Improvement in Accuracy and Speed of Image Semantic Segmentation via Convolution Neural Network Encoder-Decoder

Hanieh Zamanian
Department of Electrical and Computer Engineering., University of Birjand, Birjand, Iran
hanieh.zamanian@birjand.ac.ir
Hassan Farsi*
Department of Electrical and Computer Engineering., University of Birjand, Birjand, Iran
hfarsi@birjand.ac.ir
Sajad Mohamadzadeh
Technical faculty of Ferdows, University of Birjand, Birjand, Iran
s.mohamadzadeh @birjand.ac.ir

Received: 24/Feb/2018 Revised: 22/Aug/2018 Accepted: 16/Sep/2018

Abstract

Recent researches on pixel-wise semantic segmentation use deep neural networks to improve accuracy and speed of these networks in order to increase the efficiency in practical applications such as automatic driving. These approaches have used deep architecture to predict pixel tags, but the obtained results seem to be undesirable. The reason for these unacceptable results is mainly due to the existence of max pooling operators, which reduces the resolution of the feature maps. In this paper, we present a convolutional neural network composed of encoder-decoder segments based on successful SegNet network. The encoder section has a depth of 2, which in the first part has 5 convolutional layers, in which each layer has 64 filters with dimensions of 3×3. In the decoding section, the dimensions of the decoding filters are adjusted according to the convolutions used at each step of the encoding. So, at each step, 64 filters with the size of 3×3 are used for coding where the weights of these filters are adjusted by network training and adapted to the educational data. Due to having the low depth of 2, and the low number of parameters in proposed network, the speed and the accuracy improve compared to the popular networks such as SegNet and DeepLab. For the CamVid dataset, after a total of 60,000 iterations, we obtain the 91% for global accuracy, which indicates improvements in the efficiency of proposed method.

Keywords: Semantic Segmentation; Convolutional Neural Networks; Encoder – Decoder; Pixelwise Semantic Interpretation.

1. Introduction

Semantic segmentation for 2D images, video and even 3D data is one of the key problems in computer vision [1]. For large images, semantic segmentation is one of the high-level tasks that makes a full scene understanding [2]. The importance of the scene understanding as a major problem in computer vision is due to the fact that a large number of applications is improved or developed by the inference of image information [3,4]. Some of these include independent driving, human-machine engagement, image search engines, and virtual reality [5]. In the past, solutions were developed by using various machine learning techniques for this problem. Despite the popularity of machine learning based methods, deep learning has revolutionized the solution of these problems, so that many computer vision problems, including semantic segmentation, with the use of deep architecture, especially the convolutional neural networks (CNN), perform with even better accuracy than other approaches [6-8]. Semantic segmentation is still challenging task today. Theoretically, semantic segmentation combines two functions [9]; one is the segmentation of the image, and the other is the classification of the objects in which eventually connects parts of the image that belongs to one object class. By semantic segmentation, we can obtain the pixelwise semantic interpretation of the image [10]. Compared to the object detection, semantic segmentation is considered to be a major improvement because the distinction between objects is mentioned based on the distinction between the pixels. However, there are several problems and challenges that are mainly summarized in the following aspects: 1) Object Level: due to differences in lighting, viewing points and distance, an object in the image may be seen in very different ways. 2) Class Level: objects in one class may be different, and objects in different classes may be similar. For example, a pedestrian in front of a car divides the visual view of the car into two parts. 3) Background: a clean background helps to split, but in practice, the background is usually complicated which may be misleading [11].

Before the development of the deep learning algorithms, there were several popular ways to segment the image. Threshold splitting is one of the most basic methods of image segmentation, in which pixels are divided according to their color or gray levels [12]. The edge segmentation is the identification of some points at the edge of the objects which extracts a segmented region by using some particular algorithms. The Snake model transforms the segmentation into an energy minimization problem to find the edges [13].

* Corresponding Author

Watershed algorithm is a regional division based on morphology. Regional growth method is also a common method for regional segmentation [14]. The main idea of this algorithm is to find the growth criterion and then to search for a pixel of grain in each region. The random forest, which has multiple decision trees, is used as a classifier [15]. In image segmentation based on the graph theory, the image is depicted as an indirect weighted graph in which pixels are considered as nodes. The weight of the edge between the nodes is related to the difference between two pixels. Cutting these edges depends on the energy function. Markov Random Fields are an indirect probability graph model used to split the image. Each pixel is assigned a random value and then each pixel is categorized by using probabilistic methods.

Following the development of deep learning, a series of semantic segmentation methods based on the convolutional neural network were proposed and resulted in great progress. One of the most popular primary learning methods was the fragmentation, in which each pixel was categorized separately by using a piece of the surrounding image. The main reason for the use of patches is that the classification networks usually have fully connected layers, which require fixed-dimensional images.

In 2014, the fully convoluted (FCN) network is introduced by Long and colleagues [16], presented the well-known CNN architecture for dense prediction without fully-connected layers. In the FCN algorithm, the size of the input image is arbitrary and is faster than the fractional classification method. Almost all of the subsequent later methods of semantic segmentation somehow try to improve this pattern.

After FCN, SegNet [17], Detailed Convolutions [18], DeepLab V1 [19], DeepLab V2 [20], RefineNet [21], PSPNet [22], Big Core Problems [23] and DeepLab v3 [24] have been consecutively proposed and improved the accuracy of pixel-wised segmentation.

However, the main problems with these methods are the size of the networks and the time of calculation which are great for using them for real-time applications. Especially for semantic segmentation applications, such as independent driving, they are undesirable and sometimes impossible.

In this paper, in order to overcome to these problems, an idea has been presented in which a new architecture for the semantic segmentation, especially for city images, is introduced with a better accuracy than the successful architecture of SegNet and provides 10 times fewer parameters than SegNet. In later sections, after presenting an overview of existing architectures by using deep learning architecture, an innovative technique that has been tested in the framework of MATLAB is described and, finally, the results on the CamVid database [25] are shown by common criteria and compared to other successful methods.

2. Related Work

In order to understand the meaning of the semantic segmentation with the modern learning architecture of deep learning, it must be noted that in fact, semantic segmentation is the achievement of the correct inference, that its base is classification, and its result is a prediction of likelihood to each object class. Therefore, a ranking list of the object similarity with the objects in the image should be provided. Localization or diagnosis is the next step in deduction, not only the classes but also additional information about the location of these classes, such as their center or boundary boxes, should be taken into account. Therefore, it is clear that semantic segmentation is a natural step for achieving accurate inferences; and its purpose is dense predictions and labeling for each pixel. In this way, each pixel is labeled with the object class or region that is most similar to it.

Training a deep neural network from the beginning is often not possible for various reasons; first, a large set of data is needed for network training (and usually not available), and achieving convergence for an acceptable result is taking a long time. Second, even if a dataset is large enough to deprive its long-term convergence, it is often useful to begin with pre-trained weight training rather than randomly selecting them [26, 27]. Pre-trained weight training means initializing the weights of the network when are learned for another dataset or task instead of initializing the weights randomly and then start training the network for the special task and dataset. Initializing the weights is referred to use a pre-trained network. The first network is pre-trained network. The second one is the network which is fine-tuning. Adjusting the weight of a pre-trained network is one of the most important learning transfer scenarios by continuing the training process.

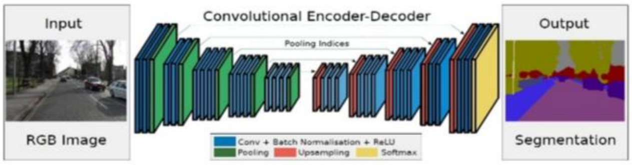

Fig. 1. SegNet architecture for pixelwise classification in semantic segmentation [17]

Yosinski et al. proved that the transfer of features, even on different issues, could be better than using a random initial value [28]. So, it is important to consider that when the difference between the previously trained problem and the goal is great, the ability to transfer features decreases.

However, the use of the transfer learning method is not simple. Using a pre-trained network contains architectural constraints that should be considered. However, since the introduction of a completely new architecture is not common, usually the architecture of the networks or their components are reused, so transfer learning is possible. On the other hand, the training process is also slightly different when the network configuration is used to be fine-tuned instead of training it from the beginning. Proper selection of layers which are usually the high levels of the network needs to be fine-tuned. Since the layers of the lower levels extract the general characteristics, lower layers also select the appropriate learning rate, which is usually a small number. Since it is expected that the pre-trained weights are relatively proper, so they do not require many changes.

Another problem for image semantic segmentation with deep networks is the learning dataset. Due to the inherent complexity of aggregation and the creation of a segmentation dataset with marked pixels, the number of the images in these datasets is not as large as the classification datasets, such as ImageNet [29,30]. This problem is even worse when dealing with colorful or 3D image datasets. So, the transfer learning and, in particular, the precise adjustment of pre-trained classified networks, is a common trend in the segmentation networks. The success of deep learning techniques in high-level issues in computer vision, in particular in supervised approaches such as CNN for image classification or object detection [31-33], induced researchers to use this technique, to check the capabilities of such networks for problems with pixel level labeling such as semantic segmentation. The key advantage of these techniques, which surpasses them from traditional methods, is the ability to learn the proper features for the desired problem. Nowadays, the most successful advanced deep learning methods for the semantic segmentation are based on a common pioneer: Fully Convolutional Network (FCN) [16]. The insight of this approach was to use existing CNNs as powerful visual models that were able to learn the hierarchy of features. They deployed existing and well-known classification models, AlexNet [31], VGG16 (16 pure layers), [32], GoogLeNet [33], and ResNet [34] into very complex networks. The result of replacing fully connected layers with a convolutional layer is achieving the network output, as a spatial map, instead of a ranking list. These maps are sampled using deconvolution [35, 36] and produce dense outputs with labeled pixels. This was considered as a milestone since it showed how CNNs can train for this problem and effectively evaluate dense predictions for the semantic segmentation for arbitrary input sizes. This method greatly improves the segmentation accuracy along with maintaining efficiency, compared to traditional methods, on different datasets such as PASCAL VOC.

For all these reasons, the FCN is the base of deep learning methods, which are applied to semantic segmentation. Despite the power and flexibility of the FCN model, this model does not have some of the necessary features which makes it difficult to use for some problems. The causes of the undesirable results of this model are inherent irregularity of its spatial form, which makes it impossible to use useful general information, and on the other hand, does not work for real-time use when the resolution is high.

Of course, FCN-based architectures are very popular and successful, but there are other alternatives that are noteworthy. In general, all of them, like the VGG-16 [32], consider a network for classification and eliminates all of its fully connected layers. This part of the new segmentation network is often called the encoder and produces a low-resolution image or a feature map. Decoding or displaying the low-resolution images is difficult to segment as pixel level predictions, and usually, the difference of these kinds of architectures is in the decoding section.

SegNet [17] is a clear example of these kinds of architectures. Figure 1 shows an overview of this architecture. The SegNet decoding part is comprised of a set of upsampling and convolution layers. The softmax classification layer which is located at the end of the network is used to predict the pixel tags of the output, and the output has the same resolution as the input image. Each layer of upsampling in the decoding section corresponds to a max-pooling in the encoder section.

These layers upsample the features by using the max-pooling indices in the encoder phase. Then upsampled maps are convolved with a set of trained filter banks to produce a map of dense features. When the feature map is returned to the original resolution, it is fed to the softmax layer to produce the final segmentation. In the SegNet encoder section, the number of convolutional layers is equal to Vgg16, only the fully connected layers in the VGG16 architecture are eliminated, which significantly reduces network dimensions and learning parameters. An important part of the SegNet architecture is its decoder section. The decoder in SegNet is hierarchically related to each step of the encoding section. Each decoder must receive max pooling indices from their respective encoders and apply non-linear upsampling to their inputs. The use of these indexes has several advantages [17]: First, it improves the boundary detections. Second, high-frequency details are maintained. Third, it reduces the training parameters. Forth, this method can be used in many encoder-encoder architectures by some modifications. Figure 2 shows how to apply the unpooling operation in SegNet architecture. As it shows, the indexes of each max pooling layer are stored in the codec section, and then in the decoding section and in the upsample layer, the unpooling operation is performed by using stored parameters.

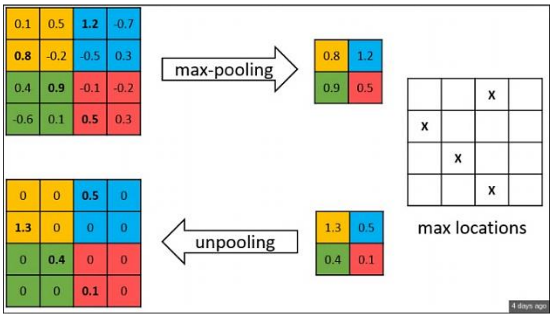

Fig. 2. Unpooling operation in SegNet architecture [17]

3. Proposed Method

In recent years, a lot of research has been done on pixel labeling for semantic segmentation of images. Some of these approaches have used deep architecture to predict pixel tags, but the results seem to be undesirable. The reason for these unacceptable results is mainly due to the existence of max pooling operators, which reduces the resolution of the feature maps. SegNet, introduced an idea, for translating the low resolution of these features to the input image resolution for pixel categorization, which results in the creation of useful features for determining the exact location of the objects boundaries in the images. SegNet is designed for pixel-wise semantic segmentation and mainly used to understand road imagery. In a typical road image, the majority of pixels are related to large classes such as roads and buildings. A desirable semantic segmentation operator must correctly classify and isolate the boundaries of objects, due to the inequality and proportion between the numbers of pixels belonging to different classes. In addition, a segmentation operator must determine the type of object in spite of its small dimensions; therefore, it must extract the correct information from the boundary of objects, so that it can correctly decide on the type of objects.

From the computational point of view, the designed network should be efficient in terms of memory and the duration of computation at the inference level. The ability to train the network based on weighing techniques, such as Stochastic Gradient Descent (SGD), is an advantage in deep learning networks such as SegNet which speeds up the convergence of network learning.

Thus, with regard to the capabilities of SegNet, which so far has been able to improve the semantic segmentation results, proposed method is a codec method that is inspired by the Segnet algorithm to better determine the parameters and the number of layers for CamVid dataset. As noted, from a computational point of view, the designed network should be efficient in terms of memory and the duration of computations in the inference step. Certainly, with having the lower number of layers and learning parameters, the network will be more efficient from the computational point of view, and will be more useful for online uses. The goal of proposed algorithm is to reduce these parameters simultaneously with increasing precision in semantic segmentation.

In proposed method, the encoder section has a depth of two, which in the first part has five convolutional layers, in which each layer has 64 filters with dimensions of 3×3. After convolution layer, there is a batch normalization layer and then a ReLU layer. A graphical representation of the suggested network graph that is compared to the SegNet network is shown in Figure 3. In Figure 4, proposed network encoding architecture is displayed.

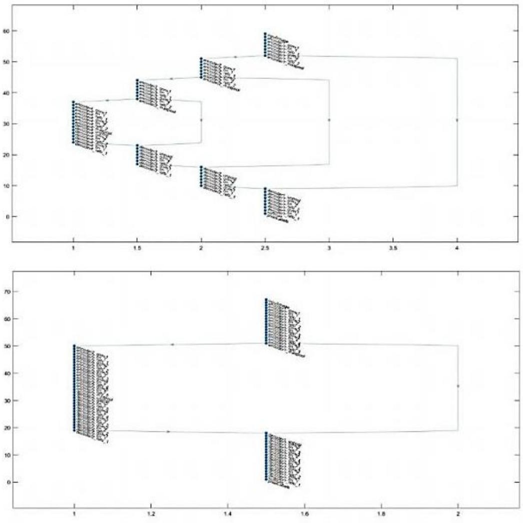

Fig. 3. SegNet architecture [17] (first diagram) vs proposed architecture (second diagram).

In the decoding section, the dimensions of the decoding filters are adjusted according to the convolutions used at each step of the encoding. So, at each step, 64 filters with the size of 3×3 are used for coding, the weights of these filters must be adjusted by network training and adapted to the training data. As shown in Figure 5, the network architecture of proposed method for decoding is displayed. At the end of the decoding section, the output is created with dimensions equal to the input image. The softmax layer performs pixel classification and the result of semantic segmentation of the input image is achieved. Due to a large size of the input image and consequently the large number of pixels that should be decided upon in the classification stage, this stage contains the most adjustable parameters.

Figure 6 shows a complete view of proposed network. One of the goals of this research is to reduce the number of parameters that need to be set during the network training. Table 1 presents a comparison between the number of parameters in different networks and proposed network. As can be seen, the number of parameters in proposed method is about 2 mega. This reduction in the number of parameters reduces the computational cost and memory needed to store network parameters as well as increasing the speed of training and even test the network.

In general, the lower numbers of parameters provide the more effective the network utilization for online and real-time applications. Table 1 shows a comparison between the numbers of adjustable parameters for several methods. As can be seen, due to the low depth of proposed network, the number of parameters that should be adjusted during the training period is much lower than the other methods, resulting in faster training and faster convergence. For example, after 100 epochs, which contain approximately 20,000 iterations, the network converges for the CamVid database.

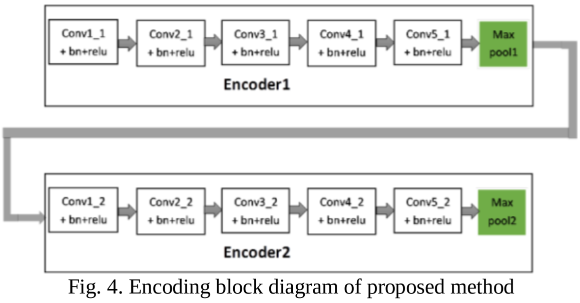

Fig. 4. Encoding block diagram of proposed method

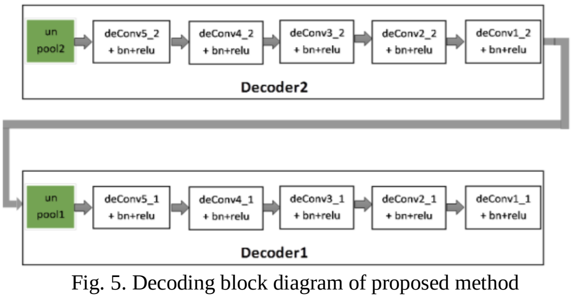

Fig. 5. Decoding block diagram of proposed method

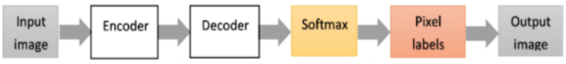

Fig. 6. Block diagram of proposed method

Table 1. Comparison between the adjustable parameters of proposed method and some others

Network name	Type	Number of parameters(*106)
SegNet [17]	convolution	14.7
ENet[37]	residual	0.36
SqueezeNet[38]	convolution	2.7
VGG 16[32]	convolution	138
Proposed method	convolution	1.41

3.1 Database

As mentioned, in this research, CamVid road scene dataset [25] was used to evaluate the performance of the network. This dataset is small and contains 701 images, in which 421 images are used for training set and 280 images are used for testing and validation sets. These images are RGB and include scenes of day and evening, with a resolution of 360 by 480 pixels. The challenge is to separate 11 classes of roads, buildings, cars, pedestrians, signs, columns, pedestrians, sky, trees, bicycles, and fence.

3.2 Network Training

For training and testing of proposed network, the CPU with Intel Core i7-6700HQ, NVIDIA GEFORCE GTX 950M graphic card, one GPU and 12GB of memory have been used. Proposed method codes are written using the MATLAB software toolkit. For training, the stochastic gradient descent (SGD) with the initial learning rate of 0.1 and its reduction by a factor of 0.1 after every 100 epochs (20,000 iterations) and momentum of 0.5 were used. In addition, crossover entropy loss [16] has been used as a target function for network training.

When there is a large variation in the number of pixels in each class in the training images (for example, for the road, sky, and building, the number of pixels in the CamVid dataset are more abundant than other objects), there is a need to weight reduction differently according to the class of objects. This method is called the class balancing. Here, the medium frequency balancing is used [39]. This means that the larger classes in the training set have the weights less than 1 and the smallest class has the highest weight value.

3.3 Comparative Metrics

To compare the quantitative performance of different types of methods, three common metrics have been used:

- Global accuracy (GA), which measures the percentage of correctly categorized pixels in the dataset [45] which is given by:

$$GA = \frac{P_C}{N} \times 100\% \qquad (1)$$

where P_c is the number of pixels correctly categorized and N is the total number of pixels in the image.

- Class Accuracy (CA), which measures the average prediction accuracy over all classes [45] which is given by:

$$CA = \frac{1}{M}\sum_{i=1}^{M}\frac{P_{c_i}}{P_{t_i}} \times 100\% \qquad (2)$$

where P_{ci} is the number of correctly categorized pixels in the ith class and P_{ti} is the total number of pixels in the ith class of the image with M different classes.

Mean Intersection Over Union (mIoU), which is used in the Pascal VOC12 challenge [40]. If A_i shows the segmented region for ith class in ground truth image and B_i shows the prediction for the segmented region for ith class according to the used algorithm, mIoU for M classes is calculated by [45]:

$$mIoU = \frac{1}{M}\sum_{i=1}^{M}\frac{A_i \cap B_i}{A_i \cup B_i} \qquad (3)$$

The mIoU criterion is more precise than the average class accuracy since it penalties pseudo-positive predictions.

4. Experiments and Results

As already mentioned, proposed network is being trained and tested by CamVid dataset images. Table 2 shows the comparison between the semantic segmentation accuracy for proposed method and some of the known algorithms for the CamVid dataset. As shown in Table 2, proposed algorithm has been able to perform more successfully than other methods in semantic segmentation. Improving the performance of this technique is particularly noticeable in the segmentation of objects with a small number of pixels, such as the fence, pole, bicyclist, and sign symbols. This is because of the low network depth and less use of the max-pooling layers. Because, this layer is actually a kind of drop in the map of features, which results in loss of information and image clarity.

Table 2. Comparison between different algorithms for the percentage of class accuracy for each class of CamVid dataset.

Architecture	sky	building	pole	road	pavement	tree	sign symbol	fence	car	pedestrian	bicyclist
SegNet [17]	96.1	89.6	32.1	96.4	93.3	83.4	52.7	53.45	87.7	62.2	36.5
SqueezeNet [38]	94.5	88.9	36.9	93.6	93.6	75.5	19.8	1	97.7	64.4	67.6
Super Parsing [41]	96.9	87	1.7	95.9	70	67.1	30.1	17.9	62.7	14.7	19.4
Proposed method	97.8	92.3	**70.2**	**97.7**	91.6	**91.1**	**70.6**	**79.8**	94.0	**81**	**82.3**

Table 3 also shows the comparison between proposed method and several other methods after performing 40,000 iterations for training. To demonstrate proposed network convergence rate compared to other methods, the overall performance of proposed method and other methods for the CamVid dataset after 40,000 iterations of training in all methods is shown in Table 3. As is clear from the results, proposed method has achieved a better accuracy compared to other methods. Except for proposed method, other results have been adopted from the SegNet article [17].

Table 4 shows the values of the comparison criteria in Table 3 with the maximum number of iterations for the best response, according to the SegNet article. This is while proposed method is only trained for 60,000 iterations. The results of the table represent the convergence rate and achievement of higher accuracy in all criteria for proposed method than the other methods. Obviously, if the number of training iterations increases the better results will be achieved.

5. Conclusion

Proposed method is a convolutional neural network architecture based on SegNet, successful architecture of encoder and encoder components. The purpose of this network design is to reduce the amount of computational cost and memory required to process and increase speed, while at the same time, the increase in the accuracy of the training and testing of the network. Therefore, due to a 15-times reduction in the number of parameters compared to the SegNet network and achieving higher accuracy than other methods in all criteria, after only 60,000 replications of the network training because of the low volume of the database, the efficiency of proposed method has been improved in both accuracy and speed.

Table 3. Comparison between proposed method and several other methods after performing 40,000 iterations for training

Architecture	GA	CA	mIoU
SegNet [17]	88.81	59.93	50.02
DeepLab-LargeFOV [19]	85.95	60.41	50.18
FCN [16]	81.97	54.38	46.59
FCN (learnt deconv) [16]	83.21	56.05	48.68
DeconvNet [42]	85.26	46.40	39.69
Proposed method	89.49	83.76	62.65

Table 4. Comparison between proposed method and several other methods after performing the maximum number of iterations for the best response for training

Architecture	GA	CA	mIoU	Iterations 1000×
SegNet [17]	88.81	59.93	50.02	140
DeepLab-LargeFOV [19]	85.95	60.41	50.18	140
FCN [16]	81.97	54.38	46.59	200
FCN (learnt deconv) [16]	83.21	56.05	48.68	160
DeconvNet [42]	85.26	46.40	39.69	260
FC-DenseNet67 [43]	90.8	-	65.8	-
G-FRNet [44]	90.8	-	68.0	-
Proposed method	91.18	84.64	65.94	60

References

[1] A. a. M. Ess, Tobias and Grabner, Helmut and Gool, Luc van, “Segmentation-Based Urban Traffic Scene Understanding,” Proceedings of the British Machine Vision Conference, pp. 84.1-84.11, 2009.

[2] A. Geiger, “Are we ready for autonomous driving? the kitti vision benchmark suite,” in 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3354–3361, 2012.

[3] 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 2016 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3213–3223, 2016.

[4] M. Oberweger, P. Wohlhart, and V. Lepetit, “Hands Deep in Deep Learning for Hand Pose Estimation,” CoRR, vol. abs/1502.06807, 2015.

[5] Y. Yoon, H.-G. Jeon, D. Yoo, J.-Y. Lee, and I. S. Kweon, “Learning a Deep Convolutional Network for Light-Field Image Super-Resolution,” in IEEE International Conference on Computer Vision Workshop (ICCVW), Santiago, Chile, pp. 57-65, 2015.

[6] J. Wan, D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. Zhang, and J. Li, “Deep Learning for Content-Based Image Retrieval: A Comprehensive Study,” in Proceedings of the 22nd ACM international conference on Multimedia, Orlando, Florida, USA, pp. 157-166, 2014.

[7] F. Ning, D. Delhomme, Y. LeCun, F. Piano, L. Bottou, and P. E. Barbano, “Toward automatic phenotyping of developing embryos from videos,” Transaction of Image. Processing, vol. 14, no. 9, pp. 1360-1371, 2005.

[8] D. C. Cire, #351, an, A. Giusti, L. M. Gambardella, #252, and r. Schmidhuber, “Deep neural networks segment neuronal membranes in electron microscopy images,” in Proceedings of the 25th International Conference on Neural Information Processing Systems, Vol. 2, Lake Tahoe, Nevada, pp. 2843-2851, 2012.

[9] C. Farabet, C. Couprie, L. Najman, and Y. LeCun, “Learning Hierarchical Features for Scene Labeling,” IEEE Transaction of Pattern Analysis, Machine Intelligence, vol. 35, no. 8, pp. 1915-1929, 2013.

[10] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous Detection and Segmentation," Computer Vision – ECCV 2014. pp. 297-312, 2014.

[11] S. Gupta, R. Girshick, P. Arbeláez, and J. Malik, "Learning Rich Features from RGB-D Images for Object Detection and Segmentation," Computer Vision – ECCV 2014. pp. 345-360, 2014.

[12] S. Bittel, V. Kaiser, M. Teichmann, and M. Thoma, “Pixel-wise Segmentation of Street with Neural Networks,” CoRR, vol. abs/1511.00513, 2015.

[13] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes: Active contour models,” International Journal of Computer Vision, vol. 1, no. 4, pp. 321-331, January 01, 1988.

[14] M. D. Levine, and S. I. Shaheen, “A Modular Computer Vision System for Picture Segmentation and Interpretation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 3, no. 5, pp. 540-556, 1981.

[15] T. K. Ho, “Random decision forests,” in Proceedings of the Third International Conference on Document Analysis and Recognition (Volume 1) - Vol. 1, pp. 278, 1995.

[16] E. Shelhamer, J. Long, and T. Darrell, “Fully Convolutional Networks for Semantic Segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence., vol. 39, no. 4, pp. 640-651, 2017.

[17] V. Badrinarayanan, A. Kendall, and RobertoCipolla, “ SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 39, pp. 2481-2495, 2017.

[18] F. Yu, and V. Koltun, “Multi-Scale Context Aggregation by Dilated Convolutions,” CoRR, vol. abs/1511.07122, 2015.

[19] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, “Semantic Image Segmentation with Deep Convolutional Nets and Fully Connected CRFs,” CoRR, vol. abs/1412.7062, 2014.

[20] 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,” CoRR, vol. abs/1606.00915, 2016.

[21] G. Lin, A. Milan, C. Shen, and I. D. Reid, “RefineNet: Multi-Path Refinement Networks for High-Resolution Semantic Segmentation,” CoRR, vol. abs/1611.06612, 2016.

[22] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, “Pyramid Scene Parsing Network,” 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6230-6239, 2017.

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

[24] L.-C. Chen, G. Papandreou, F. Schroff, and H. Adam, “Rethinking Atrous Convolution for Semantic Image Segmentation,” CoRR, vol. abs/1706.05587, 2017.

[25] G. J. Brostow, J. Fauqueur, and R. Cipolla, “Semantic object classes in video: A high-definition ground truth database,” Pattern Recogn. Lett., vol. 30, no. 2, pp. 88-97, 2009.

[26] A. Ahmed, K. Yu, W. Xu, Y. Gong, and E. Xing, "Training Hierarchical Feed-Forward Visual Recognition Models Using Transfer Learning from Pseudo-Tasks," Computer Vision – ECCV 2008. pp. 69-82, 2008.

[27] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Learning and Transferring Mid-level Image Representations Using Convolutional Neural Networks,” in Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1717-1724, 2014.

[28] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?,” in Proceedings of the 27th International Conference on Neural Information Processing Systems, vol. 2, Montreal, Canada, pp. 3320-3328, 2014.

[29] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, “ImageNet: A large-scale hierarchical image database,” 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248-255, 2009.

[30] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge,” International Journal of Computer Vision, vol. 115, no. 3, pp. 211-252, December 01, 2015.

[31] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” in Proceedings of the 25th International Conference on Neural Information Processing Systems, vol. 1, Lake Tahoe, Nevada, pp. 1097-1105, 2012.

[32] K. Simonyan, and A. Zisserman, “Very Deep Convolutional Networks for Large-Scale Image Recognition,” CoRR, vol. abs/1409.1556, 2014.

[33] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, “Going deeper with convolutions,” in 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Boston, MA, USA, pp. 1-9, 2015.

[34] K. He, X. Zhang, S. Ren, and J. Sun, “Deep Residual Learning for Image Recognition,” in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV, USA, pp. 770–778, 2016.

[35] M. D. Zeiler, G. W. Taylor, and R. Fergus, “Adaptive deconvolutional networks for mid and high level feature learning,” in Proceedings of the 2011 International Conference on Computer Vision, pp. 2018-2025, 2011.

[36] M. D. Zeiler, and R. Fergus, "Visualizing and Understanding Convolutional Networks," Computer Vision – ECCV 2014. pp. 818-833, 2014.

[37] A. Paszke, A. Chaurasia, S. Kim, and E. Culurciello, “ENet: {A} Deep Neural Network Architecture for Real-Time Semantic Segmentation,” CoRR, vol. abs/1606.02147, 2016.

[38] G. Nanfack, A. Elhassouny, and R. O. H. Thami, “Squeeze-SegNet: {A} new fast Deep Convolutional Neural Network for Semantic Segmentation,” CoRR, vol. abs/1711.05491, 2017.

[39] D. Eigen, and R. Fergus, “Predicting Depth, Surface Normals and Semantic Labels with a Common Multi-scale Convolutional Architecture,” in Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), pp. 2650-2658, 2015.

[40] 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,” International Journal of Computer Vision, vol. 111, no. 1, pp. 98-136, January 01, 2015.

[41] J. Tighe, and S. Lazebnik, "SuperParsing: Scalable Nonparametric Image Parsing with Superpixels," Computer Vision – ECCV 2010. pp. 352-365, 2010.

[42] H. Noh, S. Hong, and B. Han, “Learning Deconvolution Network for Semantic Segmentation,” in Proceedings of the 2015 IEEE International Conference on Computer Vision (ICCV), pp. 1520-1528, 2015.

[43] S. Jégou and M. Drozdzal and D. Vazquez and A. Romero and Y. Bengio, "The One Hundred Layers Tiramisu: Fully Convolutional DenseNets for Semantic Segmentation", in 2017 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), pp. 1175-1183, 2017.

[44] M. A. Islam, M. Rochan, N. D. Bruce, and Y. Wang, "Gated Feedback Refinement Network for Dense Image Labeling", in 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017.

[45] Z, Tan, B. Liu, N. Yu, "PPEDNet: Pyramid Pooling Encoder-Decoder Network for Real-Time Semantic Segmentation", in International Conference on Image and Graphics, pp. 328-339, 2017.

Hanieh Zamanian was born in Mashhad. She received the B.Sc degree in Communication engineering from Sadjad University of Technology, Mashhad, Iran in 2009. She then received M.Sc degree from University of Ferdowsi, Mashhad, Iran, in 2014. She is currently Ph.D. student in Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran. Her research interests include Image and Video Processing, Pattern Recognition, Machine Learning and Deep Learning.

Hassan Farsi received the B.Sc. and M.Sc. degrees from Sharif University of Technology, Tehran, Iran, in 1992 and 1995, respectively. Since 2000, he started his Ph.D in the Centre of Communications Systems Research (CCSR), University of Surrey, Guildford, UK, and received the Ph.D degree in 2004. He is interested in speech, image and video processing on wireless communications. Now, he works as professor in communication engineering in department of Electrical and Computer Eng., University of Birjand, Birjand, IRAN.

Sajad Mohamadzadeh received the B.Sc. degree in communication engineering from Sistan & Baloochestan, University of Zahedan, Iran, in 2010. He received the M.Sc. and Ph.D. degree in communication engineering from South of Khorasan, University of Birjand, Birjand, Iran, in 2012 and 2016, respectively. Now, he works as assistant professor in Faculty of Technical and Engineering of Ferdows, University of Birjand, Birjand, Iran. His area research interests include Image and Video Processing, Retrieval, Pattern recognition, Digital Signal Processing, Sparse Representation, and Deep Learning.