



UNIVERSIDADE FEDERAL DO CEARÁ
CAMPUS DE RUSSAS
CURSO DE GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

VICTOR EMANUEL AQUINO BEZERRA

REMOÇÃO AUTOMÁTICA DE OBJETOS INDESEJADOS EM NUVENS DE PONTOS
UTILIZANDO O ALGORITMO DE DETECÇÃO VOTENET

RUSSAS

2024

VICTOR EMANUEL AQUINO BEZERRA

REMOÇÃO AUTOMÁTICA DE OBJETOS INDESEJADOS EM NUVENS DE PONTOS
UTILIZANDO O ALGORITMO DE DETECÇÃO VOTENET

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Ciência da Computação
da Universidade Federal do Ceará, como
requisito parcial à obtenção do grau de bacharel
em Ciência da Computação.

Orientadora: Profa. Dra. Rosineide Fer-
nando da Paz

RUSSAS

2024

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

B469r Bezerra, Victor Emanuel Aquino.
Remoção automática de objetivos indesejados em nuvens de pontos utilizando o algoritmo de detecção votenet / Victor Emanuel Aquino Bezerra. – 2024.
49 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Russas, Curso de Ciência da Computação, Russas, 2024.
Orientação: Prof. Dr. Rosineide Fernando da Paz.

1. Nuvens de Pontos. 2. VoteNet. 3. Remoção Automática. 4. Detecção de Objetos. I. Título.
CDD 005

VICTOR EMANUEL AQUINO BEZERRA

REMOÇÃO AUTOMÁTICA DE OBJETOS INDESEJADOS EM NUVENS DE PONTOS
UTILIZANDO O ALGORITMO DE DETECÇÃO VOTENET

Trabalho de Conclusão de Curso apresentado ao
Curso de Graduação em Ciência da Computação
da Universidade Federal do Ceará, como
requisito parcial à obtenção do grau de bacharel
em Ciência da Computação.

Aprovada em: 03/10/2024

BANCA EXAMINADORA

Profa. Dra. Rosineide Fernando da Paz (Orientadora)
Universidade Federal do Ceará (UFC)

Prof. Dr. Marcos Vinicius de Andrade Lima
Universidade Federal do Ceará (UFC)

Prof. Ms. Carlos Eugênio Moreira de Sousa
Universidade Federal do Ceará (UFC)

Aos meus familiares, cuja presença e apoio incondicionais foram a força e inspiração necessárias para alcançar mais este marco. Com gratidão e amor.

AGRADECIMENTOS

À Profa. Dra. Rosineide Fernando da Paz, pela valiosa orientação ao longo do meu trabalho de conclusão de curso.

Aos professores do Laboratório de Reabilitação e Durabilidade das Construções (LAREB), pelo apoio essencial para o desenvolvimento deste trabalho.

Aos amigos e companheiros de pesquisa, Rafaela Braga e Carlos Matheus, pela dedicação e valiosa colaboração ao longo desta jornada.

Aos professores do Laboratório de Tecnologias Inovadoras (LTI), pela valiosa oportunidade de aprendizado, pela orientação fundamental em minha carreira e pelo apoio na minha inserção no mercado de trabalho.

À minha família, minha mãe Suely Aquino, meu pai Raimundo Bezerra, e minha irmã Suelen Vitória, por me formarem como indivíduo ao longo de minha vida, que nos momentos de minha ausência dedicados ao estudo superior, sempre fizeram entender que o futuro é feito a partir da constante dedicação no presente!

Aos amigos, especialmente Júnior Pergentino, Virna Diolino, Lucas Bezerra, Raphael Cavalcante, Felipe Damascena, pela amizade sincera e pelo apoio contínuo ao longo desta trajetória.

Aos servidores e terceirizados, pelo serviço e manutenção do espaço da universidade, pela dedicação e pelo trabalho essencial que contribui para o bom funcionamento do ambiente acadêmico.

Agradeço a todos os professores por me proporcionarem não apenas o conhecimento racional, mas também a manifestação do caráter e da afetividade da educação no processo de formação profissional. Em especial, agradeço ao Prof. Ms. Filipe Maciel, ao Prof. Dr. Alexandre Matos, ao Prof. Dr. Rafael Ivo e ao Prof. Dr. Bonfim Amaro.

Agradeço a Deus e aos Orixás pela proteção constante, pela clareza que trazem à minha jornada, pela força nas adversidades e pela orientação que transforma obstáculos em aprendizados.

“A vida é uma escolha, e a escolha é um ato de liberdade.”

(Jean-Paul Sartre)

RESUMO

Este trabalho explora a remoção automática de objetos indesejados em nuvens de pontos, uma tarefa complexa, mas essencial no campo da computação aplicada à engenharia civil e arquitetura. As nuvens de pontos são representações tridimensionais geradas por tecnologias como *scanners a laser* e fotogrametria, utilizadas amplamente na reconstrução de representações *BIM (Building Information Modeling)* e na digitalização de edificações. No entanto, essas nuvens muitas vezes contêm objetos irrelevantes que prejudicam a segmentação de elementos arquitetônicos e estruturais. O objetivo deste estudo foi a implementação do algoritmo *VoteNet*, uma técnica de detecção de objetos baseada em *deep learning*, para identificar e remover automaticamente esses objetos. Para avaliar a eficácia do método, foram utilizadas métricas como o Índice de *Jaccard* e a Distância de *Hausdorff*, que permitiram mensurar a precisão e a acurácia do processo de remoção. Os resultados indicam que o *VoteNet* apresentou bom desempenho na detecção e remoção de objetos indesejados, melhorando significativamente a qualidade e a usabilidade das nuvens de pontos para futuras etapas de segmentação e modelagem. A conclusão destaca o potencial do algoritmo em otimizar o fluxo de trabalho da automação na construção civil, tornando o processamento de dados mais eficiente e robusto.

Palavras-chave: nuvens de pontos; *VoteNet*; remoção automática; detecção de objetos.

ABSTRACT

This work explores the automatic removal of unwanted objects in point clouds, a complex but essential task in the field of computing applied to civil engineering and architecture. Point clouds are three-dimensional representations generated by technologies such as laser scanners and photogrammetry, widely used in the reconstruction of BIM (Building Information Modeling) representations and in the digitization of buildings. However, these clouds often contain irrelevant objects that impair the segmentation of architectural and structural elements. The objective of this study was to implement the *VoteNet* algorithm, an object detection technique based on deep learning, to automatically identify and remove these objects. To evaluate the effectiveness of the method, metrics such as the Jaccard Index and Hausdorff Distance were used, which allowed measuring the precision and accuracy of the removal process. The results indicate that *VoteNet* performed well in detecting and removing unwanted objects, significantly improving the quality and usability of point clouds for future segmentation and modeling steps. The conclusion highlights the potential of the algorithm to optimize the automation workflow in civil construction, making data processing more efficient and robust.

Keywords: point clouds; *VoteNet*; automatic removal; object detection.

LISTA DE FIGURAS

Figura 1 – Fachada do Teatro José de Alencar em nuvem de ponto.	16
Figura 2 – Nuvem de pontos do auditório do Teatro José de Alencar.	26
Figura 3 – Área da nuvem de pontos explorada no primeiro quadrante.	27
Figura 4 – Área da nuvem de pontos explorada no segundo quadrante.	28
Figura 5 – Área da nuvem de pontos explorada no terceiro quadrante.	28
Figura 6 – Área da nuvem de pontos explorada no quarto quadrante.	29
Figura 7 – Comparação entre o recorte manual e a área correspondente ao primeiro quadrante.	30
Figura 8 – Comparação entre o recorte manual e a área correspondente ao segundo quadrante.	30
Figura 9 – Comparação entre o recorte manual e a área correspondente ao terceiro quadrante.	31
Figura 10 – Comparação entre o recorte manual e a área correspondente ao quarto quadrante.	31
Figura 11 – Demonstração das cadeiras removidas, ilustrando o progresso do procedimento.	33
Figura 12 – Comparação entre o recorte automático e a área do primeiro quadrante.	34
Figura 13 – Comparação entre o recorte automático e a área do segundo quadrante.	34
Figura 14 – Comparação entre o recorte automático e a área do terceiro quadrante.	35
Figura 15 – Comparação entre o recorte automático e a área do quarto quadrante.	35
Figura 16 – Comparação entre recortes automático e manual no primeiro quadrante.	37
Figura 17 – Comparação da área removida no primeiro quadrante (automático e manual).	38
Figura 18 – Comparação entre recortes automático e manual no segundo quadrante.	38
Figura 19 – Comparação da área removida no segundo quadrante (automático e manual).	39
Figura 20 – Comparação entre recortes automático e manual no terceiro quadrante.	39
Figura 21 – Comparação da área removida no terceiro quadrante (automático e manual).	40
Figura 22 – Comparação entre recortes automático e manual no quarto quadrante.	40
Figura 23 – Comparação da área removida no quarto quadrante (automático e manual).	41

LISTA DE TABELAS

Tabela 1 – Comparação da Redução das Nuvens (em quantidade de pontos) nos Quadrantes.	41
---	----

LISTA DE CÓDIGOS-FONTE

Código-fonte 1	– Trecho de código adaptado de (MMDETECTION3D, 2020) para conversão arquivo com extensão de arquivo <i>.ply</i> para <i>.bin</i>	47
Código-fonte 2	– Trecho de código para remoção automática das bounding boxes detectadas pela inferência.	48
Código-fonte 3	– Trecho de código para calculo das métricas do Índice de Jaccard e distância de Hausdorff.	50

LISTA DE ABREVIATURAS E SIGLAS

BIM	<i>Building Information Modeling</i>
TLS	<i>Terrestrial Laser Scanning</i>
MAM	Modelo de Aprendizado de Máquina
AEC	Arquitetura, Engenharia e Construção
LiDAR	<i>Light Detection and Ranging</i>
GIS	Sistemas de Informação Geográfica
IDE	<i>Integrated Development Environment</i>
GPU	<i>Graphics Processing Unit</i>

SUMÁRIO

1	INTRODUÇÃO	15
2	OBJETIVOS	18
2.1	Objetivo Geral	18
2.2	Objetivos Específicos	18
3	FUNDAMENTAÇÃO TEÓRICA	19
3.1	Modelos de Dados Semânticos	19
3.2	Modelagem de Informações de Construção	20
3.3	Métodos Automatizados para Extração e Modelagem	22
3.3.1	<i>Índice de Jaccard</i>	22
3.3.2	<i>Distância de Hausdorff</i>	23
4	TRABALHOS RELACIONADOS	24
4.1	<i>PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space</i>	24
4.2	<i>Deep Hough Voting for 3D Object Detection in Point Clouds</i>	24
4.3	<i>ImVoteNet: Boosting 3D Object Detection in Point Clouds with Image Votes</i>	25
5	METODOLOGIA	26
5.1	Recorte da Nuvem em Quadrantes	26
5.2	Identificação Manual dos Objetos	29
5.3	Pré-processamento das Nuvens de Pontos de Entrada	32
5.4	Remoção de Caixas Delimitadoras de Volumes	32
5.5	Recortes Automatizados	32
5.6	Métricas para Avaliação	36
5.7	Recursos Utilizados	36
6	RESULTADOS	37
6.0.1	<i>Análise Comparativa das Remoções Automáticas e Manuais</i>	37
6.1	Análise Comparativa da Redução da Quantidade de Pontos	41
6.2	Repositório do Método Proposto	42
7	CONCLUSÃO	43
	REFERÊNCIAS	45
	APÊNDICES	47

APÊNDICE A –CÓDIGO CONVERSÃO <i>.PLY</i> PARA <i>.BIN</i>	47
APÊNDICE B –CÓDIGO PARA REMOÇÃO DE <i>BOUDING BOXES</i> .	48
APÊNDICE C –CÓDIGO PARA INFERÊNCIA DAS MÉTRICAS . . .	50

1 INTRODUÇÃO

A preservação de edifícios históricos envolve uma série de fatores complexos, que abrangem desde aspectos históricos e sociais até questões econômicas e ambientais. Nesse contexto, a conservação inadequada dessas estruturas frequentemente resulta em anomalias que comprometem seu valor patrimonial. Portanto, a documentação emerge como um elemento crucial nesse cenário, assegurando a comunicação eficaz entre os diversos envolvidos no processo de preservação. Com o avanço tecnológico, especialmente com a introdução do escaneamento a *laser* e de ferramentas tridimensionais, novas soluções têm surgido, permitindo a aquisição, visualização e disseminação de dados precisos, que superam as limitações das abordagens tradicionais (RUBENS, 2023, p. 18).

Um aspecto central da preservação é o processo de transformar uma nuvem de pontos em *Building Information Modeling* (BIM). Essa transformação envolve a conversão de dados capturados por *laser scan* e fotogrametria em modelos digitais tridimensionais de alta precisão. Inicialmente, a nuvem de pontos, gerada por técnicas como fotogrametria de curto alcance ou *Terrestrial Laser Scanning* (TLS), serve como uma base sólida para a criação de superfícies que representam com fidelidade os elementos arquitetônicos. Esses dados são, então, mapeados como objetos paramétricos no ambiente BIM, permitindo a associação de características físicas e mecânicas a cada componente. Dessa forma, essa abordagem não apenas facilita a gestão e análise das edificações, mas também enriquece a compreensão do patrimônio cultural, promovendo uma preservação mais eficiente e abrindo novas oportunidades para o desenvolvimento urbano (PEPE *et al.*, 2021).

Entretanto, a digitalização do patrimônio histórico enfrenta o desafio de capturar a autenticidade e a fidelidade das estruturas originais. É imperativo considerar aspectos como texturas, cores e materiais, que possuem um significado histórico e artístico. Nesse sentido, é crucial que as tecnologias de digitalização reproduzam esses elementos com precisão. Um aspecto importante deste processo é a etapa de limpeza e filtragem dos dados, que consiste na eliminação de informações irrelevantes das varreduras, como edifícios adjacentes, pessoas, vegetação e obstruções. Essa etapa é essencial não apenas para reduzir o tamanho do conjunto de dados, mas também para tornar o processo de registro mais eficiente (RUBENS, 2023, p. 30, 40).

O BIM, por sua vez, é um processo que integra informações ao longo do ciclo de vida de uma construção, utilizando modelos 3D dinâmicos que representam não apenas aspectos

gráficos, mas também as propriedades e funções dos elementos do edifício. Essa metodologia possibilita a simulação de desempenho, análise de sustentabilidade e detecção de conflitos antes da construção, melhorando a coordenação entre as disciplinas envolvidas. Além disso, a capacidade do BIM de armazenar dados em um ambiente colaborativo permite atualizações em tempo real, facilitando a tomada de decisões. Os benefícios do BIM incluem, portanto, a eficiência na documentação, automação de fluxos de trabalho, maior colaboração interdisciplinar e integração de cronogramas e custos, resultando em um planejamento, precisão e visualização de projetos significativamente aprimorados (MONTEIRO; MARTINS, 2011).

Neste contexto, o trabalho de Vieira *et al.* (2023) proporcionou um conjunto valioso de nuvens de pontos obtidas por técnicas de *laser scanner* e fotogrametria, especificamente no levantamento do Teatro José de Alencar, um importante patrimônio histórico localizado em Fortaleza, no estado do Ceará (VIEIRA *et al.*, 2023). Esses dados não apenas serviram como base para a detecção e remoção automática de elementos indesejados na presente pesquisa, mas também ofereceram uma representação detalhada do teatro. A Figura 1, extraída dos registros de Vieira *et al.*, ilustra a fachada do teatro em nuvem de pontos, evidenciando a riqueza dos dados coletados.

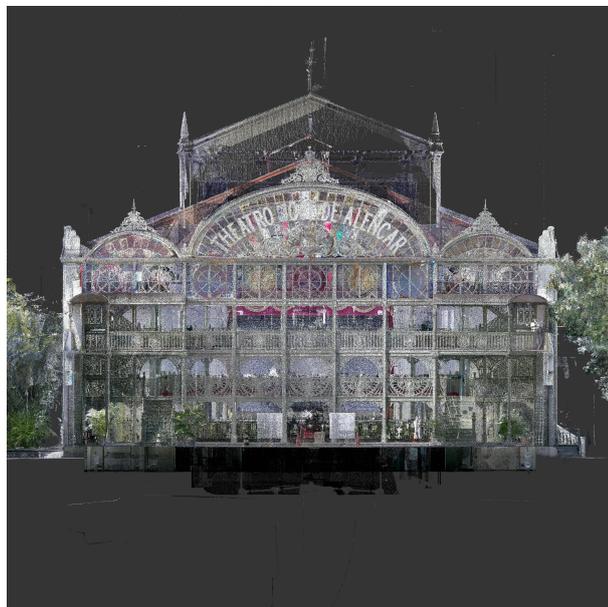


Figura 1 – Fachada do Teatro José de Alencar em nuvem de ponto.

A pesquisa em questão também se insere no campo da ciência da computação, destacando a aplicação de técnicas de aprendizado de máquina e algoritmos avançados na preservação digital. Ao explorar a interseção entre nuvens de pontos, BIM e inteligência artificial, busca-se aumentar a eficiência e a precisão da modelagem, com impacto direto na

preservação do patrimônio cultural. O uso de IA, como *deep learning*, permite o desenvolvimento de métodos automáticos para limpeza e preparação de dados, contribuindo para uma modelagem mais eficiente e precisa. Assim, além de utilizar os dados gerados por Vieira *et al.*, a pesquisa avança nas práticas de documentação digital com uma abordagem automatizada, aplicável a outros levantamentos.

O principal objetivo deste trabalho é desenvolver um método automatizado de detecção e remoção de objetos indesejados em nuvens de pontos, utilizando o algoritmo *VoteNet*. Espera-se que essa abordagem reduza significativamente o esforço manual na criação de representações digitais de edificações em BIM. As limitações da pesquisa indicam áreas futuras de investigação, como o reconhecimento de diferentes classes de objetos e melhorias na precisão, ampliando, assim, as aplicações do método.

Por fim, este trabalho está estruturado em capítulos que abordam de forma sistemática as etapas da pesquisa. O primeiro capítulo introduz o tema, ressaltando a relevância da remoção automatizada de elementos indesejados. O segundo capítulo discute o referencial teórico, apresentando conceitos de BIM e algoritmos de detecção. A metodologia detalha o desenvolvimento do método, bem como as ferramentas e configurações utilizadas. No capítulo de resultados, são analisados o desempenho e a eficácia do modelo em comparação com métodos tradicionais. Portanto, as conclusões resumem os principais achados e sugerem direções futuras para aprimorar a modelagem automatizada em BIM.

2 OBJETIVOS

2.1 Objetivo Geral

Desenvolver um método automatizado fundamentado no modelo de rede neural profunda *VoteNet* para a detecção e remoção eficiente de elementos indesejados em cenas de nuvens de pontos, visando a otimização e agilidade do processo de *scan to BIM*, o qual transforma essas nuvens em modelos digitais de BIM.

2.2 Objetivos Específicos

1. Inferir o modelo *VoteNet* para a detecção automática de objetos em cenas de nuvens de pontos tridimensionais.
2. Criar um repositório que estabeleça um ambiente de trabalho simplificado para a implementação do método, assegurando a configuração adequada.
3. Apresentar uma estratégia automatizada para a remoção de objetos indesejados em nuvens de pontos.
4. Avaliar a eficácia abordagem de remoção automática proposta em comparação com a remoção manual realizada por software especializado, analisando métricas de qualidade e eficiência.

3 FUNDAMENTAÇÃO TEÓRICA

Para embasar o desenvolvimento e a aplicação do método automatizado de remoção de objetos em nuvens de pontos, é fundamental compreender os conceitos e tecnologias subjacentes a este estudo. Esta seção explora os principais elementos teóricos e técnicos que sustentam a pesquisa, incluindo uma análise de modelos de dados semânticos, que atribuem valores representativos às informações, permitindo uma interpretação mais rica dos dados coletados. Em seguida, abordaremos a modelagem de informações de construção *BIM*, amplamente utilizada na documentação de estruturas, destacando como suas inovações tornam o processo de reconstrução e digitalização de ambientes mais eficiente e preciso. Esses componentes teóricos são essenciais para o desenvolvimento do método proposto, fundamentando sua eficácia e demonstrando sua relevância nos campos da visão computacional e preservação digital.

3.1 Modelos de Dados Semânticos

A semântica aborda a relação entre símbolos significativos (palavras, frases, sinais e símbolos) e as entidades ou conceitos que esses símbolos representam, facilitando assim a interação entre pessoas e objetos. Em outras palavras, a semântica é o estudo do significado, preocupando-se com a conexão entre significantes e as coisas que esses significantes denotam, sejam entidades, conceitos ou ideias. No contexto dos sistemas computacionais, ela desempenha um papel crucial, pois auxilia na correta interpretação dos dados e na comunicação eficiente entre diferentes sistemas (ALEXOPOULOS, 2020, p. 4).

Nesse sentido, a modelagem semântica de dados surge como uma solução importante. Ela refere-se ao desenvolvimento de representações em que o significado dos dados é explícito, tornando-os compreensíveis tanto para humanos quanto para máquinas. Além disso, essa abordagem permite que os dados sejam precisos e bem interpretados, o que é essencial para enfrentar os desafios impostos pelos sistemas de IA. Particularmente, quando combinada com Modelo de Aprendizado de Máquina (MAM), a modelagem semântica pode melhorar significativamente a precisão e a interpretação dos dados, proporcionando um avanço em termos de automação e compreensão (ALEXOPOULOS, 2020, p. 4, 7).

Consequentemente, os modelos de dados semânticos têm ganhado popularidade em diversas áreas, como processamento de linguagem natural e análise de sentimentos, pois aprimoram a funcionalidade de aplicativos de IA ao permitir o acesso a conhecimento simbólico

explícito (ALEXOPOULOS, 2020, p. 7, 146, 170). Além disso, esses modelos ajudam a padronizar e alinhar o significado de dados heterogêneos, tornando-os mais interoperáveis e utilizáveis. Na engenharia civil e na arquitetura, sua integração com tecnologias como nuvens de pontos e BIM tem otimizado processos e aumentado a eficiência no planejamento e execução de projetos.

3.2 Modelagem de Informações de Construção

Especificamente na engenharia civil, a modelagem com nuvens de pontos é uma prática cada vez mais relevante e crescente, estreitamente relacionada ao uso de modelos de dados semânticos. Este tipo de modelagem é amplamente empregada nos setores de Arquitetura, Engenharia e Construção (AEC), especialmente na criação de Modelagem da Informação da Construção BIM (PELLIS *et al.*, 2022a).

Ademais, a BIM representa digitalmente as características físicas e funcionais detalhadas de uma construção. Funciona como um recurso compartilhado para informações sobre a construção, permitindo decisões ao longo de todo o seu ciclo de vida, desde o projeto inicial até a demolição (MONTEIRO; MARTINS, 2011). É uma base confiável de dados que cobre todos os aspectos da construção e operação de uma instalação (POLJANŠEK, 2017). Esta modelagem integra dados essenciais de produtos e ativos em um modelo computacional tridimensional, facilitando a gestão eficaz e eficiente das informações de um edifício (PärN *et al.*, 2017).

O uso de BIM oferece vantagens significativas no gerenciamento de informações de construção em 3D, sendo este um modelo fundamental que facilita a comunicação entre partes interessadas com diferentes experiências em um ambiente de dados compartilhado (MONTEIRO; MARTINS, 2011). Essa abordagem possibilita a troca e o compartilhamento de informações ao longo de todo o processo de desenvolvimento de um edifício, promovendo eficiência e colaboração entre os envolvidos.

Nessa perspectiva, a abordagem BIM parte do escaneamento de um objeto para a obtenção da nuvem de pontos, que, por sua vez, serve de base para a construção do modelo 3D (RUBENS, 2023). Após a formação do modelo tridimensional, são adicionadas as informações necessárias, como propriedades e materiais, para enriquecer o modelo e torná-lo mais informativo. Além disso, nuvens de pontos são conjuntos de dados tridimensionais que representam objetos ou superfícies em um ambiente físico, capturando a posição de pontos no espaço. Essa tecnologia é comumente utilizada em levantamentos topográficos, mapeamento de terrenos, inspeção

estrutural, monitoramento de canteiros de obras e outras aplicações relevantes (ATAZADEH *et al.*, 2019).

A modelagem com nuvens de pontos envolve a aquisição desses dados através de tecnologias como o escaneamento a laser terrestre *Light Detection and Ranging* (LiDAR). Em seguida, os dados são analisados e processados para criar modelos 3D detalhados de estruturas, terrenos ou objetos. Isso pode incluir a geração de modelos digitais de terreno, edifícios em 3D, representações de infraestrutura viária e outras aplicações relevantes (PELLIS *et al.*, 2022b).

A aplicação de modelos semânticos nesse contexto envolve atribuir significado aos elementos detectados na nuvem de pontos. Por exemplo, identificar objetos como portas, janelas, escadas, entre outros, e associar informações semânticas a esses elementos (RUBENS, 2023, p. 46). Essas informações são úteis para diversas tarefas, algumas das quais são listadas a seguir.

1. **Análise e Inspeção:** Facilita a identificação e análise de elementos específicos em uma nuvem de pontos, como rachaduras em estruturas, por exemplo.
2. **Planejamento e Projeto:** Utiliza nuvens de pontos para auxiliar no planejamento de projetos, permitindo uma modelagem precisa do terreno existente e outros elementos relevantes.
3. **Monitoramento de Canteiros de Obras:** Acompanha o progresso de construções comparando nuvens de pontos coletadas em diferentes momentos, detectando desvios em relação ao projeto original.
4. **Sistemas de Informação Geográfica (GIS):** Integra nuvens de pontos em sistemas GIS para melhorar a precisão e riqueza dos dados geoespaciais.
5. **Análise de Impacto Ambiental:** Avalia impactos ambientais de projetos na engenharia civil com base em informações detalhadas de nuvens de pontos.

A detecção de objetos em nuvens de pontos é fundamental para diversas aplicações na engenharia civil, porém, o manuseio dessas nuvens apresenta desafios devido ao grande volume de dados gerados, que aumentam conforme a complexidade e o detalhamento da geometria escaneada. Quanto mais detalhado o objeto, maior é o esforço computacional necessário para sua análise, o que pode dificultar a extração de informações. Nesse contexto, a limpeza das nuvens de pontos, removendo elementos indesejados, é essencial para facilitar a visualização e reduzir a carga computacional nas etapas subsequentes de segmentação e análise, sendo um passo crucial para a elaboração de representações BIM precisas (RUBENS, 2023, p. 40).

3.3 Métodos Automatizados para Extração e Modelagem

Embora ainda sejam limitados os estudos focados especificamente na remoção de objetos indesejados em nuvens de pontos, a área tem avançado na automatização de processos BIM. Pesquisas têm discutido algoritmos voltados para a detecção, segmentação e classificação de dados em nuvens de pontos, embora a aplicação prática dessas descobertas seja menos frequente. Um exemplo disso é a revisão sistemática realizada por (LOPES *et al.*, 2023), que explora técnicas voltadas para a automação da construção de modelos BIM, evidenciando a necessidade de tornar esses processos mais acessíveis e funcionais.

Além disso, estudos como o de (WANG *et al.*, 2015) buscam automatizar a extração de componentes arquitetônicos, como paredes, portas e janelas, diretamente a partir das nuvens de pontos. Essa abordagem visa otimizar a reconstrução de modelos, permitindo a identificação e classificação automática dos elementos relevantes. O presente trabalho contribui para esse cenário ao propor a remoção automatizada de objetos desnecessários, reduzindo o volume de pontos na nuvem e, conseqüentemente, o esforço computacional exigido para o processamento e modelagem. Isso torna o processo mais ágil e menos oneroso, economizando recursos e tempo no tratamento das nuvens de pontos.

3.3.1 Índice de Jaccard

O Índice de *Jaccard* é uma métrica utilizada para quantificar a similaridade entre dois conjuntos, sendo frequentemente aplicada em visão computacional e análise de dados. No presente estudo, foi empregado para comparar as extrações automáticas e manuais de objetos em nuvens de pontos. O cálculo do Índice de *Jaccard* envolve a definição dos conjuntos A e B , o cálculo dos elementos comuns ($|A \cap B|$) e o número total de elementos ($|A \cup B|$). A fórmula é dada por:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (3.1)$$

Valores próximos de 1 indicam alta similaridade, enquanto valores próximos de 0 sugerem baixa sobreposição. Embora tradicionalmente aplicado em imagens bidimensionais, o Índice de *Jaccard* pode ser estendido para nuvens de pontos tridimensionais, como destacado por (SALARI *et al.*, 2022).

3.3.2 Distância de Hausdorff

A Distância de *Hausdorff* é uma métrica que mede a maior distância mínima entre dois conjuntos de pontos, sendo especialmente útil para comparar nuvens de pontos tridimensionais. Primeiro, definem-se os conjuntos A e B . Em seguida, para cada ponto $a \in A$, calcula-se a menor distância para os pontos em B , resultando em $\inf_{b \in B} d(a, b)$. Simetricamente, o mesmo cálculo é realizado para os pontos em B em relação a A . A fórmula final da Distância de *Hausdorff* é dada por:

$$d_H(A, B) = \max \left\{ \sup_{a \in A} \inf_{b \in B} d(a, b), \sup_{b \in B} \inf_{a \in A} d(b, a) \right\} \quad (3.2)$$

Valores mais próximos de 0 indicam maior similaridade entre as nuvens de pontos, enquanto valores maiores indicam discrepâncias. Ao contrário do Índice de Jaccard, cujos valores estão sempre entre 0 e 1, a Distância de *Hausdorff* pode variar de 0 até infinito, dependendo da separação dos conjuntos. Esta métrica é amplamente utilizada em comparações de formas complexas, como descrito por (VIEIRA, 2012), sendo uma ferramenta valiosa para avaliar a precisão de modelos gerados automaticamente.

4 TRABALHOS RELACIONADOS

O início deste trabalho foi caracterizado por uma investigação na literatura, com o objetivo de identificar algoritmos de aprendizado de máquina e inteligência artificial aplicáveis à resolução de problemas de reconstrução para modelagem em BIM. As buscas foram realizadas nas principais bases de dados acadêmicas, como *IEEE Xplore*, *ScienceDirect*, *Scopus* e *Google Acadêmico*, utilizando termos como “algoritmos de aprendizado de máquina para BIM,” “detecção e remoção em nuvens de pontos” e “inteligência artificial em modelagem de edificações.” A partir dessa busca criteriosa, os trabalhos a serem apresentados a seguir foram selecionados por sua relevância e contribuição para o avanço das tecnologias voltadas para a detecção e segmentação de estruturas em nuvens de pontos.

4.1 *PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space*

Embora o algoritmo *PointNet++* (QI *et al.*, 2017) represente uma evolução significativa em relação ao *PointNet* convencional, ele permanece como uma base essencial para os estudos nesta área. Esta rede neural convolucional é crucial para uma variedade de algoritmos e aplicações, incluindo segmentação, classificação e detecção de objetos, estruturas e cenas em nuvens de pontos. A capacidade do *PointNet++* de lidar com dados não estruturados, como as nuvens de pontos, torna-o uma ferramenta poderosa para a extração de informações geométricas e semânticas de ambientes construídos, possibilitando avanços notáveis na compreensão e representação digital desses espaços.

4.2 *Deep Hough Voting for 3D Object Detection in Point Clouds*

Nesse contexto, o *VoteNet* (QI *et al.*, 2019) surge como uma solução projetada para maximizar a eficiência, oferecendo um modelo compacto e de alta precisão. Capaz de processar grandes conjuntos de dados de nuvens de pontos, o *VoteNet* atinge resultados de ponta em benchmarks renomados, como o SUN RGB-D (SONG *et al.*, 2015). Sua abordagem se destaca por se basear exclusivamente em informações geométricas, sem depender de imagens coloridas, o que confere ao modelo uma robustez superior e menor vulnerabilidade a variações nos dados externos. Dessa forma, o *VoteNet* se consolida como uma ferramenta eficaz para a detecção de objetos em ambientes tridimensionais, complementando e expandindo as capacidades oferecidas pelo *PointNet++*.

Os principais componentes do *VoteNet* são detalhados a seguir:

1. **Rede de Pontos Profunda:** O *VoteNet* emprega redes neurais especializadas projetadas para processar diretamente nuvens de pontos, abordando eficazmente a natureza esparsa e irregular dos dados tridimensionais. Essa arquitetura facilita a extração de características relevantes, promovendo uma compreensão mais aprofundada das estruturas representadas.
2. **Votação Hough:** A técnica de votação *Hough* é essencial para aprimorar a detecção de objetos, permitindo a identificação precisa da localização e do tamanho dos mesmos. Ao acumular as votações dos pontos da nuvem, essa abordagem fortalece a previsibilidade do modelo, aumentando sua confiabilidade na detecção em ambientes complexos e variados.

4.3 *ImVoteNet: Boosting 3D Object Detection in Point Clouds with Image Votes*

De maneira correspondente, o algoritmo *ImVoteNet* (QI *et al.*, 2020) representa uma extensão aprimorada do *VoteNet*, com o objetivo de aumentar a precisão na detecção de objetos em cenas de nuvens de pontos por meio da integração de informações provenientes de imagens bidimensionais correspondentes. Este algoritmo emprega um detector de objetos em imagens 2D para identificar a localização dos objetos na imagem, transferindo subsequentemente essas informações para a nuvem de pontos, o que orienta o detector na localização dos objetos de maneira mais eficiente e precisa.

Estudos dessa natureza possuem potencial significativo para aprimorar o processo discutido neste trabalho. No entanto, o *ImVoteNet* apresenta certas limitações, como a dependência de imagens que correspondam à cena tridimensional e a exigência de um detector de objetos 2D como etapa de pré-processamento antes da inferência. Embora essas restrições sejam pertinentes, elas também sugerem oportunidades para a integração com outros métodos, visando a otimização da reconstrução e segmentação de nuvens de pontos.

5 METODOLOGIA

Este trabalho caracteriza-se como uma pesquisa de natureza explicativa e adota uma abordagem quantitativa, envolvendo a realização de experimentos controlados com o objetivo de testar hipóteses e investigar teorias de maneira prática e sistemática (GIL, 2002). Inicialmente, será destacado o processo de remoção manual de objetos indesejados em nuvens de pontos, que serve como base para a comparação.

Em seguida, será apresentada a metodologia automatizada que utiliza técnicas de *deep learning* para a detecção e remoção desses objetos, manipulando e controlando variáveis relevantes para garantir a precisão dos resultados. Por fim, serão detalhadas as métricas estabelecidas para a comparação entre os métodos manual e automatizado, permitindo uma análise crítica da eficácia das abordagens empregadas.

5.1 Recorte da Nuvem em Quadrantes

Para o presente trabalho, a nuvem de pontos foi recortada em quadrantes a fim de facilitar o manuseio, reduzir o tempo de carregamento nos softwares e permitir uma melhor análise e comparação dos resultados, possibilitando uma interpretação mais precisa das características presentes no ambiente estudado. A Figura 2 mostra a visão original do auditório, destacando a disposição das cadeiras e a visão panorâmica do palco.

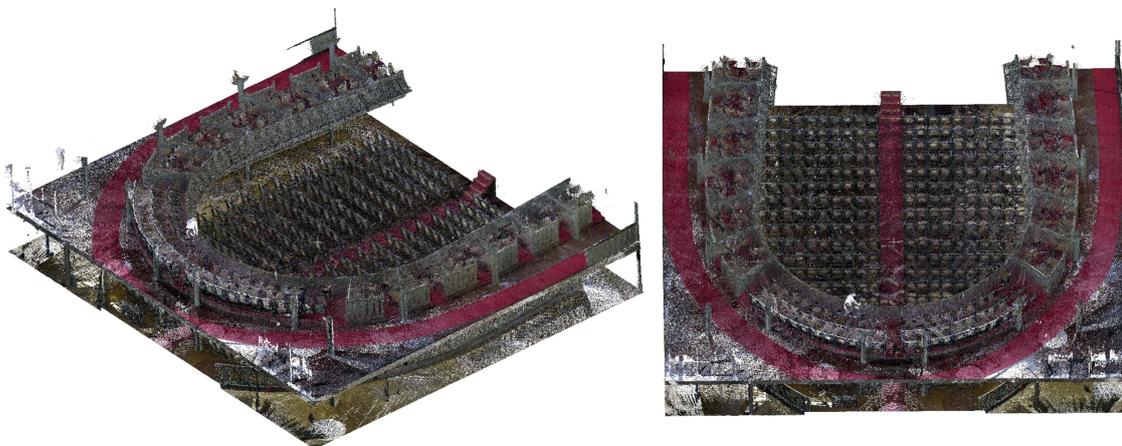


Figura 2 – Nuvem de pontos do auditório do Teatro José de Alencar.

As áreas correspondentes ao andar superior foram descartadas, de forma a concentrar a análise nas partes do auditório que apresentam maior relevância para o escopo deste estudo. A decisão de excluir essas áreas se deve à necessidade de focar nas seções que desempenham um papel central na avaliação e compreensão dos aspectos mais importantes do auditório.

O centro do auditório, por sua vez, foi segmentado cuidadosamente em quatro quadrantes, com o objetivo de facilitar uma análise mais detalhada e organizada da nuvem de pontos. Essa divisão estratégica permite que cada quadrante represente uma seção distinta do espaço, viabilizando uma inspeção minuciosa das características específicas presentes em cada área. Dessa maneira, é possível identificar particularidades em cada quadrante que poderiam passar despercebidas em uma análise mais ampla, garantindo um estudo mais preciso e criterioso.

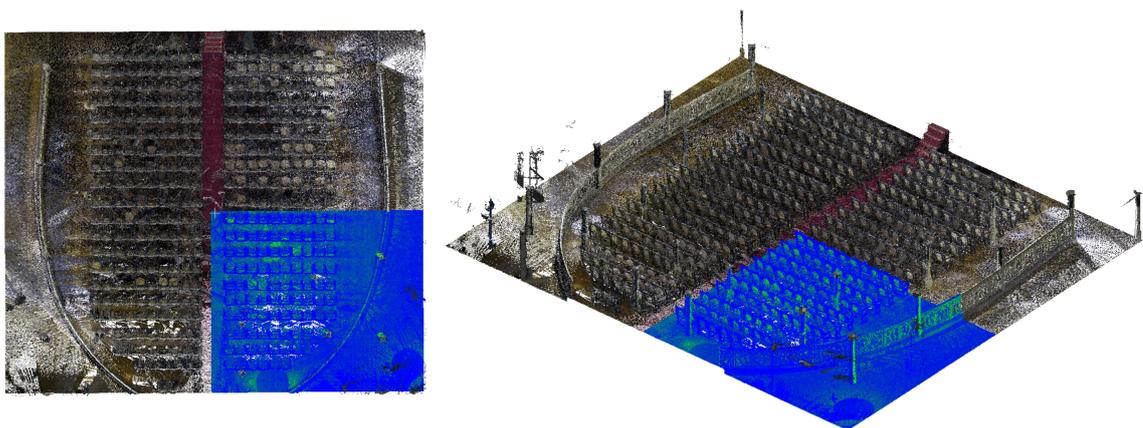


Figura 3 – Área da nuvem de pontos explorada no primeiro quadrante.

As áreas em azul das Figuras 3 e 4, representam respectivamente os quadrantes 1 e 2 referentes ao andar inferior da nuvem de pontos do Teatro José de Alencar, cada área contém aproximadamente 100 cadeiras, é possível observar outras estruturas, principalmente pilares de sustentação e guarda-corpos.

As áreas destacadas nas Figuras 5 e 6, que correspondem aos quadrantes 3 e 4, apresentam falhas visíveis no chão, como pontos faltantes. Essas falhas ocorrem principalmente nas regiões com maior concentração de cadeiras, devido à limitação do *scanner*, que, bloqueado por objetos como as cadeiras, não consegue captar adequadamente essas superfícies, resultando em lacunas nos dados. Essa limitação técnica é comum em ambientes com obstáculos que impedem a linha de visão direta do *scanner*, afetando a precisão da nuvem de pontos.

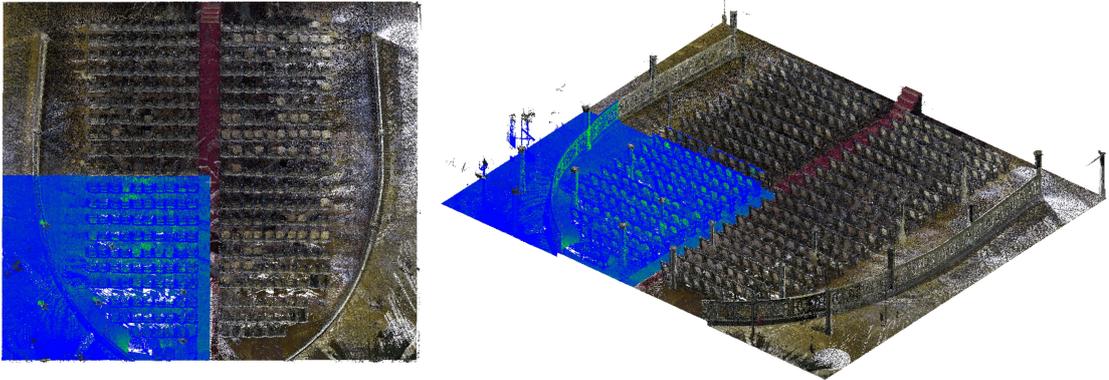


Figura 4 – Área da nuvem de pontos explorada no segundo quadrante.

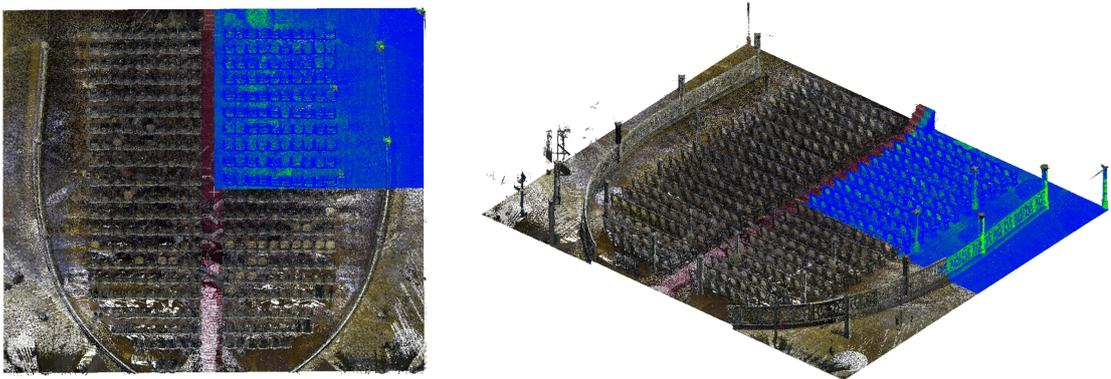


Figura 5 – Área da nuvem de pontos explorada no terceiro quadrante.

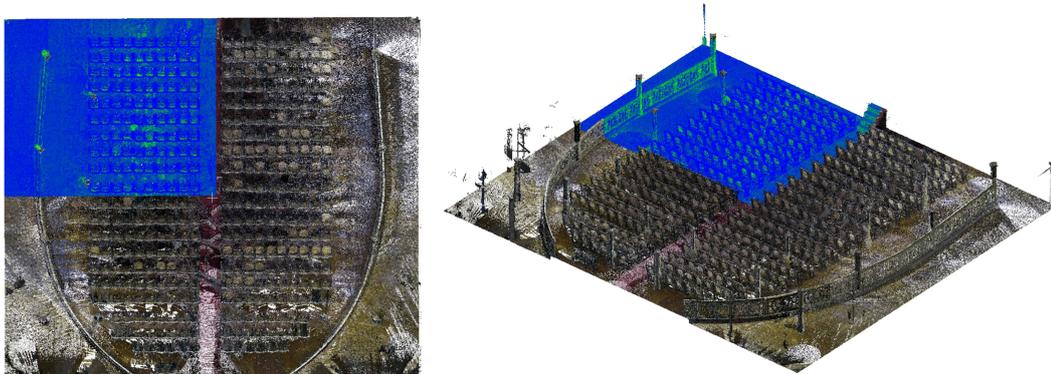


Figura 6 – Área da nuvem de pontos explorada no quarto quadrante.

5.2 Identificação Manual dos Objetos

Para a remoção manual das cadeiras nas cenas de nuvens de pontos, utilizou-se o software *CloudCompare*, uma solução de código aberto com ferramentas de recorte. Considerando a natureza manual do processo, buscou-se alcançar o melhor resultado possível, isolando os objetos indesejados e preservando as estruturas essenciais para a integridade da nuvem de pontos.

Nesta etapa, são descritos os procedimentos de recorte manual realizados em cada quadrante. A ferramenta de seleção manual de volumes foi utilizada para remover ou preservar áreas específicas, exigindo um trabalho minucioso para evitar a remoção de partes importantes das estruturas de interesse, garantindo que apenas os elementos indesejados fossem excluídos.

As Figuras 7 e 8 ilustram os quadrantes 1 e 2 após a extração manual. Com esses recortes, é possível evidenciar claramente as falhas na superfície do chão mencionadas anteriormente.

As Figuras 9 e 10 correspondem aos quadrantes 3 e 4. A Figura 9 oferece uma perspectiva superior, acentuando ainda mais as imperfeições da nuvem em relação ao solo. É importante ressaltar que fragmentos de estruturas significativas podem interferir na remoção de objetos indesejados, especialmente quando essas estruturas estão muito próximas umas das outras.

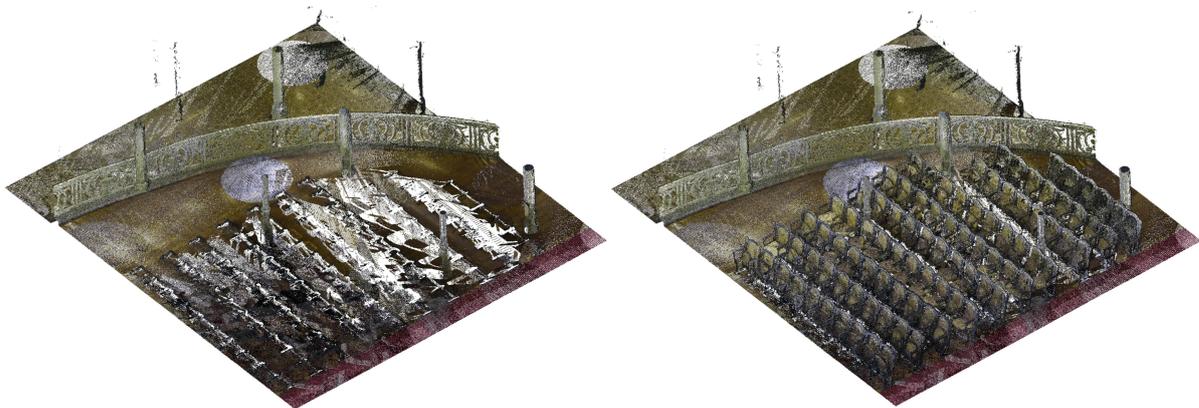


Figura 7 – Comparação entre o recorte manual e a área correspondente ao primeiro quadrante.

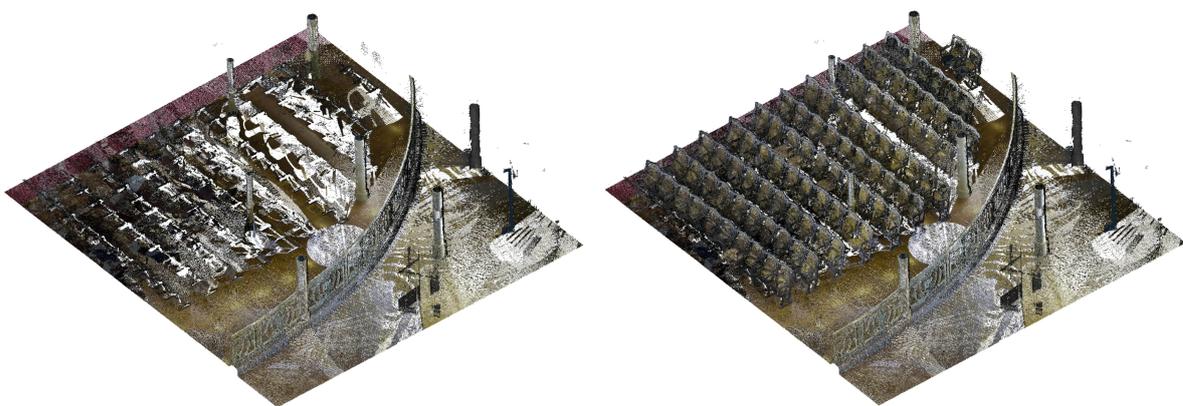


Figura 8 – Comparação entre o recorte manual e a área correspondente ao segundo quadrante.

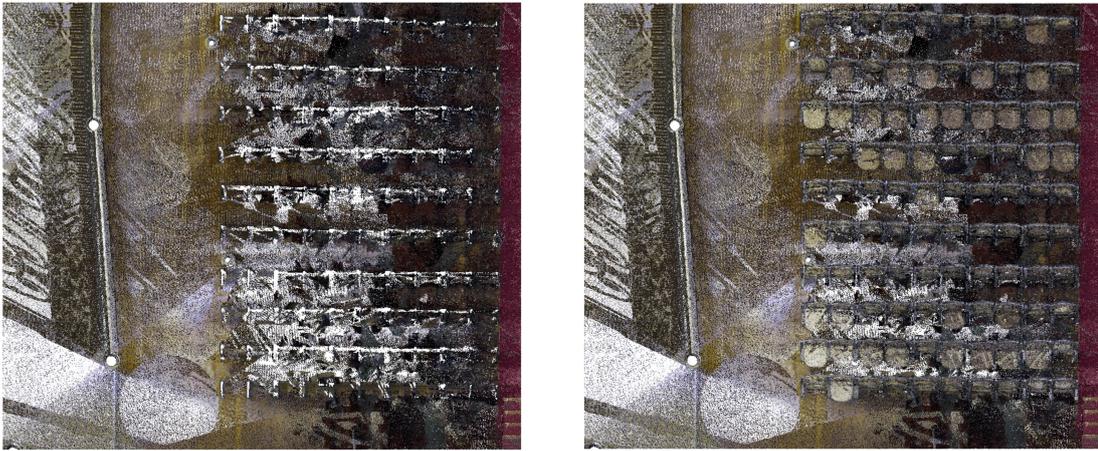


Figura 9 – Comparação entre o recorte manual e a área correspondente ao terceiro quadrante.

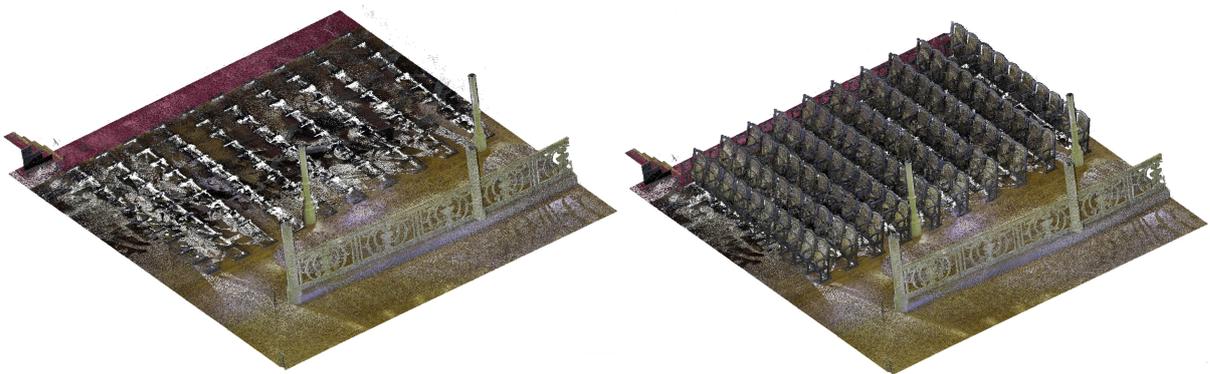


Figura 10 – Comparação entre o recorte manual e a área correspondente ao quarto quadrante.

5.3 Pré-processamento das Nuvens de Pontos de Entrada

O arquivo do auditório utilizado neste trabalho foi inicialmente fornecido em um formato que, para viabilizar a manipulação das nuvens de pontos divididas em quadrantes, precisou ser convertido para o formato *.ply*. A escolha desse formato deve-se à sua ampla compatibilidade com as ferramentas empregadas na metodologia, além de garantir uma conversão eficiente que preserva as características geométricas e cromáticas essenciais para as etapas subsequentes do processo.

No entanto, para a fase de inferência do algoritmo, foi necessário converter os arquivos de *.ply* para *.bin*, formato exigido pela biblioteca utilizada no procedimento. O código responsável pela conversão dos arquivos *.ply* para *.bin* pode ser encontrado no apêndice A.

5.4 Remoção de Caixas Delimitadoras de Volumes

Após o processo de inferência, um diretório é gerado contendo um arquivo de saída no formato *.json*, que armazena as caixas delimitadoras dos objetos detectados, juntamente com suas respectivas notas preditivas, indicativas da qualidade de cada delimitação. O processamento subsequente, necessário para a remoção das caixas correspondentes a volumes indesejados, é realizado com base nas informações contidas nesse arquivo.

Para aprimorar a precisão dos resultados, foi implementado um filtro que seleciona apenas as caixas delimitadoras com notas preditivas iguais ou superiores a 0,9. Devido à alta densidade de caixas geradas durante a inferência das nuvens de pontos, o processo de remoção foi realizado em múltiplas iterações, eliminando gradualmente as caixas com as melhores notas até que todos os elementos indesejados, como as cadeiras, fossem completamente removidos.

O código responsável pela remoção automatizada dos itens inferidos está disponível no apêndice B. Cabe ressaltar que o código é configurável, permitindo ajustes no nível de rigor do filtro, o que possibilita aumentar ou reduzir a exigência para a seleção das caixas delimitadoras conforme a necessidade da aplicação.

5.5 Recortes Automatizados

Para ilustrar as etapas do processo de inferência, a Figura 11 apresenta uma representação clara das cadeiras removidas ao longo do procedimento. Esse processo foi realizado de forma sistemática envolvendo as seguintes etapas:

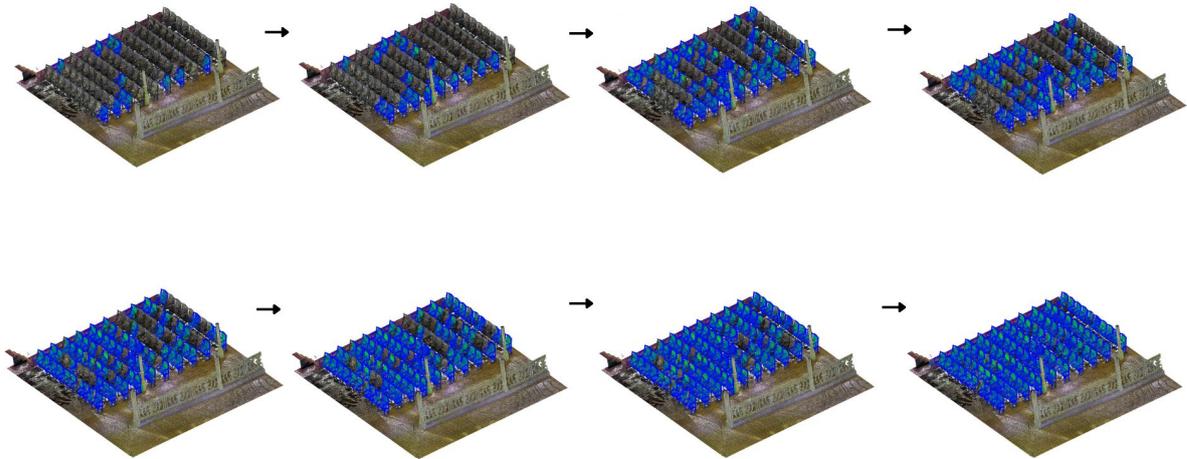


Figura 11 – Demonstração das cadeiras removidas, ilustrando o progresso do procedimento.

1. **Inferência Inicial:** Executar a primeira inferência sobre a nuvem de pontos, visando a identificação das cadeiras presentes na cena.
2. **Remoção Parcial:** Proceder com a remoção das cadeiras identificadas, com base nos dados obtidos na inferência inicial.
3. **Repetição do Processo:** Repetir o processo de inferência em cada nuvem resultante.
4. **Avaliação:** Avaliar as saídas geradas após cada inferência, realizando ajustes no processo conforme necessário para otimizar os resultados alcançados.

Nesta seção, apresentamos as saídas finais obtidas a partir da execução da inferência nas nuvens de pontos de todos os quadrantes, comparando-as com a nuvem original.

As Figuras 12 e 13 mostram a comparação das saídas da remoção automática das cadeiras nos quadrantes 1 e 2, respectivamente. Estas imagens ilustram de forma clara limpeza alcançada após a aplicação do processo de inferência, permitindo uma avaliação direta da eficácia do método automatizado.

As Figuras 14 e 15 referem-se às saídas obtidas pela remoção automática dos quadrantes 3 e 4, respectivamente. Essas imagens complementam a análise, demonstrando a consistência do desempenho do algoritmo em diferentes seções do auditório. A Figura 14 ilustra a eficácia da remoção, destacando uma visão mais limpa e livre de obstruções, enquanto a Figura 15 reforça a capacidade do método em lidar com áreas mais complexas, onde as cadeiras estavam dispostas de maneira mais densa.

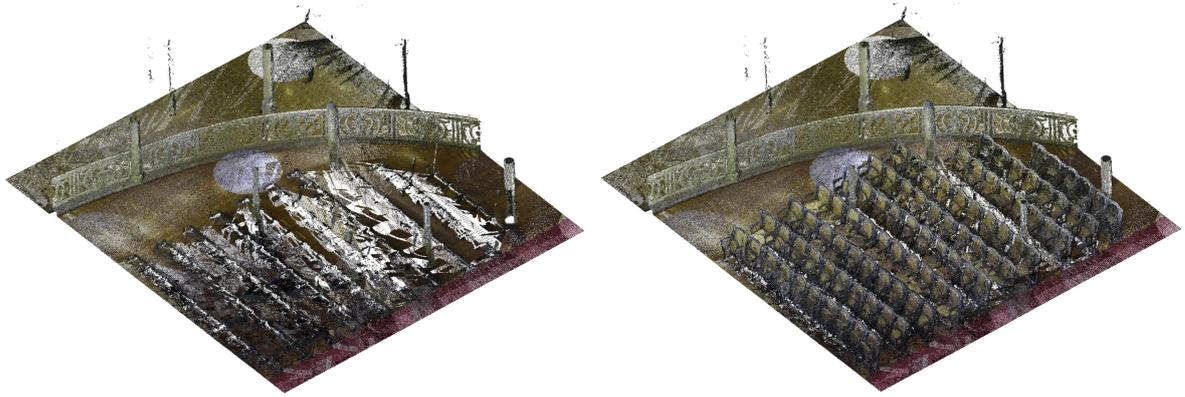


Figura 12 – Comparação entre o recorte automático e a área do primeiro quadrante.

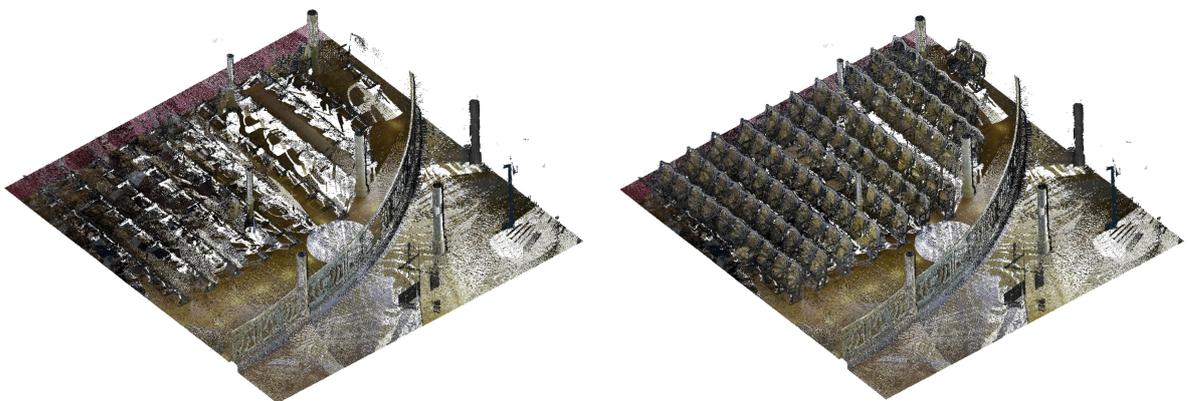


Figura 13 – Comparação entre o recorte automático e a área do segundo quadrante.

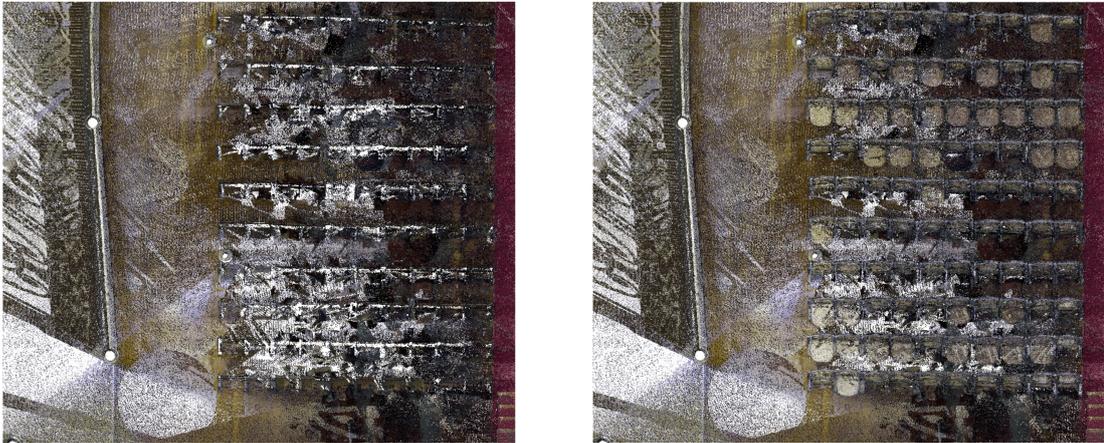


Figura 14 – Comparação entre o recorte automático e a área do terceiro quadrante.

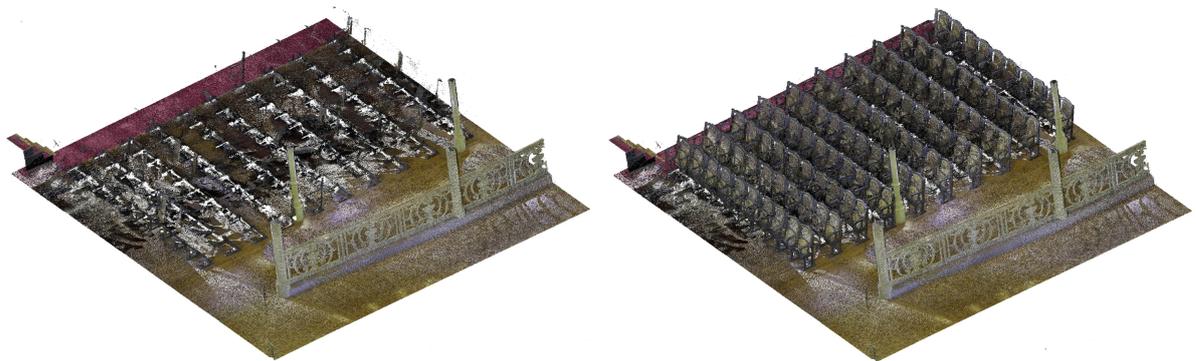


Figura 15 – Comparação entre o recorte automático e a área do quarto quadrante.

Essas comparações são fundamentais para validar a metodologia proposta e demonstrar a eficácia da inferência automatizada em cenários práticos, onde a precisão e a agilidade são cruciais para análises subsequentes. A capacidade de manter a integridade das características estruturais enquanto se removem obstruções desnecessárias representa um avanço significativo na manipulação de nuvens de pontos em ambientes complexos.

5.6 Métricas para Avaliação

Para avaliar a qualidade das inferências automáticas em comparação com as extrações manuais, foram utilizadas métricas reconhecidas na literatura, como o Índice de *Jaccard* e a Distância de *Hausdorff*. O Índice de *Jaccard* mede a similaridade e a sobreposição entre conjuntos, variando de 0 a 1; valores próximos a 1 indicam uma alta similaridade, sugerindo que os métodos automatizados conseguem replicar as características essenciais das extrações manuais. A Distância de *Hausdorff*, por sua vez, quantifica a discrepância entre as nuvens, onde valores menores indicam maior congruência, refletindo a eficácia do processo de inferência na preservação da estrutura original dos dados. O código utilizado para calcular essas métricas está disponível no Apêndice C.

5.7 Recursos Utilizados

As configurações da máquina utilizadas nas experimentações incluem um processador *Intel Core i5-10300H* de 2,50 GHz, 16 GB de RAM DDR4 em *Dual Channel*, uma placa de vídeo *NVIDIA GeForce GTX 1650* com 4 GB de memória e 1 TB de *SSD PCIe*. Para o desenvolvimento, foram usados: *Windows 10 Pro* como sistema operacional, *Ubuntu 22.03.3 LTS* no *WSL 2.0*, *Visual Studio Code (VSCode)* como *Integrated Development Environment (IDE)*, *Miniconda* como gerenciador de pacotes para *Python* e a linguagem *Python 3.8*. A plataforma de computação paralela *CUDA 11.8* foi utilizada para processamento em *Graphics Processing Unit (GPU)* e a biblioteca *PyTorch 11.8* para treinamento e inferência em visão computacional.

As principais bibliotecas para processamento de dados 3D incluem *MMDetection3D*, para detecção e segmentação de objetos em nuvens de pontos; e *Open3D*, manipulação de nuvens de pontos. Também foi utilizado o software *CloudCompare* para visualização, recorte e conversão de nuvens de pontos. É importante garantir a compatibilidade entre as versões do compilador *Python*, *CUDA* e *PyTorch*, seguindo as recomendações da *MMDetection3D*.

6 RESULTADOS

Nesta seção, será discutida a eficácia do método automatizado baseado no modelo *VoteNet* para a detecção e remoção de objetos indesejados em nuvens de pontos. A avaliação foi realizada por meio de observação direta durante o processo de implementação, com o intuito de verificar o desempenho do modelo na extração dos objetos.

6.0.1 Análise Comparativa das Remoções Automáticas e Manuais

Como resultado da inferência do modelo *VoteNet* para detecção das caixas de volume e da estratégia da remoção automatizada dos objetos detectados, serão apresentadas os comparativos obtidos pela remoção automática em relação à remoção manual, utilizando as métricas estabelecidas. Para cada quadrante, os resultados obtidos pela remoção automática serão apresentados à esquerda, enquanto os dados da remoção manual serão apresentados à direita.



Figura 16 – Comparação entre recortes automático e manual no primeiro quadrante.

As Figuras 16 e 17 ilustram o primeiro quadrante, comparando o recorte automático (à esquerda) e o manual (à direita). Os dados para este quadrante são um Índice de *Jaccard* de 0,8984 e uma Distância de *Hausdorff* de 0,8524, indicando alta similaridade e precisão entre as abordagens.

As Figuras 18 e 19 representam o segundo quadrante, comparando o recorte automático (à esquerda) e o manual (à direita). Os índices obtidos são um Índice de *Jaccard* de 0,8868 e uma Distância de *Hausdorff* de 0,8519, evidenciando precisão semelhante entre os métodos.

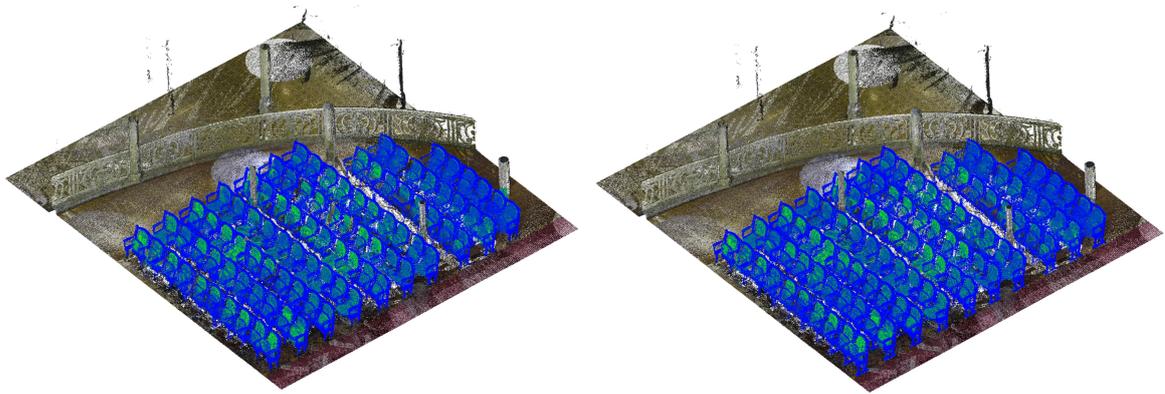


Figura 17 – Comparação da área removida no primeiro quadrante (automático e manual).

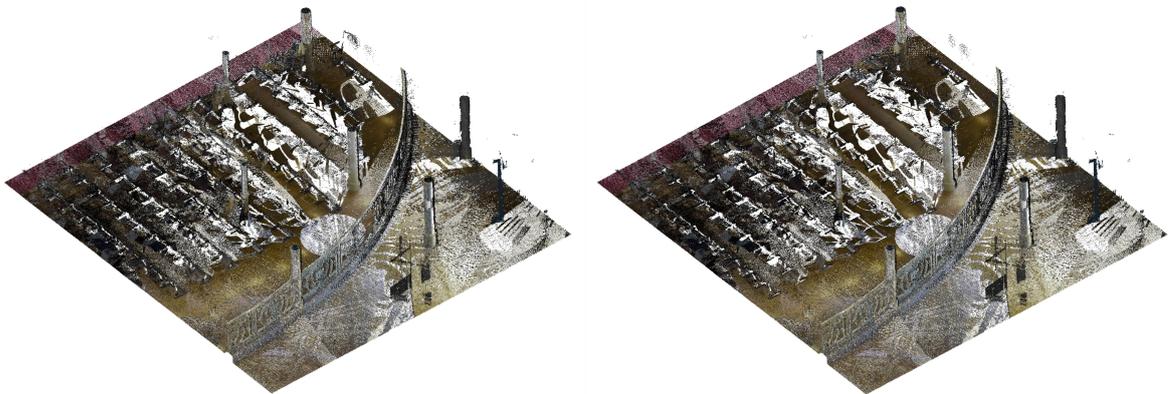


Figura 18 – Comparação entre recortes automático e manual no segundo quadrante.

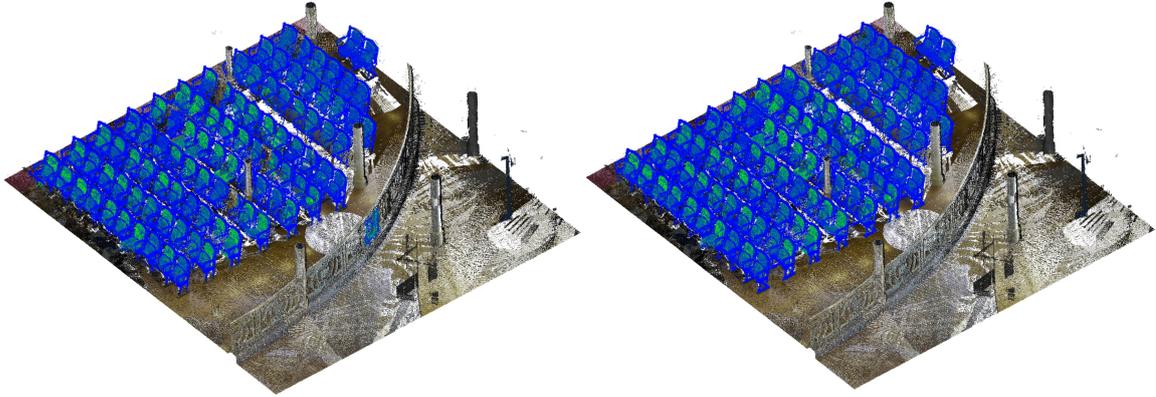


Figura 19 – Comparação da área removida no segundo quadrante (automático e manual).



Figura 20 – Comparação entre recortes automático e manual no terceiro quadrante.

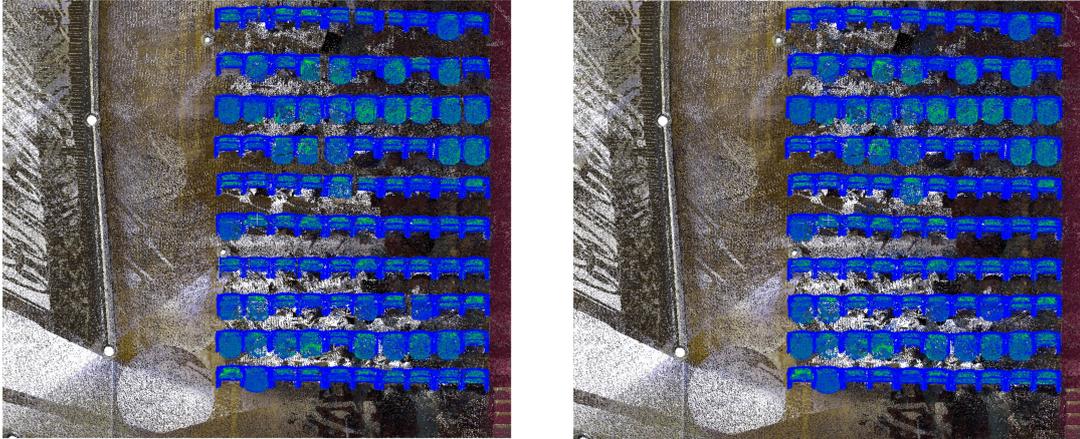


Figura 21 – Comparação da área removida no terceiro quadrante (automático e manual).

As Figuras 20 e 21 referem-se ao terceiro quadrante, apresentando uma comparação entre o recorte automático (à esquerda) e o manual (à direita). Os índices obtidos são um Índice de *Jaccard* de 0,8845 e uma Distância de *Hausdorff* de 0,8511, indicando boa correspondência entre as abordagens, com pequenas variações.

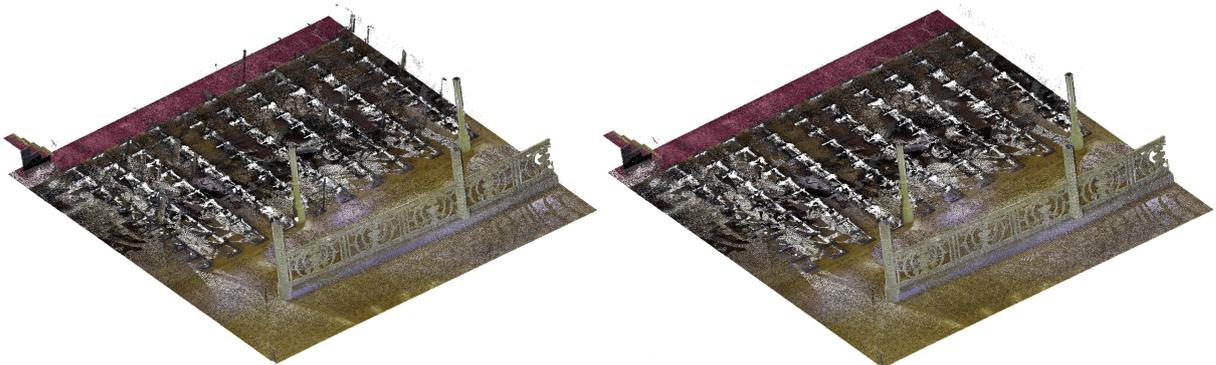


Figura 22 – Comparação entre recortes automático e manual no quarto quadrante.

As Figuras 22 e 23 estão relacionadas ao quarto quadrante, comparando o recorte automático (à esquerda) com o manual (à direita). Os índices obtidos mostram um Índice de *Jaccard* de 0,8977 e uma Distância de *Hausdorff* de 0,8505, refletindo alta similaridade e precisão entre os dois métodos.

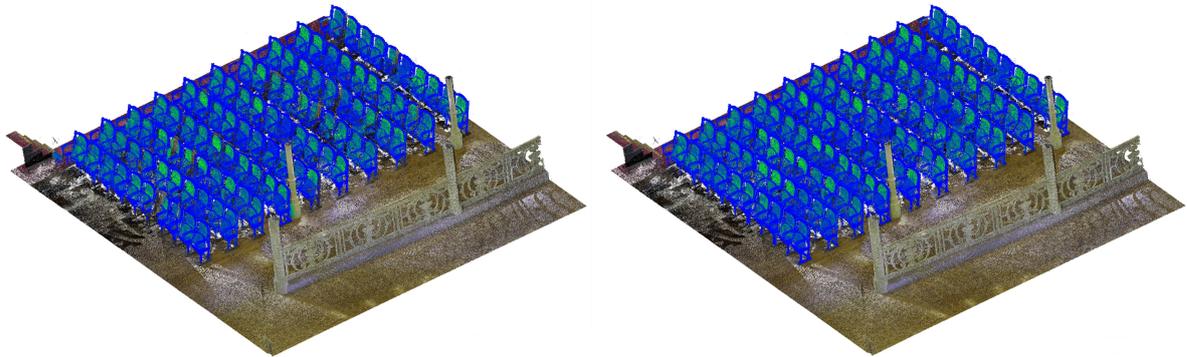


Figura 23 – Comparação da área removida no quarto quadrante (automático e manual).

6.1 Análise Comparativa da Redução da Quantidade de Pontos

No que se refere à remoção de objetos indesejados, o método demonstrou ser eficaz, gerando nuvens de pontos que se assemelham àquelas obtidas por meio de processos manuais, conforme indicado pelos índices de *Jaccard* e *Hausdorff*. Apesar do sucesso na remoção da maioria dos objetos, foram identificados alguns casos em que partes das cadeiras não foram completamente eliminadas, resultando na presença de resíduos.

Tabela 1 – Comparação da Redução das Nuvens (em quantidade de pontos) nos Quadrantes.

Quadrante	Nuvem Original	Remoção Manual	Remoção Automática	Redução Manual (%)	Redução Automática (%)
1	13.019.872	9.098.624	9.268.274	30.12	28.81
2	12.104.994	8.130.996	8.343.650	32.83	31.07
3	15.009.837	9.267.214	9.550.401	38.26	36.37
4	13.910.984	9.166.422	9.443.089	34.11	32.12

A Tabela 1, elaborada pelo autor deste trabalho, complementa as métricas estabelecidas e demonstra que a porcentagem de redução na quantidade de pontos é bastante semelhante quando se compara a extração manual com a extração automática. Em todos os quadrantes avaliados, a diferença entre os dois métodos permanece relativamente baixa, o que indica que o método automatizado foi capaz de remover uma quantidade de pontos similar à do processo manual.

Embora pequenas variações possam ser observadas, como no Quadrante 1, onde a redução manual foi de 30,12% e a automática de 28,81%, e no Quadrante 3, com uma redução de 38,26% manual contra 36,37% automática. Esses resultados reforçam a viabilidade do uso da

remoção automática para otimizar o processo, diminuindo significativamente o tempo e esforço necessário, com uma precisão comparável à intervenção manual, especialmente em situações com objetos mais simples e bem definidos.

6.2 Repositório do Método Proposto

Como resultado do desenvolvimento de um repositório que estabelece um ambiente de trabalho simplificado para a implementação do método proposto, foi criado um espaço que oferece instruções detalhadas para a preparação, manipulação e inferência das nuvens de pontos. O repositório *Point Cloud Object Detection* pode ser acessado na plataforma GitHub. Disponível em: <https://github.com/larebufc/point-cloud-object-detection>. Acesso em: 09 out. 2024.

7 CONCLUSÃO

O presente trabalho teve como objetivo o desenvolvimento de um método automatizado para a detecção e remoção de objetos indesejados em nuvens de pontos, utilizando o modelo de rede neural profunda *VoteNet*. A implementação e avaliação desse modelo evidenciaram, de maneira significativa, sua precisão na detecção de objetos, bem como sua eficácia na remoção de elementos indesejados. Entre as principais contribuições, destaca-se a redução do trabalho manual no processo de *scan to BIM*, o que resultou em uma aceleração significativa do processo de modelagem e em maior eficiência nas práticas de engenharia e arquitetura.

Além disso, quando comparado à remoção manual, o método automatizado proporcionou uma redução considerável no tempo e no esforço requeridos. A remoção manual, por sua vez, exigia um trabalho minucioso e cuidadoso para evitar a exclusão de partes desejadas, tornando o processo não apenas demorado, mas também suscetível a erros. Em contrapartida, a abordagem automatizada eliminou a necessidade de intervenções manuais extensivas, permitindo a remoção de objetos indesejados em um curto período. Assim, essa abordagem não apenas acelera o tempo de processamento, mas também oferece uma facilidade de uso considerável, garantindo resultados mais consistentes e reduzindo, portanto, a carga de trabalho sobre os operadores.

Contudo, apesar do desempenho satisfatório na detecção de objetos indesejados, o método apresenta algumas limitações. O algoritmo mostrou-se eficaz na identificação de estruturas como cadeiras, mas a variabilidade de objetos que o método pode detectar está restrita ao pré-treinamento disponível, resultando em uma gama limitada de objetos identificáveis. Diante dessa realidade, é essencial abordar essa limitação por meio do treinamento do modelo com novos conjuntos de dados, que incluam diferentes tipos de cadeiras para aprimorar a precisão na detecção, além de ampliar a capacidade de reconhecimento de outras classes de objetos.

Adicionalmente, o trabalho revelou desafios significativos na detecção de um maior número de objetos, incluindo a dificuldade em reconhecer diferentes classes e a presença de pontos residuais após o processo de remoção. Esses desafios, portanto, ressaltam a necessidade de melhorias em áreas que devem ser abordadas em pesquisas futuras, visando aprimorar tanto a eficácia quanto a abrangência do método proposto. Assim, a continuidade deste trabalho não só é desejável, mas necessária para garantir a evolução do método.

Dessa forma, este estudo representa um avanço significativo na automação da remoção de objetos indesejados em nuvens de pontos aplicadas à abordagem de BIM. Para maximizar

esse potencial, futuras pesquisas poderão se concentrar na criação de conjuntos de dados personalizados, abrangendo tipos específicos de objetos a serem removidos, além do re-treinamento do modelo para aprimorar a precisão na detecção de categorias, como cadeiras. A adoção de algoritmos mais avançados para a identificação de caixas delimitadoras de volumes também pode proporcionar melhorias substanciais. Portanto, tais avanços têm o potencial não apenas de expandir as aplicações do método desenvolvido, mas também de contribuir para o progresso contínuo na reconstrução e análise da arquitetura em BIM.

REFERÊNCIAS

- ALEXOPOULOS, P. **Semantic modeling for data**. Gravenstein Highway North, Sebastopol: O'Reilly Media, 2020.
- ATAZADEH, B.; RAJABIFARD, A.; ZHANG, Y.; BARZEGAR, M. Querying 3d cadastral information from bim models. **ISPRS International Journal of Geo-Information**, v. 8, n. 8, 2019. ISSN 2220-9964. Disponível em: <https://www.mdpi.com/2220-9964/8/8/329>.
- GIL, A. C. **Como Elaborar Projetos de Pesquisa**. 4ª. ed. São Paulo: Editora Atlas, 2002. P. 47.
- LOPES, I. T.; GROETELAARS, N. J.; CHECCUCCI, d. S.; AMORIM, A. L. Automação da modelagem bim a partir de nuvens de pontos: uma revisão sobre métodos e técnicas. **PARC Pesquisa em Arquitetura e Construção**, v. 14, n. 00, p. e023010, maio 2023. Disponível em: <https://periodicos.sbu.unicamp.br/ojs/index.php/parc/article/view/8669015>.
- MMDetection3D. **MMDetection3D: OpenMMLab next-generation platform for general 3D object detection**. 2020. <https://github.com/open-mmlab/mmdetection3d>.
- MONTEIRO, A.; MARTINS, J. P. Building information modeling (bim)-teoria e aplicação. In: **International Conference on Engineering UBI**. [S. l.: s. n.], 2011. v. 10.
- PELLIS, E.; MURTIYOSO, A.; MASIERO, A.; TUCCI, G.; BETTI, M.; GRUSSENMEYER, P. 2d to 3d label propagation for the semantic segmentation of heritage building point clouds. In: **ISPRS. XXIV ISPRS Congress “Imaging today, foreseeing tomorrow”, Commission II 2022 edition, 6–11 June 2022, Nice, France**. [S. l.], 2022. v. 43, p. 861–867.
- PELLIS, E.; MURTIYOSO, A.; MASIERO, A.; TUCCI, G.; BETTI, M.; GRUSSENMEYER, P. 2d to 3d label propagation for the semantic segmentation of heritage building point clouds. **The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**, XLIII-B2-2022, p. 861–867, 2022. Disponível em: <https://isprs-archives.copernicus.org/articles/XLIII-B2-2022/861/2022/>.
- PEPE, M.; COSTANTINO, D.; ALFIO, V. S.; RESTUCCIA, A. G.; PAPALINO, N. M. Scan to bim for the digital management and representation in 3d gis environment of cultural heritage site. **Journal of Cultural Heritage**, v. 50, p. 115–125, 2021. ISSN 1296-2074. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1296207421000881>.
- POLJANŠEK, M. Building information modelling (bim) standardization. **European Commission**, 2017.
- PärN, E.; EDWARDS, D.; SING, M. The building information modelling trajectory in facilities management: A review. **Automation in Construction**, v. 75, p. 45–55, 2017. ISSN 0926-5805. Disponível em: <https://www.sciencedirect.com/science/article/pii/S092658051630485X>.
- QI, C. R.; CHEN, X.; LITANY, O.; GUIBAS, L. J. Invotenet: Boosting 3d object detection in point clouds with image votes. In: **Proceedings of the IEEE/CVF conference on computer vision and pattern recognition**. [S. l.: s. n.], 2020. p. 4404–4413.
- QI, C. R.; LITANY, O.; HE, K.; GUIBAS, L. J. Deep hough voting for 3d object detection in point clouds. In: **Proceedings of the IEEE International Conference on Computer Vision**. [S. l.: s. n.], 2019.

QI, C. R.; YI, L.; SU, H.; GUIBAS, L. J. Pointnet++ deep hierarchical feature learning on point sets in a metric space. In: **Proceedings of the 31st International Conference on Neural Information Processing Systems**. [S. l.: s. n.], 2017. p. 5105–5114.

RUBENS, T. D. M. Digitalização do patrimônio cultural baseado em escaneamento de alta resolução. 2023.

SALARI, A.; DJAVADIFAR, A.; LIU, X.; NAJJARAN, H. Object recognition datasets and challenges: A review. **Neurocomputing**, v. 495, p. 129–152, 2022. ISSN 0925-2312. Disponível em: <https://www.sciencedirect.com/science/article/pii/S092523122200039X>. Acesso em: 10 de outubro de 2024. Disponível em: <https://doi.org/10.1016/j.neucom.2022.01.022>.

SONG, S.; LICHTENBERG, S.; XIAO, J. Sun rgb-d: A rgb-d scene understanding benchmark suite. In: **Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)**. [S. l.: s. n.], 2015.

VIEIRA, A. W. **Densidade de ocupação espacial como representação para comparação de nuvens de pontos e aplicações**. Tese (Tese de Doutorado) – Universidade Federal de Minas Gerais (UFMG), Belo Horizonte, 2012. Disponível em: <http://hdl.handle.net/1843/ESBF-92RN74>.

VIEIRA, M. M.; RIBEIRO, G.; PAULO, R.; BESSA, M.; SOUSA, F. R.; MOREIRA, E.; MESQUITA, E. Strategy for hbim implementation using high-resolution 3d architectural documentation based on laser scanning and photogrammetry of the José de Alencar Theatre. **Digital Applications in Archaeology and Cultural Heritage**, v. 30, p. e00287, 2023. ISSN 2212-0548. Disponível em: <https://www.sciencedirect.com/science/article/pii/S2212054823000322>.

WANG, C.; CHO, Y. K.; KIM, C. Automatic bim component extraction from point clouds of existing buildings for sustainability applications. **Automation in Construction**, v. 56, p. 1–13, 2015. ISSN 0926-5805. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0926580515000734>.

APÊNDICE A – CÓDIGO CONVERSÃO *.PLY* PARA *.BIN*

```
1 import numpy as np
2 import pandas as pd
3 from plyfile import PlyData
4
5 def convert_ply(input_path, output_path):
6     plydata = PlyData.read(input_path)
7     data = plydata.elements[0].data
8     data_pd = pd.DataFrame(data)
9
10    properties_to_use = ['x', 'y', 'z', 'nx', 'ny', 'nz']
11    num_points = data_pd.shape[0]
12    num_properties = len(properties_to_use)
13
14    data_np = np.zeros((num_points, num_properties), dtype=np.
15                       float64)
16
17    for i, name in enumerate(properties_to_use):
18        data_np[:, i] = data_pd[name]
19
20    data_np.astype(np.float32).tofile(output_path)
21
22 convert_ply('./input.ply', './output.bin')
```

Código-fonte 1 – Trecho de código adaptado de (MMDTECTION3D, 2020) para conversão arquivo com extensão de arquivo *.ply* para *.bin*.

APÊNDICE B – CÓDIGO PARA REMOÇÃO DE *BOUDING BOXES*

```

1 import json
2 import numpy as np
3 import open3d as o3d
4
5 point_cloud = o3d.io.read_point_cloud("input.ply")
6
7 with open("./outputs/preds/output.json", "r") as f:
8     data = json.load(f)
9
10 bboxes = data["bboxes_3d"]
11 scores = data["scores_3d"]
12
13 score_threshold = 0.9
14
15 filtered_bboxes = [bbox for bbox, score in zip(bboxes, scores)
16                  if score > score_threshold]
17
18 total_filtered_bboxes = len(filtered_bboxes)
19
20 points = np.asarray(point_cloud.points)
21 colors = np.asarray(point_cloud.colors)
22 normals = np.asarray(point_cloud.normals)
23
24 def are_points_inside_bbox(points, bbox):
25     center = np.array(bbox[0:3])
26     dim = np.array(bbox[3:6])
27     yaw = np.zeros(3)
28     yaw[2] = bbox[6]
29
30     rot_mat = o3d.geometry.get_rotation_matrix_from_xyz(yaw)
31     center[2] += dim[2] / 2
32
33     points_local = np.dot(points - center, rot_mat.T)
34     return np.all(np.abs(points_local) <= dim / 2, axis=1)

```

```
33
34 mask = np.ones(len(points), dtype=bool)
35
36 for idx, bbox in enumerate(filtered_bboxes):
37     print(f"Processando bounding box {idx + 1} de {
38         total_filtered_bboxes}")
39     mask &= ~are_points_inside_bbox(points, bbox)
40
41 filtered_points = points[mask]
42 filtered_colors = colors[mask]
43 filtered_normals = normals[mask]
44
45 new_point_cloud = o3d.geometry.PointCloud()
46 new_point_cloud.points = o3d.utility.Vector3dVector(
47     filtered_points)
48 new_point_cloud.colors = o3d.utility.Vector3dVector(
49     filtered_colors)
50 new_point_cloud.normals = o3d.utility.Vector3dVector(
51     filtered_normals)
52
53 o3d.io.write_point_cloud("result.ply", new_point_cloud)
54 print(f"Total de bounding boxes processadas: {
55     total_filtered_bboxes}")
```

Código-fonte 2 – Trecho de código para remoção automática das bounding boxes detectadas pela inferência.

APÊNDICE C – CÓDIGO PARA INFERÊNCIA DAS MÉTRICAS

```

1 import open3d as o3d
2 from scipy.spatial.distance import directed_hausdorff
3
4 def hausdorff_distance(source_points, target_points):
5     d_forward = directed_hausdorff(source_points, target_points)
6         [0]
7     d_backward = directed_hausdorff(target_points, source_points
8         ) [0]
9     return max(d_forward, d_backward)
10
11 def jaccard_index(source_points, target_points, voxel_size=0.02)
12     :
13     source_voxel = o3d.geometry.VoxelGrid.
14         create_from_point_cloud(o3d.geometry.PointCloud(o3d.
15             utility.Vector3dVector(source_points)), voxel_size)
16     target_voxel = o3d.geometry.VoxelGrid.
17         create_from_point_cloud(o3d.geometry.PointCloud(o3d.
18             utility.Vector3dVector(target_points)), voxel_size)
19
20     source_voxel_coords = set(tuple(v.grid_index) for v in
21         source_voxel.get_voxels())
22     target_voxel_coords = set(tuple(v.grid_index) for v in
23         target_voxel.get_voxels())
24
25     intersection = source_voxel_coords.intersection(
26         target_voxel_coords)
27     union = source_voxel_coords.union(target_voxel_coords)
28
29     return len(intersection) / len(union) if union else 1.0

```

Código-fonte 3 – Trecho de código para calculo das métricas do Índice de Jaccard e distância de Hausdorff.