applied sciences

MDPI

Article

One For All: A Mutual Enhancement Method for Object Detection and Semantic Segmentation

Shichao Zhang, Zhe Zhang, Libo Sun and Wenhu Qin *

School of Instrument Science and Engineering, Southeast University, Nanjing 210096, China; zhangshichaoseu@foxmail.com (S.Z.); zhangzhecnjs@gmail.com (Z.Z.); sunlibo@seu.edu.cn (L.S.)
* Correspondence: qinwenhu@seu.edu.cn; Tel.: +86-136-0518-7156

Received: 11 November 2019; Accepted: 13 December 2019; Published: 18 December 2019

Abstract: Generally, most approaches using methods such as cropping, rotating, and flipping achieve more data to train models for improving the accuracy of detection and segmentation. However, due to the difficulties of labeling such data especially semantic segmentation data, those traditional data augmentation methodologies cannot help a lot when the training set is really limited. In this paper, a model named OFA-Net (One For All Network) is proposed to combine object detection and semantic segmentation tasks. Meanwhile, using a strategy called "1-N Alternation" to train the OFA-Net model, which can make a fusion of features from detection and segmentation data. The results show that object detection data can be recruited to better the segmentation accuracy performance, and furthermore, segmentation data assist a lot to enhance the confidence of predictions for object detection. Finally, the OFA-Net model is trained without traditional data augmentation methodologies and tested on the KITTI test server. The model works well on the KITTI Road Segmentation challenge and can do a good job on the object detection task.

Keywords: "1-N Alternation" strategy; OFA-Net; object detection; segmentation; feature fusion

1. Introduction

In recent years, convolutional networks (ConvNets) contributed a lot to the dramatic improvements in computer vision-related tasks. ConvNets not only boosted image classification related tasks [1–6] but also made significant progress on object detection [7–13] and semantic segmentation tasks [14–22]. Object detection and semantic segmentation have a wide spread of applications from scene understanding to video monitoring, and they are also two important parts of autonomous driving.

He et al. proposed ResNet [4] in 2015, which overcame the difficulties of training very deep neural networks and let researchers achieve more and more complex models. Deep neural networks had been requiring large quantities of data to train [1] and more so after ResNet [4]. Most of us, however, are not always able to get enough data. Taking image semantic segmentation as an example, especially for driving environmental image segmentation task, it seems that people are lacking such data all the time. The famous KITTI semantics benchmark [23] just has 200 train images, and the KITTI road segmentation benchmark [24] only contains 289 training images, so it seems too few to train a very deep network, let alone researchers usually have to split training sets to make evaluations. With the high cost of labeling such data, it is hard work to create custom datasets for your own driving environment. Although there are several data augmentation methods, such as cropping, rotating, flipping, wrapping, and over-sampling [25,26] that can help to acquire additional training data; researchers have also explored many exciting models to get more accurate segmentation results, these models still cannot work very well when the training data is really limited.

Zeiler and Fergus [27] demonstrated that the features learned by ConvNets are hierarchical, while the bottom layers focus on low-level features like corners, edges, etc., the top layers pay more attention to high-level features. Inspired by this idea, this paper proposes a model called OFA-Net (One For All, which means One model For All results required) to do driving environment images road segmentation and object detection tasks.

In view of the fact that object detection data is much easier to label, OFA-Net utilizes KITTI 2D Object Detection Dataset [28] as a supplementary source to learn the data distribution, which can surely benefit the performance of road segmentation task. OFA-Net is a special multi-task fusion model [29] because it can output object detection and semantic segmentation results simultaneously. The model consists of three parts serving as feature extractor, detection, and segmentation, respectively. It feeds object detection and semantic segmentation data alternately, and uses two different loss functions to train each task, respectively. During this process, the model attempts to locate a proper point in the high dimensional parameter space where it performs well on both. This paper shows that by mixing object detection data with segmentation data using our "1-N Alternation" strategy, this unified multi-task learning [29] model can be trained faster, more accurate, with better generalization ability for the road segmentation task and high prediction confidence for the object detection task. In the meantime, training a model via fusing these two related datasets can be viewed as a data augmentation and model regularization method as well.

To conclude briefly, the work of this paper demonstrated that object detection and semantic segmentation can benefit from each other by borrowing some fusion features when training a unified fusion model by feeding data alternately. The advantages for the OFA-net with the "1-N Alternation" strategy are speeding up the convergence, improving segmentation accuracy and enhancing prediction confidence for object detection.

2. Related Work

Recently, Deep ConvNets play a significant role in image-related work. Hence this paper lays the emphasis on Deep ConvNets based models. Firstly, some papers on classification, object detection, and semantic segmentation are reviewed respectively, and then some work on transfer learning, multi-task learning, and simultaneous detection & segmentation efforts previously are listed carefully.

2.1. Classification, Detection, and Segmentation

Classification Recent years most image classification models utilize ConvNets after the success of AlexNet [1]. Some researchers claimed that network depth is crucial for network performance [3,5]. Unfortunately, training a very deep neural network was difficult due to the problem of vanishing/exploding gradients. ResNet [4] proposed by He et al. is the state-of-the-art structure, which allows people to train very deep networks without worrying about the problems mentioned above. Some researchers pointed out that Deep ConvNets learn image features hierarchically, which is, convolutional kernels of bottom layers extract low-level features while the top layers are to combine the low-level features as high-level representations [27].

Detection Traditional methods for object detection usually followed a two-step strategy. Firstly, the model proposed numerous region proposals [30], then classified them to get correct predictions. RCNN [7] and Fast RCNN [8] were the typical representations of this strategy. These methods were easy to understand and implement. Nevertheless, due to the multi-stage training pipeline, it is too slow and contains too much repetitive computation and cannot be trained end-to-end. Although the Faster RCNN [9] model made use of RPN (Region Proposal Network) and solved these problems partially, it was still too slow to use. Fortunately, there are several models that have been proposed to use only one Deep ConvNet trainable end-to-end to make detections directly, such as YOLO models [10,11,13] and SSD [12]. These models are superior to those region proposal-based models mainly for their fast train and inference speed, and thus more appropriate for real-time object detection work. Thus, the OFA-Net model leverages a lot from the YOLO models [10,11,13].

Segmentation For the semantic segmentation task, some early approaches attempted to adapt the model architectures designed for image classification directly to pixel-wise label prediction [31]. Despite the results surpassing traditional models, which heavily relied on low-level features hand-engineered by humans [32–35], they still appeared fairly coarse [17]. Researchers tried a lot to fine the segmentation results. FCN (Fully Convolutional Network) [14] replaced the last fully connected layer with a convolutional layer and then added an up-sampling layer to recover the image size. The biggest problem of FCN is the network lost a lot of information due to the pooling layers, thus led the results not fine enough. To aggregate multi-scale contextual information with no resolution lost, Yu and Koltun created dilated convolution modules [16]. To fine the prediction result, Badrinarayanan et al. designed an encoder-decoder architecture [15], in which the encoder network was identical to normal ConvNets and the decoder network was to learn the map between the low-resolution encoder and high-resolution feature maps whose size is identical to the input images. Recently, Chen et al. [21,22] combined the dilated convolutional module and the encoder-decoder architecture together and borrowed the idea of the SPP (Spatial Pyramid Pooling) [6,20] to create an ASPP module (Atrous Spatial Pyramid Pooling) for better performance on multiple scale segmentation.

2.2. Transfer Learning and Multi-Task Learning

Transfer Learning refers to the problem in the machine learning field that focuses on storing knowledge obtained from a task and applying it to a different but related task [36]. Transfer learning has been demonstrated on various computer vision-related tasks from image recognition [27,37] to detection and segmentation [7,38]. The reason why it works is that features learned by Deep ConvNets are hierarchical [27]. Both object detection and semantic segmentation need Deep ConvNets as the infrastructure, hence these two tasks require similar features.

Multi-Task Learning (MTL) is the strategy that aims to leverage information among multiple related learning tasks to improve the performance of all those tasks [39]. MTL obviously utilizes fusion features because it learns features useful for every related task. Simultaneous detection and segmentation can be seen as an application of multi-task learning, and more details will be given in the next section. The OFA-Net leverages this idea and demonstrates that by training detection and segmentation alternately, it is able to better the performance on both tasks in some aspects.

2.3. Simultaneous Detection and Segmentation

People have made a lot of effort to achieve simultaneous detection and segmentation. Wu and Nevatia [40] recruited the Edgelet feature to capture objects' local shape and then built detection and segmentation simultaneously on top of it. Meanwhile, they utilized boosting algorithms to enhance performance. However, due to the dependence on hand-engineered features, their model only worked on simple datasets. Yao et al. [41] resorted to the convergent message-passing algorithm to inference their joint object detection, scene classification, and semantic segmentation model, which performed better than previous work. Hariharn et al. [42] attempted to use ConvNets do simultaneous detection and segmentation. Their model was built on R-CNN [7] and appeared as multi-stage for they just used ConvNets as a fixed feature extractor. OFA-Net model described in this paper can also use detection and segmentation simultaneously but the feature extractor module was unfixed. Making the feature extractor unfixed has many advantages as it can behave not only as a feature extractor module but a bridge allowing the detection and segmentation to borrow features from each other.

3. The One for All Network (OFA-Net)

In this paper, a model named OFA-Net is proposed. OFA means "One For All", which means that all results wanted can be retrieved with just one model. Next, it is important to explain what "all results wanted" refers to. "All" not only means simultaneous detection and segmentation, but also stands for exploring the mutual effects between the two tasks, and proving that the mutual effects are beneficial to both detection and segmentation in some aspects. The OFA-Net architecture is shown in

Figure 1. OFA-Net is composed of three modules: the feature extractor module, the object detection module, and the segmentation module. Finally, an alternate training strategy called "1-N Alternation" is implemented to train the model and explore the relationships between detection and segmentation tasks. More details are illustrated in the rest of the paper.

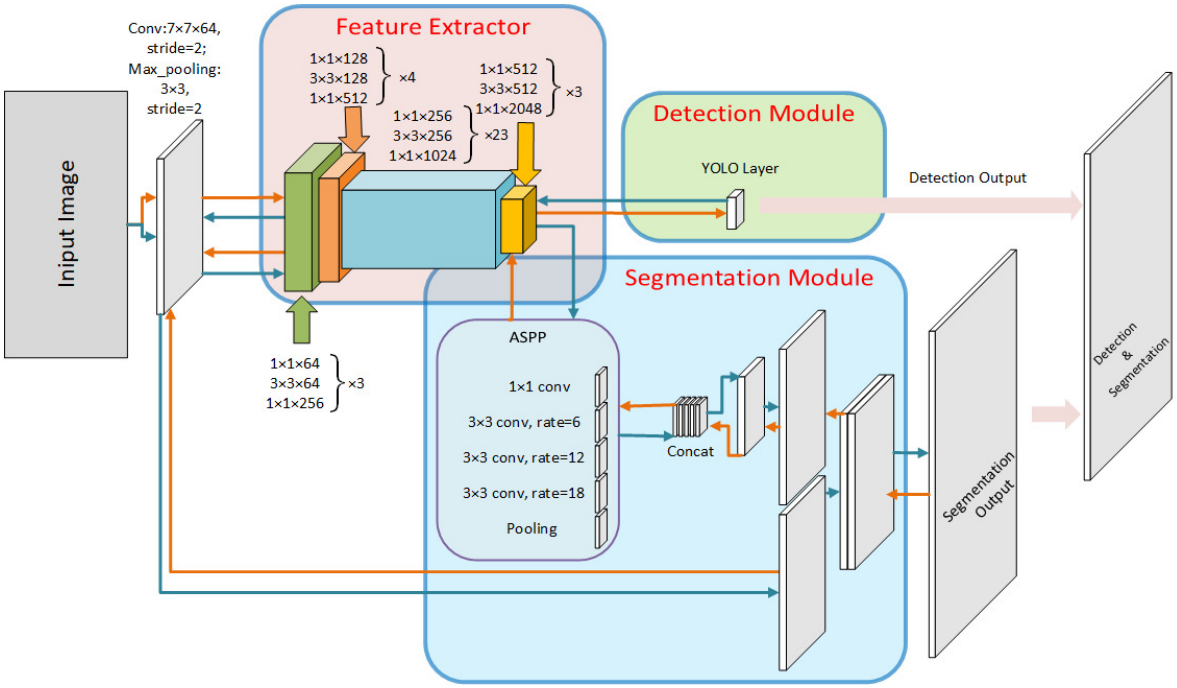


Figure 1. The One For All network (OFA-Net) architecture (rightward arrows indicate the forward process while leftward arrows indicate backward process).

Feature Extractor The feature extractor module is the first module in the OFA-Net model, and this module is almost identical to the ResNet-101 model [4]. The reason why the ResNet model is chosen as the infrastructure is that using ResNet can train a very deep network. It is not necessary to worry about the vanishing/exploding gradient problem too much and, therefore, research can focus on analyzing the reciprocal effects between detection and segmentation. The task of this module is to process input images and abstract the contextual features required by the following modules. The only change of ResNet is that the last convolutional layer is replaced with a dilate convolutional layer of dilation rate 2 for the reason that dilate convolutional module has the ability to aggregate contextual information without losing resolution [16]. Some other advantages by doing this replacement are improving the performance on the segmentation task and letting the detection module be able to access more elaborate contextual information, which can lower the confidence loss value. By training with the "1-N Alternation" strategy, this module also behaves as a fusion bridge that permits detection and segmentation tasks to borrow features from each other. More details will be given later.

Object Detection Another crucial module of the OFA-Net model is the object detection module. There are two types of object detection models currently, one is the region proposal-based method [7–9], and the other is the non-region proposal-based method [10,11,13]. The non-region proposal-based method, such as YOLO, performs training and inference faster. This research recruits the structure of YOLO and implements it as an abbreviated version. Just as YOLO models did, every input image is divided into 11×39 cells, and if the center of an object falls into a cell, that cell is in charge of predicting the corresponding object. In this work, the original YOLO model is simplified by reducing the anchors to 3 at each cell. This simplification is based on the observation that when combined with segmentation and using the dilated convolutional module, the model has already retrieved enough information for detection with a small number of anchors. Another factor in reducing the number of

anchors is that the KITTI 2D Object Detection dataset [28] does not contain so many small instances that need a large number of anchors to detect.

Segmentation The last module of the OFA-Net model is the segmentation module. Following the implementation of Chen et al., the module is illustrated in Figure 1. This module uses ASPP (Atrous Spatial Pyramid Pooling) [18,21,22] to aggregate contextual feature information at different scales. An encoder-decoder structure [15] is also adapted to combine low-level features with those high-level ones and then an up-sampling operation is utilized to restore the resolution of the feature maps generated by the former layers. You can see a lot of applications of the dilate convolutional module in the segmentation part of the OFA-Net model. Dilate convolutional modules allow the model to keep the resolution of feature maps while getting more contextual information. Additionally, it enables the OFA-Net model to enlarge the receptive field while keeping the number of parameters unchanged [16].

4. Training Details

4.1. Initialization

Initialization is crucial for Deep ConvNets [43]. As features learnt by Deep ConvNets are applicable to many related datasets and tasks, even distant tasks [44] using transfer learning to initialize the model is a good approach. The feature extractor module was initialized with pre-trained ResNet 101 parameters except for the last few layers. The Relu function [1] was utilized as the activation function in the OFA-Net model. Considering the rectifier non-linearities, Kaiming initialization [36] was adapted to initialize the blocks in the rest layers containing detection and segmentation modules.

4.2. Loss Functions and Loss Value Balancing

For the segmentation task, the OFA-Net model resorted to softmax cross-entropy loss function as many others did [14–22].

The loss function for object detection is more complex owing to the cause that the detection task requires not only classification outputs but boxes prediction as well. The Softmax cross-entropy loss function is appropriate for classification tasks whereas it is helpless when encountering the boxes prediction problem. Mean square error (or L2 loss) function is appropriate to predict bounding boxes because the bounding boxes prediction problem can be viewed as a regression problem. YOLO loss function [10] gave a good example of combining these two types of loss functions, thus the model of this paper recruited it here.

The last point about loss functions is "loss value balance." The same thinking is also contained in the YOLO loss function, which assigns different weights to the loss values according to their importance. In this work, detection and segmentation are considered equally important, therefore, an adjustment needs to be carried out to make these two loss values approximately equal. During the training process, our team discovered that the loss value of segmentation is much bigger than that of detection. Therefore, the segmentation loss value should be divided by a big constant number K to achieve our goal. To find this big constant number, the values of the two loss functions (one for the object detection and the other for the semantic segmentation) were recorded during the first several training iterations, say n iterations. Then, the big constant number K can be computed as the Formula (1) and the final loss function is shown as the Formula (2), where $Loss_{dection}$ and $Loss_{segmentation}$ indicate the original loss value of the object detection and segmentation respectively, and K is the big constant number in the Formula (1) to achieve loss value balance.

$$K = \frac{\sum_{i=0}^{n} Loss_{segmenation}}{\sum_{i=0}^{n} Loss_{detection}} \tag{1}$$

$$Loss = Loss_{detection} + \frac{1}{K} \times Loss_{segmenation} \tag{2}$$

4.3. Alternate Training Strategy

According to the research made by Zeiler [27] and Yosinski [44], the features learned by deep neural networks can be applicable to some related tasks even those distant ones. Compared with obtaining object detection data, people are always having difficulties to access massive segmentation data. Taking the KITTI dataset [23,24,28] as an example, the dataset includes 7481 training images for detection while only 289 training images for road segmentation and 200 training images for multi-class pixel-wise semantic segmentation (call "pixel-wise segmentation" later). The cause of this situation is the case that the cost is much higher to label segmentation datasets. To address this problem, an alternate training strategy was created in this paper. It firstly feeds detection data into the model, computes the detection loss function and completes the corresponding backward propagation. Following this, feed segmentation data into the model, then compute the segmentation loss function and accomplish the relevant backward propagation. And repeat these two procedures alternately. This "one batch of segmentation data feeding with one batch of detection data feeding" method is called the "1-1 Alternation" strategy. Likewise, "1-N Alternation" means one batch of segmentation data feeding with N batches of detection data feeding. To make it clear, the process is displayed in Figure 1. The rightward arrows indicate the forward process while the leftward arrows indicate the backward propagation process. Different colors (red and blue) are used to identify the alternate strategy here. For instance, feed segmentation data through blue-rightward arrows (forward process) and adjust the parameters by the red-leftward arrows (backward process), and the adjusted model is next used for the detection task following the red-rightward arrows. Using the "1-N Alternation" strategy, the OFA-Net is able to locate an acceptable point in the high dimensional parameter space where the model performs well on both tasks.

The performance of the "1-N Alternation" strategy is also tested to seek out the proper mixture ratio of detection data and segmentation data, where N is equal to 1, 2, and 5. Finally, convergence speed, IOU (Intersection Over Union) for segmentation, precision for detection, and some additional related indicators are measured.

4.4. Dataset Split and Experiments

Experiments are performed on the KITTI datasets [23,24,28]. The datasets are split in Table 1.

Table 1. Datasets split.

Dataset	Total	Train	Validation
2D object detection	7481	6000	1481
Road segmentation	289	240	49
Pixel-wise segmentation	200	160	40

The "distance" between object detection and pixel-wise segmentation was shorter than that of object detection and road segmentation because both object detection and pixel-wise segmentation have similar categories, such as cars, persons, bicycles, buses etc., while road segmentation has just two categories: road and non-road. The performances of the OFA-Net were compared in accordance with this different distance. The following parts of this paper summarize the mutual effects between these related tasks and concludes some rules to train a reliable model.

4.5. Hyper Parameters

Adam optimizer [45] was recruited to train the OFA-Net model. The model adopts a learning rate of 1×10^{-5}, and a weight decay of 5×10^{-4} is applied to all layers. The OFA-Net is also equipped with batch normalization which can stabilize the learning process. The learning rate decreases by the

Formula (3) every 10 epochs, where *current_epoch* indicates the current epoch value, and *max_epochs* indicates the maximum epoch value that our program will run.

$$lr = lr \times (\frac{1 - current\_epoch}{max\_epochs})^{0.9} \tag{3}$$

5. Results Analysis

5.1. How Does Detection Affect Segmentation?

Firstly, the model pays attention to the impacts on segmentation performance. To figure out the influences on segmentation task by mixing detection data, the IOU index is collected over training epochs. The results are presented in Figures 2 and 3. The legend "1-N" means the "1-N Alternation" strategy is employed to train the model. The models were trained and evaluated on the train and validation sets specified in Table 1.

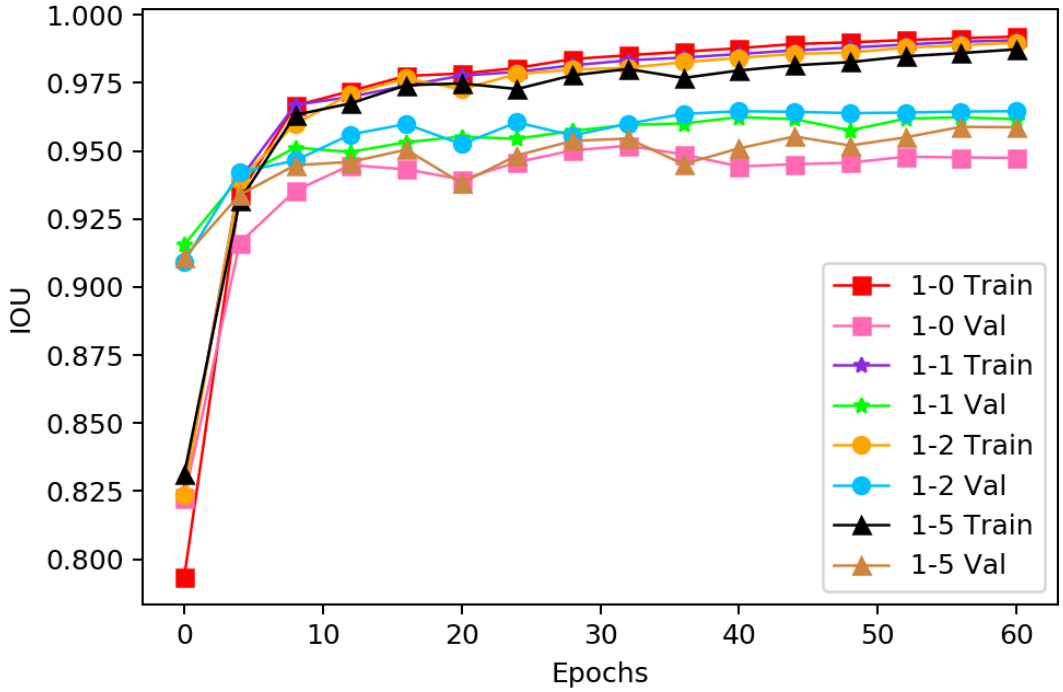


Figure 2. IOU (intersection over union) of road segmentation over epochs.

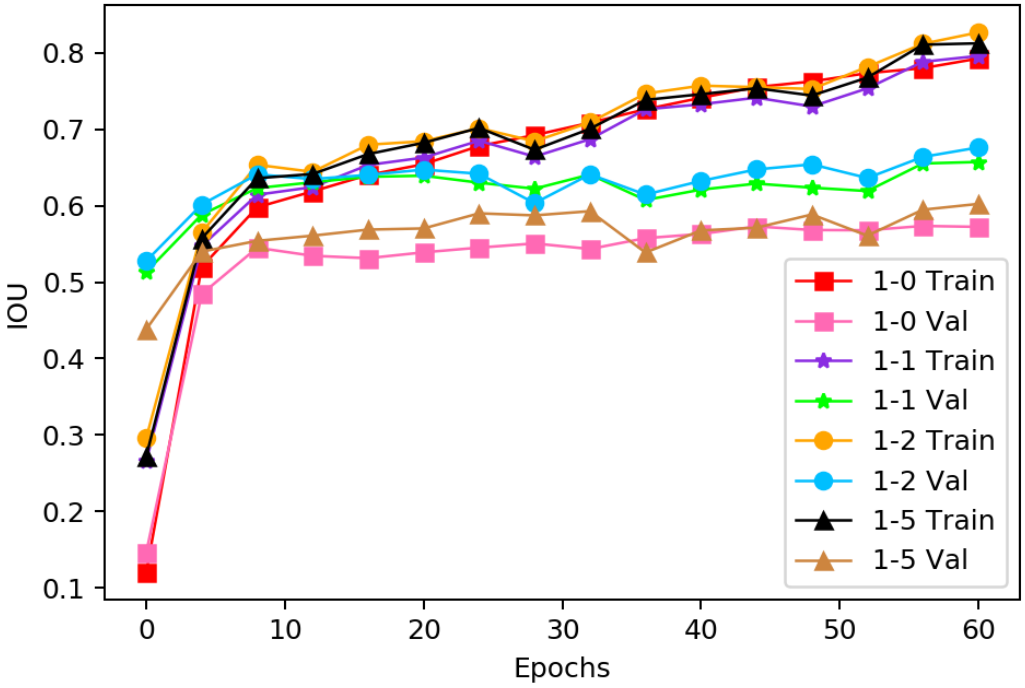


Figure 3. IOU (intersection over union) of pixel-wise segmentation over epochs.

Figure 2 displays the results of mixing road segmentation data with detection data, and the IOUs are evaluated when N is equal to 0, 1, 2, and 5. What is discovered is that by adding detection images as an additional training data source, the model performs better anyway. Specifically, when joining detection images, the model converges fairly faster compared with training only with road segmentation data, especially during the first few epochs. To be more specific, considering the performance on the validation set, the first epoch of the "1-2 Alternation" model performs almost comparable to the 10th epoch of the "1-0 Alteration" (without detection data). This phenomenon tells the fact that the features for object detection have a strong correlation with the segmentation task. Figure 2 also shows that the "1-2 Alternation" model got about 2% improvement compared with the "1-0 Alternation" model. Another thing observed is that the "1-2 Alternation" model behaves a little better than the "1-1 Alternation" model. So, is it beneficial for a bigger N all the time? To solve this question, N was increased to 5 but the result shows that it is not always better for a bigger N. If N is too big, the IOU curve will fluctuate over epochs and sometimes performs even worse than the model without detection data.

Figure 3 describes the results of mixing pixel-wise segmentation data with object detection data. Let N be equal to 0, 1, 2, and 5 again, the model produces similar results found in Figure 2. And considering the aim of KITTI pixel-wise segmentation is to pick fine area of objects like cars, persons and so on, the distance between this task and detection task is shorter than that of road segmentation and detection tasks, and as a consequence of which, the improvement of IOU is more obvious (approximately 9% improvement).

Based on the findings mentioned above, a suggestion is that when the segmentation dataset is limited or difficult to collect due to high labeling cost or some other reasons, people should try to label some detection data with the amount of one or two times of the segmentation data. By adding these detection data to the training processes, people should get a much better result. From this aspect, the "1-N Alternation" strategy can be viewed as a data augmentation methodology. Notice that in Figures 2 and 3 the IOUs on the training sets are very similar regardless of the value of N, whereas

the IOUs on the validation sets change a lot due to different values of N, which means the model generalizes better when N is appropriate. Therefore, the strategy designed in the OFA-Net model can be seen as a regularization method as well.

5.2. How Does Segmentation affect Detection?

The OFA-Net model is evaluated on the detection validation sets and the results are shown in Table 2. An interesting discovery is that when training with the segmentation data, the confidence of the objects with "enough" instances [28] (Cars, Persons) becomes higher. This arises from more detailed scene information supplied by the segmentation task. Due to the small amount of validating instances of the cyclist category, it cannot benefit from this. Besides, some cyclist objects were classified into the person category with segmentation data added as the supplementary data source. Therefore, if applying the OFA-Net model and the "1-N Alternation" training strategy to the object detection applications, "enough" validation instances for object detection are needed. Now, the question is what "enough" means. To give an answer to it, the number of instances over the number of images ratio for each category, say $R_{[category]}$, is calculated as the Formula (4). Notice that R_{Car} and R_{Person} are 4.7 and 1.3, respectively, while $R_{Cyclist}$, R_{Van}, R_{Truck}, and R_{Tram} are much smaller than 1, specifically, 0.07, 0.52, 0.17, and 0.03 respectively. What can be concluded from these ratio values is that the "enough" instances cannot be less than the number of images, or the $R_{[category]}$ at least should be greater or equal to 1. Based on this finding, one suggestion is that the amount of validating instances for each category in the detection datasets should not be smaller than the number of images in those datasets.

$$R_{[category]} = \frac{Num\ of\ Instances\ for\ Category}{Num\ of\ Images} \tag{4}$$

Table 2. Confidence changes with/without segmentation data

Confidence	Without Seg Data	With Seg Data
Car	0.9323	0.9947
Person	0.8725	0.9241
Cyclist	0.7052	0.6908
Van	0.8324	0.7925
Truck	0.8893	0.7236
Tram	0.7614	0.6867

5.3. OFA-Net Results

The OFA-Net model was tested on the official KITTI test server and the results were also compared with other's models, including the MixedCRF model [46], the ALO-AVG-MM model [47], the HybridCRF model [48], and the HID-LS model [49]. The results of the comparisons are given in Table 3 [50]. The meaning of the indexes in Table 3 is shown in Table 4. Table 3 shows that the OFA-Net model has significant advantages over other models especially on the index of MaxF and REC.

Table 3. Road segmentation evaluation results [50] *.

Benchmark	Model	MaxF	AP	PRE	REC	FPR	FNR
UM_ROAD	OFA-Net	92.08 %	82.73 %	87.87 %	**96.72** %	6.08 %	**3.28** %
	MixedCRF	91.57 %	84.68 %	90.02 %	93.19 %	4.71 %	6.81 %
	ALO-AVG-MM	91.15 %	83.82 %	89.07 %	93.33 %	5.22 %	6.67 %
	HybridCRF	90.99 %	85.26 %	90.65 %	91.33 %	4.29 %	8.67 %
	HID-LS	93.10 %	86.38 %	91.89 %	94.33 %	3.79 %	5.67 %
UMM_ROAD	OFA-Net	**95.43** %	89.10 %	92.78 %	**98.24** %	8.41 %	**1.76** %
	MixedCRF	92.75 %	90.24 %	94.03 %	91.50 %	6.39 %	8.50 %
	ALO-AVG-MM	94.05 %	90.96 %	94.82 %	93.29 %	5.60 %	6.71 %
	HybridCRF	91.95 %	86.44 %	94.01 %	89.98 %	6.30 %	10.02 %
	HID-LS	94.89 %	91.46 %	95.37 %	94.42 %	5.04 %	5.58 %
UU_ROAD	OFA-Net	**92.62** %	**83.12** %	**88.97** %	**96.58** %	3.90 %	**3.42** %
	MixedCRF	85.69 %	75.12 %	80.17 %	92.02 %	7.42 %	7.98 %
	ALO-AVG-MM	89.45 %	79.87 %	85.40 %	93.90 %	5.23 %	6.10 %
	HybridCRF	88.53 %	80.79 %	86.41 %	90.76 %	4.65 %	9.24 %
	HID-LS	89.81 %	82.33 %	88.11 %	91.58 %	4.03 %	8.42 %
URBAN_ROAD	OFA-Net	**93.74** %	85.37 %	90.36 %	**97.38** %	5.72 %	**2.62** %
	MixedCRF	90.59 %	84.24 %	89.11 %	92.13 %	6.20 %	7.87 %
	ALO-AVG-MM	92.03 %	85.64 %	90.65 %	93.45 %	5.31 %	6.55 %
	HybridCRF	90.81 %	86.01 %	91.05 %	90.57 %	4.90 %	9.43 %
	HID-LS	93.11 %	87.33 %	92.52 %	93.71 %	4.18 %	6.29 %

* The results surpass other models are labeled in bold. Lower is better in the last two columns. Higher is better in other columns.

Table 4. Meaning of the indices in Table 3 *.

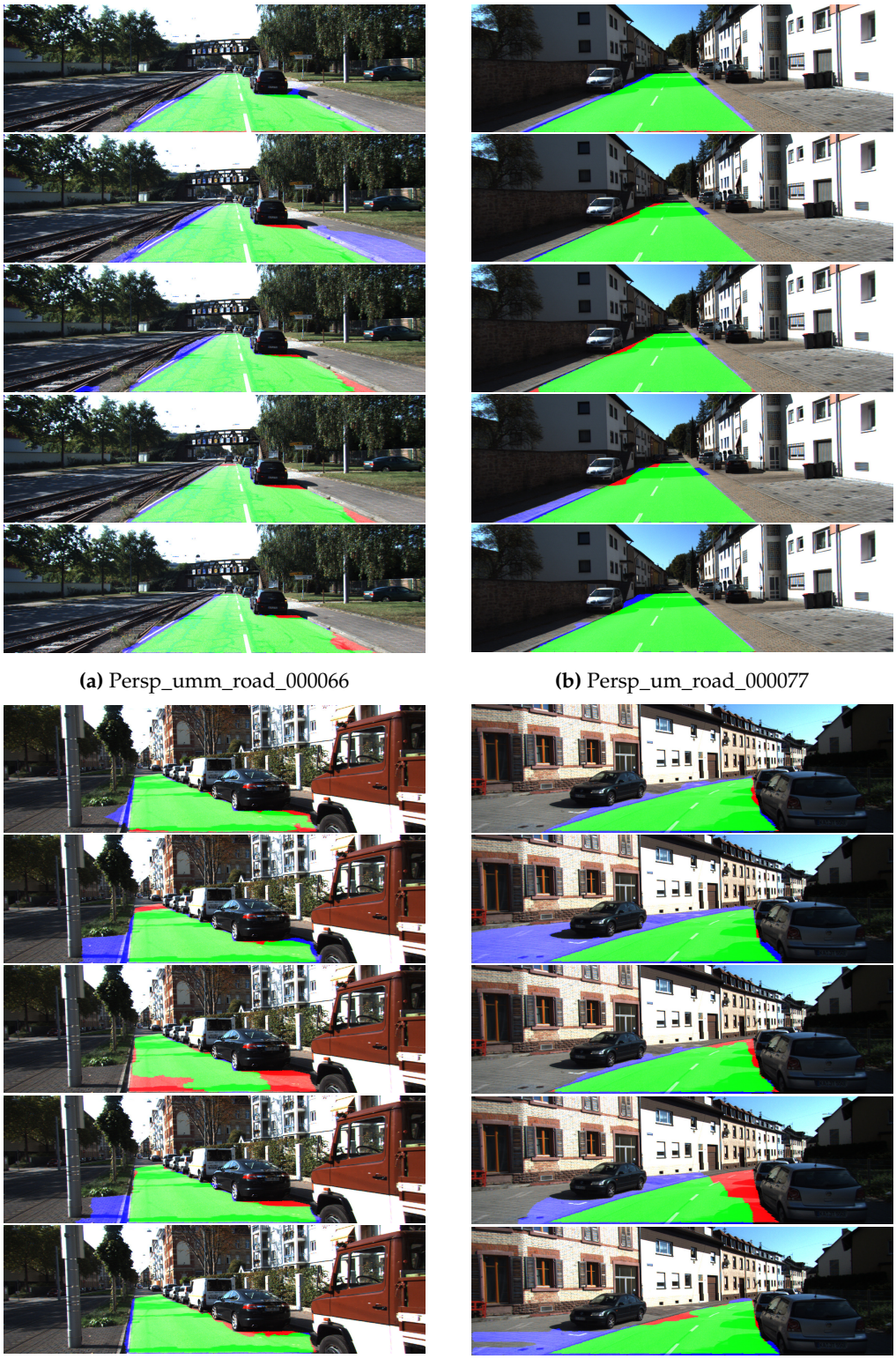
Index	Meaning
MaxF	Maximum F1-measure
AP	Average precision as used in PASCAL VOC challenges
PRE	Precision
REC	Recall
FPR	False positive rate
FNR	False negative rate

* The four latter measures are evaluated at the working point MaxF.

The OFA-Net model surpasses other models on the index of **MaxF** for the UMM_ROAD, UU_ROAD, and URBAN_ROAD benchmarks. To be specific, for the **MaxF** index, the **MaxF** value of the OFA-Net model is 0.54% higher on the UMM_ROAD benchmark, 2.81% higher on the UU_ROAD benchmark and 0.63% higher on the URBAN_ROAD benchmark than the second-best model in the five models listed. The OFA-Net model beats all other 4 models listed in Table 3 on the index of **REC**. Specifically, for the **REC** index, the **REC** value of the OFA-Net model is 2.39% higher on the UM_ROAD benchmark, 3.84% higher on the UMM_ROAD benchmark, 2.68% higher on the UU_ROAD benchmark, and 3.67% higher on the URBAN_ROAD benchmark than the second-best model in the five models mentioned above.

For a better understanding of the advantages of the OFA-Net model on segmentation tasks, there are four groups of comparisons displayed in Figure 4. Each group has the outputs of the OFA-Net model, the MixedCRF model, the ALO-AVG-MM model, the HybridCRF model, and the HID-LS model from top to bottom, respectively. Here, red areas denote false-negatives, blue areas correspond to false-positives and green areas represent true positives. What can be seen from Figure 4 is that the outputs of the OFA-Net model have fewer red areas compared with others, which means, the OFA-Net model can get lower false negative rate. This is in line with the last column of Table 3. In this paper, false-negatives indicate that the road area cannot be recognized (the road area is identified as non-road

area). The OFA-Net model here has a very low false-negative rate, thus has a very good ability to recognize road areas, which is helpful for autonomous driving.



(a) Persp_umm_road_000066 **(b)** Persp_um_road_000077

(c) Persp_uu_road_000082 **(d)** Persp_um_road_000095

Figure 4. Visualization of comparisons with other models (each group has the outputs of OFA-Net, MixedCRF, ALO-AVG-MM, HybridCRF, and HID-LS from top to bottom, respectively. Find the meaning of different road types from the research made by Fritsch, J., etc. [24].)

The object detection and pixel-wise semantic segmentation were not tested on the server because just several common categories of them were used to explore the relationships between segmentation and object detection tasks and to find a new training strategy that both tasks can benefit from. For a better intuitive feeling of the object detection effects of the OFA-Net model, some visualizations of the OFA-Net outputs are displayed in Figure 5.

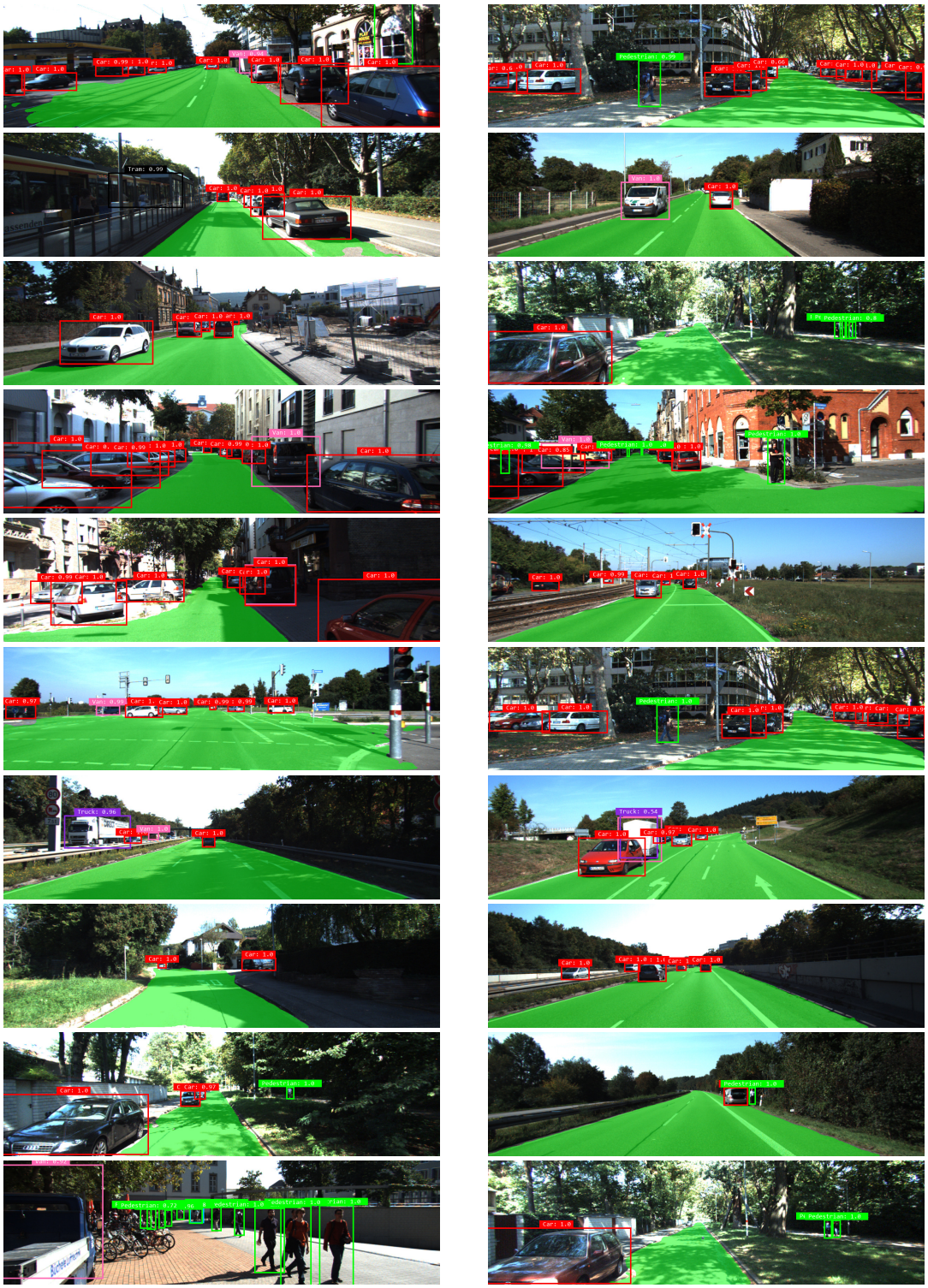


Figure 5. Visualization of the OFA-Net outputs.

6. Conclusions

In this work, a ConvNet model named OFA-Net was designed and a "1-N Alternation" strategy was created to train the OFA-Net. During which the relationships between object detection tasks and semantic segmentation tasks were explored, and the mutual enhancement effects between them were also found. Due to the segmentation task always lacking data, the findings in this paper pointed out that people can use some easily obtained object detection data as the supplementary data source to augment segmentation datasets; vice versa, the segmentation data is helpful to the detection task. What is more, the "1-N Alternation" strategy can do a feature fusion job in the OFA-Net model and can also be recruited as a regularization methodology. The OFA-Net model can get more accurate segmentation/detection results, converge more quickly, and achieve lower false negative rates, thus performing much better than the other models. In summary, the work described in this paper explored things behind related learning tasks and created a multi-task learning model named OFA-Net. The OFA-Net works well on the KITTI datasets. Furthermore, our team next is going to replace the feature extractor module with the MobileNet [51] and adapt this work to low-power devices.

Author Contributions: conceptualization, S.Z. and Z.Z.; methodology, S.Z.; software, S.Z.; validation, S.Z. and Z.Z.; formal analysis, S.Z.; investigation, S.Z.; resources, S.Z. and Z.Z.; data curation, S.Z.; writing—original draft preparation, S.Z.; writing—review and editing, L.S.; visualization, S.Z.; supervision, W.Q.; project administration, W.Q.; funding acquisition, W.Q. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the Key Research Plan of Jiangsu Province under the Grant BE2017035 and BE2019311.

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

References

1. Krizhevsky, A.; Sutskever, I.; Hinton, G. ImageNet Classification with Deep Convolutional Neural Networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
2. Sermanet, P.; Eigen, D.; Zhang, X.; Mathieu, M.; Fergus, R.; LeCun, Y. Overfeat: Integrated recognition, localization and detection using convolutional networks. *arXiv* **2013**, arXiv:1312.6229.
3. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
4. 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, Las Vegas, USA, 27–30 June 2016; pp. 770–778.
5. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
6. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. In *European Conference on Computer Vision*; Springer: Zurich, Switzerland, 2014; pp. 346–361.
7. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June2014; pp. 580–587.
8. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1440–1448.
9. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems, Montreal, QC, Canada, 7–12 December 2015; pp. 91–99.
10. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 27–30 June 2016; pp. 779–788.
11. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. *arXiv* **2017**, arXiv:1612.08242.
12. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In *European Conference on Computer Vision*; Springer: Amsterdam, Netherlands, 2016; pp. 21–37.
13. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.

14. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
15. Badrinarayanan, V.; Kendall, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv* **2015**, arXiv:1511.00561.
16. Yu, F.; Koltun, V. Multi-scale context aggregation by dilated convolutions. *arXiv* **2015**, arXiv:1511.07122.
17. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv* **2014**, arXiv:1412.7062.
18. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef]
19. Lin, G.; Milan, A.; Shen, C.; Reid, I. RefineNet: Multi-path Refinement Networks for High-Resolution Semantic Segmentation. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017.
20. Zhao, H.; Shi, J.; Qi, X.; Wang, X.; Jia, J. Pyramid scene parsing network. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017; pp. 2881–2890.
21. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
22. Chen, L.C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-decoder with atrous separable convolution for semantic image segmentation. *arXiv* **2018**, arXiv:1802.02611.
23. Alhaija, H.A.; Mustikovela, S.K.; Mescheder, L.; Geiger, A.; Rother, C. Augmented reality meets computer vision: Efficient data generation for urban driving scenes. *Int. J. Comput. Vis.* **2018**, *126*, 961–972. [CrossRef]
24. Fritsch, J.; Kuehnl, T.; Geiger, A. A new performance measure and evaluation benchmark for road detection algorithms. In Proceedings of the 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, Netherlands, 6–9 October 2013; pp. 1693–1700.
25. Perez, L.; Wang, J. The effectiveness of data augmentation in image classification using deep learning. *arXiv* **2017**, arXiv:1712.04621.
26. Wong, S.C.; Gatt, A.; Stamatescu, V.; McDonnell, M.D. Understanding data augmentation for classification: When to warp? *arXiv* **2016**, arXiv:1609.08764.
27. Zeiler, M.D.; Fergus, R. Visualizing and understanding convolutional networks. In *European Conference on Computer Vision*; Springer: Zurich, Switzerland, 2014; pp. 818–833.
28. Geiger, A.; Lenz, P.; Urtasun, R. Are we ready for autonomous driving? The kitti vision benchmark suite. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, 16–21 June 2012; pp. 3354–3361.
29. Ruder, S. An overview of multi-task learning in deep neural networks. *arXiv* **2017**, arXiv:1706.05098.
30. Uijlings, J.R.; Van De Sande, K.E.; Gevers, T.; Smeulders, A.W. Selective search for object recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
31. Farabet, C.; Couprie, C.; Najman, L.; LeCun, Y. Learning hierarchical features for scene labeling. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1915–1929. [CrossRef]
32. Shotton, J.; Johnson, M.; Cipolla, R. Semantic texton forests for image categorization and segmentation. In Proceedings of the 2008 IEEE Conference on Computer Vision and Pattern Recognition, Anchorage, AK, USA, 23–28 June 2008; pp. 1–8.
33. Brostow, G.J.; Shotton, J.; Fauqueur, J.; Cipolla, R. Segmentation and recognition using structure from motion point clouds. In *European Conference on Computer Vision*; Springer: Marseille, France, 2008; pp. 44–57.
34. Sturgess, P.; Alahari, K.; Ladicky, L.; Torr, P.H. Combining appearance and structure from motion features for road scene understanding. In Proceedings of the BMVC-British Machine Vision Conference, BMVA, London, UK, 7–10 September 2009.
35. Ladickỳ, L.; Sturgess, P.; Alahari, K.; Russell, C.; Torr, P.H. What, where and how many? Combining object detectors and crfs. In *European Conference on Computer Vision*; Springer: Heraklion, Greece, 2010; pp. 424–437.
36. He, K.; Zhang, X.; Ren, S.; Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 11–18 December 2015; pp. 1026–1034.

37. Donahue, J.; Jia, Y.; Vinyals, O.; Hoffman, J.; Zhang, N.; Tzeng, E.; Darrell, T. Decaf: A deep convolutional activation feature for generic visual recognition. In Proceedings of the International Conference on Machine Learning, Beijing, China, 22–24 June 2014; pp. 647–655.
38. Gupta, S.; Girshick, R.; Arbeláez, P.; Malik, J. Learning rich features from RGB-D images for object detection and segmentation. In *European Conference on Computer Vision*; Springer: Zurich, Switzerland, 2014; pp. 345–360.
39. Zhang, Y.; Yang, Q. A survey on multi-task learning. *arXiv* **2017**, arXiv:1707.08114.
40. Wu, B.; Nevatia, R. Simultaneous object detection and segmentation by boosting local shape feature based classifier. In Proceedings of the 2007 IEEE Conference on Computer Vision and Pattern Recognition—CVPR'07, Minneapolis, MN, USA, 17–22 June 2007; pp. 1–8.
41. Yao, J.; Fidler, S.; Urtasun, R. Describing the scene as a whole: Joint object detection, scene classification and semantic segmentation. In Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, MN, USA, 16–21 June 2012; pp. 702–709.
42. Hariharan, B.; Arbeláez, P.; Girshick, R.; Malik, J. Simultaneous detection and segmentation. In *European Conference on Computer Vision*; Springer: Zurich, Switzerland, 2014; pp. 297–312.
43. Mishkin, D.; Matas, J. All you need is a good init. *arXiv* **2015**, arXiv:1511.06422.
44. Yosinski, J.; Clune, J.; Bengio, Y.; Lipson, H. How transferable are features in deep neural networks? In *Advances in Neural Information Processing Systems*; NIPS: Montreal, QC, Canada, 2014; pp. 3320–3328.
45. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
46. Han, X.; Wang, H.; Lu, J.; Zhao, C. Road detection based on the fusion of Lidar and image data. *Int. J. Adv. Robot. Syst.* **2017**, *14*. [CrossRef]
47. Reis, F.A.; Almeida, R.; Kijak, E.; Malinowski, S.; Guimarães, S.J.F.; do Patrocínio, Z.K. Combining convolutional side-outputs for road image segmentation. In Proceedings of the 2019 International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 14–19 July 2019; pp. 1–8.
48. Xiao, L.; Wang, R.; Dai, B.; Fang, Y.; Liu, D.; Wu, T. Hybrid conditional random field based camera-LIDAR fusion for road detection. *Inf. Sci.* **2017**, *432*, 543–558. [CrossRef]
49. Gu, S.; Zhang, Y.; Yuan, X.; Yang, J.; Wu, T.; Kong, H. Histograms of the Normalized Inverse Depth and Line Scanning for Urban Road Detection. *IEEE Trans. Intell. Transp. Syst.* **2019**, *20*, 3070–3080. [CrossRef]
50. One For All [OFA]. 2019. Available online: http://www.cvlibs.net/datasets/kitti/eval_road_detail.php?result=afb1781c3a8b02a110dc5ff6ce5f18ffbbe041d9 (accessed on 16 March 2019).
51. 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, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.

© 2019 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (http://creativecommons.org/licenses/by/4.0/).