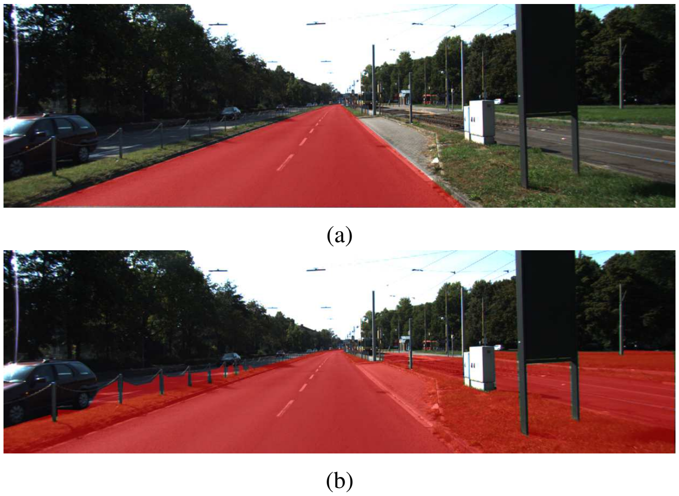
Self-Paced Cross-Modality Transfer Learning for Efficient Road Segmentation

Weiyue Wang[1], Naiyan Wang[2], Xiaomin Wu[3], Suya You[1] and Ulrich Neumann[1]

Abstract— **Accurate road segmentation is a prerequisite for autonomous driving. Current state-of-the-art methods are mostly based on convolutional neural networks (CNNs). Nevertheless, their good performance is at expense of abundant annotated data and high computational cost. In this work, we address these two issues by a self-paced cross-modality transfer learning framework with efficient projection CNN. To be specific, with the help of stereo images, we first tackle a relevant but easier task, i.e. free-space detection with well developed unsupervised methods. Then, we transfer these useful but noisy knowledge in depth modality to single RGB modality with self-paced CNN learning. Finally, we only need to fine-tune the CNN with a few annotated images to get good performance. In addition, we propose an efficient projection CNN, which can improve the fine-grained segmentation results with little additional cost. At last, we test our method on KITTI road benchmark. Our proposed method surpasses all published methods at a speed of 15fps.**


(a)

(b)

Fig. 1. (a) Example image with annotated road region. (b) Example image with annotated free-space region. Note that the segmented areas are shown in red.

I. INTRODUCTION

Road segmentation (Fig. 1a) refers to the task of labeling road regions from outdoor images, which is a crucial task in computer vision and robotics field. It has extensive applications such as autonomous driving [14] and mobile robot monocular vision navigation [29]. Such systems require high robustness, high efficiency and high accuracy to adapt various complex road patterns. Thanks to the advancement of deep learning, especially convolutional neural network (CNN), the performance of semantic segmentation has significantly improved over the last few years [28], [8], [42]. A key benefit of deep learning methods is its ability to extract complex, high-level features from massive training data. CNNs are also widely used for road segmentation and achieve great success [1], [35], [26].

Large-scale dataset is necessary for deep learning. However, manually annotating segmentation labels is time-consuming and cumbersome. For example, it was reported that it took a human labeler 90 minutes per image to annotate high-quality semantic labels for Cityscapes dataset [10], [41]. Such heavy labor prohibits building large-scale datasets for semantic segmentation. Consequently, popular image segmentation datasets [12], [10], [7] only consist of hundreds or thousands of images, which is at least two orders of magnitude smaller than other visual recognition tasks, such as image classification or object detection dataset [11]. Thus, for a special yet important segmentation task – road segmentation, *can we utilize its own characteristics to reduce the labor of human annotation?* Our answer is affirmative.

To achieve such goal, we need to introduce another closely related task – free-space detection (Fig. 1b), which segments the drivable space (or non-obstacle space) from images. Road segmentation aims at partitioning road regions, while free-space detection segments the flat drivable regions (cf. Fig. 1a and Fig. 1b). Though some areas are drivable, vehicles are not allowed in these areas based on the traffic rules or conventions (e.g. the grass in Fig. 1). Different from road, free-space is not related to the semantic meaning of an area, researchers have been dedicated to develop unsupervised methods [4], [3], [25] for this task in the last decade. Among those, most state-of-the-art methods utilize depth information from stereo images to estimate obstacle positions, and liberate human labors from annotating training labels.

In this work, we address the task of robust and accurate road segmentation from single image in real-time without large scale manually-labeled training data. In particular, we first harness stereo-based unsupervised free-space detection methods to generate large amount of noisy labels without human intervention. Although these results are not accurate, they still provide rich information of the scene that resides in the stereo images. The proposed framework essentially transfers the knowledge of ground plane within the depth

[1] Authors are with the Computer Graphics and Immersive Technologies (CGIT) laboratory of Computer Science department, University of Southern California, CA, USA. {weiyuewa;suya;uneumann}@usc.edu

[2] Author is with TuSimple. winsty@gmail.com

[3] Author is with the Department of Electrical Engineering, University of Petroleum (East China), Shandong, China. xiaominwu@s.upc.edu.cn

This work was done when W. Wang and X. Wu were interns at TuSimple.

estimation to a single RGB image. To combat with the label noise, we incorporate self-paced learning technique [24] into CNN training. It can significantly alleviate the influence of inaccurate supervision. For the choice of network structure, we modify the original ENet [36] with additional cross-block shortcuts and expanded dilated convolution [45]. These simple modifications are proven to effectively recover details of the segmentation results without resorting to computationally expensive post-processing methods, such as Conditional Random Field (CRF)[8]. Last but not least, our learning framework for road segmentation is not only limited to the network structure we use, but also compatible for all semantic segmentation models such as Fully Convolution Network (FCN) [28], DeepLab [8], SegNet [5], etc. This property makes our framework more appealing since it can leverage all future advancements in semantic segmentation.

To summarize, our contributions of this paper are as follows:

1) We design a cross-modality transfer learning framework for road segmentation. Within the framework, we only need several hundreds of annotated images to achieve superior performance.
2) We devise a robust CNN learning scheme by applying self-paced learning technique to image segmentation problem.
3) We modify the original ENet [36] structure with additional shortcuts and expanded dilated convolution to retain fine-grained details while keeping the computational efficiency.
4) Our proposed method achieves the best results among all published methods in the KITTI road segmentation dataset while running at 15fps.

The remainder of this paper is organized as follows. Section II reviews the related works. Section III presents the details of our proposed methods. Section IV shows the experiment results. Section V concludes the paper.

II. Related Works

A. Convolutional Neural Networks for Scene Parsing

Deep learning, especially Convolutional Neural Networks (CNNs) leads the evolution of computer vision in recent years. In contrast to traditional methods that manually design different features for different tasks, CNN learns the most suitable features by multiple layers of non-linear transformations. As for scene labeling or semantic segmentation task, FCN[28] is the first work to apply CNN to image segmentation. FCN employs in-network upsampling to enable pixel-wise prediction, which makes end-to-end learning and fast prediction feasible. Following this pioneer work, a series of CNN based image segmentation works [15], [8], [34], [47] are proposed. To improve the efficiency, Paszke et al. [36] proposed an efficient CNN (ENet) for real-time semantic segmentation. They deliberately designed the network structure as in ResNet [17], and used various variants of convolution operations to reduce number of parameters. Due to the real-time requirement, we choose ENet as our baseline model in this paper.

B. Self-paced Learning

Self-paced learning (SPL) [24] has attracted increasing attention from researchers in machine learning and computer vision. SPL is built on the intuition that rather than training on all samples simultaneously, the algorithm should learn the data in the order of difficulty. Just like human learning procedure, SPL learns in a self-controlled pace from easy samples to hard ones. Effectiveness of SPL to overcome training data outliers has been shown in many computer vision tasks, such as multimedia event detection [21], object detector adaptation [43], long-term tracking [19]. However the underlying mechanism of SPL remained unknown for a long time until Meng et al. [32] provided theoretical justification for SPL: SPL can be treated as optimization of a robust loss function. There are also several works trying to employ SPL in deep learning [40], [22]. However, no prior work has applied SPL to dense prediction problem (e.g. semantic segmentation) to the best of our knowledge.

C. Free-space Detection

Free space detection [25], [6] aims to estimate the non-obstacle area (a.k.a drivable space) of a scene. It defines the area that a robot or an autonomous vehicle could reach without collision.

By definition, it is nature to use depth estimation to find the continuous and flat area of a scene since the 3D information is available. The representative work of free-space detection from depth is called StixelWorld [4]. They first used stereo disparity map to build occupancy grids where each cell represents the probability of a grid is occupied. Then dynamic programming is used to find the optimal path that segments the image into free-space and obstacle area. Many subsequent works[6], [39] incorporated RGB information to reduce the error of inaccurate disparity computation. However, such methods still fail in the cases like dramatic illumination change due to the lack of context information.

There are also works to apply CNN to solve this task. Sanberg et al. [38] proposed a CNN based method with self-supervised learning and online training for free-space detection. Levi et al. [27] converted the problem to a regression problem using CNN by finding the split point of free-space and obstacle for each column of an image. Though they both share the idea of automatic label generation from depth, their performances are seriously degraded by the unsatisfied quality of labels.

D. Road Segmentation

Road segmentation is highly related to free-space detection except that road segmentation considers the additional semantic meaning of an area. For example, the grassland in Fig. 1b is drivable, but according to traffic rule, we cannot drive on it. Additional semantic constraints make road segmentation even harder than free-space detection.

Conventional methods [44], [23], [2], [46] used hand-crafted features to capture geometrical characteristics (e.g. road edge, vanishing point [23] and texture [46]) of roads to solve the problem. These methods have limitations in adapting various road conditions and complex environments. If the system meets new road patterns, new features and constraints need to be manually designed.

Not surprisingly, CNN based semantic parsing methods have been widely used in road segmentation. Mohan [33] first proposed a deep deconvolution network for this task. This method first divides an image into several regions, and then each region is trained with a separate network, which is both time and memory inefficient. Moreover, this method requires fixed-size input, and is not trained in an end-to-end manner. Subsequent works tackled these drawbacks by fully convolutional networks. Mendes et al. [31] proposed a network-in-network architecture for fast road detection; Oliveira et al. [35] proposed an efficient FCN model for road detection. They adopted VGG [42] as encoder and use a U-Net architecture[37] to increase the network's capability recovering from substantial dimension reduction in encoder.

There are also some other works that utilize existing methods for weak labels generation. Alvarez et al. [1] first trained a classifier with manually designed features on a small labeled training dataset, and then used this classifier to generate weak labels for CNN training on large unlabeled data. Laddha et al. [26] proposed a map-supervised approach to detect road. They used localization sensors and map data to generate noisy road labels, and adopted K-means to refine the noisy labels. However, the label noise issue degenerates the performance of these methods significantly. Moreover, they both generated the labels in the same modality (RGB image) of subsequent CNN training, which may lead the algorithm step in the pitfalls of hard cases in RGB images. Diversity in information modality is needed to ensure a good performance.

Above all, the performance of road segmentation has been skyrocketed with the rapid development of CNN based image segmentation technique. However, designing a road segmentation method with both high efficiency and high performance is still challenging. Moreover, the need of large-scale labeled data hamper the further improvement of performance. Both these issues call for a deliberately designed training pipeline and a network structure to incorporate the prior knowledge of road.

III. Methods

In this section, we introduce our transfer learning framework for road segmentation. The framework of our proposed approach is shown in Fig. 2. An unsupervised free-space detection method first generates the noisy training labels. Then, our proposed efficient projection CNN is trained on these noisy labels with self-paced learning. Lastly, the pre-trained CNN is fine-tuned on a few annotated images for road segmentation with normal training to adapt the difference between free-space detection and road segmentation. In testing phase, the images from single camera are fed to the CNN to get the final prediction.

A. Coarse Label Generation from Unsupervised Method

Firstly, an off-the-shelf algorithm is used to generate training labels. Specifically, we adopt Extended StixelWorld [39]. The inputs of this algorithm is a RGB image and a disparity map, which can be obtained by LiDAR or computed from stereo images. In our implementation, we use DispNet [30] to generate the disparity map from a pair of stereo images.

B. Pre-training on Efficient Projection CNN

After label generation, the left eye image is used as input to train our segmentation CNN. The benefit of pre-training the CNN with the label generated by stereo based free-space detection algorithm are in two folds: First, we transfer the rich structure knowledge of the scene resides in disparity map to single RGB image. Second, [39] only relies on local features to generate free space. It can not utilize the context information either within the same image or across images. For example, it can hardly distinguish road and sidewalk. However, if the whole image and stereo-based machine-generated label are used to train a CNN, the knowledge within RGB and stereo images are condensed into the weights of CNN. Moreover, the weights are shared across different images.

Our proposed network architecture is based on ENet [36]. ENet adopts the residual connection block of ResNet [17] as the basic block (Fig. 4b) and refers this block as bottleneck module. Each bottleneck block has a bypass branch and an main branch with convolution to learn residual. This bottleneck ensures that layers in early stage can receive gradient directly from layers in later stage. This structure facilitates the optimization of ultra deep network, so ResNet can stack even deeper than 200 layers [18].

The network architecture is presented in Table I. It is divided into several levels as separated by horizontal lines, and each level is further divided into blocks. The first three levels are encoders to condense the semantic information, while the last three levels are decoders to expand the features to final prediction. As presented in Fig. 4, there are three block types: initial, bottleneck, and projection. The initial block (Fig. 4a) is the same as in ENet. Batch Normalization [20] and PReLU [16] are placed between convolutions. The last level of the network is a single 3×3 deconvolution layer with stride 2. The final output has C feature maps, where C is the number of object classes.The difference between our network and original ENet mainly lies in the following two aspects:

First, we exploit the use of shortcut connections across different levels to recover fine-grained details. The first two stages of ENet rapidly downsample the feature maps. Although this scheme accelerates the computation, it loses the fine-grained details. Therefore it leads to over-smooth results. As is shown in Fig. 5a, ENet cannot delineate road boundaries accurately. A common approach to address such issue is to use CRF [8]. However, it introduces extra computational cost, which is undesirable for real-time prediction. Therefore, we add shortcuts from lower layers to higher layers directly. This design helps the network to propagate the detailed

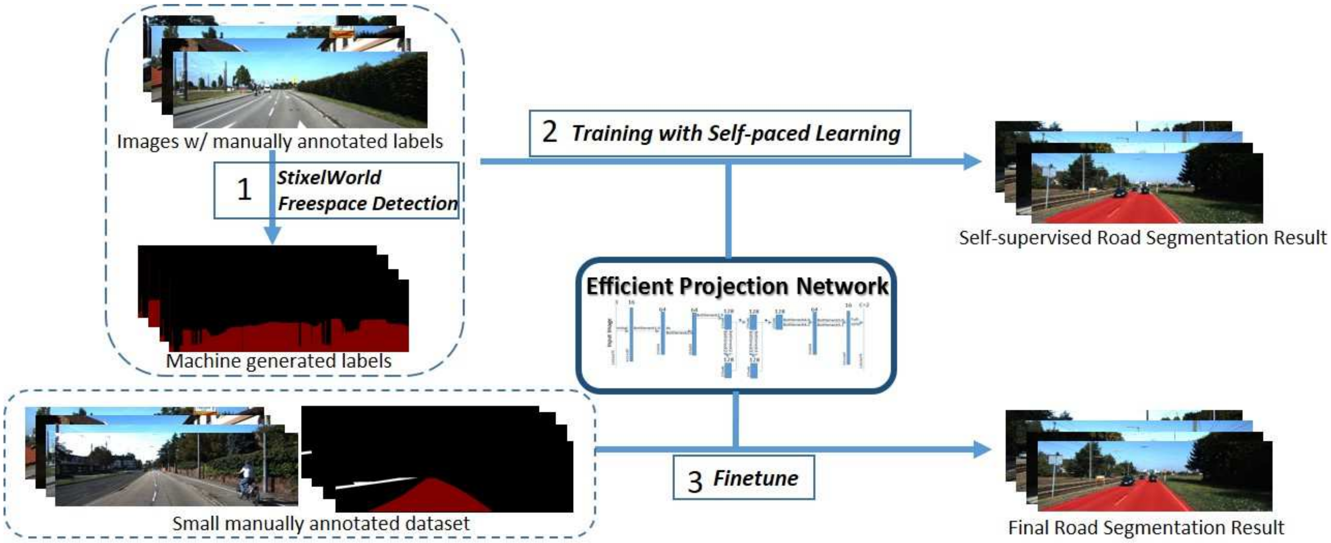

Images w/ manually annotated labels
1 StixelWorld Freespace Detection
Machine generated labels
2 Training with Self-paced Learning
Self-supervised Road Segmentation Result
Efficient Projection Network
Small manually annotated dataset
3 Finetune
Final Road Segmentation Result

Fig. 2. Framework of road segmentation approach with efficient projection network and self-paced learning.

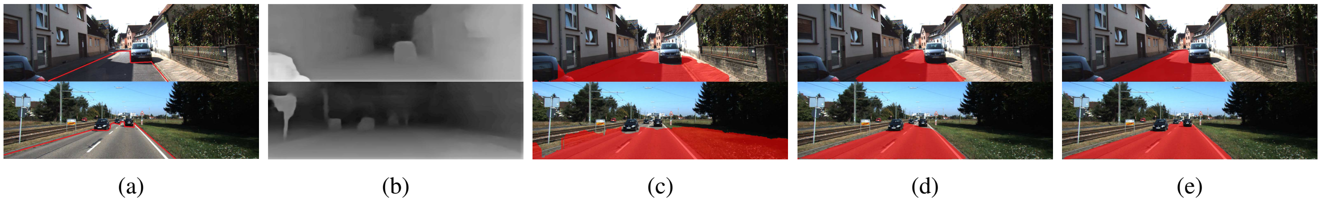

(a) (b) (c) (d) (e)

Fig. 3. Illustration of our framework. (a) Input images (The red line indicates the boundary for road and other regions). (b) Disparity estimation with DispNet [30]. (c) Unsupervised free-space detection with Extended StixelWorld [39]. (d) Road segmentation learned from free-space with self-paced learning. (e) Final road segmentation result after fine-tuning on labeled dataset.

features in low level layers to high level layers. We call our proposed network efficient projection network. As shown in Fig. 4c, our projection operation is conducted by element-wise addition of a main branch and a projection shortcut by 1 × 1 convolutions. Projection is only added on encoder levels (level 1, level 2 and level 3) after bottleneck1.4, 2.10 and 3.10.

Second, since road usually spans over large portion of the image, it needs larger receptive field to distinguish the semantic meaning of an area. For example, in order to distinguish road and sidewalk, the network even needs the global layout of the scene in the image. Consequently, we increase the receptive field by an additional dilation convolution[45]. We add bottleneck2.10 and bottleneck3.10 with dilation 32 along with regular 3 × 3 convolution bottleneck2.9 and bottleneck3.9 to expand the receptive field. As illustrated in Fig. 5, these two modification indeed improves the localization ability of the network and succeeds in recovering road boundaries.

Other differences between our proposed network and the original ENet include: 1) When dimension matching is needed in bottleneck block, we use 1 × 1 convolution to match them instead of using zero padding. 2) In original ENet, the number of filters of the convolution layer between two 1 × 1 convolution in bottleneck block is one quarter of that in these 1 × 1 convolution layers. This design severely impairs the results since the number of filters of ENet has already significantly decreased compared with ResNet. So we keep the number of filter in these three layers same in our network.

C. Self-paced Learning

The core idea of SPL is to gradually include samples from easy to hard in the training process. Then the model may learn toward the majority of the labels, and tend to ignore the outliers in the labels. Specifically, SPL reweights each sample according to its loss value during training. Formally, for training data $\{(x_i, y_i)\}_{i=1}^{n}$, x_i, y_i represent the i-th sample and i-th label, respectively. Let $L(y_i, g(x_i, \mathbf{w}))$ represent the loss function, in which $g(x_i, \mathbf{w})$ is the prediction from the model with weights $\mathbf{w}$. Suppose λ is an age parameter controlled by training iteration, $f(v; \lambda)$ is a self-paced regularizer (SP-regularizer), the main goal of SPL is to learn sample weights $v = [v_1, v_2, ..., v_n]^T$ with gradually increasing age parameter. The SPL model is composed of a weighted loss term and a self-paced regularizer term, denoted as:

$$\begin{aligned} \min_{\mathbf{v},\mathbf{w}} \quad & \sum_{i=1}^{n} (v_i L(y_i, g(x_i, \mathbf{w})) + f(v_i, \lambda)) \\ s.t. \quad & \mathbf{v} \geq 0 \end{aligned} \tag{1}$$

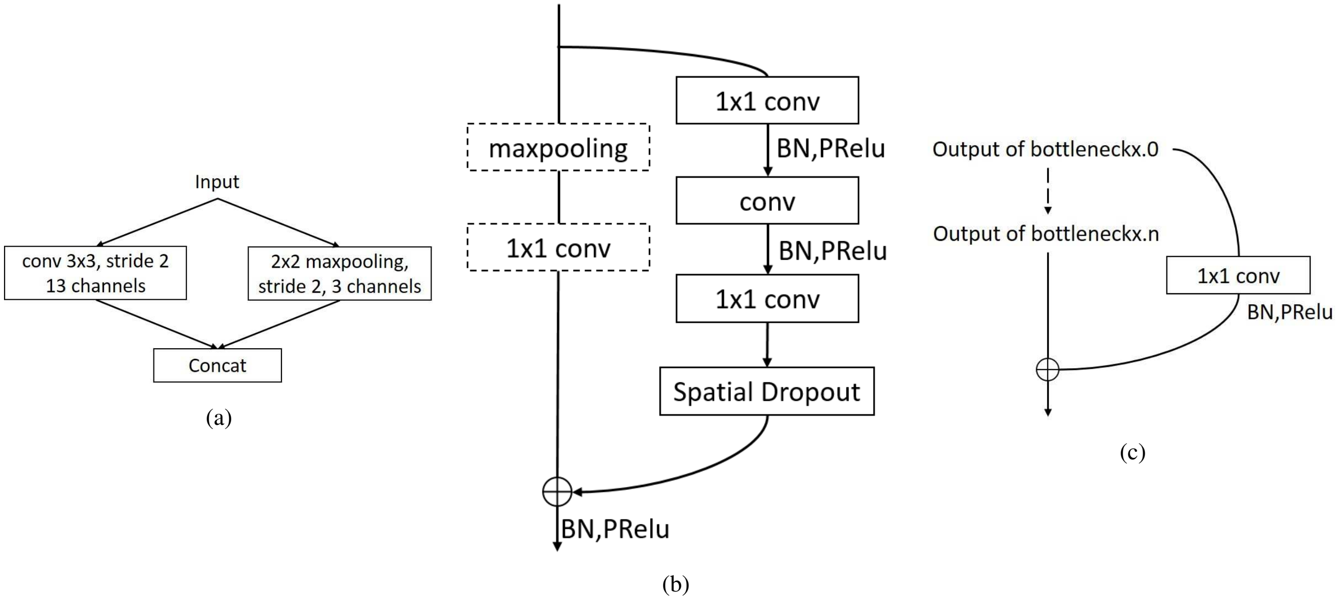


Fig. 4. (a)Initial block. (b) Bottleneck block. conv is either a regular(3×3), dilated(3×3), asymmetric(a (5×1) followed by a (1×5)) or deconvolution(3×3). (c)Projection block. 1×1 convolution is used to match dimension. Here, x denotes the level. n denotes the last block of the level.

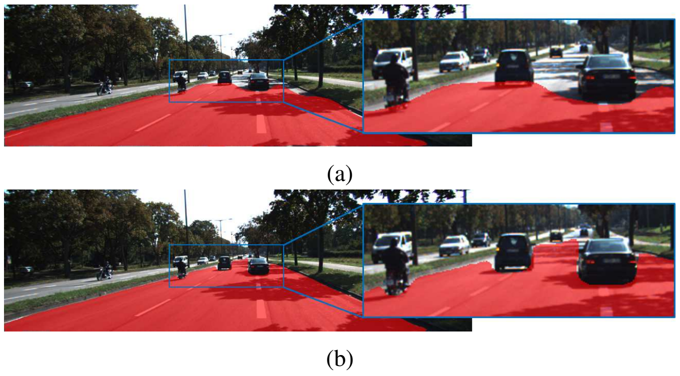


Fig. 5. (a) Segmentation result with ENet without project and expanded receptive field. (b) Segmentation result with our proposed network.

where $\mathbf{w}$ is the original model parameter, $\mathbf{v}$ is the latent sample weight, and they are jointly learned to incorporate the samples from easy to hard. To implement the idea of self-paced learning, the SP-regularizer should satisfy several properties as stated in [32]. Due to limited space, we do not repeat them here. A typical choice of SP-regularizer is called linear regularizer [21], which is defined as:

$$f(v;\lambda) = \lambda(\frac{1}{2}v^2 - v). \tag{2}$$

In our model, the function $g(\cdot)$ is the proposed efficient projection network, and the loss function $L(\cdot)$ is the last softmax layer. Since each pixel corresponds to a softmax loss, an individual sample weights is assigned for each pixel according to the corresponding loss.

For optimization of the problem, we alterate between $\mathbf{v}$ and $\mathbf{w}$. The closed form solution of $\mathbf{v}$ given $\mathbf{w}$ can be calculated as:

$$v^*(l;\lambda) = \begin{cases} -\frac{l}{\lambda}+1, & \text{if } l < \lambda \\ 0, & \text{if } l \geq \lambda \end{cases}. \tag{3}$$

The intuition behind this regularizer is that if the loss value of a sample is larger than the increasing age parameter, the sample will not be considered in training. Otherwise, its weight increases linearly as the loss value decreases. The age parameter λ is gradually increased during SPL training process to include more training samples. To optimize $\mathbf{w}$, we use back-propagation algorithm to train the network.

Fig. 6 exhibits the learning results and sample weights $\mathbf{v}$ for different training epochs. As the age parameter increasing with the training epoch, more samples are added into training. SPL effectively removes inaccurate regions in training labels by decreasing the weights of misleading labels in the supervision (darker in the figure), and increasing the weights of true road and obstacle region (brighter in the figure). As a result, the CNN inclines to learn road structures from free-space labels.

IV. Experiment

A. Dataset and Experiment Settings

KITTI [13] is a dataset designed to benchmark vision tasks for autonomous driving. We evaluate the performance of our method on KITTI road dataset, which consists of 289 training and 290 testing images. The performance is evaluated in three different categories of road scenes: single-lane road with markings (UM), single-lane road without markings (UU), multi-lane road with markings (UMM) and urban road which is defined as the average of these three. Fig. 7 shows examples of our road segmentation results for the three different categories. The performance is evaluated in bird view spacem, and the following metrics are used

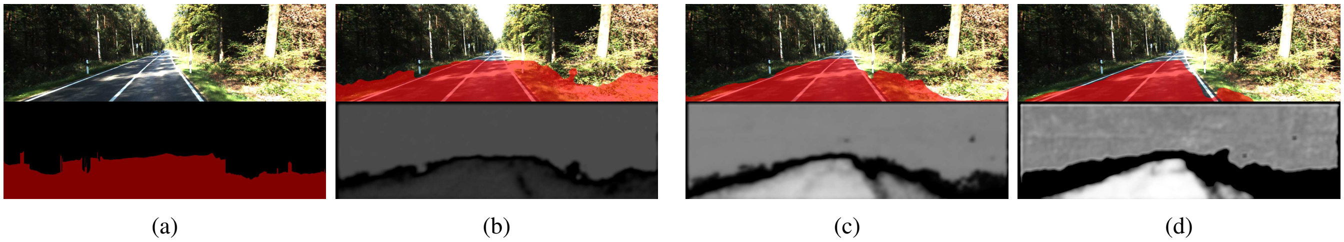


Fig. 6. Example of self-paced learning results (the first row) and the assigned sample weights **v** (the second row, the brighter the higher) with different training epoch. (a) Input image and generated noisy label. (b) Result and sample weights in the 3rd epoch. (c) Result and sample weights in the 10th epoch. (d) Result and sample weights in the 30th epoch.

TABLE I. EFFICIENT PROJECTION NETWORK ARCHITECTURE.

Name	Type	Number of Channels
Initial		16
bottleneck1.0	downsampling	64
4×bottleneck1.x		64
projection1.5		64
bottleneck2.0	downsampling	128
bottleneck2.1		128
bottleneck2.2	dilate 2	128
bottleneck2.3	asymmetric 5	128
bottleneck2.4	dilate 4	128
bottleneck2.5		128
bottleneck2.6	dilate 8	128
bottleneck2.7	asymmetric 5	128
bottleneck2.8	dilate 16	128
bottleneck2.9		128
bottleneck2.10	dilate 32	128
projection2.11		128
Repeat section 2, without bottleneck2.0		
bottleneck4.0	upsampling	64
bottleneck4.1		64
bottleneck4.2		64
bottleneck5.0	upsampling	16
bottleneck5.1		16
deconv		C

for evaluation: Maximum F1-measurement (MaxF), Average precision (AP), Precision (PRE), Recall (REC), False positive rate (FPR), and False negative rate (FNR). We show these measurements in percentage and we ignore '%' symbol in our experiment section. Results for other methods can be found on KITTI website. Methods are ranked according to their MaxF since it fuses other metrics. We also compare inference time with various methods.

Our implementation is based on MXNet [9] on NVIDIA TitanX GPU. More details and codes are available at `http://winsty.net/cmtspl_roadseg.html`

B. Implementation Details

We use all 14999 images in KITTI object detection dataset for unsupervised pre-training. Note that since we don't use the object detection labels, it is reasonable to use both training and testing dataset. After that, we fine-tune the network on KITTI road dataset (289 training images with human-annotated labels). We pre-train 60 epochs and fine-tune 1000 epochs. We use SGD with momentum of 0.9, weight decay of 0.0002 and a polynomial learning rate policy: $L_i = L_0(1 - i/n)^p$, where i is the training iterator, L_i is the learning rate of ith training epoch, $L_0 = 0.001$ is the initial learning rate, n is the maximum training epochs. We set $p = 0.9$ in our experiment. The age parameter λ is tuned as:

$$\lambda_i = 5 \times 10^{-6} i + 0.3 \tag{4}$$

where i is the training iteration, and λ_i is the age parameter in the ith iteration.

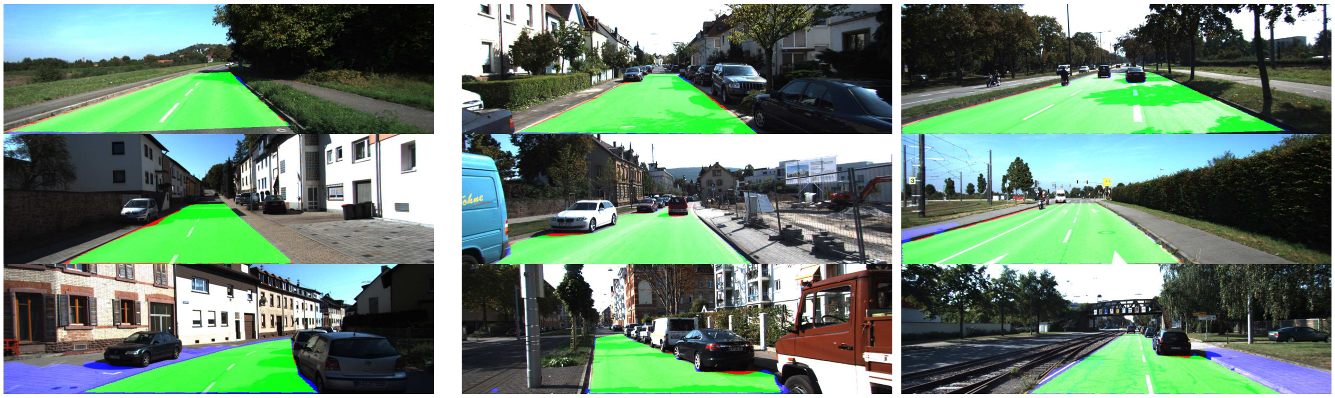
We also employ a series of data transformations to augment data: scaling (randomly scale the image by a factor between 0.9 to 1.2), mirroring (randomly flip the image horizontally), color transformation (color jittering in HSV color space, range $[-15, 15]$ for H, S, V, respectively.) and random cropping (randomly crop 320×1000 from the scaled image).

C. Self-paced Learning Results

We firstly evaluate the effectiveness of cross-modality transfer learning and the self-paced learning scheme. We compare the results of: 1) pre-training on KITTI object detection dataset with SPL and fine-tuning on KITTI road dataset, 2) the same setting with 1) without SPL, and 3) directly training KITTI road dataset with random initialization. Table II summerizes the results. Not surprisingly, SPL ranks first, then followed by the model without SPL and random initialization. In one hand, cross-modality transfer learning is able to guide the CNN to capture the rough structure of the scene. On the other hand, SPL decreases the chances for CNN to learn from misleading labels. The experiment results confirm the usefulness of cross-modality transfer learning and SPL.

TABLE II. COMPARISON OF THE EFFECT OF PRE-TRAINING AND SPL LEARNING

	MaxF	AP	PRE	REC	FPR	FNR
with SPL	**94.40**	**93.05**	**93.87**	**94.93**	**3.42**	**5.07**
w/o SPL	93.69	92.96	93.74	93.65	3.44	6.35
random init	92.72	92.78	92.76	92.68	3.98	7.32



(a) UM

(b) UU

(c) UMM

Fig. 7. Our segmentation results on different road categories. Green represents true positives, blue denotes false positives, and red denotes false negatives.

D. Comparison of different network structure

Next, we compare the results of different network structures within our framework. Note that we don't use SPL in this experiment. We test the impact of the projection operation and additional dilation while keep using convolution for dimension matching and larger number of filter for 3×3 convolution. As illustrated in Table III, the performance has been improved vastly with these two designs. The inspiring results demonstrate that our projection operation improves the performance in dense prediction task by combining fine-grained features in low levels and semantic features in high levels. In addition, the expanded receptive field by additional dilation convolutions helps to incorporate more context into prediction.

We also try to replace our efficient projection network with FCN-8s [28]. Though FCN is a more complex network with more parameters, its results are much worse than ours. This again validates the effectiveness of our designed network structure.

TABLE III. COMPARISON OF RESULTS OF DIFFERENT NETWORK STRUCTURES

	MaxF	AP	PRE	REC	FPR	FNR
ENet	93.13	**93.01**	93.15	93.12	3.77	6.88
FCN	90.89	82.32	87.00	**95.14**	7.83	**4.86**
Our model	**93.69**	92.96	**93.74**	93.65	**3.44**	6.35

E. Comparison with the state-of-art methods

We compare our method with top ranked methods submitted to KITTI website. Since most of them do not reveal the details of their methods, we only compare with those published methods. Our test results of different categories are shown in Table IV. Note that UR denotes urban roads, which is average of UM, UU and UMM. Our method exhibits significantly better results compared with other state-of-the-art methods at 15fps.[1]

F. Comparisons of Complexity of Different Methods

Table V compares the prediction time for different methods. Although these results are directly quoted from KITTI website, all these methods are run on latest GPU, so we can still roughly compare them. Our method is at the sweet point at performance and speed. Moreover, our model only has 2.4M parameters (18.8M in storage), which makes this network easy to deploy into embedded system for autonomous driving. We believe it will also benifit from more advanced compression and accleration techniques for deep learning that been developed in recent years.

V. CONCLUSIONS

In this paper, we study the problem of road segmentation for autonomous driving. A cross-modality transfer framework has been proposed to reduce human labor for segmentation label annotation. The principle behind the framework is that we transfer the rich yet inexpensive scene structure across two different modalities: from stereo images to single RGB image. Furthermore, self-paced training has been incorporated to reduce the influence of inaccurate automatic generated labels. Then, an efficient projection convolutional neural network has been further devised to achieve real-time segmentation with more accurate road boundaries. At last, our experimental results demonstrate the effectiveness and efficiency of our approaches. In particular, we rank first among all published methods in KITTI road segmentation evaluation.

REFERENCES

[1] J. M. Alvarez, T. Gevers, Y. LeCun, and A. M. Lopez. Road scene segmentation from a single image. In *ECCV*, 2012.

[2] J. Alvarez-Mozos, A. Lopez, and R. Baldrich. Illuminant-invariant model-based road segmentation. In *IEEE Intelligent Vehicles Symposium*, 2008.

[3] H. Badino, U. Franke, and R. Mester. Free space computation using stochastic occupancy grids and dynamic. In *ICCV Workshop*, 2007.

[4] H. Badino, U. Franke, and D. Pfeiffer. The Stixel world - a compact medium level representation of the 3D-world. In *DAGM Symposium on Pattern Recognition*, 2009.

[5] V. Badrinarayanan, A. Kendall, and R. Cipolla. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *arXiv preprint arXiv:1511.00561*, 2015.

[1] By 15th, Sep, 2016

TABLE IV. ROAD SEGMENTATION PERFORMANCE OF VARIOUS METHODS ON KITTI ROAD TESTING DATASET

	UM					UU					UMM					UR				
	Ours	[35]	[33]	[26]	[31]	Ours	[35]	[33]	[26]	[31]	Ours	[35]	[33]	[26]	[31]	Ours	[35]	[33]	[26]	[31]
MaxF	93.44	92.2	**93.65**	91.2	89.36	**92.97**	92.65	91.76	89.62	86.27	**95.87**	95.52	94.17	92.98	94.09	**94.40**	93.83	93.43	91.61	90.79
AP	**92.5**	88.85	88.55	90.6	78.80	**91.93**	89.2	86.84	88.93	75.37	**95.13**	92.86	92.7	92.89	90.26	**93.05**	90.47	89.67	90.96	85.83
PRE	92.81	92.57	**94.28**	91.11	89.35	92.88	92.85	**93.06**	89.1	86.65	95.64	95.37	**96.73**	91.84	94.05	93.87	94	**95.09**	91.04	90.87
REC	**94.07**	91.83	93.03	91.29	89.37	**93.07**	92.45	90.5	90.14	85.89	**96.10**	95.67	91.74	94.15	94.13	**94.93**	93.67	91.82	92.2	90.72
FPR	3.32	3.36	**2.57**	4.06	4.85	2.33	2.32	**2.20**	3.59	4.31	4.82	5.10	**3.41**	9.20	6.55	3.42	3.29	**2.61**	5.00	5.02
FNR	**5.93**	8.17	6.97	8.71	10.63	**6.93**	7.55	9.50	9.86	14.11	**3.90**	4.33	8.26	5.85	5.87	**5.07**	6.33	8.18	7.80	9.28

TABLE V. RUNNING TIME COMPARISONS WITH STATE-OF-THE-ART METHODS

	Ours	[35]	[33]	[26]	[31]
Time	70ms	80ms	2s	280ms	30ms

[6] R. Benenson, R. Timofte, and L. J. V. Gool. Stixels estimation without depth map computation. In *ICCV Workshop*, 2011.

[7] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV*, 2008.

[8] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. In *ICLR*, 2015.

[9] T. Chen, M. Li, Y. Li, M. Lin, N. Wang, M. Wang, T. Xiao, B. Xu, C. Zhang, and Z. Zhang. MXNet: A flexible and efficient machine learning library for heterogeneous distributed systems. In *NIPS Workshop*, 2015.

[10] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, and B. Schiele. The Cityscapes dataset for semantic urban scene understanding. In *CVPR*, 2016.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. Li. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009.

[12] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes challenge: A retrospective. *IJCV*, 111(1):98–136, 2015.

[13] J. Fritsch, T. Kuehnl, and A. Geiger. A new performance measure and evaluation benchmark for road detection algorithms. In *ITSC*, 2013.

[14] D. Geronimo, A. M. Lopez, A. D. Sappa, and T. Graf. Survey of pedestrian detection for advanced driver assistance systems. *TPAMI*, 32(7):1239–1258, July 2010.

[15] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Hypercolumns for object segmentation and fine-grained localization. In *CVPR*, 2015.

[16] K. He, X. Zhang, S. Ren, and J. Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, 2015.

[17] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

[18] K. He, X. Zhang, S. Ren, and J. Sun. Identity mappings in deep residual networks. In *ECCV*, 2016.

[19] J. S. Supancic III and D. Ramanan. Self-paced learning for long-term tracking. In *CVPR*, 2013.

[20] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *ICML*, 2015.

[21] L. Jiang, D. Meng, T. Mitamura, and A. G. Hauptmann. Easy samples first: Self-paced reranking for zero-example multimedia search. In *ACM MM*, 2014.

[22] L. Jiang, D. Meng, S. Yu, Z. Lan, S. Shan, and A. Hauptmann. Self-paced learning with diversity. In *NIPS*. 2014.

[23] H. Kong, J. Audibert, and J. Ponce. General road detection from a single image. *TIP*, 19(8):2211–2220, 2010.

[24] M. P. Kumar, B. Packer, and D. Koller. Self-paced learning for latent variable models. In *NIPS*, 2010.

[25] R. Labayrade and D. Aubert. Real time obstacle detection in stereovision on non flat road geometry through V-disparity representation. In *IEEE Intelligent Vehicle Symposium*, 2002.

[26] A. Laddha, M. Kocamaz, L. E. Navarro-Serment, and M. Hebert. Map-supervised road detection. In *IEEE Intelligent Vehicles Symposium*, 2016.

[27] D. Levi, N. Garnett, and E. Fetaya. StixelNet: A deep convolutional network for obstacle detection and road segmentation. In *BMVC*, 2015.

[28] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *CVPR*, 2015.

[29] A. Lookingbill, J. Rogers, D. Lieb, J. Curry, and S. Thrun. Reverse optical flow for self-supervised adaptive autonomous robot navigation. *IJCV*, 74(3):287–302, 2007.

[30] N. Mayer, E. Ilg, P. Husser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In *CVPR*, 2016.

[31] C. C. Teodoro Mendes, V. Fremont, and D. Fernando Wolf. Exploiting fully convolutional neural networks for fast road detection. In *ICRA*, 2016.

[32] D. Meng and Q. Zhao. What objective does self-paced learning indeed optimize? *arXiv preprint arXiv:1511.06049*, 2015.

[33] R. Mohan. Deep deconvolutional networks for scene parsing. *arXiv preprint arXiv:1411.4101*, 2014.

[34] H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. *arXiv preprint arXiv:1505.04366*, 2015.

[35] G. Oliveira, W. Burgard, and T. Brox. Efficient deep methods for monocular road segmentation. In *IROS*, 2016.

[36] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello. ENet: A deep neural network architecture for real-time semantic segmentation. *arXiv preprint arXiv:1606.02147*, 2016.

[37] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[38] W. P. Sanberg, G. Dubbelman, and P. H. N. de With. Free-space detection with self-supervised and online trained fully convolutional networks. *arXiv preprint arXiv:1604.02316*, 2016.

[39] W. P. Sanberg, G. Dubbelman, and P. H.N. de With. Color-based free-space segmentation using online disparity-supervised learning. In *ITSC*, 2015.

[40] E. Sangineto, M. Nabi, D. Culibrk, and N. Sebe. Self paced deep learning for weakly supervised object detection. *arXiv preprint arXiv:1605.07651*, 2016.

[41] A. Shafaei, J. J. Little, and M. Schmidt. Play and learn: Using video games to train computer vision models. *arXiv preprint arXiv:1608.01745*, 2016.

[42] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

[43] K. Tang, V. Ramanathan, F.-F. Li, and D. Koller. Shifting weights: Adapting object detectors from image to video. In *NIPS*. 2012.

[44] B. Wang, V. Fremont, and S. A. Rodriguez Florez. Color-based road detection and its evaluation on the KITTI road benchmark. In *IEEE Intelligent Vehicles Symposium Workshop*, 2014.

[45] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. In *ICLR*, 2016.

[46] J. Zhang and H-H Nagel. Texture-based segmentation of road images. In *IEEE Intelligent Vehicles Symposium*, 1994.

[47] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. Torr. Conditional random fields as recurrent neural networks. In *ICCV*, 2015.