applied sciences

MDPI

Article

Fast Drivable Areas Estimation with Multi-Task Learning for Real-Time Autonomous Driving Assistant

Dong-Gyu Lee

Department of Artificial Intelligence, Kyungpook National University, Daegu 700010, Korea; dglee@knu.ac.kr; Tel.: +82-53-950-4559

Abstract: Autonomous driving is a safety-critical application that requires a high-level understanding of computer vision with real-time inference. In this study, we focus on the computational efficiency of an important factor by improving the running time and performing multiple tasks simultaneously for practical applications. We propose a fast and accurate multi-task learning-based architecture for joint segmentation of drivable area, lane line, and classification of the scene. An encoder–decoder architecture efficiently handles input frames through shared representation. A comprehensive understanding of the driving environment is improved by generalization and regularization from different tasks. The proposed method learns end-to-end through multi-task learning on a very challenging Berkeley Deep Drive dataset and shows its robustness for three tasks in autonomous driving. Experimental results show that the proposed method outperforms other multi-task learning approaches in both speed and accuracy. The computational efficiency of the method was over 93.81 fps at inference, enabling execution in real-time.

Keywords: drivable area estimation; autonomous driving; multi-task learning; lane line detection; scene classification; real-time processing

Citation: Lee, D.-G. Fast Drivable Areas Estimation with Multi-Task Learning for Real-Time Autonomous Driving Assistant. *Appl. Sci.* **2021**, *11*, 10713. https://doi.org/10.3390/app112210713

Academic Editor: Deokwoo Lee

Received: 17 October 2021
Accepted: 11 November 2021
Published: 13 November 2021

1. Introduction

Recent studies on intelligent vehicles have advanced rapidly with the development of deep learning-based computer vision techniques and much attention to self-driving vehicles. Visual perception is playing a key role in this revolution. It enables the vehicle to understand the driving environments and estimate drivable areas. The drivable area estimation for safe driving requires a more high-level understanding than other tasks such as free space detection, vehicle detection, pedestrian detection, and lane line detection [1]. According to the current environment in real-time, a drivable area is necessarily divided into ego-lanes, other drivable lanes, sidewalks, and so forth [2].

Although many methods have shown successful results in drivable area estimation [3], it is not straightforward to apply these methods to real-world applications. Knowing the position of the lane line is essential to move the vehicle correctly on the road and avoid traffic accidents with others. Single tasks, such as lane line segmentation, object detection and semantic segmentation, alone, are insufficient to decide road affordability of driving because many areas have different conditions such as the absence of lanes [4]. Furthermore, the road is shared with all other vehicles. If the lane is occupied by an obstacle or the free space of other vehicles coming from the opposite direction, it should also be considered. Thus, understanding drivable areas to set the proper path where the ego vehicle can drive is an essential function for safe driving. Although most studies have focused on improving performance to address practical real-world autonomous driving applications that have limited resources, reducing computational cost is also a vital factor. In particular, as safety is vital for autonomous driving, the required perception tasks for autonomous driving are diversified, whereas resources for understanding them are limited.

To address this issue, herein, we adopt the multi-task learning approach to small deep network architecture to understand the environmental context of current driving vehicles

and reduce the computation for real-time inference. This network has desirable properties, such as low-power usage, real-time processing, and small model size while maintaining competitive accuracy. Multi-task learning can improve learning efficiency and prediction accuracy by learning multiple objectives from a shared representation [5]. Furthermore, the multi-task learning approach allows the model to perform in real-time by combining all loss functions to simultaneously learn multiple objectives in a single small model from shared computations. For elaborate estimation of drivable areas, we performed three tasks simultaneously as follows: (1) drivable area estimation, that pixel-wise classification of road that vehicles can forward; (2) lane line segmentation, that detecting lane lines to divide directly and alternatively drivable area, and (3) scene classification, that is, understanding the global context of current driving environments. Figure 1 shows an input image and three tasks for drivable area estimation. In Figure 1b, the red color region corresponds to the ego-lane and blue color region indicates the alternative drivable lane. In Figure 1b, the red color region corresponds to the ego-lane and the blue color region indicates the alternative drivable lane. The drivable area is more likely around the lane lines. Therefore, a unified structure that shares the representation can achieve not only better computation efficiency, but also a better performance than separate architecture.

(**a**) Input (**b**) Result

Figure 1. Our goal: Solving drivable area estimation, lane line segmentation, and scene classification through multi-task learning.

In this paper, we propose a fast, simple, efficient, and accurate end-to-end neural network architecture to enhance the simultaneous estimation of drivable areas, lane line segmentation, and scene classification based on a multi-task learning approach. We focus on computational times with competitive performances to enable safety-critical real-time applications. To apply to three tasks, we used an encoder that generates feature maps and a decoder that predicts the result of each task from shared feature maps. The comprehensive understanding of the driving environment is efficiently learned through multi-task learning. Our approach is very simple, can be trained end-to-end, and performs well in the challenging Berkeley deep drive (BDD) dataset [4], showing a much faster computational time with high performance in all three tasks. In summary, the three contributions of our work are:

- Drivable area estimation, lane line segmentation, and scene classification are integrated into one end-to-end framework;
- The proposed method can perform multi-task inference in real-time by utilizing shared representation efficiently;
- We demonstrated the effectiveness of the proposed method on the public BDD dataset.

The subsequent sections of the paper are organized as follows: In Section 2, we introduce the studies related to our work. We discuss the overall architecture in Section 3. The experimental results are detailed and discussed in Section 4. Finally, we present the conclusions of this work in Section 5.

2. Related Work

The three tasks for intelligent driving are handled separately. In this section, we introduce the research progress of the three tasks. Then some prominent multi-task networks and efficient networks are introduced

2.1. Drivable Area Estimation

Most of the published papers focus on improving performance from the learning method and network architecture. RBNet [6] learns to estimate the road and the road boundary simultaneously based on a Bayesian model. Han et al. [7] proposed a road detection method based on generative adversarial networks (GAN) and a weakly supervised learning method based on conditional GAN. Munoz et al. [8] proposed an enhanced generalization technique for road detection with a random data augmentation method.

Although significant performance has been achieved in the segmentation task based on the deep-learning approaches, the trade-off between accuracy and speed is still a challenge. Asgarian et al [9] considered the process of drivable area estimation as a row-selection task for efficient computation. A pixel-wise semantic segmentation network, SegNet [10], uses encoders and decoders to minimize the required memory and computational cost. The ErfNet [11] has fewer computational costs while providing better performances than SegNet by exploiting the skip connections and 1D kernels.

2.2. Lane Line Segmentation

Lane line segmentation is a fundamental topic in the field of intelligent vehicles. Despite the many studies in the literature, recently, many learning-based approaches have been conducted to enhance its robustness. Neven et al. [12] proposed a lane detection method based on the instance segmentation approach. In their method, each lane line is considered as an instance. Expanded Self Attention (ESA) [13] propose a self-attention mechanism that can predict the confidence of a lane along with the vertical and horizontal directions. The ESA module extracts global contextual information by predicting the confidence of the lane. SCNN [14] generalizes traditional deep layer by layer convolutions to slice-by-slice convolutions. The message-passing procedure among consecutive pixels make advantages in the segmentation performance. We argue that the main advantage of the proposed method is their ability to infer lane line and drivable area at same time in one forward path processing.

2.3. Scene Classification

Starting with the AlexNet [15], most of the image classification tasks utilize deep learning. The residual network [16] allows the training of very deep networks without exploding or vanishing gradient problems through skip connection. In the sense of scene classification, deep neural networks that can effectively process an entire input image at a fast speed are widely used [17,18]. In this paper, we classify road types to share the global representation with drivable area estimation and lane line segmentation.

2.4. Multi-Task Learning for Intelligent Vehicle

The idea of a better representation through a multi-task learning approach by exploiting many tasks has been adopted in this field. Several approaches have been proposed in the context of CNNs [19,20]. Mask R-CNN [21] performed instance segmentation and object detection at same time. BlitzNet [22] achieved real-time inference for scene understanding. However, these networks are more like computer vision methods than intelligent vehicle methods. Some researches focus on intelligent vehicle system that can be applied to traffic scenarios directly. MultiNet [1] proposed a joint classification, detection and semantic segmentation problem via a unified architecture that can run in real-time. The architecture shares the same encoder and divides into three decoders for three tasks. DLT-Net [3] simultaneously handles drivable area estimation, lane line detection, and traffic object detection problem in a single framework. The context tensor between sub-task decoders

effectively shares designate influence among tasks. Uhrig et al. learn semantic and instance segmentations under a classification setting. Multi-task deep learning has also been used for geometry and regression tasks. Fabio et al. [2] detect navigable area by estimating free space inside each lane. The homoscedastic uncertainty estimation is used to achieve better performances. In this paper, we propose an encoder and decoder-based multi-task learning architecture for the comprehensive understanding of the driving environment through shared representation.

3. Methods

In this section, the architecture of the proposed method is described. The overall architectures of both the encoder and decoder are shown in Figures 2 and 3, respectively. Our aim is to estimate drivable ares, lane lines and classify the scene. The network shares an encoder to generate shared representation. The feature maps are divided into three decoders for each task. The multi-task likelihood loss is simultaneously calculated for three different tasks.

3.1. Encoder-Decoder Structure

Our model aims at a fast and accurate estimation of drivable areas with multi-task learning approach. For this purpose, we designed an encoder and three decoders architecture to extract rich features that contain all necessary information to perform the tasks of drivable area segmentation, lane line segmentation, and scene classification from shared computation. We conduct our network based on the MobileNetV3-Large networks [18] with the lite reduced atrous spatial pyramid pooling (LR-ASPP) module to process input frames. There are two main reasons that we build our network based on this architecture as follows: (1) autonomous vehicles generally have better computational resources than mobile phones, (2) it shows sufficiently fast speed and high accuracy.

In the backbone network, depthwise separable convolutions consists of two separate layers: a light weight depthwise convolution for spatial filtering and a heavier 1×1 pointwise convolution for feature generation. The depthwise separable convolution efficiently factorizes traditional convolution by separating spatial filtering at feature generation mechanisms. The removal of the both filtering layer and the projection of the bottleneck layer enables fast computation. A linear bottleneck and inverted residual structure create an efficient hierarchical structure. We employ the NetAdapt algorithm [23] to discover the optimal number of filters per layer. A combination of these methods creates more efficient building blocks. The global network structure is found by optimizing each network block and platform-aware neural architecture search (NAS). The hard sigmoid is employed to preserve the accuracy of the fixed-point arithmetic while replacing the inefficient operation. The detail of a backbone network structure is shown in Table 1. SE denotes whether there is a Squeeze-and-Excite [24] in that block. The LR-ASPP module enables efficient encoding for the segmentation by deploying the global average pooling with a combination of squeeze and excitation in a residual layer.

In Figure 2, two branches consist of a 1×1 convolution and the global-average pooling with a large stride and only 1×1 convolution. BN denotes a batch normalization. To process three tasks in a single forward pass, the encoder generates feature maps f and f', then the decoder predicts the results of the three tasks using shared representation for the encoding efficiency. For the multi-task learning and further to obtain a faster inference speed during the encoding process, the feature map f is extracted from the fifth layer, and feature map f' comes from the layer just before the final convolution operation. The atrous convolution is applied to the last block of the network to extract denser features, and low-level features are captured to provide detailed information for the segmentation. Then, two feature maps, f and f', are used to feed the three decoders. At the decoding stage, the decoder assembles each feature map for each task. All decoders consist of following configuration; (1) 1×1 convolution to reduce channel size to the number of classes for each task; (2) batch normalization; and (3) ReLU activation functions. In Figure 3, the feature

map f is used for the multi-task inference, scene classification, drivable area estimation and lane line segmentation task, and the feature map f' is used for drivable area estimation and lane line segmentation, which are the tasks that require more detailed information for semantic segmentation (or any dense pixel prediction). Both drivable area estimation task and lane line segmentation task perform pixel wise classification. The encoder–decoder network is trained using the sum of losses in three different tasks as described in the following Multi-task learning Section.

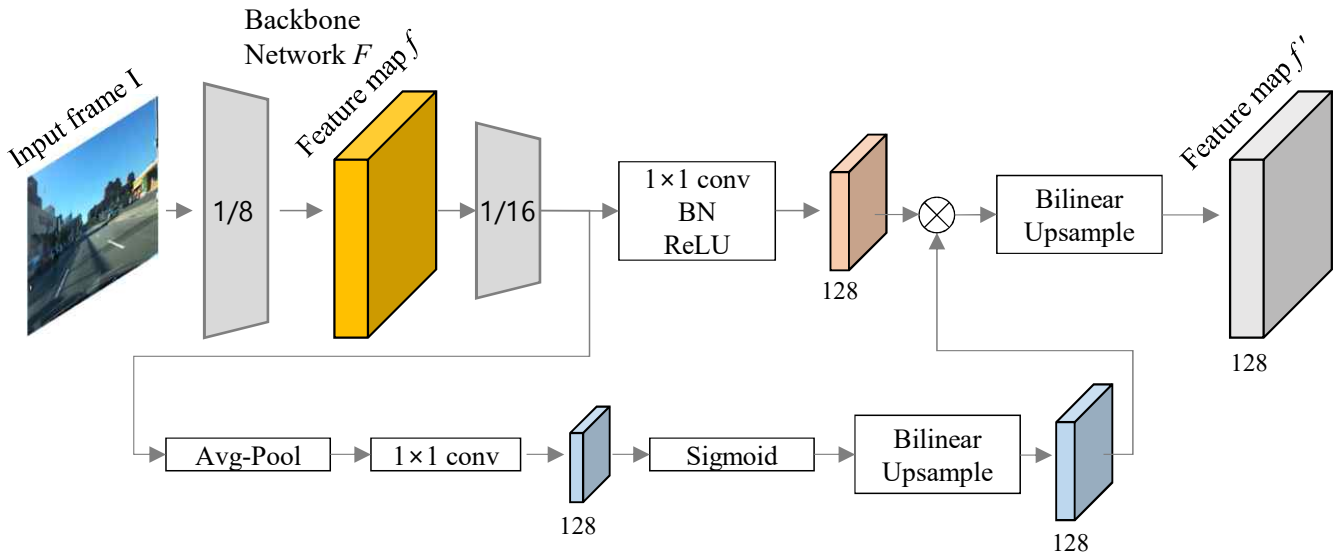

Figure 2. The architecture of encoder for the feature extraction. The light-weight segmentation head is delivered in LR-ASPP module to mix features from multiple resolutions.

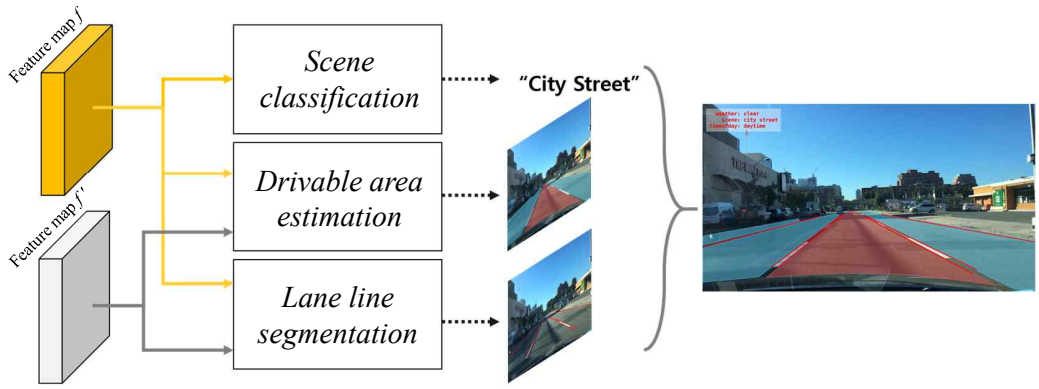

Figure 3. Composition of three decoders for multi-task learning architecture. Each task is trained simultaneously at the decoder using two feature maps having different resolutions. Multi-task learning improves accuracy because losses from different tasks are used to regularize and improve the generalization of another domain.

Table 1. Model structure.

Input	Operator	Exp Size	#out	SE	Stride
$224^2 \times 3$	conv2d	-	16	-	2
$112^2 \times 16$	bneck, 3×3	16	16	-	1
$112^2 \times 16$	bneck, 3×3	64	24	-	2
$56^2 \times 24$	bneck, 3×3	72	24	-	1
$56^2 \times 24$	bneck, 5×5	72	40	✓	2
$28^2 \times 40$	bneck, 5×5	120	40	✓	1
$28^2 \times 40$	bneck, 5×5	120	40	✓	1
$28^2 \times 40$	bneck, 3×3	240	80	-	2
$14^2 \times 80$	bneck, 3×3	200	80	-	1
$14^2 \times 80$	bneck, 3×3	184	80	-	1
$14^2 \times 80$	bneck, 3×3	184	80	-	1
$14^2 \times 80$	bneck, 3×3	480	112	✓	1
$14^2 \times 112$	bneck, 3×3	672	112	✓	1
$14^2 \times 112$	bneck, 5×5	672	160	✓	2
$7^2 \times 160$	bneck, 5×5	960	160	✓	1
$7^2 \times 160$	bneck, 5×5	960	160	✓	1
$7^2 \times 160$	bneck, conv2d, 1×1	-	960	-	1
$7^2 \times 960$	bneck, pool, 7×7	-	-	-	1
$1^2 \times 960$	bneck, conv2d 1×1 NBN	-	1280	-	1
$1^2 \times 1280$	bneck, conv2d 1×1, NBN	-	k	-	1

3.2. Multi-Task Learning

In the proposed architecture, three tasks are processed one image for training and testing, simultaneously. One-encoder and three-decoder structure shares the computation to generate generalized feature maps by learning multiple objectives. Thus, we trained our model using multi-task loss to exploit a comprehensive understanding and computational efficiency. However, because each task has different importance for and impact on the overall performance, we observed a decrease in performance when we train the network with the same weight for each task. To address this issue, we adopt multi-task likelihood loss [5] to simultaneously learn various quantities with different units or scales in both classification and segmentation. Equation (1) expressed the total loss function to train the network, and it is the sum of three tasks.

$$\begin{aligned} Loss(d,l,s) = \frac{1}{3}(&e^{-\eta_d} L_d + \eta_d \\ &+e^{-\eta_l} L_l + \eta_l \\ &+e^{-\eta_s} L_s + \eta_s). \end{aligned} \tag{1}$$

L_d and L_l are semantic segmentation loss functions, and L_s is the classification loss function. The cross-entropy is used to obtain the scene classification loss, L_s. L_d is applied to the drivable area estimation loss, and L_l is applied to the lane line segmentation loss. Simple summation of the three losses with weight is used as the total loss function that helps to learn the entire task reliably. The initial η value of each loss is set to zero.

4. Experiment and Result Analysis

4.1. Dataset

The proposed architecture was evaluated on the Berkeley deep drive dataset [4], which is proposed for autonomous driving research, regarding rich annotation for drivable areas, traffic objects, lane segments, image-level descriptive labels, and the road type for scene classification. Furthermore, this large dataset covers the varying weather, scenes, and times of the day compared to other datasets that have a relatively small number of samples or tasks. This is important to understand the current autonomous driving environment

of vehicles by distinguishing the area of the ego lane and other lanes or backgrounds. There are three classes for the drivable area estimation task: (1) direct; (2) alternative; and (3) background. The ability of the network to classify road types facilitates decisions regarding whether to use the drivable area in different lanes. The other lanes include all pixels that are part of the load, except the ego lane. The BDD dataset consists of 100 K images with a resolution of 1280 × 720. We divided this dataset into three categories: 70 K for the training set, 10 K for the validation set, and 20 K for the testing set, following official standard in the training and testing process as done in the literature [2–4].

4.2. Implementation Details

In this experiment, we trained our system on the training split of the BDD dataset from the scratch. For each image, we first performed image normalization using the value μ = (0.485, 0.456, 0.406) and σ = (0.229, 0.224, 0.225). The model consists of 1.52 M parameters. The input images were resized to 720 × 720 using a random crop technique. We used AdamOptimzer with a learning rate of 0.1×10^{-3} and a minibatch size of 30. All modules were implemented using the Pytorch framework [25]. We used an Intel(R) Xeon Gold 6148 CPU and an NVIDIA GTX Titan Xp to run the experiment.

4.3. Experimental Results

For quantitative comparison with competing methods, we evaluated the performance of the proposed system using two metrics: mean intersection over union (mIoU) for both lane line segmentation and drivable area estimation, and accuracy (Acc.) for the scene classification. The mIoU is defined as the mean of a single intersection over a union per class, $IoU = TP/(FP + TP + FN)$, where TP, FP, and FN denote *true positive*, $false$ *positive*, and $false$ *negative*, respectively, for a single class. The accuracy was calculated using the ratio of the correctly classified road to the total number of test images, $(TP + TN)/(TP + FP + TN + FN)$. We also measured the frame per second (fps), which denotes how many frames can be processed in a second, to show the running time.

The performance comparison with that of the competing drivable area estimation, lane line segmentation, and scene classification methods on the BDD dataset is listed in Table 2. We compared our method with the winners of the Autonomous Driving challenge for drivable area detection [26,27] and others [9,28]. We also compared the lane line segmentation task with [13,14,29], which are single task approaches for lane line detection or segmentation. The comparison with multi-task learning approaches [1–3] are also provided in the table. Even though competing methods shows good results, most of them are not suitable for real-time data processing, except VisLabs [2], SCNN [14], and Asgarian [9]. In particular, IBN_PSA/P is based on IBN-Net [26], and both Mapillary Research [27] and DiDi AI Labs [28] use a modified version of ResNet [16]. MultiNet [1] is a well known multi-task learning approach that performed street classification, vehicle detection and road segmentation on the KITTI dataset. DLT-Net [3] performed drivable area estimation and traffic object detection, simultaneously. To the best of our knowledge, this work is the first real-time multi-task learning approach consists of drivable area estimation, lane line segmentation, and scene classification on the BDD dataset. Here, note that the performances of competing methods were adapted from by [2,13], which might slightly vary as the implementation details are only partially public.

As shown in Table 2, compared to competing multi-task learning approaches [1–3], the proposed methods show improved performance in all three tasks. More accurate drivable area estimation is achieved with a mIoU of 84.41%, and the accuracy of scene classification reaches 78.4%, which is higher than that of the competing method. Compare to other multi-task learning approaches, our approach significantly outperformed competing methods in speed and accuracy. Specifically, the slowest version of our method, which performs three tasks (drivable area estimation, lane line segmentation, and scene classification) simultaneously executes at 93.81 fps. That is more than four times faster than the VisLabs [2]. This is a remarkable advantage for practical applications that require a

short time performance, such as in autonomous vehicles. Compared to the prevision single task drivable area estimation methods [26,27], our method shows a slightly lower mIoU. However, the competing method is not suitable for real-time application with 3.81 fps; the proposed method exhibits more than 20 times faster inference time compared to the competing methods while providing lane line segmentation and scene classification simultaneously. This implies that the proposed method has better computational efficiency, which is also one of the advantages for mobility devices considering the limited resources. We believe that it is competitive results considering that the multi-task approach can provide more rich information that can be utilized in other modules of autonomous driving than a single-task approach.

Table 2. Quantitative evaluation on the BDD dataset. DAE, Lane, Scene, Multi, and Speed refers to the drivable area estimation task, lane line segmentation task, scene classification task, multi-task learning, and running time, respectively.

Method	DAE	Lane	Scene	Multi	Speed
	mIoU(%)	mIoU(%)	Acc.(%)	Yes/No	fps
IBN_PSA/P [26]	86.18	-	-	no	3.81
Mapillary [27]	86.04	-	-	no	0.153
DiDiLabs [28]	84.01	-	-	no	3.35
Asgarian [9]	83.50	-	-	no	322
ENet-SAD [29]	-	47.72	-	no	12.1
SCNN [14]	-	47.34	-	no	123.6
ERFNet [13]	-	51.77	-	no	7.3
DLT-Net [3]	72.10	-	-	yes	9.30
MultiNet [1]	71.60	-	-	yes	8.6
VisLabs (DAE) [2]	83.35	-	-	yes	23.86
VisLabs (Scene) [2]	-	-	77.73	yes	27.58
VisLabs (DAE+Scene) [2]	82.62	-	76.53	yes	22.59
Ours (Lane + Scene)	-	74.46	76.76	yes	96.34
Ours (DAE + Scene)	82.55	-	76.51	yes	106.38
Ours (DAE + Lane)	83.34	75.66	-	yes	104.43
Ours (DAE + Lane + Scene)	84.56	78.57	78.4	yes	93.81

The results of the proposed drivable area estimation are depicted in Figure 4. The drivable areas for both ego-lane and alternative drivable lanes are sufficiently well segmented in a complex environment, for example, highway, city, night, curve, parking lot, single lane. The light reflection can causes errors where light is insufficient at night, but the drivable area is well preserved. The areas in the opposite lane are excluded from the drivable area, and it was successfully differentiated from the opposite driving lane even on a road with sparse lanes. Free space between the cars parked on the road was also well detected. Figure 5 shows the lane line segmentation result. In this experiment, following the ground truth of the BDD dataset, the crosswalk is considered as lane line. Some lane lines are occluded by other objects on the road. However, the result shows that multiple lanes lines are well estimated under various conditions even on a rainy day.

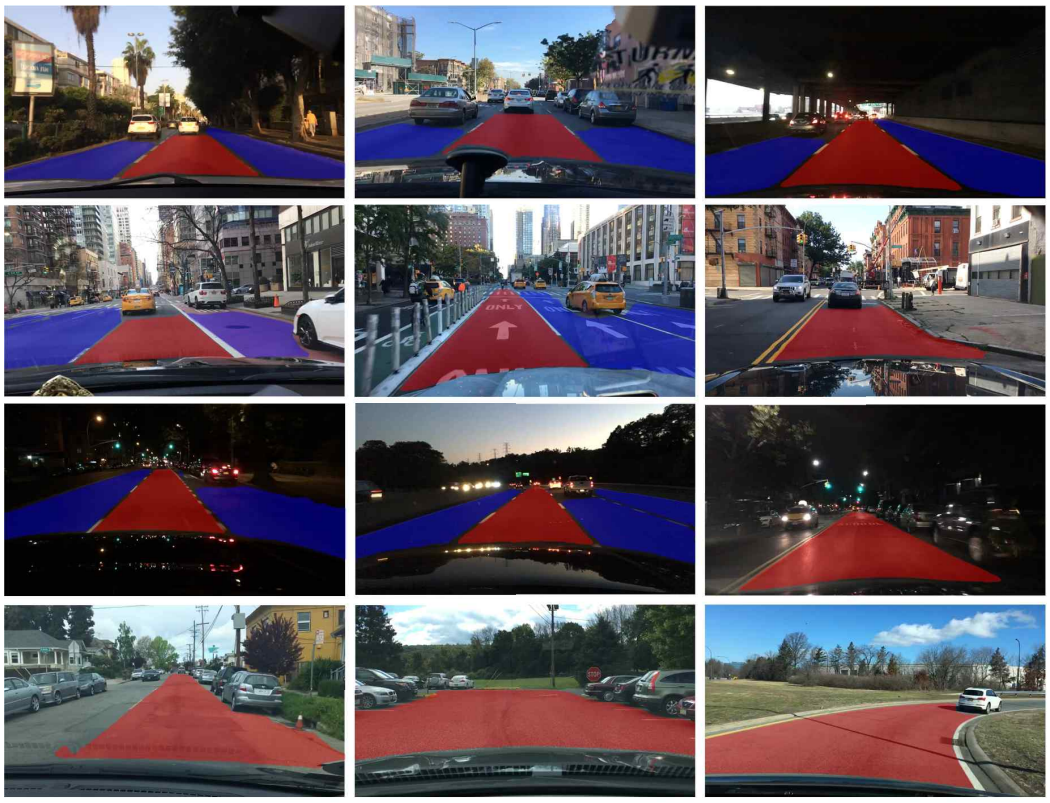

Figure 4. Visualization of drivable area detection result. The red area corresponds to the ego-lane and blue area indicates the alternative drivable area.

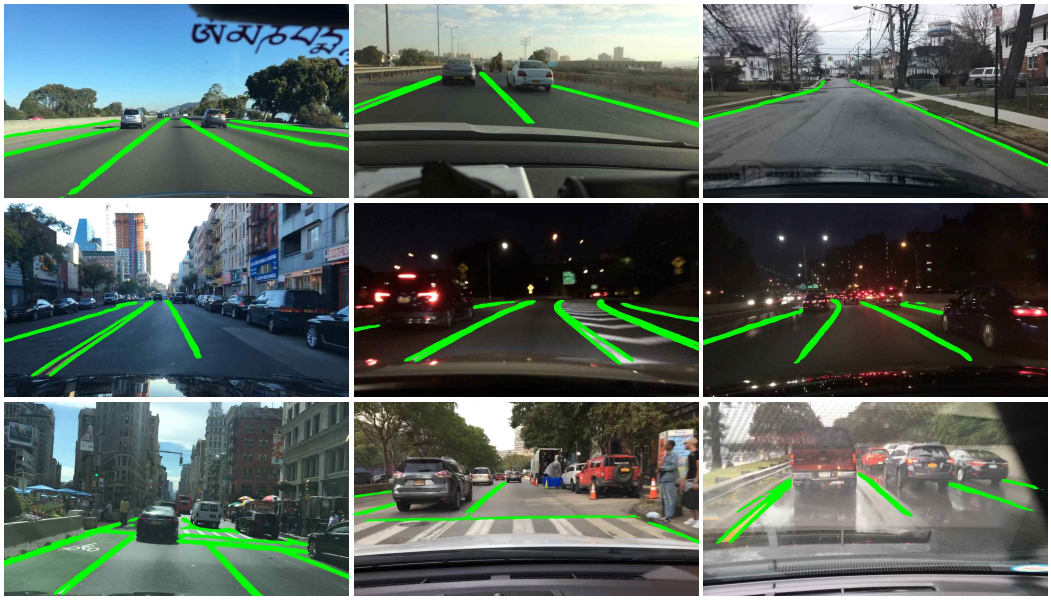

Figure 5. Visualization of lane line segmentation result.

5. Conclusions

In this study, we developed a fast and accurate unified end-to-end architecture that can simultaneously estimate drivable area, segment lane lines and classify scenes in real-time. Our simple encoder–decoder structure can learn the most important three tasks for autonomous driving, simultaneously. The decoder successfully predicts the drivable area with the ego-lane and alternative lanes. The experimental results demonstrate that our approach achieves the fastest inference speed, surpassing that of all competing methods, while maintaining competitive performances for the drivable area estimation in a very

challenging BDD dataset. Furthermore, compared to the multi-task learning approaches, the proposed method outperforms the competing method in both accuracy and speed. The computational efficiency of the proposed architecture is notable, which is the most promising advantage for practical autonomous driving. However, performance improvement is still required since the drivable area estimation is a safety-critical application. Avoiding collision is an important function for the passenger's safety. Thus, we expect that the multi-task learning architecture detecting obstacles on the road will be our future research.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korean Government (MSIT) (No.2021R1C1C1012590) and the Information Technology Research Center (ITRC) support program supervised by the Institute of Information Communications & Technology Planning & Evaluation (IITP) grant funded by the Korean Government (MSIT) (IITP-2021-2020-0-01808).

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

References

1. Teichmann, M.; Weber, M.; Zoellner, M.; Cipolla, R.; Urtasun, R. Multinet: Real-time joint semantic reasoning for autonomous driving. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 1013–1020.
2. Pizzati, F.; García, F. Enhanced free space detection in multiple lanes based on single CNN with scene identification. In Proceedings of the 2019 IEEE Intelligent Vehicles Symposium (IV), Paris, France, 9–12 June 2019; pp. 2536–2541.
3. Qian, Y.; Dolan, J.M.; Yang, M. DLT-Net: Joint detection of drivable areas, lane lines, and traffic objects. *IEEE Trans. Intell. Transp. Syst. (IVS)* **2019**, *21*, 4670–4679. [CrossRef]
4. Yu, F.; Xian, W.; Chen, Y.; Liu, F.; Liao, M.; Madhavan, V.; Darrell, T. Bdd100k: A diverse driving video database with scalable annotation tooling. *arXiv* **2018**, arXiv:1805.04687.
5. Kendall, A.; Gal, Y.; Cipolla, R. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 7482–7491.
6. Chen, Z.; Chen, Z. Rbnet: A deep neural network for unified road and road boundary detection. In *International Conference on Neural Information Processing*; Springer: Berlin, Germany, 2017; pp. 677–687.
7. Han, X.; Lu, J.; Zhao, C.; You, S.; Li, H. Semisupervised and weakly supervised road detection based on generative adversarial networks. *IEEE Signal Process. Lett.* **2018**, *25*, 551–555. [CrossRef]
8. Munoz-Bulnes, J.; Fernandez, C.; Parra, I.; Fernández-Llorca, D.; Sotelo, M.A. Deep fully convolutional networks with random data augmentation for enhanced generalization in road detection. In Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC), Yokohama, Japan, 16–19 October 2017; pp. 366–371.
9. Asgarian, H.; Amirkhani, A.; Shokouhi, S.B. Fast Drivable Area Detection for Autonomous Driving with Deep Learning. In Proceedings of the 2021 5th International Conference on Pattern Recognition and Image Analysis (ICPRIA), Kashan, Iran, 28–29 April 2021; pp. 1–6.
10. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder–decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI)* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]
11. Romera, E.; Alvarez, J.M.; Bergasa, L.M.; Arroyo, R. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Trans. Intell. Transp. Syst. (ITS)* **2017**, *19*, 263–272. [CrossRef]
12. Neven, D.; De Brabandere, B.; Georgoulis, S.; Proesmans, M.; Van Gool, L. Towards end-to-end lane detection: An instance segmentation approach. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 26–30 June 2018; pp. 286–291.
13. Lee, M.; Lee, J.; Lee, D.; Kim, W.; Hwang, S.; Lee, S. Robust lane detection via expanded self attention. *arXiv* **2021**, arXiv:2102.07037.
14. Pan, X.; Shi, J.; Luo, P.; Wang, X.; Tang, X. Spatial as deep: Spatial cnn for traffic scene understanding. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, New Orleans, LA, USA, 2–7 February 2018.
15. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst. (NeurIPS)* **2012**, *25*, 1097–1105. [CrossRef]
16. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.

17. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L.C. Mobilenetv2: Inverted residuals and linear bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
18. Howard, A.; Sandler, M.; Chu, G.; Chen, L.C.; Chen, B.; Tan, M.; Wang, W.; Zhu, Y.; Pang, R.; Vasudevan, V.; et al. Searching for mobilenetv3. In Proceedings of the IEEE International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019; pp. 1314–1324.
19. Liu, X.; Gao, J.; He, X.; Deng, L.; Duh, K.; Wang, Y.Y. Representation Learning Using Multi-Task Deep Neural Networks for Semantic Classification and Information Retrieval. In Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Denver, CO, USA, 31 May–5 June 2015.
20. Hariharan, B.; Arbeláez, P.; Girshick, R.; Malik, J. Simultaneous detection and segmentation. In *European Conference on Computer Vision (ECCV)*; Springer: Berlin, Germany, 2014; pp. 297–312.
21. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
22. Dvornik, N.; Shmelkov, K.; Mairal, J.; Schmid, C. Blitznet: A real-time deep network for scene understanding. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 4154–4162.
23. Yang, T.J.; Howard, A.; Chen, B.; Zhang, X.; Go, A.; Sandler, M.; Sze, V.; Adam, H. Netadapt: Platform-aware neural network adaptation for mobile applications. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 285–300.
24. Hu, J.; Shen, L.; Sun, G. Squeeze-and-excitation networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 7132–7141.
25. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. PyTorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*; Wallach, H., Larochelle, H., Beygelzimer, A., d'Alché-Buc, F., Fox, E., Garnett, R., Eds.; Curran Associates, Inc.: Dutchess County, NY, USA, 2019; pp. 8024–8035.
26. Pan, X.; Luo, P.; Shi, J.; Tang, X. Two at once: Enhancing learning and generalization capacities via ibn-net. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 464–479.
27. Zhao, H.; Zhang, Y.; Liu, S.; Shi, J.; Loy, C.C.; Lin, D.; Jia, J. Psanet: Point-wise spatial attention network for scene parsing. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 267–283.
28. Bulo, S.R.; Porzi, L.; Kontschieder, P. In-place activated batchnorm for memory-optimized training of dnns. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Salt Lake City, UT, USA, 18–22 June 2018; pp. 5639–5647.
29. Hou, Y.; Ma, Z.; Liu, C.; Loy, C.C. Learning lightweight lane detection cnns by self attention distillation. In Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea, 27–28 October 2019; pp. 1013–1021.