

Segmentação Semântica Integrada ao ROS2 para Aplicações em Veículos Autônomos

Oziel F. da Silva¹ (IC), Giovani Bernardes Vitor (PQ)¹

¹ Universidade Federal de Itajubá - Campus Itabira

Palavras-chave: Autonomia Veicular, Inteligência Artificial, Segmentação Semântica, Sistemas Embarcados, Visão Computacional.

Introdução

Este projeto teve como objetivo desenvolver um sistema de segmentação semântica integrado ao sistema operacional de robôs ROS2 (do inglês, *Robot Operating System 2*), com possíveis aplicações em veículos autônomos. A segmentação semântica desempenha um papel essencial ao classificar visualmente os elementos do ambiente, permitindo a identificação de obstáculos e contribuindo para a segurança e navegação autônoma em cenários urbanos (Garcia-Garcia et al., 2018).

A segmentação semântica é uma técnica fundamental na visão computacional, na qual cada pixel de uma imagem é classificado em uma categoria específica, como pedestres, sinais de trânsito ou veículos. Isso resulta em uma compreensão detalhada dos objetos e suas fronteiras dentro de uma cena, proporcionando aos sistemas autônomos uma base sólida para a tomada de decisões. Para aumentar a eficácia dos modelos de segmentação semântica, utiliza-se um *encoder*, que transforma os dados de entrada em representações compactas, capturando as características mais relevantes da imagem. O processo de treinamento dessas redes neurais é otimizado pelo uso de um agendador de taxa de aprendizado, que ajusta dinamicamente esse hiperparâmetro, determinando a intensidade das atualizações dos pesos da rede. Esse ajuste dinâmico melhora a convergência do modelo, evitando que ele fique preso em soluções subótimas.

O principal objetivo desta pesquisa foi criar um sistema eficiente de segmentação semântica, capaz de operar em tempo real em plataformas robóticas móveis. Para isso, utilizou-se o banco de dados *Cityscapes*, composto por imagens urbanas anotadas em 20 classes, o que proporciona um ambiente realista para treinamento e validação (Cordts et al., 2016). O modelo de segmentação escolhido foi o *DeepLabV3+*, que incorpora o *encoder EfficientNet-B4*, auxiliado por pesos pré-treinados no *ImageNet*, uma base de dados amplamente utilizada para *transfer learning*. Este modelo foi escolhido por seu equilíbrio entre alta precisão e desempenho computacional eficiente (Chen et al., 2018; Tan & Le, 2019).

A escolha pela implementação desse sistema se

justifica pela crescente demanda por soluções autônomas no setor de transportes.

O método adotado para o desenvolvimento envolveu o treinamento supervisionado do modelo de segmentação, utilizando técnicas de aprendizado profundo. O treinamento foi realizado ao longo de várias épocas, ajustando gradualmente os pesos da rede neural para otimizar sua performance. Ferramentas como *PyTorch* (Paszke et al., 2019), *TensorFlow* (Abadi et al., 2016), *OpenCV* (Bradski, 2000), *Albumentations* e *SegmentationModelsPytorch* foram empregadas para o desenvolvimento do modelo e sua integração com o ROS2. A arquitetura do sistema consistiu em dois nós: um nó responsável por receber imagens em tempo real de uma câmera, enquanto o segundo nó aplicava o modelo de segmentação para processá-las.

Para avaliar a eficácia do sistema, utilizou-se a métrica IoU (do inglês, *Intersection over Union*), que mede a precisão da segmentação comparando a área prevista pelo modelo com a área real das classes anotadas. A métrica IoU é amplamente utilizada em problemas de segmentação semântica, pois oferece uma medida clara da sobreposição entre a segmentação predita e a real. Os resultados foram apresentados em interfaces gráficas, permitindo uma análise tanto qualitativa quanto quantitativa do desempenho.

Durante o desenvolvimento, alguns desafios que foram enfrentados incluíram a necessidade de otimizar o sistema para processamento em tempo real e gerenciar os recursos computacionais limitados das plataformas embarcadas. Para contornar esses problemas, foram aplicadas técnicas de compressão de modelo e ajustes nos parâmetros de inferência, que reduziram o consumo de memória e tempo de execução sem sacrificar a precisão. Este projeto estabeleceu as bases para futuras implementações de comportamentos autônomos em veículos elétricos, utilizando a segmentação semântica como fonte de telemetria visual.

Metodologia

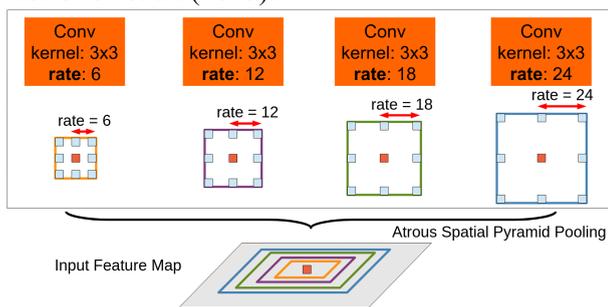
Este projeto utilizou uma abordagem baseada em aprendizado profundo para desenvolver um sistema de segmentação semântica integrado ao ROS2, visando

aplicações futuras em veículos autônomos. A seguir, são descritos os principais componentes, desde a configuração do modelo e da base de dados até o treinamento e a avaliação.

O modelo de segmentação semântica escolhido foi o *DeepLabV3+*, devido à sua arquitetura eficiente de *encoder-decoder* com convoluções separáveis e dilatadas, como mostrado na Figura 1 (atrous).

Figura 1 – Atrous Spatial Pyramid Pooling (ASPP).

Fonte: Chen et al. (2017).



Esta estrutura, sugerida por Chen et al. (2017), provou ser uma alternativa sólida para a segmentação semântica em contextos diversos, como no caso de veículos autônomos, onde é importante reconhecer várias escalas de contexto. O *encoder* escolhido foi o *EfficientNet-B4*, uma arquitetura proposta por Tan & Le (2019), que oferece uma abordagem escalável para redes neurais convolucionais. E a base de dados, do *Cityscapes*. A Tabela 1 mostra a relação entre hiperparâmetros da rede.

Tabela 1 – Parâmetros de Treinamento dos Modelos.

Fonte: Autor.

Parâmetro	Valor Usado
Nome do <i>Encoder</i>	EfficientNet-B4
Pesos do <i>Encoder</i>	ImageNet
Arquitetura do Modelo	DeepLabV3Plus
Número de Classes	20
Batch Size (Train)	8
Batch Size (Validation)	4
Batch Size (Test)	2
Número de Épocas	80
Tamanho dos Patches	512x512

O conjunto de dados foi dividido em imagens de treinamento, validação e teste, de forma dinâmica. Para aumentar a diversidade do conjunto de dados e melhorar a robustez do modelo, aplicou-se *data augmentation* utilizando técnicas como *Random Crop*, *Horizontal Flip*, rotação, ajuste de brilho/contraste, ruído gaussiano, entre outras. Essas técnicas garantem que o modelo seja treinado com variações de iluminação, ângulos e deformações, simulando diferentes condições.

Durante o treinamento do modelo de segmentação semântica, foi utilizado o agendador *CosineAnnealingLR*, proposto por Loshchilov & Hutter (2017), que ajusta a taxa de aprendizado ao longo do tempo seguindo uma curva senoidal, permitindo uma convergência mais suave. A função de perda empregada foi a *Cross Entropy Loss*, amplamente utilizada em tarefas de classificação e segmentação (Goodfellow et al., 2016), a qual penaliza previsões incorretas. Para otimizar os pesos da rede, foi escolhido o otimizador *AdamW*, que introduz a regularização por decaimento de peso, conforme descrito por Loshchilov & Hutter (2019), melhorando a generalização do modelo.

Dois nós principais foram implementados no ROS2: um para capturar imagens em tempo real das câmeras do veículo e outro para aplicar o modelo de segmentação, classificando objetos como pedestres e veículos. A comunicação eficiente entre os nós permitiu que o sistema fornecesse uma visão detalhada do ambiente.

Resultados e discussão

As figuras a seguir apresentam o resultado da aplicação de um modelo de segmentação semântica em uma cena urbana do dataset *Cityscapes*, um dos mais utilizados em tarefas de segmentação de cenas de tráfego urbano (Cordts et al., 2016). Na Figura 2, observa-se uma imagem original de uma interseção de ruas com pedestres, carros estacionados e outros. Este cenário é típico de ambientes urbanos e oferece desafios para sistemas de visão computacional, como variações de iluminação e sobreposição de objetos.

Figura 2 – Uma rua urbana com carros, pedestres, árvores e sinais de trânsito. Fonte: *Cityscapes*.



Este tipo de segmentação semântica é essencial para a criação de veículos autônomos e sistemas de assistência ao motorista, uma vez que possibilita ao sistema um entendimento aprofundado do ambiente em que está inserido, como demonstrado na Figura 3. Portanto, o desempenho desse modelo na identificação e classificação de objetos contribui diretamente para a segurança e eficiência de sistemas autônomos em ambientes urbanos diversificados (Janai et al., 2020).

Figura 3 – Resultado da segmentação semântica aplicada à Figura 1. Cada objeto foi classificado em categorias específicas: carros estão em azul, pedestres em vermelho, árvores em verde e a rua em roxo. Fonte: Autor.



A Figura 4 apresenta os resultados de uma segmentação semântica aplicada a uma cena urbana capturada a partir de uma câmera veicular. A imagem à esquerda mostra o frame original de uma interseção, com diversos elementos urbanos como pedestres, carros, prédios e sinais de trânsito. A imagem à direita exibe a mesma cena, porém com as classes segmentadas e destacadas em cores distintas. A figura também apresenta um cabeçalho descritivo na parte superior, que lista todas as classes possíveis que o modelo pode identificar, cada uma associada a uma cor.

Figura 4 – Rua de Berlim, na Alemanha, antes (à esquerda) e depois da segmentação (à direita), com objetos coloridos conforme suas categorias, como carros e pedestres. Fonte: Autor.

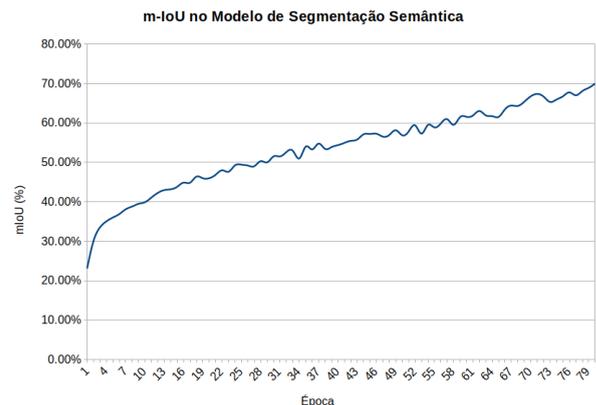


Esta estrutura visual simplifica a compreensão do âmbito da segmentação e evidencia a habilidade do modelo em identificar uma ampla gama de objetos existentes no contexto urbano.

A Figura 5 ilustra a evolução da métrica de m-IoU (*mean-IoU*) no modelo de segmentação semântica ao longo de diversas fases de treinamento,

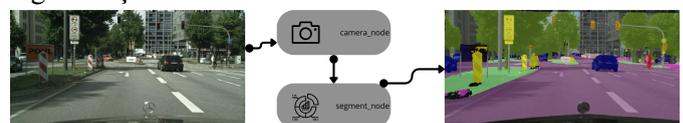
evidenciando um crescimento contínuo no rendimento. Eventualmente alcançando cerca de 70% nas fases finais. A falta de qualquer queda acentuada sugere que o modelo aprendeu de maneira eficiente e faz uma boa generalização durante o processo de formação.

Figura 5 – Métrica de validação do modelo em relação às épocas de treinamento. Fonte: Autor.



A Figura 6 ilustra o fluxo de dados entre os nós do sistema ROS2, destacando o funcionamento do nó de captura de imagens e do nó de segmentação semântica. O *camera_node* recebe as imagens diretamente das câmeras instaladas no veículo, capturando em tempo real o ambiente ao redor. Essas imagens são então publicadas em um tópico específico para serem processadas pelo *segment_node*, que aplica o modelo de segmentação semântica e classifica os elementos da cena. O resultado é exibido na parte inferior da figura, onde cada objeto foi segmentado e categorizado.

Figura 6 – Fluxo de dados entre nós ROS2 para segmentação semântica. Fonte: Autor.

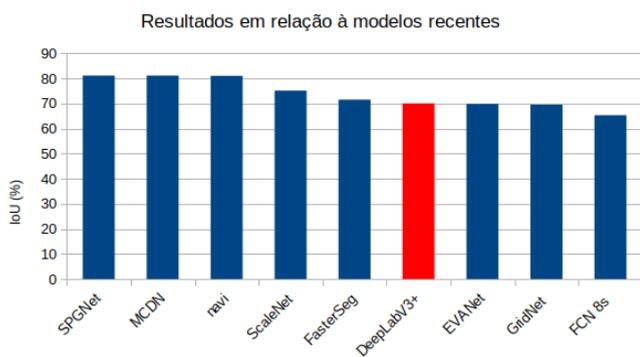


Conclusões

O modelo desenvolvido, baseado na arquitetura *DeepLabV3+* com o *encoder EfficientNet-B4*, alcançou um IoU de aproximadamente 70.01%, evidenciando um desempenho considerável quando comparado a outros modelos de segmentação no nível de pixel nas publicações feitas em relação ao *Cityscapes*, como evidenciado pela Figura 7. O sistema mostrou-se promissor o suficiente para proporcionar uma compreensão detalhada do ambiente em tempo real.

Além disso é importante destacar que o uso de outros *encoders* poderiam gerar resultados diferentes, assim como outras técnicas de regularização e *transfer learning*, outras técnicas relacionadas à otimização computacional do treinamento geram resultados diversificados no atual projeto.

Figura 7 – Comparação com outros modelos de Segmentação Semântica (modelo em destaque relativo a este projeto). Fonte: *Cityscapes*.



O projeto, portanto, estabeleceu uma base sólida para futuras otimizações, como melhorias no processamento em tempo real em plataformas embarcadas, apontando para um caminho na evolução de veículos autônomos mais seguros e eficientes.

Agradecimentos

Gostaria de expressar meus sinceros agradecimentos à FAPEMIG e à UNIFEI pelo financiamento e suporte essenciais para a realização desta pesquisa. Ao meu orientador, Giovani Bernardes Vitor, minha profunda gratidão por sua orientação inestimável, conhecimento compartilhado e dedicação ao longo deste trabalho. Agradeço também aos meus amigos pelo constante apoio e encorajamento, e à minha família pelo amor incondicional e compreensão durante toda esta jornada.

Referências

ABADI, M.; et al. TensorFlow: a system for large-scale machine learning. In: **USENIX SYMPOSIUM ON OPERATING SYSTEMS DESIGN AND IMPLEMENTATION**, 12., 2016, Savannah, GA. Proceedings... Berkeley, CA: USENIX Association, 2016. p. 265-283.

BRADSKI, G. The OpenCV library. *Dr. Dobb's Journal of Software Tools*, 2000. Disponível em: <https://www.drdobbs.com/open-source/the-opencv-library/184404319>. Acesso em: 05/10/2024.

CHEN, Liang-Chieh; PAPANDREOU, George; et al. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. 2017. Disponível em: <https://arxiv.org/abs/1606.00915>. Acesso em: 05/10/2024.

CHEN, L. C.; et al. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: **EUROPEAN CONFERENCE ON COMPUTER VISION**, 15., 2018, Munich, Germany. Proceedings... Cham: Springer, 2018.

CORDTS, M.; et al. The Cityscapes dataset for semantic urban scene understanding. In: **IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION**, 2016, Las Vegas, NV. Proceedings... New York: IEEE, 2016.

GARCIA-GARCIA, A.; et al. A review on deep learning techniques applied to semantic segmentation. *arXiv preprint*, arXiv:1704.06857, 2018. Disponível em: <https://arxiv.org/abs/1704.06857>. Acesso em: 05/10/2024.

GOODFELLOW, Ian; BENGIO, Yoshua; COURVILLE, Aaron. Deep Learning. Cambridge: MIT Press, 2016. Disponível em: <https://www.deeplearningbook.org/>. Acesso em: 05/10/2024. Seção 6.2.2 – Output Units.

LOSHCHILOV, I.; HUTTER, F. Decoupled weight decay regularization. In: **INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS**, 2019, New Orleans, LA. Proceedings... [S.l.: s.n.], 2019.

LOSHCHILOV, I.; HUTTER, F. SGDR: stochastic gradient descent with warm restarts. In: **INTERNATIONAL CONFERENCE ON LEARNING REPRESENTATIONS**, 2017, Toulon, France. Proceedings... [S.l.: s.n.], 2017.

PASZKE, A.; et al. PyTorch: an imperative style, high-performance deep learning library. In: **ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS**, 33., 2019, Vancouver, Canada. Proceedings... Red Hook, NY: Curran Associates, Inc., 2019.

ROS2. ROS 2 design. Disponível em: <https://design.ros2.org/>. Acesso em: 05/10/2024.

TAN, M.; LE, Q. EfficientNet: rethinking model scaling for convolutional neural networks. In: **INTERNATIONAL CONFERENCE ON MACHINE LEARNING**, 36., 2019, Long Beach, CA. Proceedings... New York: PMLR, 2019.