Ardiyanto and Adji *IPSJ Transactions on Computer Vision and Applications* (2017) 9:6
DOI 10.1186/s41074-017-0020-9

IPSJ Transactions on Computer Vision and Applications

EXPRESS PAPER Open Access

Deep residual coalesced convolutional network for efficient semantic road segmentation

Igi Ardiyanto* and Teguh Bharata Adji

Abstract

This paper proposes a deep learning-based efficient and compact solution for road scene segmentation problem, named deep residual coalesced convolutional network (RCC-Net). Initially, the RCC-Net performs dimensionality reduction to compress and extract relevant features, from which it is subsequently delivered to the encoder. The encoder adopts the residual network style for efficient model size. In the core of each residual network, three different convolutional layers are simultaneously coalesced for obtaining broader information. The decoder is then altered to upsample the encoder for pixel-wise mapping from the input images to the segmented output. Experimental results reveal the efficacy of the proposed network over the state-of-the-art methods and its capability to be deployed in an average system.

Keywords: Residual coalesced network, Deep learning, Road segmentation

1 Introduction

Unlike the traditional object detection and classification which globally works on an image or a patch, the scene segmentation is a pixel-wise classification which requires more accurate boundary localization of each object and area inside the images. For instance in case of the road scene segmentation, one needs to precisely separate the sidewalk for the pedestrian from the road body.

The semantic road scene segmentation, which is the part of the general image segmentation problems, attracts a lot of researchers for providing the best solution. Early works mostly depend on the pixel-wise hand-crafted features (e.g., [1]) followed by conditional random field (e.g., [2, 3]), the usage of dense depth map [4], or exploitation of the spatio-temporal parsing [5] for achieving the best acccuracy.

Since the rise of deep learning for object classification [6], several attempts were done for designing a deep network architecture for the image segmentation problem. Most of them follow the *encoder-decoder* architecture style (e.g., [7–9]). Another approach takes advantage of the image patch and spatial prior [10] for attaining better scene segmentation. Except [9] which tries to build a small model size network, all of the above works are suffered from either very large network size or slow inference time which make them inconvenient for practical applications.

Here, we aim to establish a compact and effective network for segmenting the road scene. Our approach is inpired by ResNet [11] which utilizes residual blocks, allowing it to be stacked into a very deep architecture without huge degradation problem. In the heart of our proposed architecture, three different types of convolutional layers are simultaneously coalesced in a residual fashion and stacked it into an encoder-type network for altering the receptive field. Hence, more variational functions are enabled to obtain richer information from the images. Subsequently, a decoder with a lesser architecture followed by a fully connected convolutional (*Full Conv.*) layer is appended to upsample the encoder and fine-tune the output.

Our contributions are twofold. First, we introduce a coalesced style of the convolutional layers with the residual-flavored network to build an efficient model for the semantic road segmentation. Subsequently, we exhibit an asymmetric encoder-decoder network for reducing the model size even more, unlike the conventional symmetric approach used by the previous methods, e.g., SegNet [8].

*Correspondence: igi@ugm.ac.id
Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada, Jl. Grafika No.2, Yogyakarta, Indonesia

The rest of this paper is organized as follows. Section 2 explains the overall architecture of the proposed RCC-Net. Evaluations against several *state-of-the-art* methods are described in Section 3. We then conclude the paper and give some future directions of the research in Section 4.

2 Proposed network architecture

Our proposed RCC-Net is established in a deep encoder-decoder manner. The target is to create a pixel-wise classification which maps each pixel of the input images into corresponding semantic class of the road objects. Figure 1 expresses the full architecture of the RCC-Net.

2.1 Initial stage

The idea of constructing small feature maps in the early stage of the network was heavily inspired by [9] and [12]. For this purpose, a max pooling, an average pooling, and 3 × 3 convolution layers with 13 filters are concatenated, creating a total 19 dimensional feature maps. Figure 2 represents the initial stage of the RCC-Net. Using these settings, the first stage of the RCC-Net is expected to reduce the dimensionality of the input images while extracts the relevant features for the next stages.

2.2 Residual coalesced convolutional blocks

As the core of our network, we introduce the residual coalesced convolutional (RCC) block which is intemperately instigated by Inception [13] and ResNet [11] architectures. The RCC module is composed by *projection-receptive-projection* sequences with skip connection. The projection parts are realized by 1 × 1 convolution, while the receptive section consists of a coalesced three different convolutional layers.

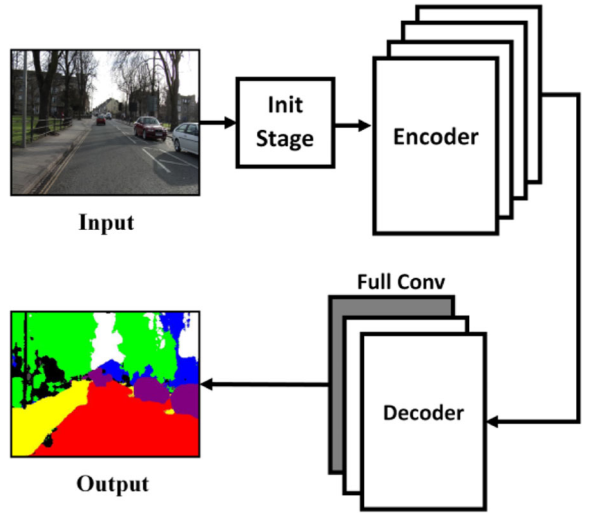

Fig. 1 Architecture of RCC-Net

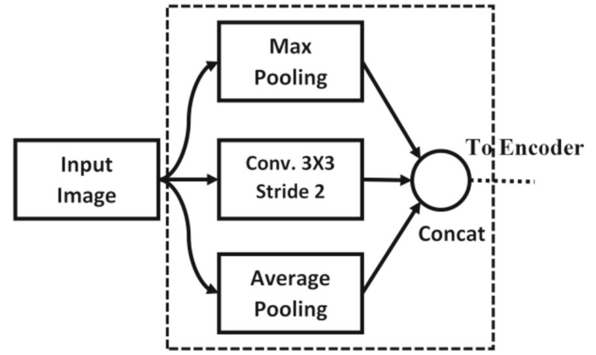

Fig. 2 Initial stage of RCC-Net

The 1 × 1 convolution is meant to aggregate the activation of each feature in the previous layer. It is eminent for infering the networks with different input size. An ordinary, an asymmetric [12], and a dilated [14] convolution layers are subsequently appended in a parallel fashion. This coalesced style is motivated by an assumption that each type of convolution layer contributes different receptive field. By coalescing them, it is expected to have a wider function to be learned, thus increasing the amount of feature information.

Let $X \in \mathbb{R}^n$ be the n-dimensional input of the coalesced convolution and $w_j^i \in \mathbb{R}^{i\times j}$ is the $i \times j$ convolution kernel. The corresponding feature output for the coalesced convolution can be denoted by

$$M = \left[\underbrace{w_j^i * X}_{\text{ordinary}} \bigcup \underbrace{w_j^1 * X}_{\text{asymmetric}} \bigcup \underbrace{w_j^i *_d X}_{\text{dilated}} \right] \tag{1}$$

where the last term is the dilated convolution with the dilation factor d. Figure 3 shows the RCC module formation.

Actually, it is interesting to investigate the proper way to combine the convolutional layers. In the experimental section, we will show how the change on its combination, by summing and concatenating them, will affect the entire network results.

The entire encoder contains three stages, where each stage is made from five RCC modules. The ordinary convolution uses 3 × 3 kernel. Dilation factor of the dilated convolutions is arranged from 2 to 32, while the asymmetric kernels are set to 5 and 7. In between the convolutional operation inside the RCC modules, a parametric rectified linear unit (PReLU) activation layer and a batch n are added. We then place a drop out layer at the end

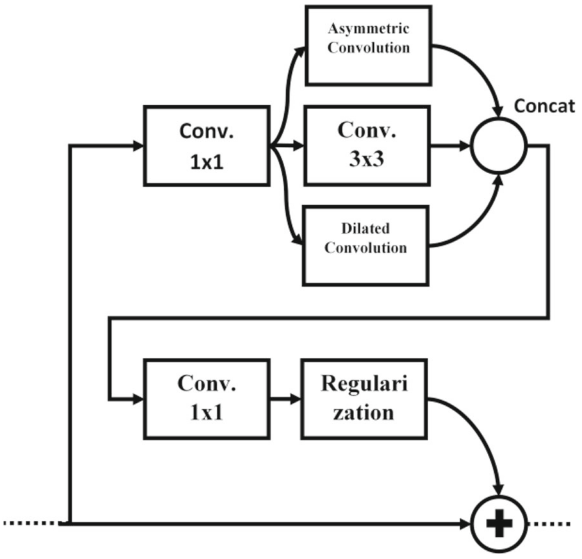

Fig. 3 Residual coalesced convolutional modules

of RCC modules for regularization. A skip connection imitating the ResNet [11] is coupled for one RCC module. A max-pooling layer is subsequently appended between each stage for downsampling the input.

2.3 Decoder

The decoder is constructed by stacking the same RCC modules as the encoder, except the coalesced convolutional part is now replaced by a deconvolutional layer and the number of stages is decreased. This setting is motivated by [9], where the role of the pixel recognition should be done mostly by the encoder. The task of the decoder is merely to upsample the output of the encoder and adjust the details. A fully connected convolutional (*Full Conv.*) layer is thus appended behind the decoder for performing pixel-wise mapping.

As summary of the proposed network, Table 1 exhibits the configuration of the RCC-Net, with 3-channel input images and 11 classes of the road scenes.

3 Results and discussions

In this section, the efficacy of our proposed architecture is demonstrated against several state-of-the-art methods on the road scene segmentation problems. All implementations of the proposed algorithm were done on a Linux PC (Ubuntu 16.04, Core i7, 32 GB RAM), with a GTX 1080 GPU and Torch7. Training was performed using Adam optimization [15] for 200 epoch with learning rate 10*e*-3, momentum 0.9, and batch size 8.

3.1 CamVid dataset benchmark

The performance of the proposed RCC-Net architecture is benchmarked on CamVid road scene dataset [16], which consists of 367 training and 233 testing images with the resolution of 480 × 360. The CamVid dataset has 11 classes depicting different objects which frequently appear on the street, such as road, cars, pedestrian, and building. Table 2 shows the comparison of several state-of-the-art methods on the CamVid road scene dataset.

Table 1 Configuration of RCC-Net

Stages	Convolution			# RCC
	Ordinary	Asymmetric	Dilated	
Input		3 × 480 × 360		n/a
Initial		19 × 240 × 180		n/a
Encoder 1	3 × 3	5 × 1; 1 × 5	2;4;8;16;32	5
Encoder 2	3 × 3	5 × 1; 1 × 5	2;4;8;16;32	5
Encoder 3	3 × 3	7 × 1; 1 × 7	2;4;8;16;32	5
Decoder 1		3 × 3 deconvolution		3
Decoder 2		3 × 3 deconvolution		2
Full Conv.		11 × 480 × 360		n/a

From Table 2, the proposed RCC-Net (concatenated version) exceeds the existing state-of-the-art methods in four different class categories and the overall class average accuracy. Three-out-four winning categories constitute the small area and objects with lesser training data. It means our proposed method is capable for capturing objects which are difficult to segment. The best class average accuracy and a comparable intersection-over-union (IoU) imply the RCC-Net has a high consistency for achieving good results in each category.

One notable result is that the RCC-Net has the best capability for recognizing the sidewalk. It is very important for an autonomous car to differentiate the road and the sidewalk, so that the safety of the pedestrians are guaranteed. Figure 4 depicts some examples of the RCC-Net prediction output on the test set of the CamVid dataset.

As we have noted in the previous section, it is intriguing to examine different ways of coalescing the convolutional layers. Both summing and concatenated convolutional layers of the RCC-Net surpass the other methods. Nevertheless, the concatenated version of RCC-Net has advantages over the summing one. One interesting result is the pedestrian segmentation of the summing version of the RCC-Net achieves the highest accuracy (70.6%). This fact may lead to a promising application in the future research, e.g., to determine the salient regions for the pedestrian detection.

3.2 Test on wild scene

To conceive the RCC-Net capabilities, we subsequently feed the network with some difficult road scenes taken from the Internet without re-train the network

Table 2 Comparison on the CamVid dataset [16] using 11 road scene categories (in percent)

Method	Sky	Building	Road	Sidewalk	Car	Pedestrian	Bicyclist	Tree	Fence	Column-pole	Sign-symbol	Class avg.	Class IoU
Local label descriptor [1]	88.8	80.7	98	12.4	16.4	1.09	0.07	61.5	0.05	4.13	n/a	36.3	n/a
Boosting+pairwise CRF [2]	94.7	70.7	94.1	79.3	74.4	45.7	23.1	70.8	37.2	13	55.9	59.9	n/a
Boosting+detection+CRF [3]	96.2	81.5	93.9	81.5	78.7	43	33.9	76.6	47.6	14.3	40.2	62.5	n/a
Dense depth map [4]	95.4	85.3	**98.5**	38.1	69.2	23.8	28.7	57.3	44.3	22	46.5	55.4	n/a
Super parsing [5]	**96.9**	87	95.9	70	62.7	14.7	19.4	67.1	17.9	1.7	30.1	51.2	n/a
SegNet-basic [8]	91.2	75	93.3	74.1	**82.7**	55	16	84.6	47.5	44.8	36.9	62	47.7
SegNet [8]	92.4	**88.8**	97.2	84.4	82.1	57.1	30.7	**87.3**	49.3	27.5	20.5	65.2	**55.6**
ENet [9]	95.1	74.7	95.1	86.7	82.4	67.2	34.1	77.8	**51.7**	35.4	51	68.3	51.3
RCC-Net (sum)	95.2	70.1	94.1	90.1	82.6	**70.6**	45.7	81.2	51	52.3	35.4	69.8	52.6
RCC-Net (concatenated)	94.3	71.8	92.6	**92.7**	79.3	57.7	**65.6**	80.5	35.7	**57.4**	**59.4**	**71.5**	53.3

The bold values show the highest accuracy for each category

model obtained from the previous CamVid benchmark. From Fig. 5, the RCC-Net produces qualitatively good segmentations, even for the scenes which are heavily cluttered. It also means the proposed network is able to transfer the model information to the new environment.

3.3 Computation time and model size

On GTX 1080, the RCC-Net took 25.5 ms for the forward inference of 480 × 360 images, including fetching and displaying the image. It is also able to run one inference on a car-deployable mini PC Zotac EN-761 in 67.5 ms with the network size of 4.9 MB, which draws out the power consumption around 62.4 watt. It means the proposed network is fast and small enough to enable the Advanced Driver Assistance System (ADAS). We plan to run the network on a GPU-based embedded system, such as NVIDIA Jetson TK1 for further investigation[1].

4 Conclusions

An efficient and compact solution for solving the semantic road segmentation problem has been presented. By coalescing different types of convolutional layers and stacking them in a deep residual network style, we achieve the high-quality results on the semantic road segmentation with relatively small model size, surpassing the existing state-of-the-art methods. In the future, we would like to examine the performance of our RCC-Net on the boarder problems, such as medical images and other challenging image segmentation dataset, for understanding its capabilities to solve more general segmentation applications.

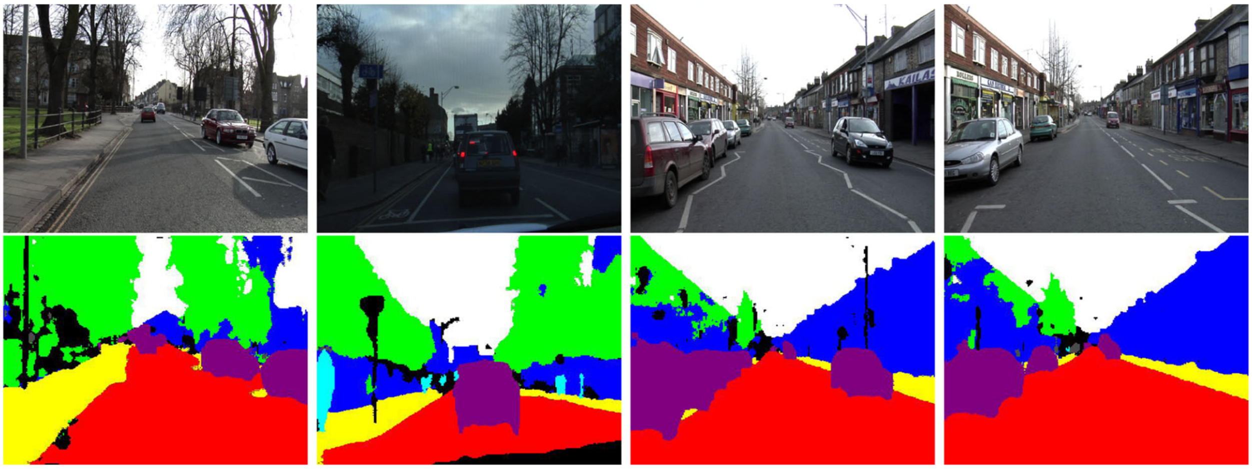

Fig. 4 RCC-Net results on CamVid [16] test set. (color code: *red* road, *yellow* sidewalk, *purple* car, *cyan* pedestrian, *blue* building, *green* tree, *white* sky, and *black* void)

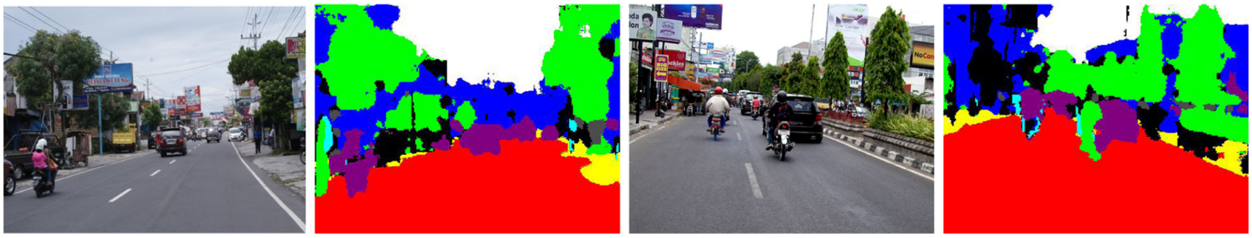

Fig. 5 RCC-Net results on wild scenes

Endnote

[1] The progress of RCC-Net performance on the embedded system can be seen at http://te.ugm.ac.id/~igi/?page_id=826

Acknowledgements

This work was supported by AUN-SEED Net Collaborative Research for Alumni (CRA) Project.

Authors' contributions

IA performed the primary development and analysis for this work and the initial drafting of the manuscript. TBA played an essential role in development of this work and editing the paper. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Received: 20 February 2017 Accepted: 7 March 2017
Published online: 15 March 2017

References

1. Yang Y, Li Z, Zhang L, Murphy C, Hoeve JV, Jiang H (2012) Local label descriptor for example based semantic image labeling. In: Proc. of European Converence on Computer Vision (ECCV). pp 361–375
2. Sturgess P, Alahari K, Ladicky L, H.S.Torr P (2009) Combining appearance and structure from motion features for road scene understanding. In: Proc. of British Machine Vision Conferenve (BMVC)
3. Ladicky L, Sturgess P, Alahari K, Russell C, Torr PHS (2010) What, where and how many? Combining object detectors and CRFs. In: Proc. of European Converence on Computer Vision (ECCV). pp 424–437
4. Zhang C, Wang L, Yang R (2010) Semantic segmentation of urban scenes using dense depth maps. In: Proc. of European Converence on Computer Vision (ECCV). pp 708–721
5. Tighe J, Lazebnik S (2013) Superparsing. Int J Comput Vision (IJCV) 101(2):329–349
6. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: Proc. of NIPS. pp 1097–1105
7. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: Proc. of IEEE Computer Vision and Pattern Recognition (CVPR). pp 3431–3440
8. Badrinarayanan V, Kendall A, Cipolla R (2015) Segnet: a deep convolutional encoder-decoder architecture for image segmentation. arXiv: 1511.00561
9. Paszke A, Chaurasia A, Kim S, Culurciello E (2016) Enet: a deep neural network architecture for real-time semantic segmentation. arXiv: 1606.02147v1
10. Brust CA, Sickert S, Simon M, Rodner E, Denzler J (2015) Convolutional patch networks with spatial prior for road detection and urban scene understanding. In: Proc. of VISAPP
11. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. arXiv: 1512.03385
12. Szegedy C, Vanhoucke V, Ioffe S, Shlens J, Wojna Z (2015) Rethinking the inception architecture for computer vision. arXiv: 1512.00567
13. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: Proc. of IEEE Computer Vision and Pattern Recognition (CVPR). pp 1–9
14. Yu F, Koltun V (2015) Multi-scale context aggregation by dilated convolutions. arXiv: 1511.07122
15. Kingma D, Ba J (2014) Adam: a method for stochastic optimization. arXiv: 1412.6980
16. Brostow GJ, Shotton J, Fauqueur J, Cipolla R (2008) Segmentation and recognition using structure from motion point clouds. In: Proc. of European Converence on Computer Vision (ECCV). pp 44–57