Simultaneous Road Edge and Road Surface Markings Detection Using Convolutional Neural Networks

René Pihlak(✉) and Andri Riid

Tallinn University of Technology, Ehitajate tee 5, 19086 Tallinn, Estonia
{rene.pihlak,andri.riid}@taltech.ee

Abstract. Accurate road surface markings and road edges detection is a crucial task for operating self-driving cars and for advanced driver assistance systems deployment (e.g. lane detection) in general. This research proposes an original neural network based method that combines structural components of autoencoders, residual neural networks and densely connected neural networks. The resulting neural network is able to concurrently detect and segment accurate road edges and road surface markings from RGB images of road surfaces.

Keywords: Road surface markings · Road edge · Neural networks · Segmentation · Machine learning · Autoencoder

1 Introduction

Road surface markings (RSMs) and road edges (REs) are among the most important elements for guiding autonomous vehicles (AVs). High-Definition maps with accurate RSM information are very useful for many applications such as navigation, prediction of upcoming road situations and road maintenance [23]. RSM and RE detection is also vital in the context of pavement distress detection to ensure that RSMs are not confused with pavement defects and to eliminate the areas where the cracks, potholes and other defects cannot possibly be found [17].

The detection of stationary objects of interest related to roadways is usually addressed by using video streams or still images acquired by digital cameras mounted on a vehicle. For example, Reach-U Ltd.—Estonian company specializing in geographic information systems, location based solutions and cartography—has developed a fast-speed mobile mapping system employing six high-resolution cameras for recording images of roads. The resulting orthoframes assembled from recorded panoramic images are available to Estonian Road

This study was partially supported by the Archimedes Foundation and Reach-U Ltd. in the scope of the smart specialization research and development project #LEP19022: "Applied research for creating a cost-effective interchangeable 3D spatial data infrastructure with survey-grade accuracy".

T. Robal et al. (Eds.): DB&IS 2020, CCIS 1243, pp. 109–121, 2020.
https://doi.org/10.1007/978-3-030-57672-1_9

Administration via a web application called EyeVi [16] and are used as the source data in present study.

RSM extraction methods are often based on intensity thresholds that are subject to data quality, including lighting, shadows and RSM wear. RE detection is generally considered even more difficult than lane mark detection due to lack of clear boundary between road and RE because of variations in road and roadside materials and colors [9]. Due to recent advances in deep learning, semantic segmentation of RSMs and REs is increasingly implemented within the deep learning framework [21].

The present study combines both RSM and RE detection by a hierarchical convolutional neural network to enhance the method's predictive ability. On narrow rural roads without RSMs, RE detection alone can provide sufficient information for navigation but REs might escape the field of view of cameras on wider highways and REs cannot provide sufficient information for lane selection. Simultaneous detection of REs and RSMs can also improve the accuracy of these tasks, e.g. RSMs suggested by the network outside the road can be eliminated either implicitly within the network or explicitly during the post-processing.

2 Related Work

In recent years, several studies have tackled tasks of RSM and RE detection for a variety of reasons. For example, [3,19] employ RSMs for car localization, [1] applies RSMs to detect REs, [2,4,6,26] classify RSMs whereas [7,9,12,20,21,27] focus on pixel level segmentation to detect the exact shapes and locations of RSMs.

These works have used different input data. For example, [12,22,24] use 3D data collected by LiDAR, [15,25] use 3D spacial data generated from images captured by stereo camera, [11] uses radar for detecting metal guardrails and [9,21,26] use 2D still images captured by a camera. Radars have an advantage over cameras as they can be deployed regardless of lighting conditions. However, metal guardrails are rarely found on smaller rural roads. LiDARs can also operate under unfavorable lighting conditions that can cause over- or under-lit road surface, however, high quality LiDARs are expensive. Both 2D cameras and 3D (stereo) cameras are adversely affected by shadows, over- and under-lit road surface and while 3D cameras can provide additional information, this information requires more processing power. Consequently, 2D cameras are still the most suitable equipment for capturing data for RSMs and RE detection provided that the detection method is "smart" enough to cope with even the most adverse lighting condition such as hard shadows, over- and under-lit surface of the road, etc. Thus the RSM and RE detection methods must address these issues.

Possible RSMs and RE detection methods and techniques include, for example, filtering [10] and (NN) based detection [26]. While filtering was the method of choice in older research, the NN based methods have started to dominate recent research. Even though the NN based methods have gained popularity, the way NNs are implemented for RSM and RE detection has evolved significantly over the years. Earlier approaches used classifier networks. The segmentation of images was achieved by applying a sliding window classifier over the whole input image. To improve quality of the predictions, fully convolutional (AEs) that can produce a segmented output image directly, have replaced the sliding window method. However, since the 'encoding' part of the AE is subject to data loss, the AEs have been phased out by residual networks (e.g., [5,8]) with direct forward connections between layers of *encoder* and *decoder* for higher quality image segmentation.

3 Methodology

3.1 Neural Network Design

The proposed architecture consists of three connected neural networks of identical structure (Fig. 1) that combines the architectures of symmetrical AE (with dimension reducing *encoder* and expanding *decoder*), (ResNet) [5] and DenseNet [8] having shortcut connections between *encoder* and *decoder* layers by feature map concatenation.

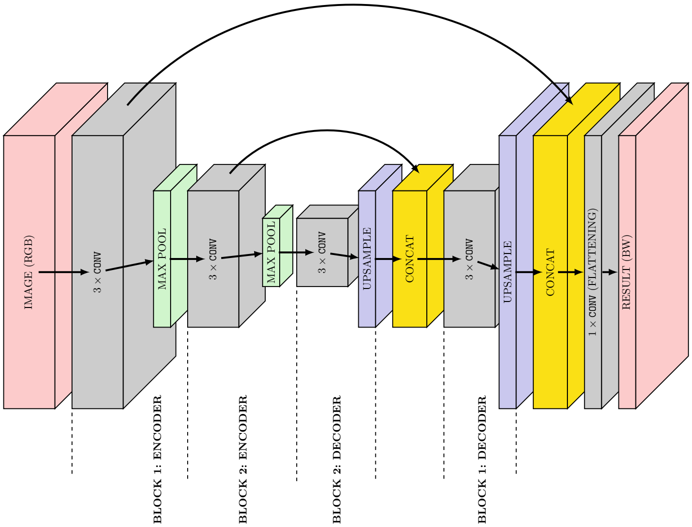

Fig. 1. Illustration of sub-neural network design using 2 blocks with 3 convolutional filters

This NN can be further divided into sub-blocks (hereafter *blocks*) that all have a similar structure. In essence, a block consists of a set of convolutional layers followed by a size transformation layer (either `max pooling` for *encoder* or `upsampling` for *decoder*). As the NN is symmetric, *encoder* and *decoder* have equal number of blocks. Thus a NN with "two blocks" would be a NN that has two encoding and two decoding blocks. Each of these blocks has an equal number of convolutional layers with 3 × 3 kernel. After each decoding block, there is a concatenation layer in order to fuse the upsampled feature map with the corresponding convolutional layer's feature map from the *encoder* part of the network.

The overall design of the sub-NN is as follows:

1. Input layer (RGB image)
2. $n\times$ encoding block
3. $n\times$ decoding block + concatenation layer
4. flattening convolutional layer
5. Output layer (gray scale image).

All the convolutional layers (except the last one) use LeakyReLU ($\alpha = 0.1$) as activation function. It is observed that a large gradient flowing through a more common ReLU neuron can cause the weights to update in such a way that the neuron will never activate on any datapoint again. Once a ReLU ends up in this state, it is unlikely to recover. LeakyReLU returns a small value when its input is less than zero and gives it a chance to recover [13]. The convolutional layers' padding is 'same', i.e., the height and width of input and output feature maps are the same for each convolutional layer within a block.

The U-Net architecture proposed in [18] utilizes a similar idea, however, the order of layers is different in the *decoder* part of their NN:

- $\boxed{\texttt{upsample}} \rightarrow \boxed{\texttt{concatenate}} \rightarrow \boxed{\texttt{convolution}}$ (U-Net)
- $\boxed{\texttt{convolution}} \rightarrow \boxed{\texttt{upsample}} \rightarrow \boxed{\texttt{concatenate}}$ (Proposed)

The number of computations required by a convolutional layer increases as the size of inputs (feature maps from the previous layer) increases. The size of layer's inputs is given by Eq. 1, where *width* and *height* are width and height of the feature map from the previous layer and *depth* is the number of convolutional filters in the previous (convolutional) layer.

$$V = width \times height \times depth \tag{1}$$

Because U-Net performs upsampling and concatenation before convolutional layer, the input size (hence the required computational power) is more than four times bigger (Eq. 2) than with the proposed NN. Upsampling with kernel size (2 × 2) doubles both height and width and concatenation further increases depth by adding $depth_{enc}$ layers from an encoder block to the feature map.

$$\begin{aligned} V_{\texttt{UNet}} &= width_{\texttt{UNet}} \times height_{\texttt{UNet}} \times depth_{\texttt{UNet}} \\ &= (2 \times width) \times (2 \times height) \times (depth + depth_{enc}) \end{aligned} \tag{2}$$

Since RSMs and REs have different characteristics, these two tasks are performed using two different NNs. Both NNs have a RGB input of size (224 × 224 × 13) and a single (224 × 13) output, i.e., a gray scale mask (Fig. 1). One pixel corresponds roughly to 3.37 mm, thus the width and height of the input segment are about 75.42 cm.

The resulting outputs from RE and RSMs detection are concatenated with the original input image and fed to a third NN (REFNET) to refine the final results (Fig. 2). Hence, the final structure of the whole NN combines three NNs: two parallel NNs (RENN and RSMNN) and a REFNET. REFNET has the same general design as the RENN and RSMNN but the number of blocks and number of convolutional filters per block differ (Table 1). It must be noted that adding REFNET incurs relatively small overhead because it has considerably smaller number of trainable parameters compered to RENN and RSMNN even though REFNET has higher block count. Table 1 describes the chosen architecture that had the best performance.

Table 1. Neural network parameters

Value	RENN	RSMNN	REFNET
Trainable parameters	148489	208201	66058
Block count	3	4	5
Conv layers per block	6	2	2
Number of filters	(8,16,32)	(8,16,32,64)	(8,8,16,16,32)

Both RENN and RSMNN are pre-trained for 20 epochs. Several combinations of blocks and layers were tested and best performing RENN and RSMNN were chosen. Next, all three NNs – RENN, RSMNN and REFNET – are trained for additional 60 epochs. The NNs were trained using Adadelta optimizer with initial learning rate of 1.0 and binary crossentropy as loss function. Automatic learning rate reduction was applied if validation loss did not improve in 15 epochs. Each reduction halved the current learning rate.

3.2 Evaluation Methodology

There are several metrics for evaluating segmentation quality. The most common of those is accuracy (Eq. 3) that calculates the percent of pixels in the image that are classified correctly

$$A_{cc} = \frac{TP + TN}{TP + FP + TN + FN}, \tag{3}$$

where TP = True Positives (pixels correctly predicted to belong to the given class), TN = True Negatives (pixels correctly predicted not to belong to the given class), FP = False Positives (pixels falsely predicted to belong to the given class), and FN = False Negatives (pixels falsely predicted not to belong to the given class). In cases where there is a class imbalance in the evaluation data,

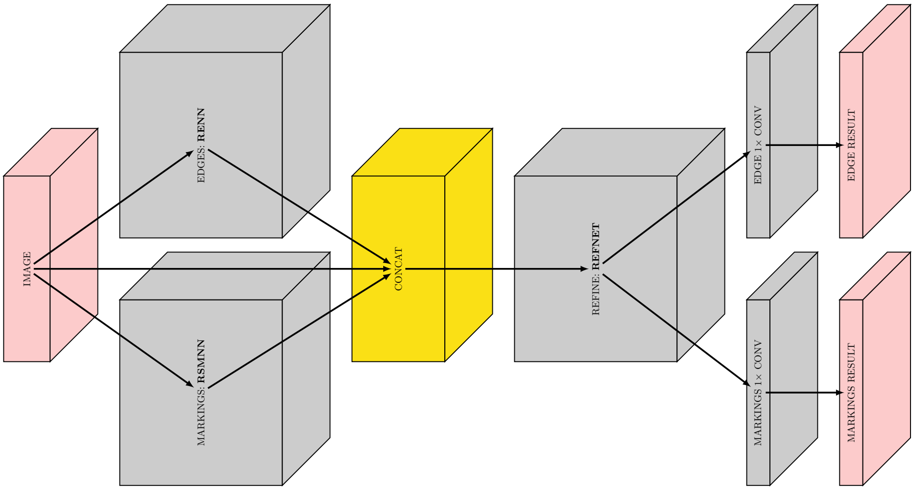

Fig. 2. Overall structure of the final neural network

e.g. TN ≫ TP, FP, FN (which is often the case in practice), accuracy is biased toward an overoptimistic estimate.

For class-imbalanced problems, precision (Eq. 4) and recall (Eq. 5) (showing how many of predicted positives were actually true positives and how many of true positives were predicted as positives, respectively)

$$P_r = \frac{TP}{TP + FP}, \tag{4}$$

$$R_c = \frac{TP}{TP + FN}, \tag{5}$$

or balanced accuracy that is an average of recall (also known a sensitivity or True Positive Rate) and specificity (also known as True Negative Rate) (Eq. 6)

$$Acc_{bal} = \frac{TPR + TNR}{2} = \left(\frac{TP}{TP + FN} + \frac{TN}{TN + FP}\right) \div 2, \tag{6}$$

give a better estimate

In addition, Jaccard similarity coefficient or Intersection over Union (IoU) (Eq. 7) can be used

$$IoU = \frac{|X \bigcap Y|}{|X \bigcup Y|} = \frac{TP}{TP + FP + FN}. \tag{7}$$

IoU is the area of overlap between the predicted segmentation (X) and the ground truth (Y) divided by the area of union between the predicted segmentation and the ground truth.

For binary or multi-class segmentation problems, the mean IoU is calculated by taking the IoU of each class and averaging them.

$$mIoU = n^{-1} \sum_{i=1}^{n} IoU_i \tag{8}$$

The same applies for the other metrics (Eq. 6, Eq. 4, Eq. 5).

4 Experimental Results

4.1 Setup of the Experiments

The proposed methods are evaluated on a dataset that contains a collection of 314 orthoframe images of Estonian roads that each have a size of 4096 × 4096 pixels.

The images in the dataset are manually annotated to generate ground truth masks for both RSMs and RE. The annotation is performed by using Computer Vision Annotation Tool (CVAT) (Fig. 3). These vector graphics annotations are then converted into image masks with separate masks for RSM and RE.

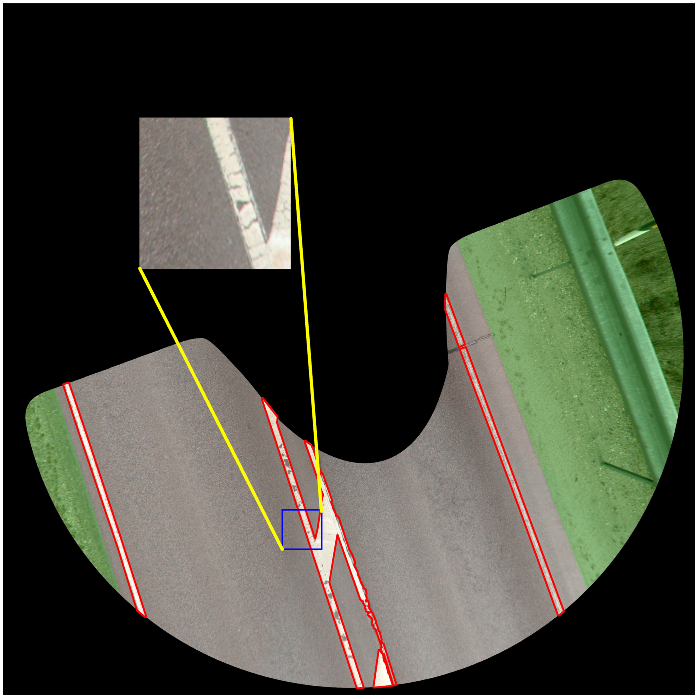

Fig. 3. Image annotation. The annotated RSMs are outlined by red color and the area outside the road masked out by the annotation mask is painted green for illustration. (Color figure online)

As a preliminary step before training and testing the proposed NN, automatic image pre-processing (i.e. white balancing) is applied to each individual input image by using gray world assumption with saturation threshold of (0.95). The goal of image pre-processing is to mitigate color variance (e.g. due to color difference of sun light).

Next, 249 of 314 pre-processed images are used to build the training/validation dataset of image segments. This training/validation dataset contains 36497 image segments and each segment has a size of 224 × 224 pixels (Fig. 3). It must be noted that not all segments of a pre-processed image are included in the training/validation dataset. Since large part of the 4096 × 4096 pixel image is masked out (depicted by black color in Fig. 3), only segments with 20% or more non-masked pixels are considered. These segments are then divided into two groups: 1) those that include annotations and 2) those that do not include annotations. A segment is considered to be annotated if and only if at least 5% of its pixels contain annotations. In order to prevent significant class imbalance in training data, the two groups (i.e., segments with and without annotations) have to be of equal size. The segments in the two groups do not overlap.

Built-in image pre-processing methods of TensorFlow/Keras library are randomly applied on each 224 × 224 × 3 segment before training in order to perform data augmentation. These image pre-processing methods include horizontal flip, vertical flip, width shift (±5%), height shift (±5%) and rotation (<90°).

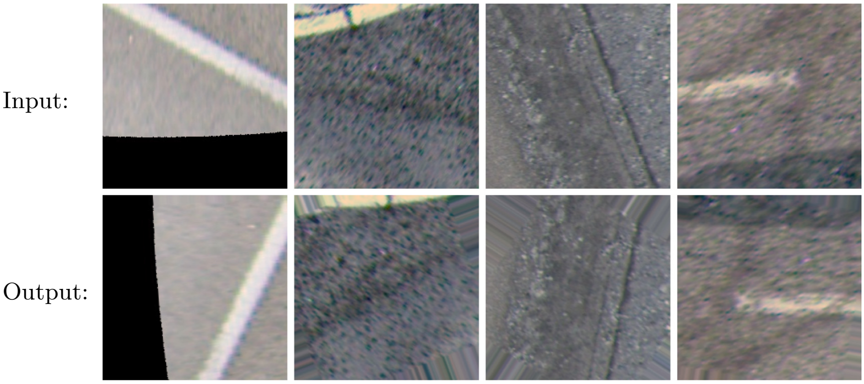

Fig. 4. Augmentation of input data

Rotation and/or shifting without zooming in can lead to situations where part of the modified segment does not include the pixels from the original segment. Therefore, these pre-processing methods are performed using 'nearest' fill method (Fig. 4).

4.2 Image Post-processing and Testing Results

Testing is performed on 65 images which were not included in training or validation data sets. The images are turned into 224 × 224 × 3 RGB image segments similarly to the process described in Sect. 4.1. The main difference is that after pre-processing (i.e., white balance), the testable segments are generated from all segments that are not 100% black. These testable segments are then passed to the NNs and resulting output of the NNs is re-combined such that input segments that were 100% black are also fully black in the combined 4096 × 4096 output image. This process is repeated on each test image three times. Each time a different amount of padding is used. These three outputs are cropped to original size and averaged to assure that final generated masks/images are smooth.

Before the final evaluation, the re-combined outputs undergo image post-processing. First, thresholding using Otsu's method [14] is applied on both RE and RSM images to produce binary images (Fig. 5). Next, holes are filled in the RE image. Then RE contours are detected and small objects are filtered out based on their area. Next, the refined RE image is used as a mask to rule out false positive RSMs. Finally, RSM contours are detected and small objects are filtered out based on the object's area (Fig. 6)

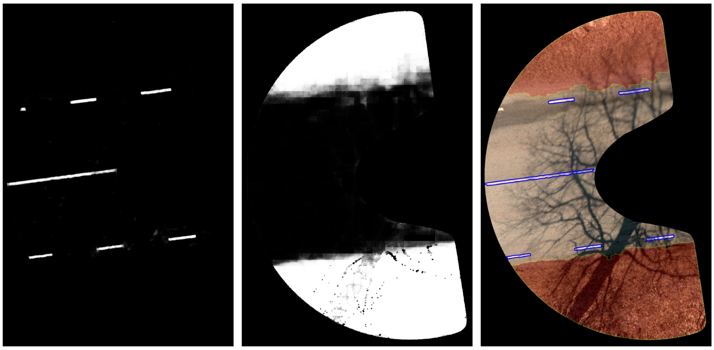

Fig. 5. Road surface markings and road edges masks before applying Otsu's method (left and middle, respectively) and the combined result after post-processing (right).

The number of pixels, classified as true positives (TP), true negatives (TN), false positives (FP) and false negatives (FN) are determined over all test images (Table 2). Based on the resulting figures, recall (R_c), precision (P_r), balanced accuracy (ACC_{bal}) and IoU are calculated and given in Table 3. The figures in Table 3 imply that RE detection has somewhat better quality than RSMs detection. This is, however, deceptive, because of the smaller overall size of RSMs (TP + FN in Table 2), minor detection imperfections result in a greater degradation of performance indices.

In contrast to performance measures over all test images in Table 3, Table 4 provides performance analysis on orthoframe basis. First, performance measures were calculated for each image separately. Next, statistical measures such as average, minimum, maximum value and standard deviation (σ) were calculated. These additional characteristics show that RE detection accuracy is in fact less consistent and more sensitive to hard shadows in particular as Fig. 7 demonstrates. Standard deviation (σ) for RSM detection is lower in all categories compared to RE detection.

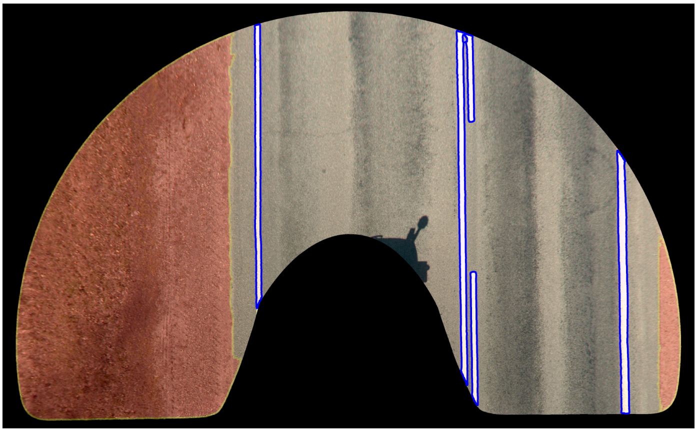

Fig. 6. Near-perfect detection of road surface markings (outlined by blue color) and road edges (pale red). (Color figure online)

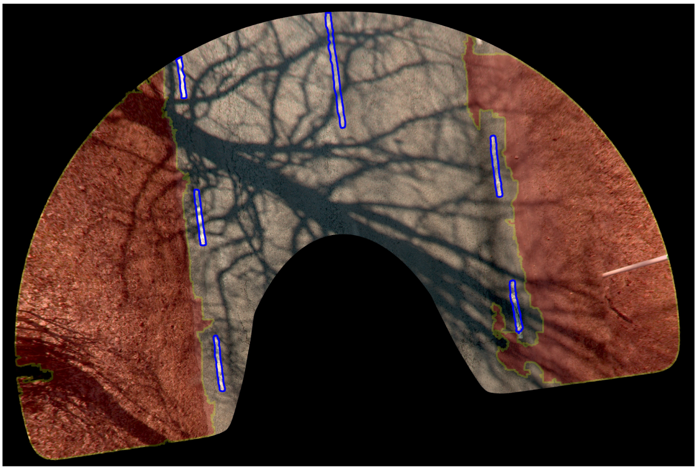

Fig. 7. Detection of road surface markings (outlined by blue color) and road edges (pale red) in presence of hard shadows. (Color figure online)

Table 2. Pixel classification after post-processing

Type	TP	TN	FP	FN
Edges	159647475	915666344	8170998	7034223
Markings	7501060	1081635670	581563	800747

Table 3. Detection performance indices after post-processing

Type	R_c	P_r	Acc_{bal}	IoU
Edges	0.96	0.95	0.97	0.88
Markings	0.90	0.93	0.95	0.84

Table 4. Detailed measures of per image results

Value	Edges				Markings			
	R_c	P_r	TNR	Acc_{bal}	R_c	P_r	TNR	Acc_{bal}
Avg	0.92	0.95	0.99	0.96	0.90	0.93	1.00	0.95
σ	0.18	0.08	0.02	0.09	0.06	0.06	0.00	0.03
Min	0.13	0.42	0.88	0.57	0.64	0.71	1.00	0.82
Max	1.00	0.99	1.00	1.00	0.98	1.00	1.00	0.99

This was expected because RE detection is considered to be harder problem to solve. The detection of RENN did not improve when NN size (block and/or convolutional layer count) were increased. However, adding contextual information (RSMNN) increased the proposed method's predictive ability for both RE and RSM detection when the outputs of RENN and RSMNN were combined in REFNET.

5 Conclusion

In this study, we proposed a convolutional neural network-powered method for concurrent road edge (RE) and road surface markings (RSMs) detection from orthoframes even under severely adverse conditions such as bad lighting conditions (shadows, over-lit road surface, etc) and RSM wear. The measures of IoU (88% and 84%, respectively) indicate that the network performs well in most conditions, however, hard shadows, cast either by trees or buildings alongside the road, present a problem, particularly for RE detection.

In future works, we intend to research how to improve the predictive ability of the method by using increased contextual awareness either by incorporating data about the neighboring image segments or by using the whole orthoframe image as input.

References

1. Álvarez, J.M., López, A.M., Gevers, T., Lumbreras, F.: Combining priors, appearance, and context for road detection. IEEE Trans. Intell. Transp. Syst. **15**(3), 1168–1178 (2014). https://doi.org/10.1109/TITS.2013.2295427
2. De Paula, M.B., Jung, C.R.: Automatic detection and classification of road lane markings using onboard vehicular cameras. IEEE Trans. Intell. Transp. Syst. **16**(6), 3160–3169 (2015). https://doi.org/10.1109/TITS.2015.2438714
3. Deng, L., Yang, M., Hu, B., Li, T., Li, H., Wang, C.: Semantic segmentation-based lane-level localization using around view monitoring system. IEEE Sens. J. **19**(21), 10077–10086 (2019). https://doi.org/10.1109/JSEN.2019.2929135
4. Gupta, A., Choudhary, A.: A framework for camera-based real-time lane and road surface marking detection and recognition. IEEE Trans. Intell. Veh. **3**(4), 476–485 (2018). https://doi.org/10.1109/tiv.2018.2873902
5. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778 (2015)
6. Hoang, T.M., Nam, S.H., Park, K.R.: Enhanced detection and recognition of road markings based on adaptive region of interest and deep learning. IEEE Access **7**, 109817–109832 (2019). https://doi.org/10.1109/access.2019.2933598
7. Hu, J., Yang, M., Xu, H., He, Y., Wang, C.: Mapping and localization using semantic road marking with centimeter-level accuracy in indoor parking lots. In: 2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019, pp. 4068–4073. Institute of Electrical and Electronics Engineers Inc. (October 2019). https://doi.org/10.1109/ITSC.2019.8917529
8. Huang, G., Liu, Z., Weinberger, K.Q.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2261–2269 (2016)
9. Jiang, W., Wu, Y., Guan, L., Zhao, J.: DFNet: semantic segmentation on panoramic images with dynamic loss weights and residual fusion block. In: Proceedings - IEEE International Conference on Robotics and Automation, vol. 2019, pp. 5887–5892. Institute of Electrical and Electronics Engineers Inc. (May 2019). https://doi.org/10.1109/ICRA.2019.8794476
10. Kim, Z.W.: Robust lane detection and tracking in challenging scenarios (2008). https://doi.org/10.1109/TITS.2007.908582
11. Lin, J., Chien, S., Chen, Y., Chen, C.C., Sherony, R.: 24 GHz and 77 GHz radar characteristics of metal guardrail for the development of metal guardrail surrogate for road departure mitigation system testing. In: 2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019, pp. 3340–3346. Institute of Electrical and Electronics Engineers Inc. (October 2019). https://doi.org/10.1109/ITSC.2019.8916960
12. Ma, L., Li, Y., Li, J., Zhong, Z., Chapman, M.A.: Generation of horizontally curved driving lines in HD maps using mobile laser scanning point clouds. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **12**(5), 1572–1586 (2019). https://doi.org/10.1109/JSTARS.2019.2904514
13. Maas, A.L., Hannun, A.Y., Ng, A.Y.: Rectifier nonlinearities improve neural network acoustic models. In: ICML Workshop on Deep Learning for Audio, Speech and Language Processing (2013)
14. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. Syst. Man Cybern. **9**(1), 62–66 (1979). https://doi.org/10.1109/TSMC.1979.4310076

15. Ozgunalp, U., Fan, R., Ai, X., Dahnoun, N.: Multiple lane detection algorithm based on novel dense vanishing point estimation. IEEE Trans. Intell. Transp. Syst. **18**(3), 621–632 (2017). https://doi.org/10.1109/TITS.2016.2586187
16. Reach-U Ltd.: Eyevi – mobile mapping based visual intelligence. https://www.reach-u.com/eyevi.html. Accessed 12 Feb 2020
17. Riid, A., Lõuk, R., Pihlak, R., Tepljakov, A., Vassiljeva, K.: Pavement distress detection with deep learning using the orthoframes acquired by a mobile mapping system. Appl. Sci. **9**(22), 4829 (2019)
18. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. ArXiv (May 2015)
19. Rose, C., Britt, J., Allen, J., Bevly, D.: An integrated vehicle navigation system utilizing lane-detection and lateral position estimation systems in difficult environments for GPS. IEEE Trans. Intell. Transp. Syst. **15**(6), 2615–2629 (2014). https://doi.org/10.1109/TITS.2014.2321108
20. Suleymanov, T., Kunze, L., Newman, P.: Online inference and detection of curbs in partially occluded scenes with sparse LIDAR. In: 2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019, pp. 2693–2700. Institute of Electrical and Electronics Engineers Inc. (October 2019). https://doi.org/10.1109/ITSC.2019.8917086
21. Tran, L.A., Le, M.H.: robust U-Net-based road lane markings detection for autonomous driving. In: Proceedings of 2019 International Conference on System Science and Engineering, ICSSE 2019, pp. 62–66. Institute of Electrical and Electronics Engineers Inc. (July 2019). https://doi.org/10.1109/ICSSE.2019.8823532
22. Uzer, F., Benmokhtar, R., Moujtahid, S., Perrotton, X.: Dempster shafer grid-based hybrid fusion of virtual lanes for autonomous driving. In: 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3760–3765. IEEE (November 2019). https://doi.org/10.1109/IROS40897.2019.8967610, https://ieeexplore.ieee.org/document/8967610/
23. Wen, C., Sun, X., Li, J., Wang, C., Guo, Y., Habib, A.: A deep learning framework for road marking extraction, classification and completion from mobile laser scanning point clouds. ISPRS J. Photogramm. Remote Sens. **147**, 178–192 (2019). https://doi.org/10.1016/j.isprsjprs.2018.10.007
24. Yu, Y., Li, J., Guan, H., Jia, F., Wang, C.: Learning hierarchical features for automated extraction of road markings from 3-D mobile LiDAR point clouds. IEEE J. Sel. Top. Appl. Earth Obs. Remote Sens. **8**(2), 709–726 (2015). https://doi.org/10.1109/JSTARS.2014.2347276
25. Yuan, C., Chen, H., Liu, J., Zhu, D., Xu, Y.: Robust lane detection for complicated road environment based on normal map. IEEE Access **6**, 49679–49689 (2018). https://doi.org/10.1109/ACCESS.2018.2868976
26. Zhang, F., Wu, X., Gu, C.: Detection of road surface identifiers based on deep learning. In: 2019 International Conference on Artificial Intelligence and Advanced Manufacturing (AIAM), pp. 66–70. Institute of Electrical and Electronics Engineers (IEEE) (January 2020). https://doi.org/10.1109/aiam48774.2019.00020
27. Zhang, W., Mi, Z., Zheng, Y., Gao, Q., Li, W.: Road marking segmentation based on siamese attention module and maximum stable external region. IEEE Access **7**, 143710–143720 (2019). https://doi.org/10.1109/ACCESS.2019.2944993