Revolutionizing navigation: Deep Learning for lane detection in mobile robots

Mohammed Hammoud, Ola Haydar, Ammar Asad, Zeinab Khadra, Sergey S. Lupin

***Abstract*—Mobile robots are becoming increasingly vital across various industries because of their ability to perform tasks safely, efficiently, and autonomously. A key feature of these robots and self-driving cars is lane detection, which can be particularly challenging owing to changing lighting and weather conditions. This study focuses on detecting lanes using deep learning-based image segmentation techniques. It employs transfer learning from pre-trained models on the ImageNet dataset, specifically examining architectures such as LinkNet, Unet, and Unet++. The study also explored various model backbones, including ResNet, VGG, Inception, and MobileNet. A benchmark dataset was utilized alongside data augmentation techniques to enhance the training volume and ensure robust validation. The results indicated that U-Net demonstrated the highest performance among all tested models, achieving an Intersection over Union (IoU) score of 0.848. This study contributes significantly to the development of reliable lane detection systems for mobile robotic applications.**

***Keywords*—Self-driving cars, Navigation, Mobile robots, Perception, Deep learning, Image segmentation**

I. INTRODUCTION

In recent years, advancements in autonomous vehicles and mobile robots have significantly transformed transportation and automation. Road recognition and lane detection are essential tasks, enabling them to understand and navigate their surroundings effectively. These technologies allow robots to operate on unstructured roads, thereby enhancing their versatility. Moreover, driver assistance systems have been developed to reduce the human errors and reduce the accidents risk. These systems rely on sensors, cameras, and algorithms to identify and track various road features such as road surfaces and lane markings. This information is crucial for controlling the steering, acceleration, and braking of a vehicle, enabling safe navigation and obstacle avoidance.

As our cities evolve, the road systems become more intricate, which means we need reliable detection systems that can work well in all sorts of situations. These situations can include a variety of lane designs, different driving environments, unpredictable weather, and shifting lighting conditions—like shadows, bright sunshine, and nighttime lights.

All of these factors can make lane markings tough to spot or even completely hidden by poor lighting or nearby vehicles and unique road layouts.

This sets the stage for understanding just how crucial lane detection is in building smart transportation systems, which could transform the way we move in the future.

Traditional or conventional methods for lane detection typically rely on visual cues and image-processing techniques. These approaches primarily use low-level features, such as image gradients and colors to extract valuable information from visual data [1]–[3].

By combining lane detection with Machine Learning (ML) and computer vision techniques, the capabilities of self-driving cars and mobile robots can be improved. Using ML- and computer vision-based lane detection techniques, robots can make real-time decisions based on their understanding of the environment with better performance than traditional methods [4]–[6].

EfficientNet is a cutting-edge convolutional neural network (CNN) known for its strong performance in computer vision tasks such as image classification and object detection. It achieves high efficiency through strategies such as compound scaling, which balances the network's depth, width, and resolution and employs the "swish" activation function. By contrast, MobileNet is specifically designed for mobile and embedded devices with limited resources. It uses depth-wise separable convolutions to lower computational complexity and reduce the number of parameters, resulting in smaller, faster models while maintaining high accuracy for tasks such as classification and detection. Both EfficientNet and MobileNet are pre-trained on large datasets, such as ImageNet, and can be fine-tuned effectively with minimal data.

This study aims to improve the lane detection capabilities of mobile robots and self-driving cars using Deep learning (DL)-based image segmentation techniques. Specifically, this study leveraged transfer learning from pre-trained models on the ImageNet dataset. It will evaluate various neural network architectures, including UNet, U-Net++, PSPNet, MANet, FPNet, and LinkNet, while also exploring different model backbones, such as ResNet, VGG, and MobileNet. The major contributions of our study are summarized as follows:

- Explore the application of DL-based image segmentation methods for effective lane detection in mobile robots and self-driving vehicles.
- Utilize transfer learning strategies with pre-trained models on the ImageNet dataset to improve the lane detection performance of mobile robots.
- Examine the efficacy of different model backbones such as ResNet, VGG, and MobileNet in enhancing lane detection accuracy.

The remainder of this paper is organized as follows. Section II reviews the literature, and Section III discusses the implementation details, including the dataset and evaluation metrics. Lastly, Section IV presents the results obtained and their discussion.

II. RELATED WORKS

A. Lane detection

Lane detection approaches can be categorized into several methods: segmentation-based methods [7]–[10], anchor-based lane detectors [11]–[13], row-based methods [12]–[18], parametric representation [19]–[21], keypoint association [10], [22]–[24], and curve-based methods [19]–[21].

Segmentation-based methods involve dividing an image into several segments, each corresponding to a different object or part of the image. These methods predict pixel-wise labels using supervised learning, with a focus on pixel-level predictions. However, they cannot be directly adapted to predict multiple lanes without modifying their model. The ultimate image segmentation goal is simplifying the image to make it easier to analyze and understand.

In contrast, anchor-based methods search for a limited number of key features using predefined anchor points, which helps reduce computation time. They estimated lane shapes by regressing the relative coordinates of these anchors, allowing for accurate lane detection through refinement of predefined anchors.

Row-based methods define lanes by dividing the image into slices or cells, and rely on predefined lines or row anchors to guide lane detection. This task can be addressed through either offset regression or rowwise classification. These methods optimize lane shape by regressing the relative coordinates.

Parametric representation methods represent lanes as a set of curve parameters. Lane detection involves fitting these parameters to the actual lane shapes and regressing them accordingly.

Keypoint-based methods focus on detecting critical points (keypoints) along the boundaries of lane lines. They treated lane line prediction as a key point estimation task. These methods can adapt to complex road conditions by first predicting all potential key points that likely belong to lane lines and subsequently inferring lane shapes by grouping these key points into distinct lane instances.

Curve-based methods can also be utilized, where lane detection is approached by fitting curves to the detected features in the image.

In summary, each category of lane detection method has distinct approaches and focuses, allowing them to adapt to different challenges in lane detection tasks. This study focused on image-segmentation-based methods.

B. Image segmentation-based methods

There are two main types of DL image segmentation: semantic and instance. Semantic segmentation is pixel-based classification, where each pixel is labelled according to its object category (e.g., cars, pedestrians, buildings, and lanes), aiding object detection and scene understanding. In contrast, instance segmentation also labels pixels, but distinguishes between individual instances of the same object, such as identifying multiple cars in one image. This method is useful for tracking and counting the objects.

U-Net [25] is a DL model widely used for image segmentation, particularly in medical imaging, and performs well with small datasets. The architecture includes an encoder that downsamples the input image and a decoder that upsamples the output connected by skip connections to preserve spatial information. Similarly, U-Net++ [26] is the same as Unet, but with nested skip pathways to enhance feature propagation and fine detail learning, leading to better segmentation. Although Unet is effective for simpler tasks, U-Net++ offers greater accuracy and complexity, making it more suitable for intricate segmentation challenges.

Feature Pyramid Network (FPN) is a top-down architecture that creates high-level semantic feature maps at different scales, which helps to improve multiscale image segmentation. By enriching the feature maps with both high- and low-resolution information, FPN enhanced the model's ability to detect objects of various sizes. This architecture is particularly powerful for tasks that require detection and segmentation, such as those found in instance segmentation frameworks. Pyramid Scene Parsing Network (PSPNet) [27] excels at understanding the global context through its pyramid pooling strategy, making it superior for natural image segmentation. In contrast, FPN is particularly beneficial in multitasking scenarios, where there is significant variability in the object scale.

Multi-Attention Network (MANet) [28] incorporates multi-attention mechanisms to enhance feature extraction and improve segmentation accuracy. This model focuses on different areas of an image to prioritize the important features while minimizing noise. MANet is designed to effectively handle contextual information, which helps improve segmentation results in challenging environments. Although its complexity may require more computational resources, this results in significant gains in precision.

LinkNet [29] was designed for efficient segmentation and balancing performance with high computational efficiency. Using a lightweight encoder-decoder architecture, the encoder and decoder were connected through shortcut connections to facilitate an effective information flow. Although LinkNet may not achieve the same level of detail as more complex models, it is well-suited for real-time applications due to its speed and lower resource requirements. This enables the training of models with fewer parameters while allowing extensive calculations to be performed efficiently. LinkNet outperforms other architectures, such as SegNet, ENet, Dilation 8/10, and DeepLab CRF. Additionally, it is applicable to embedded systems that have been tested on NVIDIA TX1 and NVIDIA Titan X. Although LinkNet is more lightweight, it may sacrifice some accuracy compared to models with more complex architectures, such as U-Net++ and PSPNet, making it less suitable for applications where precision is paramount.

While MANet's attention mechanisms provide notable improvements in handling contextual information, the additional computational expense may limit their suitability for real-time scenarios compared to faster models such as LinkNet. In summary, selecting a model is largely determined by segmentation task requirements, the hardware specifications, and the tradeoff between accuracy and efficiency. Each model has unique strengths and weaknesses, offering a range of tools for different challenges in segmentation tasks.

III. Methods and Methodology

A. Dataset and Implementation details

We used a public dataset [30]. It consists of street images of roads under various conditions in real-life scenarios, as illustrated in Figure 1. After processing (removing the images that do not have corresponding masks), we obtained approximately 3264 images with their corresponding masks, with various resolutions.

Fig. 1. Samples from the dataset showing the images with its corresponding masks.

As mentioned previously, the images have various dimensions; therefore, as the first step, the images are resized to a size of 512 × 512. As a preprocessing step, the images were enhanced using Contrast Limited Adaptive Histogram Equalization (CLANE). CLANE is a technique utilized in image processing to enhance images' contrast. CLANE is a modified version of the classical histogram equalization method that adapts to the local contrast of an image. In CLANE, the image is split into small areas called tiles, and histogram equalization is applied to each tile separately. This helps to prevent over-enhancement of the image and preserves the local contrast. For DL models, we investigated the use of various model architectures with transfer learning. These models include Unet, Unet++, PSPNet, Linknet, MAnet, and FPN. The pretrained models on the famous ImageNet dataset are applied using various backbones (encoder network weights), such as VGG, ResNet, and MobileNet. The data were split into training, validation, and testing sets in a ratio of 0.7:0.1:0.2. We used a focal loss function [31] as a loss function for training, as shown in Equation 1, where we used γ = 2. Focal loss is helpful because we are working with imbalanced data. It places greater emphasis on hard-to-detect classes, often the minority class. It achieves this by down weighting easy-to-predict classes, allowing for more focus on those that are challenging to detect. The weight of the best model was saved based on the best-acquired validation loss.

Python [1] is used as a programming language, while PyTorch [2], Keras, and segmentation models API [3] are used for model implementation. Additionally, OpenCV [4] is used to perform image processing. Data augmentation was not applied in this study because of the insufficient dataset size.

$$\mathrm{FL}(p_t) = - \alpha\, (1 - p_t)^{\gamma} \log(p_t) \qquad (1)$$

$$\mathrm{IoU} = 2 * \frac{\mathrm{P} \cap \mathrm{G}}{\mathrm{P} \cup \mathrm{G}} \qquad (2)$$

$$\mathrm{F1}_{\mathrm{score}} = 2 * \frac{\mathrm{Precision} * \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}} \qquad (3)$$

$$\mathrm{Precision} = \frac{\mathrm{T_P}}{\mathrm{T_P} + \mathrm{F_P}} \qquad (4)$$

$$\mathrm{Recall} = \frac{\mathrm{T_P}}{\mathrm{T_P} + \mathrm{F_N}} \qquad (5)$$

$$\mathrm{Specificity} = \frac{\mathrm{T}_N}{\mathrm{T_N} + \mathrm{F_P}} \qquad (6)$$

As evaluation metrics, we used three metrics, represented by Intersection over union or Jaccard index (IoU), F1 score, precision, recall, and specificity. The IoU is the main metric to select the best model in our evaluation.

IV. Results and Discusioin

In this study, we evaluated various DL architectures for lane street detection, as listed in Table I. The U-Net architecture with the timmmobilenetv3_large_075 encoder demonstrated the highest performance across all metrics, achieving an IoU of 0.8479, an F1 score of 0.9161, precision of 0.9019, sensitivity of 0.9353, specificity of 0.9985, and an accuracy of 0.9973. The model size was recorded at 113.46 MB, indicating a good balance between model efficiency and detection accuracy.

The UNet++ architecture paired with the ResNet18 encoder followed closely, yielding an IoU of 0.8356 and an F1 score of 0.9087. This model exhibited slightly lower precision (0.8855) and sensitivity (0.9387) than the top-performing U-Net model but still maintained commendable specificity (0.9981) and accuracy (0.9969). The model size for this configuration was larger, at 161.70 MB.

The performance of LinkNet models showed a consistent trend with respect to IoU and F1 scores across different encoder types. The best-performing configuration was the one using vgg11, which achieved an IoU of 0.8298 and an F1 score of 0.9054. Precision ratings across the LinkNet configurations ranged from 0.9002 to 0.9051, while sensitivity values were more variable, indicating that some configurations were better at detecting lane markings than others. The model size here

[1] https://www.python.org/

[2] https://pytorch.org/

[3] https://smp.readthedocs.io/en/latest/index.html

[4] https://opencv.org/

was notably smaller at 41.99 MB, indicating high efficiency in resource utilization.

In contrast, the FPN models showed a significant performance drop, with the mobilenet_v2 encoder yielding only an IoU of 0.1785 and an F1 score of 0.2691, which underlines that FPN architectures are struggling for this particular task, as they probably cannot capture necessary features for lane detection in an appropriate way.

TABLE. I. THE TESTING RESULTS

Model	**Backbone**	**iou**	**F1**	**Prc**	**Sen**	**Spec**	**Acc**	**Model size (MB)**
unet	timm-mobilenetv3_large_075	0.848	0.916	0.902	0.935	0.999	0.997	113.460
UnetPlusPlus	resnet18	0.836	0.909	0.886	0.939	0.998	0.997	161.703
UnetPlusPlus	mobilenet_v2	0.831	0.905	0.903	0.914	0.998	0.997	137.946
Linknet	vgg11	0.830	0.905	0.905	0.913	0.999	0.997	41.998
Linknet	mobilenet_v2	0.822	0.901	0.908	0.900	0.999	0.997	17.570
Linknet	resnet34	0.822	0.900	0.900	0.906	0.998	0.997	87.274
MAnet	resnet18	0.819	0.898	0.914	0.890	0.999	0.997	200.356
FPN	resnet18	0.818	0.897	0.892	0.909	0.998	0.997	52.282
unet	vgg11	0.816	0.897	0.904	0.898	0.998	0.997	181.358
unet	resnet34	0.816	0.897	0.901	0.900	0.998	0.997	192.277
unet	resnet18	0.814	0.895	0.922	0.877	0.999	0.997	151.781
Linknet	resnet18	0.813	0.894	0.918	0.880	0.998	0.997	46.779
UnetPlusPlus	resnet34	0.809	0.892	0.924	0.870	0.999	0.996	202.199
unet	mobilenet_v2	0.805	0.890	0.897	0.891	0.998	0.997	136.193
MAnet	mobilenet_v2	0.800	0.886	0.893	0.890	0.998	0.996	296.601
FPN	resnet34	0.762	0.861	0.933	0.807	0.999	0.996	92.778
PSPNet	resnet18	0.745	0.848	0.855	0.847	0.998	0.996	45.415
PSPNet	resnet34	0.743	0.845	0.864	0.833	0.998	0.996	85.911
MAnet	resnet34	0.720	0.828	0.757	0.948	0.996	0.995	240.851
PSPNet	mobilenet_v2	0.712	0.823	0.857	0.799	0.998	0.995	9.317
FPN	mobilenet_v2	0.179	0.269	0.191	0.842	0.872	0.871	17.124

Fig. 2. Visual representation of the performance of the best-model on some testing data, where the green and magneta colors stand for the true lane and the predicted lane, respectively.

An important observation in the results is the very high specificity across almost all models, which signifies that the models effectively recognize when lane markings are not present. The specificity of all the models is greater than 0.998, which reflects that all models do a great job of reducing false positives. The sensitivity values have varied quite significantly among the models, with the highest being from U-Net at 0.9353. In contrast, some models showed very low sensitivity, such as FPN with mobilenet_v2 at 0.8416, indicating possible weaknesses in detecting lane markings under certain conditions.

Interestingly, while U-Net and U-Net++ have better accuracy and detection metrics, this comes at the cost of model size. The MAnet, despite poorer results, has a relatively bigger model size of 296.60 MB and can be a problem when deployed in resource-constrained environments. On the other hand, a smaller LinkNet (41.99 MB) showcases its applicability to real-time scenarios where efficiency and accuracy at detection are crucial.

The results demonstrate the U-Net architecture, when combined with a timm-mobilenetv3_large_075 encoders, is the most successful one for lane street detection in view of superior metrics and performance with respect to model size. Other architectures, such as PSPNet and MAnet, still need extensive tuning to arrive at performance comparable to U-Net. Future work can involve fine-tuning these models, designing hybrid architectures, or preparing better training datasets for better general performance in lane detection tasks.

A. Challenges and Limitations

While our transfer-learning-based approach has shown impressive performance, validating these findings on a more extensive dataset is crucial to ensure robustness and generalizability. Additionally, a thorough comparison with existing research is necessary to accurately position our work within the broader context of the field. By addressing these aspects, we can enhance the credibility of our approach and emphasize its unique contributions to advancing our understanding in this area. Moreover, the effectiveness of this model in real-time applications has yet to be tested. A significant drawback of image-based methods for lane detection is their inability to represent lane lines as masks accurately.

Although these approaches are stable and can detect visible lanes, they need lots of computational power and memory to be better for systems with limited resources. The latency of deep learning models varies based on their complexity and the hardware used, which can hinder their ability to respond in real time. Besides that, the DL model's performance depends on the data quality, so these models must be trained on big data containing all possible scenarios of real-life situations, or else they will overfit and have poor generalization and performance under new conditions.

V. Conclusion

This study underscores the pivotal role of deep learning-based image segmentation techniques in advancing lane detection capabilities essential for mobile robots and self-driving vehicles. By harnessing transfer learning from pre-trained models on the ImageNet dataset and systematically exploring multiple neural network architectures, we achieved significant improvements in both the reliability and accuracy of lane detection systems. Notably, the U-Net and Unet++ architectures demonstrated exceptional performance, with the U-Net achieving an IoU score of 0.8479, thereby illustrating the effectiveness of modern deep learning frameworks in navigating the complexities present in diverse driving environments and challenging conditions.

As the demand for autonomous systems escalates, this study lays a robust foundation for the ongoing development of intelligent transportation solutions. These advancements pave the way for safer and more efficient navigation strategies for mobile robots across various settings. Future research should aim to enhance the adaptability and robustness of these lane detection models, focusing on addressing nuances posed by dynamic road conditions, varying environmental factors, and diverse traffic scenarios. Emphasizing real-time applications, this work contributes significantly to the evolution of smart mobility solutions and the broader field of autonomous navigation, ultimately fostering a more reliable and integrated intelligent transportation ecosystem. The source code for this work can be found here [5].

REFERENCES

[1] Y. He, H. Wang, and B. Zhang, “Color-based road detection in urban traffic scenes,” IEEE Transactions on intelligent transportation systems, vol. 5, no. 4, pp. 309–318, 2004.

[2] A. Bar Hillel, R. Lerner, D. Levi, and G. Raz, “Recent progress in road and lane detection: a survey,” Machine vision and applications, vol. 25, no. 3, pp. 727–745, 2014.

[3] T.-Y. Sun, S.-J. Tsai, and V. Chan, “Hsi color model based lane-marking detection,” in 2006 ieee intelligent transportation systems conference. IEEE, 2006, pp. 1168–1172.

[4] A. Borkar, M. Hayes, and M. T. Smith, “A novel lane detection system with efficient ground truth generation,” IEEE Transactions on Intelligent Transportation Systems, vol. 13, no. 1, pp. 365–374, 2011.

[5] H. Jung, J. Min, and J. Kim, “An efficient lane detection algorithm for lane departure detection,” in 2013 IEEE Intelligent vehicles symposium (IV). IEEE, 2013, pp. 976–981.

[6] J. Niu, J. Lu, M. Xu, P. Lv, and X. Zhao, “Robust lane detection using two-stage feature extraction with curve fitting,”Pattern Recognition, vol. 59, pp. 225–233, 2016.

[7] X. Pan, J. Shi, P. Luo, X. Wang, and X. Tang, “Spatial as deep: Spatial cnn for traffic scene understanding,” in Proceedings of the AAAI conference on artificial intelligence, vol. 32, no. 1, 2018.

[8] D. Neven, B. De Brabandere, S. Georgoulis, M. Proesmans, and L. Van Gool, “Towards end-to-end lane detection: an instance segmentation approach,” in 2018 IEEE intelligent vehicles symposium (IV). IEEE, 2018, pp. 286–291.

[9] T. Zheng, H. Fang, Y. Zhang, W. Tang, Z. Yang, H. Liu, and D. Cai, “Resa: Recurrent feature-shift aggregator for lane detection,” in Proceedings of the AAAI conference on artificial intelligence, vol. 35, no. 4, 2021, pp. 3547–3554.

[10] Y. Ko, Y. Lee, S. Azam, F. Munir, M. Jeon, and W. Pedrycz, “Key points estimation and point instance segmentation approach for lane detection,” IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 7, pp. 8949–8958, 2021.

[11] D. Jin, W. Park, S.-G. Jeong, H. Kwon, and C.-S. Kim, “Eigenlanes: Data-driven lane descriptors for structurally diverse lanes,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 17 163–17 171.

[12] L. Tabelini, R. Berriel, T. M. Paixao, C. Badue, A. F.De Souza, and T. Oliveira-Santos, “Keep your eyes on the lane: Real-time attention-guided lane detection,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 294–302.

[13] X. Li, J. Li, X. Hu, and J. Yang, “Line-cnn: End-to-end traffic line detection with line proposal unit,” IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 1, pp. 248–258, 2019.

[14] J. Philion, “Fastdraw: Addressing the long tail of lane detection by adapting a sequential prediction network,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 11 582–11 591.

[15] Z. Qin, H. Wang, and X. Li, “Ultra fast structure-aware deep lane detection,” in Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXIV 16. Springer, 2020, pp. 276–291.

[16] T. Zheng, Y. Huang, Y. Liu, W. Tang, Z. Yang, D. Cai, and X. He, “Clrnet: Cross layer refinement network for lane detection,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 898–907.

[17] L. Liu, X. Chen, S. Zhu, and P. Tan, “Condlanenet: a top-to-down lane detection framework based on conditional convolution,” in Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 3773–3782.

[18] H. Honda and Y. Uchida, “Clrernet: improving confidence of lane detection with laneiou,” in Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, 2024, pp. 1176–1185.

[19] Z. Feng, S. Guo, X. Tan, K. Xu, M. Wang, and L. Ma, “Rethinking efficient lane detection via curve modeling,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 17 062–17 070.

[20] R. Liu, Z. Yuan, T. Liu, and Z. Xiong, “End-to-end lane shape prediction with transformers,” in Proceedings of the IEEE/CVF winter conference on applications of computer vision, 2021, pp. 3694–3702.

[21] L. Tabelini, R. Berriel, T. M. Paixao, C. Badue, A. F. De Souza, and T. Oliveira-Santos, “Polylanenet: Lane estimation via deep polynomial regression,” in 2020 25th International Conference on Pattern Recognition (ICPR). IEEE, 2021, pp. 6150–6156.

[22] S. Xu, X. Cai, B. Zhao, L. Zhang, H. Xu, Y. Fu, and X. Xue, “Rclane: Relay chain prediction for lane detection,” in European Conference on Computer Vision. Springer, 2022, pp. 461–477.

[23] J. Wang, Y. Ma, S. Huang, T. Hui, F. Wang, C. Qian, and T. Zhang, “A keypoint-based global association network for lane detection,” in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 1392–1401.

[24] Z. Qu, H. Jin, Y. Zhou, Z. Yang, and W. Zhang, “Focus on local: Detecting lane marker from bottom up via key point,” in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2021, pp. 14 122–14 130.

[25] O. Ronneberger, P. Fischer, and T. Brox, “U-net: Convolutional networks for biomedical image segmentation,” in Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18. Springer International Publishing, 2015, pp. 234–241.

[26] Z. Zhou, M. M. Rahman Siddiquee, N. Tajbakhsh, and J. Liang, “Unet++: A nested u-net architecture for medical image segmentation,” in Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: 4th International Workshop, DLMIA 2018, and 8th International Workshop, ML-CDS 2018, Held in Conjunction with MICCAI 2018, Granada, Spain, September 20, 2018,

[5] https://github.com/Hammoudmsh/Revolutionizing-Navigation-Deep-Learning-for-Lane-Detection-in-Mobile-Robots.git

[27] Proceedings 4. Springer, 2018, pp. 3–11. [27] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid scene parsing network,” in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2881–2890.

[28] R. LI, S. Zheng, C. Zhang, C. Duan, J. Su, L. Wang, and P. Atkinson, “Multiattention network for semantic segmentation of fine-resolution remote sensing images,” IEEE Transactions on Geoscience and Remote Sensing, vol. PP, pp. 1–13, 07 2021.

[29] A. Chaurasia and E. Culurciello, “Linknet: Exploiting encoder representations for efficient semantic segmentation,” in 2017 IEEE visual communications and image processing (VCIP). IEEE, 2017, pp. 1–4.

[30] M. OKUYAR, “Lane segmentation,” 2022. [Online]. Available: https://www.kaggle.com/dsv/4048355

[31] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Doll´ar, “Focal loss for dense object detection,” in Proceedings of the IEEE international conference on computer vision, 2017, pp. 2980–2988.

Paper received 16 April 2026.

Mohammed Hammoud, Ph.D. student, Skolkovo Institute of Science and Technology, (e-mail: Mohammed.Hammoud@skoltech.ru).

Ola Haydar, Ph.D. student, National Research University of Electronic Technology «MIET», (e-mail: oulahaider@gmail.com).

Ammar Asad, Ph.D. student, Moscow Polytechnic University, (e-mail: ammarasad09@gmail.com).

Zeinab Khadra, Ph.D. student, Moscow Polytechnic University, (e-mail: zeinabkhadrah@gmail.com).

Sergey S. Lupin, Ph.D., Ass. Professor, National Research University of Electronic Technology «MIET», (e-mail: Sergeylupin.jr@gmail.com);