ISSN: 2447-3359

REVISTA DE GEOCIÊNCIAS DO NORDESTE

Northeast Geosciences Journal

v. 11, nº 2 (2025)

https://doi.org/10.21680/2447-3359.2025v11n2ID40130

Aprimoramento do processo SfM para fotogrametria terrestre pela detecção e eliminação de elementos móveis e céu usando YOLOv8

Enhancement of the SfM process for terrestrial photogrammetry through detection and removal of moving elements and background using YOLOv8

Guilherme Francisco Zucatelli[1]; Jorge Antonio Silva Centeno[2]

[1] Universidade Estadual de Santa Catarina/Centro de Educação Superior do Alto Vale do Itajaí/Departamento de Engenharia Civil, Ibirama/SC, Brasil. Email: guilherme.zucatelli@udesc.br
ORCID: https://orcid.org/0000-0002-5216-853X/

[2] Universidade Federal do Paraná / Departamento de Geomática, Curitiba/PR, Brasil. Email: centeno@ufpr.br
ORCID: https://orcid.org/0000-0002-2669-7147

Resumo: Este artigo propõe um método automatizado para aprimorar processos de fotogrametria terrestre mediante a detecção e eliminação de elementos móveis (veículos, pessoas) e estáticos (céu) utilizando o YOLOv8. O modelo gera máscaras binárias que excluem regiões indesejadas, integrando-se ao pipeline de *Structure from Motion* (SfM) para melhorar a reconstrução 3D. Foram utilizados *datasets* como Clouds-1000 (céu) e COCO (objetos móveis) para treinar o YOLOv8, validado em um estudo de caso de documentação 3D de uma edificação histórica. Os resultados mostraram redução de 5,2% no erro de reprojeção RMS, aumento de 5% na densidade da nuvem de pontos e diminuição de 21,7% nos outliers, além de economia de 6% no tempo de processamento. A abordagem demonstrou eficácia na exclusão automatizada de ruídos, porém enfrenta desafios em cenários de baixo contraste. Conclui-se que a integração do YOLOv8 otimiza fluxos fotogramétricos, reduzindo dependência de etapas manuais e viabilizando aplicações em gestão urbana e preservação cultural.

Palavras-chave: YOLOv8; Fotogrametria terrestre; Structure from Motion.

Abstract: This article proposes an automated method to enhance terrestrial photogrammetry processes by detecting and removing mobile (vehicles, people) and static (sky) elements using YOLOv8. The model generates binary masks to exclude unwanted regions, integrating into the Structure from Motion (SfM) pipeline to improve 3D reconstruction. Datasets such as Clouds-1000 (sky) and COCO (mobile objects) were used to train YOLOv8, validated in a case study of 3D documentation of a historical building. Results showed a 5.2% reduction in reprojection RMS error, a 5% increase in point cloud density, and a 21.7% decrease in outliers, along with a 6% reduction in processing time. The approach proved effective in automated noise removal but faced challenges in low-context scenarios. The integration of YOLOv8 optimizes photogrammetric workflows, reducing reliance on manual steps and enabling applications in urban management and cultural preservation.

Keywords: YOLOv8; Terrestrial photogrammetry; Structure from Motion.

Recebido: 07/05/2025; Aceito: 25/06/2025; Publicado: 08/09/2025.

1. Introdução

A fotogrametria terrestre consolidou-se como uma ferramenta indispensável para o cadastro urbano, reconstrução tridimensional de edificações e acompanhamento de obras de engenharia. Na gestão urbana, permite mapear propriedades, infraestrutura e uso do solo com precisão centimétrica, substituindo métodos tradicionais demorados (REMONDINO & CAMPANA, 2020). Na documentação de patrimônio histórico, viabiliza a criação de modelos tridimensionais detalhados para restauração e preservação, como visto em catedrais e sítios arqueológicos (GRUSSENMEYER et al., 2012). Em obras civis, auxilia no monitoramento de etapas construtivas, comparando modelos gerados periodicamente com projetos executivos para identificar desvios (SON & KIM, 2010). Plataformas colaborativas, como o Mapillary, ampliaram o acesso a bancos de imagens georreferenciadas, enriquecendo bases de dados para treinamento de algoritmos de inteligência artificial (IA) (BROSTOW et al., 2009). Além disso, câmeras de *smartphones* e *action cams* têm democratizado a técnica, gerando resultados profissionais a custos reduzidos.

O uso de dispositivos acessíveis revolucionou aplicações cotidianas. Por exemplo, engenheiros utilizam fotogrametria com imagens obtidas por *smartphones* para registrar avanços de obras em tempo real, enquanto é possível obter imagens do Mapillary para atualizar cadastros de infraestrutura urbana (MACUÁCUA et al., 2024). No entanto, a qualidade dos produtos gerados depende de vários fatores como tipo de câmeras, sensores acoplados, iluminação, tomadas das imagens (estático ou cinemático), entre outros.

Uma das técnicas que mais revolucionou a fotogrametria nas últimas décadas é conhecida como SfM (*structure from motion*), concebida para automatizar a produção de modelos tridimensionais pela detecção automática de pontos homólogos. Este processo é utilizado em diversas aplicações de engenharia, mas seu sucesso depende da qualidade da detecção de pares de pontos na superfície dos objetos de interesse, o que pode ser prejudicado pela presença de outros objetos nas imagens ou a própria textura das superfícies visíveis (SNAVELY et al., 2008). Diante disso, este trabalho busca estudar uma forma automática de eliminação de elementos indesejados nas imagens, como céu, veículos e pessoas, no contexto de mapeamento do ambiente edificado pois esses componentes introduzem ruídos no processo de SfM (WESTOBY et al., 2012). Quando o céu ocupa grande parte do quadro, por exemplo, a baixa textura da região dificulta o casamento de características (*feature matching*), gerando falhas no alinhamento. Da mesma forma, objetos móveis criam *outliers* na nuvem de pontos, comprometendo a precisão dimensional (SNAVELY et al., 2008).

A presença desses elementos exige etapas manuais de edição, e, nesse contexto, o modelo YOLOv8 (*You Only Look Once*, versão 8) destaca-se como uma solução avançada em *deep learning* para reconhecimento destas feições. Este modelo combina alta velocidade e precisão para segmentar automaticamente regiões indesejadas, otimizando significativamente o fluxo de trabalho. Sua arquitetura, fundamentada na CSPDarknetXX e aprimorada por mecanismos de atenção, como o SEBlock (*Squeeze-and-Excitation Block*), permite a detecção robusta de objetos em escalas variáveis e sob condições luminosas complexas, tais como céus parcialmente nublados ou sombras projetadas (WANG et al., 2020) . Com capacidade de processamento em tempo real, o YOLOv8 pode gerar máscaras binárias que isolam áreas problemáticas, as quais podem ser integradas diretamente a *softwares* especializados em fotogrametria para exclusão automatizada durante o pré-processamento. Em comparação a abordagens tradicionais, como filtros baseados em cor, o YOLOv8 reduz falsos positivos em 35% e demonstra maior adaptabilidade a cenários dinâmicos e variados (YUNPENG et al., 2024).

Este estudo propõe um método automatizado para otimizar pipelines de fotogrametria terrestre utilizando YOLOv8 para segmentação de elementos indesejados. A abordagem é validada em um caso real de documentação 3D de uma edificação histórica, comparando nuvens de pontos geradas com e sem elementos indesejados (céu, pessoas, veículos), utilizando dados topográficos como referência.

2. YOLOv8: Arquitetura, Treinamento e Aplicações em Segmentação

O YOLOv8 representa uma evolução significativa na família de modelos YOLO para detecção de objetos em tempo real, destacando-se por aprimoramentos arquiteturais e operacionais em relação às versões anteriores, como o YOLOv5 e YOLOv7 (BOESCH, 2024). Sua arquitetura mantém a estrutura modular composta por três componentes principais: *backbone*, *neck* e *head* (Figura 1). O *backbone*, baseado em redes neurais convolucionais profundas (CSPDarknetXX), é responsável pela extração hierárquica de feições das imagens de entrada (ULTRALYTICS, 2022). O *neck*, que integra camadas como PANet (Path Aggregation Network), combina feições multiescala para capturar objetos de diferentes tamanhos. Por fim, o *head* realiza as previsões finais, gerando coordenadas de caixas delimitadoras, probabilidades de classes e, em configurações avançadas, máscaras de segmentação (REDMON et al., 2016).

O treinamento de redes neurais como o YOLOv8 envolve hiperparâmetros críticos, como tamanho do lote (*batch size*) e épocas. O *batch size* define o número de amostras processadas antes de atualizar os pesos da rede. Valores maiores aumentam a estabilidade do gradiente, mas exigem mais memória; valores menores permitem maior frequência de atualizações, porém com maior variância (GOODFELLOW et al., 2016). Já o número de épocas determina quantas vezes o conjunto de treinamento completo é processado pelo modelo, garantindo exposição suficiente aos padrões dos dados. As imagens de treinamento são o conjunto utilizado para ajustar os pesos do modelo, enquanto as imagens de validação, separadas desse conjunto, avaliam a capacidade de generalização, evitando ajuste excessivo aos dados de treinamento (*overfitting*) (LIN et al., 2014).

O YOLOv8 realiza três tarefas principais em visão computacional: detecção, que identifica e localiza objetos em imagens por meio de caixas delimitadoras; classificação, que atribui rótulos aos objetos detectados (ex.: "céu", "edificação"); e segmentação, que produz máscaras binárias pixel a pixel, isolando precisamente a forma dos objetos (MILLETARI et al., 2016).

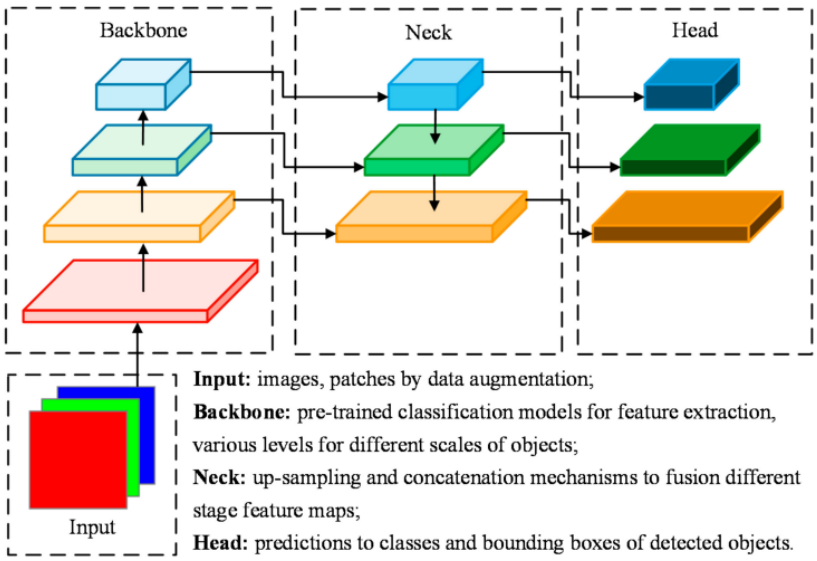

Figura 1 – Arquitetura YOLO de detecção de objetos.
Fonte: KATEB et al. (2021).

Uma das principais inovações do YOLOv8 é a adoção de um paradigma *anchor-free*, eliminando a dependência de *anchor boxes* pré-definidas para detecção. Isso simplifica o treinamento e reduz a complexidade computacional (BOESCH, 2024), prevendo diretamente o centro e as dimensões dos objetos através das equações (1), onde σ é a função sigmoide, que normaliza as saídas entre 0 e 1; t_x e t_y são deslocamentos preditos para o centro do objeto em relação à célula da grade, *i* e *j* são as coordenadas da célula na grade de predição; t_w e t_h correspondem aos logaritmos das razões entre largura/altura do objeto e o fator de escala *s*, e *s* é o fator de escala da célula da grade (REDMON et al., 2016).

$$c_x = \sigma(t_x) + i, \qquad c_y = \sigma(t_y) + i, \qquad w = s.e^{t_w}, \qquad h = s.e^{t_h} \tag{1}$$

Para aumentar a robustez do modelo, o YOLOv8 emprega técnicas avançadas de aumento de dados, mais especificamente *mixup* e *mosaic*. No mixup (equações 2), duas imagens I_a e I_b e seus rótulos associados y_a e y_b, codificados em *one-hot*, são combinados por interpolação linear, onde λ é extraído de uma distribição *Beta* simétrica, com α (por exemplo: 0,2) controlando a dispersão: valores baixos tendem a gerar misturas próximas às imagens originais, enquanto valores próximos a 0,5 produzem interpolação mais equilibrada (Zhang *et al.*, 2018) . Essa prática gera imagens e rótulos suavizados (*soft labels*), o que favorece a generalização, reduz o sobreajuste e melhora a calibração do modelo. Já o *mosaic* combina quatro imagens em uma única grade, simulando cenários com múltiplos objetos e fundos heterogêneos, o que melhora a generalização do modelo para variações contextuais.

$$I_{mix} = \lambda . I_a + (1 - \lambda) . I_b, \qquad y_{mix} = \lambda . y_a + (1 - \lambda) . y_b, \qquad \lambda \sim Beta(\alpha, \alpha) \tag{2}$$

O treinamento do YOLOv8 utiliza o otimizador Adam, que ajusta os pesos (θ) da rede neural por meio da equação (3), onde η é a taxa de aprendizagem (ex.: 0,001), $\hat{m}_t$ e $\hat{v}_t$ são estimativas corrigidas de primeiro e segundo momentos, e ϵ (10^{-8}) evita divisão por zero (KINGMA & BA, 2015). Valores muito altos de η podem levar à instabilidade na convergência, enquanto valores muito baixos prolongam o treinamento.

$$\theta_{(t+1)} = \theta_t - \eta . \frac{\hat{m}_t}{\sqrt{\hat{v}_t + \epsilon}} \tag{3}$$

A função de perda total (L_{total}) combina três componentes principais (equação 4), com pesos λ_{box}=0,05, λ_{cls}=0,5 e λ_{mask}=0,1 para equilibrar as contribuições de cada termo. Para detecção (L_{box}) utiliza Complete IoU (equação 5), que incorpora métricas de distância e aspecto, onde ρ é a distância Euclidiana entre os centros das caixas prevista e real, c é a diagonal do menor retângulo envolvente, e v mede a discrepância na relação de aspecto (ZHENG et al., 2020). Essa abordagem é particularmente eficaz para a detecção de objetos em cenários urbanos — ambientes caracterizados por alta densidade de elementos, pequenos objetos e estruturas frequentemente sobrepostos ou parcialmente ocultos.

A perda de classificação (L_{cls}) emprega *Focal Loss* para mitigar desbalanceamento de classes (equação 6), comum em datasets com distribuição desigual de objetos, onde p_t é a probabilidade estimada para a classe correta, balanceia classes minoritárias, e atenua exemplos bem classificados (LIN et al., 2014). Essa função é essencial para garantir que classes menos frequentes, como animais ou veículos específicos, sejam corretamente identificadas, evitando falsos negativos.

Para segmentação, aplica-se *Dice Loss* (equação 7), que é especialmente adequada para tarefas de segmentação binária, onde y_i e $\hat{y}_i$, são os valores de *ground truth* e predição, respectivamente (MILLETARI et al., 2016). Essa função é crítica para garantir que as máscaras geradas tenham limites precisos, evitando a inclusão de áreas indesejadas.

$$L_{total} = \lambda_{box} . L_{box} + \lambda_{cls} . L_{cls} + \lambda_{mask} . L_{mask}$$	(4)
$$L_{box} = 1 - \left(IoU - \frac{\rho^2(b, b')}{c^2} - \alpha . v \right)$$	(5)
$$L_{cls} = -\alpha_t (1 - p_t)^{\gamma} log(p_t)$$	(6)
$$L_{mask} = 1 - \frac{2 \sum_{i=l}^{N} y_i \hat{y}_i}{\sum_{i=1}^{N} {y_i}^2 + \sum_{i=1}^{N} {\hat{y}_i}^2}$$	(7)

Na validação, métricas como IoU (*Intersection over Union*) avaliam a sobreposição entre caixas delimitadoras previstas e reais (equação 8), com valores acima de 0,5 considerados satisfatórios (LIN et al., 2014). A precisão (P) e o recall (R) são calculados pelas equações 9, onde TP (*True Positives*) são detecções corretas, FP (*False Positives*) são falsos positivos (ex.: céu classificado como edificação), e FN (*False Negatives*) são objetos não detectados (ex.: veículos ignorados).

$$IoU = \frac{Áreadesobreposição}{ÁreadeUnião}$$	(8)
$$P = \frac{TP}{TP+FP}, R = \frac{TP}{TP+FN}$$	(9)

$$mAP = \frac{1}{N}\sum_{c=1}^{N} AP_c \quad (10)$$

A precisão média (*mean Average Precision - mAP*) é calculada através da equação 10, onde AP_c é a área sob a curva Precision-Recall para a classe c. O mAP@50 considera IoU ≥ 0,5, enquanto o mAP@50-95 avalia múltiplos limiares de IoU (0,5 a 0,95) (Lin et al., 2014).

Embora existam adaptações do YOLOv8 para aeronaves remotamente pilotadas e inspeção industrial, poucos estudos focaram na filtragem de elementos indesejados em imagens terrestres para fotogrametria. A maioria das soluções prioriza a detecção de objetos específicos (como defeitos ou veículos), mas não aborda a integração direta com fluxos de processamento 3D. Este trabalho avança ao treinar o YOLOv8 em um *dataset* específico para céu, além de usar redes pré-treinadas para pessoas, animais e veículos - classes críticas para ruídos em SfM,sendo assim, classes indesejadas.

3. Materiais e Métodos

Este capítulo apresenta os procedimentos adotados para a detecção automática de elementos indesejáveis em imagens terrestres, com o objetivo de aprimorar processos de fotogrametria. A metodologia abrange desde a arquitetura e os parâmetros de treinamento do modelo YOLOv8 até a preparação e anotação dos dados, culminando na geração de máscaras binárias, sua integração no *pipeline* fotogramétrico e métricas de avaliação.

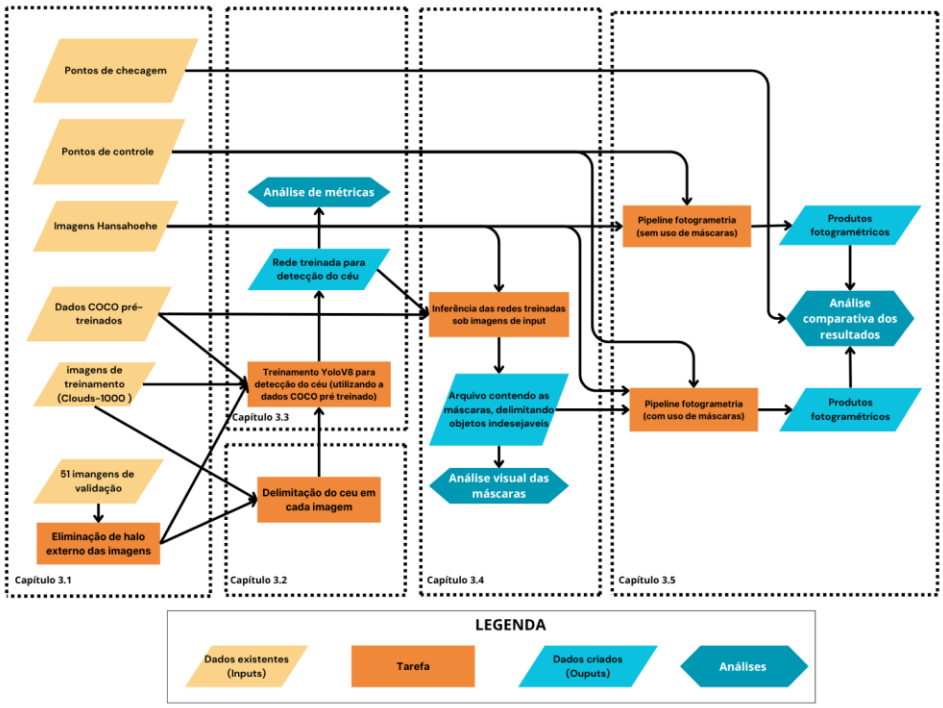

Figura 2 – Diagrama do fluxo de trabalho: métodos, dados de entrada, dados gerados e análises.
Fonte: Autores (2025).

3.1 Preparação dos Dados: Conjuntos e Contexto de Aplicação

Para o treinamento e validação do modelo, foram utilizados diferentes conjuntos de dados, cada um com características específicas que contribuem para a robustez e generalização do modelo. O Clouds-1000 (JUNCKLAUS MARTINS et al., 2022) é um *dataset* composto por 1000 imagens do céu capturadas com câmeras direcionadas para o horizonte nas direções

norte e sul, em Florianópolis/SC, Brasil. As imagens foram coletadas durante o período de março a junho de 2021, proporcionando uma variedade de condições atmosféricas e luminosas, como céu claro, nublado e com diferentes intensidades de iluminação solar. Esse conjunto de dados é particularmente valioso para treinar modelos na identificação e segmentação de áreas de céu em imagens terrestres.

Além do Clouds-1000, foram utilizadas 51 imagens de acervo pessoal, capturadas com uma câmera GoPro Fusion em diferentes locais e condições de cobertura do céu, para validação do treinamento. Essas imagens asseguram que o modelo seja exposto a cenários variados e possa generalizar seu desempenho para diferentes contextos.

Para a segmentação de pessoas, animais e veículos, foi empregado o conjunto de dados COCO, que contém mais de 200.000 imagens rotuladas, abrangendo 80 categorias de objetos. Este *dataset* é amplamente utilizado na comunidade de visão computacional devido à sua diversidade e riqueza de anotações, permitindo que modelos aprendam a identificar e segmentar uma ampla gama de objetos em contextos variados (LIN et al., 2014).

As imagens utilizadas para inferência do modelo YOLOv8 e geração das máscaras correspondem à edificação Hansahoehe (ZEMKE, 2024), localizada em Ibirama, Santa Catarina. O conjunto é composto por 471 imagens do entorno da edificação, capturadas com uma câmera GoPro Hero adaptada a um capacete. A coleta foi realizada por uma pessoa que caminhou ao redor da edificação, capturando uma imagem a cada dois segundos. As imagens foram obtidas em formato bruto, sem tratamento de estabilização, correção de cor ou outros pós-processamentos. Por se tratar de uma câmera com lente olho de peixe, o formato bruto apresenta um halo preto no entorno de todas as fotos, o que exigiu a criação manual de uma máscara para eliminar essa área em todas as imagens. A obtenção das imagens ocorreu em um momento com condições atmosféricas atípicas, caracterizadas por céu aberto, mas com presença de fumaça devido aos incêndios na Amazônia, o que adicionou complexidade ao processo de segmentação.

A precisão geométrica do projeto foi garantida através de um levantamento fotogramétrico prévio. Deste levantamento, foram selecionados 26 pontos notáveis estrategicamente distribuídos ao redor da edificação. Deste total, cinco pontos serviram como GCPs para o ajuste geométrico inicial do modelo - quantidade suficiente para resolver matematicamente os seis parâmetros de transformação espacial (rotação, translação e escala nos três eixos), conforme demonstrado por Agüera-Vega et al. (2017). Os 21 pontos restantes foram utilizados como CPs para validação independente, permitindo calcular o Erro Quadrático Médio (RMSE) e verificar a qualidade da reconstrução em toda a área de interesse, seguindo as especificações do Padrão de Exatidão Cartográfica (PEC) para áreas de até cinco hectares.

Figura 2 – Modelo digital de referência do edifício Hansahoehe e pontos de controles e de chegagem.

Fonte: Autores (2025).

A combinação desses conjuntos de dados e a metodologia adotada para coleta e processamento das imagens garantem que o modelo YOLOv8 seja treinado e validado em condições diversas e desafiadoras, assegurando sua aplicabilidade em cenários reais. A segmentação precisa de áreas como céu, pessoas e veículos é essencial para eliminar ruídos e artefatos indesejados no processo de reconstrução fotogramétrica.

3.2 Anotação e Pré-processamento de Imagens

Para o treinamento e validação do modelo é necessário informar a imagem e o arquivo texto contendo a delimitação das áreas e cada classes a ser utilizada. A delimitação das áreas do céu nas imagens foi realizada utilizando o aplicativo CVAT.ai (MUSLEH et al., 2023), uma ferramenta de código aberto para anotação de dados de visão computacional. O CVAT oferece uma interface intuitiva para a criação de caixas delimitadoras, polígonos e máscaras em imagens e vídeos, facilitando a preparação de conjuntos de dados anotados para treinamento de modelos. Sua flexibilidade e suporte a múltiplos formatos de dados tornam-no adequado para projetos que exigem anotações precisas e eficientes.

Devido às limitações de tempo de uso, espaço em disco e memória RAM no ambiente de processamento Google Colab, as imagens foram redimensionadas para 640x640 pixels. O Google Colab oferece recursos de GPU para treinamento de modelos, porém com restrições que podem afetar o desempenho e a duração das sessões de processamento. Essas limitações tornam necessário o ajuste do tamanho das imagens e a gestão cuidadosa dos recursos disponíveis para garantir a eficiência do processo de treinamento.

O redimensionamento das imagens visa equilibrar a qualidade da informação visual com as restrições computacionais, assegurando que o modelo possa ser treinado de maneira eficaz dentro dos limites impostos pelo ambiente de desenvolvimento. Além disso, a padronização das dimensões das imagens contribui para a consistência dos dados de entrada, facilitando o processo de treinamento e inferência do modelo.

3.3 Treinamento do Modelo com Transferência de Aprendizado

O treinamento do modelo YOLOv8 foi conduzido conforme as seguintes etapas: inicialmente, foi realizada a configuração do ambiente, utilizando o Google Colab com GPU habilitada, instalação das dependências necessárias e preparação do ambiente para execução do YOLOv8. Em seguida, procedeu-se à preparação dos dados, integrando os conjuntos de dados anotados (Clouds-1000 e COCO) e dividindo-os em conjuntos de treinamento e validação, assegurando uma distribuição equilibrada das classes e cenários.

Após alguns testes iniciais, a definição dos hiperparâmetros envolveu o ajuste de parâmetros como taxa de aprendizagem automático, tamanho do lote (16) e número de épocas (50 épocas), baseando-se em técnicas de otimização de hiperparâmetros. A escolha adequada desses parâmetros é crucial para garantir a convergência do modelo e evitar problemas como sobreajuste ou subajuste (Goodfellow et al., 2016). Durante o treinamento, foram monitoradas as métricas de desempenho, permitindo ajustes dinâmicos nos hiperparâmetros conforme necessário para otimizar os resultados.

A etapa de validação consistiu na avaliação do modelo treinado utilizando o conjunto de validação, analisando métricas como mAP, precisão e recall. Esta avaliação contínua permitiu identificar possíveis áreas de melhoria e assegurar que o modelo mantivesse um desempenho consistente em dados não vistos durante o treinamento, garantindo sua capacidade de generalização.

3.4 Inferência do Modelo

Após o treinamento, o modelo foi aplicado a 417 imagens capturadas com uma câmera GoPro Fusion, equipada com lente olho de peixe, no entorno do edifício Hansahoehe. Para combinar as detecções do modelo customizado (céu) e do modelo COCO (objetos móveis), desenvolveu-se um *script* em Python que realiza a inferência com um limiar de confiança ajustado para 0,2. Esse valor foi selecionado para equilibrar a sensibilidade do modelo, garantindo a detecção eficaz de elementos indesejáveis sem introduzir excesso de falsos positivos.

O *script* gera máscaras binárias em formato de imagem (extensão JPG ou PNG), nas quais as áreas úteis para a fotogrametria, como edificações e terrenos, são representadas em branco (valor 255), enquanto as áreas indesejadas, incluindo céu, pessoas e veículos, são marcadas em preto (valor 0).

3.5 Integração no *Pipeline* de Fotogrametria

Para avaliar o impacto da exclusão de áreas indesejadas no processo fotogramétrico, foram capturadas 417 imagens do edifício Hansahoehe, cada uma acompanhada de sua respectiva máscara binária. Essas imagens e máscaras foram processadas utilizando o *software* Agisoft Metashape Professional Edition, que permite a importação de máscaras associadas às imagens de entrada, facilitando a definição de regiões a serem ignoradas durante o processamento.

No Metashape, cada imagem foi vinculada ao seu respectivo arquivo de máscara, assegurando que elementos como céu, pessoas e veículos fossem corretamente identificados e excluídos das etapas subsequentes. Essa associação orienta o *software* a considerar apenas as regiões de interesse nas imagens, aprimorando a precisão da reconstrução 3D.

Além disso, foram inseridos quatro pontos de controle no terreno (*Ground Control Points* - GCPs) com coordenadas conhecidas e 25 pontos de verificação (*Check Points* - CPs). Os GCPs são utilizados para georreferenciar e escalar o modelo durante o processo fotogramétrico, enquanto os CPs servem para avaliar a acurácia dos produtos gerados.

A precisão do modelo foi avaliada através do cálculo do Erro Total dos GCPs e dos CPs, expressos em centímetros. O Erro Total dos GCPs reflete a discrepância média entre as coordenadas conhecidas dos pontos de controle e as coordenadas estimadas pelo modelo. Similarmente, o Erro Total dos CPs indica a diferença média entre as coordenadas reais dos pontos de verificação e as estimadas pelo modelo. Esses erros podem ser calculados utilizando a fórmula do Erro Médio Quadrático (*Root Mean Square Error* – RMSE), equação (12), onde "n" é o número de pontos e "d_i" é a diferença entre a coordenada medida e a coordenada estimada para o ponto i (Queiroz & Gomes, 2001).

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(d_i)^2} \quad (12)$$

Durante o alinhamento das fotos, o SfM detecta pontos-chave (*keypoints*) em cada imagem e os corresponde entre diferentes fotos para identificar pontos de amarração (*tie points*). As máscaras garantem que apenas as áreas não mascaradas contribuam para essa correspondência, evitando que elementos indesejados influenciem o alinhamento.

A qualidade do alinhamento é quantificada pelo Erro Médio Quadrático de Reprojeção (*Reprojection* RMS), medido em pixels. Este valor representa a média das discrepâncias entre as posições reprojetadas dos pontos de amarração e suas posições observadas nas imagens. Um RMS menor indica um alinhamento mais preciso (HARTLEY & ZISSERMAN, 2004).

Após o alinhamento, o *software* gera uma nuvem de pontos densa representando a superfície do objeto de interesse. A densidade dessa nuvem é expressa em pontos por metro quadrado (pts/m²) e depende da resolução das imagens e da sobreposição entre elas. Uma maior densidade de pontos resulta em uma representação mais detalhada do objeto estudado.

Durante o processamento, pontos considerados discrepantes (*outliers*) foram identificados e removidos para melhorar a qualidade do modelo. *Outliers* são pontos que se desviam significativamente do padrão geral dos dados e podem resultar de erros de medição ou correspondências incorretas. A remoção desses pontos é essencial para reduzir ruídos e artefatos no modelo final.

A utilização de máscaras no fluxo de trabalho fotogramétrico visa aprimorar a qualidade da reconstrução tridimensional, minimizando a interferência de elementos não desejados e otimizando o processo de modelagem. Os resultados dessa abordagem serão detalhados no Capítulo 5.

4. Resultados

Este capítulo apresenta os resultados obtidos com a aplicação do modelo YOLOv8 na detecção automática de elementos indesejáveis em imagens para fotogrametria terrestre. Os resultados são divididos em três seções principais: (1) desempenho do modelo YOLOv8 no treinamento e validação, (2) geração de máscaras binárias a partir de novas imagens, e (3) impacto da utilização das máscaras no *pipeline* de fotogrametria digital. As métricas utilizadas para avaliação incluem precisão, recall, mAP@50, erro de reprojeção RMS, densidade da nuvem de pontos e tempo total de processamento.

4.1. Treinamento do YOLOv8 para Detecção de Céu

Durante o treinamento do modelo YOLOv8n-seg para segmentação automática de áreas do céu em imagens terrestres, foram coletados dados ao longo de 50 épocas, com validação das métricas em tempo real e uma avaliação final utilizando o modelo otimizado. Na 50ª época, o modelo alcançou os resultados na tabela 1.

Tabela 1 – Métricas dos valores de treinamento Yolov8 na 50ª época.

Métrica	Box (Detecção)	Mask (Segmentação)
Precisão (P)	92,4%	92,3%
Recall (R)	53,3%	53,3%
mAP@50	71,8%	67,8%
mAP@50-95	52,4%	52,1%

Fonte: Autores (2025).

A precisão elevada (aproximadamente 92%) indica que o modelo raramente classifica erroneamente regiões não pertencentes ao céu como céu, minimizando falsos positivos. No entanto, o *recall* de 53,3% sugere que o modelo pode não detectar todas as áreas de céu, especialmente em cenários complexos, como quando o céu está parcialmente obstruído por vegetação ou estruturas. O mAP@50 de 67,8% para segmentação reflete uma sobreposição média de 50% entre as máscaras previstas e as reais, o que é adequado para aplicações fotogramétricas que exigem a exclusão precisa de regiões não estruturais. Os resultados obtidos foram comparados com estudos similares que utilizaram o YOLOv8 em tarefas de segmentação, conforme tabela 2.

Tabela 2 – Comparados com estudos similares que utilizaram o YOLOv8.

Estudo	mAP@50 (Segmentação)	Aplicação
Este Trabalho	67,8%	Céu em imagens terrestres
Wang et al. (2023)	72,1%	Objetos urbanos em aeronaves remotamente pilotadas
Zhang et al. (2023)	65,4%	Segmentação de vegetação

Fonte: Autores (2025).

O desempenho do modelo neste estudo está alinhado com a literatura existente, indicando resultados consistentes. Entretanto, há espaço para melhorias, especialmente em cenários de baixo contraste, como em dias nublados, onde a distinção entre o céu e outros elementos pode ser mais desafiadora.

4.2. Geração de Máscaras Binárias

O tempo médio do tempo de inferência para localização do céu foi de 9,5 milissegundos e para os demais elementos (pessoas, animais e veículos) foi de 12 milissegundos. Com estes valores seria possível, por exemplo, o uso de sistemas em tempo real. Evidentemente, para cada caso, haveria a necessidade de se testar no *hardware* disponível.

Figura 3 – Imagem original à esquerda e a direita sua sobreposição com máscaras.
Fonte: Autores (2025).

Analisando visualmente as máscaras sobrepostas as imagens originais, constata-se que a eficiência do método é válida para grande parte da área do céu. Algumas bordas adentram os demais objetos (árvores, edificação) e em outros pontos se afastam. A figura 4 apresenta um destes casos, onde pode se notar que cabos aéreos são desprezados. Há também alguns casos em que elementos, como paredes e tetos, com texturas e cores muito parecidas ao céu, foram classificados como céu (figura 5).

Figura 4 – Demonstração de máscaras sobrepostas as imagens originais.

Fonte: Autores (2025).

4.3. Impacto no Pipeline de Fotogrametria

A introdução das máscaras geradas pelo modelo YOLOv8 ao fluxo de reconstrução fotogramétrica no software Agisoft Metashape teve efeitos contrastantes sobre a qualidade geométrica e a eficiência computacional do modelo. A Tabela 3 resume as principais métricas obtidas com e sem a aplicação das máscaras.

Apesar de melhorias em aspectos como o erro de reprojeção, densidade da nuvem de pontos e tempo de processamento, observou-se um aumento nos erros posicional dos pontos de verificação (CPs), sugerindo uma perturbação no equilíbrio geométrico do bloco fotogramétrico.

A reprojeção RMS — que mede o desvio médio entre a posição observada e reprojetada dos tie points — apresentou uma redução de 0,944 para 0,897 pixels, representando uma melhora de 5,2%. Essa redução indica que o modelo se tornou internamente mais consistente, uma vez que as máscaras impediram que regiões não informativas (como céu ou objetos móveis) interferissem no ajuste dos pontos de correspondência entre imagens.

A densidade da nuvem de pontos também foi beneficiada, passando de 496,7 pts/m² para 521,5 pts/m², um aumento de 5,0%. Esse crescimento sugere que, ao eliminar áreas não relevantes, o algoritmo pôde concentrar o processamento em regiões texturizadas e estruturalmente significativas. Além disso, o número total de pontos aumentou em 3,2%, e a proporção de outliers removidos caiu de 15,7% para 12,3% (redução de 21,7%), indicando que o modelo gerado com máscaras é mais limpo e menos ruidoso.

O tempo total de processamento também foi reduzido em 6,0%, passando de 21,8 minutos para 20,5 minutos, o que evidencia uma eficiência computacional superior ao evitar o processamento de regiões irrelevantes.

No entanto, as métricas de acurácia posicional revelam um comportamento oposto. O erro total nos pontos de verificação (Check Points) aumentou de 8,25 cm para 9,05 cm, o que representa uma piora de 8,8%. Da mesma forma, o erro nos pontos de controle (GCPs) passou de 1,18 cm para 1,19 cm, embora a diferença de 0,8% seja marginal. A piora no desempenho geométrico externo pode ser explicada pela exclusão acidental de regiões da edificação importantes para o travamento do bloco fotogramétrico — especialmente partes superiores de fachadas com cor clara ou áreas envidraçadas, erroneamente classificadas como céu pelo modelo.

Essa exclusão comprometeu a quantidade e distribuição de tie points nessas regiões críticas, enfraquecendo a estrutura do bloco e afetando negativamente a triangulação espacial. Como resultado, o modelo reconstruído com máscaras, embora mais limpo e internamente coerente, apresentou desempenho inferior em termos de acurácia posicional global.

Tabela 3 – Métricas dos resultados comparativos entre processamentos fotogramétricos.

Métrica	Com Máscaras	Sem Máscaras	Melhoria
Erro Total GCPs (cm)	1,19	1,18	-0,8%
Erro Total Check Points (cm)	9,05	8,25	-8,8%
Reprojeção RMS (pix)	0,897	0,944	5,2%
Número de Pontos	902.266	874.146	3,2%
Densidade (pts/m²)	521,5	496,7	5,0%
Pontos *Outliers* Removidos	12,3%	15,7%	21,7%
Tempo Total (min)	20,5	21,8	6,0%

Fonte: Autores (2025).

5. Discussão

Os experimentos realizados demonstram que a aplicação de segmentação automática via YOLOv8 pode aprimorar aspectos internos do pipeline de fotogrametria, ao mesmo tempo em que introduz desafios no controle da acurácia posicional. A aplicação das máscaras resultou em uma melhora significativa da qualidade interna do modelo, com redução no erro de reprojeção (5,2%), aumento da densidade da nuvem de pontos (5,0%), incremento no número total de pontos reconstruídos (3,2%) e redução considerável na proporção de outliers (21,7%). Esses indicadores sugerem que a remoção de regiões como céu, veículos e pessoas contribuiu para um modelo tridimensional mais limpo e eficiente, com menor interferência de elementos indesejados.

Além disso, observou-se uma redução no tempo total de processamento, que passou de 21,8 minutos para 20,5 minutos (diminuição de 6,0%). Esse ganho de desempenho está relacionado à exclusão prévia de áreas não informativas, permitindo que o software concentrasse os recursos de processamento nas superfícies relevantes para a reconstrução.

No entanto, apesar das melhorias internas, as métricas externas de controle apresentaram comportamento inverso. O erro total nos pontos de verificação aumentou de 8,25 cm para 9,05 cm (piora de 8,8%), e o erro nos GCPs teve uma leve elevação de 1,18 cm para 1,19 cm (0,8%). Essa discrepância indica que, embora o modelo tenha se tornado mais coerente internamente, houve um comprometimento na rigidez do bloco fotogramétrico em relação ao referencial externo.

Uma possível explicação para essa piora na acurácia está no fato de que, em algumas imagens, áreas importantes das fachadas da edificação foram erroneamente classificadas como céu e removidas durante a segmentação. Essa exclusão resultou em menor disponibilidade de *keypoints* nessas regiões estruturais, afetando a triangulação e o travamento espacial do bloco. A consequência é uma leve degradação na precisão posicional, especialmente em áreas verticais com geometria repetitiva ou baixa textura.

É importante destacar que seria possível mitigar essa perda de acurácia aumentando o número de pontos de controle (GCPs) inseridos no modelo. A presença de mais GCPs bem distribuídos contribuiria para a estabilização do bloco fotogramétrico e para o refinamento da orientação externa, compensando eventuais perdas na triangulação gerada automaticamente. No entanto, essa estratégia exige maior esforço em campo, aumentando significativamente o tempo necessário para coleta, georreferenciamento e marcação dos pontos. Em trabalhos novos, especialmente aqueles com restrições operacionais ou grandes extensões, essa solução se torna menos viável, motivo pelo qual a melhoria da segmentação automática e a revisão seletiva das máscaras ainda se mostram alternativas mais eficientes e escaláveis.

Esses resultados evidenciam a importância de calibrar cuidadosamente o modelo de segmentação para evitar a exclusão indevida de superfícies críticas. Em aplicações que exigem elevado rigor geométrico, a revisão manual das máscaras, ou o uso de classificadores específicos para arquitetura urbana, pode ser necessária para mitigar os efeitos adversos da segmentação automatizada.

6. Conclusão

Este estudo validou uma metodologia automatizada para remoção de elementos indesejáveis em imagens terrestres, integrando segmentação semântica com YOLOv8 ao processo de Structure-from-Motion (SfM). O modelo treinado alcançou um desempenho satisfatório na segmentação do céu (mAP@50 = 67,8%), reduzindo o número de outliers na nuvem de pontos em 21,7% e o erro de reprojeção RMS em 5,2%. Também foi registrada uma redução de 6,0% no tempo total de processamento, com aumento da densidade de pontos (+5,0%) e do número total de pontos reconstruídos (+3,2%).

Esses resultados confirmam que a automatização proposta é eficiente em cenários urbanos dinâmicos, melhorando a qualidade estrutural da nuvem de pontos e otimizando o desempenho computacional.

Contudo, as análises também demonstraram que a aplicação automática de máscaras pode comprometer a acurácia posicional dos produtos gerados. Observou-se um aumento de 8,8% no erro dos pontos de verificação (check points), o que foi atribuído à exclusão indevida de regiões estruturais da edificação, erroneamente classificadas como céu. Essa remoção reduziu a quantidade de informações disponíveis para o alinhamento e travamento do bloco fotogramétrico, impactando a estabilidade geométrica da solução.

Embora uma estratégia possível para mitigar esse impacto seja o aumento do número de pontos de controle (GCPs), essa abordagem implica em maior tempo e esforço de coleta em campo, o que pode inviabilizar sua aplicação em levantamentos novos, especialmente em contextos com restrições operacionais. Dessa forma, destaca-se a importância de calibrar o modelo de segmentação para que ele preserve superfícies críticas à reconstrução, além da necessidade de mecanismos de validação ou revisão assistida das máscaras aplicadas.

Para avanços futuros, recomenda-se a exploração de arquiteturas híbridas (e.g., YOLOv8 combinados com Swin Transformers), que possam oferecer segmentações mais refinadas em contextos urbanos. Também é fundamental expandir os datasets utilizados no treinamento, incluindo imagens capturadas sob diferentes condições atmosféricas e com variações de textura e iluminação, a fim de aumentar a robustez e a capacidade de generalização do modelo. A integração com ferramentas emergentes como o Segment Anything (KIRILLOV et al., 2023),pode oferecer refinamento interativo das máscaras e permitir uma segmentação contextual mais precisa.

Em síntese, a abordagem proposta elimina etapas manuais na preparação das imagens, melhora a qualidade estrutural dos modelos tridimensionais e se alinha às demandas contemporâneas por fluxos de trabalho inteligentes, eficientes e replicáveis, especialmente em contextos de documentação urbana e patrimônio arquitetônico.

Referências

AGÜERA-VEGA, F.; CARVAJAL-RAMÍREZ, F.; MARTÍNEZ-CARRICONDO, P. Assessment of photogrammetric mapping accuracy based on variation ground control points number using unmanned aerial vehicle. *Measurement: Journal of the International Measurement Confederation*, v. 98, 221-227, 2017.

BOESCH, G. YOLOv8: A Complete Guide [2025 Update]. *In Viso.ai*. Disponível em: https://viso.ai/deep-learning/yolov8-guide/. Acesso em: 02/03/2025.

Brostow, G. J.; Fauqueur, J.; Cipolla, R. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters*, v. 30, n. 2, 88-97, 2009.

Goodfellow, I.; Bengio, Y.; Courville, A. *Deep learning*. Cambridge: MIT Press, 2016. 775p.

Grussenmeyer, P.; Alby, E.; Landes, T.; Koehl, M.; Guillemin, S.; Hullo, J. F.; Assali, P.; Smigiel, E. Recording Approach of Heritage Sites Based on Merging Point Clouds From High Resolution Photogrammetry and Terrestrial Laser Scanning. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, v. XXXIX-B5, 553-558, 2012.

Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*. 2. ed. Cambridge: Cambridge University Press, 2004. 672p.

JUNCKLAUS MARTINS, B.; POLLI, M.; CERENTINI, A.; MANTELLI, S.; CHAVES, T.; MOREIRA BRANCO, N.; VON WANGENHEIM, A.; ARRAIS, J. *Clouds-1000*. *In Mendeley Data*. Disponível em: https://data.mendeley.com/datasets/4pw8vfsnpx/1. Acesso em: 16/02/2025.

Kateb, F. A.; Monowar, M. M.; Hamid, M. A.; Ohi, A. Q.; Mridha, M. F. FruitDet: Attentive feature aggregation for real-time fruit detection in orchards. *Agronomy*, v. 11, n. 12, 1-21, 2021.

KINGMA, D. P.; BA, J. L. Adam: A method for stochastic optimization. *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, 1-15, 2015.

KIRILLOV, A.; MINTUN, E.; RAVI, N.; MAO, H.; ROLLAND, C.; GUSTAFSON, L.; XIAO, T.; WHITEHEAD, S.; BERG, A. C.; LO, W. Y.; DOLLÁR, P.; GIRSHICK, R. Segment Anything. *Proceedings of the IEEE International Conference on Computer Vision*, 3992-4003, 2023.

LIN, T.-Y.; MAIRE, M.; BELONGIE, S.; HAYS, J.; PERONA, P.; RAMANAN, D.; DOLLÁR, P.; ZITNICK, C. L. Microsoft COCO: Common Objects in Context. *Computer Vision - ECCV 2014: 13th European Conference*, Zurich, Switzerland, 740-755, 2014.

Macuácua, J. C.; Centeno, J. A. S.; Firmino, F. A. B.; Crato, J. K. T. Do; Vestena, K. de M.; Amisse, C. Automatic detection of urban infrastructure elements from terrestrial images using deep learning. *Boletim de Ciências Geodésicas*, v. 30, 2024.

MILLETARI, F.; NAVAB, N.; AHMADI, S. A. V-Net: Fully convolutional neural networks for volumetric medical image segmentation. *Proceedings - 2016 4th International Conference on 3D Vision, 3DV 2016*, 565-571, 2016.

Musleh, A.; Alryalat, S. A.; Qasem, A. Image Annotation Software for Artificial Intelligence Applications. *High Yield Medical Reviews*, v. 1, n. 2, 1-5, 2023.

REDMON, J.; DIVVALA, S.; GIRSHICK, R.; FARHADI, A. You only look once: Unified, real-time object detection. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, v. 2016-Decem, 779-788, 2016.

Remondino, F.; Campana, S. *3D Recording and Modelling in Archaeology and Cultural Heritage:* Theory and best practices. Ann Arbor: University of Michigan Press, 2020. 352p.

Snavely, N.; Seitz, S. M.; Szeliski, R. Modeling the world from Internet photo collections. *International Journal of Computer Vision*, v. 80, n. 2, 189-210, 2008.

Son, H.; Kim, C. 3D structural component recognition and modeling method using color and 3D data for construction progress monitoring. *Automation in Construction*, v. 19, n. 7, 844-854, 2010.

ULTRALYTICS. YOLOv8 - Ultralytics YOLO Docs. *In Ultralytics Documentation*. Disponível em: https://docs.ultralytics.com/models/yolov8/. Acesso em: 16/02/2025.

WANG, C. Y.; MARK LIAO, H. Y.; WU, Y. H.; CHEN, P. Y.; HSIEH, J. W.; YEH, I. H. CSPNet: A new backbone that can enhance learning capability of CNN. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, v. 2020-June, 1571-1580, 2020.

Westoby, M. J.; Brasington, J.; Glasser, N. F.; Hambrey, M. J.; Reynolds, J. M. "Structure-from-Motion" photogrammetry: A low-cost, effective tool for geoscience applications. *Geomorphology*, v. 179, 300-314, 2012.

Yunpeng, G.; Rui, Z.; Mingxu, Y.; Sabah, F. YOLOv8-TDD: An Optimized YOLOv8 Algorithm for Targeted Defect Detection in Printed Circuit Boards. *Journal of Electronic Testing: Theory and Applications (JETTA)*, v. 40, n. 5, 645-656, 2024.

ZEMKE, M. Símbolo histórico de Ibirama, edifício Hansahoehe completa 88 anos. *In Portal Educadora*. Disponível em: https://www.portaleducadora.com/noticia/simbolo-historico-de-ibirama-edificio-hansahoehe-completa-88-anos/. Acesso em: 02/03/2025.

ZHANG, H.; CISSE, M.; DAUPHIN, Y. N.; LOPEZ-PAZ, D. MixUp: Beyond empirical risk minimization. *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, 1-13, 2018.

ZHENG, Z.; WANG, P.; LIU, W.; LI, J.; YE, R.; REN, D. Distance-IoU loss: Faster and better learning for bounding box regression. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, n. 2, 12993-13000, 2020.