

---

# Semantic Segmentation With Multi Scale Spatial Attention For Self Driving Cars

---

**Abhinav Sagar\***

Vellore Institute of Technology  
Vellore, Tamil Nadu, India  
abhinavsagar4@gmail.com

**RajKumar Soundrapandiyan<sup>†</sup>**

Vellore Institute of Technology  
Vellore, Tamil Nadu, India  
rajkumar.s@vit.ac.in

## Abstract

In this paper, we present a novel neural network using multi scale feature fusion at various scales for accurate and efficient semantic image segmentation. We used ResNet based feature extractor, dilated convolutional layers in downsampling part, atrous convolutional layers in the upsampling part and used concat operation to merge them. A new attention module is proposed to encode more contextual information and enhance the receptive field of the network. We present an in depth theoretical analysis of our network with training and optimization details. Our network was trained and tested on the Camvid dataset and Cityscapes dataset using mean accuracy per class and Intersection Over Union (IOU) as the evaluation metrics. Our model outperforms previous state of the art methods on semantic segmentation achieving mean IOU value of 74.12 while running at >100 FPS.

## 1 Introduction

Convolutional neural networks has seen a lot of success in tasks involving classification, detection and segmentation. These include bounding box object detection, pose estimation, keypoint prediction and image segmentation. CNN-based neural networks advances, such as dropout (Srivastava et al., 2014) and batch normalization (Ioffe and Szegedy, 2015) have helped avoid some of the common challenges faced earlier like the curse of dimensionality and vanishing gradient problem while training neural networks.

Convolutional networks are now leading many computer vision tasks, including image classification (Deng et al., 2009), object detection (Girshick et al., 2014), (Zhu et al., 2015) and (Liu and He, 2015) and semantic image segmentation (Chen et al., 2014), (Li et al., 2014) and (Zhao et al., 2017). Semantic segmentation is also known as scene parsing, which aims to classify each and every pixel present in the image. It is one of the most challenging and important tasks in computer vision. The famous fully convolutional network (FCN) (Long et al., 2015) for semantic segmentation is based on VGG-Net (Simonyan and Zisserman, 2014), which is trained on the famous ImageNet dataset (Deng et al., 2009).

Segmentation task is different from classification task because it requires predicting a class for each and every pixel of the input image, instead of only discrete classes for the whole input images. In order to predict what is present in the image for each and every pixel, segmentation needs to find not only what is in the input image, but also where it is. It has a number of potential applications in the fields of autonomous driving, video surveillance, medical imaging etc. This is a challenging problem as there is often a tradeoff between accuracy and speed. Since the model eventually needs to be deployed in real world setting, hence both accuracy and speed should be high.

---

\*Website of author - <https://abhinavsagar.github.io/>

<sup>†</sup>Website of author - <https://sites.google.com/site/rajkumarsrajkumar/>

## 2 Related Work

State-of-the-art methods on semantic segmentation have heavily relied on CNN models trained on large labeled datasets. Fully convolutional networks (FCN) trained pixels-to-pixels using skip connections that combines semantic information from a deep, coarse layer with appearance information from a shallow, fine layer to produce accurate and detailed segmentations. Convolution layers with a kernel size of  $1 \times 1$  take the place of fully connected layers, followed by unpooling layers to recover the spatial resolution of the feature maps. The success of FCN is due to the great improvements in performance and because it showed that CNN can efficiently learn how to make dense class predictions for semantic segmentation.

After FCN, recently proposed models are mainly designed by (1) bringing out novel decoder structure of the networks (Girshick et al., 2014) and (Badrinarayanan et al., 2017); (2) adopting more efficient basic classification models (Liu et al., 2015) and (Bittel et al., 2015); (3) adding integrating context knowledge with some independent modules (Zhu et al., 2015) and (Ronneberger et al., 2015). SegNet (Badrinarayanan et al., 2017) used an alternative decoder variant, in which an encoder decoder convolution path was proposed. Another deconvolution network was used in (Noh et al., 2015) with a similar decoder path as SegNet, but they adopted deconvolution modules to implement upsampling operations.

(Ronneberger et al., 2015) added a  $2 \times 2$  up-convolution layer, with a concatenation with corresponding pooling layer in U-Net. FCCN (Lin et al., 2016) could also be regarded as an alternative decoder structure. (Chen et al., 2018) used atrous spatial pyramid pooling to embed contextual information at various scales which consist of parallel dilated convolutions with different dilation rates. (Zhao et al., 2017) used multi-scale contextual information by combining feature maps generated using different dilated convolutions and pooling operations.

(Lin et al., 2017) proposed to fuse mid-level and high-level semantic features using an encoder decoder architecture. (Paszke et al., 2016) reduced the number of downsampling times to get an extremely tight fusion structure. (Zhao et al., 2018) uses multi-scale images as input and a cascade network to raise efficiency. (Li et al., 2019) uses Subnetwork Aggregation and Sub-stage Aggregation to achieve very high FPS and high accuracy using modified Xception bottleneck. (Yu et al., 2018a) uses spatial path to recover spatial information and to implement real-time calculation.

We summarize our main contributions as follows:

- We propose a new model architecture which used dilated convolutional layers in downsampling part and atrous convolutional layers in upsampling at multiple scales.
- Concat operator is used for merging the feature maps for context encoding. We also propose our very own attention module which encodes channel wise information to model more contextual information and enlarge the receptive field.
- We present the layer wise details, optimization and ablation study of our neural network.
- On evaluating our network using Camvid dataset and Cityscapes dataset using mean accuracy per class and IOU as evaluation metrics, our model outperforms previous state of the art model architectures while running at  $> 100$  FPS.

## 3 Background

### 3.1 Feature Fusion

Feature fusion have been used successfully in semantic segmentation networks. As the increase of the depth of network, the fusion and reuse of features show significant advantages. The upsampling in decoder recovers the spatial information from the downsampling in encoder. (Lin et al., 2017) proposes a refine network module to finely fuse features. (Li et al., 2019) proposes two feature fusion methods (Sub-network Aggregation and Substage Aggregation) to enhance feature extraction capabilities. Using feature fusion, the interaction between different layers in terms of spatial information and semantic information improves thus obtaining better results. A depthwise separable convolution layer with kernel size  $k = 3$  is used for feature fusion.

### 3.2 Context Encoding

(Hu et al., 2018) used channel wise attention to take into account channel wise information. This work achieved state of the art results in image classification. The challenge in semantic segmentation is to enhance the receptive field and the classification ability. The goal is to extract more contextual information using multi scale feature maps. We present our own attention based module which handles the aforementioned problem well.

### 3.3 Attention Modules

The advantage of using attention based models are that they are very effective in modelling long range dependencies. (Vaswani et al., 2017) first used self attention module for machine translation. (Wang et al., 2018b) used attention module for images and video using non local operation. In this work, we use attention module for more efficient image segmentation by taking into account more contextual information thus enlarging the receptive field.

## 4 Proposed Method

### 4.1 Dataset

1. **Camvid dataset:** The Cambridge-driving Labeled Video Database (CamVid) is a collection of videos with object class semantic labels, complete with metadata. The database provides ground truth labels that associate each pixel with one of 32 classes. The images are of size  $360 \times 480$ . The original images are taken as ground truth. For any algorithm, the metrics are always evaluated in comparison to the ground truth data. The ground truth information is provided in the dataset for the training and test set. A sample image from dataset is shown in Figure 1:

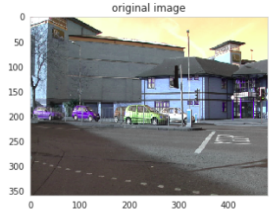


Figure 1: Sample image from dataset.

For semantic segmentation problems, the ground truth includes the image, the classes of the objects in it and a segmentation mask for each and every object present in a particular image. Since there is a lot of overlaps in between the labels, hence for the sake of convenience we have gone with 12 labels in this work. These images are shown in binary format for each label separately in Figure 2:

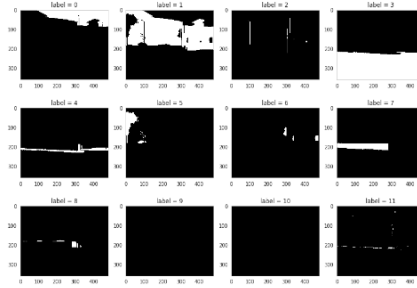


Figure 2: Sample image converted to binary class mask.

The classes chosen from the dataset are Sky, Building, Pole, Road, Pavement, Tree, SignSymbol, Fence, Car, Pedestrian and Bicyclist.

**2. Cityscapes dataset:** This dataset contains urban street scenes images from 50 different cities. The images are divided into 5,000 finely annotated images and 19,998 coarsely annotated images. The total number of classes in the dataset is 30, but we have only used 19 classes for both training and evaluation. The images in training, validation and test set are 2,975, 500, 1,525 respectively.

## 4.2 Network Architecture

We split the dataset into 2 parts with 85 percent images in the training set and 15 percent images in the test set. The loss function used is categorical cross entropy. We used dilated convolutions in downsampling part to reduce the feature maps and atrous convolutions in upsampling part to recover back the features. Concat operation is used to merge the features at different scales thus encoding more contextual information. We use our very own designed attention module for enlarging the receptive field and encode more contextual information. The attention module used in this work is shown in Figure 3:

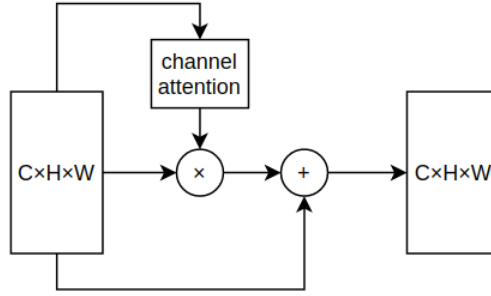


Figure 3: Illustration of our attention module. Here  $\times$  denotes matrix multiplication and  $+$  denotes element wise sum. C, W and H respectively denotes channel, width and height of a layer respectively.

For the dilated convolutional layer we didn't use any padding, used  $3 \times 3$  filter and use relu as the activation function. For the max pooling layer, we used  $2 \times 2$  filters and strides of  $2 \times 2$ . ResNet is used as the backbone and the feature extractor. In the upsampling path we used atrous convolutions layers with  $4 \times 4$  kernel size and strides of  $4 \times 4$ . Softmax is used as the activation function in the last layer to output discrete probabilities of whether an object is present in a particular pixel location or not. We used Adam as the optimizer for training our network. The network architecture used in this work is shown in Figure 4:

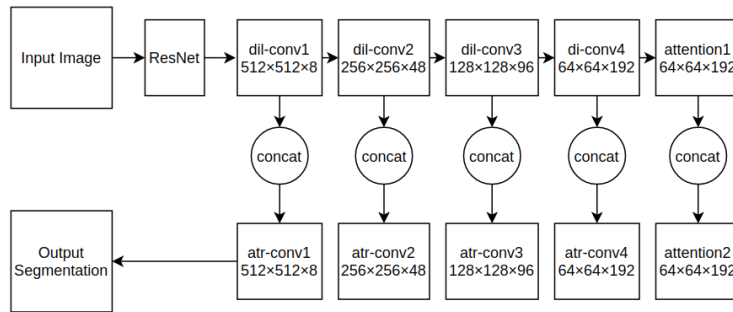


Figure 4: Illustration of our neural network architecture. Here dil-conv represents dilated convolutions and atr-conv represents atrous convolutions. attention1 and attention2 are the two channel wise attention modules used in this work.

### 4.3 Optimization

Suppose given a local feature  $C$ , we feed it into a convolution layers to generate two new feature maps  $B$  and  $C$  respectively. After that we perform a matrix multiplication between the transpose of  $A$  and  $B$ , and apply a softmax layer to calculate the spatial attention map as shown in Equation 1:

$$s_{ji} = \frac{\exp(A_i \cdot B_j)}{\sum_{i=1}^N \exp(A_i \cdot B_j)} \quad (1)$$

We then perform a matrix multiplication between the transpose of  $X$  and  $A$  and reshape their results. Then we multiply the result by a scale parameter  $\beta$  and perform an element-wise sum operation with  $A$  to obtain the final output as shown in Equation 2:

$$E_j = \alpha \sum_{i=1}^N (s_{ji} D_i) + C_j \quad (2)$$

The Equation 2 shows that the resultant feature of each channel is a weighted sum of the features of all channels and models the semantic dependencies between feature maps at various scales. For a single backbone  $\phi n(x)$ , a stage process, the stage in the previous backbone network and sub-stage aggregation method can be formulated as shown in Equation 3:

$$x_n^i = \begin{cases} x_n^{i-1} + \phi_n^i(x_n^{i-1}) \\ [x_n^{i-1}, x_{n-1}^i] + \phi_n^i([x_n^{i-1}, x_{n-1}^i]) \end{cases} \quad (3)$$

Here  $i$  refers to the index of the stage.

### 4.4 Loss Functions

The loss function used is Softmax loss as shown in Equation 4:

$$Loss = \frac{1}{N} \sum_i -\log \left( \frac{e^{p_i}}{\sum_j e^{p_j}} \right) \quad (4)$$

where  $p$  is the output prediction of the network,  $i$  and  $j$  are stages in the network.

### 4.5 Ablation Studies

The effect of the number of pooling layers on Intersection Over Union(IOU) using Camvid dataset is shown in Table 1. As can be noted, using more pooling layers increases IOU but it's effect is not consistent.

Table 1: Results on Camvid dataset with different numbers of pooling in each stage of the backbone, “ $\times N$ ” means the number of pooling.

Number of pooling	mIoU(%)
Pooling $\times 0$	70.4
Pooling $\times 1$	71.3
Pooling $\times 2$	73.8
Pooling $\times 3$	73.4
Pooling $\times 4$	74.9
Pooling $\times 5$	75.6

The effect of varying the number of branches and fusion methods used in model architecture on IOU using Cityscapes dataset is shown in Table 2. Using more number of branches and concat fusion instead of not using one increases the IOU.

We investigate the effect of each component in our proposed network. We use VGG16 and ResNet50 as the base network and evaluate our method on the Camvid validation dataset as shown in Table 3:

Table 2: Results on Cityscapes dataset with with different number of branches and fusion methods.

Number of branches	Fusion methods	mIoU(%)
1	None	74.4
1	concat	75.8
2	None	75.7
2	concat	77.5

Table 3: Accuracy and parameter analysis of our baseline model: VGG16 and ResNet50 on Camvid validation dataset. Here we use FCN-32s as the base structure. FLOPS are estimated for input of size  $3 \times 640 \times 360$ .

Method	BaseModel	FLOPS	Parameters	Mean IOU(%)
FCN-32s	VGG16	47.5M	1.4M	61.28
FCN-32s	ResNet50	12.5G	22.5M	61.73

#### 4.6 Implementation Details

The initial learning rate was set to 0.001 in all our experiments. Momentum and weight decay coefficients are set to 0.9 and 0.0001 respectively. Batch size of 8 was chosen for all our experiments. Data augmentation operations like shearing, cropping and flipping was performed to artificially increase the dataset size.

## 5 Results

In this section we present the results of our work and compare the results we achieved with previous state of the art. The model is trained for 40 epochs and reaches a training mean pixel accuracy of 93 percent and validation mean pixel accuracy of 88 percent. The loss and pixel wise accuracy (both training and test) are plotted as a function of epochs in Figure 5:

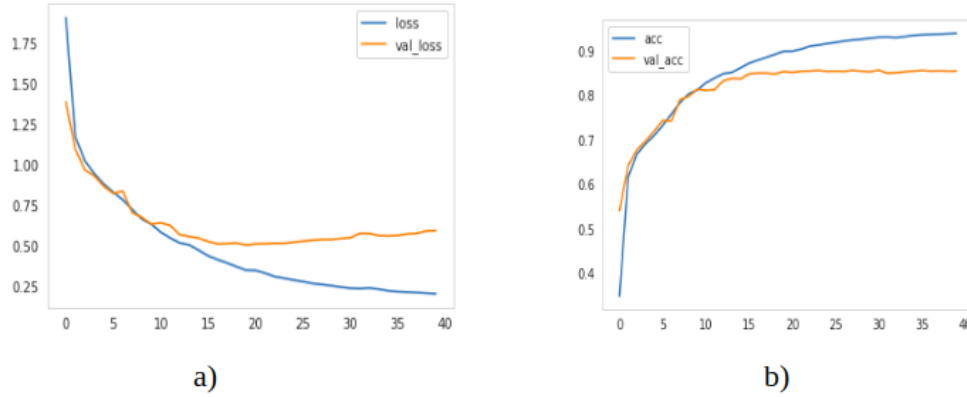


Figure 5: a) Loss vs epochs b) Accuracy vs epochs

For evaluating the performance of our model architecture, we used two evaluation metrics:

1. **Mean Accuracy per-class** - This metric outputs the class wise prediction accuracy per pixel.
2. **Mean IOU** - It is a segmentation performance parameter that measures the overlap between two objects by calculating the ratio of intersection and union with ground truth masks. This metric is also known as Jaccard Index.

The class wise IOU values were calculated using Equation 5.

$$IoU = \frac{TP}{(TP + FP + FN)} \quad (5)$$

Where  $TP$  denotes true positive,  $FP$  denotes false positive,  $FN$  denotes false negative and  $IOU$  denotes Intersection over union value.

We present the class wise IOU values for all the twelve classes present using CamVid dataset in Table 4.

Table 4: IOU values for all classes

Class	1	2	3	4	5	6	7	8	9	10	11	12
IOU	0.923	0.905	0.232	0.947	0.831	0.344	0.569	0.792	0.283	0.261	0.457	0.527

The effect of using multiple blocks, FLOPS and parameters on IOU using Cityscapes dataset is shown in Table 5. Here FLOPS and parameters are a measure of computation required by our model architecture.

Table 5: Detailed performance comparison of our proposed aggregation strategy on Cityscapes dataset. ' $\times N$ ' means that we replicate N backbones to implement feature aggregation

Model	FLOPs(G)	Params(M)	mIoU(%)
Backbone A	1.4	2.2	64.7
Backbone A $\times 2$	2.3	4.5	65.3
Backbone A $\times 3$	2.7	7.4	62.1
Backbone A $\times 4$	2.9	10.6	57.8
Backbone B	0.8	1.4	59.5
Backbone B $\times 2$	1.2	3.3	61.5
Backbone B $\times 3$	1.4	4.7	56.4
Backbone B $\times 4$	1.5	6.1	52.7

A comparative analysis using CamVid dataset achieved by previous state of the art model architectures vs ours is shown in Table 6.

Table 6: Accuracy and speed analysis on CamVid test dataset. Ours is  $512 \times 768$  input and others are  $768 \times 1024$  input.

Model	Frame(fps)	mIoU(%)
DPN (Yu et al., 2018b)	1.2	60.1
DeepLab (Chen et al., 2017)	4.9	61.6
ENet (Paszke et al., 2016)	-	51.3
ICNet (Zhao et al., 2018)	27.8	67.1
BiSeNet1 (Yu et al., 2018a)	-	65.6
BiSeNet2 (Yu et al., 2018a)	-	68.7
DFANet A (Li et al., 2019)	120	64.7
DFANet B (Li et al., 2019)	160	59.3
SwiftNet pyr (Orsic et al., 2019)	-	72.85
SwiftNet (Orsic et al., 2019)	-	73.86
Ours	124	74.12

A comparative analysis using Cityscapes dataset achieved by previous state of the art model architectures vs ours is shown in Table 7.

The results comparing the predicted segmentations and ground truth segmentation using CamVid dataset is shown in Fig 6.

Table 7: Accuracy and speed analysis on Cityscapes test dataset.

Model	InputSize	FLOPs	Params	Time(ms)	Frame(fps)	mIoU(%)
PSPNet (Zhao et al., 2017)	$713 \times 713$	412.2G	250.8M	1288	0.78	81.2
DeepLab (Chen et al., 2017)	$512 \times 1024$	457.8G	262.1M	4000	0.25	63.1
SegNet (Badrinarayanan et al., 2017)	$640 \times 360$	286G	29.5M	16	16.7	57
ENet (Paszke et al., 2016)	$640 \times 360$	3.8G	0.4M	7	135.4	57
CRF-RNN (Zheng et al., 2015)	$512 \times 1024$	-	-	700	1.4	62.5
FCN-8S (Long et al., 2015)	$512 \times 1024$	136.2G	-	500	2	63.1
FRRN (Pohlen et al., 2017)	$512 \times 1024$	235G	-	469	0.25	71.8
ICNet (Zhao et al., 2018)	$1024 \times 2048$	28.3G	26.5M	33	30.3	69.5
BiSeNet1 (Yu et al., 2018a)	$768 \times 1536$	14.8G	5.8M	13	72.3	68.4
BiSeNet2 (Yu et al., 2018a)	$768 \times 1536$	55.3G	49M	21	45.7	74.7
DFANet A (Li et al., 2019)	$1024 \times 1024$	3.4G	7.8M	10	100	71.3
DFANet B (Li et al., 2019)	$1024 \times 1024$	2.1G	4.8M	8	120	67.1
Ours	$1024 \times 1024$	1.8G	5.5M	6	134	72.4

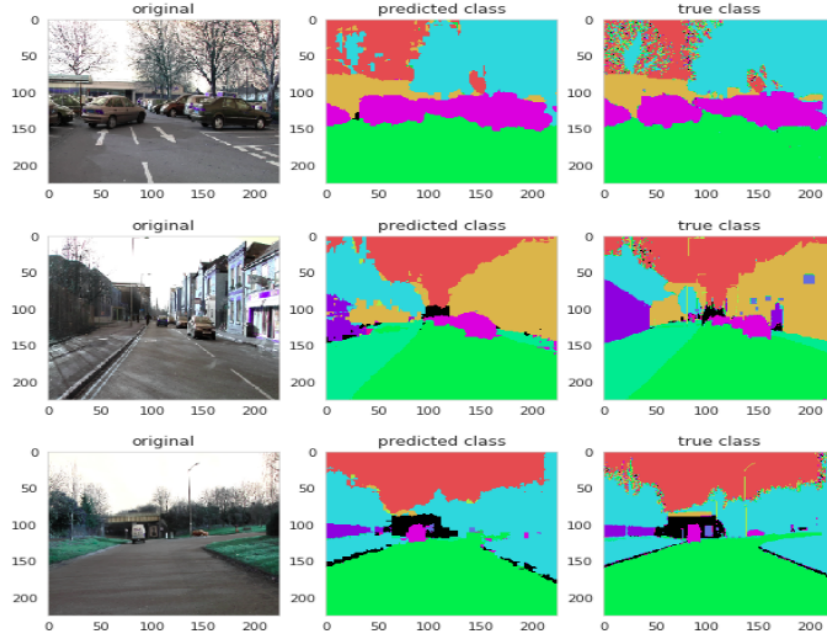


Figure 6: Results using CamVid dataset. First column: input image from dataset, second column: predicted segmentation from our network and third column: ground truth segmentation.

## 6 Conclusions

In this paper, we proposed a semantic segmentation network using multi scale attention feature maps and validated its performance on Camvid dataset and Cityscapes dataset. We used a downsampling and upsampling structure with dilated and atrous convolutional layers respectively with combinations between corresponding pooling and unpooling layers. We also propose our own attention module to enlarge the receptive field and encode more contextual information. Multi scale feature maps are merged using concat operator for encoding more contextual information. We present loss function, optimization details, ablation studies and evaluation metrics used. Our network achieves mean IOU value of 74.12 which is better than the previous state of the art on semantic segmentation while running at >100 FPS.



## Acknowledgments

We would like to thank Nvidia for providing the GPUs.

## References

- A. F. Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.
- V. Badrinarayanan, A. Kendall, and R. Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.
- S. Bittel, V. Kaiser, M. Teichmann, and M. Thoma. Pixel-wise segmentation of street with neural networks. *arXiv preprint arXiv:1511.00513*, 2015.
- L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062*, 2014.
- L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE transactions on pattern analysis and machine intelligence*, 40(4):834–848, 2017.
- L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.
- J. Dai, K. He, Y. Li, S. Ren, and J. Sun. Instance-sensitive fully convolutional networks. In *European Conference on Computer Vision*, pages 534–549. Springer, 2016.
- J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.
- M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International journal of computer vision*, 88(2):303–338, 2010.
- R. Gadde, V. Jampani, and P. V. Gehler. Semantic video cnns through representation warping. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4453–4462, 2017.
- R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014.
- J. Hu, L. Shen, and G. Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.
- S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *arXiv preprint arXiv:1502.03167*, 2015.
- S. Jégou, M. Drozdal, D. Vazquez, A. Romero, and Y. Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 11–19, 2017.
- D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- H. Li, R. Zhao, and X. Wang. Highly efficient forward and backward propagation of convolutional neural networks for pixelwise classification. *arXiv preprint arXiv:1412.4526*, 2014.
- H. Li, P. Xiong, H. Fan, and J. Sun. Dfanet: Deep feature aggregation for real-time semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9522–9531, 2019.

- G. Lin, C. Shen, A. Van Den Hengel, and I. Reid. Efficient piecewise training of deep structured models for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3194–3203, 2016.
- G. Lin, A. Milan, C. Shen, and I. Reid. Refinenet: Multi-path refinement networks for high-resolution semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1925–1934, 2017.
- B. Liu and X. He. Multiclass semantic video segmentation with object-level active inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4286–4294, 2015.
- Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang. Semantic image segmentation via deep parsing network. In *Proceedings of the IEEE international conference on computer vision*, pages 1377–1385, 2015.
- 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*, pages 3431–3440, 2015.
- H. Noh, S. Hong, and B. Han. Learning deconvolution network for semantic segmentation. In *Proceedings of the IEEE international conference on computer vision*, pages 1520–1528, 2015.
- M. Orsic, I. Kreso, P. Bevandic, and S. Segvic. In defense of pre-trained imagenet architectures for real-time semantic segmentation of road-driving images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 12607–12616, 2019.
- 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.
- C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun. Large kernel matters—improve semantic segmentation by global convolutional network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4353–4361, 2017.
- T. Pohlen, A. Hermans, M. Mathias, and B. Leibe. Full-resolution residual networks for semantic segmentation in street scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4151–4160, 2017.
- S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.
- E. Romera, J. M. Alvarez, L. M. Bergasa, and R. Arroyo. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):263–272, 2017.
- O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- A. Sagar. Bayesian multi scale neural network for crowd counting. *arXiv preprint arXiv:2007.14245*, 2020a.
- A. Sagar. Learning to detect 3d objects from point clouds in real time. *arXiv preprint arXiv:2006.01250*, 2020b.
- A. Sagar. Monocular depth estimation using multi scale neural network and feature fusion. *arXiv preprint arXiv:2009.09934*, 2020c.
- K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.
- N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1): 1929–1958, 2014.

- C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.
- A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.
- P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. Cottrell. Understanding convolution for semantic segmentation. In *2018 IEEE winter conference on applications of computer vision (WACV)*, pages 1451–1460. IEEE, 2018a.
- X. Wang, R. Girshick, A. Gupta, and K. He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018b.
- C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. Bisenet: Bilateral segmentation network for real-time semantic segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 325–341, 2018a.
- C. Yu, J. Wang, C. Peng, C. Gao, G. Yu, and N. Sang. Learning a discriminative feature network for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1857–1866, 2018b.
- F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.
- H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal. Context encoding for semantic segmentation. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 7151–7160, 2018.
- H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017.
- H. Zhao, X. Qi, X. Shen, J. Shi, and J. Jia. Icnet for real-time semantic segmentation on high-resolution images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 405–420, 2018.
- S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1529–1537, 2015.
- Y. Zhu, R. Urtasun, R. Salakhutdinov, and S. Fidler. segdeepm: Exploiting segmentation and context in deep neural networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4703–4711, 2015.