Saqib MAMOON, Muhammad Arslan MANZOOR, Fa-en ZHANG, Zakir ALI, Jian-feng LU, 2020. SPSSNet: a real-time network for image semantic segmentation. *Frontiers of Information Technology & Electronic Engineering*, 21(12):1770-1782. https://doi.org/10.1631/FITEE.1900697

SPSSNet: a real-time network for image semantic segmentation

Front Inform Technol Electron Eng

Key words: Real-time semantic segmentation; Stage-pooling; Feature reuse

Saqib MAMOON
E-mail: saqibmamoon@njust.edu.cn
ORCID: https://orcid.org/0000-0002-8392-5118

Real-time segmentation & its challenges

- In recent years, the application of deep neural networks in mobile systems has attracted more attention, and research increasingly focused on building light-weight, fast networks with an acceptable accuracy for segmentation in real time.
- A vast number of deep neural network (DNN) architectures based on FCN were proposed, and demonstrated exceptional progress in this field; however, they are often implemented offline due to their low inference speed and high-dimensional feature vectors.
- To keep the model light-weight and fast, small feature maps and low-resolution images are often used, which can significantly reduce the accuracy. Additionally, in encoder-decoder architectures, the image goes through abundant down-sampling and up-sampling operations, thereby losing much of the finer image structure.
- Consequently, the critical information from early and intermediate layers is completely neglected, leading to the reduction in the overall performance of the network.

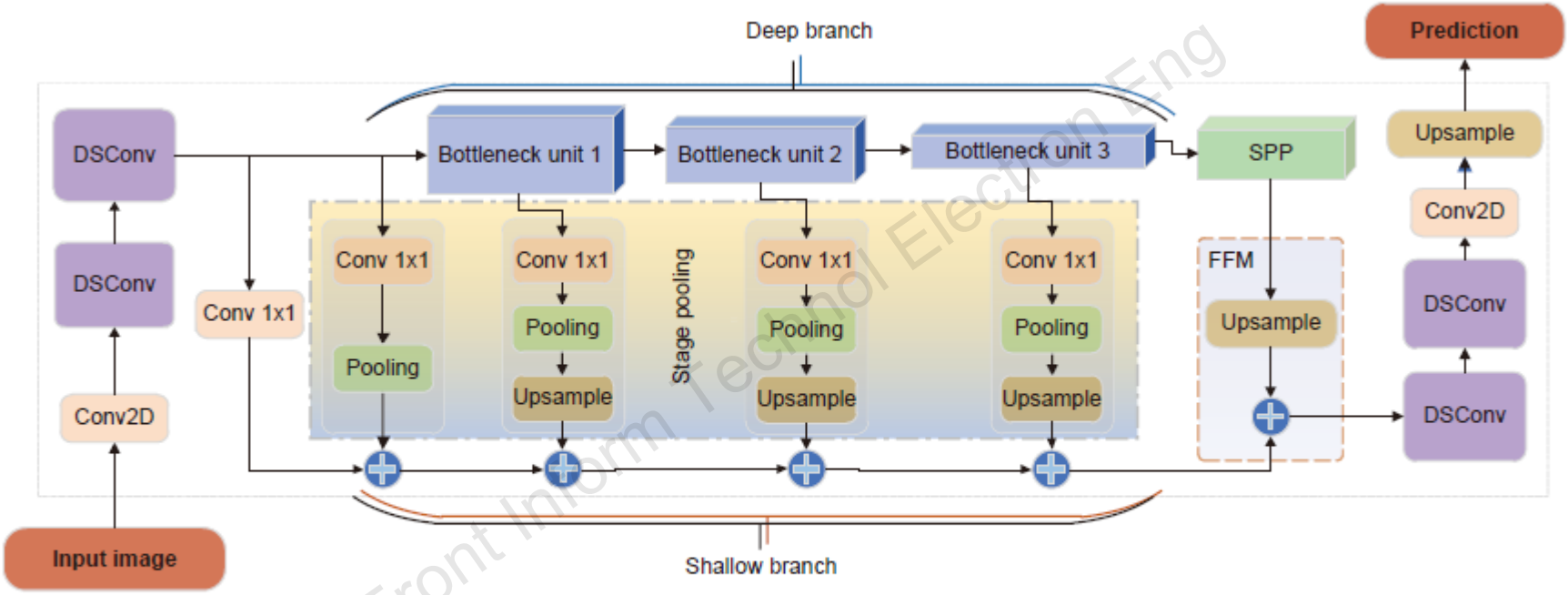
Front Inform Technol Electron Eng

Motivation

- Real-time semantic segmentation models proposed recently are mostly multi-branch architectures, with each branch working at a different resolution.
- The deep branch consists of a series of convolution and max-pooling layers, which significantly reduces the spatial resolution. To refine and recover the spatial details of segmentation results, the shallow branch is employed.
- Note that early layers of DNNs extract low-level features, such as corners, edges/colors, and textures. Moreover, a receptive field of limited size is used in earlier layers, and feature pooling strategies such as spatial pyramid pooling (SPP) and atrous spatial pyramid pooling (ASPP) are often employed at the end of a network to make it more mobile-friendly.
- Consequently, maximum use of early and intermediate features at a high resolution is not considered. Therefore, we introduce a stage-pooling network that can efficiently extract the features from early layers at a high spatial resolution, resulting in sharp edges and corners, which is of utmost importance in semantic segmentation.

Front Inform Technol Electron Eng

SPSSN: stage-pooling semantic segmentation network



Front Inform Technol Electron Eng

Fig. 1 Architecture of a stage-pooling semantic segmentation network (SPSSN): the overall network architecture contains four depth-wise separable convolutions (DSConv), three bottleneck units, four stage-pooling modules, one spatial pyramid pooling (SPP), pointwise convolutions, and one feature fusion module (FFM)

SPSSN architecture

- SPSSN consists of the following major parts:
  - Learning to downsample module
  - Deep branch
  - Stage-pooling modules
  - Shallow branch
  - Feature fusion module

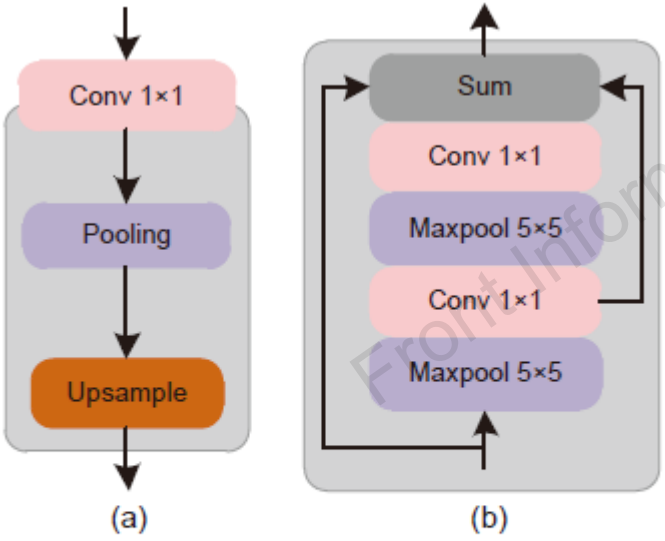


Fig. 2 Stage-pooling module with a pointwise convolution, a pooling module, and an up-sampling layer is shown in (a), and CRP as a pooling module used in (a) is shown in (b)

Table 1 Network architecture of SPSSN

Name	Input size	Output size	n
Conv2D	$2048\times1024\times3$	$1024\times512\times32$	1
DSConv	$1024\times512\times32$	$512\times256\times48$	1
DSConv	$512\times256\times48$	$256\times128\times64_{db,sb}$	1
Stage-pooling 1	$256\times128\times64_{db}$	$256\times128\times128_{sb}$	1
Bottleneck	$256\times128\times64_{db}$	$128\times64\times64_{db}$	3
Stage-pooling 2	$128\times64\times64_{db}$	$256\times128\times128_{sb}$	1
Bottleneck	$128\times64\times64_{db}$	$64\times32\times96_{db}$	3
Stage-pooling 3	$64\times32\times96_{db}$	$256\times128\times128_{sb}$	1
Bottleneck	$64\times32\times96_{db}$	$64\times32\times128_{db}$	3
Stage-pooling 4	$64\times32\times128_{db}$	$256\times128\times128_{sb}$	1
SPP	$64\times32\times128_{db}$	$64\times32\times128_{db}$	1
FFM	$(64\times32\times128)^{4\times}_{db}$ $+256\times128\times128_{sb}$	$256\times128\times128$	1
DSConv	$256\times128\times128$	$256\times128\times128$	2
Conv2D	$256\times128\times128$	$256\times128\times c$	1

SPSSN consists of standard convolutions (Conv2D), depth-wise separable convolutions (DSConv), residual bottleneck convolutions (Bottleneck), stage-pooling modules, a spatial pyramid pooling (SPP) module, and a feature fusion module (FFM). n represents the number of blocks. c is the size of classifier. "db" and "sb" represent the deep branch and shallow branch, respectively. $^{4\times}$ means up-sampling four times

Front Inform Technol Electron Eng

Datasets

❑ Cityscapes dataset

- ❑ It contains 5000 finely annotated images and 20 000 coarsely annotated images, with a resolution of 2048×1024 pixels.
- ❑ Following the standard settings of cityscapes, the finely annotated images are split into 2979, 500, and1525 images for training, validation, and testing, respectively.

❑ CamVid dataset

- ❑ It consists of 701 manually annotated images, split into 367, 101, and 233 images for training, validation, and testing, respectively. The images have a maximum resolution of 960×720 pixels.
- ❑ CamVid consists of 32 classes originally; however, due to the rare occurrence of some classes, most literature focused on only 11.

Front Inform Technol Electron Eng

Evaluation criteria

❑ Evaluation metrics

We adopted primarily three evaluation metrics for performance measurement:

1. The mean intersection-over-union (mIoU) of classes and categories as the segmentation accuracy metric
2. The number of parameters as the computation complexity metric
3. The number of frames per second (FPS) as the speed metric.

Front Inform Technol Electron Eng

Experimental results

Table 2 Results of class and category in terms of mean intersection-over-union and mean instance intersection-over-union, and the number of parameters on the Cityscapes test set

Network	mIoU (%)		miIoU (%)		Number of parameters ($\times 10^6$)
	Class	Category	Class	Category	
BiSeNet2 (Yu CQ et al., 2018)	74.7	–	–	–	49
SegNet (Badrinarayanan et al., 2017)	57.0	79.1	32.0	61.9	29.5
ENet (Paszke et al., 2016)	58.3	80.4	34.4	–	0.36
ESPNet (Mehta et al., 2018)	60.3	82.2	31.8	63.1	0.4
ThunderNet (Xiang et al., 2019)	64.0	84.1	40.4	69.3	4.7
ContextNet (Poudel et al., 2018)	66.1	82.7	36.8	64.3	0.85
ERFNet (Romera et al., 2018)	68.0	86.5	40.4	70.4	2.1
Fast-SCNN (Poudel et al., 2019)	68.0	84.7	37.9	63.5	1.11
BiSeNet1 (Yu CQ et al., 2018)	68.4	–	–	–	5.8
ICNet (Zhao et al., 2018)	69.5	86.4	–	–	7.8
SPSSN (Ours)	69.4	86.0	41.8	70.2	1.42

mIoU: mean intersection-over-union; miIoU: mean instance intersection-over-union

Front Inform Technol Electron Eng

Experimental results (Cont'd)

Table 5 Average class accuracy and global accuracy, mean intersection-over-union, the number of parameters, and the number of frames per second (FPS) on the CamVid test set

Network	mIoU (%)	Class avg. (%)	Global avg. (%)	Number of parameters ($\times 10^6$)	FPS
ENet (Paszke et al., 2016)	51.3	68.3	–	0.36	61.2
SegNet (Badrinarayanan et al., 2017)	55.6	65.2	88.5	29.5	4.6
FCN-8s (Long et al., 2014)	57.0	–	88.0	134.5	–
FC-DenseNet56 (Jégou et al., 2017)	58.9	–	88.9	1.5	27
DeepLab-LFOV (Chen et al., 2018)	61.6	–	–	37.3	4.9
Dilation8 (Yu F and Koltun, 2016)	65.3	–	79.0	140.8	4.4
ESPNet (Mehta et al., 2018)	55.6	68.3	–	–	205
ERFNet (Romera et al., 2018)	53.1	65.8	86.3	–	–
ERFNet* (Romera et al., 2018)	62.7	72.2	89.4	–	–
ICNet (Zhao et al., 2018)	67.1	–	–	–	27.8
SPSSN (Ours)	64.3	73.55	91.1	1.4	105

mIoU: mean intersection-over-union; Class avg.: average class accuracy; Global avg.: average global accuracy; FPS: number of frames per second. * Pre-trained on ImageNet

Front Inform Technol Electron Eng

Experimental results (Cont'd)

Table 4 Comparison of individual class results on the Cityscapes test set with baseline networks

Network	IoU (%)									
	Road	Sidewalk	Building	Wall	Fence	Pole	Traffic light	Traffic sign	Vegetation	Terrain
SegNet	96.4	73.2	84.0	28.4	29.0	35.7	39.8	45.1	87.0	63.8
ENet	96.3	74.2	75.0	32.2	33.2	43.4	34.1	44.0	88.6	61.4
ESPNet	97.0	77.5	76.2	35.0	36.1	45.0	35.6	46.3	90.8	63.2
ERFNet	97.2	80.0	89.5	41.6	45.3	56.4	**60.5**	64.6	91.4	**68.7**
ThunderNet	97.2	77.3	88.3	41.1	38.3	48.5	55.6	60.8	90.66	67.7
ICNet	97.1	79.2	89.7	43.2	48.9	**61.5**	60.4	63.4	**91.5**	68.3
ContextNet	97.4	79.6	89.5	**44.1**	**49.8**	45.5	50.6	64.6	90.2	59.4
SPSSN	**97.7**	**80.8**	**89.8**	43.9	46.5	53.1	58.8	**64.7**	**91.5**	**68.7**

Network	IoU (%)								
	Sky	Person	Rider	Car	Truck	Bus	Train	Motorcycle	Bicycle
SegNet	91.8	62.8	42.8	89.3	38.1	43.1	44.1	35.8	51.9
ENet	90.6	65.5	38.4	90.6	36.9	50.5	48.1	8.8	55.4
ESPNet	92.6	67.0	40.9	92.3	38.1	52.5	50.1	41.8	57.2
ERFNet	**94.2**	76.1	56.4	92.4	45.7	60.6	27.0	48.7	61.8
ThunderNet	92.9	71.3	46.6	91.6	39.31	49.9	49.8	45.5	62.3
ICNet	93.5	74.6	56.1	92.6	51.3	**72.7**	51.3	**53.6**	**70.5**
ContextNet	93.4	70.9	43.1	91.8	**65.2**	71.9	**64.5**	41.9	66.1
SPSSN	**94.2**	**76.2**	**59.0**	**92.7**	53.5	71.0	59.2	52.9	63.8

IoU: intersection-over-union. The best results are in bold

Front Inform Technol Electron Eng

Experimental results (Cont'd)

Table 3 Comparison of single feed forward time and runtime between SPSSN and several baseline networks

Network	Time (ms)	FPS
SegNet (Badrinarayanan et al., 2017)	67	15
ThunderNet (Xiang et al., 2019)	10.1	96
ENet (Paszke et al., 2016)	13	77
ESPNet (Mehta et al., 2018)	18	54
ERFNet (Romera et al., 2018)	21	41.7
ICNet (Zhao et al., 2018)	33	30
Fast-SCNN (Poudel et al., 2019)	9.4	106.2
ContextNet (Poudel et al., 2018)	24.4	41.9
SPSSN (Ours)	16	59

FPS: number of frames per second

Table 6 Evaluation results of the stage-pooling and stream-pooling branches on the Cityscapes validation dataset as an ablation study

Module	mIoU (%)		Number of parameters ($\times 10^6$)
	Class	Category	
Fast-SCNN	68.62	–	1.11
Stream-pooling	69.70	85.6	1.47
Stage-pooling	70.40	86.4	1.42

Table 7 Inference speed and FPS of SPSSN at different image resolutions on NVIDIA TITAN Xp (Pascal)

Resolution (pixels)	Time (ms)	FPS
2048×1024	16	59
1024×1024	9	111.44
1024×512	7	135.30

FPS: number of frames per second

Front Inform Technol Electron Eng

Visual results

Fig. 3 Visual results on the Cityscapes validation set (left to right: original input image, ground truth, SPSSN results)

Visual results (Cont'd)

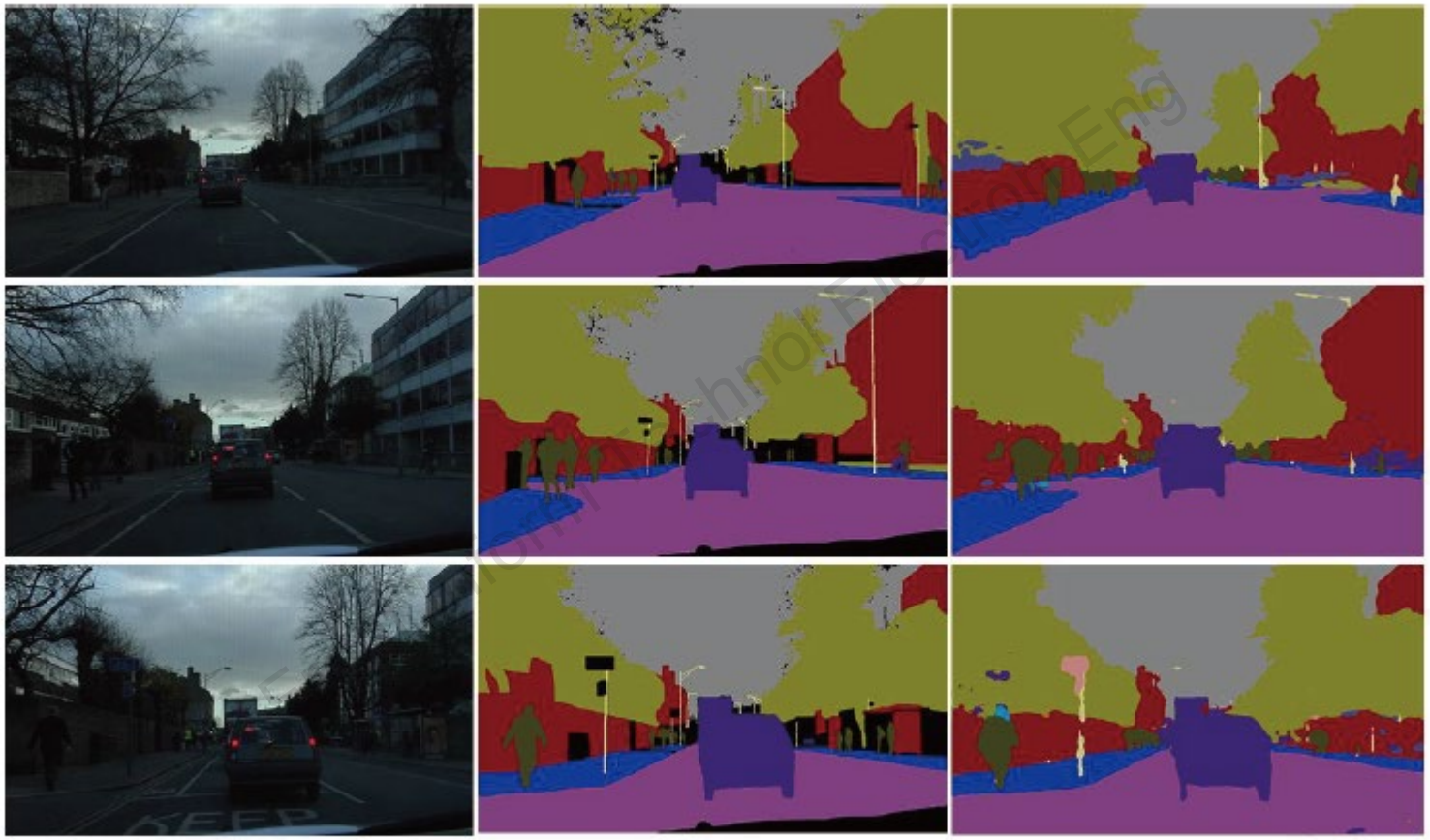


Fig. 4 Visual results on the CamVid test set (left to right: original input image, ground truth, SPSSN results)

Conclusions

In this paper, we have proposed a novel stage-pooling architecture for real-time image semantic segmentation. The experimental results demonstrated that stage-pooling increases the networks ability to use low- and intermediate-level features at high resolution. Pooling at different spatial dimensions and stages benefits the model to better segment the multiple objects in real time for diversely changing environments. The stage-pooling technique can increase the efficiency of a network and can also be used for other tasks, such as image classification and object detection, which will be our future work.

Front Inform Technol Electron Eng