Proceedings of the 26th International Conference on Automation & Computing, University of Portsmouth, Portsmouth, UK, 2-4 September 2021

Deep Ensembles for Semantic Segmentation on Road Detection

Deniz Uzun
Department of Computer Science
University of Aberdeen
Aberdeen, United Kingdom
denideni21@abv.bg

Dewei Yi
Department of Computer Science
University of Aberdeen
Aberdeen, United Kingdom
dewei.yi@abdn.ac.uk

***Abstract*—Semantic segmentation is a significant technique that can provide valuable insights into the context of driving scenes. This work discusses several mechanisms: *data augmentation*, *transfer learning*, *transposed convolutions* and *focal loss function* for improving the performance of neural networks for image segmentation. Experiments on two traditional model architectures – U-net and MobileUNetV2 – are conducted and the results are evaluated in terms of – Intersection-over-Union (IoU) and F-score. The KITTI Road dataset is utilised for training and testing the algorithms on road segmentation. More specifically, data augmentation and the task-specific focal loss provide the highest improvement of 6.68% and 5.23%, respectively. To further enhance segmentation performance, an ensemble scheme is adopted where several models are executed simultaneously and their outputs are fused together to derive the final prediction. Such a design can reduce incorrect predictions of individual models and produce more precise segmentation masks.**

Keywords-semantic segmentation; focal loss; transfer learning; ensemble scheme; data augmentation

I. INTRODUCTION

In recent years, autonomous vehicles (AV) have attracted intensive attention both from academia and industry. For AV, it is an essential task to understand the surrounding environment. Semantic segmentation (SS) plays an important role in road detection. It generates a segmentation map for a given image where each pixel is assigned a label corresponding to a class. The created segmentation masks contain information about the region occupied by each object in the image. The pixel-wise classification problem involves correctly recognising an object in an image and precisely identifying its boundaries in the corresponding segmentation mask. Therefore, it is essential that for each pixel in the image the model can retain both the categorical information that defines it and its exact spatial coordinates. In real-world scenarios, the image data is affected by different conditions like variable weather, accidental camera rotation, etc. This requires the classifier of the model to be robust to image transformations and generate correct spatial predictions even in the presence of distorting factors and noise [1].

Due to the great success of deep neural networks (DNN) for feature extraction, many models have achieved state-of-the-art performance as mentioned in [2]–[4]. In this work, the U-net model is used due to its compelling performance on semantic segmentation tasks. U-net is based on an encoder-decoder architecture, which consists of two paths: a) **contracting** path and b) **expansive** path. The former gradually reduces the size of the input image while extracting different levels of contextual feature representations and hence the data at the deepest layer of the encoder has a very low resolution and contains the richest features. The latter is responsible for expanding the small image back to its original size by upsampling which positions the extracted features on the generated output mask. This "resizing" method, however, can result in inaccurate predictions due to the loss of fine-grained spatial information during the downsampling stage. To tackle this problem, U-net utilises skip connections that merge intermediate products of contracting layers with outputs of corresponding expansive layers. Such a design allows high-resolution local features to be matched with low-resolution global ones.

This paper explores the superiority of focal loss, transfer learning, transposed convolution and ensemble scheme and then combines their advantages to improve the performance of semantic segmentation for road detection in terms of per-pixel accuracy and prediction quality. An ablation study is also conducted to identify the effect of the various modifications. The key contributions of this work are summarised as follows. First, *Focal loss* is adopted to optimise the network rather the using only the conventional cross-entropy. In addition, data augmentation is employed to avoid the problem of overfitting. Second, the generalisation ability of the segmentation model is amplified by using *transfer learning* and *finetuning*. Third, the original interpolation operation is replaced with a *transposed convolution* along with experimental evaluation. Fourth, an *ensemble scheme* is introduced to combine diverse models. The final predictions are determined by all ensemble models.

II. RELATED WORK

A. Fully Convolutional Network (FCN)

The rising interest in semantic segmentation and the significant breakthroughs in deep learning have led to the development of various solutions in the field. One of the most successful inventions is the fully convolutional network (FCN). It has been proposed in [2] and since then has been utilised in adapting many state-of-the-art classification models (e.g., AlexNet, ResNet, VGG) to the task of image segmentation. This conversion is performed in two steps: (1) all fully connected dense layers of the origin networks are discarded and new convolutional ones are added in their place, (2) the final layer (the classifier) is replaced by a 1×1 convolution with channel dimension equal to the number of possible classes for each pixel. In

this way, the receptive field of the model is expanded and rather than outputting a class prediction for the whole image, the network returns a pixel-wise classification map.

U-net [3] builds upon the findings of the FCN. It is characterised by its U-like shape where the *contracting* (downsampling) path is symmetrical to the *expansive* (upsampling) path. Moreover, it improves the FCN model by extending the capabilities of the decoder (the resizing component) and by refining the quality of the products of skip connections. Also, the researchers, driven by the detrimental effects of the *"small dataset problem"* on learning outcomes, apply data augmentation and observe significant gains in terms of both accuracy and generalisation ability. Other studies [5], [6] also discover a strong correlation between the number of training samples and the performance of the resulting neural networks.

B. Transfer Learning

Transfer learning is a tool that addresses the imbalance in data availability between different domains in machine learning and encourages the principle of reusability.

Since there exist many state-of-the-art networks for image segmentation, this creates the opportunity to "transfer" top-level knowledge representations from already trained models to new problem solvers in a different but closely related domain. As a result, this approach decreases the training time of the (knowledge) receiver networks substantially and reduces the required number of labelled samples from the new domain [7]. Furthermore, this facilitates the efficient and quick capture of complex relationships present in the examples from the unseen problem space by the new models [8]. This is possible since the bottom layers in every model contain low- and mid-level feature representations (in image processing these include edges, simple shapes, line junctions, object components) which are generally applicable to all entities in the problem space.

In the architectures of deep neural networks, each layer constructs its representation based on that of its preceding layer. Therefore, in transfer learning, the low-level feature extractors of the (knowledge) sharer network can be "frozen". That is, their weights are not updated during future training and only the top (deepest) ones are randomly reinitialized and retrained. In [8], the authors demonstrate that even for the most distant problems, initialising a network with transferred features is still superior to employing random weight values.

C. Transposed Convolutions

A transposed convolution is an operation that takes the output of some convolutional layer and increases its spatial dimensions by a specified stride factor. It performs upsampling on feature maps by utilising trainable parameters. The forward pass of a standard convolutional layer is the backward pass of a transposed convolutional layer and vice versa.

This technique has been utilised for the task of semantic segmentation in many studies [2], [9]. Despite the success of transposed convolutions, other studies argue that they have poor performance due to the presence of checkerboard artefacts in their outputs. In [10], the authors confirm these limitations and propose a simple and yet effective alternative that mitigates this issue, where an interpolation operation (bilinear or nearest-neighbour) is followed by a convolution. However, they also both recognise that their propositions might introduce other limitations. The contradictory reports on the performance of this tool imply this question remains open for discussion and further research is required to fully address it.

D. Ensemble Scheme

Ensemble learning is to combine the predictions of multiple models for improving accuracy and robustness [11]. Each ensemble is a distinct entity - either a different model architecture or a network trained with different data. Since the captured feature representations vary between deep learning models due to many factors (e.g., random initialisation of kernel weights), a combination of predictions ensures to filter most individual random errors (misclassifications) [12]. Consequently, the adaptive performance of models and the accuracy of the produced predictions improves. As clarified in [13], an ensemble of networks has a statistical, computational and representational dominance over any single model.

Although this mechanism promotes great performance, it is not widely utilised in image segmentation applications because of its computational complexity. A few studies in medical research employ this method for the segmentation of brain images and detection of lesions [11], [12]. All studies report increased robustness of the networks (e.g., to noisy or low-quality data) and a better (in most cases) performance than state-of-the-art solutions.

III. Proposed Method

A. Network Architecture

In recent years, most of the state-of-the-art performance for semantic segmentation tasks is achieved by deep neural networks (DNN). Therefore, this paper also focuses on enhancing semantic segmentation performance by utilising DNN. In this paper, U-net and MobileUNetV2 are used to detect the road segments in images as baseline models. The encoder of the U-net is a standard feature extractor – each level consists of convolutions activated by the rectified linear unit (ReLU) function followed by a max-pooling operation. For the decoder component, each block is constructed using the *resize-convolution* method (nearest-neighbour interpolation followed by a convolution) like in [3] and as proposed in [14]. The modified U-net relies on a pretrained MobileNetV2 [4] for its encoder while its upsampling part comprises transposed convolutions. Skip connections between symmetric encoder-decoder blocks are also implemented in both models. The total number of parameters of each network are 31 million (U-net) and 6.5 million (MobileNetV2).

B. Implementation Details

The main objective of this work is to train several models on a given dataset and test their prediction performance on a different one from a close domain. This means that the road scene features present in the two sets must be different. For example, in the KITTI road

benchmark (discussed in Section IV.A), the three categories exactly match the above description – same image-capturing setup, different road scene classes. For all experiments (unless explicitly stated), category II - marked urban road images (**um:** $\boldsymbol{n} = 95$) is defined as the training set while category III - multi-lane suburban and urban images (**umm:** $\boldsymbol{n} = 96$) is used for evaluation. This tests the ability of the models to generalise from single-lane roadways to multi-lane motorways.

To achieve a comprehensive comparison, the baseline model is implemented and trained which provides the benchmark of semantic segmentation performance. This allows for a clear measurement of the improvement achieved by creating new modified versions of each segmentation network. To positively impact the prediction accuracy and generalization ability of these models, several different sets of modifications are defined. From this section, the word **configuration** specifies the set of hyperparameters, modifications and training procedures applied to a base network. All algorithms are implemented under the TensorFlow framework. The training process of all models is optimized using Adam with a learning rate of 0.0001 for U-net and 0.001 for MobileUNetV2.

1) Data Augmentation

The operations of data enhancement include horizontal flip, random rotation and random zoom (crop). The techniques are selected specifically based on the findings presented in [15].The occurrence probability of all is set to 1 which ensures that for each original sample in the training data, a new augmented version is generated, thus doubling the dataset size. It is essential to preserve the consistency between the training samples and their corresponding targets. Therefore, the exact same transformations are applied simultaneously to the input image and its annotated segmentation mask. The predictions, from the models of *"U-NET AUG"* and *"MobileNetV2 AUG"*, are generated and evaluated. Table II provides brief definitions of all derived variants.

2) Network Optimisation

After analysing the performance of all models from the previous task, *"U-NET AUG"* is the one with the highest-quality predictions. It is trained with two loss functions – a standard *Binary Cross-Entropy* loss and a *Binary Focal* loss. The loss function is responsible for computing the error of the model according to the training data and hence guiding its whole learning process.

The focal loss is proposed in [16] as a solution to an object detection problem associated with dense neural networks. The focal loss is a variation of the cross-entropy loss function that introduces a focusing parameter γ which reduces the contribution to the loss from easy examples. Consequently, the misclassified hard examples are penalized more severely and the model's focus is diverted to them, while the reward for guessing the easy ones correctly is reduced. The value of the focusing parameter determines the strength of its effect with 0 being equivalent to standard cross-entropy. In this paper, models are trained with three values of γ - 0.5, 1 and 2.

3) Domain adaptation with Transfer Learning

In this experiment, the knowledge captured by the base network from the **um** road class is transferred to a new model. Subsequently, the weights of some layers (different set sizes are investigated) are *frozen* (prevented from updating) and only the remaining are trained on 10% of the **umm** data. To achieve a more comprehensive comparison, configurations are defined as follows.

TABLE I. TRAINING CONFIGURATIONS (A LIST OF CONFIGURATIONS FOR APPLYING TRANSFER LEARNING TO AN AUGMENTED MODEL FROM EXPERIMENT 2 (SECTION III.*B*.2).

Configuration List	Frozen layers	Epochs	Validation split %	Focal Loss γ
Version 1 (**v1**)	1 to 9	30	20	0.5
Version 2 (**v2**)	1 to 7	30	20	0.5
Version 3 (**v3**)	1 to 5	30	20	0.5

The training samples, drawn from the new domain (**umm**), are different for each model due to the randomisation applied to the partial sampling algorithm. Since the training and evaluation data are from the same distribution, it is essential to ensure that the model is evaluated only on never-before-seen images.

All models (from Table I) are trained for 30 epochs (or less if early stopping is engaged) with a focusing parameter of 0.5 and a training-validation split of 30%, however, the frozen layers vary. The reasons behind these decisions are presented in the description list below.

Validation Split: The most popular train-validation split ratios are 80:20 and 70:30. However, this depends a lot on the size of the whole dataset. [17] investigates the effect of the split size on the correct classification rate of models and observes that it is very sensitive to the size of the partitions. Therefore, considering the samples in the KITTI dataset, it is reasonable to define a larger validation set size (30% of the total volume).

Frozen Set Size: The number of frozen layers is a factor that is found to influence the results of transfer learning in [18]. The scientists also investigate how different frozen set sizes affect the generalisation capabilities of the new model and observe a favourable increase in performance parameters when the weights of the bottom 3 transferred layers are not randomly reinitialised. However, expanding the frozen set further introduces a significant decrease in accuracy. It is acknowledged that the optimal number of frozen layers depends on the similarity between problems and the quantity of training data from the new domain. Three models are created during this set of experiments (one for each configuration) – *"U-NET TF* [v1-v3]*"*.

4) Learnable Upsampling

The upsampling blocks of *"U-NET AUG"* are replaced with transposed convolutions. The current interpolation operation is predefined (there are no learnable weights in the upsampling layers) and therefore the substitution is expected to result in better spatial positioning of predicted features which in turn will improve the quality of the segmentation masks. Additionally, the effect of two different values for γ (0.5, 1) on the pixel-wise accuracy of transposed decoders is also investigated.

Furthermore, transfer learning and finetuning are also applied to the above two networks. The benefits of unfreezing all layers of a receiver model and initiating a second training cycle with a decreased learning rate (by a factor of 10) on the predictive precision are investigated. Finetuning is expected to perform tiny adjustments to the weights of the already trained model which ensures that most redundant features, existing due to noise, for example, are pruned.

5) Ensemble Scheme

An ensemble is formed by grouping several diverse models together. *"U-NET TF v2"* and *"v3"* and *"U-NET Transposed FINETUNED"*, which are defined in Table II, are selected due to their outperformance on pixel-level classification. The finetuned U-net has a higher *TP*- and lower *FN*-rate (more road pixels predicted correctly and less misclassified background pixels) than the other two. On the other hand, both *v2* and *v3* are found to distinguish non-road pixels more precisely (higher mean *TN*) and to over-estimate the road class less severely (fewer *FP* predictions) with around 6000 pixels less on average (39.62%). Therefore, an ensemble of these models is expected to combine their strengths and produce more accurate segmentation maps.

For this experiment, a majority voting ensemble scheme is employed. The final prediction of the three models is derived from their individual segmentation masks. Each pixel in the predicted masks is encoded by a single value (0 for background or 1 for road class) which allows hard voting to be utilised. In hard voting, the final prediction for a pixel is determined by the class identifier that receives more than half of the votes. For instance, if pixel p with coordinates (x, y) is classified as follows [0], [1], [0] by the three models, the class assigned to the pixel at (x, y) in the final output would be 0 (2 > 1 votes). This is implemented by taking the sum of all class predictions for $p_{(x,y)}$ and dividing it by the number of total networks in the ensemble. The above result is calculated as follows:

$$p_{(x,y)} = \frac{\sum_{i=1}^{N} p_{(xi,yi)}}{N} = 0.33,$$

where $\boldsymbol{N}$ is the total number of models and $\boldsymbol{i}$ is the index of a certain model. For binary problems, usually, if the quotient is less than 0.5, p is encoded with a 0 (background class), while if $p_{(x,y)} \geq$ the threshold (0.5), it receives a 1.

TABLE II. TABLE OF ALL TRAINED MODELS' NAMES AND THEIR DEFINITIONS. COMMAS IN THE MODEL NAMES SEPARATE INSTANCES OF THE SAME MODEL WITH VARIATIONS IN CONFIGURATION PARAMETERS.

Model Name	Description
Base models ▪ U-NET simple ▪ MobileNetV2 simple	Base models with no modifications applied are marked with '*simple*'. They are trained with BCE loss on the original version of the KITTI dataset.
Augmented versions ▪ U-NET AUG ▪ MobileNetV2 AUG	*'AUG'* specifies that the models are trained on the augmented version of the KITTI road dataset.
Focal Loss ▪ U-NET BCE ▪ U-NET AUG	*'BCE'* stands for Binary Cross-Entropy loss while the other model is the augmented U-NET trained with Binary Focal loss.
Transfer Learning ▪ U-NET TF v1, v2, v3	*'TF'* indicates Transfer Learning, while '*v'* specifies the used training configuration
Transposed Convolutions ▪ U-NET Transposed G-0.5, G-1 ▪ U-NET Transposed TF, FINETUNED	Models where transposed convolution is used in the decoder have, '*Transposed*' in their names. '*G*' shows the value of the focusing parameter (gamma) used for the loss function.

IV. EXPERIMENTAL EVALUATION

A. Dataset

All the models presented in this document are trained and tested on images from the KITTI dataset [19]. It is developed as a vision benchmarking suite for various autonomous tasks associated with AV. The images used in this research are from the road detection data class. This benchmark contains 289 training samples divided into three categories - urban scenes with no lane markings (**uu**), marked urban road images (**um**) and multi-lane suburban and urban images (**umm**). A segmentation mask has been provided for each of these samples. Three distinct classes can be assigned to a pixel in the original labels - the current roadway, the opposing roadway and the non-road background. However, for the purposes of this study, a distinction only between the current roadway and the background is sought. Therefore, all opposing road pixels in the segmentation masks are marked as non-road. This reduces the problem to a binary segmentation task.

B. Performance Metrics

The following metrics are adopted for measuring the performance of the models – Precision, Recall, Pixel Accuracy and F_1-measure (harmonic mean of recall and precision). They are widely employed for the evaluation of pixel-based tasks. The developers of the KITTI road benchmark also incorporate these criteria into their model comparison framework [19]. Another metric, namely intersection over union (**IoU**) or Jaccard index, is also a very accurate performance measure for segmentation tasks which represents the overlap between predictions and ground truths in percentages. It is adopted in [2] as a standard for evaluating the predictive ability of FCNs.

Both the IoU and the F_1 measures can be calculated based on the Recall and Precision metrics, which can be expressed using the numbers of true positives (*TP)*, true negatives (*TN*), false positives (*FP*) and false negatives (*FN*). These values are retrieved by overlaying each prediction on the top of their corresponding ground truths and counting the matched (*TP, TN*) and misclassified (*FP, FN*) pixels. The precision metric indicates how pure a prediction is while recall describes how well the model distinguishes between the different classes in the image. The values of all metrics are calculated by averaging the total score for all images in the testing set.

C. Comparative Studies

Table III illustrates the performance achieved by all modified networks. Furthermore, Fig. 1 graphically presents the results by ranking them based on IoU scores and the F_1 measures. The most distinguishable difference is observed in terms of IoU scores.

TABLE III. COMPARISON OF THE DIFFERENT MODELS

Model Names	F_1	Precision	Recall	PA	IoU
MobileNet AUG	0.588	0.822	0.923	0.981	0.768
MobileNet simple	0.565	0.728	**0.938**	0.975	0.695
U-NET BCE	0.591	0.835	0.929	0.982	0.783
U-NET Transposed G-0.5	0.600	0.879	0.920	0.984	0.816
U-NET Transposed G-1	0.601	0.906	0.908	0.985	0.832
U-NET AUG	0.603	0.892	0.918	0.985	0.824
U-NET Transposed FINETUNED	0.595	0.982	0.857	0.985	0.844
U-NET TF v1	0.592	**0.965**	0.858	0.984	0.832
U-NET TF v2	0.592	0.924	0.894	0.985	0.834
U-NET TF v3	**0.609**	0.957	0.900	**0.988**	**0.865**
U-NET Transposed TF	0.596	0.967	0.865	0.985	0.841
U-NET simple	0.570	0.865	0.859	0.978	0.761

From the results, it can be deduced that transfer learning provides one of the easiest and most effective ways of improving models' performance as it does not require high volumes of data or sufficient training time. An increase in performance of 4.71% is observed when the tool is applied on *"U-NET AUG"*. Additionally, the transposed versions of U-net also follow that trend with 1.37% improvement when transfer learning is utilised. Furthermore, it is noticeable that the Precision metric is positively influenced in all models trained using this approach, while, on the other hand, a decrease in Recall values is present in these networks. This means that the *FN*-rate reduces and *FP*-rate increases where the models are overpredicting the road class and under-segmenting the background (non-road) class. The *"U-NET TF v3"* model performs the best with regards to F_1 score, IoU and pixel accuracy.

Next, it appears that binary focal loss dominates the traditional binary cross-entropy. The performance gained from replacing the standard loss function with a task-specific one is 5.28%. In both *"U-NET AUG"* and *"U-NET Transposed"*, $\gamma = 1$ provides the most promising results.

Another successful experiment shows the tremendous advantage of applying data augmentation when data availability is an existing limitation. Training both U-net and MobileUNetV2 on an artificially extended dataset (with less than 100 added images) increases the percentage of overlap between predictions and targets by 6.69% on average - 10.5% (MobileUNetV2) and 2.88% (U-net).

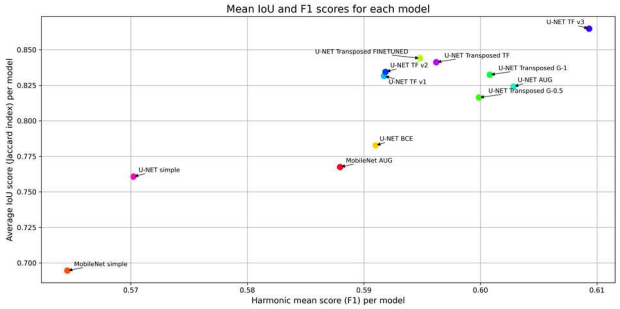

Figure 1. A scatter plot of all neural networks and their mean scores (F_1 measure on the horizontal axis and IoU on the vertical axis). The best performing models are positioned at the top-right corner of the image.

When transposed convolutions are utilised in the expansive path of U-net instead of nearest-neighbour interpolation, very little or no improvement is observed. The reason for this might be the growth in trainable parameters (more than 4 million added weights) caused by the addition of more convolutions (learnable kernels) to the network. In that case, more iterations over the data (epochs of training) can allow the models sufficient time to learn and potentially mitigate this issue.

By employing the ensemble scheme an additional 1.5% increase in F_1-measure and IoU is achieved. Moreover, the results in Table IV show a significant decrease (12.40%) in false negatives and an optimistic step towards reducing false positives (a 3.06% improvement). The other two metrics TP and TN are not affected by the ensemble approach. Additional tests can be performed with soft voting, where the confidence scores of pixels are summed instead of their class labels.

TABLE IV. THE IMPROVEMENT OF USING ENSEMBLE SCHEME

Model Names	TP	TN	FP	FN
U-NET Ensemble 2	104183	344942	11289	4493
U-NET TF v3	103646	344511	11645	5129
Percentage of change	**0.52%**	**0.13%**	**-3.06%**	**-12.40%**

Finally, the inference time of all neural networks is compared. The milliseconds (ms) taken by each model to generate a prediction are recorded on 2000 prediction attempts and then averaged. MobileNetV2 achieves the highest score with a mean inference time of around 60ms (16.6 frames per second). All variations of U-net have a very similar performance of around 70ms (14.3 frames per second). Fig. 2 illustrates the mean inference times of all modifications sorted from fastest to slowest. Compared to the top three vision-based state-of-the-art solutions submitted to the KITTI Road benchmark for the **umm** class – SNE-RoadSeg+ (40ms), FDS-DeepLabV3+ (60ms) and ZongNet (100ms), both U-net and MobileUNetV2 perform very well. The ensemble of models can process 4.3 frames per second. Reducing this requires an optimisation of the vote-counting algorithm. Model fusion is also another solution.

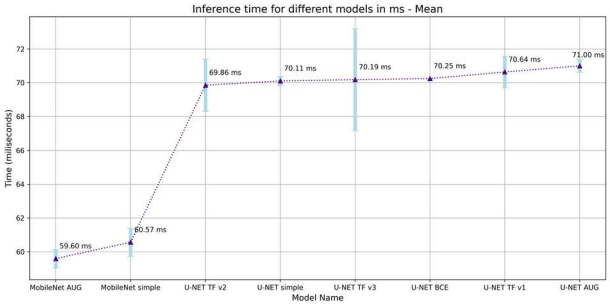

Figure 2. The figure of the mean inference time in milliseconds (ms) for all models for a single image (each score is averaged from 2000 attempts). The vertical bars at each point represent the standard deviation in runtimes between the different samples or prediction instances.

V. CONCLUSION AND FUTURE WORK

This paper improves the semantic segmentation performance of convolutional neural networks for road detection by introducing an ensemble scheme along with transfer learning and transposed convolutions. Two models, U-net and MobileNetV2-based U-net, are modified and evaluated over a range of performance metrics. The dataset utilised in training and testing is

provided by the KITTI benchmarking suite. All conducted experiments, especially those that involve transfer learning and data augmentation, indicate great improvements in terms of IoU scores. More precisely, the data augmentation introduces a 6.69% increase in Intersection-over-Union while domain adaptation (transfer learning) achieves 4.71% improvement for the same metric. This demonstrates that synthetically generating training samples by augmenting existing ones can alleviate the problem of overfitting. Transfer learning also tackles the issue with low data availability and at the same time reduces the training time sufficiently by reusing low- and mid-level features of existing models. Moreover, the focusing parameter of the focal loss improves segmentation accuracy, especially when class imbalance is present in the data. The proposed method has similar inference times to state-of-the-art methods. The main contribution of this work is the proposed ensemble scheme, where the segmentation maps of several versions of a certain model derived utilising different combinations of modifications are fused to reduce pixel-level errors. The ensemble method achieves a decrease of 12.40% in false negatives and 3.06% in false positives. Therefore, it can be deduced that combining multiple segmentation models utilises their individual strengths and reduces the impact of their limitations on the accuracy of the predicted masks. In the future, a more advanced network for road segmentation will be adopted. Moreover, a more comprehensive study of hyperparameters will also be conducted. To emphasise the advantages of the proposed integration-ensemble scheme in real scenarios, experiments on larger datasets will be required.

REFERENCES

[1] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large Kernel Matters — Improve Semantic Segmentation by Global Convolutional Network," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1743–1751.

[2] E. Shelhamer, J. Long, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 4, pp. 640–651, 2017.

[3] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*, 2015, pp. 234–241.

[4] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 4510–4520.

[5] Y. Zhang and C. Ling, "A strategy to apply machine learning to small datasets in materials science," *Npj Comput. Mater.*, vol. 4, no. 1, pp. 1–8, 2018.

[6] C. Sun, A. Shrivastava, S. Singh, and A. Gupta, "Revisiting Unreasonable Effectiveness of Data in Deep Learning Era," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 843–852.

[7] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1717–1724.

[8] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How Transferable Are Features in Deep Neural Networks?," in *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, 2014, pp. 3320–3328.

[9] H. Gao, H. Yuan, Z. Wang, and S. Ji, "Pixel Transposed Convolutional Networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 5, pp. 1218–1227, 2020.

[10] A. P. Aitken, C. Ledig, L. Theis, J. Caballero, Z. Wang, and W. Shi, "Checkerboard artifact free sub-pixel convolution: A note on sub-pixel convolution, resize convolution and convolution resize," *ArXiv*, vol. abs/1707.0, 2017.

[11] J. V Manjón *et al.*, "MRI white matter lesion segmentation using an ensemble of neural networks and overcomplete patch-based voting," *Comput. Med. Imaging Graph.*, vol. 69, pp. 43–51, 2018.

[12] J. Dolz, C. Desrosiers, L. Wang, J. Yuan, D. Shen, and I. Ben Ayed, "Deep CNN ensembles and suggestive annotations for infant brain MRI segmentation," *Comput. Med. Imaging Graph.*, vol. 79, p. 101660, 2020.

[13] T. G. Dietterich, "Ensemble methods in machine learning," in *International workshop on multiple classifier systems*, 2000, pp. 1–15.

[14] A. Odena, V. Dumoulin, and C. Olah, "Deconvolution and Checkerboard Artifacts," *Distill*, 2016.

[15] L. Taylor and G. Nitschke, "Improving Deep Learning with Generic Data Augmentation," in *2018 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2018, pp. 1542–1547.

[16] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal Loss for Dense Object Detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 318–327, 2020.

[17] Y. Xu and R. Goodacre, "On Splitting Training and Validation Set: A Comparative Study of Cross-Validation, Bootstrap and Systematic Sampling for Estimating the Generalization Performance of Supervised Learning," *J. Anal. Test.*, vol. 2, pp. 249–262, 2018.

[18] D. Soekhoe, P. Van Der Putten, and A. Plaat, "On the impact of data set size in transfer learning using deep neural networks," in *International Symposium on Intelligent Data Analysis*, 2016, pp. 50–60.

[19] J. Fritsch, T. Kühnl, and A. Geiger, "A new performance measure and evaluation benchmark for road detection algorithms," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*, 2013, pp. 1693–1700.