Transferring Visual Knowledge for a Robust Road Environment Perception in Intelligent Vehicles

Wei Zhou, Roberto Arroyo, Alex Zyner, James Ward, Stewart Worrall, Eduardo Nebot and Luis M. Bergasa

***Abstract*— Vision-based urban scene recognition is capable of providing semantic information that will have a significant impact for intelligent transportation systems. Semantic segmentation can provide high-level information from images of urban scenes, but we have discovered that existing models trained on public datasets often do not adapt well to other environments. This work explores the transferability of Convolution Neural Network (CNN) features by retraining the network using a minimal dataset incorporating training data specific to the local environment. A new local dataset is manually annotated and used to train a neural network for pixel-level semantic image information. Since data annotation is time-consuming, we evaluate the transferability of CNNs and the performance of different data augmentation methods for dataset expansion. Small datasets are normally considered not sufficient for training a neural network from scratch. This paper presents an incremental fine-tunning algorithm to update the pre-trained network. The experimental results shows that it is possible to successfully transfer semantic features to a different environment by incorporating a relatively small number of local images.**

I. Introduction

Autonomous cars need to be capable of recognizing the semantic context of a local environment and navigating within this environment. This ability relies heavily on vehicle-mounted sensor systems including lasers, radars, cameras, etc. Due to their low cost and high information content, cameras have been widely used for object classification [1] and scene understanding [2] in intelligent transportation systems (ITS).

Convolutional Neural Networks (CNNs) [3] are increasing in popularity for ITS applications. This is largely due to the rapid increase in available computational power from graphics processing units (GPUs) that enable the training and real-time implementation of CNNs. Another factor is the availability of very large annotated datasets such as ImageNet [4] which has around one million images with bounding box annotations. Models trained using these advances are increasingly superior to most traditional algorithms for image classification, detection, localization and other vision-based tasks. Recent research in CNNs has explored pixel-level semantic information to provide a high-level understanding of a visual scene.

W. Zhou, A. Zyner, J. Ward, S. Worrall and E. Nebot are with the Australian Centre for Field Robotics (ACFR) at the University of Sydney (NSW, Australia). E-mails: {w.zhou, a.zyner, j.ward, s.worrall, e.nebot}@acfr.usyd.edu.au.

R. Arroyo and L. M. Bergasa are with the Department of Electronics at the University of Alcalá (Madrid, Spain). E-mails: {roberto.arroyo, bergasa}@depeca.uah.es.

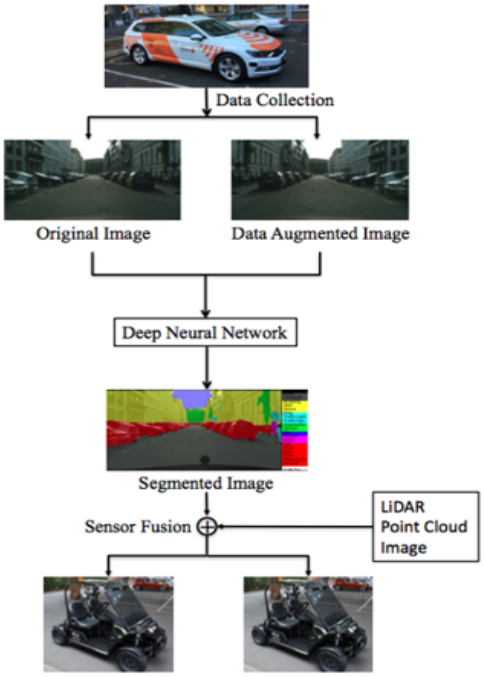


Fig. 1: Project Flowchart. The SMART City campus project involves data collection/processing, sensor fusion and electrical vehicles self driving. This paper presents the first phase of the project and demonstrates the challenges of understanding local environment for autonomous vehicles.

Inspired by the compelling performance of CNN-based feature classification, it is now possible for intelligent vehicles to process camera images and provide essential information to advanced driver assistance systems (ADAS) with the aim of giving alerts in the event of high risk scenarios [5] and to provide some level of automation. Some state-of-the-art methods such as SegNet [6], ResNet-38 [7] and PSPNet [8] have dramatically improved segmentation accuracy, and ENet [9] has claimed to achieve semantic segmentation in real-time.

The performance of pixel-level semantic segmentation is still restricted by a number of factors. As deep neural networks rely heavily on a large amount of labeled data to learn features and classify categories, the first challenge is achieving acceptable results with the limited number of segmentation datasets currently available. The most widely used CamVid dataset [10], KITTI dataset [11] and Cityscapes dataset [12] are far from complete in covering all traffic

Fig. 2: Semantic segmentation on University of Sydney campus.

environments. Another challenge is to generate a model that is compatible with other cities or regions that have local variations which are not covered by the training datasets. As the last few layers of the deep neural network architecture are designed to learn specific features, a model trained using data from a specific environment may not transfer to a different environment that varies significantly from the training data [13]. As a result of this, the performance of a model may vary widely from city to city.

In this paper, we explore the transferability of CNN features from publicly available datasets taken in other cities to our local environment. The motivation behind this is the measurably poor performance of using existing models for ITS applications when used in other regions. We demonstrate that a small amount of locally annotated images can be expanded using data augmentation to improve the model performance [3]. We also measure the influence of different data augmentation algorithms and demonstrate the improvement of neural network fine-tuned with a reasonably small dataset.

In collaboration with Ibeo Automotive Systems GmbH [14], we have retrofitted a perception vehicle with cameras and 360-degree LiDAR sensing for naturalistic driving data collection. The first phase of our project involves using this vehicle to prepare our own urban scene dataset, train models and test these in real urban scenarios. We are also working with a number of innovative electrical vehicles fitted with sensors to demonstrate self-driving algorithms and SMART city concepts at the University of Sydney (USyd). The second phase will involve fusing the available sensors information and finally implementing vehicle safety applications and autonomous driving (project flowchart shown in Fig. 1).

II. Related Work

The purpose of urban scene semantic segmentation is to help intelligent vehicles understand the high level semantic meaning of objects in real-world traffic and assist in vehicle control and safe driving applications. In this section, we provide related works in these areas.

A. CNN Architectures for Semantic Segmentation

Recent semantic segmentation algorithms are almost exclusively using deep neural networks. A basic strategy is to use a deep convolutional encoder-decoder scheme to classify a scene into a number of different categories and retrieve a segmented image which has the same size with the input image from a low-resolution encoder output. The VGG16 [15] network is one of the most popular neural networks used as an encoder to extract image features, while decoder networks for upsampling have been demonstrated using different architectures.

The encoder of the fully convolutional networks (FCNs) [16] is a pioneering architecture based on the VGG16 network. The decoder combines an upsampled feature map in the current layer with a feature map from the corresponding encoder layer to produce a new input for the next layer. The drawback of this network is the high demand of memory to store all encoder feature maps, which also restricts real-time implementations. SegNet [6], as another deep CNN architecture, has the first 13 layers topologically identical to the VGG16 network for the encoder and a symmetrical architecture for the decoder. Instead of storing all feature maps, SegNet uses max pooling indices obtained from the encoder to upsample the corresponding feature maps for decoder, which dramatically reduces the memory and computational cost required for training.

ENet [9] claims to enable the implementation of semantic segmentation in real-time. Adopting views from ResNets [17], ENet is constructed with multiple bottleneck modules which can be used for either downsampling or upsampling images. Unlike SegNet's symmetric architecture, ENet has larger encoder than decoder as it is believed the decoder is only required to fine-tune the details in the image [9]. This simplified structure allows ENet to dramatically reduce the number of parameters and save additional processing time and memory costs. Considering the real world application of intelligent vehicles, this project follows the concept of the ENet architecture for semantic segmentation.

B. Transferability of Features in Deep Neural Networks

Large annotated datasets are not always available for a given application. Using recent techniques from transfer learning [18], the CNN architecture can be first trained on a large existing dataset and then enhanced with additional set of images for a more specific task [19]. This technique was first developed for object recognition [20] and then used for both instance and semantic segmentation [21], [22].

There are two popular ways to make use of transfer learning techniques. The first one is to freeze the network as a feature extractor, remove the last fully connected layer (classifier) and attach a new classifier to train on a new dataset. The second strategy also removes the classifier, but instead of freezing the structure, it fine-tunes all the weights in the pre-trained model to better adapt to the new data. We use the second strategy by fine-tuning models trained on the publicly available datasets to fit our local environment.

C. Self-driving Cars with Deep Learning Algorithms

For vision-based autonomous driving, there are several paradigms to make a control decision that have been pro-

posed. The mediated perception approach, which consists of multiple image recognition tasks, involves parsing the camera images and providing this information to other vehicle systems for making driving control decisions [23].

Another paradigm is the behavior reflex approach which allows vehicles to directly react to an input image [23]. As deep neural networks are capable of learning models end-to-end, the system could learn to drive on local roads after building a model from the reactions of a human driver. NVIDIA Corporation [24] has trained a CNN model which directly maps the input camera image to steering commands.

Our project is based on the first paradigm but uses more precise pixel-level semantic segmentation to assign each pixel with a corresponding scene category. The colour index in Fig. 2 shows an example of the segmentation results for each category. This information is then fused with other sensors to improve the integrity of the perception process.

III. Method

Implementing real-time semantic segmentation is a trade-off between accuracy and efficiency. A concise network architecture with fewer parameters and sufficient labeled data are fundamental to train a good model. The preparation and process of training semantic segmentation models are presented in this section.

A. CNN Architecture

We start exploration with ENet architecture [9] for semantic segmentation. The crucial design insight of this network is to downsample the input image at an early stage and use only a small number of feature maps to save computational cost. The network includes an initial block which combines the results of an initial max pooling and convolution step. The second main block consists of several bottleneck modules adapted from ResNet [17]. Each bottleneck is designed to have a projection to reduce the dimension, a convolution layer and an expansion to match the dimension. The convolution layer could be regular, dilated or deconvolutional depending on the purpose. The overall architecture is shown in Fig. 3. The encoder has 22 bottlenecks and uses max pooling to downsample the images. For the decoder, 5 bottleneck modules are designed and max pooling is replaced by max unpooling to upsample the feature maps.

B. Data Preparation

We use publicly available datasets (Cityscapes [12] and CamVid [10]) to train the primary models, benchmark the performance and transfer CNN features to local areas in Sec. IV. The original Cityscapes dataset has around 30 classes while not all of them are meaningful in our scenario. Therefore, we combine and remap the original Cityscapes dataset to 12 classes including 'sky', 'building', 'pole', 'road', 'undrivable road' (sidewalk, grass, etc.), 'vegetation' (tree, hedge, etc.), 'sign symbol', 'fence', 'vehicle' (car, truck, bus, etc.), 'pedestrian', 'rider' (cyclist, motorcyclist, etc.) and 'void'.

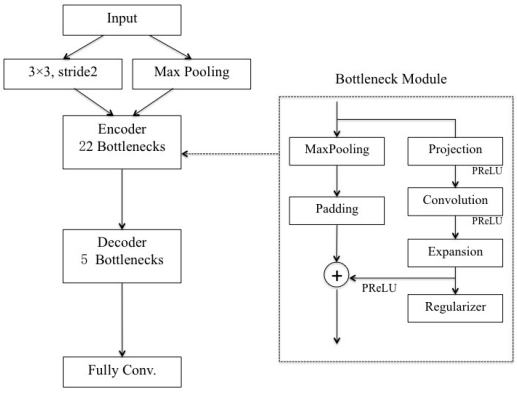


Fig. 3: CNN Architecture for Semantic Segmentation

In addition to those two datasets, we have also recorded some local traffic scenes around USyd campus. The data collection vehicle is equipped with Point Grey Blackfly cameras and 12.5mm industrial manual lenses. The selected 150 images are manually annotated using the on-line labeling tool LabelMe [25] and the annotation follows the same rule for Cityscapes category remapping. We keep the original Cityscapes image ratio but reduce the resolution to 512x256 for the model training. The input image sizes can be arbitrary since they will be rescaled during the training process.

Since data annotation is prohibitively expensive, some commonly used data augmentation algorithms are used to expand our local dataset and also minimize the occurrence of overfitting [17], [3], [15]. In this paper, we employ four types of image transformations: center cropping, left-right flipping, additive noise and Gaussian blur (shown in Figure 4) to augment the labeled data. These algorithms preserve the set of underlying classes and introduce additional examples for each class.

Instead of random cropping, we crop the center region of images so that objects in the path of the vehicle can be emphasized. Left-right flipping or mirroring changes the image structures but maintains the number of classes in the dataset. We also add Gaussian noise and apply Gaussian blur respectively to our dataset since neural networks are generally influenced by these two transformations [26].

C. Training

The original encoder and decoder of ENet were trained separately. We combine the encoder and decoder so that parameters for the entire network can be fine-tuned together on our local dataset. We also improve the way to decay the learning rate. In the original implementation, the learning rate is set to decay after a certain number of training epochs. This could induce some risks that the training stops at a local minimum or the validation accuracy has not topped off. Therefore, instead of setting a fixed number of epochs to decay learning rate, we decay it when the validation accuracy shows no improvement for N epochs. N is a hyper-parameter which can be adjusted by different batch sizes, initial learning

(a) Original Image (b) Blurred Image (c) Cropped Image (d) Flipped Image (e) Additive Noise Image

Fig. 4: Data Augmentation on USyd Dataset

	Custom Class Balancing						Median Frequency Balancing						Natural Frequency Balancing					
	Train			Test			Train			Test			Train			Test		
	G	C	IoU	G	C	IoU	G	C	IoU	G	C	IoU	G	C	IoU	G	C	IoU
SegNet	n/a	n/a	n/a	n/a	n/a	n/a	*94.3*	*95.8*	*92.0*	*83.4*	*63.6*	*48.5*	*95.3*	*80.9*	*68.9*	*84.2*	*56.5*	*47.7*
ENet*	n/a	n/a	n/a	n/a	*68.3*	*51.3*	89.0	89.9	64.8	80.8	69.8	48.3	91.1	51.5	45.9	85.9	46.8	40.8
ENetAug	93.8	90.5	76.5	86.9	72.3	57.2	91.2	92.7	71.3	83.4	75.5	53.6	94.6	82.5	75.3	88.7	63.0	55.1

TABLE I: Quantitative results trained and tested on CamVid dataset with different class balancing techniques described in subsection III-C. Values in italics are original results from SegNet [6] and ENet [9]. ENetAug is trained on augmented CamVid dataset. The performance is quantified by global accuracy (G), class average accuracy (C) and intersection over union (IoU). The results are shown as percentages.

rates, the total number of training epochs, etc.

Considering different classes occupy different portions of pixels in the dataset, both SegNet [6] and ENet [9] networks use class balancing schemes to weight each class in the loss function. The first technique is the custom class weighing scheme introduced in ENet and defined as $w_{class} = \frac{1}{ln(c+p_{class})}$ [9], where c is a hyper-parameter. This hyper-parameter is set to be 1.02 for Cityscapes dataset and 1.04 for CamVid dataset, therefore all class weights are restricted in the interval of [1, 50] [9]. The SegNet model uses median frequency balancing [27] where the per-class weight is the median of all class frequencies divided by each class's frequency. They also use natural frequency balancing which is equivalent to training without balancing class weight. The accuracy varies when using different class balancing techniques and the comparison results are shown in Section IV.

IV. Experiments and Results

The training is implemented using two NVIDIA GTX 1080 GPUs. With the image resolution of 512x256, it takes around 40ms to learn a sample in USyd dataset and the model can achieve an average of 20 fps without any optimization during inference. The initial learning rate is set to be 5e-6, the L2 weight decay is 2e-4 and the batch size keeps as 5 for all experiments. The training process adopts the idea of 5-fold cross validation and all fine-tuned results are averaged in this section.

A. Class Balancing Techniques Analysis

The first experiment is to illustrate the influence of different class balancing techniques on segmentation accuracies using CamVid dataset [10]. The results in Table I show slightly higher global accuracy when using natural frequency balancing. This phenomenon can be explained by the dominance of good estimation from large portion classes such as 'sky', 'tree' and 'road'. The class average accuracies and the intersection of union (IoU) are improved by using another two techniques as small portion categories are balanced to have higher weights in the training set. The accuracy can be further improved by augmenting the CamVid dataset, which inspires us to evaluate the performance of different data augmentations on our local dataset. The median frequency balancing is utilized for the rest of the experiments to eliminate the selection of hyper-parameter c in custom class balancing.

B. Data Augmentation Analysis

The training dataset was expanded using data augmentation as illustrated in Figure 4. The model was firstly trained on the Cityscapes dataset [12] and then fine-tuned on either original USyd dataset or augmented USyd datasets.

From Figure 5, adding flipped or cropped images show good improvements in most classes. These two algorithms make considerable changes to the image structure so more variations are introduced to the training set. Smaller improvements were observed for some of the blurred or noise images, though in some cases the classifier performance was worse. The 'rider' class was particularly affected by the blur and noise, this was observed to be a result of misclassification with the 'pedestrian' class. It is very challenging to distinguish cyclists from pedestrians after the images are blurred or noised. The 'fence' category in our specific local environment was often constructed as a solid concrete wall, which was easily confused by the appearance of buildings. This category shows significant improvement after augmenting the dataset.

C. Qualitative Analysis of CNN Transferability

As shown in Figure 6, the primary model trained on Cityscapes dataset is capable of performing an acceptable level of classification for categories with common features such as vegetations, roads, sky and vehicles. In our local traffic environment however, there are more roundabout

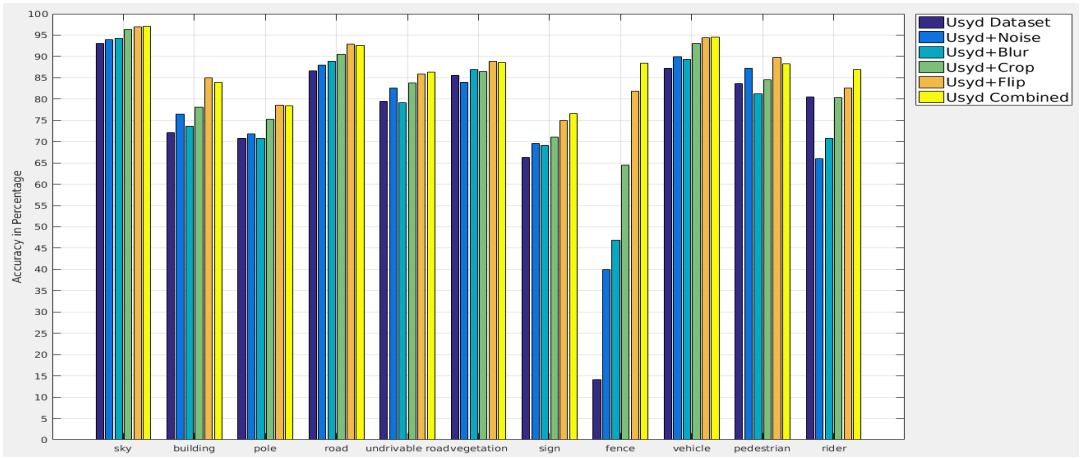
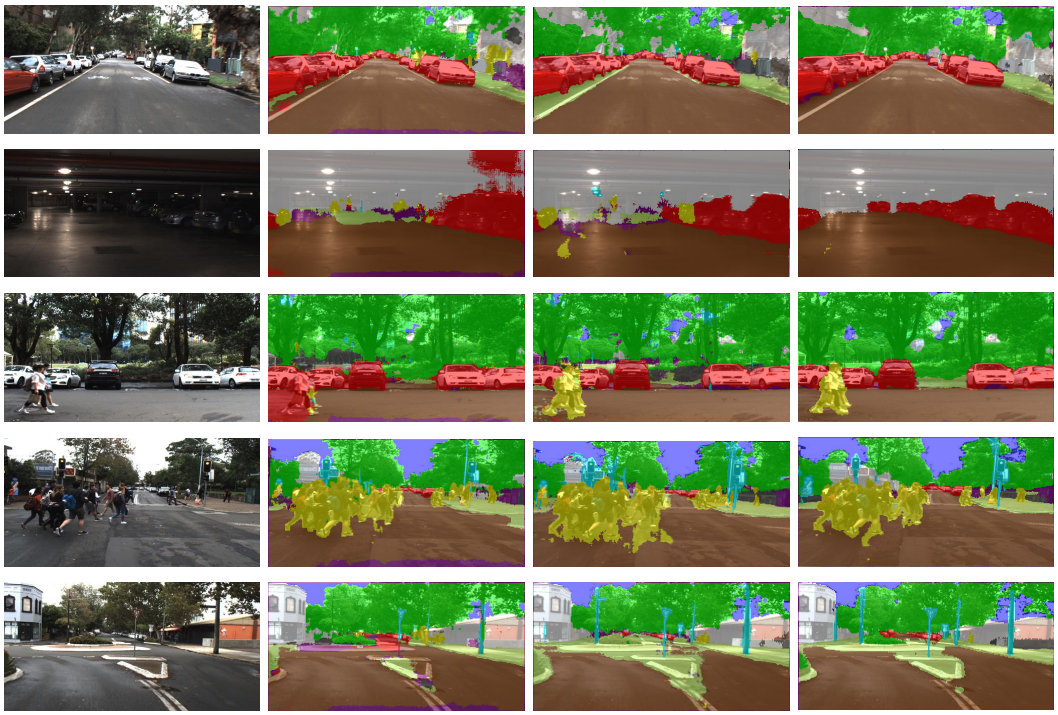


Fig. 5: The influence of data augmentation algorithms on each class in USyd Dataset



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

Fig. 6: Semantic segmentation results. Column (a) is selected image frames from several video sequences taken around University of Sydney for testing. Column (b) is result images from model trained only on Cityscapes dataset. Column (c) is fine-tuned results on USyd dataset with 150 images. Column (d) is fine-tuned on augmented USyd dataset with 750 images. Red is for vehicles, white is for buildings, brown is for roads, green is for vegetations, blue is for sky, neon green is for undrivable roads, yellow is for pedestrian and riders, cyan is for poles and sign symbols, gray is for fence and purple is for misc or unlabeled pixels.

structures which are considered as 'undrivable road'. Also, pedestrians (students) usually appear in groups on/around campus with their backpacks or handbags. The classification of these two categories are enhanced after incorporating USyd annotated data into our model. In addition, we had a few frames from a nearby car park building. The results show that even with a small amount of images (less than 10 car park images annotated), the segmentation accuracy can be greatly improved.

V. Conclusions and Future Works

In this paper, we proposed a CNN architecture for semantic segmentation trained for our particular urban environment. The performance of a neural network is task-specific

and models trained on publicly available datasets do not always generalise for every scenario. Focused on our self-driving vehicle application, we collected and annotated a 150-image dataset around University of Sydney to add training samples for features specific to our local environment. With a small dataset, it is not sufficient to reliably train a model from scratch. Therefore, we trained a primary model on Cityscapes dataset and fine-tuned this model using our local dataset.

Four data augmentation algorithms were implemented to expand the training set. We measured the influence of different data augmentation algorithms and found a greatly improved classification performance for most classes. The combination of multiple augmentation algorithms outperformed the single augmentation in most cases, though some negative affects for certain algorithms were identified. In general, it was determined that adding flipped or cropped augmented images improved segmentation accuracy performance much more than blurring or adding noise.

To optimize the process of training and fine-tuning, we combined and modified the encoder and decoder of ENet so that the fine-tuning can start and stop at any time. This also reduced the computational cost for cross validation in order to minimize the biased results towards any single validation set. In general, by using transferred CNN features and augmenting a small dataset, the accuracy of semantic segmentation in local areas can be greatly improved.

For the next stage of this project, we plan to use the laser point cloud data to constrain the accuracy of 2D semantic segmentation and construct a 3D semantic map for local areas. This will provide a robust information for navigating and controlling our autonomous cars around University of Sydney.

VI. Acknowledgment

This work has been funded by the Australian Centre for Field Robotics, University of Sydney and Australian Government through the Australian Research Council Discovery Grant DP160104081. It was also funded by the Spanish MINECO through the SmartElderlyCar project (TRA2015-70501-C2-1-R) and the Community of Madrid through the RoboCity2030 III-CM project (S2013/MIT-2748).

References

[1] W. Maddern, A. Stewart, C. McManus, B. Upcroft, W. Churchill, and P. Newman, "Illumination invariant imaging: Applications in robust vision-based localisation, mapping and classification for autonomous vehicles," in *Proceedings of the Visual Place Recognition in Changing Environments Workshop, IEEE International Conference on Robotics and Automation (ICRA), Hong Kong, China*, vol. 2, 2014, p. 3.

[2] G. Ros, S. Ramos, M. Granados, A. Bakhtiary, D. Vazquez, and A. M. Lopez, "Vision-based offline-online perception paradigm for autonomous driving," in *2015 IEEE Winter Conference on Applications of Computer Vision*. IEEE, 2015, pp. 231–238.

[3] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NIPS)*, December 2012, pp. 1106–1114.

[4] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*. IEEE, 2009, pp. 248–255.

[5] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

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

[7] Z. Wu, C. Shen, and A. v. d. Hengel, "Wider or deeper: Revisiting the resnet model for visual recognition," *arXiv preprint arXiv:1611.10080*, 2016.

[8] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," *arXiv preprint arXiv:1612.01105*, 2016.

[9] 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.

[10] G. J. Brostow, J. Fauqueur, and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[11] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The kitti dataset," *The International Journal of Robotics Research*, p. 0278364913491297, 2013.

[12] 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," *arXiv preprint arXiv:1604.01685*, 2016.

[13] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, "How transferable are features in deep neural networks?" in *Advances in Neural Information Processing Systems (NIPS)*, December 2014, pp. 3320–3328.

[14] Ibeo automotive systems gmbh. [Online]. Available: www.ibeo-as.de

[15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[16] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 3431–3440.

[17] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[18] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[19] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, "Learning and transferring mid-level image representations using convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014, pp. 1717–1724.

[20] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell, "Decaf: A deep convolutional activation feature for generic visual recognition." in *Icml*, vol. 32, 2014, pp. 647–655.

[21] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 38, no. 1, pp. 142–158, 2016.

[22] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *European Conference on Computer Vision*. Springer, 2014, pp. 297–312.

[23] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deepdriving: Learning affordance for direct perception in autonomous driving," in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 2722–2730.

[24] M. Bojarski, D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba, "End to end learning for self-driving cars," *Computing Research Repository (CoRR)*, vol. arXiv:1604.07316, pp. 1–9, April 2016.

[25] B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman, "Labelme: a database and web-based tool for image annotation," *International Journal of Computer Vision*, vol. 77, no. 1-3, pp. 157–173, May 2008.

[26] S. Dodge and L. Karam, "Understanding how image quality affects deep neural networks," in *Quality of Multimedia Experience (QoMEX), 2016 Eighth International Conference on*. IEEE, 2016, pp. 1–6.

[27] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in *The IEEE International Conference on Computer Vision (ICCV)*, December 2015.