IEEE Xplore

Notice to Reader

"Stacked Deconvolutional Network for Semantic Segmentation, by Jun Fu, Jing Lu, Yuhang Wang, Jin Zhou, Changyong Wang, and Hanqing Lu,
published in the ***IEEE Transactions on Image Processing*** **Early Access**

Digital Object Identifier: 10.1109/TIP.2019.2895460

This article will not be published in final form due to unauthorized changes made to the authorship following acceptance of the paper. It should not be considered for citation purposes.

We regret any inconvenience this may have caused.

Gaurav Sharma
Editor-in-Chief
IEEE Transactions on Image Processing

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

Stacked Deconvolutional Network for Semantic Segmentation

Jun Fu, Jing Liu, *Member, IEEE*, Yuhang Wang, Jin Zhou, Changyong Wang, and Hanqing Lu, *Senior Member, IEEE*

***Abstract*—Recent progress in semantic segmentation has been driven by improving the spatial resolution under Fully Convolutional Networks (FCNs). To address this problem, we propose a Stacked Deconvolutional Network (SDN) for semantic segmentation. In SDN, multiple shallow deconvolutional networks, which are called as SDN units, are stacked one by one to integrate contextual information and bring the fine recovery of localization information. Meanwhile, inter-unit and intra-unit connections are designed to assist network training and enhance feature fusion since the connections improve the flow of information and gradient propagation throughout the network. Besides, hierarchical supervision is applied during the upsampling process of each SDN unit, which enhances the discrimination of feature representations and benefits the network optimization. We carry out comprehensive experiments and achieve the new state-of-the-art results on four datasets, including PASCAL VOC 2012, CamVid, GATECH, COCO Stuff. In particular, our best model without CRF post-processing achieves an intersection-over-union score of 86.6% in the test set.**

***Index Terms*—Semantic Segmentation, Deconvolutional Neural Network, Dense Connection, Hierarchical Supervision.**

I. Introduction

SEMANTIC image segmentation has been one of the most important fields in computer vision, which is to predict the categories of individual pixels in an image. Recently, Deep Convolutional Neural Networks (DCNNs) [1] have strong learning ability to obtain high-level semantic features, and make remarkable advances in computer vision, including image classification [2], [3], [4], object detection [5], [6] and keypoint prediction [7], [8]. For semantic segmentation tasks, DCNNs based methods mainly utilize the architecture of Full Convolutional Networks (FCNs) [9] which usually adopt a certain pretrained classification network and output a probability map per class for an arbitrary-sized input. However, the classification network with downsampling operations sacrifices the spatial resolution of feature maps to obtain the invariance to image transformations. The resolution reduction results in poor object delineation and small spurious regions in segmentation outputs.

Many approaches have been proposed to solve the above problems. One way is to apply dilated convolutions [10], [11], [12], [13]. This type of solutions is to upsample the filters with different dilation factors, while the number of filter parameters per position remains unchanged. Accordingly, the receptive fields are enlarged and the larger contextual information is captured without losing the spatial resolution. However, these methods output a coarse sub-sampling feature maps which still lose details of object delineation. Moreover, many methods [14], [12], [11], [15] incorporate multi-scale or global feature maps to capture contextual information effectively and fit objects at multiple scales. Although this strategy further improves the receptive field and captures multi-scale information, it still outputs low-resolution feature maps which impede the generation of detailed boundaries.

Another type of methods is to recover the spatial resolution by an upsampling or deconvolutional path [16], [17], [18], [19], which can generate high-resolution feature maps for dense prediction. By improving upsampling process, the low resolution of the feature maps can be restored to the input resolution for pixel-wise classification, which is useful for accurate boundary localization. In the above upsampling solutions, the deconvolutional and unpooling layers are appended with a symmetric structure of the corresponding convolutional and pooling layers. Consequently, the parameter size of the new network is twice as large as the original convolutional structure. Furthermore, the deconvolutional networks are usually built by simply stacking layers, leading to the degradation problem when the depth increases [4]. To make the model easy to convergence, Wang et al. [16] used the VGG16 network [3] as pretrained weights to obtain better initial parameters of deconvolutional network, and Noh et al. [18] used a two-stage training strategy on single object images and multi-object images, respectively. However, expanding the above network to a deeper networks, unless use special designs likes skip connections [4] , is also restricted because of the difficulties on network training.

To overcome the above issues, in this paper, we propose a Stacked Deconvolutional Network (SDN) for semantic image

J. Fu is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China and the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: jun.fu@nlpr.ia.ac.cn).

J. Liu (corresponding author) is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, (e-mail: jliu@nlpr.ia.ac.cn).

Y. Wang is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China and the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: yuhang.wang@nlpr.ia.ac.cn).

J. Zhou is with Department of Neural Engineering and Biological Interdisciplinary Studies, Institute of Military Cognition and Brain Sciences, Academy of Military Medical Sciences, 27 Taiping Rd, Beijing 100850, PR China (e-mail: sisun819@yahoo.com).

C. Wang is with Department of Neural Engineering and Biological Interdisciplinary Studies, Institute of Military Cognition and Brain Sciences, Academy of Military Medical Sciences, 27 Taiping Rd, Beijing 100850, PR China (e-mail: wcy2000_zm@163.com).

H. Lu is with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: luhq@nlpr.ia.ac.cn).

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

segmentation. SDN is a deeper deconvolutional network but easier to optimize compared with the above deconvolutional networks. Instead of building a single deep encoder-decoder network, we design an efficient shallow deconvolutional network (called as SDN unit), and stack multiple SDN units one by one with dense connections, thus making the proposed SDN capture more contextual information and easily optimized. The designing details of the proposed architecture are shown in Fig. 1. A SDN unit is an encoder-decoder network. The encoder module is operated as a downsampling process, aiming to exploit multi-scale features and capture contextual information as much as possible, while the decoder module is operated as an upsampling process to recover the spatial resolution, aiming for accurate boundary localization. To benefit from the impressive performance of the pretrained deep model on ImageNet [20], we adopt DenseNet-161 [21] without its last downsampling operations as the encoder module of the first SDN unit, while other encoder and decoder modules consist of some simple downsampling blocks and upsampling blocks, respectively. Typically, each downsampling (upsampling) block includes a max-pooling layer (deconvolutional layer), several convolutional layers, and a compression layer, where the compression layer is a convolution operation for reducing the number of feature maps.

As the number of the stacked SDN unit increases, the difficulty on model training becomes a major problem. Two strategies have been taken to ameliorate the situation. First, hierarchical supervision is added to upsampling blocks of each SDN unit. Specifically, the compression layers are mapped to pixel-wise labeling maps by a classification layer. With this structure, the network could be trained in a more refined way while maintaining the discrimination ability as much as possible. Second, we import intra-unit and inter-unit connections to help the network training. The connections are shortcut paths from early layers to later layers, and they are beneficial to the flow of information and gradient propagation throughout the network. Within an upsampling or a downsampling block of a given SDN unit, the intra-unit connections is a short-range dense connection which is the direct link from the inputs of previous convolutional layers to the ones of back convolutional layers. The inter-unit connection is a long-range skip connection between certain two SDN units. Considering the different intentions for information transmission, there are two forms of inter-unit connections. One is to link decoder and encoder modules between any two adjacent SDN units to promote the network training. The other is to connect the multi-scale feature maps from the encoder module of the first SDN unit to the corresponding decoder module of each SDN unit, thus maintaining the low-level details for the high-resolution prediction. With the above two strategies, our proposed SDN can be trained efficiently and effectively, and achieves impressive performance on four popular benchmarks including PASCAL VOC 2012 dataset [22], CamVid dataset [23] , GATECH dataset [24] and COCO Stuff [25] . In particular, a Mean IoU score of 86.6% without CRF post-processing on PASCAL VOC 2012 test set, which is the best one on the official leaderborad to the time of our submission (29-July-2017).

Our main contributions can be summarized as follows:

- We propose a novel network (Stacked Deconvolutional Network) for semantic segmentation, which stacks multiple shallow deconvolutional networks to capture multi-scale context, and improve the recovery of localization information.
- Our proposed SDN adopts intra-unit and inter-unit connections to enhance the flow of information and gradients throughout the network. In particular, inter-unit connections make the multi-scale information across different units efficient to reuse.
- The hierarchical supervision for each SDN unit results in better training of the proposed network, since early layers of the network can obtain more gradient feedback. Besides, it enhances the discrimination of the feature maps from the upsampling process of each unit.
- Extensive experimental evaluations demonstrate that the proposed SDN model achieves new state-of-the-art performance on all the four benchmarks.

The remainder of this paper is organized as follows: In Section II, we mainly review the related work about DCNNs based methods for semantic image segmentation and scan the basic architecture of DenseNet [21]. In Section III, we will introduce our proposed Stacked Deconvolutional Network, including the designing details of a single SDN unit, the connections stacking multiple SDN units and the multi-scale hierarchical supervision. To verify the effectiveness of our work, the experimental evaluations and necessary analysis are presented in Section IV. Finally, we summarize our work in Section V.

II. Related Work

Recently, DCNNs based methods make great progress in pixel-level segmentation, including semantic segmentation and instance segmentation. The former aims to assign a class label to each pixel in an image, while the latter is more challenging since it not only assigns a class label to each pixel but also distinguish different objects in an image. Therefore, the recent methods of instance segmentation are mainly solved with a combination of object detection and semantic segmentation [26], [27], [28], represented by Mask-RCNN[27]. From this view, semantic segmentation is a primary problem, which is also our focus in this paper.

In the semantic segmentation task, the main methods are based on FCNs [9] which apply a fully convolutional structure and bilinear interpolation to realize pixel-wise prediction. However, the architecture of FCNs results in the problems of rough edges and object vanishing. Recently, many works try to alleviate these problems.

Some networks imported dilated convolutions and reduced down-sampling operations, which enable context aggregation and retain more spatial information. Deeplab [10] and DilatedNet [13] adopted dilated convolutions to enlarge receptive fields and captured larger contextual information without losing resolution. Wang et al. [29] employed hybrid dilated convolutions to further enlarge receptive fields of the network. Further on, Dai et al. [30] proposed a deformable convolution to adjust the receptive fields according to the scale of objects.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

The above methods removed some downsampling operations for increasing the resolution of outputs, which helps to generate detailed object delineation.

Some networks explored multi-scale or global features to capture contextual information for performance improvement. ParseNet [14] employed global pooling operations to extract image-level information. Deeplabv2 [12] proposed atrous spatial pyramid pooling (ASPP) to embed contextual information, which consists of parallel dilated convolutions with different dilated rates. PSPNet [15] designed a pyramid pooling module to collect the effective contextual prior, containing information with different scales. Deeplabv3 [11] proposed an augment ASPP module with image-level features to further capture global context. Graph learning [31], [32] is also an important way to improve recognition.

Some networks employed deconvolution operations in upsampling path, thus they are able to realize high-resolution prediction and obtain refined object delineation. OA-Seg [16] and SegNet [17] applied unpooling operations to upsample the low-resolution features and learn deconvolutional layers to improve the performance of upsampling. U-Net [33] exploited multi-level features by skip connections in a deconvolutional network. RefineNet [34] further refined coarse semantic features with fine-grained low-level features in a multi-path refined architecture. Yu et al. [35] adopted a channel mechanism to select the more discriminative features in decoder stage and employed additional edge information [36] to improve the details. Bilinski et al [37] introduced dense decoder shortcut connections for full multi-level feature fusion and effective information propagation. Ding et al. [38] proposed a context contrasted local feature and gating mechanism in decoder stage for improving inconspicuous objects and background stuff. In human pose estimation tasks, Newell et al. [8] stacked multiple encoder-decoder architectures to capture multi-scale information, where the nearest neighbor interpolation is employed in its upsampling path. DCDN [39], which is our previous work, stacked many shallow deconvolutional networks for enhancing the learning ability of the network.

In this work, we stack multiple shallow deconvolutional networks one by one to improve accurate boundary localization. Another important work [8] adopted hourglass-shaped structure with nearest neighbor upsampling process and achieves good performance in the pose estimation task. Different from their design, our proposed network for semantic segmentation adopt deconvolutional layer to learn refined recovery of the spatial resolution, hierarchical supervision to enhance the discrimination of the feature maps in the upsampling path, and the inter-unit connections between any two adjacent SDN units to promote full feature fusion and network optimization. Besides, compared with our previous work DCDN [39], we expand it by redesigning the deconvolutional network with intra-unit and inter-unit connections, introducing hierarchical supervision, and inheriting the ImageNet [20] pre-trained network. All these designs make our network produce more discriminative features and easy to optimize.

Our proposed SDN imports dense connections to make the stacked deep model easy to optimize. This is inspired by the work of DenseNet [21]. Accordingly, we will overview the basic idea of DenseNet in the following. DenseNet adopts dense connections to avoid the vanishing gradient problem and improve the flow of information, and it mainly consists of dense blocks and transition layers. The input of each convolutional layer within a dense block is the concatenation of all feature outputs of its previous layers at a given resolution. Consider x_l is the output of the l^{th} layer in a dense block, x_l can be computed as follows:

$$x_l = H_l\left([x_0, x_1, \ldots, x_{l-1}]\right) \tag{1}$$

where $[x_0, x_1, \ldots, x_{l-1}]$ stands for the concatenation of the feature maps $x_0, x_1, \ldots, x_{l-1}$, and x_0 is the inputs of the dense block. Meanwhile, H_l is defined as a composite function of operations: BN, ReLU, a 1×1 convolution operation followed by BN, ReLU, a 3×3 convolution operation. Each dense block is followed by a transition layer, which do convolution and pooling to change the number and the size of feature maps. Finally, a softmax classifier is attached to make prediction.

III. Our Approach

A. Overview

We propose a new framework called as Stacked Deconvolutional Network (SDN), which aims to capture more contextual information and recover high-resolution prediction progressively by stacking multiple shallow deconvolutional networks one by one.

As illustrated in Fig. 1, three (but not limited) deconvolutional networks, called as *SDN units*, are piled up from end to end. Such a stacked structure has two advantages. On one hand, it makes the network deep ,thus enhancing learning ability of the network. On the other hand, intermediate supervision can help training the network. Meanwhile, the network has more downsampling operations to capture contextual information, which benefits for object recognition. The network also have multiple decoder blocks, each decoder block take spatial information from shallow layers, which continuously refined object delineation and learn a feature at multiple scales. In addition, the intra-unit connections and the inter-unit connections are jointly employed. Such a connected way improves the flow of information and gradients throughout the network, which makes networks easy to train [21]. Meanwhile, it also enhances feature reuse for capturing rich multi-scale information and refining object segmentation edges. Moreover, we add hierarchical supervision during upsampling process of each deconvolution network. Motivated by previous research [40], we adopt deeply supervised structure to help the model training. Besides, such a supervision structure also suppresses the noises in the features from the shallow layers and produces accurate semantic predictions.

In order to obtain high-quality semantic representation and be consistent with the SDN unit in connection pattern, we adopt DenseNet-161, pre-trained on ImageNet [20], as the encoder module of the first deconvolutional network. In the inference step, we only use the highest-resolution results of the last unit as the final prediction. In the following subsections, we will elaborate the designing details of each SDN unit, intra-unit and inter-unit connections, and the hierarchical supervision.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

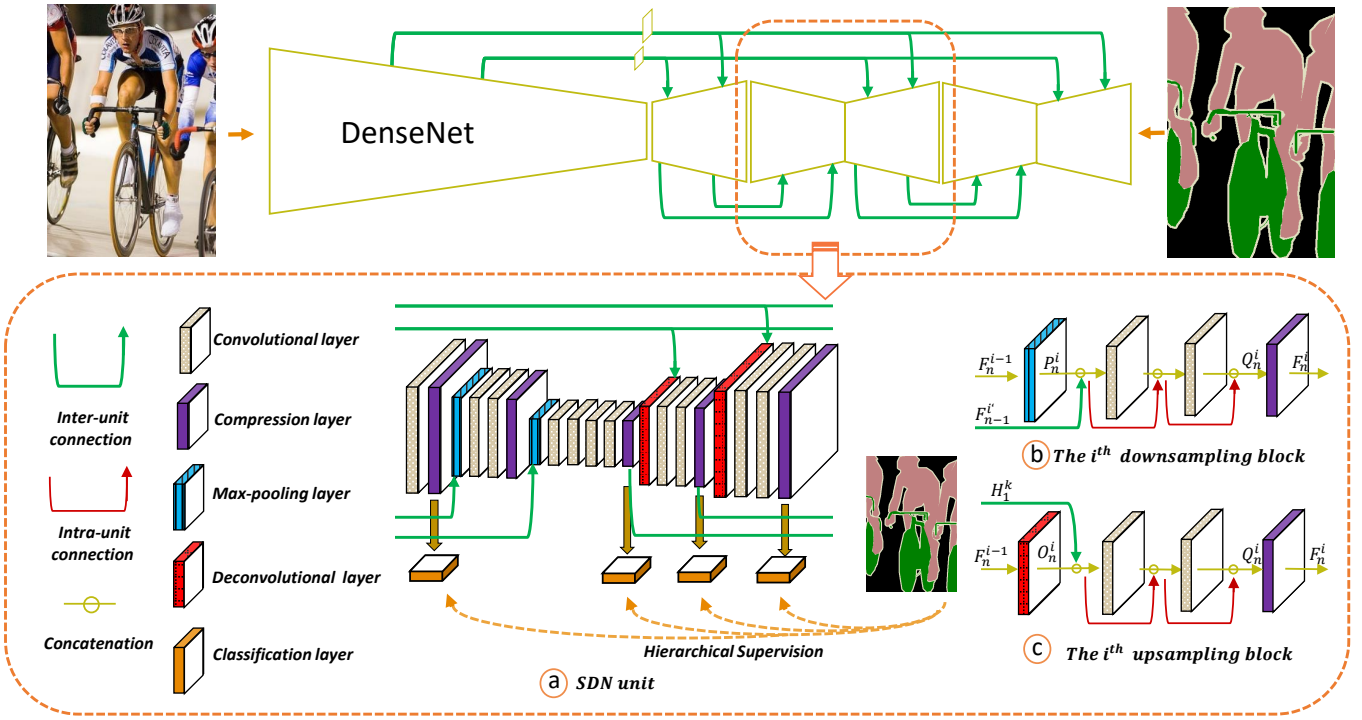

Fig. 1. Overall architecture of our approach. The upper part indicates the structure of proposed stacked deconvolutional network (SDN), the lower part indicates the detailed structure of the SDN unit (a), the downsampling block (b), the upsampling block (c). (Best viewed in color.)

B. Deconvolutional Network (SDN unit)

We design a shallow deconvolutional network referred to as a SDN unit to capture contextual information and refine poor object delineation, whose structure is illustrated in Fig. 1(a). One SDN unit has an encoder module and a corresponding decoder module. In a encoder module, we stack two downsampling blocks to enlarge the receptive fields of the network, resulting in the low resolution of the feature maps. In a decoder module, upsampling blocks are used twice to achieve a more refined reconstruction of the feature maps.

For a given SDN unit, its encoder module takes the outputs of its previous unit as inputs and produces low-resolution feature maps with larger receptive fields. Here, we employ downsampling blocks twice, resulting in $\frac{1}{16}$ spatial resolution of the input image. The structure of downsampling block is shown in Fig. 1(b). A downsampling block consists of a max-pooling layer and 2 (or more) convolutional layers, and a compression layer. First, the feature map F_n^{i-1} from the $(i-1)^{th}$ block is fed into the max-pooling layer in the i^{th} downsampling block of the n^{th} unit and sub-sampled by a factor of 2 to produce a new feature map P_n^i. Second, behind the max-pooling layer, we cascade 2 convolutional layers with intra-unit connections. Concretely, the input of the convolutional layer is the concatenation of the input and output of its previous convolutional layer. Such a densely connected structure is beneficial to feature reuse, i.e., the multi-scale appearance of objects can be better captured to obtain effective semantic segmentation. The densely connected structure takes the feature map P_n^i and the output $F_{n-1}^{i'}$ of the i'^{th} block in the $(n-1)^{th}$ unit as inputs and outputs an feature map Q_n^i, where the i'^{th} block is the backward nearest block to the i^{th} block in the same resolution. However, intra-unit connections bring the linear growth in the channel number of the feature maps, resulting in too much GPU demanding memory . Finally, in order to decrease the computational cost and the memory demanding, we perform a convolution with fewer filters, which we refer to it as a compression layer, to reduce the channel number of the feature map Q_n^i, to generate a feature map F_n^i. In short, the i^{th} downsampling block takes two feature maps F_n^{i-1} and $F_{n-1}^{i'}$ as its inputs, and outputs a new feature map F_n^i, the operations can be summarized as follows:

$$\begin{aligned} P_n^i &= Max(F_n^{i-1}), \\ Q_n^i &= Trans([P_n^i, F_{n-1}^{i'}]), \\ F_n^i &= Comp(Q_n^i). \end{aligned} \tag{2}$$

where $[P_n^i, F_{n-1}^{i'}]$ stands for the concatenation of the feature maps P_n^i and $F_{n-1}^{i'}$. $Max(\cdot)$ denotes a max-pooling operation. $Trans(\cdot)$ denotes a transformation function of the densely connected structure, in which two sequences of BN, ReLU, a 3×3 convolution, dropout, and concatenation operations are performed. $Comp(\cdot)$ refers to a 3×3 convolution operation. It should be noted that our encoder module of the first SDN unit employs full convolutional DenseNet-161 to obtain high-semantic features, while the other SDN units adopt the encoder module described above.

In the decoder module, we apply upsampling blocks to progressively upsample feature maps to larger resolution. The upsampling blocks are also used twice to enlarge resolution back to $\frac{1}{4}$ spatial resolution of the input image. An upsampling

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

block consists of a deconvolutional layer and several convolutional layers and a compression layer. As shown in Fig. 1(c), in the i^{th} upsampling block, we first apply deconvolutional operation on the output F_n^{i-1} of the $(i-1)^{th}$ block and produce a high-resolution feature map O_n^i. Then the feature map O_n^i is concatenated with the feature map H_1^k from the k^{th} block in the encoder module of the first SDN unit, where the k^{th} block have the same resolution with O_n^i. Finally, the concatenated feature maps are fed to the subsequent convolutional layers and compression layer, which adopt the similar connection structure as the downsampling block. The output F_n^i of the i^{th} upsampling block can be computed as follow:

$$\begin{aligned} O_n^i &= Deconv(F_n^{i-1}), \\ Q_n^i &= Trans([O_n^i, H_1^k]), \\ F_n^i &= Comp(Q_n^i). \end{aligned} \tag{3}$$

where $Deconv(\cdot)$ refers to a deconvolutional operation. For the last upsamlping block of the last SDN unit, we abandon its compression layer for better prediction. Note that, our highest resolution of SDN unit is set to a quarter of input images. One important reason for this design is that we can reduce GPU memory usage of a single SDN unit to stack more units.

Contrary to traditional deconvolutional networks [16], [18], [17], which have difficulty in network training and require additional aids, our deconvolutional network achieves great improvements in network training. First, our deconvolutional network is shallow encoder-decoder framework, which only includes two simple downsampling and upsampling blocks. Then, intra-unit connections are performed between convolutional layers, and this enables effective backward propagation of the gradients through a SDN unit. All these design improve end-to-end training of all network blocks.

C. Densely Connecting SDN units

To enhance the learning ability of network, we stack some shallow SDN units into a very deep model. Meanwhile, the inter-unit connections are imported to make the multi-scale information across different units efficient to reuse. As shown in Fig. 1, there are two types of inter-unit skip connections in the proposed framework. One is the skip connections between any two adjacent SDN units, and the other is a kind of skip connections from the first SDN units to others. In the following, we will introduce them in detail.

The skip connections between any two adjacent SDN units are used to promote the flows of high-level semantic information and improve the optimization of encoder modules. For a given SDN unit, its encoder module exploits the intermediate features of the decoder module in its previous unit by a shortcut path. Specifically, the feature map P_n^i from the i^{th} block of the n^{th} unit is concatenated with the feature map $F_{n-1}^{i'}$ from the i'^{th} block of the $(n-1)^{th}$ unit. Such a concatenation operation is shown in Fig. 1(b). We adopt the skip connection twice according to the number of downsampling blocks. With such skip connections, the gradients can be directly propagated to the previous unit, thus promoting the optimization of

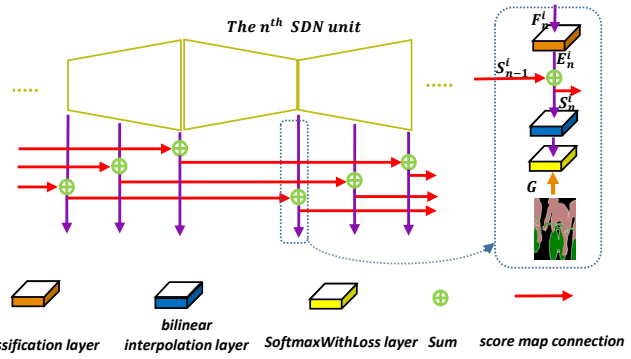

Fig. 2. Hierarchical supervision with score map connections during upsampling process. (Best viewed in color.)

network. Besides, the fusion of features from adjacent units would efficiently capture multi-scale information.

Meanwhile, we adopt the skip connections from the first SDN unit to others to fuse low-level representations and high-level semantic features, resulting in refined object segmentation edges. As illustrated in Fig. 1, we feed the low-level visual features from the k^{th} block of the first encoder module into a convolutional layer, where the convolution operations generate a feature map H_1^k. Then the feature map H_1^k is concatenated with the outputs of the deconvolutional layer in the corresponding resolution. Here, we combine the features from the first encoder module with the upsampling features from each decoder module. The reason is that the mid-level representations in first encoder module have more spatial visual information. Such inter-unit connections contribute to detailed boundaries for high-resolution prediction.

D. Hierarchical Supervision

Deeper networks lead to better performance. However, the difficulty in training deeper network becomes a major problem. We stack multiple shallow deconvolutional networks with random initialization, which leads to additional optimization difficulty. Previous work has suggested that short connections [4], [21] and auxiliary supervision [40] could help training very deep networks. In our network structure, inter-unit and intra-unit connections are used to assist training.

In order to alleviate this problem further, we add hierarchical supervision in each SDN unit. Specifically, three supervision branches is added at $\frac{1}{16}$, $\frac{1}{8}$, $\frac{1}{4}$ resolution in the upsampling process of each SDN unit, respectively. Each supervision branches is consist of a convolutional layer, a bilinear interpolation layer and a per-pixel cross-entropy loss layer. The convolutional layer makes per-pixel prediction and we refer to it as a classification layer. In the i^{th} supervision branch of n^{th} the SDN unit, the features from compression layer is fed to a classification layer to obtain a feature map E_n^i, Then E_n^i is upsampled to match the size of the input image with bilinear interpolation, and finally supervised with pixel-wise groundtruth. In short, the pixel-wise cross-entropy loss $\mathcal{L}^i$ in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

the i^{th} supervision branch of the n^{th} SDN unit is computed as follows:

$$\mathcal{L}^i = \ell(Bi(E_n^i), G), E_n^i = Classify(F_n^i) \tag{4}$$

where ℓ denotes a cross-entropy loss function, $Classify(\cdot)$ denotes a classifier with a 3×3 convolution operation, $Bi(\cdot)$ denotes a bilinear interpolation operation, and G refers to a ground truth map. With such the multiple additional supervision, the loss of the network would be a sum of the overall loss, and benefit network training. In addition, the noise from shallow features can be restrain by these additional supervision.

In order to further improve the performance of the network, we enhance score map fusion between two adjacent SDN units, depicted in Fig. 2. Specifically, in the i^{th} supervision branch of n^{th} the SDN unit, the output E_n^i of the classification layer are fused with the features S_{n-1}^i in the $(n-1)^{th}$ unit by element-wise sum operation to produce an new fused feature S_n^i, and then the fused feature S_n^i is upsampled with bilinear interpolation, and also supervised with pixelwise groundtruth. Therefore, we can compute the loss $\mathcal{L}^i$ as follows

$$\begin{aligned} \mathcal{L}^i &= \ell(Bi(S_n^i), G), \\ S_n^i &= E_n^i \oplus S_{n-1}^i \end{aligned} \tag{5}$$

where $\oplus$ denotes element-wise sum. Such a score map fusion not only enhances information flow, since the auxiliary supervision of the latter units could guide the learning of the previous units, but also improves final segmentation results by fusing multiple unit score maps

We carefully add supervision to different SDN unit and different resolution of each SDN unit, we find that above design of hierarchical supervision is better than only adding supervision to last SDN unit or the highest resolution of each SDN unit. In the testing phase, we only use the highest-resolution result of the last unit as the final prediction. It can be found that, with the hierarchical supervision, the feature maps restrain the noise and obtain effective end-to-end training during upsampling process.

E. Implementation Details

In each SDN unit, we stack 4 convolutional layers in the block of the lowest resolution for better global perspective, while the other blocks have 2 convolutional layers. The convolutional layers in a block is composed of BN, ReLU, and a 3×3 convolution operation followed by dropout with the probability of 0.2. The filter numbers of the convolutional layers are all set to 48. In two downsampling blocks, the compression layers are 3×3 convolution operations and their filter numbers are set to 768 and 1024 respectively. Max-pooling with 2×2 window and stride 2 is preformed. Meanwhile, in two upsampling blocks, the compression layers also are 3×3 convolution operations and their filter number are set to 768 and 576 respectively. Upsampling operation can be done by a 4×4 deconvolutional operation with stride 2. The convolutional layer in inter-unit connections from the first SDN unit to others is composed of BN, ReLU, and a 3×3 convolution operation. We employ the full convolutional DenseNet-161 network, pre-trained on ImageNet, as the first encoder module. Meanwhile, the last downsampling operations is removed and dilation convolutions is used.

The proposed SDN is implemented with Caffe [41]. Similar to [10], we optimize it using the "ploy" learning rate policy, with batchsize of 10. We set power to 0.9, momentum to 0.9 and weight decay to 0.0005. We apply data augmentation in the training step. Here, random crops of 320×320 are used, and horizontal flip is also applied. In the inference step, we pad images with mean value to make the length divisible by 16 before feeding full images into the network.

IV. Experiments

To show the effectiveness of our approach, we carry out comprehensive experiments on PASCAL VOC 2012 dataset [22], CamVid dataset [23], GATECH dataset [24] and COCO Stuff dataset [25]. Moreover, we perform a series of ablation evaluations to inspect the impact of various components on PASCAL VOC 2012 dataset. Experimental results demonstrate our proposed SDN achieves new state-of-the-art performance on four datasets.

A. Dataset and Evaluation Metrics

1) Dataset: PASCAL VOC 2012 : The dataset has 1,464 images for training, 1,449 images for validation and 1,456 images for testing, which involves 20 foreground object classes and one background class. Meanwhile, following the common procedure [9],we augment the training set with extra labeled PASCAL VOC images provided by Semantic Boundaries Dataset [42], resulting in 10,582 images for training.

CamVid: The dataset is a street scene understanding dataset which consists of 5 video sequences. Following [17], we split the dataset into 367 training images, 100 validation images, and 233 test images. The resolution of each image is 360×480 and all images belong to 11 semantic categories. Compared with PASCAL VOC 2012 dataset, CamVid dataset has more strong spatial-relationship among different categories.

GATECH: The dataset is a large video set of outdoor scenes which consists of 63 videos with 12241 frames for training and 38 videos with 7071 frames for testing. The dataset is labeled with 8 semantic classes which are sky, ground, solid, porous, cars, humans, vertical mix, and main mix.

COCO Stuff: The dataset is a recent scene understanding dataset which contains 10000 images from Microsoft COCO dataset [43], out of which 9000 images are for training and 1000 images for testing. This dataset contains 171 categories including objects and stuff annotated to each pixel.

2) Evaluation Metrics: We only report the results of Mean IoU on PASCAL VOC 2012 dataset for the common convention, while we evaluate other datasets with Global Avg and Mean IoU.

- Global Avg: Percentage of correctly classified pixels over the whole dataset. i.e. $\frac{\sum_i t_{ii}}{\sum_i T_i}$.
- Mean IoU: Ratio of correctly classified pixels in a class over the union set of pixels predicted to this class

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

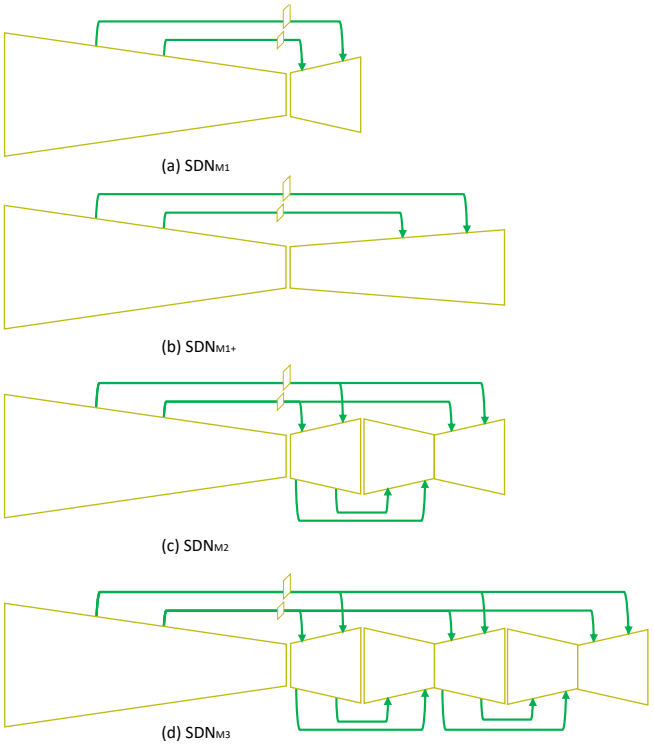

Fig. 3. Different stacked SDN structure. (Best viewed in color.)

and groundtruth, and then averaged over all classes. i.e. $\frac{1}{N}\sum_i \frac{t_{ii}}{T_i+\sum_j t_{ji}-t_{ii}}$.

where N is the number of semantic classes, and T_i is the total number of pixels in class i, while t_{ij} indicates the number of pixels which belong to class i and predicted to class j.

B. Results on PASCAL VOC 2012 dataset

In this subsection, we first verify the effectiveness of each component in our approach, including the effects of stacking multiple SDN units, stacked network design and hierarchical supervision. Then we report the experimental results on PASCAL VOC 2012 test set. Here, we set the initial learning rate to 0.00025, and further apply data augmentation by randomly transforming the input images with 5 scales {0.6, 0.8, 1, 1.2, 1.4}, and 5 aspect rations {0.7, 0.85, 1, 1.15, 1.3}.

1) Stacking multiple SDN units: We stack multiple SDN units to refine the segmentation maps. The key of the stacked architecture is that each unit can capture more contextual information and recover the high-resolution features. To verify this intuition, we gradually increase the number of the SDN units and test the performance respectively. As shown in Fig. 3, we refer to the network stacking k SDN units as SDN$_{Mk}$, and the number of the SDN units is up to 3.

The results are shown in Table I, it is observed that the performance consistently increases with the growth of SDN unit number. Specially, the performance of SDN$_{M1}$ network is only 78.2%. When we increase SDN unit number from 1 to 3, the performance improves from 78.2% to 79.2% to 79.9%. Meanwhile, we show some predicted semantic maps

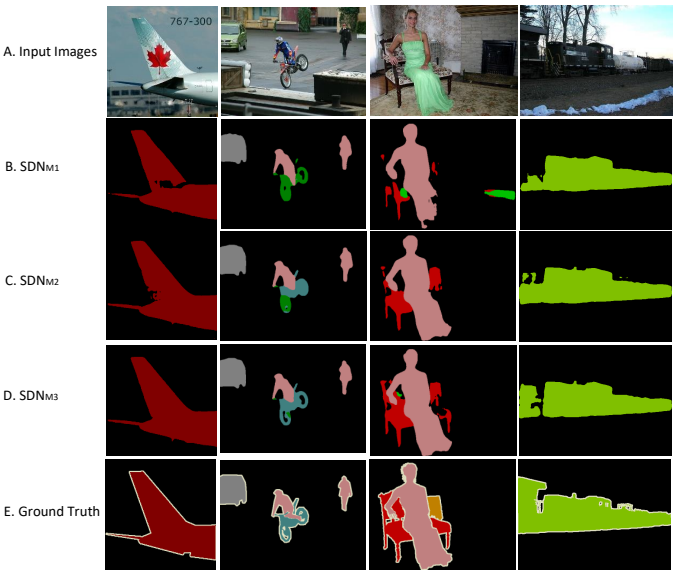

Fig. 4. Results on PASCAL VOC 2012 val set. For every column we list input images (A), the semantic segmentation results of SDN$_{M1}$ network (B), SDN$_{M2}$ network (C), SDN$_{M3}$ network (D), and Ground Truth (E).

under different networks in Fig. 4. With the growth of SDN unit number, the object edges are ameliorated (the first and fourth column), and the object discriminate is enhanced (the second and third column). These factors bring improvement of the semantic maps. The noticeable trend indicates that, with increasing number of the stacked units, the model benefits from the deeper network. Moreover, stacking multiple SDN units makes a coarse-to-fine prediction process, thus improving the network performance. However, it also leads to more computational stress and GPU demanding memory , thus we stack up to 3 SDN units as more units bring too slight improvements.

TABLE I
THE PERFORMANCE COMPARISON BETWEEN DIFFERENT NETWORKS ON PASCAL VOC 2012 VAL SET.

	SDN$_{M1}$	SDN$_{M2}$	SDN$_{M3}$	SDN$_{M1}+$
Depth	169	185	201	185
Parameters (M)	58.7	109.3	160.0	135.6
Mean IoU	78.2	79.2	79.9	78.6

2) Stacked network design: From the experiments above, the network depth and its parameter size increase with the number of stacked units. We want to explore that the improvements are mainly brought by more parameters or stacked network design. To address this problem, we design a single encoder-decoder network, named as SDN$_{M1}+$, which has the same network depth as the network stacking two SDN units, i.e., SDN$_{M2}$. SDN$_{M1}+$ also employ DenseNet-161 as the encoder module, meanwhile, the decoder module consists of 6 trivial convolutional layers and 2 upsampling blocks, the number of layers in upsampling block are set to 11 and 7 respectively. The results for the performance comparison between SDN$_{M1}+$ and SDN$_{M2}$ are listed in Table I. From

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

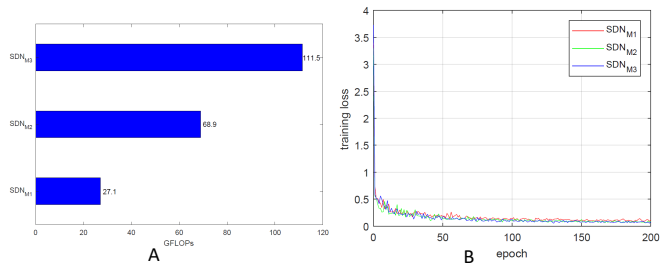

Fig. 5. Comparison of the three network about GFLOPs (crop 224×224) and the training curves on pascal voc 2012.

the results we can see that the performance of SDN_{M2} is 0.6 percent higher than $\text{SDN}_{M1}+$, although the former has a smaller size of parameters than the later. The comparison demonstrates that our structural design is the key factor of the performance of model.

Our design makes the network not only have a better performance but easy convergence. We provide the comparison of the three network about flops and the training curves on pascal voc 2012 as illustrated in Fig.5. The loss value is the average of all losses in networks. It reveals that our model could achieve a good convergence. In addition, although SDN_{M3} has more parameters, it has lower training loss than SDN_{M2} and SDN_{M1}.

3) Hierarchical supervision: To further alleviate the difficulty on optimization, we add hierarchical supervision to the upsampling process in each unit. In order to explore the impact of hierarchical supervision, we take SDN_{M1} as an example to test the performance of different supervision, Specifically, we modify SDN_{M1} network by adding supervision at different upsampling scales, and refer corresponding settings to as SDN_{M1_1}, SDN_{M1_2}, SDN_{M1} respectively. Here, we denote by *up_ratio* the ratio of the input image spatial resolution to the output resolution of the block. In SDN_{M1} network, we add supervision at *up_ratio* = 16, 8, 4. And for SDN_{M1_2} network, we add supervision at *up_ratio* = 8, 4. In SDN_{M1_1} network, we only add supervision at *up_ratio* = 4. In the inference step, we output the prediction at different *up_ratio*.

Results are shown in Table II, we can find that, in SDN_{M1} network, the performance at *up_ratio* = 4 is 2.3 percent higher than the performance at *up_ratio* = 16, it is clear that the high-resolution prediction, which use more low-level visual features, performs better than the low-resolution prediction during upsampling process of the unit. Meanwhile, the performance keeps increasing with the number of the supervision. the performance of SDN_{M1_2} is 0.5 percent higher than SDN_{M1_1}, and there is modest improvement in performance from 78.0 to 78.2 when we add supervision at all *up_ratio*. The results prove the effectiveness of hierarchical supervision, and we employ hierarchical supervision during upsampling process of each unit.

We also explore the effects of score map fusion, depicted in Fig. 2. We compare the performance between networks with or without score map fusion. To this end, we stack two SDN units without score map connection, and refer to the corresponding network as $\text{SDN}_{M2}-$. The results are shown in Table III, the

TABLE II
THE PERFORMANCE COMPARISON BETWEEN DIFFERENT SUPERVISION ON PASCAL VOC 2012 VAL SET.

	up_ratio	SDN_{M1_1}	SDN_{M1_2}	SDN_{M1}
Mean IoU	16	–	–	75.9
	8	–	77.1	77.7
	4	77.5	78.0	78.2

performance of SDN_{M2} is 0.4 percent higher than $\text{SDN}_{M2}-$. From the results we can see that the score map fusion is benefit for the segmentation result.

TABLE III
THE PERFORMANCE COMPARISON NETWORKS WITH OR WITHOUT SCORE MAP CONNECTIONS ON PASCAL VOC 2012 VAL SET.

	SDN_{M2}	$\text{SDN}_{M2}-$
Mean IoU	79.2	78.8

4) Some improvement strategies: In the section, detailed evaluations are performed on PASCAL VOC 2012 val dataset. Here, we adopt several steps to improve segmentation performance further based on the SDN_{M2} network: (1) UP: We further restore high resolution features by cascading a upsampling block, and we refer to the network as $\text{SDN}_{M2}*$. (2) MS_Flip: We average the segmentation probability maps from 5 image scales $\{0.5, 0.8, 1, 1.2, 1.4\}$ as well as their mirrors for inference. (3) COCO: For fair comparison with other state-of-the-art models, we also pretrain the model of $\text{SDN}_{M2}*$ on MS-COCO dataset [43]. We evaluate how each of these factors affects val set performance in Table IV.

TABLE IV
THE PERFORMANCE COMPARISON BETWEEN DIFFERENT MEASURES ON PASCAL VOC 2012 VAL SET.

Up	MS_Flip	COCO	Mean IoU
			79.2
✓			79.6
✓	✓		80.7
✓	✓	✓	84.8

Increasing an upsampling block gives 0.4% gain, and segmentation map fusion brings another 1.2% improvement. Moreover, when we pretrain the model of $\text{SDN}_{M2}*$ on MS-COCO dataset, the performance attains 84.8%. Compared with current well-known method Deeplabv3 [11](82.7% pre-trained on MS-COCO), our method of the $\text{SDN}_{M2}*$ outperforms theirs by 2.1%, which shows the strength of our proposed SDN network.

5) Comparing with the state-of-the-art methods : We further compare our method with the state-of-the-art methods on PASCAL VOC 2012 test set. Here, based on two settings,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

TABLE V
EXPERIMENTAL RESULTS ON PASCAL VOC 2012 TEST SET.

Method	aero	bike	bird	boat	bottle	bus	car	cat	chair	cow	table	dog	horse	mbike	person	plant	sheep	sofa	train	tv	mIoU
Only using VOC data																					
FCN[9]	76.8	34.2	68.9	49.4	60.3	75.3	74.7	77.6	21.4	62.5	46.8	71.8	63.9	76.5	73.9	45.2	72.4	37.4	70.9	55.1	62.2
DeepLabv2[12]	84.4	54.5	81.5	63.6	65.9	85.1	79.1	83.4	30.7	74.1	59.8	79.0	76.1	83.2	80.8	59.7	82.2	50.4	73.1	63.7	71.6
CRF-RNN[44]	87.5	39.0	79.7	64.2	68.3	87.6	80.8	84.4	30.4	78.2	60.4	80.5	77.8	83.1	80.6	59.5	82.8	47.8	78.3	67.1	72.0
DeconvNet[18]	89.9	39.3	79.7	63.9	68.2	87.4	81.2	86.1	28.5	77.0	62.0	79.0	80.3	83.6	80.2	58.8	83.4	54.3	80.7	65.0	72.5
GCRF[45]	85.2	43.9	83.3	65.2	68.3	89.0	82.7	85.3	31.1	79.5	63.3	80.5	79.3	85.5	81.0	60.5	85.5	52.0	77.3	65.1	73.2
DPN[46]	87.7	59.4	78.4	64.9	70.3	89.3	83.5	86.1	31.7	79.9	62.6	81.9	80.0	83.5	82.3	60.5	83.2	53.4	77.9	65.0	74.1
Piecewise[47]	90.6	37.6	80.0	67.8	74.4	92.0	85.2	86.2	39.1	81.2	58.9	83.8	83.9	84.3	84.8	62.1	83.2	58.2	80.8	72.3	75.3
ResNet38[48]	94.4	72.9	**94.9**	68.8	**78.4**	90.6	**90.0**	92.1	40.1	90.4	71.7	89.9	93.7	**91.0**	89.1	71.3	90.7	61.3	87.7	**78.1**	82.5
PSPNet[15]	91.8	71.9	94.7	71.2	75.8	95.2	89.9	95.9	39.3	90.7	71.7	90.5	**94.5**	88.8	89.6	**72.8**	89.6	**64.0**	85.1	76.3	82.6
SDN	**96.2**	**73.9**	94.0	**74.1**	76.1	**96.7**	89.9	**96.2**	**44.1**	**92.6**	**72.3**	**91.2**	94.1	89.2	**89.7**	71.2	**93.0**	59.0	**88.4**	76.5	**83.5**
Using VOC+COCO data																					
CRF-RNN[44]	90.4	55.3	88.7	68.4	69.8	88.3	82.4	85.1	32.6	78.5	64.4	79.6	81.9	86.4	81.8	58.6	82.4	53.5	77.4	70.1	74.7
Dilation8[13]	91.7	39.6	87.8	63.1	71.8	89.7	82.9	89.8	37.2	84.0	63.0	83.3	89.0	83.8	85.1	56.8	87.6	56.0	80.2	64.7	75.3
DPN[46]	89.0	61.6	87.7	66.8	74.7	91.2	84.3	87.6	36.5	86.3	66.1	84.4	87.8	85.6	85.4	63.6	87.3	61.3	79.4	66.4	77.5
Piecewise[47]	94.1	40.7	84.1	67.8	75.9	93.4	84.3	88.4	42.5	86.4	64.7	85.4	89.0	85.8	86.0	67.5	90.2	63.8	80.9	73.0	78.0
DeepLabv2[12]	92.6	60.4	91.6	63.4	76.3	95.0	88.4	92.6	32.7	88.5	67.6	89.6	92.1	87.0	87.4	63.3	88.3	60.0	86.8	74.5	79.7
RefineNet[34]	95.0	73.2	93.5	78.1	84.8	95.6	89.8	94.1	43.7	92.0	77.2	90.8	93.4	88.6	88.1	70.1	92.9	64.3	87.7	78.8	84.2
ResNet38[48]	96.2	75.2	95.4	74.4	81.7	93.7	89.9	92.5	48.2	92.0	79.9	90.1	95.5	91.8	91.2	73.0	90.5	65.4	88.7	80.6	84.9
PSPNet[15]	95.8	72.7	95.0	78.9	84.4	94.7	**92.0**	95.7	43.1	91.0	**80.3**	91.3	96.3	**92.3**	90.1	71.5	**94.4**	66.9	88.8	82.0	85.4
DeepLabv3[11]	96.4	76.6	92.7	77.8	**87.6**	96.7	90.2	95.4	47.5	93.4	76.3	91.4	**97.2**	91.0	**92.1**	71.3	90.9	**68.9**	**90.8**	79.3	85.7
EncNet [49]	95.3	76.9	94.2	**80.2**	85.3	96.5	90.8	96.3	47.9	93.9	80.0	92.4	96.6	90.5	91.5	70.9	93.6	66.5	87.7	80.8	85.9
SDN+	**96.9**	**78.6**	**96.0**	79.6	84.1	**97.1**	91.9	**96.6**	**48.5**	**94.3**	78.9	**93.6**	95.5	92.1	91.1	**75.0**	93.8	64.8	89.0	**84.6**	**86.6**

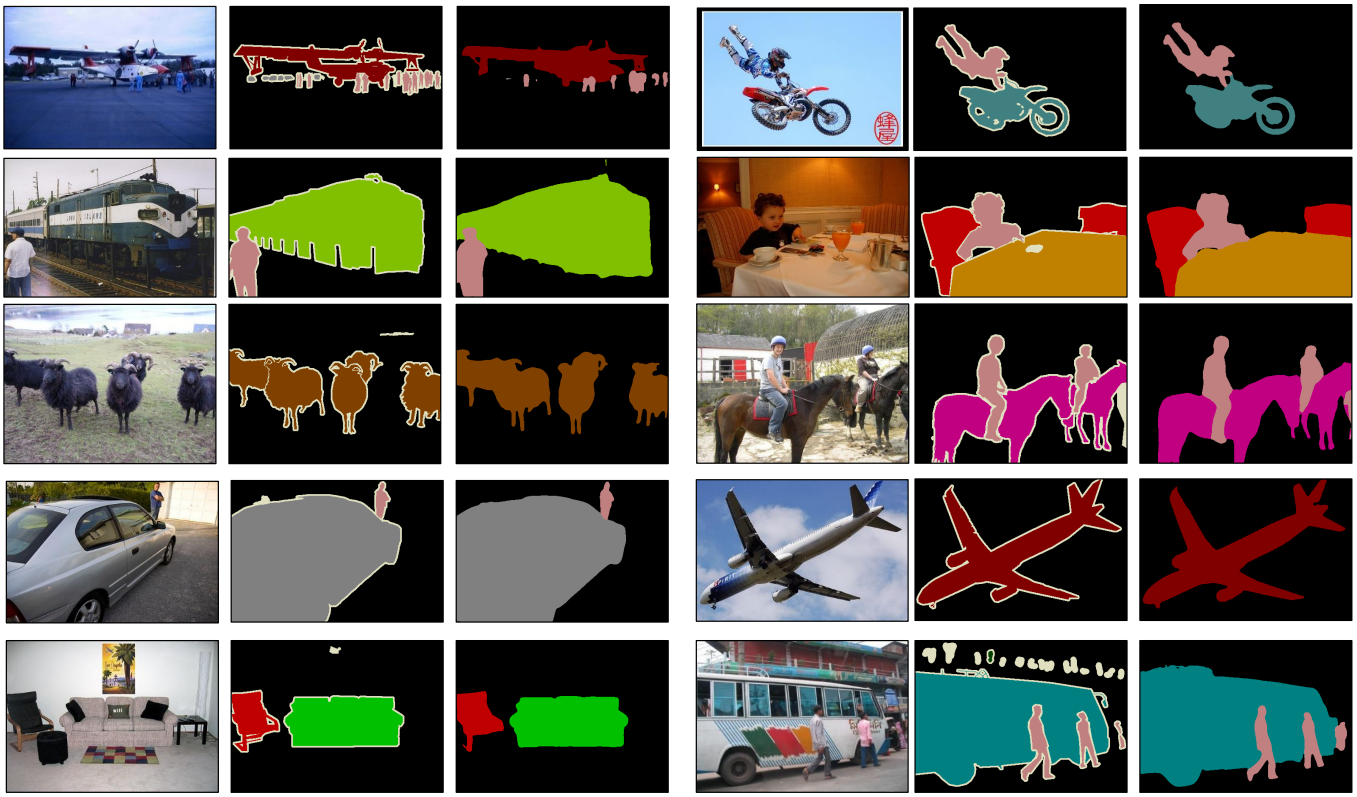

Fig. 6. Results on PASCAL VOC 2012 dataset. The images in each row from left to right are: (1) input image (2) groundtruth (3) semantic segmentation result.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

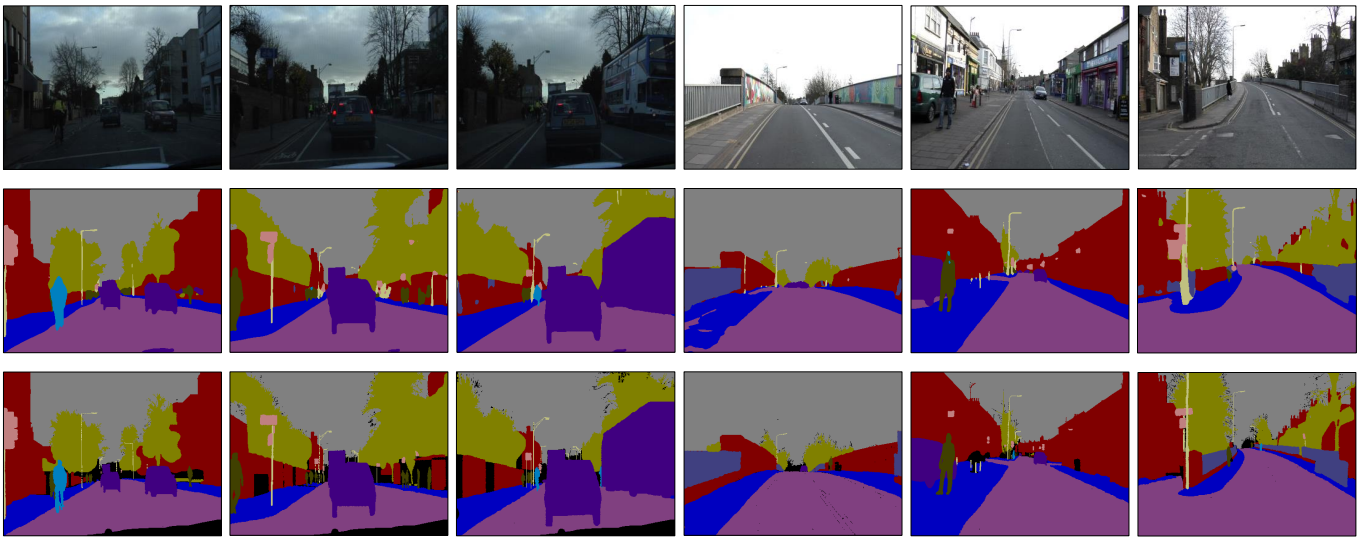

Fig. 7. Results on CamVid dataset. The images in each column from top to down are: (1) input image (2) semantic segmentation result (3) groundtruth.

TABLE VI
EXPERIMENTAL RESULTS ON CAMVID TEST SET.

Method	Building	Tree	Sky	Car	Sign	Road	Pedestrian	Fence	Pole	Sidewalk	Bicyclist	Mean IoU	Global Avg
SegNet[17]	68.7	52.0	87.0	58.5	13.4	86.2	25.3	17.9	16.0	60.5	24.8	46.4	62.5
DeconvNet[18]	–	–	–	–	–	–	–	–	–	–	–	48.9	85.6
ReSeg[50]	–	–	–	–	–	–	–	–	–	–	–	58.8	88.7
DeepLab-LFOV[10]	81.5	74.6	89.0	82.2	42.3	92.2	48.4	27.2	14.3	75.4	50.1	61.6	–
Bayesian SegNet[19]	–	–	–	–	–	–	–	–	–	–	–	63.1	86.9
Dilation8[13]	82.6	76.2	89.0	84.0	46.9	92.2	56.3	35.8	23.4	75.3	55.5	65.3	79.0
HDCNN-448+TL [51]	–	–	–	–	–	–	–	–	–	–	–	65.6	90.9
Dilation8+FSO[52]	84.0	77.2	91.3	85.6	49.9	92.5	59.1	37.6	16.9	76.0	57.2	66.1	88.3
FC-DenseNet103[53]	83.0	77.3	**93.0**	77.3	43.9	94.5	59.6	37.1	37.8	82.2	50.5	66.9	91.5
G-FRNet[54]	82.5	76.8	92.1	81.8	43.0	94.5	54.6	**47.1**	33.4	82.3	59.4	68.0	–
DCDN[39]	–	–	–	–	–	–	–	–	–	–	–	68.4	91.4
SDN	84.45	76.9	92.2	88.2	51.6	93.4	62.4	37.0	37.5	77.8	64.4	69.6	91.7
SDN+	**85.2**	**77.5**	92.3	**90.2**	**53.9**	**96.0**	**63.8**	39.8	**38.4**	**85.36**	**66.9**	**71.8**	**92.7**

i.e., with or without pre-trained with MS-COCO dataset, we fine tune our the model of SDN_{M3} on PASCAL VOC 2012 trainval set, and submit our test results to the official evaluation server. Results are shown in Table V. In the two settings, our method both outperforms all the other methods. Trained with only PSACAL VOC 2012 data, we achieve a Mean IoU score of 83.5% [1]. When we pretrain the model of SDN_{M3} on MS-COCO dataset, it reaches 86.6% [2] in Mean IoU. Specifically, our model outperforms the RefineNet [34] by 2.4%, and RefineNet employs a deep encoder-decoder structure and refines results with DenseCRF post-processing. Meanwhile, our model also outperforms the Deeplabv3 [11] by 0.9% and EncNet [49] by 0.7% under the equivalent training data. In addition, Deeplabv3 improves segmentation accuracy remarkably by bootstrapping method for rare and finely annotated classes. EncNet focuses on capturing the global context of scenes and selectively highlights class-dependent feature maps by an attention mechanism. It should be noticed that DenseCRF post-processing, the bootstrapping method and the attention mechanism are all not applied to our model. These comparisons indicate that our proposed SDN network can more effectively capture contextual information and generate accurate boundary localization.

[1]http://host.robots.ox.ac.uk:8080/anonymous/Z9RDVZ.html
[2]http://host.robots.ox.ac.uk:8080/anonymous/GRWV3B.html

C. Results on CamVid dataset

In this subsection, we carry out experiments on the CamVid [23] dataset to further evaluate the effectiveness of our method. The experimental settings are as follows: we apply $SDN_{M2}*$ network, and train the network with 367 training images, and test it with 233 test images. The initial learning rate is changed to 5e-4. Meanwhile, we also adopt data augmentation to reduce overfitting by multi-scale translation. Note that we compare SDN with the previous state-of-the-art methods on two settings, i.e., initializing the network with or without the model pretrained on PASCAL VOC 2012 [22].

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

Following [39], [53], [17], Mean IoU and Global Avg are employed to evaluate our method on this dataset. We compare our method with previous ones in Table VI. Results show that our method on two settings both outperforms other methods. The model, without pretrained on PASCAL VOC 2012(marked by SDN), achieves 69.6% in Mean IoU and 91.7% in Global Avg. Particularly, the classes, including *Car*, *Sign*, *Pedestrian*, *Bicyclist*, have a major boost in performance. When we initialize the network pretrained on PASCAL VOC dataset (marked by SDN+), the performance further improves by 2.2 percent in Mean IoU and 1.0 percent in Global Avg, and here nine out of eleven object categories achieve best performance in Mean IoU. Note that, the CamVid dataset sampling in video frames contains temporal information, and some works [52], [51] mine temporal information to aid segmentation results. Our method still outperforms these works by a relative large margin. Besides, the spatio-temporal information of the dataset is complementary to our method and could bring additional improvements.

Some test images along with ground truth and our predicted semantic maps are shown in Fig. 7. We can find that our network can well sketch multi-scale appearance of objects, including large-scale objects, i.e. building and car, and shape objects, i.e. poles and pedestrians. All the results on this dataset show that our network can capture more contextual information and learn better spatial-relationship.

TABLE VII
EXPERIMENTAL RESULTS ON GATECH TEST SET.

Method	Temporal Info	Global Avg	Mean IoU
3D-V2V-scratch [55]	Yes	66.7	-
3D-V2V-finetune [55]	Yes	76.0	-
FC-DenseNet103 [53]	No	79.4	-
HDCNN-448+TL [51]	Yes	82.1	48.2
DCDN[39]	No	83.5	49.0
SDN	No	84.6	53.5
SDN+	No	**86.3**	**55.9**

D. Results on GATECH dataset

In order to further verify the generalization of our models, we evaluate our network on GATECH [24] dataset, which is much larger than CamVid dataset, but has a lot of noisy annotations. We employ the SDN_{M2*} network with the same training strategy on CamVid. We also compare our models with previous state-of-the-art methods on two settings, i.e., initializing the network with (marked by SDN) or without (marked by SDN+) the model pretrained on PASCAL VOC 2012 [22].

Results are shown in Table VII, we can find that our method on two settings both outperforms other methods. SDN yields 53.5% in mean IoU and 84.6% in Global Avg. and SDN+ improves the result significantly, which gives 1.7% gain in Global Avg and 2.4% in Mean IoU. Specially, our model, without using any temporal, performs better than the models [55], [51] which exploit spatio-temporal relationships between video frames. All these comparisons confirm the proposed SDN a robust and effective model for coarse-annotation dataset. Some test images along with ground truth and our predicted semantic maps are shown in Fig. 8.

E. Results on COCO Stuff dataset

We also conduct an additional experiment on the COCO Stuff dataset to evaluate our network, we employ SDN_{M3} network and train the network with the initial learning rate 2.5e-4. Our approach is compared with some of the state-of-the-art methods in Table.VIII. Result shows that our model achieves 35.9% in Mean IoU and 67.6% in Global Avg, which outperforms the previous methods.

TABLE VIII
EXPERIMENTAL RESULTS ON COCO STUFF VAL SET.

Method	Global Avg	Mean IoU
FCN [9]	52.0	22.7
DeepLab [12]	57.8	26.9
DAG-RNN [56]	63.0	31.2
RefineNet [34]	65.2	33.6
SDN	**67.6**	**35.9**

V. CONCLUSION

In this paper, we have presented a stacked deconvolutional network (SDN), a novel deep network architecture for semantic segmentation. We stack multiple SDN units to make network deeper and realize a coarse-to-fine learning process, meanwhile, intra-unit and inter-unit connections and hierarchical supervision are adopted to promote network training. The ablation experiments show that those designs effectively capture contextual information and recover the spatial resolution for accurate boundary localization, which benefit network performance. Our best model outperforms all previous works on four public benchmarks.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (61872366) and Beijing Municipal Natural Science Foundation (4192059)

REFERENCES

[1] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, 1998.
[2] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Neural Information Processing Systems*, 2012, pp. 1106–1114.
[3] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015.
[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.
[5] R. B. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 580–587.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TIP.2019.2895460, IEEE Transactions on Image Processing

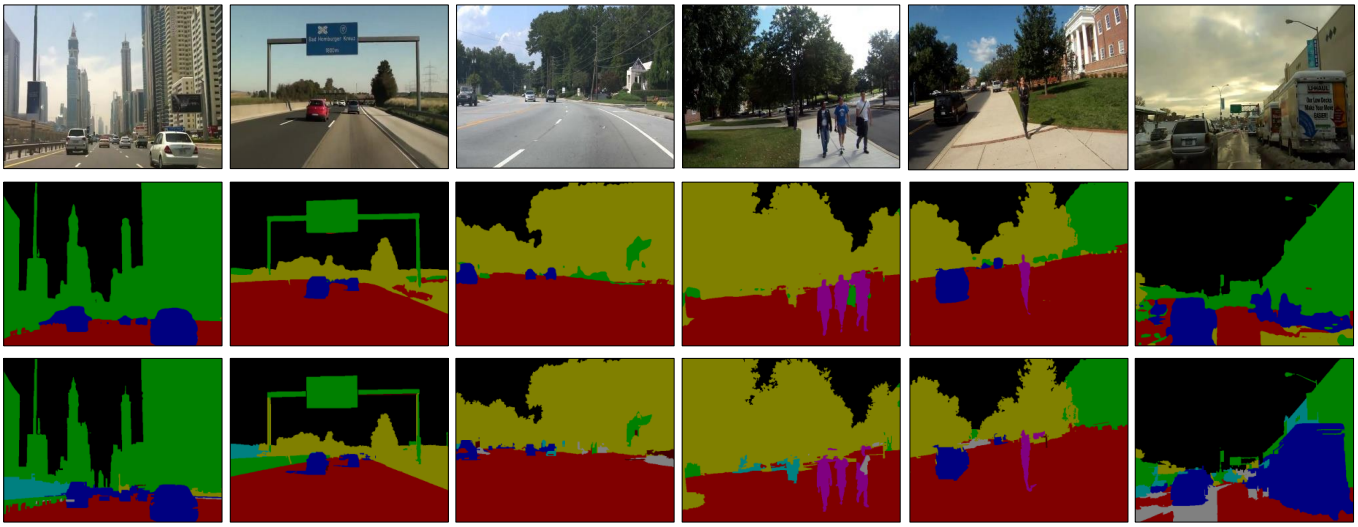

Fig. 8. Results on GATECH dataset. The images in each column from top to down are: (1) input image (2) semantic segmentation result (3) groundtruth.

[6] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," in *Neural Information Processing Systems*, 2015, pp. 91–99.

[7] A. Toshev and C. Szegedy, "Deeppose: Human pose estimation via deep neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1653–1660.

[8] A. Newell, K. Yang, and J. Deng, "Stacked hourglass networks for human pose estimation," in *European Conference on Computer Vision*, 2016, pp. 483–499.

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

[10] L. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected crfs," in *ICLR*, 2015.

[11] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, vol. abs/1706.05587, 2017.

[12] L. 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.

[13] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," vol. abs/1511.07122, 2015.

[14] W. Liu, A. Rabinovich, and A. C. Berg, "Parsenet: Looking wider to see better," vol. abs/1506.04579, 2015.

[15] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," pp. 6230–6239, 2017.

[16] Y. Wang, J. Liu, Y. Li, J. Yan, and H. Lu, "Objectness-aware semantic segmentation," in *ACM International Conference on Multimedia*, 2016, pp. 307–311.

[17] 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, 2017.

[18] H. Noh, S. Hong, and B. Han, "Learning deconvolution network for semantic segmentation," in *International Conference on Computer Vision*, 2015, pp. 1520–1528.

[19] A. Kendall, V. Badrinarayanan, and R. Cipolla, "Bayesian segnet: Model uncertainty in deep convolutional encoder-decoder architectures for scene understanding," *CoRR*, vol. abs/1511.02680, 2015.

[20] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and F. Li, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[21] G. Huang, Z. Liu, L. van der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2261–2269.

[22] M. Everingham, L. J. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman, "The pascal visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.

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

[24] S. H. Raza, M. Grundmann, and I. A. Essa, "Geometric context from videos," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3081–3088.

[25] H. Caesar, J. R. R. Uijlings, and V. Ferrari, "Coco-stuff: Thing and stuff classes in context," *CoRR*, vol. abs/1612.03716, 2016.

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

[27] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," in *International Conference on Computer Vision*, 2017, pp. 2980–2988.

[28] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3150–3158.

[29] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, and G. W. Cottrell, "Understanding convolution for semantic segmentation," *CoRR*, vol. abs/1702.08502, 2017.

[30] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," *CoRR*, vol. abs/1703.06211, 2017.

[31] J. Liu, M. Li, Q. Liu, H. Lu, and S. Ma, "Image annotation via graph learning," *Pattern Recognition*, vol. 42, no. 2, pp. 218–228, 2009.

[32] J. Liu, Y. Jiang, Z. Li, Z. Zhou, and H. Lu, "Partially shared latent factor learning with multiview data," *IEEE Trans. Neural Netw. Learning Syst.*, vol. 26, no. 6, pp. 1233–1246, 2015.

[33] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *MICCAI Medical Image Computing and Computer-Assisted Intervention*, 2015, pp. 234–241.

[34] G. Lin, A. Milan, C. Shen, and I. D. Reid, "Refinenet: Multi-path refinement networks for high-resolution semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 5168–5177.

[35] 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*, 2018.

[36] S. Xie and Z. Tu, "Holistically-nested edge detection," in *ICCV*, 2015, pp. 1395–1403.

[37] P. Bilinski and V. Prisacariu, "Dense decoder shortcut connections for single-pass semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[38] H. Ding, X. Jiang, B. Shuai, A. Q. Liu, and G. Wang, "Context contrasted feature and gated multi-scale aggregation for scene segmentation," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2018.

[39] J. Fu, J. Liu, Y. Wang, and H. Lu, "Densely connected deconvolutional network for semantic segmentation," in *ICIPInternational Conference on Image Processing*, 2017, pp. 3085–3089.

[40] L. Wang, C. Lee, Z. Tu, and S. Lazebnik, "Training deeper convolutional networks with deep supervision," *CoRR*, vol. abs/1505.02496, 2015.

[41] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. B. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," *CoRR*, vol. abs/1408.5093, 2015.

[42] B. Hariharan, P. Arbelaez, L. D. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *International Conference on Computer Vision*, 2011, pp. 991–998.

[43] T. Lin, M. Maire, S. J. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: common objects in context," in *ECCVEuropean Conference on Computer Vision*, 2014, pp. 740–755.

[44] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, and P. H. S. Torr, "Conditional random fields as recurrent neural networks," in *ICCV International Conference on Computer Vision*, 2015, pp. 1529–1537.

[45] R. Vemulapalli, O. Tuzel, M. Liu, and R. Chellappa, "Gaussian conditional random field network for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3224–3233.

[46] Z. Liu, X. Li, P. Luo, C. C. Loy, and X. Tang, "Semantic image segmentation via deep parsing network," in *ICCV International Conference on Computer Vision*, 2015, pp. 1377–1385.

[47] G. Lin, C. Shen, A. van den Hengel, and I. D. Reid, "Efficient piecewise training of deep structured models for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 3194–3203.

[48] Z. Wu, C. Shen, and A. van den Hengel, "Wider or deeper: Revisiting the resnet model for visual recognition," *CoRR*, vol. abs/1611.10080, 2015.

[49] H. Zhang, K. J. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal, "Context encoding for semantic segmentation," vol. abs/1803.08904, 2018.

[50] F. Visin, A. Romero, K. Cho, M. Matteucci, M. Ciccone, K. Kastner, Y. Bengio, and A. C. Courville, "Reseg: A recurrent neural network-based model for semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 426–433.

[51] Y. Wang, J. Liu, Y. Li, J. Fu, M. Xu, and H. Lu, "Hierarchically supervised deconvolutional network for semantic video segmentation," *Pattern Recognition Letters*, vol. 64, pp. 437–445, 2017.

[52] A. Kundu, V. Vineet, and V. Koltun, "Feature space optimization for semantic video segmentation," in *CVPRConference on Computer Vision and Pattern Recognition*, 2016, pp. 3168–3175.

[53] S. Jégou, M. Drozdzal, D. Vázquez, A. Romero, and Y. Bengio, "The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation," *CoRR*, vol. abs/1611.09326, 2016.

[54] M. A. Islam, M. Rochan, N. D. B. Bruce, and Y. Wang, "Gated feedback refinement network for dense image labeling," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 4877–4885.

[55] D. Tran, L. D. Bourdev, R. Fergus, L. Torresani, and M. Paluri, "Deep end2end voxel2voxel prediction," in *IEEEConference on Computer Vision and Pattern Recognition*, 2016, pp. 402–409.

[56] B. Shuai, Z. Zuo, B. Wang, and G. Wang, "Scene segmentation with dag-recurrent neural networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 6, pp. 1480–1493, 2018.

Jun Fu eceived the B.E. degree from Huazhong University of Science and Technology, Wuhan, China, in 2015. He is currently pursuing the Ph.D. degree at National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. His current research interests include deep learning, image or video segmentation and understanding.

Jing Liu received the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 2008. She is a Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. Her current research interests include deep learning, image content analysis and classification, multimedia understanding and retrieval.

Yuhang Wang received the Ph.D. degree from the Institute of Automation, Chinese Academy of Sciences, Beijing, in 2017. His current research interests include deep learning, image or video segmentation and understanding.

Jin Zhou is Ph.D., Professor, supported by National Outstanding Youth Fund. She receive the Ph.D. degree in 2010 and her research field is neural engineering pioneering at the construction of bioactive neural network in vitro and neural signal analysis, acquisition and analysis of intracortical neural signal and EEG, intelligent control of brain-machine interfaces based on multi-type neural information.

Changyong Wang is PhD., professor, supported by National Distinguished Youth Fund, the Pearl-river scholar. He received doctoral degree in 1998. A series of scientific achievements have been made in the field of intelligent control through brain-machine interface, efficient acquisition of brain information, accurate decoding and neural feedback, and the development of robots, artificial limbs, and other intelligent control systems based on different types of brain neural signals.

Hanqing Lu received the Ph.D. from Department of Electronic and Information Science in Huazhong University of Science and Technology. He is a Professor with the National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences. His current research interests include image similarity measure, video analysis, multimedia technology and system.