

# Segmentação Semântica de Nuvens de Pontos LiDAR

Uma Abordagem Híbrida Combinando Processamento Geométrico com Aprendizado Profundo

Matheus Leonel de Andrade<sup>1</sup>

<sup>1</sup>Instituto de Informática  
Universidade Federal de Goiás

2025

# Sumário



1. Introdução
2. Fundamentação Teórica
3. Metodologia
4. Resultados
5. Conclusão e Trabalhos Futuros



# Introdução

# Motivação

- A percepção 3D do ambiente é um pilar fundamental para sistemas autônomos (veículos autônomos, robótica móvel).





# Motivação

- A percepção 3D do ambiente é um pilar fundamental para sistemas autônomos (veículos autônomos, robótica móvel).
- Nesse sentido os sensores LiDAR (Light Detection and Ranging) se destacam pois fornecem nuvens de pontos 3D, permitindo a análise detalhada da geometria do ambiente.



# Motivação

- A percepção 3D do ambiente é um pilar fundamental para sistemas autônomos (veículos autônomos, robótica móvel).
- Nesse sentido os sensores LiDAR (Light Detection and Ranging) se destacam pois fornecem nuvens de pontos 3D, permitindo a análise detalhada da geometria do ambiente.
- **Desafio:** Processar e interpretar o grande volume de dados das nuvens de pontos de forma eficiente, já que sensores LiDAR modernos podem capturar mais de 100.000 pontos por varredura.

# Problema de Pesquisa

- Algoritmos de segmentação e classificação de nuvens de pontos ainda possuem limitações em robustez, escalabilidade e generalização.





# Problema de Pesquisa

- Algoritmos de segmentação e classificação de nuvens de pontos ainda possuem limitações em robustez, escalabilidade e generalização.
- Abordagens baseadas em Deep Learning (e.g., PointNet, KPConv) alcançam uma boa acurácia, mas exigem alto custo computacional.



# Problema de Pesquisa

- Algoritmos de segmentação e classificação de nuvens de pontos ainda possuem limitações em robustez, escalabilidade e generalização.
- Abordagens baseadas em Deep Learning (e.g., PointNet, KPConv) alcançam uma boa acurácia, mas exigem alto custo computacional.
- A segmentação do solo e a clusterização de objetos são etapas críticas e desafiadoras.

# Problema de Pesquisa

- Algoritmos de segmentação e classificação de nuvens de pontos ainda possuem limitações em robustez, escalabilidade e generalização.
- Abordagens baseadas em Deep Learning (e.g., PointNet, KPConv) alcançam uma boa acurácia, mas exigem alto custo computacional.
- A segmentação do solo e a clusterização de objetos são etapas críticas e desafiadoras.
  - A identificação do solo, serve como base para detecção de obstáculos, e planejamento de rotas, por exemplo.



# Problema de Pesquisa

- Algoritmos de segmentação e classificação de nuvens de pontos ainda possuem limitações em robustez, escalabilidade e generalização.
- Abordagens baseadas em Deep Learning (e.g., PointNet, KPConv) alcançam uma boa acurácia, mas exigem alto custo computacional.
- A segmentação do solo e a clusterização de objetos são etapas críticas e desafiadoras.
  - A identificação do solo, serve como base para detecção de obstáculos, e planejamento de rotas, por exemplo.
  - Da mesma forma, a clusterização e classificação de objetos, é fundamental para a tomada de decisões seguras em sistemas autônomo.

# Objetivos

- **Objetivo Principal:** Desenvolver e avaliar um pipeline para segmentação e classificação semântica de nuvens de pontos LiDAR.





# Objetivos

- **Objetivo Principal:** Desenvolver e avaliar um pipeline para segmentação e classificação semântica de nuvens de pontos LiDAR.
- **Abordagem Híbrida:** Combinar a eficiência de métodos clássicos de processamento geométrico com o poder de generalização de técnicas de aprendizado profundo.



# Objetivos

- **Objetivo Principal:** Desenvolver e avaliar um pipeline para segmentação e classificação semântica de nuvens de pontos LiDAR.
- **Abordagem Híbrida:** Combinar a eficiência de métodos clássicos de processamento geométrico com o poder de generalização de técnicas de aprendizado profundo.
- **Hipótese:** Investigar se a redução da dimensionalidade, através da clusterização, preserva informação semântica suficiente para uma classificação eficaz, equilibrando acurácia e custo computacional.



# Fundamentação Teórica

# Conceitos Fundamentais



## Nuvens de Pontos 3D

- Conjunto de pontos com coordenadas  $(x, y, z)$ .

# Conceitos Fundamentais



## Nuvens de Pontos 3D

- Conjunto de pontos com coordenadas (x, y, z).
- Representação direta da geometria da cena.

# Conceitos Fundamentais



## Nuvens de Pontos 3D

- Conjunto de pontos com coordenadas (x, y, z).
- Representação direta da geometria da cena.
- Dados não estruturados, esparsos e com densidade variável.

# Conceitos Fundamentais



## Nuvens de Pontos 3D

- Conjunto de pontos com coordenadas (x, y, z).
- Representação direta da geometria da cena.
- Dados não estruturados, esparsos e com densidade variável.

## Segmentação e Clusterização

# Conceitos Fundamentais



## Nuvens de Pontos 3D

- Conjunto de pontos com coordenadas (x, y, z).
- Representação direta da geometria da cena.
- Dados não estruturados, esparsos e com densidade variável.

## Segmentação e Clusterização

- **Segmentação:** Atribuir um rótulo semântico a cada ponto (e.g., solo, carro, prédio).

# Conceitos Fundamentais



## Nuvens de Pontos 3D

- Conjunto de pontos com coordenadas (x, y, z).
- Representação direta da geometria da cena.
- Dados não estruturados, esparsos e com densidade variável.

## Segmentação e Clusterização

- **Segmentação:** Atribuir um rótulo semântico a cada ponto (e.g., solo, carro, prédio).
- **Clusterização:** Agrupar pontos próximos que pertencem ao mesmo objeto físico.

# Principais Desafios



## Características dos Dados

- Irregularidade (sem estrutura de grade)

# Principais Desafios



## Características dos Dados

- Irregularidade (sem estrutura de grade)
- Densidade variável com distância

# Principais Desafios



## Características dos Dados

- Irregularidade (sem estrutura de grade)
- Densidade variável com distância
- Grande volume de dados

# Principais Desafios



## Características dos Dados

- Irregularidade (sem estrutura de grade)
- Densidade variável com distância
- Grande volume de dados

## Limitações Práticas

# Principais Desafios



## Características dos Dados

- Irregularidade (sem estrutura de grade)
- Densidade variável com distância
- Grande volume de dados

## Limitações Práticas

- Oclusões e dados incompletos

# Principais Desafios



## Características dos Dados

- Irregularidade (sem estrutura de grade)
- Densidade variável com distância
- Grande volume de dados

## Limitações Práticas

- Oclusões e dados incompletos
- Processamento em tempo real

# Principais Desafios



## Características dos Dados

- Irregularidade (sem estrutura de grade)
- Densidade variável com distância
- Grande volume de dados

## Limitações Práticas

- Oclusões e dados incompletos
- Processamento em tempo real
- Generalização entre ambientes

# Características dos Algoritmos Utilizados

- Ground Plane Fitting (GPF) - Segmentação do Solo:

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares
- **Scan Line Run (SLR)** - Clusterização:

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares
- **Scan Line Run (SLR)** - Clusterização:
  - Explora estrutura sequencial das scanlines LiDAR

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares
- **Scan Line Run (SLR)** - Clusterização:
  - Explora estrutura sequencial das scanlines LiDAR
  - Agrupa pontos por proximidade dentro de cada linha

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares
- **Scan Line Run (SLR)** - Clusterização:
  - Explora estrutura sequencial das scanlines LiDAR
  - Agrupa pontos por proximidade dentro de cada linha
  - Fusão de clusters entre scanlines adjacentes

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares
- **Scan Line Run (SLR)** - Clusterização:
  - Explora estrutura sequencial das scanlines LiDAR
  - Agrupa pontos por proximidade dentro de cada linha
  - Fusão de clusters entre scanlines adjacentes
- **PointNet** - Classificação Semântica:

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares
- **Scan Line Run (SLR)** - Clusterização:
  - Explora estrutura sequencial das scanlines LiDAR
  - Agrupa pontos por proximidade dentro de cada linha
  - Fusão de clusters entre scanlines adjacentes
- **PointNet** - Classificação Semântica:
  - Processa coordenadas 3D diretamente (sem projeção)

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares
- **Scan Line Run (SLR)** - Clusterização:
  - Explora estrutura sequencial das scanlines LiDAR
  - Agrupa pontos por proximidade dentro de cada linha
  - Fusão de clusters entre scanlines adjacentes
- **PointNet** - Classificação Semântica:
  - Processa coordenadas 3D diretamente (sem projeção)
  - Invariante a permutações

# Características dos Algoritmos Utilizados

- **Ground Plane Fitting (GPF)** - Segmentação do Solo:
  - Modela solo como plano estimado por SVD
  - Limitado a terrenos regulares
- **Scan Line Run (SLR)** - Clusterização:
  - Explora estrutura sequencial das scanlines LiDAR
  - Agrupa pontos por proximidade dentro de cada linha
  - Fusão de clusters entre scanlines adjacentes
- **PointNet** - Classificação Semântica:
  - Processa coordenadas 3D diretamente (sem projeção)
  - Invariante a permutações
  - T-Net para alinhamento espacial



# Metodología

# Pipeline Proposto



1. Nuvem de Pontos Bruta (Entrada)



2. Pré-processamento

2.1 GPF (Segmentação do Solo)



2.2 SLR (Clusterização)



2.3 Extração de Características



3. Classificação com PointNet (Saída)

# Dataset: SemanticKITTI

- Benchmark de referência para percepção 3D em direção autônoma.



# Dataset: SemanticKITTI

- Benchmark de referência para percepção 3D em direção autônoma.
- Extensão do KITTI Vision Benchmark.



# Dataset: SemanticKITTI

- Benchmark de referência para percepção 3D em direção autônoma.
- Extensão do KITTI Vision Benchmark.
- Fornece anotações semânticas densas para nuvens de pontos capturadas por um LiDAR Velodyne HDL-64E.

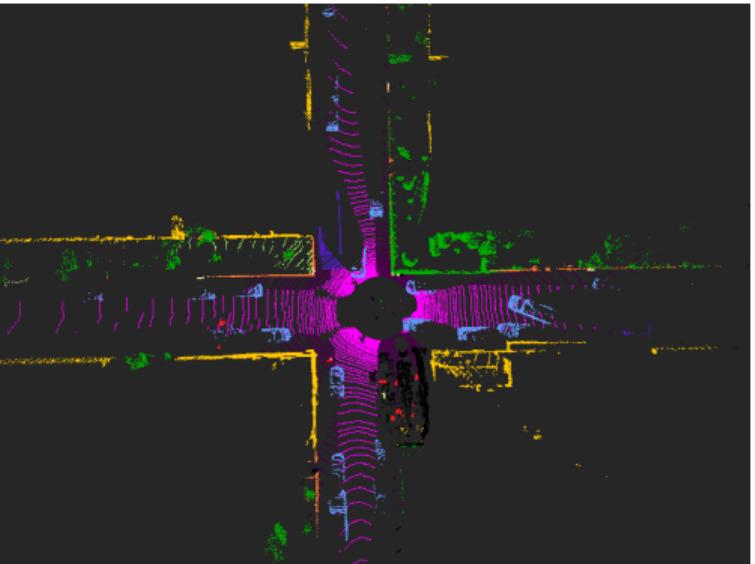


# Dataset: SemanticKITTI



- Benchmark de referência para percepção 3D em direção autônoma.
- Extensão do KITTI Vision Benchmark.
- Fornece anotações semânticas densas para nuvens de pontos capturadas por um LiDAR Velodyne HDL-64E.
- Foco do trabalho em 20 classes semânticas relevantes para navegação.

# Dataset: SemanticKITTI



**Figure 1:** Exemplo de uma cena do SemanticKITTI com rótulos semânticos.  
(sequencia 7 - frame 993)

# Pré-processamento: Segmentação do Solo (GPF)



## Objetivo

Identificar os pontos pertencentes ao solo.

## Algoritmo

Ground Plane Fitting (GPF).

# Pré-processamento: Segmentação do Solo (GPF)



## Objetivo

Identificar os pontos pertencentes ao solo.

## Algoritmo

Ground Plane Fitting (GPF).

## Como funciona

# Pré-processamento: Segmentação do Solo (GPF)



## Objetivo

Identificar os pontos pertencentes ao solo.

## Algoritmo

Ground Plane Fitting (GPF).

## Como funciona

1. Seleção dos pontos sementes.

# Pré-processamento: Segmentação do Solo (GPF)



## Objetivo

Identificar os pontos pertencentes ao solo.

## Algoritmo

Ground Plane Fitting (GPF).

## Como funciona

1. Seleção dos pontos sementes.
2. Estimativa do plano: calcula o LRP, em seguida aplica SVD para obter o vetor normal do plano.

# Pré-processamento: Segmentação do Solo (GPF)



## Objetivo

Identificar os pontos pertencentes ao solo.

## Algoritmo

Ground Plane Fitting (GPF).

## Como funciona

1. Seleção dos pontos sementes.
2. Estimativa do plano: calcula o LRP, em seguida aplica SVD para obter o vetor normal do plano.
3. Refinamento iterativo: inclui pontos próximos do plano (< limiar) em iterações sucessivas, e repete o item 2.

# Pré-processamento: Segmentação do Solo (GPF)



## Objetivo

Identificar os pontos pertencentes ao solo.

## Algoritmo

Ground Plane Fitting (GPF).

## Como funciona

1. Seleção dos pontos sementes.
2. Estimativa do plano: calcula o LRP, em seguida aplica SVD para obter o vetor normal do plano.
3. Refinamento iterativo: inclui pontos próximos do plano (< limiar) em iterações sucessivas, e repete o item 2.
4. Classificação final: pontos do solo recebem label específico.

# Pré-processamento: Clusterização de Objetos (SLR)



## Objetivo

Agrupar os pontos remanescentes (não-solo) em clusters que representam objetos.

## Algoritmo

Scan Line Run (SLR).

# Pré-processamento: Clusterização de Objetos (SLR)



## Objetivo

Agrupar os pontos remanescentes (não-solo) em clusters que representam objetos.

## Algoritmo

Scan Line Run (SLR).

## Como funciona

# Pré-processamento: Clusterização de Objetos (SLR)



## Objetivo

Agrupar os pontos remanescentes (não-solo) em clusters que representam objetos.

## Algoritmo

Scan Line Run (SLR).

## Como funciona

1. Agrupamento inicial por scanline preservando estrutura sequencial LiDAR.

# Pré-processamento: Clusterização de Objetos (SLR)



## Objetivo

Agrupar os pontos remanescentes (não-solo) em clusters que representam objetos.

## Algoritmo

Scan Line Run (SLR).

## Como funciona

1. Agrupamento inicial por scanline preservando estrutura sequencial LiDAR.
2. Identificação de "runs": segmentos contínuos com distância < limiar.

# Pré-processamento: Clusterização de Objetos (SLR)



## Objetivo

Agrupar os pontos remanescentes (não-solo) em clusters que representam objetos.

## Algoritmo

Scan Line Run (SLR).

## Como funciona

1. Agrupamento inicial por scanline preservando estrutura sequencial LiDAR.
2. Identificação de "runs": segmentos contínuos com distância < limiar.
3. Propagação de rótulos entre scanlines adjacentes (< limiar).

# Pré-processamento: Clusterização de Objetos (SLR)



## Objetivo

Agrupar os pontos remanescentes (não-solo) em clusters que representam objetos.

## Algoritmo

Scan Line Run (SLR).

## Como funciona

1. Agrupamento inicial por scanline preservando estrutura sequencial LiDAR.
2. Identificação de "runs": segmentos contínuos com distância < limiar.
3. Propagação de rótulos entre scanlines adjacentes (< limiar).
4. Resolução de equivalências via tabela para fundir rótulos duplicados.

# Pré-processamento: Clusterização de Objetos (SLR)



## Objetivo

Agrupar os pontos remanescentes (não-solo) em clusters que representam objetos.

## Algoritmo

Scan Line Run (SLR).

## Como funciona

1. Agrupamento inicial por scanline preservando estrutura sequencial LiDAR.
2. Identificação de "runs": segmentos contínuos com distância < limiar.
3. Propagação de rótulos entre scanlines adjacentes (< limiar).
4. Resolução de equivalências via tabela para fundir rótulos duplicados.
5. Atribuição final: cada cluster recebe identificador único.

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):
  - Centroide (x, y, z) - posição central

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):
  - Centroide (x, y, z) - posição central
  - Bounding box: pontos mín/máx (x, y, z)

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):
  - Centroide (x, y, z) - posição central
  - Bounding box: pontos mín/máx (x, y, z)
- Distribuição espacial (15 valores):

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):
  - Centroide (x, y, z) - posição central
  - Bounding box: pontos mín/máx (x, y, z)
- Distribuição espacial (15 valores):
  - Histogramas X, Y, Z (5 bins cada)

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):
  - Centroide (x, y, z) - posição central
  - Bounding box: pontos mín/máx (x, y, z)
- Distribuição espacial (15 valores):
  - Histogramas X, Y, Z (5 bins cada)
  - Capturam forma e orientação do cluster

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):
  - Centroide (x, y, z) - posição central
  - Bounding box: pontos mín/máx (x, y, z)
- Distribuição espacial (15 valores):
  - Histogramas X, Y, Z (5 bins cada)
  - Capturam forma e orientação do cluster
- Rótulos (2 valores):

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):
  - Centroide ( $x, y, z$ ) - posição central
  - Bounding box: pontos mín/máx ( $x, y, z$ )
- Distribuição espacial (15 valores):
  - Histogramas X, Y, Z (5 bins cada)
  - Capturam forma e orientação do cluster
- Rótulos (2 valores):
  - True label (anotação ground truth)

# Pré-processamento: Extração de Características



## Objetivo

Extrair vetor de características para cada cluster gerado pelo SLR.

## Saída

Vetor de 24 características + 2 rótulos por cluster.

## Características Extraídas

- Posição espacial (9 valores):
  - Centroide (x, y, z) - posição central
  - Bounding box: pontos mín/máx (x, y, z)
- Distribuição espacial (15 valores):
  - Histogramas X, Y, Z (5 bins cada)
  - Capturam forma e orientação do cluster
- Rótulos (2 valores):
  - True label (anotação ground truth)
  - Predicted label (placeholder para classificação)

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

## Modificações Implementadas

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

## Modificações Implementadas

- Expansão da capacidade:

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

## Modificações Implementadas

- Expansão da capacidade:
  - Maior largura (mais canais)

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

## Modificações Implementadas

- Expansão da capacidade:
  - Maior largura (mais canais)
  - Maior profundidade (mais camadas)

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

## Modificações Implementadas

- Expansão da capacidade:
  - Maior largura (mais canais)
  - Maior profundidade (mais camadas)
  - compensar a redução da dimensionalidade

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

## Modificações Implementadas

- Expansão da capacidade:
  - Maior largura (mais canais)
  - Maior profundidade (mais camadas)
  - compensar a redução da dimensionalidade
  - Redução significativa na dimensionalidade mitiga custo computacional da expansão do modelo.

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

## Modificações Implementadas

- Expansão da capacidade:
  - Maior largura (mais canais)
  - Maior profundidade (mais camadas)
  - compensar a redução da dimensionalidade
  - Redução significativa na dimensionalidade mitiga custo computacional da expansão do modelo.
- Adaptação da entrada para aceitar as 24 características

# Classificação: PointNet Modificada



## Entrada

Vetores de 24 características

## Saida

Clusters classificados em 20 categorias semânticas.

## Modificações Implementadas

- Expansão da capacidade:
  - Maior largura (mais canais)
  - Maior profundidade (mais camadas)
  - compensar a redução da dimensionalidade
  - Redução significativa na dimensionalidade mitiga custo computacional da expansão do modelo.
- Adaptação da entrada para aceitar as 24 características
- Normalização atualizada de BatchNorm para InstanceNorm1d



# Resultados

# Saída do GPF

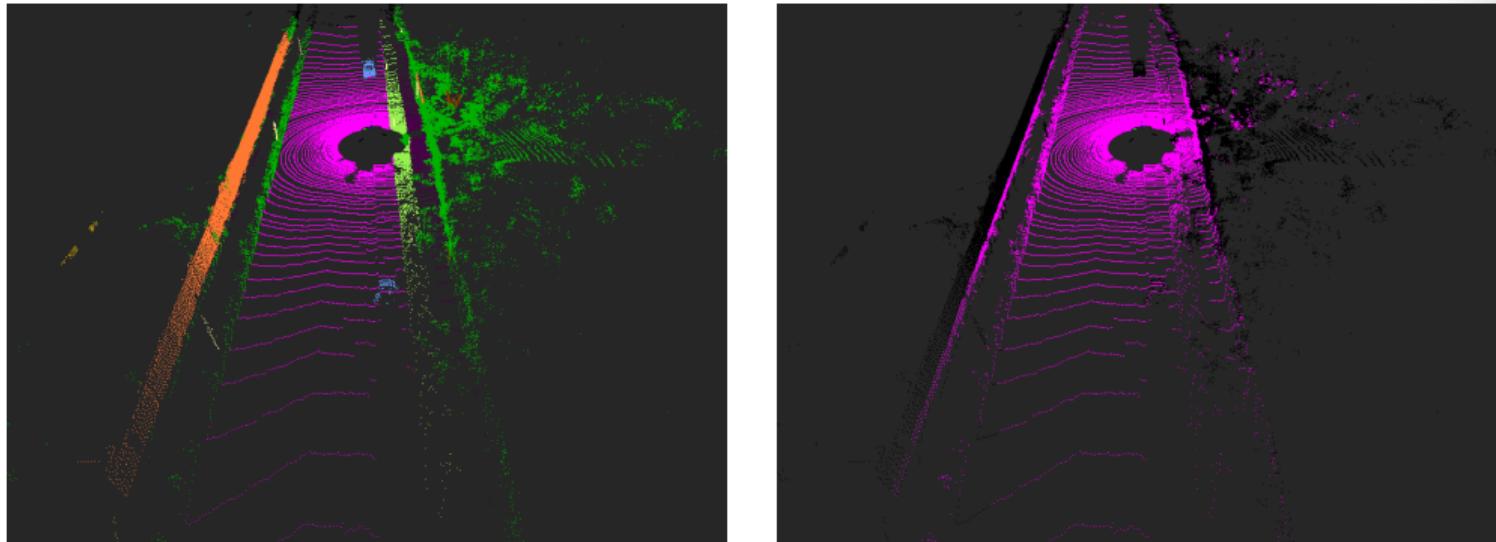


Figure 2: Comparação entre rótulos reais (esquerda) segmentação predita pelo GPF (direita), sequencia 4 - frame 75.

# Saida do SLR

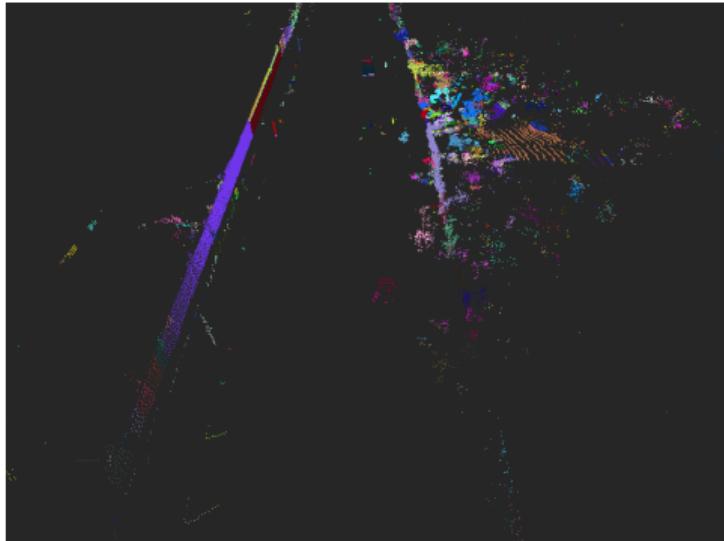


Figure 3: Saída do SLR, sequencia 4 - frame 75.

# Analise dos Clusters Inconsistentes

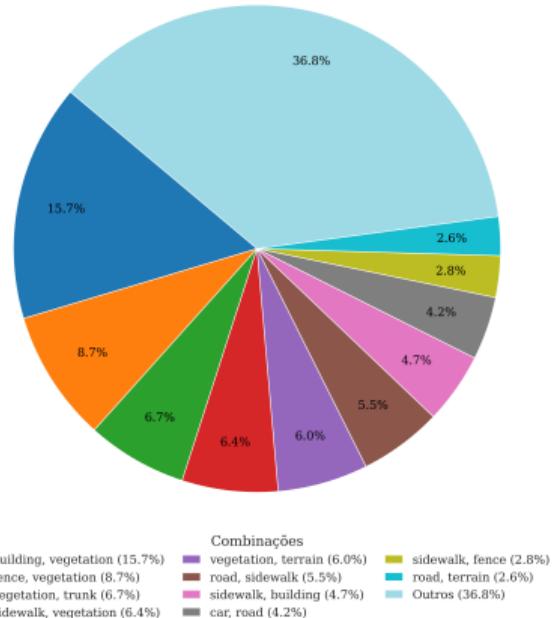


Figure 4: Combinações mais frequentes dos clusters inconsistentes, sequencia 0.

# Analise dos Clusters Inconsistentes

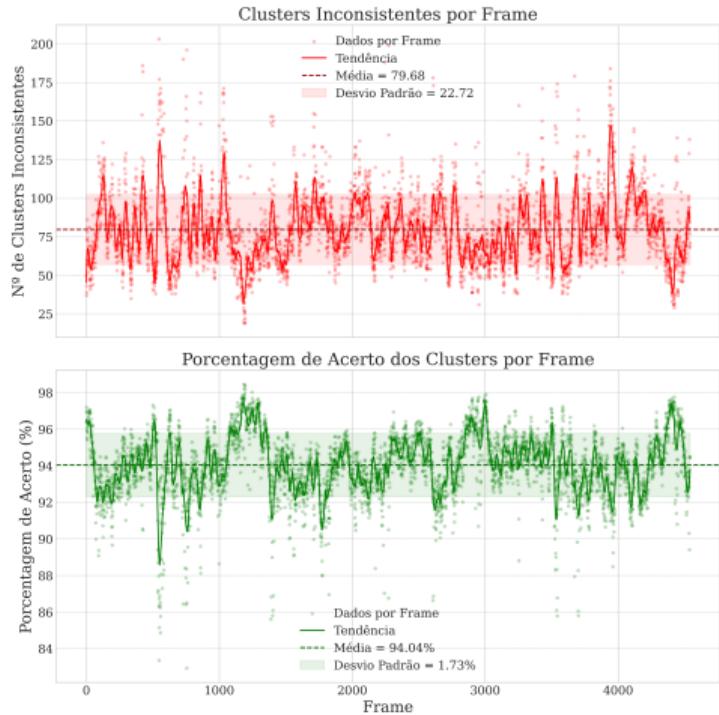


Figure 5: Gráfico de analise dos clusters inconsistentes, sequencia 0.

# Clusters após Extração de Características

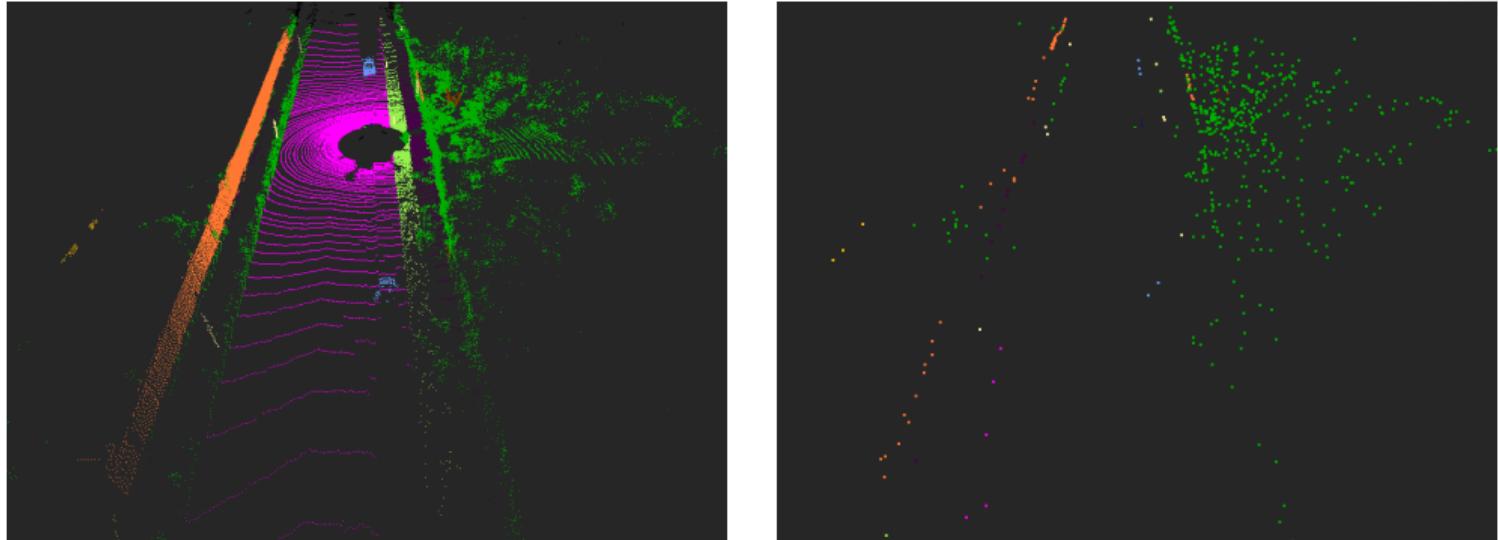


Figure 6: Comparação dos dados originais (esquerda) com os clusters após extração de características (direita), sequencia 4 - frame 75.

# Avaliação da Redução de Dimensionalidade

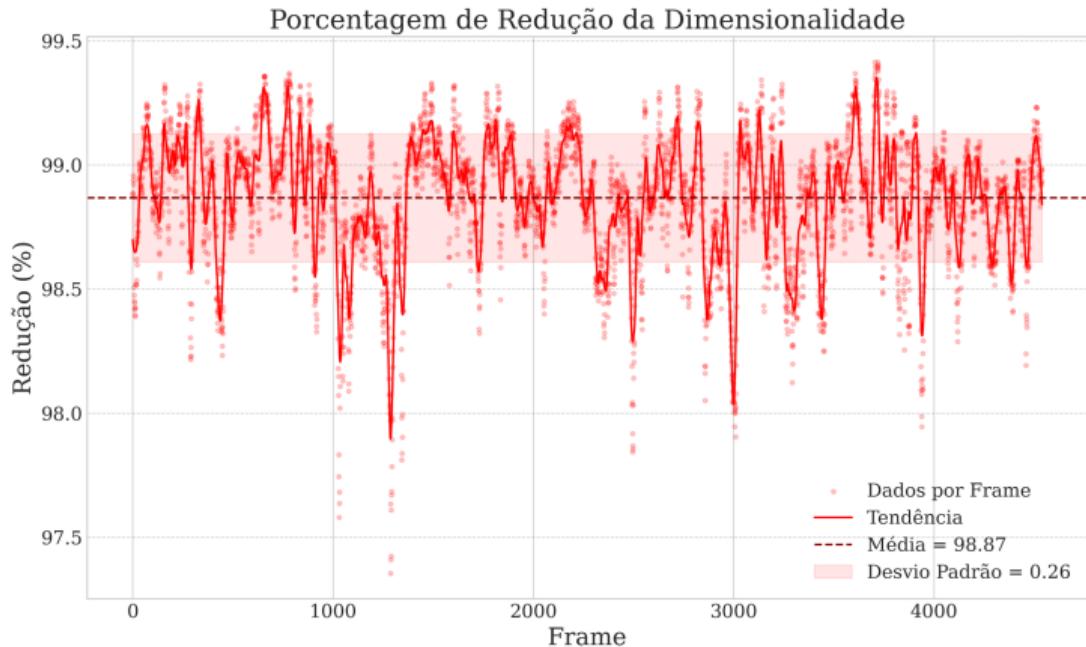


Figure 7: Gráfico de análise da redução de dimensionalidade, sequencia 0.

# Classificação

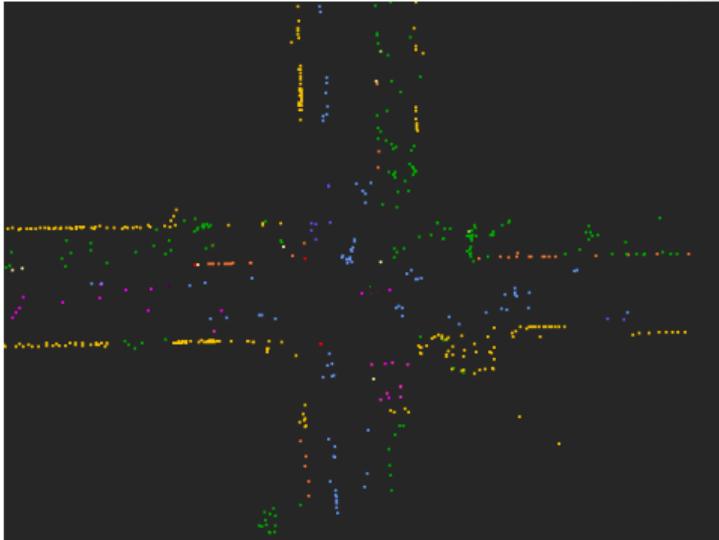
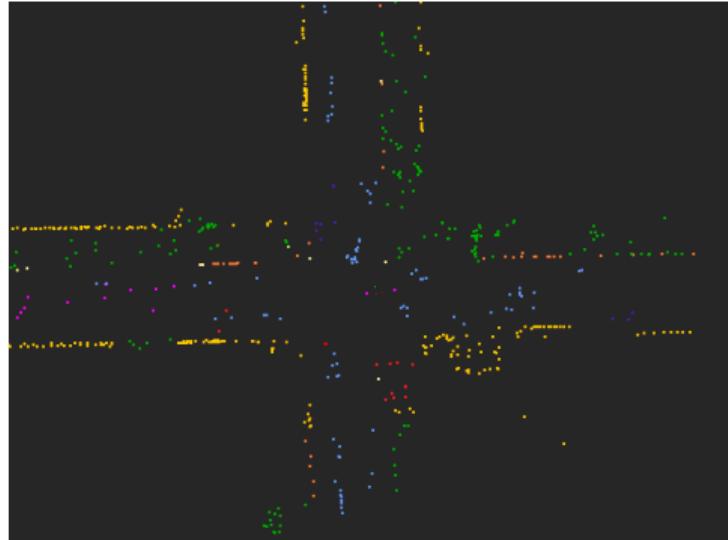


Figure 8: Comparação entre cluster obtidos (esquerda) com a classificação do modelo (direita), sequencia 7 - frame 993.

# Desempenho do Modelo

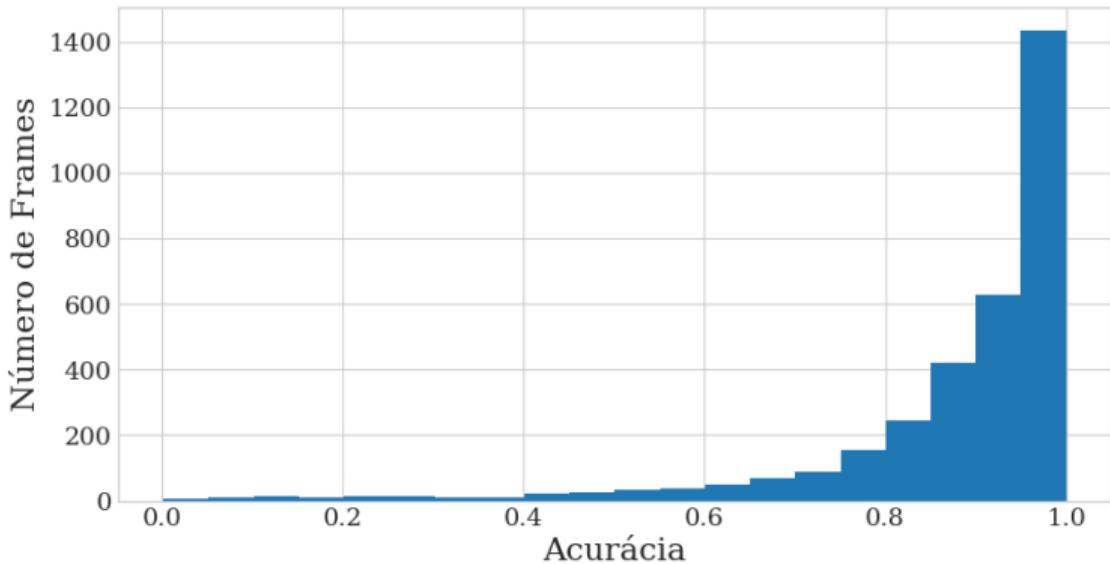


Figure 9: Gráfico da distribuição de acu, sequencia 0.



# Conclusão e Trabalhos Futuros

# Conclusões

- O trabalho validou um pipeline híbrido que combina processamento geométrico clássico com aprendizado profundo para segmentação semântica de nuvens de pontos.





# Conclusões

- O trabalho validou um pipeline híbrido que combina processamento geométrico clássico com aprendizado profundo para segmentação semântica de nuvens de pontos.
- Os resultados (redução de 98.8%, acurácia de 93.3%) confirmam a viabilidade da metodologia para aplicações que demandam eficiência.



# Conclusões

- O trabalho validou um pipeline híbrido que combina processamento geométrico clássico com aprendizado profundo para segmentação semântica de nuvens de pontos.
- Os resultados (redução de 98.8%, acurácia de 93.3%) confirmam a viabilidade da metodologia para aplicações que demandam eficiência.

## Principal Contribuição

Demonstração de que é possível comprimir drasticamente nuvens de pontos LiDAR em representações de clusters, preservando a informação semântica essencial para a classificação.

# Limitações e Trabalhos Futuros

## Limitações

- Propagação de erros: falhas na segmentação do solo afetam a clusterização e classificação.

## Trabalhos Futuros



# Limitações e Trabalhos Futuros

## Limitações

- Propagação de erros: falhas na segmentação do solo afetam a clusterização e classificação.
- A clusterização pode fragmentar objetos grandes ou complexos.

## Trabalhos Futuros



# Limitações e Trabalhos Futuros

## Limitações

- Propagação de erros: falhas na segmentação do solo afetam a clusterização e classificação.
- A clusterização pode fragmentar objetos grandes ou complexos.
- Validação restrita ao dataset SemanticKITTI.

## Trabalhos Futuros



# Limitações e Trabalhos Futuros

## Limitações

- Propagação de erros: falhas na segmentação do solo afetam a clusterização e classificação.
- A clusterização pode fragmentar objetos grandes ou complexos.
- Validação restrita ao dataset SemanticKITTI.
- Implementação não otimizada para tempo real.

## Trabalhos Futuros



# Limitações e Trabalhos Futuros

## Limitações

- Propagação de erros: falhas na segmentação do solo afetam a clusterização e classificação.
- A clusterização pode fragmentar objetos grandes ou complexos.
- Validação restrita ao dataset SemanticKITTI.
- Implementação não otimizada para tempo real.

## Trabalhos Futuros

- Aprimorar a segmentação do solo (modelos mais flexíveis).



# Limitações e Trabalhos Futuros

## Limitações

- Propagação de erros: falhas na segmentação do solo afetam a clusterização e classificação.
- A clusterização pode fragmentar objetos grandes ou complexos.
- Validação restrita ao dataset SemanticKITTI.
- Implementação não otimizada para tempo real.

## Trabalhos Futuros

- Aprimorar a segmentação do solo (modelos mais flexíveis).
- Enriquecer as características dos clusters (e.g., intensidade, contexto).

# Limitações e Trabalhos Futuros



## Limitações

- Propagação de erros: falhas na segmentação do solo afetam a clusterização e classificação.
- A clusterização pode fragmentar objetos grandes ou complexos.
- Validação restrita ao dataset SemanticKITTI.
- Implementação não otimizada para tempo real.

## Trabalhos Futuros

- Aprimorar a segmentação do solo (modelos mais flexíveis).
- Enriquecer as características dos clusters (e.g., intensidade, contexto).
- Otimizar o pipeline para hardware embarcado.

# Limitações e Trabalhos Futuros



## Limitações

- Propagação de erros: falhas na segmentação do solo afetam a clusterização e classificação.
- A clusterização pode fragmentar objetos grandes ou complexos.
- Validação restrita ao dataset SemanticKITTI.
- Implementação não otimizada para tempo real.

## Trabalhos Futuros

- Aprimorar a segmentação do solo (modelos mais flexíveis).
- Enriquecer as características dos clusters (e.g., intensidade, contexto).
- Otimizar o pipeline para hardware embarcado.
- Validar em outras plataformas robóticas e com diferentes sensores.

# Obrigado!

## Perguntas?