Submission date: 06/12/2025
Resubmission date: 01/05/2026
Camera ready submission date: 05/08/2026

1st round notification: 16/03/2026
2nd round notification: 17/07/2026
Available online: 07/09/2026
Publication date: 07/09/2026

Section: regular article

Segmentação Semântica de Flanges em Nuvens de Pontos Industriais com PointNet++

Title: Semantic Segmentation of Flanges in Industrial Point Clouds Using PointNet++

Davi Ribeiro de Carvalho[1], **Caio Eduardo Pinheiro Costa**[1], **Leonardo Santana Almeida da Silva**[1]

[1]Curso de Ciência da Computação – Universidade Jorge Amado (UNIJORGE)
Salvador – BA – Brasil

daviribeiro3002@gmail.com, caio.costa@unijorge.edu.br, leoalm@gmail.com

Abstract. *Industry 4.0 has fostered the use of 3D point clouds for inspection, as-built modeling and decision support in engineering. However, the automated interpretation of such data remains challenging, particularly in complex industrial environments where the manual identification of components is time-consuming and error-prone. Although recent studies address the semantic segmentation of industrial scenes, none of them treats the flange as an isolated target class, which is the specific gap addressed here. This work investigates the use of PointNet++ for point-wise semantic segmentation of flanges in industrial equipment point clouds. We propose a pipeline that preserves physical scale (metric coordinates), computes geometric features with fixed physical radii, and samples spherical chunks of variable radius during training to capture multi-scale context. The network is trained with a combined Focal + Lovász-Softmax loss and uses GroupNorm to stabilise inference with small batch sizes. On the test set, the final model achieved an overall accuracy of* 90.41% *and a mean IoU of* 70.47%*, with a recall of* 90.75% *and an F1 of* 68.09% *for the flange class. Comparative experiments against four traditional machine learning baselines (Random Forest, Gradient Boosting, Logistic Regression and XGBoost) demonstrated the clear superiority of the deep learning approach, which outperformed the best classical model by over* 37 *percentage points in mean IoU. An ablation study comparing three levels of data augmentation revealed that heavy domain randomisation can hurt performance in small, homogeneous datasets, motivating the adoption of a lightweight training pipeline. From an Information Systems perspective, the results demonstrate the feasibility of transforming raw scanning data into semantically enriched information capable of supporting Scan-to-BIM workflows, asset management and maintenance planning in industrial plants. The study also discusses organisational and socio-technical implications, as well as limitations related to dataset size and class imbalance.*

Cite as: Carvalho, D. R., Costa, C. E. P. & Silva, L. S. A. (2026). Semantic Segmentation of Flanges in Industrial Point Clouds Using PointNet++ (Segmentação Semântica de Flanges em Nuvens de Pontos Industriais com PointNet++). iSys: Revista Brasileira de Sistemas de Informação (iSys - Journal of Information Systems), 19(1), 11:1-11:27. DOI: 10.5753/isys.2026.7126.

Keywords. *3D point cloud; PointNet++; Semantic segmentation; Scan-to-BIM; Industry 4.0; Information Systems.*

Resumo. *A Indústria 4.0 tem impulsionado o uso de nuvens de pontos 3D para inspeção, modelagem as-built e apoio à tomada de decisão em engenharia. No entanto, a interpretação automatizada desses dados ainda é desafiadora, especialmente em ambientes industriais complexos, onde a identificação manual de componentes é demorada e suscetível a equívocos. Embora estudos recentes abordem a segmentação semântica de cenas industriais, nenhum deles trata o flange como classe-alvo isolada, lacuna específica endereçada neste trabalho. Este trabalho investiga o uso da arquitetura PointNet++ para segmentação semântica ponto a ponto de flanges em equipamentos industriais. Propõe-se uma pipeline que preserva a escala física dos dados (coordenadas em metros), calcula features geométricas com raios físicos fixos, e amostra chunks esféricos de raio variável durante o treino para capturar contexto em múltiplas escalas. A rede é treinada com uma combinação de Focal Loss e Lovász-Softmax, e utiliza GroupNorm para garantir inferência estável com batch sizes reduzidos. No conjunto de teste, o modelo final obteve acurácia global de* $90{,}41\%$ *e mIoU de* $70{,}47\%$*, com recall de* $90{,}75\%$ *e F1 de* $68{,}09\%$ *para a classe flange. Experimentos comparativos com quatro baselines de aprendizado de máquina tradicional (Random Forest, Gradient Boosting, Logistic Regression e XGBoost) demonstraram a clara superioridade da abordagem baseada em aprendizado profundo, que superou o melhor modelo clássico em mais de* 37 *pontos percentuais em mIoU. Um estudo ablativo comparando três intensidades de data augmentation revelou que a domain randomization agressiva pode prejudicar o desempenho em conjuntos pequenos e homogêneos, motivando a adoção de uma pipeline de treino mais leve. Do ponto de vista de Sistemas de Informação, os resultados demonstram a viabilidade de transformar dados brutos de escaneamento em informação semanticamente enriquecida, capaz de apoiar fluxos Scan-to-BIM, gestão de ativos e planejamento de manutenção em plantas industriais. O estudo também discute implicações organizacionais e sociotécnicas, bem como limitações relacionadas ao tamanho do dataset e ao desbalanceamento entre classes.*

Palavras-Chave. *Nuvem de pontos 3D; PointNet++; Segmentação semântica; Scan-to-BIM; Indústria 4.0; Sistemas de Informação.*

1. Introdução

A digitalização de ativos físicos e a integração entre sistemas de aquisição de dados e modelos digitais são pilares centrais da Indústria 4.0, impulsionando processos mais dinâmicos e automatizados [Jung et al. 2021]. Em setores como construção civil, petróleo e gás e manutenção industrial, nuvens de pontos 3D obtidas por sensores como *Terrestrial Laser Scanning* (TLS) tornaram-se fundamentais para representar a geometria real de equipamentos e instalações. Essas nuvens permitem criar modelos *as-built* mais fiéis, apoiar inspeções e aumentar a confiabilidade de processos de projeto e operação [Wang and Kim 2019].

No contexto de Scan-to-BIM, nuvens de pontos são combinadas a metodologias de *Building Information Modeling* (BIM) para gerar modelos digitais que agregam geometria e metadados [Valero et al. 2021]. Isso reduz o esforço manual na modelagem, facilita análises de desempenho e torna intervenções em campo mais seguras e rastreáveis. Apesar desse potencial, a interpretação manual de nuvens de pontos industriais é demorada e sujeita a equívocos, especialmente quando se trata de grandes ambientes com alta densidade de dados [Maalek et al. 2019]. Tarefas como localizar componentes específicos, por exemplo flanges em linhas de tubulação, ainda exigem trabalho intensivo de especialistas.

Sob a perspectiva de Sistemas de Informação, esse desafio não se limita a um problema de visão computacional. Trata-se também de um problema de transformação de dados brutos em informação útil para processos organizacionais, uma vez que nuvens de pontos precisam ser semanticamente enriquecidas para apoiar usos em BIM, operação e manutenção e gestão de ativos [Stojanovic et al. 2019]. Quando a identificação de componentes em nuvens de pontos depende predominantemente de inspeção manual, a atualização de modelos digitais tende a ser mais lenta, mais trabalhosa e mais suscetível a inconsistências. Assim, automatizar o reconhecimento de flanges pode contribuir para melhorar a qualidade informacional de fluxos *Scan-to-BIM*, oferecendo suporte mais ágil a atividades de inspeção de ativos, planejamento de intervenções e gestão de informação técnica em plantas industriais [Rashdi et al. 2022, Abreu et al. 2023].

Diante desse cenário, técnicas recentes de aprendizado profundo (*Deep Learning*), como a arquitetura PointNet, permitem operar diretamente sobre conjuntos de pontos sem a necessidade de convertê-los para grades voxelizadas ou imagens multivistas, reduzindo a perda de informação e etapas de pré-processamento [Qi et al. 2017a, Liu et al. 2021]. Isso abre caminho para automatizar tarefas de segmentação e detecção de objetos 3D em ambientes industriais complexos.

Embora a literatura recente registre avanços expressivos em segmentação semântica de nuvens de pontos industriais, os trabalhos disponíveis concentram-se em cenas multiclasse completas (tubos, vasos, vigas, bombas) ou na classificação de catálogos de componentes, sem tratar o flange como classe-alvo isolada em nuvens reais de plantas industriais, conforme detalhado na Seção 2.4. A lacuna é relevante porque o flange ocupa uma fração diminuta dos pontos e compartilha assinaturas geométricas locais com outros elementos salientes, ao mesmo tempo em que sua posição, tipo e dimensões condicionam a qualidade dos modelos *as-built* usados em planejamento de manutenção e verificação de conformidade normativa.

Nesse contexto, este trabalho parte da seguinte questão central: em que medida uma arquitetura PointNet++ adaptada é capaz de segmentar, ponto a ponto, flanges em nuvens de pontos industriais reais sob forte desbalanceamento entre classes? Dela derivam três desdobramentos: (i) qual parcela do desempenho é atribuível à modelagem de contexto espacial hierárquico, em contraste com classificadores ponto a ponto treinados sobre exatamente as mesmas *features*; (ii) qual o efeito da intensidade do *data augmentation* em um *dataset* pequeno e homogêneo; e (iii) que implicações organizacionais e sociotécnicas decorrem da incorporação dessa capacidade a fluxos *Scan-to-BIM*.

O objetivo principal desta pesquisa é implementar e avaliar uma arquitetura da

família PointNet++ para segmentação semântica ponto a ponto de flanges em nuvens de pontos industriais. Como contribuições específicas, apresentam-se: (a) uma *pipeline* completa que integra rotulação manual validada por especialista, pré-processamento preservando escala física real, amostragem em *chunks* esféricos multi-escala e treinamento de uma rede PointNet++ adaptada com *GroupNorm*; (b) uma comparação sistemática contra quatro *baselines* de aprendizado de máquina tradicional (*Random Forest*, *Gradient Boosting*, *Logistic Regression* e *XGBoost*) sobre o mesmo conjunto de *features*; (c) um estudo ablativo que quantifica o impacto de três intensidades de *data augmentation* sobre o desempenho final; (d) uma análise das limitações do conjunto de dados (desbalanceamento, pequena extensão geométrica dos flanges, número reduzido de equipamentos) e das implicações organizacionais e sociotécnicas da automação proposta, em linha com o escopo de Sistemas de Informação.

O restante deste artigo está organizado da seguinte forma: a Seção 2 apresenta a fundamentação teórica sobre nuvens de pontos, *Scan-to-BIM*, aprendizado profundo aplicado a dados 3D, trabalhos relacionados e o arcabouço de Sistemas de Informação adotado; a Seção 3 descreve os materiais e métodos adotados, incluindo o conjunto de dados, o pré-processamento e a arquitetura implementada; a Seção 4 apresenta os resultados experimentais; a Seção 5 apresenta as conclusões, limitações e direções para trabalhos futuros. A questão central e os desdobramentos (i) e (ii) são respondidos nas Seções 4.1, 4.2 e 4.3; o desdobramento (iii) é fundamentado na Seção 2.5 e discutido na Seção 5.

2. Fundamentação Teórica

2.1. Nuvem de Pontos

Nuvens de pontos são representações tridimensionais compostas por um conjunto denso de pontos espaciais, cada um definido por coordenadas que descrevem sua posição exata, bem como atributos adicionais, como cor ou intensidade. Durante os últimos anos, o avanço de tecnologias de sensoriamento, como scanners a laser e fotogrametria de alta precisão, impulsionou a qualidade e a versatilidade das nuvens de pontos no mapeamento e modelagem 3D [Liu et al. 2021].

De acordo com Wang e Kim [Wang and Kim 2019], essas representações permitem captar a geometria de edificações, ambientes urbanos e equipamentos industriais, fornecendo bases para análise, documentação e diagnóstico de estados físicos em áreas como engenharia, arquitetura e conservação do patrimônio histórico. A Figura 1 ilustra o escaneamento de um vaso industrial, enquanto a Figura 2 demonstra a aplicação tecnológica em uma mansão histórica.

As aplicações das nuvens de pontos abrangem diversos campos, incluindo a geração de modelos tridimensionais para realidade virtual e aumentada [Chen et al. 2020], inspeções industriais e manutenção preditiva de equipamentos [Becerik-Gerber and Rice 2010], bem como estudos arqueológicos e geográficos para a documentação e análise de sítios históricos e paisagens [Remondino and Campana 2014]. Além disso, tais dados auxiliam em avaliações estruturais, planejamento de projetos de engenharia e mapeamento de áreas urbanas, sendo frequentemente combinados com sistemas de informação geográfica e algoritmos de aprendizado de máquina para aprimorar

Figura 1. Vaso industrial em formato de nuvem de pontos.
Fonte: Autores, 2026.

Figura 2. Mansão histórica em nuvem de pontos.
Fonte: Autodesk, 2024.

análises e tomadas de decisão [Guo et al. 2016, Beraldin et al. 2010]. Mais recentemente, as nuvens de pontos têm sido adotadas como insumo central na construção de *digital twins* industriais e de manufatura, integrando técnicas de aprendizado profundo para acelerar a geração de modelos digitais de chão de fábrica [Zhao et al. 2024], enquanto *surveys* recentes consolidam o estado da arte em métodos de aprendizado geométrico aplicados a essa modalidade de dado [Saranti et al. 2024].

2.2. BIM, Scan-to-BIM e Flanges

De acordo com a Autodesk [Autodesk 2024], *Building Information Modeling* (BIM) é um processo que engloba a criação e o gerenciamento de informações de um empreendimento em construção. O BIM integra dados de diferentes disciplinas para criar uma representação digital de um objeto ao longo de todas as suas etapas, facilitando a colaboração entre as partes interessadas, desde a fase de planejamento e projeto até a construção e as operações, proporcionando uma plataforma digital unificada para o gerenciamento de informações. Portanto, a metodologia Scan-to-BIM é um processo que envolve a conversão dos dados de nuvem de pontos para modelos BIM, a fim de extrair parâmetros precisos de construção, principalmente geométricos, criando assim modelos detalhados [Badenko et al. 2019].

Conforme Wang et al. [Wang et al. 2019], esse processo possibilita a criação de modelos BIM *as-built* precisos, que são essenciais para análises de desempenho de edificações, estudos de consumo de energia, avaliações de acessibilidade e melhorias na confiabilidade estrutural.

Na Figura 3, pode-se observar a transição de um escaneamento para sua representação em modelagem 3D com alta precisão, adicionando novas informações como cores que podem representar estados, tipos e alterações dos objetos encontrados

em cena.

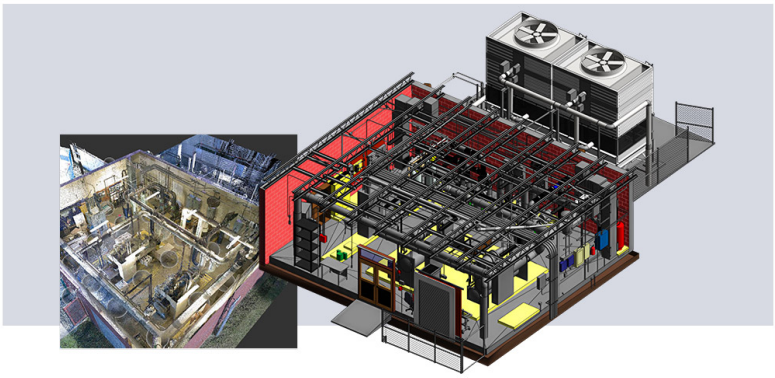

Figura 3. Representação de um escaneamento para um modelo BIM. Fonte: SCANBIM (2017).

Além disso, a integração de técnicas de visão computacional e inteligência artificial no processo Scan-to-BIM permite a automação da geração de modelos BIM com base em nuvens de pontos e fotografias de edificações, aumentando a eficiência e a precisão [Valero et al. 2021].

Em ambientes industriais, esse mesmo paradigma é aplicado a sistemas de tubulações, estruturas metálicas e equipamentos. Entre os diversos componentes presentes em instalações de dutos, destacam-se os flanges como elementos de conexão entre segmentos de tubulação, válvulas e outros dispositivos. De acordo com a norma ASME B16.5, esses flanges são componentes mecânicos circulares com furos para parafusos, padronizando dimensões, materiais e requisitos de projeto para garantir vedação e resistência mecânica nas junções [Matthews 2023]. A identificação correta da posição, do tipo e das dimensões de flanges em nuvens de pontos é, portanto, crucial para a modelagem BIM *as-built* de plantas industriais, conforme ilustrado na Figura 4.

Figura 4. Exemplos de flanges: à esquerda, flange modelado em 3D; à direita, tubulação flangeada em planta industrial. Fonte: Autores (2026) e Mistras Group (2024).

Assim, a combinação entre Scan-to-BIM e a detecção automática de flanges em nuvens de pontos permite não apenas reconstruir a geometria de tubos e conexões, mas

também enriquecer o modelo com informação semântica de alto nível sobre esses componentes. Modelos BIM *as-built* mais ricos e consistentes são particularmente relevantes para atividades de planejamento de manutenção, substituição de peças, inspeções e controle de qualidade em instalações industriais, reduzindo o esforço manual de inspeção e atualização de plantas e preparando o terreno para soluções automatizadas de monitoramento, diagnóstico e apoio à decisão técnica e operacional.

2.3. Aprendizado Profundo em Nuvens de Pontos

O Aprendizado Profundo (*Deep Learning*), impulsionado pelas Redes Neurais Convolucionais (CNNs), consolidou-se como estado da arte em tarefas de visão computacional ao longo da última década. Conforme Géron [Géron 2021], essas arquiteturas são eficazes no processamento de dados estruturados, como imagens 2D, onde a disposição regular dos pixels facilita a extração de características espaciais por meio de operações de convolução e *pooling*.

No entanto, a transição desse sucesso para o ambiente tridimensional impõe desafios significativos. Diferentemente das imagens 2D, as nuvens de pontos 3D são conjuntos de dados desordenados e não estruturados. Tentativas iniciais de aplicar CNNs nesse domínio dependiam da conversão dos dados para formatos regulares, como grades volumétricas (*voxels*) ou projeções em múltiplas vistas 2D (*multiview*). Contudo, tais abordagens introduzem artefatos de quantização, perda de detalhes geométricos finos, alto custo computacional e consumo excessivo de memória [Liu et al. 2021, Guo et al. 2021].

Para superar essas limitações, Qi et al. [Qi et al. 2017a] propuseram a PointNet, uma arquitetura pioneira projetada para processar nuvens de pontos brutas diretamente, eliminando a necessidade de voxelização. A estrutura da rede utiliza *Multi-Layer Perceptrons* (MLPs) compartilhados para aprender características de cada ponto individualmente, na prática implementados como convoluções 1D de *kernel* unitário por eficiência computacional, mas matematicamente equivalentes a um MLP aplicado ponto a ponto, seguidos por uma função simétrica (camada de *max pooling*) que agrega essas informações em um vetor de características globais, tornando o modelo invariante à ordem de entrada dos pontos. Apesar de sua importância, a PointNet original apresenta uma limitação relevante: ao aplicar *max pooling* global, ela captura apenas a estrutura geométrica geral da nuvem, sem modelar relações locais entre pontos vizinhos [Poux 2025].

Para superar essa limitação, Qi et al. [Qi et al. 2017b] propuseram a PointNet++, cuja arquitetura é ilustrada na Figura 5. A PointNet++ introduz uma hierarquia de aprendizado por meio de camadas de *Set Abstraction*. Cada camada realiza três operações: (i) *Farthest Point Sampling* (FPS), que seleciona um subconjunto de pontos centrais de forma a cobrir a nuvem espacialmente; (ii) agrupamento dos vizinhos mais próximos em torno de cada ponto central, formando vizinhanças locais; e (iii) aplicação de MLPs compartilhados sobre cada vizinhança, implementados computacionalmente por meio de convoluções 1×1, seguidos de Batch Normalization e ReLU, seguidos de *max pooling* local, produzindo descritores que capturam padrões geométricos em diferentes escalas.

Essa estrutura hierárquica permite que a rede aprenda características tanto locais (detalhes de superfície, curvatura) quanto globais (forma geral do objeto), sendo parti-

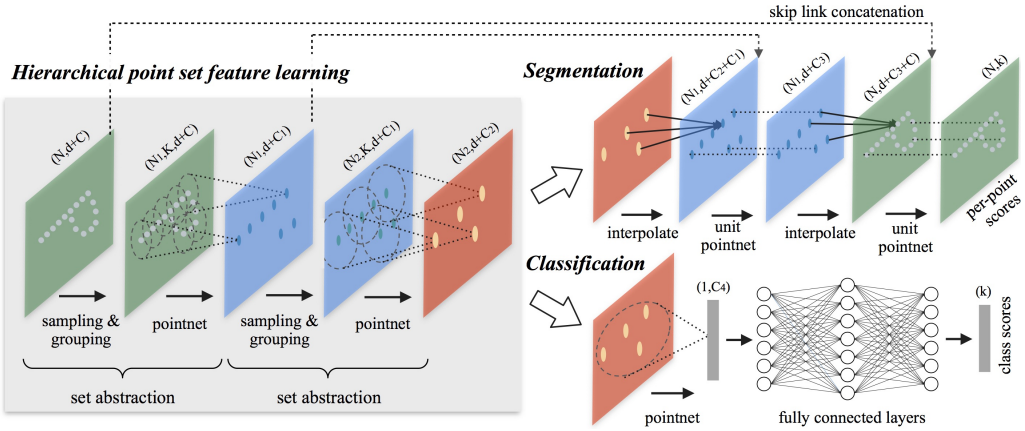

Figura 5. Arquitetura da PointNet++ aplicada à segmentação. Fonte: [Qi et al. 2017b].

cularmente adequada para cenários industriais onde componentes como flanges possuem assinaturas geométricas locais distintas, como discos com furos para parafusos, que precisam ser diferenciadas de trechos de tubulação predominantemente cilíndricos.

2.4. Trabalhos Relacionados

Os trabalhos relacionados a esta pesquisa podem ser organizados em três eixos principais: revisões de *Scan-to-BIM* e enriquecimento semântico de nuvens de pontos, métodos de aprendizado profundo para segmentação 3D, e aplicações específicas em ambientes industriais.

No contexto de *Scan-to-BIM*, Rashdi et al. [Rashdi et al. 2022] apresentam uma revisão abrangente do processo, cobrindo aquisição, registro, amostragem e segmentação semântica de nuvens de pontos. Abreu et al. [Abreu et al. 2023] discutem aplicações de modelagem procedural em *Scan-to-BIM* e *Scan-vs-BIM*, destacando limitações como oclusão e desafios de interoperabilidade. Já Stojanovic et al. [Stojanovic et al. 2019] reforçam a importância do enriquecimento semântico para a conversão de nuvens em informação útil para modelos digitais e gestão de ativos.

No eixo de métodos baseados em aprendizado profundo, Yin et al. [Yin et al. 2021] propõem a ResPointNet++, integrando aprendizado residual à arquitetura PointNet++. Utilizando operadores de agregação local e módulos *bottleneck* residuais em um *dataset* LiDAR industrial, a rede alcançou uma *Overall Accuracy* (OA) de 94% e mIoU de 87% (melhorias de 23% e 42% sobre a arquitetura tradicional). Em trabalho posterior, Yin et al. [Yin et al. 2022] apresentam a SE-PseudoGrid para classificação de componentes de tubulação. Ao substituir operadores padrão por mecanismos *Squeeze-Excite* (SE-LAO), o modelo atingiu OA de $96{,}3\%$ e acurácia média por classe de $97{,}5\%$ no *dataset Pipework*. Complementarmente, Shigeta et al. [Shigeta et al. 2023] utilizam Redes Neurais Recorrentes (RNN) baseadas na RSNet, processando a nuvem em fatias ao longo do eixo central da tubulação. Treinada com dados sintéticos aumentados e reais, a abordagem alcançou um *F-measure* de $87{,}6\%$ para flanges, valor que subiu para $89{,}8\%$ quando concatenado com características da

PointNet++.

Em contraste com essas abordagens baseadas em aprendizado, trabalhos anteriores seguiram uma linha mais geométrica e orientada a regras. Lee et al. [Lee et al. 2020] exploram a detecção por similaridade global e algoritmos ICP. Especificamente, Lee et al. realizaram o registro de conexões (*fittings*) utilizando os *port points* como referência secundária para ajustar os bicos das tubulações adjacentes ao longo de seus eixos, resolvendo qualitativamente problemas de lacunas e sobreposições nos modelos *as-built*. Mais recentemente, Noichl et al. [Noichl et al. 2024] abordaram a escassez de dados industriais gerando dados sintéticos via simulação de escaneamento a laser (Helios++) sobre modelos 3D CAD, utilizando a arquitetura KP-FCNN com 14 classes semânticas. Dados baseados em simulação superaram os baseados em amostragem em 7 pontos percentuais de mcIoU, e quando combinados com apenas 12% de dados reais anotados manualmente, o mcIoU alcançou 65%, próximo ao *benchmark* de 69% obtido com treinamento puramente em dados reais.

Do ponto de vista de avaliação comparativa de arquiteturas, Zoumpekas et al. [Zoumpekas et al. 2022] analisam cinco modelos representativos: PointNet, PointNet++, KPConv, PPNet e RSConv, nos *datasets* ShapeNet-part e S3DIS, avaliando acurácia, eficiência (tempo e memória de GPU) e desempenho sob variações nos dados. Os resultados indicaram que modelos baseados em convolução com operadores de agregação local obtiveram melhores métricas nas condições experimentais avaliadas do que os baseados em MLPs ponto a ponto, embora estes últimos apresentem maior eficiência computacional; o RSConv alcançou o melhor ImIoU de $85{,}47\%$ em segmentação de partes. Sarker et al. [Sarker et al. 2024] complementam esse panorama com uma revisão abrangente de técnicas de aprendizado profundo para segmentação semântica de nuvens de pontos 3D.

A Tabela 1 sintetiza os principais resultados quantitativos reportados pelos trabalhos discutidos nesta seção, com a finalidade de oferecer um quadro de referência ao leitor. É importante ressaltar que esses números *não* constituem uma comparação cabeça a cabeça com o presente trabalho: cada estudo utiliza um *dataset* distinto, com conjuntos de classes, escalas de ambiente e protocolos de avaliação diferentes. Em particular, os trabalhos referenciados realizam segmentação multiclasse de componentes industriais (tubos, vasos, vigas, bombas, entre outros), enquanto o presente estudo se concentra na segmentação binária da classe *flange*, que ocupa uma fração diminuta dos pontos da nuvem. Ainda assim, os valores reportados situam o desempenho obtido neste trabalho em uma faixa coerente com o estado da arte para tarefas relacionadas em nuvens de pontos industriais.

Diante do exposto, a PointNet++ original (não adaptada), reportada por Yin et al. [Yin et al. 2021] sobre seu *dataset* industrial, atinge $70{,}6\%$ de OA e $45{,}5\%$ de mIoU; arquiteturas mais recentes, como ResPointNet++ e SE-PseudoGrid, alcançam patamares superiores ao custo de modificações significativas no *backbone* (aprendizado residual e *Squeeze-Excite*, respectivamente). O desempenho obtido neste trabalho ($90{,}4\%$ de OA e $70{,}5\%$ de mIoU) situa-se em faixa intermediária a essas referências, sob a ressalva de que a tarefa abordada aqui é binária e que o *dataset* utilizado é proprietário e de menor escala.

Tabela 1. Resultados de segmentação semântica reportados na literatura em nuvens de pontos industriais.

Trabalho	Arquitetura	Dataset / Tarefa	OA (%)	mIoU (%)
Yin et al. (2021)	PointNet	Industrial LiDAR (5 cls.)	53,0	21,1
Yin et al. (2021)	PointNet++	Industrial LiDAR (5 cls.)	70,6	45,5
Yin et al. (2021)	ResPointNet++	Industrial LiDAR (5 cls.)	94,0	87,3
Yin et al. (2022)	SE-PseudoGrid	*Pipework* (cls.)	96,3	—
Shigeta et al. (2023)	RSNet	Pipe slices (F1 flange)	—	F1 87,6
Noichl et al. (2024)	KP-FCNN (real)	Cooling plant (14 cls.)	—	69,0
Noichl et al. (2024)	KP-FCNN (sint.+12% real)	Cooling plant (14 cls.)	—	65,0
Este trabalho	**PointNet++ adaptada**	**Flanges (binário)**	**90,4**	**70,5**

Embora os trabalhos revisados abordem segmentação semântica de cenas industriais completas ou classificação de catálogos de componentes, nenhum deles trata o flange como classe-alvo isolada em nuvens de pontos industriais. O presente trabalho se diferencia ao investigar especificamente essa tarefa de segmentação semântica ponto a ponto, utilizando uma arquitetura PointNet++ adaptada com *GroupNorm* para inferência estável em regime de *batch size* reduzido, combinada a uma estratégia de amostragem em *chunks* esféricos multi-escala e *loss* composta por *Focal* e *Lovász-Softmax*.

2.5. Perspectiva de Sistemas de Informação: adoção tecnológica e confiança

Para além dos aspectos computacionais, a incorporação de um módulo de segmentação automática a fluxos de trabalho reais configura um fenômeno organizacional, para cuja análise a área de Sistemas de Informação dispõe de arcabouço teórico consolidado. O *Technology Acceptance Model* (TAM) [Davis 1989] sustenta que a utilidade percebida e a facilidade de uso percebida são antecedentes críticos da intenção comportamental de uso de uma tecnologia. A *Unified Theory of Acceptance and Use of Technology* (UTAUT) [Venkatesh et al. 2003] amplia esse quadro ao tratar expectativa de desempenho, expectativa de esforço, influência social e condições facilitadoras como determinantes da intenção e do uso efetivo, moderados por fatores individuais e organizacionais.

No domínio específico de *Architecture, Engineering and Construction* (AEC), Succar [Succar 2009] caracteriza a adoção de BIM como um processo multidimensional, que envolve maturidade tecnológica, processos de trabalho e políticas organizacionais, e não apenas capacidades algorítmicas isoladas. Sob essa lente, a disponibilidade de um modelo preciso é condição necessária, porém não suficiente, para a geração de valor informacional: a incorporação efetiva a fluxos *Scan-to-BIM* depende da integração com ferramentas BIM já estabelecidas, da definição clara dos papéis entre engenheiro-validador e modelo, e da revisão dos processos de inspeção e controle de qualidade institucionalizados nas plantas industriais.

Um terceiro eixo relevante é a confiança em sistemas de inteligência artificial. A literatura sobre o tema [Glikson and Woolley 2020] indica que a aceitação operacional de saídas algorítmicas em contextos de alto risco depende fortemente da *calibração* da confiança, isto é, da capacidade do usuário de compreender quando o modelo é provavelmente correto e quando deve ser questionado. Esse arcabouço, composto por TAM, UTAUT, maturidade BIM e confiança calibrada, orienta o desdobramento (iii) e é reto-

mado na Seção 5, quando os resultados quantitativos são interpretados quanto às suas implicações organizacionais e sociotécnicas.

3. Metodologia

A metodologia proposta adota uma abordagem quantitativa experimental, estruturada em um fluxo de segmentação semântica ponto a ponto. O fluxo de trabalho, ilustrado na Figura 6, compreende as etapas de preparação e rotulação dos dados, construção do conjunto de dados, pré-processamento, treinamento da rede neural e avaliação das métricas de desempenho.

O desenvolvimento foi realizado na linguagem Python, utilizando o *framework* de aprendizado profundo **PyTorch** para a implementação e treinamento da arquitetura PointNet++. Adicionalmente, foram empregadas as bibliotecas *NumPy* para operações matriciais, *Scikit-learn* para métricas, particionamento dos dados e cálculos geométricos auxiliares (como KDTree), e *Open3D* para manipulação e processamento de estruturas tridimensionais. Os *baselines* clássicos utilizaram *XGBoost* adicionalmente. Todos os experimentos foram executados em um computador equipado com GPU NVIDIA RTX 4060 (8 GB de memória).

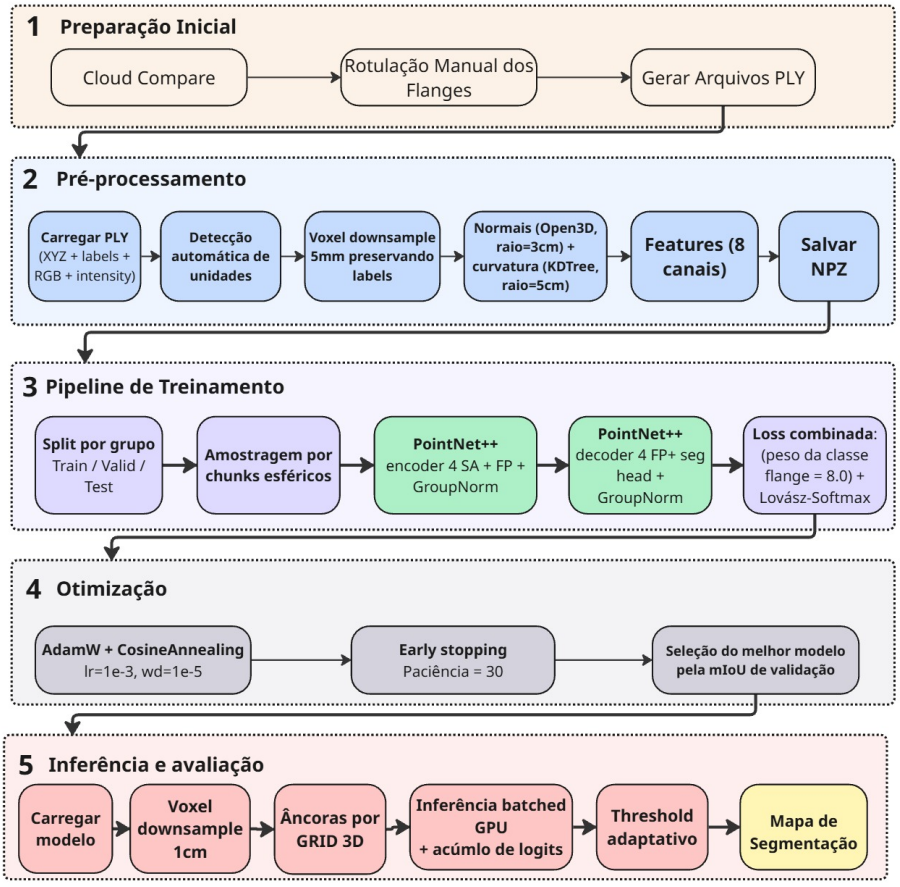

Figura 6. Fluxograma da metodologia proposta: do pré-processamento à avaliação. Fonte: Autores (2026).

3.1. Conjunto de Dados

O *dataset* utilizado neste estudo é proprietário, construído pelos autores a partir de escaneamentos reais de equipamentos industriais em formato *PLY*, previamente rotulados ponto a ponto. O conjunto é composto por 28 equipamentos distintos (vasos de pressão, trechos de tubulação, estruturas de apoio), somando aproximadamente 179 milhões de pontos antes do pré-processamento. A rotulação ponto a ponto é binária: 1 indica pertencimento à classe *flange* e 0 à classe *equipamento* (todo o restante).

Quanto ao processo de rotulação (*ground truth*), esta foi conduzida no software CloudCompare [Girardeau-Montaut 2024] por um dos autores, com base em experiência prévia em processamento de nuvens de pontos, e posteriormente validada por um engenheiro mecânico atuante na área de tubulações industriais, que revisou amostras de cada equipamento e indicou ajustes em casos de bordas ambíguas entre flange e tubulação. A Figura 7 exemplifica a rotulação em uma tubulação industrial, destacando os flanges em visão geral e em detalhe.

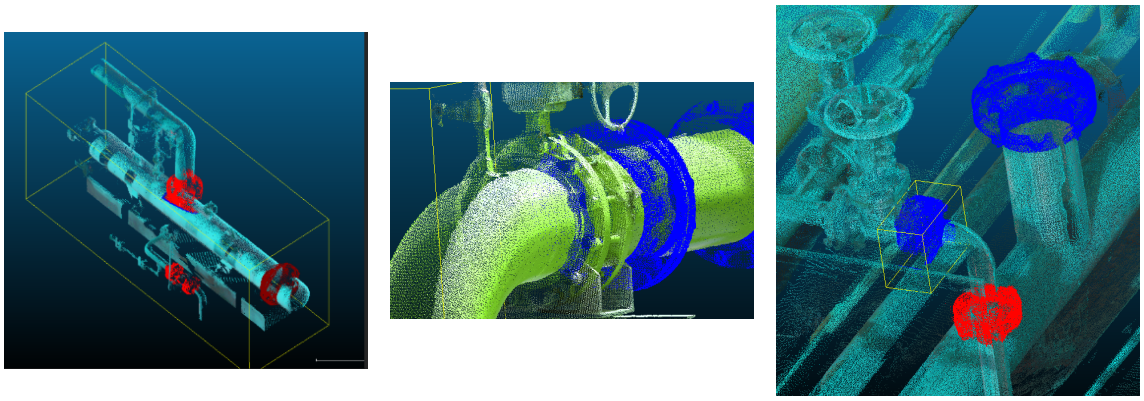

Figura 7. Processo de rotulação manual de flanges em tubulação industrial no CloudCompare: visão geral da cena (esquerda) e dois detalhes locais dos flanges (centro e direita). Fonte: Autores (2026).

Além disso, no que se refere à organização experimental, a divisão foi realizada por equipamento, e não por ponto, para evitar vazamento de informação entre conjuntos: *chunks* do mesmo equipamento podem compartilhar *features* muito correlacionadas e alocá-los em conjuntos distintos superestimaria o desempenho. Com semente fixa (SEED $=$ 42), os 28 equipamentos foram distribuídos em proporções aproximadas de 65% para treino, 15% para validação e 20% para teste, resultando em 19, 4 e 5 equipamentos, respectivamente.

Após o *voxel downsampling* de 5 mm com preservação de rótulos (Seção 3.2), o conjunto totaliza aproximadamente 69,4 milhões de pontos, dos quais 3,69% pertencem à classe *flange* e 96,31% à classe *equipamento*, o que corresponde a uma razão aproximada de 1:26 entre as classes. A Tabela 2 detalha a distribuição por conjunto. Observa-se que, embora o desbalanceamento global seja acentuado, sua intensidade varia entre os conjuntos por força do *split* ter sido conduzido por equipamento: o conjunto de treino apresenta a maior assimetria (1:30), enquanto validação e teste, por incluírem um número

menor de equipamentos com proporção relativa de flanges mais elevada, chegam a razões de 1:16 e 1:12, respectivamente. Esse forte desbalanceamento persiste mesmo após a subamostragem por *voxel* e motiva o uso da *Focal Loss* com peso $\alpha_{\text{flange}} = 8{,}0$, descrita na Seção 3.2.5.

Tabela 2. Distribuição de pontos por classe nos conjuntos de treino, validação e teste, após *voxel downsampling* de 5 mm com preservação de rótulos.

Conjunto	Equipamentos	Total de pontos	% Flange	Razão
Treino	19	59.727.318	3,20	1:30
Validação	4	4.786.189	5,92	1:16
Teste	5	4.905.077	7,43	1:12
Total	**28**	**69.418.584**	**3,69**	**1:26**

3.2. Modelo de Segmentação Semântica

3.2.1. Pré-processamento

Cada arquivo *PLY* passa por uma *pipeline* de pré-processamento *offline*, cujos produtos finais são arquivos *NPZ* contendo coordenadas, *features* geométricas e *labels*. Inicialmente, realiza-se a detecção automática de unidades por meio da inspeção da caixa delimitadora da nuvem: se qualquer dimensão exceder 100 unidades, assume-se que os dados estão em milímetros, sendo aplicada a conversão para metros por divisão por 1000; caso contrário, considera-se que a nuvem já se encontra em metros. Em seguida, aplica-se *voxel downsampling* com preservação de rótulos, utilizando um *voxel grid* de tamanho $v = 5$ mm. Para cada *voxel*, as coordenadas do ponto representativo são calculadas pela média dos pontos contidos, enquanto o rótulo é definido pela regra *or*: se qualquer ponto original do *voxel* pertencer à classe flange, o *voxel* resultante também é rotulado como flange. Essa estratégia evita a diluição da classe minoritária durante a subamostragem e contribui para a preservação de flanges pequenos.

Após essa etapa, as coordenadas x e y são centralizadas por subtração da média, enquanto a coordenada z é deslocada de modo que seu valor mínimo seja zero, mantendo-se a nuvem em metros reais, sem normalização global de escala. Essa escolha é importante para que *features* geométricas baseadas em raio físico permaneçam consistentes entre diferentes equipamentos. Na sequência, os vetores normais são estimados com a biblioteca *Open3D* por meio de busca híbrida com radius $= 3$ cm e $k_{\max} = 30$, assegurando que a vizinhança considerada corresponda a uma escala geométrica física constante, independentemente da densidade local de pontos. A curvatura aproximada de cada ponto é então calculada a partir dos autovalores $\lambda_0 \leq \lambda_1 \leq \lambda_2$ da matriz de covariância da vizinhança esférica de raio 5 cm, segundo $\kappa_i = \frac{\lambda_0}{\lambda_0+\lambda_1+\lambda_2+\varepsilon}$, com $\varepsilon = 10^{-8}$ para evitar divisão por zero. Esse descritor captura o grau de planaridade local da superfície, assumindo valores próximos de zero em regiões planas e próximos de $1/3$ em regiões isotrópicas.

Por fim, quando disponível, a intensidade é normalizada de forma robusta com base na mediana e no intervalo interquartil, isto é, $I'_i = (I_i - \text{mediana})/\text{IQR}$, reduzindo

a influência de *outliers* do sensor. Cada equipamento é então salvo em formato `.npz`, contendo coordenadas em metros, vetores normais, curvatura, RGB (ou zeros, quando ausente), intensidade normalizada (ou zeros, quando ausente) e os respectivos *labels*. Desse modo, toda a *pipeline* passa a operar sobre uma representação uniforme com 8 canais de *features* por ponto, correspondentes a 3 normais, 1 curvatura, 3 componentes RGB e 1 intensidade. Em tempo de carregamento, duas *features* adicionais são derivadas desses canais, a verticalidade ($|n_z|$) e o z-relativo, totalizando as 10 *features* por ponto efetivamente fornecidas tanto à rede quanto aos *baselines* clássicos (Seção 3.3).

3.2.2. Amostragem por *chunks* esféricos multi-escala

Processar diretamente um equipamento industrial completo é inviável em termos de memória de GPU. Por essa razão, durante o treinamento, cada amostra é definida como um *chunk* esférico extraído da nuvem de pontos. Inicialmente, sorteia-se um raio $r \in \{0{,}5;\ 1{,}0;\ 2{,}0\}$ metros, de modo a expor a rede a diferentes escalas de contexto, variando desde o detalhe de um único flange até regiões contendo múltiplos componentes. Em seguida, seleciona-se um ponto-âncora c no equipamento: com probabilidade $0{,}5$, esse ponto é escolhido aleatoriamente em toda a nuvem e, com probabilidade $0{,}5$, é escolhido entre os pontos pertencentes à classe flange. Essa estratégia de amostragem balanceada aumenta a exposição do modelo a *chunks* contendo flanges, sem impor uma razão artificial fixa entre as classes.

Uma vez definido o ponto-âncora, todos os pontos cuja distância euclidiana a c seja inferior a r são reunidos para compor o *chunk*. A partir desse conjunto, são subamostrados exatamente $N = 8192$ pontos, com reposição quando necessário, de forma a produzir uma entrada de tamanho fixo para a rede. Por fim, as coordenadas do *chunk* são centralizadas em relação a c e divididas por r, de modo que a entrada do modelo passe a ocupar sempre uma esfera unitária. Com isso, a escala física original é indiretamente preservada pelo valor de r, enquanto a rede recebe uma representação espacial normalizada.

Na validação e no teste, o procedimento de amostragem torna-se determinístico, com raio fixo de $r = 1{,}0$ m, âncora escolhida por gerador pseudoaleatório com semente fixa e ausência de transformações adicionais.

3.2.3. Aumento de Dados

As transformações de aumento de dados, cuja aplicação a nuvens de pontos é revisada em detalhe por Zhu et al. [Zhu et al. 2023], são aplicadas ***online***, ou seja, a cada *chunk* sorteado durante o treino, e nunca geram amostras persistidas em disco. Isso implica que 100% dos *chunks* de treino passam por alguma combinação aleatória das transformações, e o número efetivo de variantes únicas observadas pelo modelo ao longo de N épocas é aproximadamente $N \times |D|$, onde $|D|$ é o número de *chunks* sorteados por época.

Para investigar o impacto do *augmentation*, foram definidas três configurações distintas, descritas abaixo e comparadas empiricamente na Seção 4.3:

Configuração *No-Aug*. Sem nenhuma transformação. As *features* são usadas como produzidas pela amostragem de *chunks* (Seção 3.2.2).

Configuração *Light*. Apenas transformações fisicamente realistas, de baixa intensidade:

- *Rotação em torno do eixo z*: $x_i^{(r)} = R_z(\theta)\, x_i$, com $\theta \sim \mathcal{U}(0, 2\pi)$. Aplicada a coordenadas e normais; estas últimas são renormalizadas.
- *Ruído gaussiano (jittering)*: $x_i^{(j)} = x_i^{(r)} + \varepsilon_i$, com $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$, $\sigma = 0{,}003$ (no espaço normalizado do *chunk*, correspondendo a aproximadamente 3 mm no espaço físico para o raio $r = 1{,}0$ m).
- *Feature dropout* em RGB com probabilidade $0{,}3$: os três canais RGB são zerados, simulando cenários de escaneamento monocromático.

Configuração *Heavy*. *Pipeline* completa de *domain randomization*, combinando as transformações da configuração *Light* com: rotação pequena em torno dos eixos x e y ($\pm 8{,}6°$), escala anisotrópica por eixo ($\mathcal{U}(0{,}85;\ 1{,}15)$), *jitter* variável ($\sigma \in [0{,}002;\ 0{,}015]$), *outliers* esparsos ($0{,}5\%$–$2\%$ dos pontos), *point dropout* de até 50% dos pontos, oclusão de octante (probabilidade $0{,}3$ de zerar um oitavo aleatório do *chunk*) e *feature dropout* independente para RGB, intensidade e curvatura.

Todas as equações de transformação aplicam-se igualmente às coordenadas e, onde relevante, aos vetores normais (que são renormalizados após rotações).

3.2.4. Arquitetura

A arquitetura de segmentação segue o paradigma *encoder-decoder* da PointNet++ [Qi et al. 2017b], com uma modificação relevante: todas as camadas de normalização utilizam *GroupNorm* [Wu and He 2018] com $G = 16$ grupos, em vez de *Batch Normalization*. Essa escolha é motivada pelo *batch size* reduzido ($B = 4$): a *Batch Normalization* depende de estatísticas calculadas sobre o *mini-batch*, que se tornam instáveis com poucos exemplos, comprometendo a inferência em modo de avaliação. A *GroupNorm* normaliza dentro de cada amostra individualmente, eliminando essa dependência.

O *encoder* é composto por quatro blocos de *Set Abstraction* (SA) que realizam subamostragem progressiva da nuvem de pontos; o *decoder* utiliza quatro módulos de *Feature Propagation* (FP) que interpolam as características de volta às posições originais por meio de *skip connections*. A Tabela 3 apresenta a configuração de cada bloco.

O *decoder* interpola as *features* dos níveis mais abstratos de volta às posições originais dos pontos, com MLPs de dimensões $[512, 512] \to [256, 256] \to [128, 128] \to [128, 128]$ (blocos FP4 a FP1). O *head* de segmentação é composto por `Conv1D` $(128 \to 128)$ + *GroupNorm* + ReLU + *Dropout* $(0{,}5)$ + `Conv1D` $(128 \to 2)$, produzindo, para cada ponto, um vetor de pontuações (*logits*) para as classes “equipamento” e “flange”.

Tabela 3. Configuração dos blocos do *encoder* do modelo de segmentação.

Bloco	Pontos	Vizinhos	MLP
SA1	$8.192 \rightarrow 2.048$	32	[64, 64, 128]
SA2	$2.048 \rightarrow 512$	64	[128, 128, 256]
SA3	$512 \rightarrow 128$	128	[256, 256, 512]
SA4	$128 \rightarrow 1$ (global)	—	[512, 512, 1024]

3.2.5. Treinamento

O treinamento do modelo foi conduzido com uma função de perda composta e com estratégias explícitas de regularização, de modo a lidar com o forte desbalanceamento entre as classes e favorecer a generalização.

Nesse contexto, a função de perda combina dois termos complementares, adequados a problemas de segmentação binária desbalanceada. O primeiro termo é a *Focal Loss*, que penaliza mais fortemente exemplos difíceis e mitiga o efeito do desbalanceamento [Lin et al. 2018]:

$$\mathcal{L}_{\text{focal}} = -\frac{1}{|M|} \sum_{i \in M} \alpha_{y_i} \, (1 - p_{y_i})^{\gamma} \, \log p_{y_i} \tag{1}$$

onde p_{y_i} é a probabilidade atribuída pela rede à classe verdadeira do ponto i, $\gamma = 2$ é o parâmetro de *focusing* e $\boldsymbol{\alpha} = [1{,}0;\ 8{,}0]$ é o vetor de pesos por classe, com maior peso para a classe flange.

Além disso, emprega-se a *Lovász-Softmax* [Berman et al. 2018], que otimiza diretamente uma aproximação diferenciável da *Intersection-over-Union* (IoU), sendo mais adequada a métricas baseadas em sobreposição do que a *Dice Loss* tradicional [Milletari et al. 2016]:

$$\mathcal{L}_{\text{lov}} = \frac{1}{|C|} \sum_{c \in C} \Delta_{J_c}(\mathbf{m}_c) \tag{2}$$

onde Δ_{J_c} é a extensão de Lovász do complemento da IoU da classe c e $\mathbf{m}_c$ é o vetor de erros do *softmax*. Em conjunto, a perda total é definida como:

$$\mathcal{L} = \mathcal{L}_{\text{focal}} + \mathcal{L}_{\text{lov}} \tag{3}$$

Essa combinação é menos sensível ao desbalanceamento do que a *cross-entropy* pura e alinha explicitamente o objetivo de treinamento com a métrica de avaliação adotada (mIoU). Cabe observar que o peso $\alpha_{\text{flange}} = 8{,}0$ foi definido empiricamente durante a experimentação preliminar e não corresponde estritamente à razão de classes observada no conjunto de treino ($1{:}30$); valores mais agressivos foram avaliados, mas tenderam a aumentar a quantidade de falsos positivos sem ganho proporcional de *recall*, levando à escolha de um valor intermediário entre o peso unitário e o sugerido pela razão de classes.

Quanto à configuração experimental, a Tabela 4 sintetiza os principais hiperparâmetros utilizados no treinamento do modelo de segmentação.

Adicionalmente, foram adotadas múltiplas estratégias para reduzir o sobreajuste. Além do *dropout* $(0{,}5)$ aplicado na cabeça de segmentação, a *pipeline* emprega: *weight*

Tabela 4. Hiperparâmetros do modelo de segmentação.

Parâmetro	Valor
Pontos por amostra (N)	8.192
Batch size	4
Épocas (máximo)	150
Chunks por época (treino)	300
Chunks por época (validação)	80
Learning rate inicial	1×10^{-3}
Weight decay	1×10^{-5}
Otimizador	AdamW
Scheduler	*CosineAnnealingLR* ($\eta_{\min} = 10^{-6}$)
Função de perda	*Focal* + *Lovász-Softmax*
Peso da classe flange (α_1)	8,0
Parâmetro γ da *Focal*	2
Dropout	0,5
Normalização	*GroupNorm* ($G = 16$)
Gradient clipping (norma)	1,0
Early stopping (paciência)	30 épocas
Semente	42

decay de 1×10^{-5} no otimizador AdamW; *GroupNorm* em todas as camadas convolucionais; *gradient clipping* global por norma ($\|\nabla\|_2 \leq 1$); *early stopping* baseado no mIoU de validação, com paciência de 30 épocas; *CosineAnnealingLR* para reduzir progressivamente a *learning rate*; e *split* por equipamento, e não por ponto ou por *chunk*, evitando vazamento de informação entre os conjuntos de treino, validação e teste.

Cabe observar que a arquitetura herda da PointNet++ [Qi et al. 2017b] a estrutura hierárquica de *Set Abstraction* e *Feature Propagation*, bem como as MLPs compartilhadas implementadas como convoluções 1×1 aplicadas ponto a ponto, que constituem a operação fundamental herdada da PointNet original [Qi et al. 2017a]. As T-Nets de transformação de entrada e de *features* presentes na PointNet original não são utilizadas, seguindo a prática da PointNet++, que delega a captura de variações geométricas locais aos módulos de *Set Abstraction*.

3.2.6. Inferência

Na etapa de inferência, a nuvem completa de um equipamento, composta por centenas de milhares a milhões de pontos, é processada por meio de um esquema de cobertura baseado em um *grid* 3D. Inicialmente, aplica-se um *voxel downsampling* leve de 1 cm, com o objetivo de reduzir a densidade da nuvem sem comprometer a geometria dos flanges. Em seguida, as *features* geométricas, tais como vetores normais e curvatura, são recalculadas utilizando os mesmos raios físicos empregados no pré-processamento, de forma a manter consistência entre treinamento e inferência.

Na sequência, gera-se um *grid* 3D de âncoras com espaçamento $s = 0{,}4$ m. Como o raio adotado para cada *chunk* é $r = 1{,}0$ m, obtém-se uma sobreposição de aproximadamente 60% entre *chunks* vizinhos, o que assegura cobertura próxima de 100% dos pontos da nuvem. Âncoras cuja região de influência contém menos de 200 pontos são

descartadas. Para cada âncora válida, extrai-se o *chunk* correspondente, conforme descrito na Seção 3.2.2, e os *logits* produzidos pelo modelo são acumulados sobre os pontos da nuvem original.

Posteriormente, a probabilidade final de cada ponto é obtida pela média dos *logits* acumulados, seguida da aplicação da função *softmax*. A decisão binária é então realizada por meio de um *threshold* adaptativo, definido por $t = \max(0{,}35;\ 0{,}7 \cdot p_{\max})$, em que $p_{\max}$ denota a maior probabilidade observada na nuvem. Essa estratégia busca evitar as limitações de limiares fixos em cenários nos quais a calibração do modelo pode variar entre equipamentos distintos. Por fim, a predição obtida sobre a nuvem subamostrada é propagada de volta aos pontos originais por meio da busca do vizinho mais próximo.

3.3. *Baselines* de aprendizado de máquina tradicional

Para avaliar a competitividade da PointNet++ frente a abordagens mais simples, foram implementados quatro *baselines* clássicos com classificação ponto a ponto: *Random Forest*, com 300 árvores, profundidade máxima 20 e `class_weight="balanced"`; *Gradient Boosting*, com 100 estimadores, profundidade 5 e *learning rate* de $0{,}1$; *Logistic Regression*, com regularização ℓ_2, balanceamento entre classes e padronização das *features* por meio de *StandardScaler*; e *XGBoost*, com 300 estimadores, profundidade 8, `scale_pos_weight` ajustado pela razão entre classes e `tree_method=hist`.

Para garantir uma comparação justa, os quatro classificadores foram alimentados com exatamente as mesmas 10 *features* por ponto utilizadas na entrada da PointNet++: vetores normais (n_x, n_y, n_z), verticalidade ($|n_z|$), curvatura, componentes RGB, intensidade e z-relativo, definido como a altura normalizada em relação ao ponto mais alto do equipamento. Assim, a diferença de desempenho entre os modelos reflete principalmente a contribuição da arquitetura, isto é, o uso de contexto espacial hierárquico na PointNet++ em contraste com a classificação ponto a ponto dos métodos tradicionais, e não uma diferença nas *features* de entrada.

Além disso, o *split* por equipamento foi mantido idêntico ao utilizado no treinamento da PointNet++. Para viabilizar o tempo de treinamento dos classificadores clássicos, aplicou-se um limite de 800.000 pontos na construção do conjunto de treino, com balanceamento $50/50$ entre as classes.

4. Resultados

4.1. Segmentação Semântica

A Figura 8 apresenta a evolução das métricas de treinamento e validação ao longo das épocas para a configuração final adotada (sem *augmentation*, conforme justificado na Seção 4.3). Observa-se que a perda de treinamento decresce de forma gradual e relativamente estável ao longo do treinamento. Já a perda de validação apresenta oscilações mais acentuadas nas épocas iniciais, mas com tendência global de queda, estabilizando-se em valores menores ao final do processo. De maneira semelhante, a *Overall Accuracy* (OA) e a *mean Intersection-over-Union* (mIoU) de validação exibem alta variabilidade no início do treinamento, seguida de crescimento progressivo e posterior estabilização. O melhor valor de mIoU de validação é observado em torno da época 115.

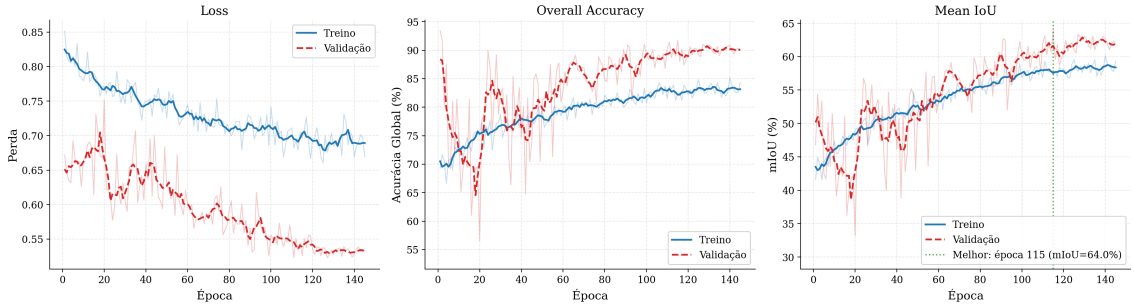

Figura 8. Curvas de treinamento e validação do modelo de segmentação semântica: *loss*, *Overall Accuracy* (OA) e *mean Intersection-over-Union* (mIoU). Fonte: Autores (2026).

Nota-se que, ao longo de grande parte do treinamento, as métricas de validação permanecem ligeiramente superiores às de treinamento, especialmente em OA e mIoU, enquanto a perda de validação se mantém inferior à perda de treino. Esse comportamento não caracteriza o padrão clássico de sobreajuste; em vez disso, sugere um regime de treinamento mais desafiador, possivelmente influenciado pelos mecanismos de regularização adotados e pela própria estratégia de amostragem dos *chunks*. Ainda assim, a estabilidade global das curvas e a evolução consistente da mIoU de validação indicam que o modelo foi capaz de aprender padrões geométricos relevantes para a identificação dos flanges. A substituição de *Batch Normalization* por *GroupNorm*, aliada ao *early stopping* e ao *dropout*, contribuiu para tornar o treinamento estável mesmo no regime de *batch size* reduzido.

No conjunto de teste, o modelo final obteve os resultados apresentados na Tabela 5. A *Overall Accuracy* (OA) foi de $90{,}41\%$, enquanto a *mean Intersection-over-Union* (mIoU) atingiu $70{,}47\%$. Para a classe flange, observou-se *recall* de $90{,}75\%$, indicando elevada sensibilidade na detecção dos pontos pertencentes a esse componente. Por outro lado, a precisão de $54{,}49\%$ foi inferior ao *recall*, o que sugere a presença de falsos positivos em regiões geometricamente ambíguas, especialmente nas transições entre flange e tubulação. Ainda assim, o valor de $F1 = 68{,}09\%$ indica um compromisso razoável entre sensibilidade e precisão, compatível com a dificuldade do problema e o forte desbalanceamento entre as classes, o que responde à questão central.

Tabela 5. Métricas do modelo de segmentação no conjunto de teste.

Métrica	**Valor (%)**
Overall Accuracy (OA)	90,41
mean IoU (mIoU)	70,47
$\text{IoU}_{\text{equipamento}}$	89,31
$\text{IoU}_{\text{flange}}$	51,62
$\text{Precision}_{\text{flange}}$	54,49
$\text{Recall}_{\text{flange}}$	90,75
$\text{F1}_{\text{flange}}$	68,09

A Figura 9 ilustra exemplos qualitativos de segmentação em três equipamentos

de teste, organizados em subfiguras (a), (b) e (c). Cada linha mostra, da esquerda para a direita, o *ground truth*, a predição do modelo e o respectivo mapa de erros, no qual os pontos em cinza correspondem a classificações corretas e os pontos em amarelo destacam falsos positivos e falsos negativos. Em (a), um trecho horizontal de tubulação com diversos flanges distribuídos ao longo da estrutura, todos os flanges principais são recuperados e os erros se concentram nas bordas entre flange e tubulação, padrão consistente com o *recall* elevado e a precisão moderada da Tabela 5. Em (b), uma tubulação curva com múltiplos flanges contíguos de diâmetros distintos, os erros aparecem nas separações entre flanges adjacentes, e não em confusões com a tubulação, comportamento coerente com o raio das vizinhanças locais agregadas pelos módulos de *Set Abstraction*. Em (c), um cenário com flanges isolados em tubulações curvas, observa-se o melhor desempenho qualitativo, com mapa de erros predominantemente cinza e falsos positivos esparsos no entorno. No conjunto, os erros do modelo são sistemáticos e localizados em regiões de transição geométrica ambígua, sustentando a leitura quantitativa das métricas e o uso da rede como uma primeira passada, com a inspeção humana concentrada nas regiões indicadas pelo mapa de erros.

4.2. Comparação com *baselines* de aprendizado de máquina tradicional

Para situar o desempenho da PointNet++ em relação a abordagens mais simples baseadas estritamente nas mesmas *features* ponto a ponto, conduzimos uma comparação sistemática com quatro classificadores clássicos. A Figura 10 compara a PointNet++ com quatro *baselines* clássicos avaliados sobre o mesmo *split* por equipamento e o mesmo conjunto de *features* por ponto descrito na Seção 3.3, sendo a métrica da PointNet++ a da *pipeline* final, com *threshold* adaptativo e agregação por *grid* 3D. A PointNet++ supera todos os *baselines* em todas as métricas: atinge $70{,}47\%$ de mIoU contra $33{,}5\%$ do melhor classificador clássico (*XGBoost*), um ganho de aproximadamente 37 pontos percentuais, e o F1 da classe flange passa de $20{,}9\%$ para $68{,}09\%$, ganho de cerca de 47 pontos percentuais.

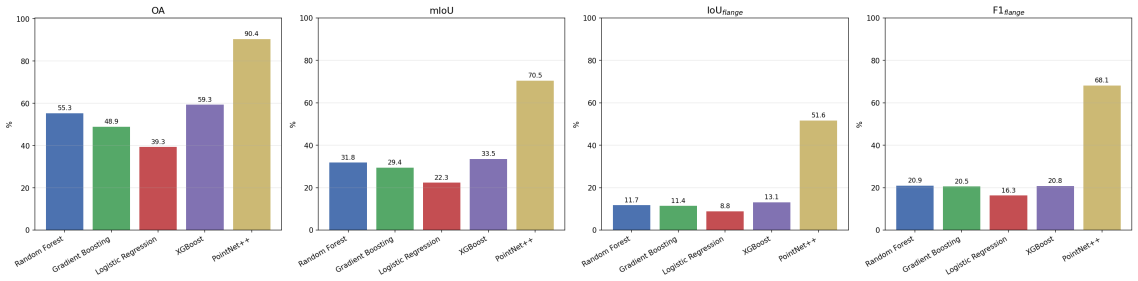

Figura 10. Comparação entre a PointNet++ e quatro *baselines* de aprendizado de máquina tradicional em *Overall Accuracy*, mIoU, IoU da classe flange e F1 da classe flange. Fonte: Autores (2026).

Esses resultados indicam que *features* ponto a ponto isoladas não são suficientes para distinguir flanges de outras estruturas geometricamente semelhantes, como ressaltos em vasos, suportes e conexões pequenas, que compartilham assinaturas locais de normais e curvatura próximas às de flanges reais. A modelagem do contexto espacial

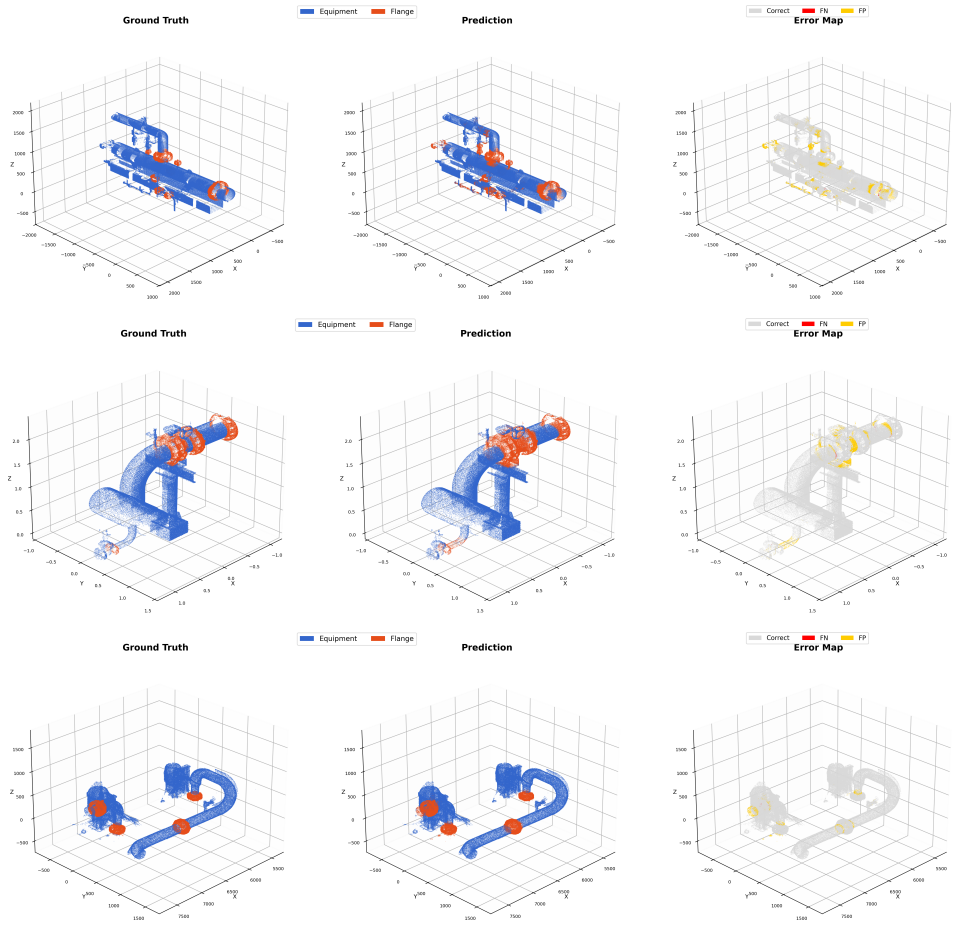

Figura 9. Resultados de segmentação semântica em três equipamentos de teste: (a) flanges bem espaçados ao longo de tubulação horizontal; (b) múltiplos flanges de diâmetros variados próximos entre si em tubulação curva; (c) flanges isolados em tubulações curvas. Da esquerda para a direita em cada linha: *ground truth*, predição do modelo e mapa de erros. Pontos em azul representam equipamento, pontos em vermelho representam flange e pontos em amarelo (mapa de erros) indicam classificações incorretas. Fonte: Autores (2026).

hierárquico dos módulos de *Set Abstraction* da PointNet++, ao agrupar pontos vizinhos em múltiplas escalas, captura o arranjo geométrico característico de um flange (anel perpendicular à tubulação com furos regulares), informação que classificadores ponto a ponto, por construção, não conseguem representar, o que responde ao desdobramento (i).

4.3. Estudo ablativo: aumento de dados

A Tabela 6 apresenta os resultados do estudo ablativo sobre o impacto do *data augmentation* na segmentação. Os modelos são idênticos em todos os aspectos (arquitetura, *loss*, otimizador, *scheduler*, sementes, *split*), variando apenas a configuração de aumento de dados (*No-Aug*, *Light*, *Heavy*, conforme Seção 3.2.3). Cada execução utilizou 60 épocas, com *early stopping* de paciência 15, e as três foram avaliadas sobre o mesmo conjunto de *chunks* de teste. Devido ao custo computacional do treinamento completo em GPU,

cada configuração foi executada com uma única semente fixa (42), o que limita a precisão estatística das comparações absolutas, mas não invalida a tendência qualitativa observada entre as três configurações.

Pode-se observar que a configuração *No-Aug* obteve o melhor desempenho geral, com 90,41% de OA, 70,47% de mIoU, 51,62% de IoU da classe flange e 68,09% de F1 da classe flange. A configuração *Light* apresentou desempenho intermediário, enquanto a configuração *Heavy* teve os piores resultados em mIoU e F1, embora tenha alcançado o maior *recall* da classe flange.

Esse comportamento indica que o aumento de dados mais agressivo elevou a sensibilidade do modelo, mas também aumentou a quantidade de falsos positivos. Como consequência, houve queda de precisão e piora das métricas de sobreposição. No cenário avaliado, em que treino e teste seguem a mesma distribuição de captura, o *augmentation* introduziu variabilidade adicional sem produzir ganho efetivo de generalização, o que responde ao desdobramento (ii).

Tabela 6. Estudo ablativo sobre *data augmentation* na segmentação. Cada linha corresponde a uma *pipeline* de aumento diferente, com todas as demais variáveis mantidas constantes (mesma arquitetura, *loss*, otimizador, *split* e semente 42).

Configuração	OA (%)	mIoU (%)	$\text{IoU}_{\text{flange}}$ (%)	$\text{Recall}_{\text{flange}}$ (%)	$\text{F1}_{\text{flange}}$ (%)
No-Aug	**90,41**	**70,47**	**51,62**	90,75	**68,09**
Light	88,03	66,13	45,57	88,81	62,61
Heavy	83,74	60,42	38,98	**92,31**	56,10

Com base nesses resultados, a configuração *No-Aug* foi adotada na *pipeline* final.

5. Conclusão

Este trabalho investigou o uso de uma arquitetura PointNet++ adaptada para segmentação semântica ponto a ponto de flanges em nuvens de pontos industriais, alinhado ao cenário da Indústria 4.0 e ao paradigma *Scan-to-BIM*. Foi proposta uma *pipeline* completa que integra rotulação manual validada, pré-processamento preservando escala física real, cálculo de *features* geométricas por raio físico, amostragem em *chunks* esféricos multi-escala e treinamento com *loss* combinada *Focal + Lovász-Softmax*.

No conjunto de teste, o modelo obteve *Overall Accuracy* de 90,41% e mIoU de 70,47%, com IoU de 51,62% e F1 de 68,09% para a classe flange. O *recall* elevado (90,75%) indica que o modelo detecta a grande maioria dos pontos pertencentes aos flanges, enquanto a precisão de 54,49% reflete falsos positivos concentrados nas regiões de transição entre flange e tubulação, onde a geometria local é ambígua. A substituição de *Batch Normalization* por *GroupNorm*, aliada a *dropout*, *weight decay*, *gradient clipping* e *early stopping*, estabilizou o treinamento no regime de *batch size* reduzido imposto pela memória de GPU disponível.

A comparação com quatro *baselines* de aprendizado de máquina tradicional (*Random Forest*, *Gradient Boosting*, *Logistic Regression* e *XGBoost*), treinados exatamente

sobre as mesmas *features* por ponto, evidenciou uma superioridade expressiva da abordagem baseada em aprendizado profundo, com ganho de 37 pontos percentuais em mIoU e 47 pontos em F1 da classe flange sobre o melhor classificador clássico. Essa diferença confirma que a detecção de flanges é fundamentalmente uma tarefa *contextual*: a assinatura semântica do componente não está em um único ponto, mas no arranjo espacial característico de um anel perpendicular à tubulação com furos regulares para parafusos. Modelar esse contexto requer uma arquitetura capaz de agrupar informação local em múltiplas escalas, capacidade oferecida pelos módulos *Set Abstraction* da PointNet++ e ausente em classificadores ponto a ponto.

O estudo ablativo sobre *data augmentation*, cobrindo três intensidades (*No-Aug*, *Light*, *Heavy*), produziu um achado não trivial: em cenário de teste homogêneo com o treino, o *augmentation* agressivo degrada o desempenho, e o melhor resultado foi obtido sem nenhuma transformação. Essa observação matiza a prática comum de adotar *augmentation* pesada por padrão e sugere que sua intensidade deve ser calibrada conforme o cenário de deslocamento de domínio esperado em produção: para *datasets* pequenos e homogêneos, *augmentation* leve ou nulo pode ser preferível; para cenários *cross-domain*, a *domain randomization* tende a ser crítica.

Do ponto de vista de Sistemas de Informação, os resultados evidenciam o potencial de transformar dados brutos de escaneamento em informação semanticamente enriquecida, capaz de alimentar fluxos organizacionais de modelagem *as-built*, planejamento de manutenção preventiva, gestão de ativos (*Enterprise Asset Management*) e verificação de conformidade normativa (por exemplo, ASME B16.5). Em fluxos atuais, a identificação manual de flanges em equipamentos de porte médio é uma atividade reconhecidamente custosa em termos de tempo de especialista, podendo consumir várias horas por equipamento; a inferência automatizada em poucos minutos desloca esse esforço para uma etapa de *validação* e correção pontual, alterando o papel do engenheiro ou técnico de modelagem de rotulador exaustivo para *validador* crítico, com sua expertise concentrada em casos de exceção (flanges pequenos, oclusões, componentes atípicos). Lidos à luz do arcabouço da Seção 2.5, contudo, esses ganhos não decorrem da capacidade algorítmica isolada: na chave do TAM e da UTAUT, a utilidade percebida do módulo depende de quanto ele reduz efetivamente o tempo de especialista, e sua facilidade de uso depende da integração com ferramentas BIM já institucionalizadas; na chave da maturidade BIM proposta por Succar, a adoção exige revisão de processos e definição de papéis, e não apenas a entrega de um modelo treinado. Além disso, o regime de *recall* elevado e *precision* moderada (Tabela 5) torna a confiança calibrada um requisito de projeto: em contextos de alto risco, como a verificação de componentes pressurizados, o uso adequado do modelo é o de *first-pass* acompanhado da exposição da incerteza por ponto, e não o de classificador autônomo. Trata-se, portanto, de implicações potenciais, cuja confirmação exige avaliação empírica em contexto organizacional real.

Algumas limitações devem ser reconhecidas. O conjunto de dados, composto por 28 equipamentos proprietários, é restrito em volume e diversidade, insuficiente para cobrir a variedade de configurações industriais reais (vasos, trocadores, colunas, *skids*) e proveniente de um conjunto homogêneo de sensores. Mesmo após *voxel downsampling* com preservação de *labels*, o desbalanceamento entre classes persiste e contribui para o

gap entre *recall* e *precision*. Flanges pequenos (como drenos *2"*) ocupam muito poucos pontos, tornando sua detecção inerentemente mais difícil.

Tendo em vista a limitação de tempo de GPU, os experimentos principais foram conduzidos com uma única semente fixa (SEED $= 42$), o que impede o reporte de média e desvio-padrão e introduz uma fonte de incerteza estatística sobre os números reportados. Em particular, não é possível afirmar que o resultado de $70{,}47\%$ de mIoU obtido neste estudo corresponda à média esperada do procedimento de treinamento proposto, e o intervalo de confiança real em torno desse valor não pode ser estimado a partir dos dados disponíveis. Resultados em trabalhos correlatos sobre nuvens de pontos industriais sugerem variações típicas entre sementes da ordem de 1 a 3 pontos percentuais em métricas baseadas em sobreposição. Recomenda-se, em trabalhos futuros, a repetição do procedimento com pelo menos três sementes distintas, de modo a permitir uma comparação com maior suporte estatístico entre configurações.

Como perspectivas de trabalhos futuros, três direções são consideradas prioritárias: (a) a ampliação e diversificação do *dataset*, incluindo novos equipamentos, sensores e condições de aquisição, bem como estratégias semissupervisionadas ou de aprendizado ativo para reduzir o custo de rotulação, endereçando diretamente a principal limitação de generalização identificada; (b) a repetição dos experimentos com múltiplas sementes, reportando média e desvio-padrão, de modo a conferir suporte estatístico às comparações entre configurações; e (c) a avaliação empírica em contextos organizacionais reais, medindo adoção tecnológica, confiança dos usuários e ganhos efetivos em fluxos de trabalho de inspeção e manutenção, o que permitiria converter as implicações potenciais discutidas acima em evidência sobre o desdobramento (iii). Secundariamente, permanecem em aberto a comparação com arquiteturas mais recentes (DGCNN, KPConv e redes baseadas em *Transformers*), a investigação de funções de perda mais sensíveis a bordas finas entre componentes, a integração do módulo de segmentação a ferramentas BIM comerciais e a reavaliação do *augmentation* em cenários de teste *cross-domain*.

Em síntese, o estudo demonstrou a viabilidade de aplicar uma rede PointNet++ adaptada a nuvens de pontos industriais para segmentação semântica de flanges, superando abordagens tradicionais baseadas em *features* geométricas por margens expressivas, e analisou criticamente o papel do *data augmentation* no contexto de *datasets* pequenos. Os resultados abrem caminho para pesquisas que busquem tornar tais soluções mais competitivas e escaláveis, consolidando o papel do aprendizado profundo como ferramenta de apoio à gestão de informação técnica em plantas industriais.

Referências

[Abreu et al. 2023] Abreu, N., Pinto, A., Matos, A., and Pires, M. (2023). Procedural point cloud modelling in Scan-to-BIM and Scan-vs-BIM applications: a review. *ISPRS International Journal of Geo-Information*, 12(7):260.

[Autodesk 2024] Autodesk (2024). Building information modeling (BIM). `https://www.autodesk.com.br/solutions/bim`. Acesso em: 23 maio 2025.

[Badenko et al. 2019] Badenko, V. et al. (2019). Scan-to-BIM methodology adapted for different applications. *ISPRS - International Archives of the Photogrammetry Remote*

Sensing and Spatial Information Sciences, XLII-5/W2:1–7.

[Becerik-Gerber and Rice 2010] Becerik-Gerber, B. and Rice, S. (2010). The perceived value of building information modeling in the U.S. building industry. *Journal of Information Technology in Construction (ITcon)*, 15:185–201.

[Beraldin et al. 2010] Beraldin, J. A. et al. (2010). Advances in 3D imaging and modeling: acquisition, processing, and visualization. *Remote Sensing*, 2(8):2334–2388.

[Berman et al. 2018] Berman, M., Triki, A. R., and Blaschko, M. B. (2018). The Lovász-Softmax loss: a tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4413–4421.

[Chen et al. 2020] Chen, J. et al. (2020). Integration of building information modeling and 3D point cloud data for augmented reality-based facility management. *Automation in Construction*, 119:103308.

[Davis 1989] Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3):319–340.

[Géron 2021] Géron, A. (2021). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*. O'Reilly Media, 3 edition.

[Girardeau-Montaut 2024] Girardeau-Montaut, D. (2024). CloudCompare – open-source project. `https://www.danielgm.net/cc/`. Acesso em: 20 set. 2025.

[Glikson and Woolley 2020] Glikson, E. and Woolley, A. W. (2020). Human trust in artificial intelligence: review of empirical research. *Academy of Management Annals*, 14(2):627–660.

[Guo et al. 2016] Guo, Y. et al. (2016). A comprehensive performance evaluation of 3D local feature descriptors. *International Journal of Computer Vision*, 116(1):66–89.

[Guo et al. 2021] Guo, Y. et al. (2021). Deep learning for 3D point clouds: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(12):4338–4364.

[Jung et al. 2021] Jung, H. et al. (2021). Application of machine learning techniques in injection molding quality prediction: implications on sustainable manufacturing industry. *Sustainability*, 13(8):4120.

[Lee et al. 2020] Lee, I. D., Lee, I., and Han, S. (2020). 3D reconstruction of as-built model of plant piping system from point clouds and port information. *Journal of Computational Design and Engineering*, 8(1):195–209.

[Lin et al. 2018] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2018). Focal loss for dense object detection.

[Liu et al. 2021] Liu, S. et al. (2021). *3D point cloud analysis: traditional, deep learning, and explainable machine learning methods*. Springer Nature, Cham, Switzerland, 1 edition.

[Maalek et al. 2019] Maalek, R. et al. (2019). Extraction of pipes and flanges from point clouds for automated verification of pre-fabricated modules in oil and gas refinery projects. *Automation in Construction*, 103:150–167.

[Matthews 2023] Matthews, C. (2023). *The API ICP exam handbook: complete guide to passing the API 510/570/653 ICP exams*. ASME. DOI: https://doi.org/10.1115/1.862API_ch19.

[Milletari et al. 2016] Milletari, F., Navab, N., and Ahmadi, S.-A. (2016). V-Net: fully convolutional neural networks for volumetric medical image segmentation.

[Noichl et al. 2024] Noichl, F., Collins, F. C., Braun, A., and Borrmann, A. (2024). Enhancing point cloud semantic segmentation in the data-scarce domain of industrial plants through synthetic data. *Computer-Aided Civil and Infrastructure Engineering*, 39(10):1530–1549.

[Poux 2025] Poux, F. (2025). *3D data science with Python: building accurate digital environments with 3D point cloud workflows*. O'Reilly Media, Inc.

[Qi et al. 2017a] Qi, C. R., Su, H., Mo, K., and Guibas, L. J. (2017a). PointNet: deep learning on point sets for 3D classification and segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 652–660.

[Qi et al. 2017b] Qi, C. R., Yi, L., Su, H., and Guibas, L. J. (2017b). PointNet++: deep hierarchical feature learning on point sets in a metric space. *CoRR*, abs/1706.02413.

[Rashdi et al. 2022] Rashdi, R., Martínez-Sánchez, J., Arias, P., and Qiu, Z. (2022). Scanning technologies to building information modelling: a review. *Infrastructures*, 7(4):49.

[Remondino and Campana 2014] Remondino, F. and Campana, S. (2014). *3D recording and modelling in archaeology and cultural heritage – theory and best practices*. BAR International Series 2598. Archaeopress.

[Saranti et al. 2024] Saranti, A., Pfeifer, B., Gollob, C., Stampfer, K., and Holzinger, A. (2024). From 3D point-cloud data to explainable geometric deep learning: state-of-the-art and future challenges. *WIREs Data Mining and Knowledge Discovery*, 14(6):e1554.

[Sarker et al. 2024] Sarker, S., Sarker, P., Stone, G., Gorman, R., Tavakkoli, A., Bebis, G., and Sattarvand, J. (2024). A comprehensive overview of deep learning techniques for 3D point cloud classification and semantic segmentation. *Machine Vision and Applications*, 35(4).

[Shigeta et al. 2023] Shigeta, K., Nagumo, T., and Masuda, H. (2023). Point cloud segmentation for pipelines in industrial facilities using recurrent networks. *Computer-Aided Design and Applications*, pages 786–796.

[Stojanovic et al. 2019] Stojanovic, V., Trapp, M., Richter, R., and Döllner, J. (2019). Service-oriented semantic enrichment of indoor point clouds using octree-based multiview classification. *Graphical Models*, 105:101039.

[Succar 2009] Succar, B. (2009). Building information modelling framework: a research and delivery foundation for industry stakeholders. *Automation in Construction*, 18(3):357–375.

[Valero et al. 2021] Valero, E. et al. (2021). An integrated scan-to-BIM approach for buildings energy performance evaluation and retrofitting. In *Proceedings of the 38th International Symposium on Automation and Robotics in Construction (ISARC)*. International Association for Automation and Robotics in Construction (IAARC).

[Venkatesh et al. 2003] Venkatesh, V., Morris, M. G., Davis, G. B., and Davis, F. D. (2003). User acceptance of information technology: toward a unified view. *MIS Quarterly*, 27(3):425–478.

[Wang et al. 2019] Wang, Q., Guo, J., and Kim, M.-K. (2019). An application-oriented scan-to-BIM framework. *Remote Sensing*, 11(3):365.

[Wang and Kim 2019] Wang, Q. and Kim, M.-K. (2019). Applications of 3D point cloud data in the construction industry: a fifteen-year review from 2004 to 2018. *Advanced Engineering Informatics*, 39:306–319.

[Wu and He 2018] Wu, Y. and He, K. (2018). Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[Yin et al. 2022] Yin, C., Cheng, J. C., Wang, B., and Gan, V. J. (2022). Automated classification of piping components from 3D LiDAR point clouds using SE-PseudoGrid. *Automation in Construction*, 139:104300.

[Yin et al. 2021] Yin, C., Wang, B., Gan, V. J., Wang, M., and Cheng, J. C. (2021). Automated semantic segmentation of industrial point clouds using ResPointNet++. *Automation in Construction*, 130:103874.

[Zhao et al. 2024] Zhao, Z., Zhang, Z., Nie, Q., Liu, C., Zhu, H., Chen, K., and Tang, D. (2024). Probing a point cloud based expeditious approach with deep learning for constructing digital twin models in shopfloor. *Advanced Engineering Informatics*, 62:102748.

[Zhu et al. 2023] Zhu, Q., Fan, L., and Weng, N. (2023). Advancements in point cloud data augmentation for deep learning: a survey. arXiv preprint arXiv:2308.12113. Acesso em: 20 set. 2025.

[Zoumpekas et al. 2022] Zoumpekas, T., Salamó, M., and Puig, A. (2022). Rethinking design and evaluation of 3D point cloud segmentation models. *Remote Sensing*, 14(23):6049.