engineering proceedings

MDPI

Proceeding Paper

Enhancing Autonomous Vehicle Segmentation with Compact UNET Models and Temporal Input Fusion †

Matúš Čavojský [1,*], Štefan Mikula [2], Juraj Gazda [2] and Gabriel Bugár [1]

[1] Department of Electronics and Multimedia Telecommunications, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 040 01 Košice, Slovakia; gabriel.bugar@tuke.sk
[2] Department of Computers and Informatics, Faculty of Electrical Engineering and Informatics, Technical University of Košice, Letná 9, 040 01 Košice, Slovakia; stevo.mikula@gmail.com (Š.M.); juraj.gazda@tuke.sk (J.G.)
* Correspondence: matus.cavojsky@tuke.sk
† Presented at the Sustainable Mobility and Transportation Symposium 2025, Győr, Hungary, 16–18 October 2025.

Abstract

This work addresses the challenge of video semantic segmentation, a critical component in applications such as autonomous driving. The primary aim was to explore the role of temporal awareness in video sequences and its impact on split computing. To achieve this, we analyzed existing deep neural networks for semantic segmentation and their computational demands and proposed a split computing architecture that leverages high-accuracy segmentation results from a remote server to enhance performance on mobile devices. To validate the proposed approach, we developed and tested four U-Net modifications on the CamVid dataset. Our results demonstrate that incorporating segmentation masks from previous frames significantly improves accuracy in split computing scenarios. In particular, masks warped using optical flow yielded the best results, increasing segmentation accuracy from 81.1% to 84.1% with minimal additional computational cost. These findings highlight the potential of time-aware split computing to enhance video semantic segmentation performance in resource-constrained IoT environments.

Keywords: convolutional neural networks; deep learning; optical flow; split computing; semantic segmentation; video semantic segmentation

Academic Editors: Szabolcs Kocsis-Szürke, Dániel Csikor and Boglárka Eisinger Balassa

Published: 14 November 2025

Citation: Čavojský, M.; Mikula, Š.; Gazda, J.; Bugár, G. Enhancing Autonomous Vehicle Segmentation with Compact UNET Models and Temporal Input Fusion. *Eng. Proc.* **2025**, *113*, 71. https://doi.org/10.3390/engproc2025113071

1. Introduction

In autonomous driving systems, perception is the first stage, bridging the sensory input data acquired by the equipped sensors to enable autonomous vehicles to make informed decisions about their surroundings. The perception data is then fed into a decision-making system, which makes the best judgment possible given a certain scenario to avoid potential collisions. Recent years have seen major advances in computer vision, especially in image segmentation, the foundation of image processing and computer vision. The goal of image segmentation is to create a dense pixel-by-pixel segmentation map of a picture, with each pixel allocated to a different class or object. Identifying and separating various objects from the background and temporal events in a video, as well as providing a more detailed and organized representation of the visual content, are all accomplished through the process of video image segmentation, which divides video frames (images) into multiple segments or objects based on specific characteristics. Traditionally, motion vectors [1,2], feature point trajectories [3], and color descriptors [4,5] are used to segment video objects based on their

motion and appearance. Object areas are typically derived through complex and fragile inference techniques, frequently with predetermined assumptions about object and camera motion, depending on the availability and quality of these inputs [6]. Since autonomous vehicles must have a thorough understanding of their environment to make informed decisions, precise object identification and segmentation within video frames is essential for improving machine perception and environmental understanding.

2. Materials and Methods

2.1. Related Work

Although extensive research studies have been done in the area of video semantic segmentation for autonomous mobility, they are limited to achieving single objective, mostly improved accuracy or improved efficiency. Garcia et al. [7] and Lateef and Ruichek [8] present extensive surveys on image segmentation techniques, discussing their characteristics, advancements, and a significant shift noted by Garcia et al. in the field of semantic image segmentation. Hao et al. [9] discuss five distinct tasks within computer vision, characterized by the nature and volume of output information they provide and specifically focus on the rate of learning with a teacher. These tasks include image classification, object detection, semantic segmentation, instance segmentation, and panoptic segmentation. Image classification serves as the foundational task, involving the categorization of an image into one or multiple classes that represent real-world or abstract objects. Moreover, the work of Sultana et al. [10] offers an overview of semantic segmentation methods, considering the context of instance segmentation. This review examines the techniques employed in these tasks and provides insights into their performance and applicability.

Numerous CNNs designed for image segmentation rely on an encoder–decoder architecture due to its robust and efficient approach to semantic pixel-wise segmentation. Badrinarayanan et al. [11] pioneered the concept of an encoder–decoder network architecture for semantic segmentation, utilizing an encoder based on the pre-trained VGG16 architecture. The encoder network extracts features from the input image, while the decoder network maps low-resolution encoder feature maps to full input resolution feature maps for pixel-wise classification. This method resembles fully convolutional networks, which was initially proposed by Long et al. [12], in which the deconvolutional layer effectively serves as its decoder, albeit employing a single layer. Many segmentation architectures [13–17] share the same encoder network and they only vary in the form of their decoder network. Despite achieving state-of-the-art accuracy and fast inference speeds, many encoder–decoder architectures have slow inference due to their large parameter count and complex architectural design [18]. Kasar et al. [19] in their U-Net and SegNet comparison have shown that U-Net has better segmentation results than SegNet in both accuracy and dice loss coefficient. A common strategy involves applying a segmentation method to individual frames independently and subsequently incorporating awareness of previous frames into the model. However, in the context of Video Semantic Segmentation (VSS), they observed that many existing systems naively apply the same segmentation algorithm independently to each frame of the video. This approach overlooks the temporal dependencies and correlations among consecutive frames, resulting in sub-optimal results. Recognizing the importance of temporal information in video segmentation, more recent studies have made progress in this area, as highlighted by Zhou et al. [20] in their survey on video segmentation using deep neural networks.

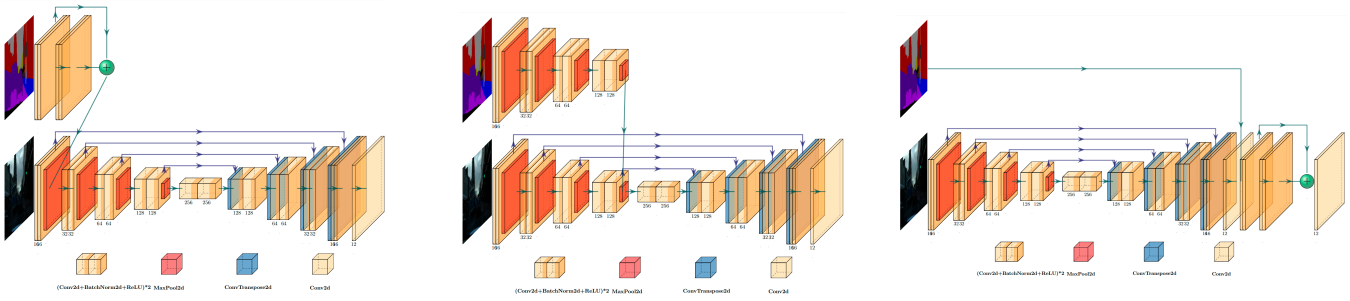
2.2. Approach

In this study, we chose the U-Net architecture due to its proven segmentation accuracy, even with complex objects and small datasets, as well as its efficient training, testing,

and compatibility with data augmentation techniques. Although not originally designed for semantic video segmentation, temporal awareness can be introduced by fusing prior segmentation outputs with current inputs. To explore this, we propose three modifications to a pruned U-Net, each incorporating previous segmentation outputs in different ways to evaluate their impact on video segmentation accuracy.

Baseline U-Net and Model Pruning–In U-Net, convolutional and deconvolutional layers form a symmetric encoder–decoder. The encoder relies on convolution blocks (Conv → BatchNorm → ReLU) with max-pooling every two blocks to reduce spatial dimensions and double channel count. This continues until reaching the smallest resolution. The decoder mirrors this process, using transposed convolutions to upsample while halving the channels and concatenating corresponding encoder outputs for richer context. Each decoder block has convolution blocks similar to the encoder for feature extraction. The final layer is a single-channel convolution that outputs a predicted segmentation mask matching the input image size. Three-by-three filters are used for all convolutions, with encoder channels progressing from 64 to 1024. This configuration is referred to as UNET1. We also define a lighter version, UNET2, with reduced channels (16 to 256), serving as the base for further modifications to improve efficiency.

U-Net with Early Fusion–UNET3 introduces early fusion, merging the input image and a previous segmentation mask (or its optical flow variant) before feature extraction. The mask is preprocessed using a residual block (two Conv-BatchNorm-ReLU layers) to extract features. This early integration enables the model to learn relationships between raw inputs at the earliest stage (Figure 1a).



(**a**) (**b**) (**c**)

Figure 1. Diagrams of the neural networks: (**a**) UNET3; (**b**) UNET4; (**c**) UNET5.

U-Net with Mid Fusion–UNET4 employs feature-level fusion. The prior mask is processed through a network replicating the first four layers of UNET2's encoder. Its output is then linearly combined with the RGB encoder's fourth-layer output at the bottleneck, enabling more refined integration of temporal features. The architecture of the UNET4 modification is visually depicted in Figure 1b.

U-Net with Late Fusion–UNET5 performs late fusion by appending the prior mask output at the end of the UNET2 pipeline. The combined output passes through a residual block and a final 1×1 convolution layer. This approach tests whether adding temporal information post-segmentation can still enhance accuracy (Figure 1c).

For model evaluation, we opted for the widely used mean Intersection over Union (mIoU) metric, as supported by prior works [21–23] and surveys [7,20]. To handle class imbalance and avoid division-by-zero issues, we used image-wise micro mIoU. Additionally, we monitored per-class performance using macro mIoU, allowing us to assess class-specific accuracy without averaging across classes. With the selected metric in mind, we adopted the Jaccard loss function due to its resilience to imbalanced classes, interpretability, similarity to mIoU (inversely related), and differentiability.

3. Results

The unmodified U-Net (UNET1) achieved 81.3% accuracy with a 10.5% standard deviation using only RGB input. As the original architecture lacks inherent support for semantic video segmentation, temporal information was incorporated by appending the previous segmentation output to the input, alongside the image intended for segmentation—either directly (MASK1, WARPED1) or as one-hot encoded (MASK2, WARPED2). All modifications improved accuracy and reduced variance, with WARPED2 yielding the best results—84.6% accuracy and a 7.5% standard deviation—marking a 3.3% accuracy gain and a 3.0% variance reduction. Notably, UNET1 with WARPED2 input achieved the highest per-class accuracy in 8 out of 12 categories, including buildings, vehicles, and pedestrians.

The pruned U-Net (UNET2) achieved 81.1% accuracy with a 7.7% standard deviation using RGB input. Slight improvements were seen with MASK2, WARPED1, and WARPED2 inputs, while MASK1 led to a negligible 0.02% drop in accuracy. WARPED1 yielded the most notable gain, improving accuracy by 1.23% and maintaining the same standard deviation. Class-specific analysis showed UNET2 performed best on 4 of 12 classes with both RGB and WARPED1 inputs, though the sets of classes differed. MASK2 achieved top accuracy for road, sidewalks, and cyclists, while WARPED2 was most effective for fences.

The first major modification to the U-Net architecture, UNET3, introduced a distinct structure incorporating a residual block to preprocess the previous segmentation mask. This block consisted of two consecutive Conv2d-BatchNorm2d-ReLU layers with 3×3 filters. Its output is fused with the input image and passed through a network identical to UNET2. The goal was to enhance performance by refining the mask before it enters the main network. Testing with WARPED2 input yielded the highest accuracy at 82.6%, a 1.5% improvement over UNET2 with RGB input, and reduced the standard deviation to 8.0%—a 1.9% decrease. UNET3 with WARPED2 performed best in 8 of 12 classes (void, buildings, road, sidewalks, vegetation, fences, vehicles, and cyclists), while UNET2/RGB led in the remaining four (sky, pillars, road signs, and pedestrians). Visual comparisons in Figure 2 show improved segmentation, particularly of pillars, when using UNET3/WARPED2, although both models struggled with motorcycles.

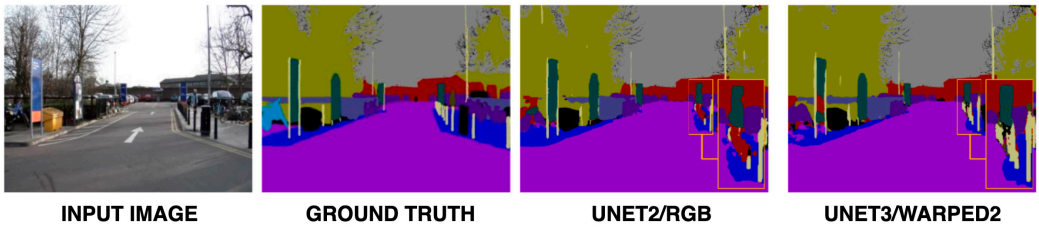


Figure 2. Visual comparison of UNET2/RGB and UNET3/WARPED2 models.

The UNET4 modification involves more extensive preprocessing of the previous segmentation output using a subnetwork identical to the first four encoder layers of UNET2. Its output, matched in dimension to the fourth layer of the RGB encoder, is linearly combined with it at the bottleneck. The highest accuracy was achieved with the WARPED1 input at 84.1%, a 3.0% improvement over UNET2 with RGB input. WARPED2 also performed well, reaching 84.0% accuracy with a standard deviation of 7.6%. These results indicate that optical flow-based warping and additional preprocessing enhance performance. In class-specific segmentation, UNET4 with WARPED1 input led in 6 of 12 classes (void, sky, buildings, road, vegetation, vehicles), while WARPED2 input performed best in 4 classes (sidewalks, traffic signs, fences, pedestrians). The MASK1 and MASK2 inputs surpassed the accuracy of other combinations for individual classes, cyclists, and poles, respectively. For comparison, the visualization of the UNET2/RGB combination is also presented in

the image. Visually, the UNET4/WARPED1 combination exhibits superior segmentation, particularly in the accurate delineation of poles and traffic signs as can be seen on Figure 3.

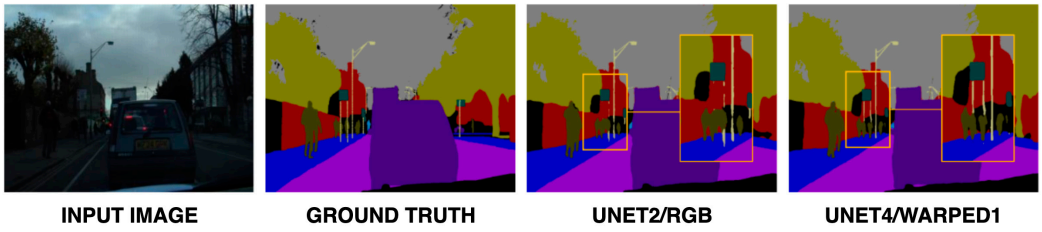


Figure 3. Visual comparison of UNET2/RGB and UNET4/WARPED1 models.

The final modification, UNET5, extends UNET2 by appending mask channels to its output and processing them through a residual block (as in UNET3), followed by a 1×1 convolution layer. This approach explores late-stage integration of temporal information (Figure 4). UNET5 achieved its highest accuracy of 82.0% with WARPED1 input—1.0% higher than UNET2/RGB—and reduced the standard deviation by 1.3% to 8.6%. Optical flow-based inputs (WARPED1 and WARPED2) outperformed MASK1 and MASK2, confirming mask shifting effectiveness. For class-specific accuracy, UNET5 with WARPED1 led in 6 out of 12 classes (void, buildings, pillars, vegetation, signs, vehicles). WARPED2 input excelled for roads and sidewalks, MASK2 for sky and fences, and MASK1 for pedestrians. However, UNET5 did not surpass UNET2/RGB in segmenting cyclists.

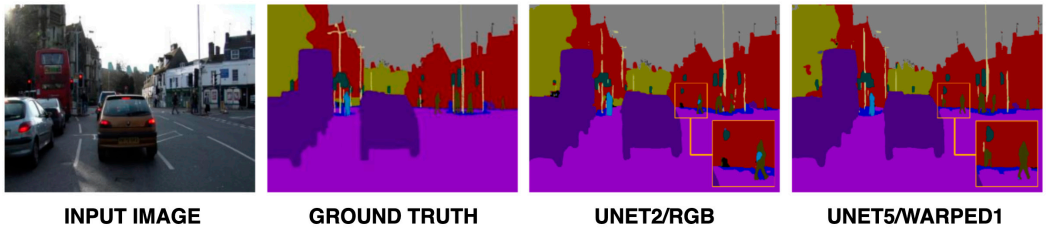


Figure 4. Visual comparison of UNET2/RGB and UNET5/WARPED1 models.

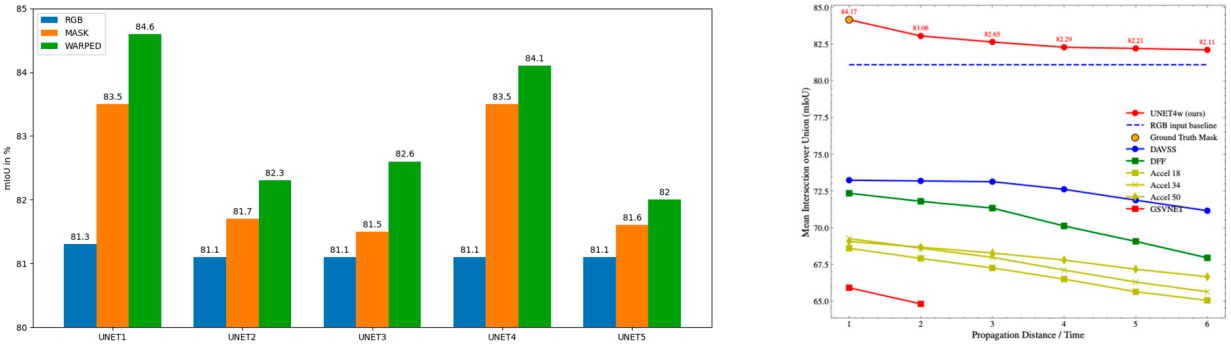
4. Discussion

In this study, we evaluated model performance based on the number of trainable parameters and GFLOPs for several reasons. The parameter count reflects memory requirements and the feasibility of deployment on resource-constrained devices such as mobile or embedded systems. GFLOPs quantify the computational workload during inference, directly measuring the model's real-world efficiency.

The original U-Net architecture (UNET1) contains 31.0 million parameters and incurs a computational cost of 510.2 GFLOPs when processing standard 3-channel RGB input. This increases marginally to 510.6 GFLOPs for 4-channel input (RGB + mask) and 515.0 GFLOPs for 15-channel input (RGB + one-hot encoded mask). The pruned version, UNET2, significantly reduces model size to 1.9 million parameters and achieves a substantial improvement in computational efficiency, requiring only 32.5 GFLOPs for 3-channel input—approximately 15.7 times more efficient than UNET1. For 4-channel and 15-channel inputs, UNET2 maintains this efficiency with GFLOPs of 32.6 and 33.7, respectively. UNET3 retains the same parameter count (1.9M) as UNET2 but introduces a residual block for early fusion. While GFLOPs are not applicable for 3-channel input due to architectural design, the model requires 32.6 GFLOPs for 4-channel input and 35.6 GFLOPs for 15-channel input, with a relative improvement factor of 14.47$\times$ compared to UNET1. UNET4 introduces mid-fusion processing and slightly increases the parameter count to 2.2 million. The computational cost rises to 41.6 GFLOPs for 4-channel input and 42.7 GFLOPs for 15-channel

input, corresponding to approximately 12.3× and 12.1× improvements over UNET1, respectively. Lastly, UNET5, which applies late fusion, contains 2.0 million parameters. It requires 34.8 GFLOPs for 4-channel input and 40.1 GFLOPs for 15-channel input, yielding improvements of about 14.7× and 12.8× over the baseline.

In terms of segmentation accuracy, the original U-Net architecture (UNET1) achieved the highest individual accuracy for RGB input images, with and without optical flow-based temporal enhancements, but required substantially more computational resources (trainable parameters and FLOPs). Among modified networks, the proposed UNET4w achieves a competitive mIoU of 84.17%, while notably maintaining the lowest parameter count (2.20M) and a GFLOPs value of 41.60. Among modified networks, the proposed UNET4w achieves a competitive mIoU of 84.17%, while notably maintaining the lowest parameter count (2.20M) and a GFLOPs value of 41.60. Although slightly outperformed by SERNet-Former [24] (84.62% mIoU) in terms of accuracy, UNET4w is highlighted as the optimal choice due to its superior trade-off between accuracy and computational efficiency. This selection emphasizes UNET4w's efficiency, scalability, and robustness, as further detailed in Figure 5a,b and Table 1.



(**a**) (**b**)

Figure 5. (**a**) The best mIoU for each network and input type; (**b**) Complexity of proposed U-Net modifications.

Table 1. Comparison to other video-based and image-based methods on CamVid dataset.

Method	mIoU (%)	Parameters (M)	GFLOPs
SERNet-Former	84.62	44.20	N/A
UNET4w (Ours)	84.17 (83.06t + 1)	2.20	41.60
PIDNet	80.10	7.60	47.60
PSPNet-101	79.70	48.88	55.44
ETC-MobileNet	78.20	3.20	0.615
TMANet-50	76.50	73.30	754.00
PSPNet-50	76.24	29.88	29.16
TD2-PSP50	76.00	65.37	541.00
DAVSS	72.46	56.00	272.19
PSPNet-18	71.00	12.82	12.88
TD4-PSP18	70.13	54.83	363.00
Accel-50	67.70	25.60	906.85
GSVNET	65.90	48.80	16.70

For video segmentation accuracy, we propose to use propagation distance vs. accuracy curve (PDA curve), which indicates how the segmentation accuracy changes along different propagation distances (Figure 5b). Unlike traditional methods such as Accel-50 [25] and GSVNET [26], which show substantial accuracy degradation with increasing propagation

distance, UNET4w maintains strong performance for short intervals. However, a notable accuracy drop occurs after two consecutive frames, with mIoU decreasing from 84.17% to 82.65% by the third frame. This suggests optimal performance is achieved when UNET4w is combined with an edge model providing near-ground truth segmentation every alternate frame, balancing accuracy and computational load. Computational efficiency is another significant advantage, as UNET4w's GFLOPs (~42.00) are substantially lower than other video-specific methods like TMANet-50 [27] (754.00 GFLOPs) and TD2-PSP50 [28] (541.00 GFLOPs).

Overall, these findings highlight that UNET4w effectively balances accuracy, computational cost, and temporal awareness, making it particularly suitable for real-world applications in autonomous systems. The ablation analysis in Table 1 confirms that UNET4w offers state-of-the-art performance while maintaining practical feasibility for real-world deployment.

All experiments were performed in Python 3.9 using PyTorch 1.12.1 (CUDA 11.3, cuDNN 8), with TorchVision, Torchaudio, and PyTorch Lightning; training and evaluation were tracked with TensorBoard. The computer-vision pipeline utilized OpenCV and Albumentations, together with segmentation-models-pytorch, timm, and efficientnet-pytorch. Data processing and analysis were conducted with NumPy, pandas, SciPy, scikit-learn, and Matplotlib in JupyterLab.

5. Conclusions

In this paper, we propose a distributed architecture for video semantic segmentation, leveraging temporal knowledge transfer by integrating server-generated segmentation masks into subsequent mobile device frames. We developed and evaluated four mobile-optimized U-Net variations (UNET2–UNET5), testing four mask incorporation methods (MASK1, MASK2, WARPED1, WARPED2) using the CamVid dataset. UNET4 consistently showed superior accuracy, particularly using the WARPED1 method, which, despite computational overhead from optical flow, effectively enhanced performance for small objects critical to autonomous driving. Our approach emphasizes computational efficiency through split computing, periodically employing a larger, accurate model to significantly reduce resource demands. Although training and validation used ground-truth masks, the strategy shows promise for efficient segmentation by offloading heavy computation to servers every 3–6 frames, enabling compact models to handle interim frames.

Author Contributions: Conceptualization, M.Č. and Š.M.; methodology, J.G. and G.B.; software, Š.M.; validation, M.Č.; formal analysis, M.Č.; investigation, Š.M.; resources, J.G. and G.B.; data curation, M.Č.; writing—original draft preparation, Š.M.; writing—review and editing, M.Č. and G.B.; visualization, M.Č. and Š.M.; supervision, J.G. and G.B.; project administration, G.B.; funding acquisition, J.G. and G.B.; All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Ministry of Education, Science, Research and Sport of the Slovak Republic, and the Slovak Academy of Sciences under Grant VEGA 1/0685/23 and by the Slovak Research and Development Agency under Grant APVV SK-CZ-RD-21-0028 and APVV-23-0512, and by Research and Innovation Authority VAIA under Grant 09I03-03-V04-00395.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets generated during and/or analyzed during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Duan, L.-Y.; Yu, X.-D.; Min, X.; Qi, T. Foreground Segmentation Using Motion Vectors in Sports Video. In *Advances in Multimedia Information Processing—PCM 2002*; Chen, Y.C., Chang, L.W., Hsu, C.T., Eds.; Springer: Berlin/Heidelberg, Germany, 2002; pp. 751–758.
2. Papazoglou, A.; Ferrari, V. Fast Object Segmentation in Unconstrained Video. In Proceedings of the 2013 IEEE International Conference on Computer Vision, Sydney, NSW, Australia, 1–8 December 2013; IEEE: Piscataway, NJ, USA, 2013; pp. 1777–1784. [CrossRef]
3. Zhang, G.; Yuan, Z.; Chen, D.; Liu, Y.; Zheng, N. Video Object Segmentation by Clustering Region Trajectories. In Proceedings of the 21st International Conference on Pattern Recognition (ICPR2012), Tsukuba, Japan, 11–15 November 2012; pp. 2598–2601. [CrossRef]
4. Liu, Y.; Liu, G.; Liu, C.; Sun, C. A Novel Color-Texture Descriptor Based on Local Histograms for Image Segmentation. *IEEE Access* **2019**, *7*, 160683–160695. [CrossRef]
5. Wu, T.; Gu, X.; Shao, J.; Zhou, R.; Li, Z. Color Image Segmentation Based on a Convex K-Means Approach. *arXiv* **2021**, arXiv:2103.09565. [CrossRef]
6. Wang, W.; Shen, J.; Yang, R.; Porikli, F. Saliency-Aware Video Object Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 20–33. [CrossRef] [PubMed]
7. Garcia-Garcia, A.; Orts-Escolano, S.; Oprea, S.; Villena-Martinez, V.; Rodríguez, J.G. A Review on Deep Learning Techniques Applied to Semantic Segmentation. *arXiv* **2017**, arXiv:1704.06857. [CrossRef]
8. Lateef, F.; Ruichek, Y. Survey on Semantic Segmentation Using Deep Learning Techniques. *Neurocomputing* **2019**, *338*, 321–348. [CrossRef]
9. Hao, S.; Zhou, Y.; Guo, Y. A Brief Survey on Semantic Segmentation with Deep Learning. *Neurocomputing* **2020**, *406*, 302–321. [CrossRef]
10. Sultana, F.; Sufian, A.; Dutta, P. Evolution of Image Segmentation Using Deep Convolutional Neural Network: A Survey. *Knowl.-Based Syst.* **2020**, *201–202*, 106062. [CrossRef]
11. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
12. Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, 7–12 June 2015; pp. 3431–3440. [CrossRef]
13. Shelhamer, E.; Long, J.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 640–651. [CrossRef] [PubMed]
14. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef] [PubMed]
15. Noh, H.; Hong, S.; Han, B. Learning Deconvolution Network for Semantic Segmentation. In Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, Chile, 7–13 December 2015; pp. 1520–1528. [CrossRef]
16. Zhou, L.; Zhang, C.; Wu, M. D-LinkNet: LinkNet with Pretrained Encoder and Dilated Convolution for High Resolution Satellite Imagery Road Extraction. In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), Salt Lake City, UT, USA, 18–22 June 2018; pp. 182–186. [CrossRef]
17. Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv* **2015**, arXiv:1505.04597. [CrossRef]
18. Ha, Q.; Watanabe, K.; Karasawa, T.; Ushiku, Y.; Harada, T. MFNet: Towards Real-Time Semantic Segmentation for Autonomous Vehicles with Multi-Spectral Scenes. In Proceedings of the 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Vancouver, BC, Canada, 24–28 September 2017; pp. 5108–5115. [CrossRef]
19. Kasar, P.; Jadhav, S.; Kansal, V. Brain Tumor Segmentation Using UNET and SEGNET: A Comparative Study. *Preprints* **2021**. [CrossRef]
20. Zhou, T.; Porikli, F.; Van Gool, L.; Wang, W.; Crandall, D. A Survey on Deep Learning Technique for Video Segmentation. *arXiv* **2021**, arXiv:2107.01153. [CrossRef] [PubMed]
21. Gadde, R.; Jampani, V.; Gehler, P.V. Semantic Video CNNs Through Representation Warping. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 4463–4472. [CrossRef]
22. Ding, M.; Wang, Z.; Zhou, B.; Shi, J.; Lu, Z.; Luo, P. Every Frame Counts: Joint Learning of Video Segmentation and Optical Flow. *AAAI Conf. Artif. Intell.* **2020**, *34*, 10713–10720. [CrossRef]
23. Chandra, S.; Couprie, C.; Kokkinos, I. Deep Spatio-Temporal Random Fields for Efficient Video Segmentation. *arXiv* **2018**, arXiv:1807.03148. [CrossRef]
24. Erisen, S. SERNet-Former: Semantic Segmentation by Efficient Residual Network with Attention-Boosting Gates and Attention-Fusion Networks. *arXiv* **2024**, arXiv:2401.15741. [CrossRef]

25. Jain, S.; Wang, X.; Gonzalez, J.E. Accel: A Corrective Fusion Network for Efficient Semantic Segmentation on Video. In Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 8858–8867. [CrossRef]
26. Lee, S.P.; Chen, S.C.; Peng, W.H. GSVNET: Guided Spatially-Varying Convolution for Fast Semantic Segmentation on Video. In Proceedings of the 2021 IEEE International Conference on Multimedia and Expo (ICME), Shenzhen, China, 5–9 July 2021; pp. 1–6. [CrossRef]
27. Wang, H.; Wang, W.; Liu, J. Temporal Memory Attention for Video Semantic Segmentation. In Proceedings of the 2021 IEEE International Conference on Image Processing (ICIP), Anchorage, AK, USA, 19–22 September 2021; pp. 2254–2258. [CrossRef]
28. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid Scene Parsing Network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 6230–6239. [CrossRef]