IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received August 20, 2019, accepted September 3, 2019, date of publication September 27, 2019, date of current version October 29, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2944219

A Novel Semantic Segmentation Algorithm Using a Hierarchical Adjacency Dependent Network

JIANJUN LI[1], JIE YU[1], DAN YANG[1], WANYONG TIAN[2], LULU ZHAO[2], AND JUNFENG HU[2]
[1]School of Computing Science and Technology, Hangzhou Dianzi University, Hangzhou 310018, China
[2]CETC Key Laboratory of Data Link Technology, Xi'an 710000, China

Corresponding author: Jianjun Li (jianjun.li@hdu.edu.cn)

This work was supported in part by the National Science Fund of China under Grant 61871170, and in part by the National Defense Basic Research Program under Grant JCKY2017210A001.

ABSTRACT Recent semantic segmentation networks mainly focus on how to fuse multi-level features from classification networks to improve segmentation accuracy. Some researches evenly emphasize the correlation of pixels in a global region, such as conditional random field (CRF). However, the strong correlation feature of pixels in a limited region is less considered in the previous researches and the remedy ability of the correlation of local pixels in semantic segmentation is severely ignored. To deal with this problem, we introduce a hierarchical adjacency dependent network (HadNet), in which an adjacency dependency module (ADM) is constructed by calculating and utilizing the impact fact of the pixel in different directions to classify the pixel. We explored the correlation of adjacent pixels and feature coverage in different feature levels to improve the segmentation accuracy. We evaluate our method on the popular Pascal VOC 2012 test set, and achieve a comparable result of mIOU accuracy of 79.8% with the state of art methods, such as DeepLabv3+ and Exfuse. Further, we discuss and analyze the data distribution of COCO dataset for deeply understanding the feature correlation and coverage in semantic segmentation.

INDEX TERMS Semantic segmentation, hierarchical adjacency dependent network, adjacency dependency module.

I. INTRODUCTION

Semantic segmentation under deep learning is often considered as a pixel-level classification task. Since the introduction of the fully convolutional network (FCN) [1] framework, many innovative works [2]–[8] have made a great progress based on the framework even though for web images, such as [9]. As a typical encoding-decoding structure, at first, the FCN-like architecture generates a higher semantic feature map by an encoder and then decode it into a segmentation result of the original resolution. In order to recover the missing details caused by convolution and pooling, many of the latest works [10]–[14] enrich the spatial information in higher semantic features by simply fusing a high-level feature layer with a low-level one. However, there are still drawbacks in that the prediction results are discontinuous, especially in edges of segmentation. As shown in Fig. 1(a), it is very easy for the human being to distinguish the wheel of the bicycle from the grass background, but it is over segmented apparently as shown in the figure. The car in Fig. 1(d) presents a regular metal texture, but the prediction result presents an overlap of the background and the car. The proposed experimental results show that these mispredicted pixels in segmentation may be effectively remedied when considering a major correlation of region features. It has been observed that region features of objects are often similar and are also semantically consistent. The features in a small region have a very similar representation, such as color, edge, contrast while texture, material and structure have a similar representation in a larger region. Pixels with similar representations are more strongly related and therefore semantically tend to be consistent. On the contrary, if the feature relationship of the pixels is weaker and they are less affected by each other so that it has higher possibility for them to belong to different categories.

Based on the above observation, we propose a hierarchical adjacency dependent network (HadNet), which is different with previously simply mixing high and low features to produce a discriminating feature. In order to avoid being constrained by high-level errors but strong semantic features, we abandon the introduction of the highest-level features and select the coarser segmentation result directly from the end

The associate editor coordinating the review of this manuscript and approving it for publication was Huazhu Fu.

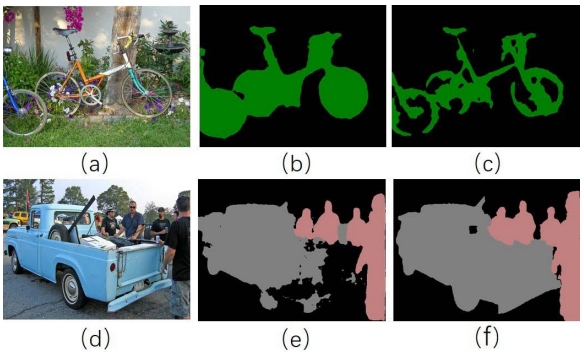


FIGURE 1. Examples of problematic semantic segmentation. The second column is the prediction result of DeepLabv3+. The third column is the prediction of our method.

of the encoder. Since the different stages of the convolutional neural network has different semantic features, which are related to the local representation of different ranges of objects. We model the pixel relations of low-level features by calculating the inter-pixel similarity so that some pixels with error classification can be re-segmented correctly.

Our contributions are as follows:

- Unlike previous works, in order to balance the impact of high-level semantics and low-level ones, we lower the impact of the high-level features and try to use low-level features to optimize segmentation results in the process of decoding of the segmentation.
- We propose a new decoder network structure, HadNet, to improve the segmentation result from a new perspective.
- The experimental results show that the proposed method obtains a comparable improvement in mIOU accuracy with 79.8% in Pascal VOC 2012 test set, compared with the state-of-the-art methods, e.g. DeepLabv3+ and Exfuse.

II. RELATED WORKS

Inspired by the progressive deep neural network [15]–[18], as a backbone network, the semantic segmentation task for extracting advanced features has also made great progress. Based on FCN, the model architecture of the fully connected layer in the classification is replaced by a convolutional layer. Many variants have also been derived from FCN. There are two main kinds of classifications:

A. ENCODER-DECODER

Deep neural networks can encode different levels of features. This type of model mainly utilizes the features of each level to gradually recover the missing spatial information due to pooling and convolution. Earlier SegNet [19] used saved pooling indexes to recover the reduced spatial information. U-Net [20] has a more regular network structure that splits each layer of the encoder's results into the corresponding decoder layer using skip links for better results. Recently, it has been considered that most of these types of methods are only a gradual fusion of features at each stage, ignoring the differences in their representation. A channel attention block has been proposed in [21] to change the weights of the features on each stage to emphasize the consistency. ExFuse [22] advocates the introduction of more semantic information into low-level features and embeds more spatial information into advanced features to bridge the gaps of semantic and resolution between low-level and high-level feature fusion, thereby improving the efficiency of feature fusion.

B. PYRAMID SPATIAL POOLING

Inspired by capturing multi-scale contextual convolutional features [23], enhanced multi-scale semantic segmentation by using dense connected conditional random fields (CRF) [24], spatial pyramid pooling [25], [26], DeepLabv2 proposes ASPP [27], which concatenates convolutions of different expansion rates in parallel to enhance multi-scale context aggregation for final prediction. Image-level branches [28], [29] are extended in DeepLabv3 [30] to further capture the global context. Similarly, PSPNet [31] uses four spatial pyramid pooling layers in parallel to aggregate information from multiple scales, which are then assigned to each pixel by upsampling to obtain a uniform resolution. Due to its excellent multi-scale feature fusion capability, it is used as an encoder in our network model.

However, most models treat semantic segmentation as an independent classification task of pixels, and the only connection between pixels is the overlap of its receptive fields. In fact, they are highly correlated for pixels in the image segmentation, and treating it as a purely independent pixel prediction will make the prediction result too blunt. Recently, more and more creatives for modeling the relationship between different pixels have been emerging. References [32], [33] proposes the self-attention mechanism to establish a spatial-wise relationship and learns the pixel-by-pixel global similarity map to estimate the target segmentation. Reference [34] proposes the concept of Adaptive Affinity Fields (AAF) in order to introduce the structural reasoning of labels directly into network modeling. The aforementioned works consider the relation between pixels globally. However, based on our observation that both the global correlation and local correlation between pixels have different impacts in semantic segmentation. The tradeoff between the impacts of global and local play an important role in different situations. Therefore, we firstly confirm that the loose constrains in the higher-level semantic is almost good enough in the semantic segmentation and then we pay more attention to the correlation of the adjacent pixels in a local region using the hierarchical feature of the network. We explore the adjacent dependencies in different levels of visual representation from a local perspective.

III. METHODS

In this section, we first introduce the proposed adjacency dependency module in details, including how to learn the adjacency correlation of features from different layers and

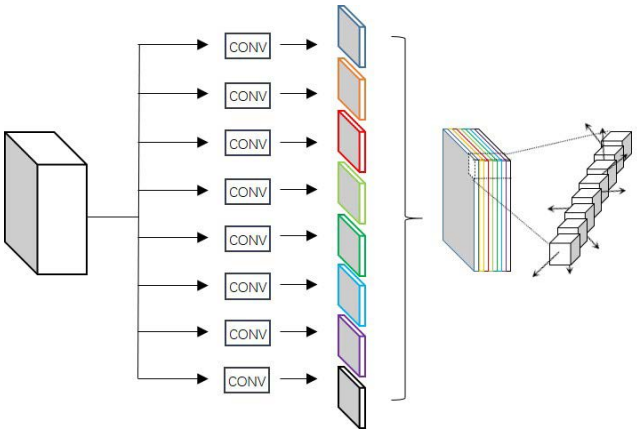


FIGURE 2. Overview of Adjacency Dependent Modules. Input is the extracted feature map. CONV is a 1x1 convolution kernel that specifies different directions of influence. By cascading the results obtained, each channel value at each point in space is a radiation value that affects the fixed direction.

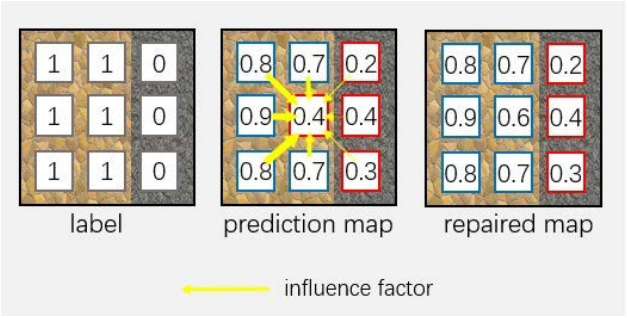


FIGURE 3. Adjacent pixels with similar local attributes are often consistent in the final classification. The numbers represent the prediction possibilities for a certain class. The blue and red boxes represent the types of predictions that are judged to be different. The thickness of the arrow represents the size of the impact factor.

improve the coarse prediction results. Then, we describe the complete encoder-decoder network architecture.

A. ADJACENCY DEPENDENT MODULE

The problem of predicting discontinuously is mainly due to the fact that the final classification for each pixel have a wide range of receptive fields. It is a comprehensive and abstract expression of a wide region and it is impossible to notice local detail representation. Especially in the junction situation, different object pixels are similar or overlapped in that the respective features are easily covered each other and thus lack spatial discrimination. Therefore, we need to further improve the spatial information through the local receptive domain. Considering the strong correlation between adjacent pixels, we utilize the feature outputs from different stages of the network to calculate the relationship between the pixel and its surround pixels in eight different directions. To determine that in which direction the pixel has the strongest dependency, the relationships in different directions can be calculated by using different convolution kernels. The dependency relationship of pixels in the entire image can be presented by a graph in which the nodes represent pixels and the weighted edges between the two nodes refer to the dependency between the two pixels. All nodes are divided into groups with the largest weighted edges inside a group and the smallest weighted edges between the groups. It should be noted that there are two weighted edges between two nodes and the edges has directions. We take the value of the weighted edge from node A to node B as the impact factor α of the pixel A on the pixel B. As shown in Fig. 2, the final feature map is with a channel number of 8 and each pixel carries eight impact factors referring to eight propagation directions.

Our Adjacency Dependent Module (ADM) aims to remedy the image details by enhancing the correlation between pixels. The final classification result of each pixel is obtained by weighting its surrounding pixels and the impact factors of its specific directions. The formula is as follows:

$$\overline{\alpha}_{i,j,c} = \frac{\alpha_{i,j,c}}{\sum_{i=a-1,j=b-1,c=0,i\neq a,j\neq b}^{i=a+1,j=b+1,c=7} |\alpha_{i,j,c}|} \tag{1}$$

$$\overline{\alpha}_{a,b} = 1 - \left| \sum_{i=a-1,j=b-1,c=0,i\neq a,j\neq b}^{i=a+1,j=b+1,c=7} \overline{\alpha}_{i,j,c} \right| \tag{2}$$

$$F_{a,b,k} = \sum_{i=a-1,j=b-1,c=0,i\neq a,j\neq b}^{i=a+1,j=b+1,c=7} \overline{\alpha}_{i,j,c} * P_{i,j,k} + \overline{\alpha}_{a,b} * P_{a,b,k} \tag{3}$$

where, $P \in R^{m\times n\times K}$ is a rough segmentation result and the number of channels is K, which represents a segmentation prediction map of the K-type of objects. $P_{-,-,k}$ refers to the segmentation prediction map of the $k-1th$ type. a, b refer to the spatial coordinates of a pixel. $\alpha \in R^{m\times n\times c}$ is the value of each channel of a point on the space and it represents the impact factor of the point on a point in a fixed direction. The central pixel of (a, b) is affected by the surrounding 8 pixels. Here we normalize these impact factors, such as Eq. 1, and obtain the confidence value of the central pixel according to Eq. 2, which is regarded as its impact factor on itself. Finally, as shown in the Eq. 3, the impact factors are weighted and summed with the spatially corresponding class prediction values, thereby obtaining the updated prediction values. We followed the steps of Eq. 3 for the prediction graphs of all classes.

It can be observed from the above formula that adjacent pixels with similar region features have large impact factors and thus, the prediction results have higher similarity while the pixel has less impact factor with others and it will maintain the original prediction result. The remedy result usually happens in those ambiguous pixels as shown in Fig. 3.

In fact, the above procedures are only the comprehensive evaluation results for each pixel in the range of 3x3 pixels. In order to expand the range of influence of the propagation, it is possible to repeat equation 3. Since all pixels are updated at the same time, the range of influence propagation of the pixels increases linearly with the number of iterations. However, it should be noted that our starting point is based on the strong correlation of pixels in the region, and an iteration more than 11 may generate downside effects.

B. PROPOSED DECODER

The backbone network has different recognition capabilities at different stages of extracting features. Just as humans' visual understanding of the world is multi-layered, humans not only recognize the whole objects immediately, but also identify details such as parts, textures, and materials. Today's identification network is often divided into several stages according to the size of the feature map, which can be considered as a multi-level perception of the entire picture, typically such as Resnet. In the higher level layer, the receptive field is large, the feature semantic is high, and the discrimination is strong. However, due to downsampling and spatial invariance, the prediction result is very coarse and abstract in spatial structure. In the lower level, the feature encodes more detailed attributes. Although the semantic feature is small due to the smaller receptive field and cannot be directly used for pixel classification, it contains more detailed spatial information and adjacent pixels with similar detail attributes are often consistent in the final classification, such as the continuous wooden material of the table and chair, the skin texture exposed on the human body and so on. In addition, since there are different semantic meanings in different feature layers, the parts and capabilities of remedying segmentation in different layers are thus different. Therefore, we can improve the segmentation accuracy by fusing features from layer to layer. In general, the influence of the high-level features is large, and it is better to be applied to remedy over-segmentation or under-segmentation than the lower-layer ones. On the contrary, the low-level features have a smaller influence range and are less affected by feature coverage, which is more suitable to be applied in details remedy. The remedy process of the whole segmentation must be one of from global to local, from wide to narrow and from coarse to details.

We follow the work of DeepLabv3+ [35] and the encoding structure is unchanged. We also use Xception [36] pre-trained on ImageNet [37] as the basic network, followed by the ASPP module for extracting multi-layer features. We re-designed the decoding structure and extract a number of low-level feature branches from different stages of the network. Through the adjacency dependent module introduced in Section 3.1, the coarse results after bilinear interpolation upsampling are repaired in a top-down manner. Several branches are "$entry\_flow/block2/unit\_1/conv$", "$middle\_flow/block1/unit\_16/conv$" (before striding), which refer to feature maps of $1/4$, $1/8$ of the original size respectively. The entire segmentation model architecture is shown as Fig. 4.

IV. EXPERIMENTAL RESULTS

We evaluate our approach on the public PASCAL VOC 2012 semantic segmentation benchmark [38]. The original dataset contains 20 object classes and one background, involving 1,464 images for training, 14, 456 images for validation and 1,456 images for testing. The dataset is augmented by [39], resulting in 10,582 images for training.

A. IMPLEMENTATION DETAILS

We follow the same training strategy as DeepLabv3+ during training.

1) TRAINING PHASE

In order to improve training efficiency, the encoder part is firstly trained. We apply $output\_stride = 16$ to train the encoder for 30K iterations with a batch size of 16 in the augmented dataset to further speed the training up. Here, we denote $output\_stride$ as the ratio of the input image spatial resolution to the final output resolution. Then, we fix the parameters of batch normalization and apply $output\_stride = 8$ to the 30K iterations for the entire network training on PASCAL VOC 2012 to make a refinement adjustment of the uneven learning rate. Finally, make the training for 10K iterations with a unified learning rate.

2) CROP SIZE

Zhao *et al.* [30] shows that large crop size is necessary to maintain the validity of the parameters of the dilated convolution using large rates without degrading its performance. A batch size of 513×513 pixels has been adopted on the training and the experimental results have shown that large crop size does improve accuracy.

3) LEARNING RATE POLICY

A learning rate strategy as "polynomial" has been adopted in the training, i.e $lr = base\_lr \times \left(1 - \frac{iter}{\max\_iter}\right)^{power}$. The power is set to 0.9. The initial learning rate, $base\_lr$, is set to 0.007 in the first 30K iterations and 0.00005 in the last 30K iterations. At the same time, the learning rate of the decoder part is set to 140 times of the initial learning rate. The initial learning rate for the last 1K iterations are 0.00005.

Data Augmentation: In order to augment the dataset during the training, we randomly scale the input images from 0.5 to 2.0 and mirror the images from left to right.

B. ABLATION STUDY

1) SINGLE BRANCH

In order to verify the performance of the ADM module, we have adopted the same encoder to extract low-level features as DeepLabv3+. Difference from DeepLabv3+ is that the simple cascade structure by combining high-level and low-level features to fill the details to increase the discrimination has been given up. Instead, a coarse segmentation result at the end of the feature extraction network of DeepLabv3 is directly output and then fused with the output of ADM as shown in Fig. 2. That is because that the prediction result by local correlation in low-level features is usually more reliable than the roughness caused by feature coverage of high-level semantic segmentation. This property that the low-level feature semantic is able to remedy the high-level segmentation map is based on the fact that the local feature of a pixel is predictable by its neighboring feature distribution of the lower-level features. Yu etc., [21] proposes that refinement residual blocks (RRBs) not only unify the number of output channels

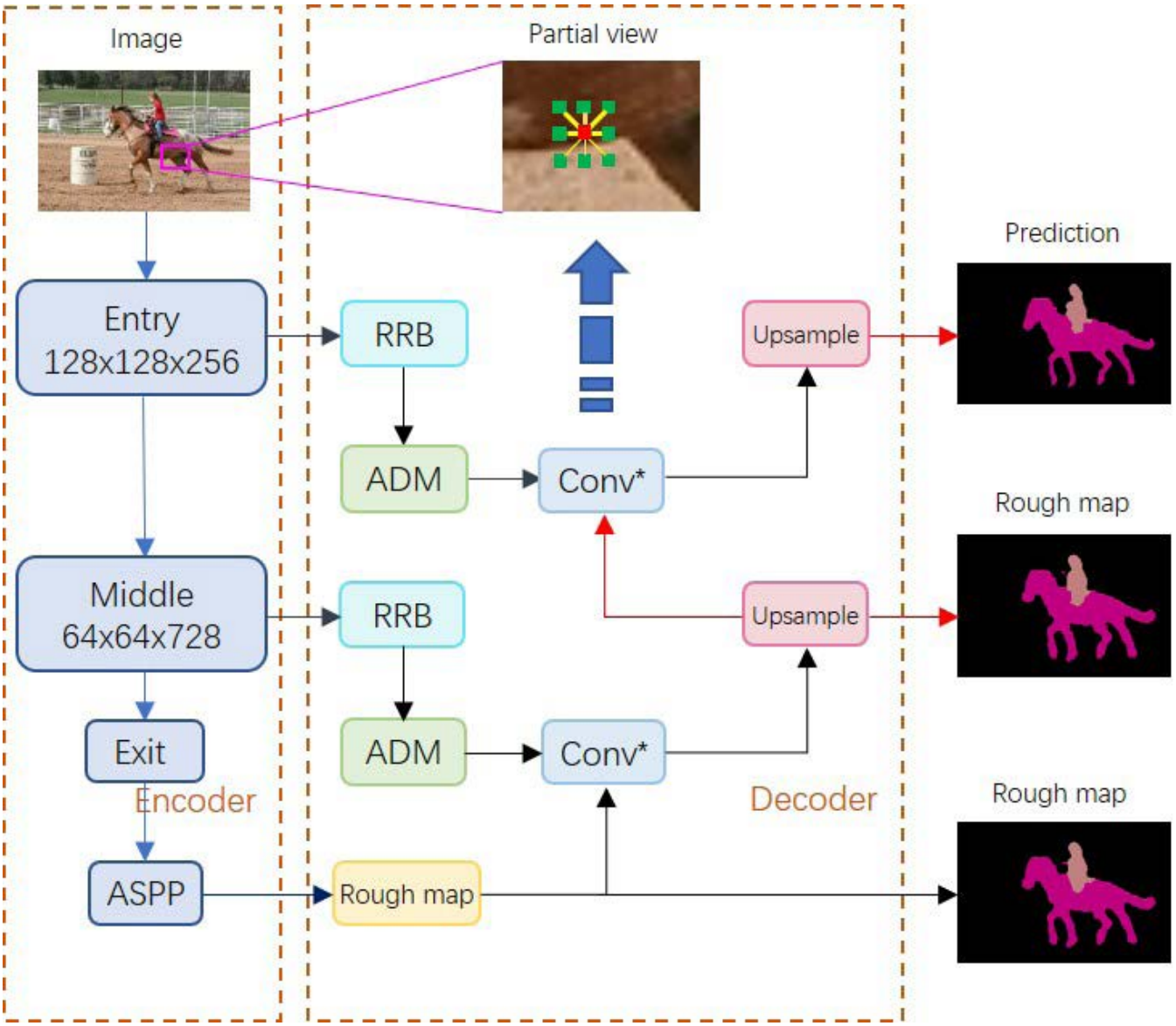


FIGURE 4. Hierarchical adjacency dependent network architecture. The ADM module is shown in Fig. 2. Conv* is a 3x3 convolution, but is not trainable. Initializes the value in the specified channel of the corresponding pixel in the ADM module. The blue line refers to the data flow direction in the feature extraction network with a downsampling operation. The red line refers to the upsampling. The black line does not change the feature map size. *Entry*, *Middle*, and *Exit* are the three parts of the Xception network. *E*.2, *M*.16 are the data streams derived in the 2nd and 16th units of the corresponding part, respectively.

TABLE 1. Comparison of performances of backbone+E.2 with and without RRB on VOC 2012 val set.

Method	RRB	mIOU
backbone+E.2		80.43%
backbone+E.2	✓	80.94%

into a unified number, say 512, but also refine and capture the multi-scale feature maps coming from their previous stages. In our proposed design, the refined and unified feature maps from RRBs can be directly input to the ADM modules for predicting the weights of the adjacent pixels on the current pixels. The performance is improved from 80.43% to 80.94% as shown in Tab. 1.

Furthermore, experiment shows that increasing the iterations of remedy reasonably can further improve the segmentation accuracy as Equation 3. The reason is that the pixels remedied can be propagated in limited range. Appropriately increasing the number of iterations can significantly achieve better performance. In our experiment, the number "3" of iterations has the optimum result as shown in Tab. 2.

2) MULTI-BRANCH

We explore the relationship at different levels of features. The initials *E* and *M* to represent the entry and middle parts of the feature extraction network separately. The number refers to the unit number of each part, for example, *E*.2 represents the feature data from the second unit in the entry part. Since the

TABLE 2. Effect of remedy times on VOC 2012 val set.

Times	1	2	3	4	5
mIOU	80.43%	80.71%	80.94%	80.86%	80.84%

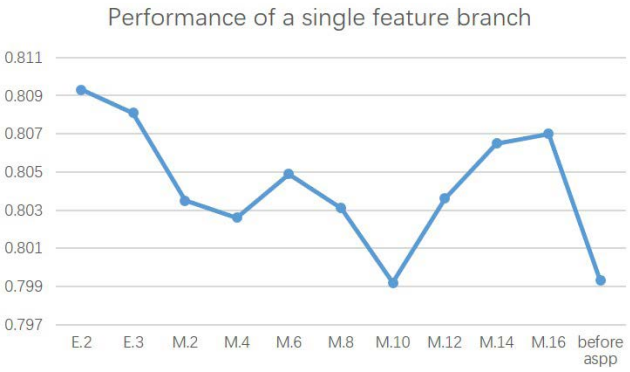


FIGURE 5. Individual remedy performance curves for different layers.

correlation of inter-pixel and intra-pixels are from the semantic correlation for different levels of features. High-level features contain abstract semantic while low-level features contain detailed semantics. It is apparent that it can improve the accuracy of semantic segmentation by adding multiple level features. However, the remedy capability of each layer is different and it is not always a positive remedy for all pixels due to the difference in the feature coding of each layer and the feature coverage effect of the feature itself. The remedy of each layer not only corrects some wrongly segmented pixels, but also makes some pixels worse. Therefore, sometimes it can not improve the segmentation accuracy only by simply adding feature layers. Fig. 5 shows the performance evaluated when each branch is utilized separately. The results obtained at E.2 and M.16 are the best. It is because that the features at E.2 are lower, such as textures and detailed features. Here, the pixel representation is tight and the correlation is strong in the local visual representation and therefore the credibility is higher. At M.16 point, unlike other mediate levels, the overall semantics of the various classes at this level are more uniform and the feature coding is more compact. As for the feature layer before ASPP, at this point, due to the multi-scale representation at the top of the network, the feature coverage is severe, the correlation reliability is not strong, and the effect of segmentation remedy is not better. Fig. 7 presents some examples of semantic segmentation results.

3) PERFORMANCE ON VOC 2012 VAL SET

A unsuccessful example is that the segmentation result was not improved when we applied both E.2 and E.3 simultaneously. The reason is that their semantic features from E.2 and E.3 are closer. We also observe that the performance is almost not improved even though the layers with similar semantics are superimposed because of their similar remedy ability and feature coverage. We also superimposed M.16 with the other middle layers. However, as shown in Tab. 3, we

TABLE 3. Performance comparison of multi-feature layer fusion on VOC 2012 val set.

M.16	M.6	E.3	E.2	mIOU
✓				80.70%
	✓			80.49%
		✓		80.81%
			✓	80.94 %
✓	✓			80.60%
✓		✓		81.02%
✓			✓	81.37%
		✓	✓	80.45%
✓	✓	✓		80.82%
✓	✓		✓	80.97%

TABLE 4. Performance comparison on VOC 2012 val set. MS: Adding multi-scale inputs. Flip: Adding left-right flipped inputs. COCO: Pre-trained on MS-COCO dataset. *E*.2, *M*.16 : The branch name used in decoder.

Decoder		MS	Flip	COCO	mIOU
E.2	M.16				
					79.64%
✓					80.94%
✓		✓			82.22%
✓		✓	✓		82.42%
	✓				80.70%
	✓	✓			81.97%
	✓	✓	✓		82.24%
✓	✓				81.37%
✓	✓			✓	84.86%
✓	✓	✓	✓		83.01%
✓	✓	✓	✓	✓	86.03%

did not find a particularly significant improvement. Finally, we reached the best performance by combining M.16 and E.2. This brings about an improvement of about 1.73% in mIOU compared with the only Resnet-like net used (79.64% in mIOU). The entire network framework is shown in Fig. 4. More evaluation details are shown in Tab. 4. We visualized the effects of HadNet. Fig. 6 presents some examples of semantic segmentation results. Obviously, our approach is more effective in restoring details. We achieve mIoU of 81.37% when training with 10582 images from PASCAL VOC 2012. In addition, we follow the procedure of pre-training on MS-COCO dataset [40]. We also apply the multi-scale inputs (with scales 0.5,0.75,1.0,1.5,1.75) and horizontally flip the inputs to further improve the performance. We eventually obtain a mIoU of 86.03% on PASCAL VOC 2012 validation set, which is 2.45% better than DeepLabv3+ and 0.23% better than Exfuse. Please find more detailed evaluation results in Tab. 4.

4) PERFORMANCE ON VOC 2012 TEST SET

As for evaluation on test set of PASCAL VOC 2012, we use the PASCAL VOC 2012 trainval set to further fine-tune our proposed model. The performance of 87.9% with MS-COCO fine-tuning has been achieved as shown in Tab. 5.

C. DISCUSSION

The reasons and phenomenon about the remedy capability of the proposed method are very interesting and we have to explore as the followings:

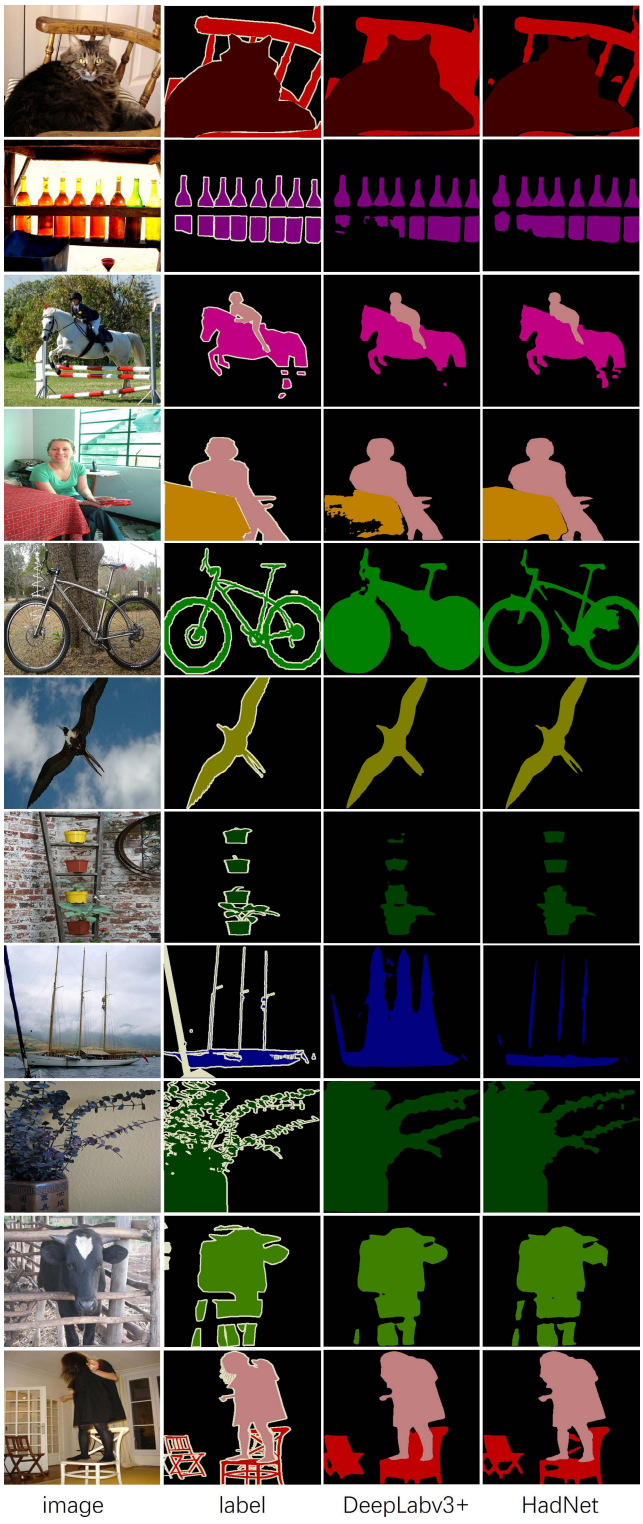


FIGURE 6. Example results on the PASCAL VOC 2012 val set.

1) GRADUAL REMEDY EFFECT OF SEGMENTATION OF BRANCHES

As described in Section. III-B, we consider the remedy capability of segmentation as a step-by-step process. We convert the output of prediction probability at each branch into heat maps as shown in Fig. 7. The stronger the color, the larger the probability value of the pixel's prediction as a non-background category. We can find that there are often fuzzy blocks inside or around the object, which have a lower prediction probability. After being processed by the M.16 branch, the prediction of the object tends to be more solid, which helps to alleviate the phenomenon of over-segmentation or under-segmentation. Furthmore, after the E.2 branch, the details predicted have been significantly improved, especially at the edges.

TABLE 5. Comparison of performances on PASCAL VOC 2012 test set.

Method	mIOU
Large Kernel Matters [10]	83.6%
Multipath RefineNet [9]	84.2%
PSPNet [30]	85.4%
Deeplabv3 [29]	85.7%
SDN [13]	86.6%
DFN [20]	86.2%
Deeplabv3+ [34]	87.8%
Ours	87.9%

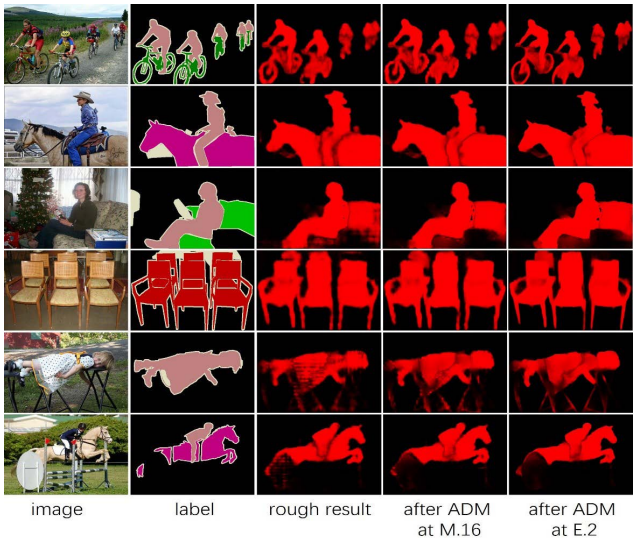


FIGURE 7. Example of remedy process from stage to stage on PASCAL VOC 2012 dataset.

2) SEGMENTATION WITH MULTI-BRANCH FUSION

Although the increasing the times of remedy can improve the segmentation accuracy according to Equation. 3, the infinite increase of the number of iterations can NOT always improve the accuracy and it will lower the segmentation accuracy since some previously correctly segmented pixels have been drawn back wrongly. Fig. 8 shows the statistics at each branch. Obviously, the overall effect will gradually become saturated as more branches are added. The blue, brown and green color refer to the number of remedied pixels of M.16, M.6 and E.2 separately and the yellow one refers to the sum of the remedied pixels. Apparently, the final remedied number of pixels is much lower than the sum of three when we applied M.16, M.6 and E.2 to the module simultaneously.

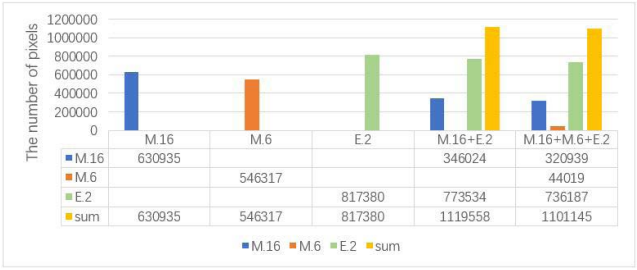


FIGURE 8. Example results in the stage-wise refinement process on PASCAL VOC 2012 dataset.

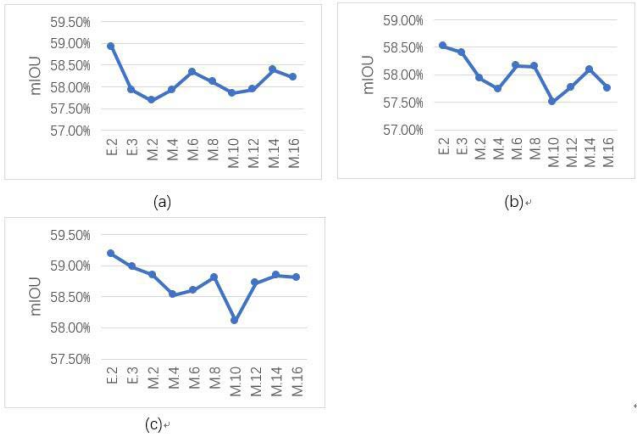


FIGURE 9. Comparison of segmentation accuracy of three random groups in different feature layers.

FIGURE 10. Comparison of segmentation accuracy of "people" and "bicycle" in different feature layers.

3) DATA DISTRIBUTION vs. SEGMENTATION

In order to further explore the correlation between the remedy capability and the data distribution, we randomly selected three sets of data sets in the COCO, each group has 500 pictures and the distribution of each category is even. We use the trained model to obtain the performance of each group as shown in Fig. 9. The overall trend is also roughly similar to Fig. 5. That means that the data distribution of VOC 2012 is relatively even.

4) CATEGORIES vs. SEGMENTATION

Besides, we also evaluated the datasets for a single category. Fig. 10 shows the performance evaluation curves of people and bicycles separately. The curves trend is completely different from that of Fig. 5. It can be seen that the performance improvement brought by each branch in Fig. 5 should be the result of the trade-offs of each category. The remedy capability of multi-branch fusion is related to different categories of datasets.

V. CONCLUSION

This paper proposes an efficient semantic segmentation net, HadNet, which is different with previous encoder-decoder structure. HadNet uses the DeepLabv3+ as the feature extractor to encode richer context information and directly outputs a coarse segmentation result. By the introduction of ADM module, HadNet takes advantage of pixel correlation in that the local feature correlation and the final prediction distribution is often consistent, and the higher-level semantic information can be remedied by the lower-level semantic information. Besides, the paper also explored the ability to remedy with multiple feature layers and the problem of feature coverage. The experimental results show that the proposed model outperforms most of the state of the art methods [21], [35] of mIOU accuracy on the PASCAL VOC 2012 test set.

REFERENCES

[1] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.

[2] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," 2015, *arXiv:1511.07122*. [Online]. Available: https://arxiv.org/abs/1511.07122

[3] N. Dvornik, K. Shmelkov, J. Mairal, and C. Schmid, "BlitzNet: A real-time deep network for scene understanding," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 4154–4162.

[4] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille, "Attention to scale: Scale-aware semantic image segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 3640–3649.

[5] 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 *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1529–1537.

[6] J. Dai, K. He, and J. Sun, "Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1635–1643.

[7] S. M. Glynn, "A contextual view of adult learning and memory," Tech. Rep., 1980.

[8] Z. Liu, X. Li, P. Luo, C.-C. Loy, and X. Tang, "Semantic image segmentation via deep parsing network," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1377–1385.

[9] C. Yan, L. Li, C. Zhang, B. Liu, Y. Zhang, and Q. Dai, "Cross-modality bridging and knowledge transferring for image understanding," *IEEE Trans. Multimedia*, vol. 21, no. 10, pp. 2675–2685, Oct. 2019.

[10] G. Lin, A. Milan, C. Shen, and I. Reid, "RefineNet: Multi-path refinement networks for high-resolution semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 1925–1934.

[11] C. Peng, X. Zhang, G. Yu, G. Luo, and J. Sun, "Large kernel matters—Improve semantic segmentation by global convolutional network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4353–4361.

[12] T. Pohlen, A. Hermans, M. Mathias, and B. Leibe, "Full-resolution residual networks for semantic segmentation in street scenes," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 4151–4160.

[13] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2015, pp. 1520–1528.

[14] J. Fu, J. Liu, Y. Wang, J. Zhou, C. Wang, and H. Lu, "Stacked deconvolutional network for semantic segmentation," *IEEE Image Process.*, to be published.

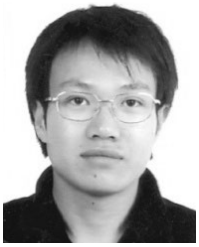
[15] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015, pp. 1–14.
[17] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. E. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1–9.
[18] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
[19] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 12, pp. 2481–2495, Dec. 2017.
[20] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. Int. Conf. Med. Image Comput. Comput.-Assist. Intervent.*, 2015, pp. 234–241.
[21] C. Yu, J. Wang, P. Chao, C. Gao, and S. Nong, "Learning a discriminative feature network for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2018, pp. 1857–1866.
[22] Z. Zhang, X. Zhang, P. Chao, X. Xue, and S. Jian, "ExFuse: Enhancing feature fusion for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 269–284.
[23] W. Yang, Q. Zhou, Y. Fan, G. Gao, S. Wu, W. Ou, H. Lu, J. Cheng, and L. J. Latecki, "Deep context convolutional neural networks for semantic segmentation," in *Proc. CCF Chin. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2017, pp. 696–704.
[24] Q. Zhou, W. Yang, G. Gao, W. Ou, H. Lu, J. J. Chen, and L. J. Latecki, "Multi-scale deep context convolutional neural networks for semantic segmentation," *World Wide Web*, vol. 22, no. 2, pp. 555–570, Mar. 2018.
[25] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 2, Jun. 2006, pp. 2169–2178.
[26] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2015.
[27] 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 Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2017.
[28] Q. C. Lin and S. Yan, "Network in network," 2013, *arXiv:1312.4400*. [Online]. Available: https://arxiv.org/abs/1312.4400
[29] W. Liu, A. Rabinovich, and A. C. Berg, "Parsenet: Looking wider to see better," 2015, *arXiv:1506.04579*. [Online]. Available: https://arxiv.org/abs/1506.04579
[30] L. C. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," 2017, *arXiv:1706.05587*. [Online]. Available: https://arxiv.org/abs/1706.05587
[31] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2016, pp. 2881–2890.
[32] H. Xu, S. Xie, L. Shu, and P. S. Yu, "Dual attention network for product compatibility and function satisfiability analysis," in *Proc. 32nd AAAI Conf. Artif. Intell.*, Apr. 2017, pp. 1–8.
[33] Y. Yuan and J. Wang, "OCNet: Object context network for scene parsing," 2018, *arXiv:1809.00916*. [Online]. Available: https://arxiv.org/abs/1809.00916
[34] T.-W. Ke, J.-J. Hwang, Z. Liu, and S. X. Yu, "Adaptive affinity fields for semantic segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 587–602.
[35] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 801–818.
[36] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2016, pp. 1251–1258.
[37] 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," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[38] 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," *Int. J. Comput. Vis.*, vol. 111, no. 1, pp. 98–136, Jan. 2014.
[39] B. Hariharan, P. Arbelaez, L. D. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 991–998.
[40] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollar, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.

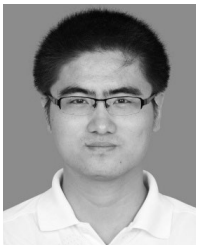
JIANJUN LI received the B.Sc. degree in information engineering from the Xi'an University of Electronic Science and Technology, Xi'an, China, the M.Sc. degree in electrical and computer from The University of Western Ontario, Canada, and the Ph.D. degree in electrical and computer from the University of Windsor, Canada. He is currently the Chair Professor with Hangzhou Dianzi University. His research interests include microelectronics, audio, and video and image processing algorithms and implementation.

JIE YU received the B.Sc. degree in communication engineering from Wenzhou University, Wenzhou, China. He is currently pursuing the master's degree with Hangzhou Dianzi University. His research interests include computer vision, artificial intelligence, and machine learning.

DAN YANG received the B.Sc. degree from the School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, China, where he is currently pursuing the M.Sc. degree. His research interests include machine learning, computer vision, and video and image processing algorithms and implementation.



WANYONG TIAN received the B.Sc. degree in software engineering from Northwest University, Xi'an, China, and the Ph.D. degree in computer science and technology from the University of Science and Technology, Hefei, China. His research interests include task scheduling algorithms and DEVS modeling.



LULU ZHAO received the B.Sc. degree in intelligence science and technology and the master's degree in control theory and control engineering from Nankai University, Tianjin, China. His research interest includes M&S about netcentric system of systems.

JUNFENG HU received the B.Sc. degree in electronic mechanical engineering and the M.Sc. and Ph.D. degrees in information and communication engineering from Xidian University, Xi'an, China, in 1997, 2004, and 2009, respectively. He is currently a Research Fellow with CETC, No.20 Research Institute. His research interests include the IoT and uRLLC.