

# Métodos de *Deep Learning* aplicados à Segmentação Semântica de Imagens para Percepção de Veículos Autônomos

Gabriel Toffanetto França da Rocha

Laboratório de Mobilidade Autônoma – LMA

Faculdade de Engenharia Mecânica, Universidade Estadual de Campinas

Campinas, Brasil

g289320@dac.unicamp.br

**Abstract—**

**Index Terms—**Deep learning, Visão computacional, Segmentação Semântica de Imagem, Robótica móvel, Veículos autônomos

## I. INTRODUÇÃO

Veículos com capacidade de se guiarem de forma autônoma estão cada vez mais presentes no dia a dia da sociedade contemporânea, possibilitando que o motorista possa realizar outras atividades durante a navegação, ou que o mesmo seja assistido em caso de alguma falha humana do condutor. Para que o automóvel seja capaz de se mover por conta própria, o mesmo deve ser capaz de perceber o ambiente, e sensores como sonares, radares, LiDARs e câmeras podem ser utilizados para tal. Porém, a câmera se faz como uma solução mais viável economicamente, e como visto na literatura, apresenta soluções que contemplam os desafios da navegação autônoma de veículos em ambientes urbanos, como visto nos trabalhos de [8] e [17].

Para que um veículo autônomo possa entender o ambiente à sua volta, é necessário que ele saiba reconhecer as entidades que o compõem, como por exemplo: estrada, veículos, calçadas, pedestres e vegetação, para que assim, o mesmo saiba diferenciar área navegável de obstáculos [12]. Para isso, o emprego da técnica de segmentação semântica de imagens, onde cada pixel da imagem é classificado de acordo com a entidade do ambiente da qual ele faz parte [10]. A Fig. 1 mostra a aplicação da técnica de segmentação semântica fundida à informação de profundidade dada por uma câmera *stereo*, permitindo a obtenção de um *grid* de percepção dinâmica local (DLP), que projeta no plano 2D o ambiente contendo a detecção de múltiplos objetos para que o veículo consiga planejar seu caminho [19].

Existem métodos de processamento de imagens que realizam o mascaramento de cada entidade da imagem, porém a definição de qual é a classe de cada segmento se faz desafiadora, sendo anteriormente empregada a utilização de redes neurais artificiais (ANNs) para tal, como feito por [18]. Porém, devido à utilização de ANNs somente para a classificação final, era necessário muito pré-processamento para realização da segmentação semântica. Com o desenvolvimento das redes neurais profundas (DNNs), obteve-se métodos com poder

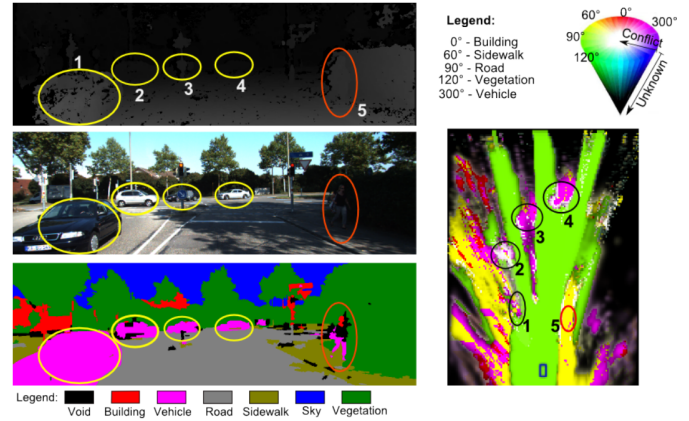


Fig. 1. DLP com ênfase na detecção múltipla de objetos móveis obtido com a fusão da imagem semanticamente segmentada e as informações de profundidade [19].

suficiente para que, dada uma imagem bruta de entrada e uma imagem de referência (*ground truth*) segmentada para comparação, a rede profunda consegue aprender como realizar a segmentação da imagem do ambiente urbano, como nas várias arquiteturas mostradas por [14]. Com a popularização desses métodos, já existem diversos conjuntos de dados para treinamento das DNNs, como os [5] e [2], [3]. Existem também *datasets* que trazem cenas ainda mais desafiadoras, como o [13] que apresenta imagens urbanas durante noites chuvosas.

Dessa forma, a percepção do ambiente por meio de visão computacional se faz indispensável para o desenvolvimento dos veículos autônomos, e com isso, os métodos de *deep learning* se fazem uma grande ferramenta para conseguir-se reconhecer as entidades de uma cena urbana com robustez às variações de luz e reflexos, sendo assim uma solução a ser explorada. Além do desempenho da segmentação semântica, o tempo demandado para tal operação também é vital, uma vez que durante a navegação, todos os módulos operam em tempo real, e a quantidade de *frames* segmentados por segundo é uma informação importante. Dessa forma, esse trabalho propõem a utilização de redes neurais profundas com diferentes arquiteturas para a segmentação semântica de imagens de cenas

urbanas, utilizados *datasets* da literatura e para testes finais, utilizando imagens reais adquiridas pelo autor. Com esse teste, espera-se poder julgar se o treinamento da rede neural com conjuntos de dados da literatura conseguem gerar máquinas que generalizam bem para localidades não vistas durante o treinamento.

Este trabalho é dividido em seis partes, onde na Seção I é realizada a motivação e contextualização da pesquisa e na Seção II é apresentado o estado da arte, discorrendo sobre as soluções utilizadas atualmente. Com isso, a Seção III apresenta a metodologia a ser utilizada neste trabalho, seguida dos resultados obtidos na Seção IV e sua análise na Seção V. Por fim, são apresentadas as conclusões na Seção VI e as referências utilizadas.

## II. ESTADO DA ARTE

### III. METODOLOGIA

Para a solução do problema foi escolhida a rede STDC1-50, que apresentou maior velocidade nos testes realizados pelos autores do *survey* [14], com um desempenho de segmentação considerável. **justificar a escolha da rede**

#### A. Arquitetura

A rede STDC1 [7] é baseada na arquitetura de *two-branch network*, onde existe uma bifurcação da rede em dois ramos, um ligado à extração de informações de contexto e o outro aplicado para obtenção de informações espaciais da entrada. A informação de contexto demanda de uma maior profundidade, implementando mecanismos de atenção em dois níveis para extração dos atributos necessários para a identificação das regiões da imagem, como mostrado na Fig. 2(a).

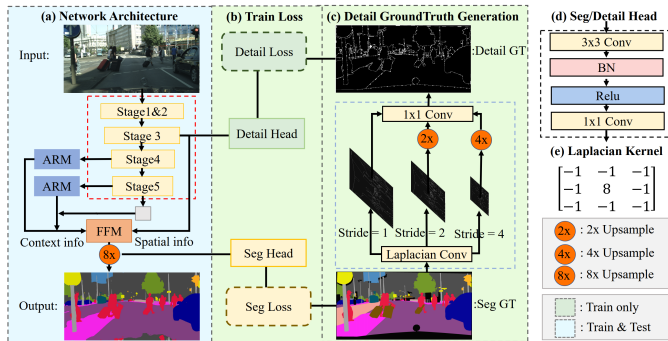


Fig. 2. Arquitetura completa da rede neural utilizada [7].

Porém, o fato de se processar a informação de entrada em dois ramos, traz um inconveniente para o problema de segmentação em tempo real, que é justamente o tempo de inferência da rede. Desta forma, a rede aplica uma estratégia elegante ao utilizar no lugar de um ramo com novas camadas, uma *skip connection* do Stage 3, para o bloco de fusão dos *feature maps* oriundos dos dois braços da rede. Para forçar a captura das informações espaciais, a saída da *skip connection* é utilizada em uma tarefa auxiliar de detecção de bordas de cada região semântica da imagem, por meio de uma *detail head*, como visto na Fig. 2(b). Tal procedimento é realizado por meio

de treinamento supervisionado, onde com a aplicação de filtros laplacianos na imagem rótulo de segmentação semântica, se obtém a imagem rótulo com as bordas de cada segmento da imagem de entrada, procedimento ilustrado na Fig. 2(c). Tal informação explicita os detalhes à serem capturados pelo ramo espacial, e a perda é computada por meio da saída da *detail head* e o rótulo obtido. Reforça-se que esse procedimento é realizado apenas em treinamento, enquanto durante a inferência a rede neural irá utilizar as habilidades aprendidas por meio do gradiente sobre a perda atrelada à segmentação e à detecção de bordas, sem realizar explicitamente a extração das bordas de cada região da imagem.

O módulo principal da rede é o *Short-Term Dense Concatenate Module* (STDC), que possui quatro camadas internas de convolução (convolução + *batch normalization* + ReLU), e os *feature maps* de todas as camadas convolucionais do bloco são concatenadas na saída, formando assim um bloco com característica densa, como descrito na Fig. 3. Dada uma entrada de  $M$  canais, a primeira camada possui *kernels*  $1 \times 1$ , gerando  $N/2$  *feature maps*. A segunda camada realiza *downsampling*, apresentando *stride* = 2 e *kernel*  $3 \times 3$ , assim como as consecutivas, porém contribuindo com  $N/4$  canais de saída. Por fim, as duas ultimas camadas contribuem cada uma com  $N/8$  mapas de ativação, e com isso, o bloco concatena todos os canais de saída, apresentando uma saída com  $N$  canais. Tal configuração se fez para valorizar a tarefa de segmentação de imagem, onde nas primeiras camadas existem mais filtros, e com isso, conseguindo explorar melhor a extração de informação multi-escala [7].

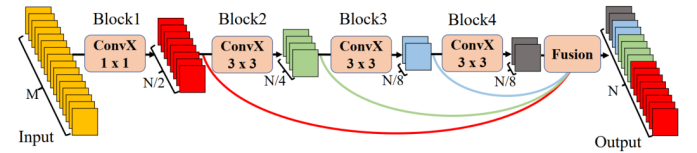


Fig. 3. Estrutura do módulo STDC. Adaptado de [7].

O estágio 1 e 2 realizam a operação de convolução, *batch normalization* e ReLU aos dados de entrada, com *stride* igual a 2. Já os estágios 3, 4 e 5 apresentam realizações em série de módulo STDC. O mecanismo de atenção e o bloco de FFM são baseados na rede BiSeNet [22], **explicar brevemente funcionamento desses blocos**

Por fim é realizado um *upsampling* de 8 vezes para a recuperação da dimensão de entrada.

Sua implementação se deu com base nos códigos fonte disponibilizados pelos autores da rede, modificados para o problema em questão, e disponibilizados no GitHub<sup>1</sup>. Utilizou-se o *framework* PyTorch, juntamente dos pacotes de CUDA necessários para utilização da GPU NVIDIA em treinamento e inferência.

#### B. Dataset de treinamento e dados de teste

O conjunto de dados Cityscapes [5] foi escolhido como base para treinamento e avaliação do modelo, sendo um

<sup>1</sup>[github.com/toffanetto/STDC-Seg](https://github.com/toffanetto/STDC-Seg)

*dataset* que contempla imagens de alta resolução, de contextos urbanos, capturados em diversas cidades da Europa. A base de dados conta com a anotação de 30 classes, sendo que 19 estão disponíveis para o problema de segmentação semântica, sendo elas: estrada, calçada, pedestre, piloto/conductor, carro, caminhão, ônibus, trem, moto, bicicleta, edifício, muro, cerca, poste, placa de transito, semáforo, vegetação, terreno e céu, cuja legenda está disponível na Fig. 4. Estão disponíveis no mesmo, 5000 imagens com anotação fina dos segmentos semânticos, divididos em treinamento (2975), validação (500) e teste (1525). A resolução das imagens é de  $2048 \times 1024$  pixels, sendo uma entrada desafiadora para a inferência em tempo real.

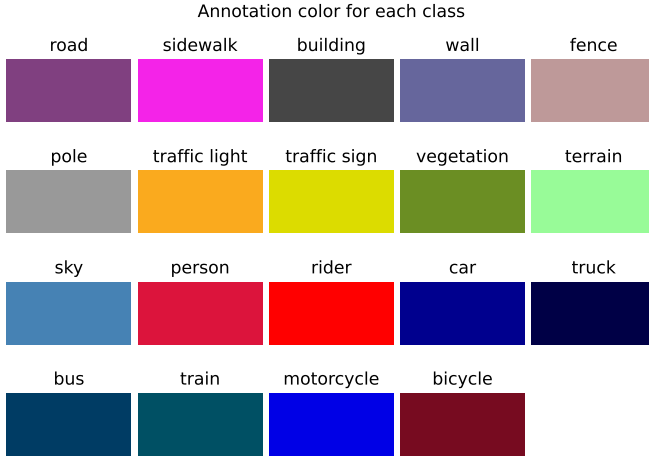


Fig. 4. Legenda de cores referentes à cada classe do *dataset*.

Foram implementadas estratégias de *data augmentation* pelos autores da rede neural [7], utilizando alteração de cor, brilho e contraste, inversão horizontal, aplicação de escala e corte, buscando a inserção de robustez e maximização da capacidade de generalização, evitando *overfitting*. A aplicação de escala também é utilizada para reduzir a dimensão dos dados de entrada, facilitando a predição do mapa de segmentação semântico em tempo real pela rede, onde foi escolhida a escala de 0,75 por fornecer um desempenho maior.

Para teste da capacidade de generalização do modelo, realizou-se uma coleta de dados na mesma forma da realizada pelos autores do *dataset*, no *campus* da Unicamp, sendo assim, imagens de locais que não foram vistos em treinamento, e aplicou-se a inferência da máscara de segmentação semântica das mesmas por meio da rede STCD1.

### C. Método de treinamento

O treinamento da rede neural foi realizado por meio do algoritmo de gradiente descendente estocástico (SGD), com *momentum* de 0,9, decaimento de pesos de  $5e^{-4}$ , com atualização dos pesos por *mini-batch* de 4 amostras, devido às limitações de memória da GPU utilizada. O treinamento é limitado por número de iterações, onde com base na rede original, foi limitado em 60000 iterações, onde é adotada uma estratégia de *warmup*. Com isso, a taxa de aprendizado é incrementada

de um valor pequeno até o valor alvo durante as primeiras 1000 iterações para estabilização do processo de otimização e evitar a divergência do mesmo. Além disso, durante o treinamento a taxa de aprendizado é adaptada por meio de uma política polinomial, dada por  $(1 - iter/max\_iter)^{power}$ , onde foi considerada uma potência de 0.9 [7].

Como função de custo para ajuste dos pesos, é considerada a entropia cruzada, por se tratar de um problema de classificação, porém utilizando *Online Hard Example Mining* para a otimização da tarefa de aprendizado da segmentação semântica. **Explicar rapidamente o OHEM** [16].

Também é computada a função de custo para o problema de detecção dos detalhes, realizado pela *detail head*. Nesse caso, não é utilizado a entropia cruzada, devido a ser um problema altamente desbalanceado, onde a mesma não apresenta bons resultado. Nesse caso, são aplicadas as métricas de entropia cruzada binária e *dice loss*, conforme (1). A *dice loss*, enunciada em (2) se baseia em medir a sobreposição entre o mapa predito e seu respectivo rótulo, variando no intervalo [0, 1], sendo assim robusta ao número de pixels de cada classe. O equacionamento é realizado considerando uma entrada de  $H \times W$  pixels, obtendo assim uma predição do mapa de detalhes  $p_d$ , sendo  $g_d$  seu respectivo rótulo, e  $\epsilon$  uma constante para evitar a divisão por zero [6].

$$L_{detail}(p_d, g_d) = L_{dice}(p_d, g_d) + L_{bce}(p_d, g_d) \quad (1)$$

$$L_{dice}(p_d, g_d) = 1 - \frac{2 \sum_i^{H \times W} p_d^i g_d^i + \epsilon}{\sum_i^{H \times W} (p_d^i)^2 + \sum_i^{H \times W} (g_d^i)^2 + \epsilon} \quad (2)$$

### D. Métricas de avaliação

Sendo um problema de segmentação com  $k + 1$  classes considerando o fundo da imagem, têm-se que  $p_{ij}$  é o número de pixels pertencentes à classe  $i$  que foram preditos para a classe  $j$ , logo,  $i = j$  representa uma classificação correta da classe do pixel.

Com isso, as duas métricas mais famosas para segmentação semântica, sendo elas a *Intersection over Union* (IoU) e *mean Intersection over Union* (mIoU). A IoU, enunciada em (3), é obtida pela quantidade de pixels preditos corretamente para uma certa classe (interseção predição e *ground truth*), dividido pela quantidade de pixels preditos incorretamente somado ao *ground truth* (união) [14].

$$IoU = \frac{\sum_{i=0}^k p_{ii}}{\sum_{i=0}^k \sum_{j=0}^k (p_{ij} + p_{ji}) - \sum_{i=0}^k p_{ii}} \quad (3)$$

Ao realizar a média da IoU para  $k + 1$  classes, se obtém a mIoU, conforme (4). Devido à informação de um desempenho médio para todas as classes, essa métrica foi escolhida para realização da medida de efetividade da segmentação semântica.

$$mIoU = \frac{1}{k + 1} \sum_{i=0}^k \frac{p_{ii}}{\sum_{j=0}^k (p_{ij} + p_{ji}) - p_{ii}} \quad (4)$$

## IV. RESULTADOS

### A. Métricas de teste

### B. Cityscapes

### C. Unicamp

## V. ANÁLISE DOS RESULTADOS

## VI. CONCLUSÕES

## AGRADECIMENTOS

## REFERÊNCIAS

- [1] BADRINARAYANAN, V., KENDALL, A., AND CIPOLLA, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, Oct. 2016.
- [2] BROSTOW, G. J., FAUQUEUR, J., AND CIPOLLA, R. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognition Letters* 30, 2 (Jan. 2009), 88–97.
- [3] BROSTOW, G. J., SHOTTON, J., FAUQUEUR, J., AND CIPOLLA, R. Segmentation and Recognition Using Structure from Motion Point Clouds. In *Computer Vision – ECCV 2008*, D. Forsyth, P. Torr, and A. Zisserman, Eds., vol. 5302. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008, pp. 44–57.
- [4] CHAO, P., KAO, C.-Y., RUAN, Y., HUANG, C.-H., AND LIN, Y.-L. HardNet: A Low Memory Traffic Network. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)* (Seoul, Korea (South), Oct. 2019), IEEE, pp. 3551–3560.
- [5] CORDTS, M., OMRAN, M., RAMOS, S., REHFELD, T., ENZWEILER, M., BENENSON, R., FRANKE, U., ROTH, S., AND SCHIELE, B. The Cityscapes Dataset for Semantic Urban Scene Understanding, Apr. 2016.
- [6] DENG, R., SHEN, C., LIU, S., WANG, H., AND LIU, X. Learning to predict crisp boundaries, July 2018.
- [7] FAN, M., LAI, S., HUANG, J., WEI, X., CHAI, Z., LUO, J., AND WEI, X. Rethinking BiSeNet For Real-time Semantic Segmentation, Apr. 2021.
- [8] GARCIA, O., VITOR, G. B., FERREIRA, J. V., MEIRELLES, P. S., AND DE MIRANDA NETO, A. The VILMA intelligent vehicle: An architectural design for cooperative control between driver and automated system. *Journal of Modern Transportation* 26, 3 (Sept. 2018), 220–229.
- [9] GÉRON, A. *Hands-on Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems*, second ed. O'Reilly, 2020.
- [10] HE, H.-J., ZHENG, C., AND SUN, D.-W. Image Segmentation Techniques. In *Computer Vision Technology for Food Quality Evaluation*. Elsevier, 2016, pp. 45–63.
- [11] JANAI, J., GÜNEY, F., BEHL, A., AND GEIGER, A. Computer Vision for Autonomous Vehicles: Problems, Datasets and State of the Art. *Foundations and Trends® in Computer Graphics and Vision* 12, 1–3 (2020), 1–308.
- [12] JEBAMIKYOUS, H.-H., AND KASHEF, R. Autonomous vehicles perception (AVP) using deep learning: Modeling, assessment, and challenges. *IEEE Access* 10 (2022), 10523–10535.
- [13] JIN, J., FATEMI, A., LIRA, W., YU, F., LENG, B., MA, R., MAHDAVI-AMIRI, A., AND ZHANG, H. RaidaR: A Rich Annotated Image Dataset of Rainy Street Scenes, Oct. 2021.
- [14] PAPADEAS, I., TSOCHATZIDIS, L., AMANATIADIS, A., AND PRATIKAKIS, I. Real-Time Semantic Image Segmentation with Deep Learning for Autonomous Driving: A Survey. *Applied Sciences* 11, 19 (Sept. 2021), 8802.
- [15] POUDEL, R. P. K., BONDE, U., LIWICKI, S., AND ZACH, C. ContextNet: Exploring Context and Detail for Semantic Segmentation in Real-time, Nov. 2018.
- [16] SHRIVASTAVA, A., GUPTA, A., AND GIRSHICK, R. Training Region-based Object Detectors with Online Hard Example Mining, Apr. 2016.
- [17] VITOR, G. B. *Urban environment and navigation using robotic vision: conception and implementation applied to autonomous vehicle = Percepção do ambiente urbano e navegação usando visão robótica: concepção e implementação aplicado à veículo autônomo*. Doutor em Engenharia Mecânica, Universidade Estadual de Campinas, Campinas, SP, Sept. 2014.
- [18] VITOR, G. B., LIMA, D. A., VICTORINO, A. C., AND JANITO V. FERREIRA. A 2D/3D Vision Based Approach Applied to Road Detection in Urban Environments. In *2013 IEEE Intelligent Vehicles Symposium (IV)* (Gold Coast City, Australia, June 2013), IEEE, pp. 952–957.
- [19] VITOR, G. B., VICTORINO, A. C., AND FERREIRA, J. V. Modeling evidential grids using semantic context information for dynamic scene perception. *Knowledge-Based Systems* 215 (Mar. 2021), 106777.
- [20] WANG, Y., ZHOU, Q., AND WU, X. ESNet: An Efficient Symmetric Network for Real-time Semantic Segmentation, June 2019.
- [21] YU, C., GAO, C., WANG, J., YU, G., SHEN, C., AND SANG, N. BiSeNet V2: Bilateral Network with Guided Aggregation for Real-time Semantic Segmentation, Apr. 2020.
- [22] YU, C., WANG, J., PENG, C., GAO, C., YU, G., AND SANG, N. BiSeNet: Bilateral Segmentation Network for Real-time Semantic Segmentation, Aug. 2018.