

As classes "Telha extra", "Alçapão aberto", "Biofilme" e "Fissura na laje" apresentaram mAP50 superior a 50% (Figura 8), desempenho atribuível a características intrínsecas que facilitam a detecção. Por exemplo, a classe "Alçapão aberto" (74,8% de mAP50 médio) beneficia-se de alto contraste visual e formato geométrico regular em relação ao contexto, conforme descrito em estudos sobre detecção de anomalias baseada em características morfológicas. Analogamente, a precisão elevada para essa classe (Figura 9) reflete a baixa ambiguidade em suas representações visuais, reduzindo falsos positivos.

Entretanto, em cenários de detecção em tempo real, a excelência em precisão ou mAP50 não constitui critério único de seleção. Como evidenciado na Seção 4.2, modelos com alta acurácia podem exibir tempos de inferência incompatíveis com aplicações embarcadas, demandando uma avaliação holística que pondere *trade-offs* entre métricas de desempenho e eficiência computacional

Figura 9: Precisão em cada classe por modelo



Fonte: os autores

4.2 Comparação entre os modelos

A análise comparativa dos modelos YOLO, conforme sumarizada na Tabela 4, revela disparidades significativas no equilíbrio entre desempenho preditivo e eficiência computacional. O Modelo 4 destacou-se como o mais robusto em termos agregados, registrando a maior precisão (66,6%), recall (52,2%) e mAP@50 (53,9%) entre todas as versões avaliadas. Contudo, esse modelo apresentou o maior tempo de inferência (9,0 ms), fator crítico para aplicações em tempo real. Em contrapartida, o Modelo 1 posicionou-se como segunda melhor opção em precisão (66,2%), enquanto os Modelos 3 e 5 exibiram desempenho intermediário em mAP@50 (52,2% e 52,7%, respectivamente), com o Modelo 5 demonstrando superioridade em tempo de inferência (8,5 ms). O recall foi incluído nesta análise entre os modelos, com o intuito de identificar omissão de detecções que podem comprometer o desempenho do modelo, dado que o recall avalia a capacidade do modelo de identificar todos os objetos que deveriam ser detectados.

Tabela 4: Modelos desenvolvidos: Precisão, Recall e mAP50 para todas as classes

Modelos	Precisão	Recall	mAP@50	Inferência (ms)
#1 YOLOv5s	66,2%	48,5%	50,5%	6,6
#2 YOLOv6s	54,6%	42,8%	40,9%	7,4
#3 YOLOv8s	65,3%	51,9%	52,2%	7,1
#4 YOLOv9s	66,6%	52,2%	53,9%	9
#5 YOLOv10s	62,8%	51,8%	52,7%	8,5
#6 YOLOv11s	65,3%	46,6%	49,0%	7,5

Existe uma relação, ainda que discreta, entre maior tempo de inferência e maior mAP50. Calculando o coeficiente de Pearson (dada pela fórmula abaixo) para essas duas variáveis, temos aproximadamente 0,39. O valor calculado (r=0,39) indica uma correlação positiva moderada (Figura 9), sugerindo que modelos com maior complexidade arquitetural tendem a melhorar a acurácia às custas de eficiência computacional, padrão alinhado com observações em redes neurais profundas.

$$p = \frac{cov(mAP,\ Inf)}{\sigma_{mAP} \cdot \sigma_{Inf}}$$

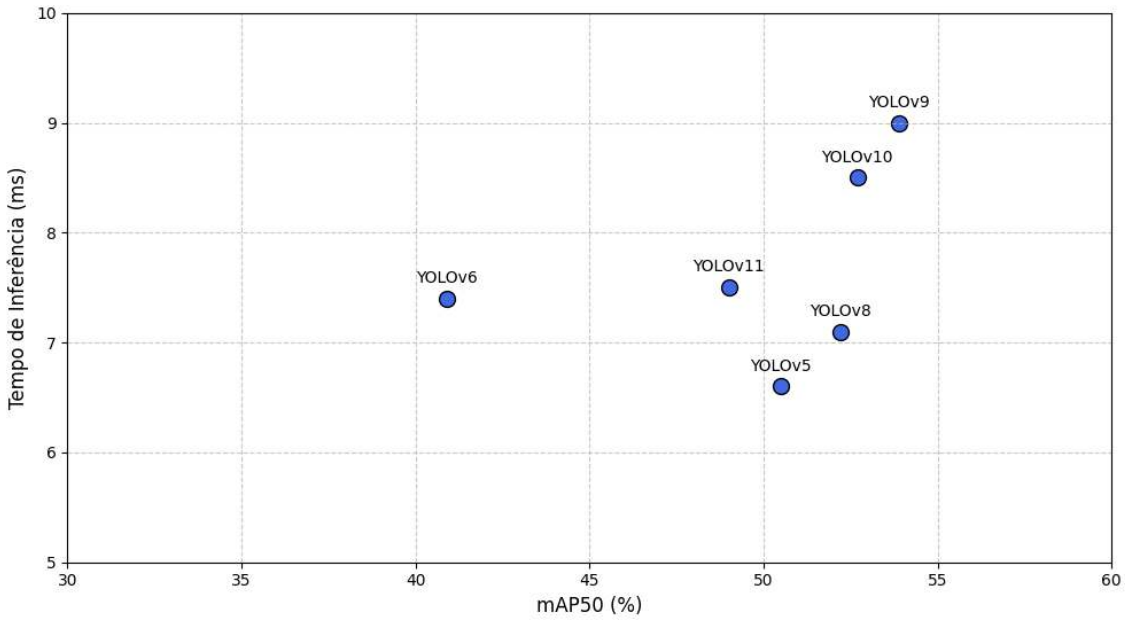
$$cov = Covariância\ entre\ o\ mAP\ e\ Inferência$$

$$\sigma_{mAP} = desvio\ padrão\ do\ mAP$$

$$\sigma_{Inf} = desvio\ padrão\ do\ Tempo\ de\ Inferência$$

Na figura 10, tem-se a relação entre o mAP50 (%) e tempo de inferência (ms). Quanto maior mAP50 e menor tempo de inferência, melhor. Nota-se que o modelo 4, relativo a YOLOv9, apresenta maior mAP, mas também apresenta o maior tempo de inferência. O modelo 3 e 5 (YOLOv8 e YOLOv10, respectivamente) apresentam desempenhos próximos em mAP, mas com tempos de inferência significativamente diferentes, com o modelo 3 com o menor valor. A análise comparativa entre os modelos YOLO foi realizada considerando duas métricas principais: a relação entre mAP@50 e tempo de inferência (Figura 7) e a relação entre mAP@50 e precisão (Figura 10). Essas métricas são fundamentais para avaliar o equilíbrio entre desempenho e eficiência computacional.

Figura 10: Relação entre mAP50 e Tempo de Inferência

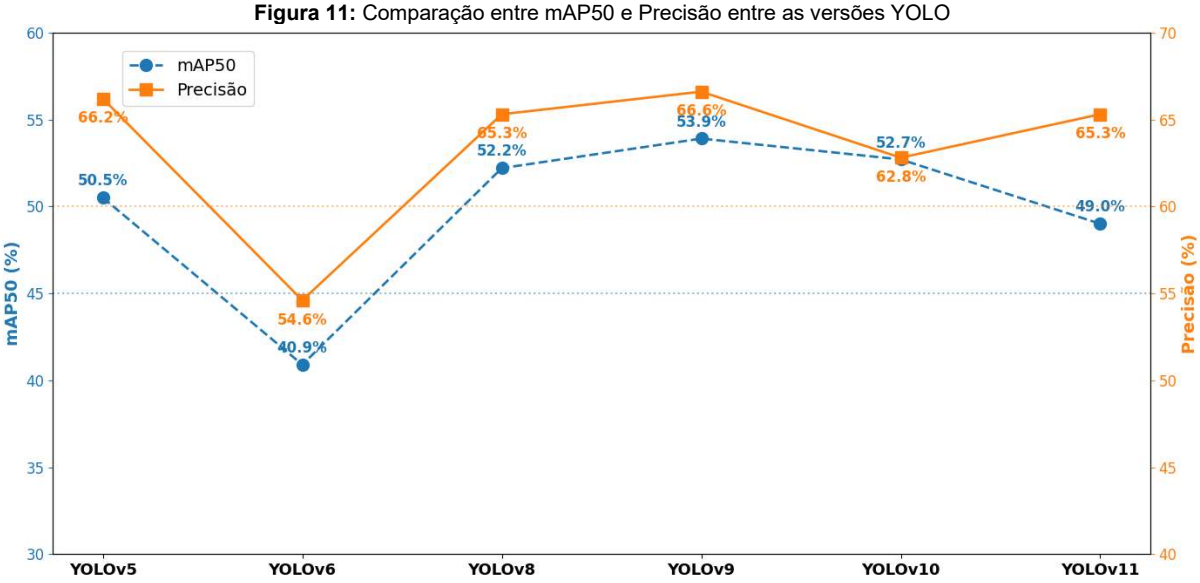


Fonte: Os autores.

Observa-se que os modelos com melhor desempenho em termos de mAP50 (YOLOv8, YOLOv9 e YOLOv10) apresentam um tempo de inferência maior, o que pode impactar sua aplicação em cenários de tempo real, com exceção da versão 8, que apresenta o segundo menor tempo de inferência. Em contraste, o YOLOv5 apresenta o tempo de inferência mais baixo, embora seu mAP@50 seja ligeiramente inferior a modelos mais recentes. O YOLOv6, por sua vez, apresenta o pior desempenho em mAP@50, o que sugere que essa versão não é a mais adequada para a detecção específica deste conjunto de dados.

A Figura 11 ilustra a variação entre mAP@50 e precisão para cada versão do YOLO. Nota-se que, embora o YOLOv9 apresente o maior mAP@50, o YOLOv5 e o YOLOv8 se destacam em termos de precisão, o que pode ser um fator relevante dependendo do contexto de aplicação. O YOLOv6 se destaca negativamente, apresentando tanto um mAP@50 quanto uma precisão mais baixas em relação às demais versões. Modelos como YOLOv10 e YOLOv11 apresentam um desempenho intermediário, mostrando valores equilibrados entre precisão e mAP@50. É possível notar, também, uma tendência de menor mAP@50 e maior precisão na YOLOv11, o que pode indicar que o modelo acerta as classes com mais corretude, mas não as localiza de forma precisa.

Figura 11: Comparação entre mAP50 e Precisão entre as versões YOLO



Fonte: Os autores.

4.3 Viabilidade de Implementação em Sistemas Embarcados:

O Modelo 4 (YOLOv9), embora destaque-se com a maior precisão (66,6%) e mAP@50 (53,9%), apresenta um tempo de inferência de 9,0 ms, o que pode ser crítico para implementação em dispositivos embarcados em drones com recursos computacionais limitados. Em comparação, o Modelo 3 (YOLOv8), apresenta um tempo de inferência mais baixo (7,1 ms), com um mAP@50 (52,2%) sendo o segundo maior apresentado pelos modelos, oferecendo um equilíbrio mais adequado entre desempenho e latência, alinhando-se a requisitos de processamento em tempo real. Nesse contexto, mesmo modelos com desempenho ligeiramente inferior (e.g., YOLOv8 com 7,1 ms) podem ser preferíveis, dependendo da criticidade da detecção de falsos positivos.

No contexto de detecção de fissuras, Yang et al. (2022) obteve 3% de mAP@50 com imagens térmicas, 13,6% com imagens RGB de mAP@50, por fim, o conjunto RGB e termais atingiram 16,3% de mAP@50, utilizando modelos desenvolvidos a partir da YOLOv3. Comparativamente, foi possível alcançar um resultado superior com todos os modelos desenvolvidos, variando entre 52% e 60% em mAP@50.

O objetivo deste comparativo é encontrar o melhor modelo que possa ser embarcado em um dispositivo, e assim realizar detecção em tempo real das classes citadas. Dado a necessidade de ser em tempo real, é necessário que o modelo seja o mais rápido possível, aliado à uma boa precisão.

4.4 Comparação com Benchmark

Os resultados deste estudo divergem significativamente dos reportados nas curvas oficiais do YOLO para a base COCO (Tabela 5). Enquanto o YOLOv8 atinge 37,3% de mAP@50 no COCO (Lin et al., 2014), o modelo apresentado alcançou 23,2%, diferença atribuível a dois fatores principais:

1. Complexidade do Domínio: A base COCO inclui 80 classes genéricas (e.g., "pessoa", "carro"), enquanto nosso conjunto foca em defeitos estruturais sutis (e.g., "oxidação", "biofilme"), que exigem granularidade maior.

2. Desequilíbrio de Classes: Defeitos como "fissura na laje" representam menos de 5% do dataset, reduzindo o mAP@50 em comparação à COCO (Ultralytics, 2023) onde as classes são balanceadas.

Tabela 5: Comparação entre modelos desenvolvidos e Benchmark

Modelos	mAP@50-95 (Conjunto de dados utilizado neste estudo)	MAP@50-95 (Conjunto de dados COCO)
Modelo 1 (YOLOv5)	22,9%	37,4%
Modelo 2 (YOLOv6)	16,8%	45,0%
Modelo 3 (YOLOv8)	23,2%	44,9%
Modelo 4 (YOLOv9)	24,2%	46,5%
Modelo 5 (YOLOv10)	23,8%	46,7%
Modelo 6 (YOLOv11)	21,9%	47,0%

Fonte: Os autores.

Essa disparidade ressalta a importância de ajustes específicos para domínios especializados, como técnicas de *fine-tuning* com camadas de atenção para defeitos de baixo contraste ou aumento de dados sintéticos para classes minoritárias.

CONCLUSÃO

Este trabalho teve como objetivo avaliar o desempenho de diferentes versões da arquitetura YOLO (v5 a v11) na detecção automática de defeitos em telhados e coberturas, utilizando imagens capturadas por drones. Para isso, foram treinados e comparados seis modelos em termos de precisão, mAP50 e tempo de inferência, com foco em sua viabilidade para implementação em sistemas embarcados. A análise priorizou o equilíbrio entre acurácia e eficiência computacional, contextualizando os resultados em cenários práticos de inspeção predial.

Os resultados demonstraram que o YOLOv9 (Modelo 4) destacou-se com a maior precisão (65,6%) e mAP50 (23,2%), porém com o maior tempo de inferência (9,0 ms). Em contraste, o YOLOv8 (Modelo 3) apresentou o segundo menor tempo de processamento (7,1 ms), mantendo desempenho competitivo (mAP50 = 52,2%),

o que o torna adequado para aplicações em tempo real. Observou-se uma correlação positiva moderada (p = 0,39) entre mAP50 e tempo de inferência, indicando que ganhos em acurácia muitas vezes exigem custos computacionais mais elevados. Além disso, classes como "Alçapão aberto" e "Biofilme" atingiram mAP50 superior a 50%, enquanto defeitos sutis (e.g., "Oxidação") apresentaram desafios significativos, refletindo a complexidade do domínio especializado.

Entre as limitações do estudo, destaca-se a ausência de técnicas de data augmentation, que poderiam melhorar a generalização dos modelos, e a restrição a testes em ambientes controlados, sem validação em campo com drones operacionais. Como direcionamentos futuros, sugere-se: i) Aplicar técnicas de otimização (*pruning*, quantização) para reduzir o custo computacional sem comprometer a acurácia; ii) Aumentar a disponibilidade de imagens das classes minoritárias, com dados reais ou sintéticos. Esses avanços contribuiriam para a adoção efetiva de sistemas autônomos de inspeção, alinhando-se às demandas por soluções precisas e energeticamente eficientes na indústria da construção civil, no contexto da gestão da manutenção.

AGRADECIMENTOS

Os autores agradecem a CAPES e ao CNPq por meio do número 443942/2024-2, pelo suporte financeiro para a pesquisa e pela equipe da Superintendência de Meio Ambiente e Infraestrutura da UFBA (SUMAI-UFBA).

REFERÊNCIAS

AFTAB, Muhammad; CHEN, Chien; CHAU, Chi-Kin; RAHWAN, Talal. **Automatic HVAC control with real-time occupancy recognition and simulation-guided model predictive control in low-cost embedded system**. *Energy and Buildings*, v. 154, p. 141-156, 2017. DOI: https://doi.org/10.1016/j.enbuild.2017.07.077.

ANIL KUMAR, Jakkani. **Artificial Intelligence and Its Applications in the Field of Internet of Things (IoT)**. *International Journal of Research in Science & Engineering*, v. 4, n. 5, p. 49-61, 2024. https://doi.org/10.55529/ijrise.45.49.61.

BADUGE, Shanaka Kristombu et al. **Artificial intelligence and smart vision for building and construction 4.0: Machine and deep learning methods and applications**. *Automation in Construction*, v. 141, p. 104440, 2022. DOI: https://doi.org/10.1016/j.autcon.2022.104440.

BOCHKOVSKIY, Alexey; WANG, Chien-Yao; LIAO, Hong-Yuan Mark. **YOLOv4: Optimal Speed and Accuracy of Object Detection**. *arXiv preprint arXiv:2004.10934*, 2020. Disponível em: https://arxiv.org/abs/2004.10934. Acesso em: 24 jan. 2025.

ĆOROVIĆ, A.; ILIĆ, V.; ĐURIĆ, S.; MARIJAN, M.; PAVKOVIĆ, B. **The real-time detection of traffic participants using YOLO algorithm**. In: **TELECOMMUNICATIONS FORUM (TELFOR), 26.**, 2018, Belgrade, Serbia. Anais [...]. Belgrade: IEEE, 2018. p. 1-4. DOI: 10.1109/TELFOR.2018.8611986.

CHUA, W. P.; CHEAH, C. C. **Deep-Learning-Based Automated Building Construction Progress Monitoring for Prefabricated Prefinished Volumetric Construction**. *Sensors*, v. 24, n. 21, p. 7074, 2024. https://doi.org/10.3390/s24217074.

DAVE, Jay; PATEL, Dr; RAVAL, Dr. **Towards Precise Water Quality Assessment: A Deep Learning Approach with Feature Selection in Smart Monitoring Systems**. I*nternational Journal of Scientific Research in Computer Science, Engineering and Information Technology*, v. 10, p. 100-114, jul. 2024. DOI: https://doi.org/10.32628/CSEIT241045.

GARG, C.; BANSAL, V.; GERA, N.; BANSAL, H.; GOEL, K. **A Deep Learning and Image Content Analysis enabled approach for Sustainable Transportation**. In: 2024 *IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*. Bangalore, Índia, 2024. p. 1-6. doi: 10.1109/CONECCT62155.2024.10677327.

HALEEM, Muhammad Salman. **Advances in Artificial Intelligence, Machine Learning and Deep Learning Applications**. *Electronics*, v. 12, n. 18, p. 3780, 2023. https://doi.org/10.3390/electronics12183780.

JIANG, S.; ZHANG, J. **Real-time crack assessment using deep neural networks with wall-climbing unmanned aerial system**. *Computer-Aided Civil and Infrastructure Engineering*, v. 35, p. 549–564, 2020. https://doi.org/10.1111/mice.12519.

JENCY, S.; RAMKUMAR, G. **Enhancing Deep Learning Model Performance for Surface Crack Detection**. In: *2024 5th International Conference on Smart Electronics and Communication (ICOSEC)*. Trichy, Índia, 2024. p. 1669-1674. doi: 10.1109/ICOSEC61587.2024.10722142.

JOCHER, Glenn; QIU, Jing. Ultralytics YOLO11: version 11 [software]. 2024. Disponível em: https://github.com/ultralytics/ultralytics. Licença AGPL-3.0. Acesso em: 04 jun. 2025.

KHOOJASTE-SARAKHSI, M.; HAGHIGHI, S. S.; GHOMI, S. M. T. F.; MARCHIORI, E. **Deep learning for Alzheimer's disease diagnosis: A survey**. *Artificial Intelligence in Medicine*, v. 130, p. 102332, 2022. https://doi.org/10.1016/j.artmed.2022.102332.

LECUN, Y.; BENGIO, Y.; HINTON, G. **Deep learning**. *Nature*, v. 521, p. 436–444, 2015. https://doi.org/10.1038/nature14539.

LIN, T. Y. et al. **Microsoft COCO: Common Objects in Context**. *arXiv preprint arXiv:1405.0312*, 2014. Disponível em: http://arxiv.org/abs/1405.0312.

MEHTA, Mihir. **Intelligent waste management: Maximize profits and minimize waste using IoT and AI**. World *Journal of Advanced Research and Reviews*, v. 24, n. 01, p. 693–701, 2024. https://doi.org/10.30574/wjarr.2024.24.1.3062.

MELO, R.; COSTA, D. **Integrating resilience engineering and UAS technology into construction safety planning and control**. *Engineering, Construction and Architectural Management,* v. 26, 2019.

PEINADO, H. S. et al. **Potential application of deep learning and UAS for guardrail safety inspection**. In: *Proceedings of the 31st Annual Conference of the International Group for Lean Construction (IGLC31)*. 2023. p. 387–398. doi: 10.24928/2023/0148.

PEREZ, Husein; TAH, Joseph H. M.; MOSAVI, Amir. **Deep Learning for Detecting Building Defects Using Convolutional Neural Networks**. *Sensors*, v. 19, n. 16, p. 3556, 2019. https://doi.org/10.3390/s19163556.

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.* 2016. p. 779–788.

SANZANA, Mirza Rayana et al. **Application of deep learning in facility management and maintenance for heating, ventilation, and air conditioning**. *Automation in Construction*, v. 141, p. 104445, 2022. DOI: https://doi.org/10.1016/j.autcon.2022.104445.

SANTOS, Lara Monalisa Alves dos; ZANONI, Vanda Alice Garcia; BEDIN, Eduardo; PISTORI, Hemerson. **Deep learning applied to equipment detection on flat roofs in images captured by UAV**. *Case Studies in Construction Materials*, v. 18, p. e01917, 2023. ISSN 2214-5095.

SCAIFE, Anthony D. **Improve predictive maintenance through the application of artificial intelligence: a systematic review**. *Results in Engineering*, v. 17, p. 101645, 2024. Disponível em: https://doi.org/10.1016/j.rineng.2023.101645

SHAH, Deval. **Mean Average Precision (mAP) Explained: Everything You Need to Know**. *V7 Labs*, [s.d.]. Disponível em: https://www.v7labs.com/blog/mean-average-precision. Acesso em: mar. 2025.

SHEHAB, M. et al. **Machine learning in medical applications: A review of state-of-the-art methods**. *Computers in Biology and Medicine*, v. 145, p. 105458, 2022. https://doi.org/10.1016/j.compbiomed.2022.105458.

SILVA, Alisson Souza et al. Modelo baseado em CNN para reconhecimento automatizado de anomalias na execução de fachadas visando apoio à Gestão da Qualidade. **Ambiente Construído**, v. 25, p. e143288, 2025.

SILVA, A. S.; MELO, R.; COSTA, D. B. Automated facade inspection: application and challenge in using Artificial Intelligence for construction defect recognition. In: **XX International Conference on Building Pathology and Constructions Repair (CINPAR 2024)**, Fortaleza, Ceará, 2024.

STAFFA, L.; COSTA, D. B.; NOGUEIRA, J. **Web platform for building roof maintenance inspection using UAS and artificial intelligence**. *International Journal of Building Pathology and Adaptation*, v. 1, p. 1, 2023.

ULTRALYTICS. Performance metrics. In: Ultralytics YOLO11 Documentation. 2024. Disponível em: https://docs.ultralytics.com/models/yolo11/#performance-metrics. Acesso em: mar 2025.

YANG, Z. et al. **A Real-time Tunnel Surface Inspection System using Edge-AI on Drone**. In: *2022 IEEE International Conference on Mechatronics and Automation (ICMA)*. Guilin, China, 2022. p. 749-754. doi: 10.1109/ICMA54519.2022.9856230.

YANG, W., & Jiachun, Z. (2018, July). **Real-time face detection based on YOLO**. In 2018 1st IEEE international conference on knowledge innovation and invention (ICKII) (pp. 221-224). IEEE.

YIN, Robert K. **Case study research and applications: design and methods**. 6. ed. Los Angeles: SAGE, 2018.

ZHANG, Guoping; ZHANG, Qiuyue. **Building Engineering Cost Prediction Based On Deep Learning: Model Construction and Real-Time Optimization**. *Journal of Electrical Systems*, v. 20, n. 5s, p. 151-164, 2024.