applied sciences

MDPI

Article

Semantic Segmentation Based on Depth Background Blur

Hao Li [1], Changjiang Liu [1,*] and Anup Basu [2]

[1] School of Mathematics and Statistics, Sichuan University of Science and Engineering, Zigong 643000, China; stulihao@163.com
[2] Department of Computing Science, University of Alberta, Edmonton, AB T6G 2H1, Canada; basu@ualberta.ca
* Correspondence: liuchangjiang@189.cn; Tel.: +86-189-9002-4310

Abstract: Deep convolutional neural networks (CNNs) are effective in image classification, and are widely used in image segmentation tasks. Several neural netowrks have achieved high accuracy in segementation on existing semantic datasets, for instance PASCAL VOC, CamVid, and Cityscapes. However, there are nearly no studies on semantic segmentation from the perspective of a dataset itself. In this paper, we analyzed the characteristics of datasets, and proposed a novel experimental strategy based on bokeh to weaken the impact of futile background information. This crucial bokeh module processed each image in the inference phase by selecting an opportune fuzzy factor σ, so that the attention of our network can focus on the categories of interest. Some networks based on fully convolutional networks (FCNs) were utilized to verify the effectiveness of our method. Extensive experiments demonstrate that our approach can generally improve the segmentation results on existing datasets, such as PASCAL VOC 2012 and CamVid.

Keywords: bokeh; fully convolutional networks; semantic segmentation

Citation: Li, H.; Liu, C.; Basu, A. Semantic Segmentation Based on Depth Background Blur. *Appl. Sci.* **2022**, *12*, 1051. https://doi.org/10.3390/app12031051

Academic Editors: Nunzio Cennamo, Yangquan Chen, Simone Morais, Subhas Mukhopadhyay, Junseop Lee and M. Jamal Deen

Received: 4 December 2021
Accepted: 17 January 2022
Published: 20 January 2022

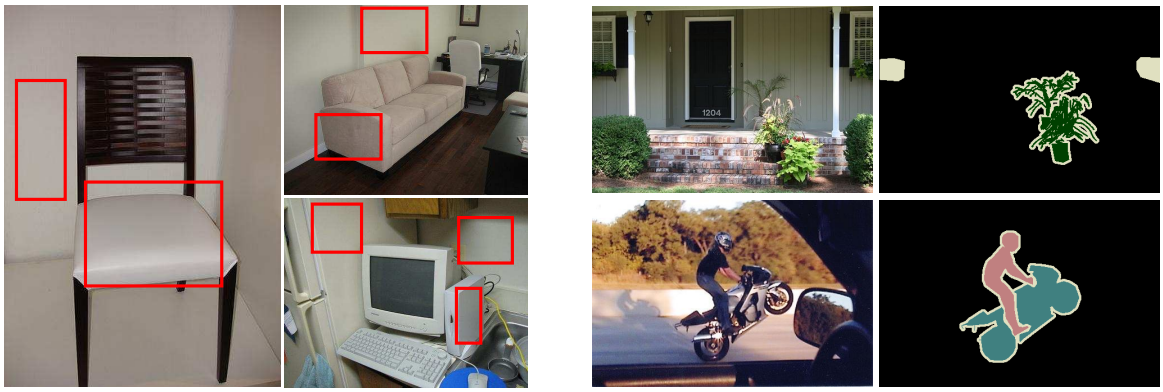
1. Introduction

In recent years, an increasing number of researchers have applied convolutional neural networks (CNNs) to resolve pixelwise and end-to-end image segmentation tasks, e.g., semantic segmentation [1–4]. Semantic segmentation can be understood as the need to segment each object in an image and annotate it with different colors. For instance, people, displays, and aircraft in the PASCAL VOC 2012 dataset were marked in pink, blue, and red respectively. As a significant role in computer vision, semantic segmentation has been widely implemented for fields like autonomous driving [5], robot perception [6], augmented reality [7], and video surveillance [8].

Since the advent of fully convolutional networks (FCN [9]), they have greatly simplified the conventional approach to address the conundrum of semantic segmentation. Various end-to-end network architectures derived from FCN have been proposed over the years. Based on existing datasets, the segmentation accuracies are relatively high or even the maximum possible. The series of DeepLab [10–13] proposes atrous convolution with dilation to improve the problem of a scarce receptive field caused by an insufficient amount of down-sampling. The proposed atrous spatial pyramid pooling, to carry out multi-scale feature fusion, significantly advances the accuracy of network segmentation. Yu et al. [14] proposed the bilateral segmentation network, which better preserves the spatial information of the original image while ensuring a sufficient receptive field. From semantic segmentation to real-time semantic segmentation [14–16], considering redundant to lean network architectures, existing scholars accomplish better segmentation by designing and improving the structure of the network itself and adopting massive data augmentation methods. However, they ignored the impact of the characteristics of the dataset itself on the segmentation results.

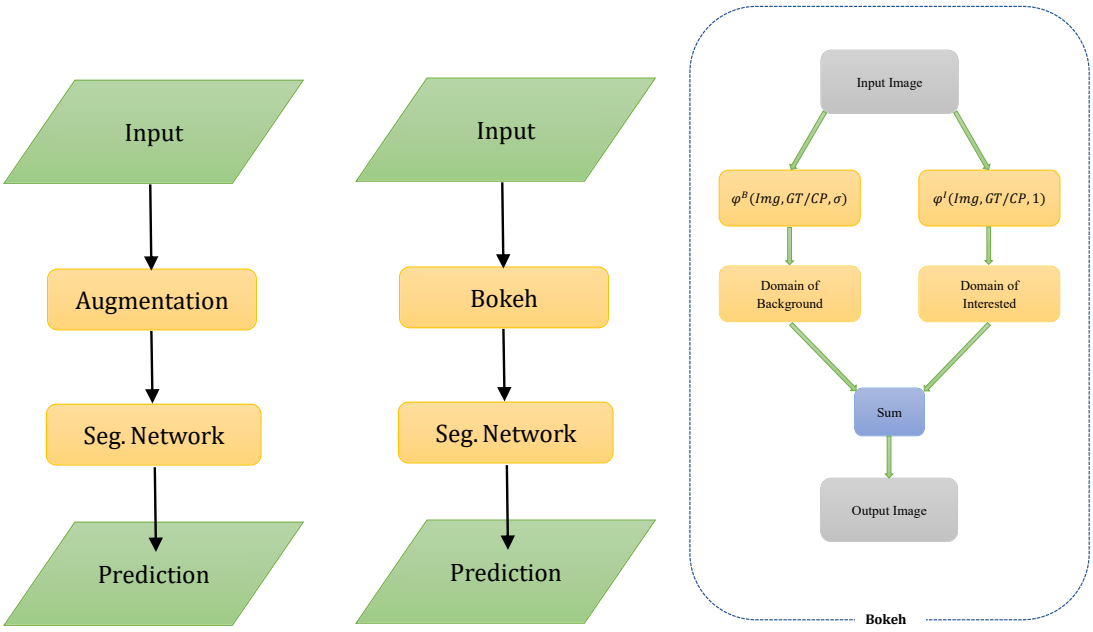
Semantic segmentation, as a pixelwise classification task, requires the classification of every pixel. Nevertheless, not every pixel is of interest to us. A substantial amount

of background information during the training phase not only increases the difficulty of learning, but also leads to misclassifications, see Figure 1. In view of the aforementioned issues, and motivated by the excellence of the self-attention mechanism in the segmentation task [17,18], we apply the attention mechanism to the dataset itself. Through in-depth analysis of the dataset, we propose the background blur module bokeh. The overall structure is shown in Figure 2, and a feasible strategy for selecting the fuzzy factor σ is proposed in Section 3.



(**a**) (**b**)

Figure 1. An example in the PASCAL VOC 2012 dataset. (**a**) Chairs, sofas, and optical modems in the field of interest have the same color and shape as the walls and wardrobes in the background. (**b**) Areas of interest only occupy an unobtrusive part of the whole picture.



(**a**) (**b**) (**c**)

Figure 2. (**a**) Existingmainstream semantic segmentation network architecture. (**b**) The segmentation network architecture of this paper. (**c**) bokeh module, see Section 3 for algorithm details.

Humans, as the most sophisticated creatures on earth, have a natural advantage in patten recognition. Relying on foveated and active vision, our visual center always focusses on the area of interest to us, rather than on the background.

The proposed bokeh module mainly performs a certain degree of background blurring according to the distribution of various categories of the dataset itself, without any prior knowledge of the domain of interest. It combines the blurred background with the domain of interest as the subsequent segmentation network input. Specifically, during the training stage, background and foreground are divided accurately by real semantic labels provided by the training set. In the validation phase, the original segmentation network is able to separate the background and foreground, based on the coarse segmentation. The reconstructed segmentation network with the bokeh module performs the final semantic segmentation. The visualization of our bokeh module is shown in Figure 3.

(**a**) Image (**b**) Image-I

(**c**) Image-B (**d**) Image-bokeh

Figure 3. Image visualization after adding the bokeh module: (**a**) original image, (**b**) the domain of interest φ^I, (**c**) the background blurred domain φ^B, and (**d**) Image-bokeh, namely a combination of the latter two.

We demonstrate the effectiveness of our approach on two datasets, PASCAL VOC 2012 [19] and CamVid [20], and on several existing end-to-end network architectures.

The main contributions of our paper are:

1. Semantic segmentation is viewed from the dataset itself, and an inference strategy based on the background blurring module (bokeh) is easy to embed into existing semantic networks.
2. According to the characteristics of each dataset, an appropriate strategy for selecting the fuzzy factor σ is proposed.
3. It is verified from the FCN-based network that our bokeh module can improve the segmentation quality of the network without changing any network structure. The segmentation results of BiSeNet [14] on CamVid are improved by 3.7 points, while the performance of HyperSeg [21] on the PASCAL VOC 2012 is improved by 5.2 points after adding the bokeh module.

2. Related Work

Recently, FCN-based approaches have achieved state-of-the-art segmentation results on most existing datasets. We present the segmentation performance of several classical networks on PASCAL VOC 2012 and CamVid datasets, shown in Table 1. This section gives

a brief introduction to FCN [9], and introduces the inspiration of this paper, namely the attention mechanism [22]. This concept is applied to the network to improve the sensitivity of the domain of interest.

Table 1. Accuracies of segmentation models on the PASCAL VOC [19] 2012 val and CamVid [20] test dataset.

Method	Backbone	Mean IOU	
		PASCAL VOC	CamVid
FCN [9]	VGG-16	62.2	-
DFANet [16]	Xception-A	-	64.7
BiSeNet [14]	ResNet-18	-	68.7
DeepLabV3 [12]	ResNet-101	78.5	-
Hyperseg-s [21]	EfficientNet-B1	-	78.4
DeepLabV3+ [13]	Xception-71	80.0	-

2.1. Fully Convolutional Networks

FCN [9] fine-tunes all types of classification as segmentation networks enabling it to fulfil a variety of recognition tasks, and takes full advantage of the classification ability of CNNs. At the same time, three different skip connection methods are used to realize the fusion of diverse scale features. Finally, deconvolution is utilized to restore the size and produce outputs of corresponding sizes. Thus, multi-scale fusion is introduced to effectively combine global information from a deep layer with local information from a shallow layer.

2.2. Attention Mechanism

As an important component of artificial intelligence, the attention mechanism has been extensively used in natural language, image and video processing. By using the advanced information of sample data to guide the forward propagation process of a network, a long-term dependency relationship is established to improve the segmentation accuracy. By introducing the attention mechanism [17,18], our network is more targeted, and we can pay more attention to the object that we concentrate on. For example, in DANNet [23], the authors introduced the Position Attention Module and the Channel Attention Module to selectively aggregate the features of each location and emphasized the interdependence of the network, so as to improve the ability of network segmentation.

2.3. Background Blur

Background blur [24] is to make the depth of field lighter and focus on the subject itself, while weakening the influence of background information of an image. Its mechanism is similar to the attention mechanism in a neural network. Note that the background blurring is performed on the image datasets themselves, while the attention mechanism operates on the high-level feature map. In the subsequent experiments, the combination of background blur and the attention mechanism will be investigated.

2.4. A Change in Thinking

In recent studies, scholars have attempted to solve semantic segmentation with a structure different from FCN (a standard encoder–decoder architecture). Nirkin et al. [21] introduced hypernetworks into semantic segmentation for the first time and proposed a novel network model, HyperSeg. Hypernetworks are networks that generate weight for another network. The crucial role of their HyperSeg is to have the encoder generate weights for the decoder with an advanced features encoded. Zheng et al. [25] serialized the input image into a sequence of image patches and adopted a pure transformer structure,

called SETR, to solve semantic segmentation. Based on the idea of mathematical statistics, we explore the inherent characteristics of the dataset itself and use the proposed bokeh method to improve the performance of existing semantic segmentation models.

3. Proposed Method

In this section, we will elaborate on the bokeh algorithm. We will begin with a brief description of the symbols used. Suppose the whole image is divided into interest and background domains. φ^I denotes the collection of all the pixels that we are interested in; otherwise, the collection of all pixels that we are not interested in is φ^B. The distribution relationship can be simply expressed as Figure 4. For two matrices $A = [a_{ij}] \in \mathbb{R}^{m\times n}$ and $B = [b_{ij}] \in \mathbb{R}^{m\times n}$, matrix $C = [c_{ij}] \in \mathbb{R}^{m\times n}$ is called the Hadamard product [26] of matrices A and B if matrix C satisfies the following condition:

$$c_{ij} = a_{ij} \times b_{ij} \tag{1}$$

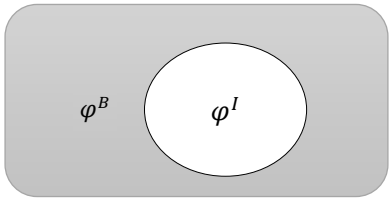


Figure 4. The distribution map of the interested domain φ^I and background domain φ^B.

We found that, on some of the datasets used for semantic segmentation (e.g., PASCAL VOC 2012, etc.), there is an imbalance between different categories in the dataset, and an imbalance between the interest and background domains. For example, of Bicycle (0.29%) and Person (4.57%), both of which are of interest, the former is only one sixteenth of the latter. Moreover, the ratio of the interest domain to background domain is about 1:3, as shown in Table 2. This is not favorable for the segmentation task. A considerable amount of background information either increases the difficulty of training or has no effect on the improvement of segmentation accuracy. Some areas of the background domain may resemble some areas of the interest domain. Learning more background information weakens the role of the usable information. We are more inclined to play down the impact of background information on learning. Learning more and more useful information improves the segmentation accuracy of all categories. We use the effective labeling information in the existing labels to obtain the interest domain of a training set.

Table 2. of 21 categories (with background) on the PASCAL VOC 2012 [19] training dataset.

Class	Background	Aeroplane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair
P(%)	74.81	0.68	0.29	0.85	0.58	0.58	1.67	1.33	2.57	1.09
Cow	**Dining Table**	**Dog**	**Horse**	**MotorBike**	**Person**	**Potted Plant**	**Sheep**	**Sofa**	**Train**	**Monitor**
0.79	1.29	1.66	0.87	1.10	4.57	0.64	0.86	1.38	1.52	0.90

3.1. The Algorithm of Bokeh

For any input image and the corresponding label in the training stage, $img \in \mathbb{R}^{H\times W\times C}$, $GT \in \mathbb{R}^{H\times W}$, $\varphi \in \mathbb{R}^{H\times W\times C}$, where $H \times W$ is the size of the image, and C represents the number of image channels (for RGB, $C = 3$), the background blur module bokeh can be summarized as follows

$$\varphi(img, GT, \sigma) = \varphi^I(img, GT, 1) + \varphi^B(img, GT, \sigma) \tag{2}$$

$$\varphi^I(img, GT, 1) = img(i, j, k) \times [I(i, j) - BGL(i, j)] \times 1 \tag{3}$$

$$\varphi^B(img, GT, \sigma) = img(i, j, k) \times BGL(i, j) \times \sigma \tag{4}$$

where:

$$0 \le i \le H-1,\ 0 \le j \le W-1,\ k = 1, 2, \cdots, C,\ \sigma \in (0, 1]$$

We denote $\varphi^I(img, GT, 1)$ as the interest domain, and $\varphi^B(img, GT, \sigma)$ as the background blur domain. When $BGL(i, j) = 1$, it means that the pixel at (i, j) belongs to the background; otherwise, it belongs to the interest domain. I is an H-by-W matrix of ones. "$*$" is the matrix Hadamard product operator. σ denotes the fuzzy factor, whose value is inversely proportional to the degree of blur. Its selection strategy will be given later. Background label variable $BGL(i, j)$ can be presented as:

$$BGL(i, j) = \begin{cases} 1, & GT(i, j) = 0 \\ 0, & Otherwise \end{cases} \tag{5}$$

Assume that R^B is the proportion of the background field in an image, and R^I is the proportion of the field of interest. Obviously, we obtain $R^B + R^I = 1$, where R^B and R^I are defined as follows:

$$R^B = \frac{Num(\varphi^B)}{H \times W}, R^I = \frac{Num(\varphi^I)}{H \times W} \tag{6}$$

where $Num(\varphi^B)$ is the sum of the pixel number of the background domain, and $Num(\varphi^I)$ is the sum of the pixel number of the interested domain. For PASCAL VOC 2012 train dataset, $Num(\varphi^B) = Num(pixel = 0) + Num(pixel = 255)$, $Num(\varphi^I) = \sum_{i=1}^{20} Num(pixel = i)$.

For the selection of the fuzzy factor σ, we initially set R^{B^*} equal to the background rate of the whole dataset (e.g., for PASCAL VOC train dataset, $R^{B^*} = 0.7481$). Suppose $\sigma \in [1 - R^{B^*}, 1]$. The degree of background blur of each image depends on the distribution of its own background. If its background proportion is larger, the background blurring degree should be aggravated. Hence, σ should be smaller. Conversely, σ should be greater. Specifically, when $R^B = 0$, only the field of interest is involved, the corresponding background blurred factor σ should be maximized. When $R^B = 1$, the background domain is barely included, the σ should be minimized. Let R^B and σ satisfy the linear relation:

$$\sigma = -\alpha \times R^B + \beta \tag{7}$$

such that

$$\begin{cases} R^B = 0, \sigma = 1 \\ R^B = 1, \sigma = 1 - R^{B^*} \end{cases}$$

where $\alpha, \beta \in \mathbb{R}$.

We solve (7), and obtain $\alpha = R^{B*},\ \beta = 1$, Thus, σ could be recasted by

$$\sigma = -R^{B^*} \times R^B + 1 \tag{8}$$

Substituting (3), (4) and (8) into (2), we obtain the formula for the evaluation of the bokeh:

$$\varphi(img, GT, \sigma) = img(:, :, k) \times [I(:, :) - R^{B^*} \times R^B \times BGL(:, :)] \tag{9}$$

where $BGL(:, :)$ is defined in (5).

3.2. The Main Mechanism of Bokeh

The reason why CNNs can achieve various classification tasks is that, after a series of convolution and pooling operations, networks are able to infer the abstract representation (also called advanced feature map) of the input image. The ability of abstract representation depends not only on the performance of the network but also on the characteristics of

the input image. Finding differences between similar objects is much more difficult than finding differences between different objects. For example, we use the same dichotomous network to classify apples and bananas, or tomatoes and cherry tomatoes. The latter is obviously more difficult, precisely because similarities weaken the differences between different objects.

Since semantic segmentation often requires a large amount of sample data, similarities between categories inevitably exist. Therefore, the proposed bokeh method enables differences between categories amplified. Assuming that there are two similar objects in an image, A_b and A_o, where A_b is marked as background and A_o is marked as category I in GT. In a training iteration of the network model, abstract representation learned from A_b is denoted as F_{A_b}, and the abstract representation learned from A_o is denoted as F_{A_o}. Obviously, F_{A_b} and F_{A_o} also have some representation elements in common. The segmentation network learns similar high-level features from two different categories of objects, so it is hard for the network to figure out the features of the category of interest A_o. After bokeh and fuzzy operations imposed on A_b, the similarity between F'_{A_b} and F_{A_o} is abated during iterations, and the network can gradually learn the features of A_o properly.

3.3. Embed into an Existing Network

According to (8), the selection of fuzzy factor σ is merely related to the background rate(R^{B^*} and R^B). Therefore, for different datasets, we only need to calculate the proportion of each category before using the bokeh method, which can be used as a general method. With the FCN network, the overall network architecture after adding our bokeh module is shown in Figure 5. For each block, the convolution stride is one and the stride of pooling is two. There are two more convolution operations with stride 1 after "Conv5 + pool". After each downsampling, the size becomes half of the original. The decoder consists of outputs of three different structures. First, FCN-32s(Output_1) is obtained from the results of "Conv5 + pool" through 32x upsampling. The output of Conv5 + pool with a 2x upsampling is added to the output of "Conv4 + pool" to obtain *Add*_1. Then, FCN-16s(Output_2) is acquired from *Add*_1 through 16x of upsampling. Similarly, as shown in the figure above, we can obtain FCN-8s (Output_3). Detailed structure is shown in Table 3. The network backbone can use AlexNet [27], VGGnet [28], and GoogLeNet [29].

Table 3. Detailed structure of the network. Backbone: VGG16, Input size: 512 × 512, *C*: the number of object classes.

Name	Details	Kernel Size	Stride	Output Size
Conv1 + pool	Conv + BN + ReLU × 2	3 × 3 × 64	1	64 × 512 × 512
	MaxPool	2 × 2	2	64 × 256 × 256
Conv2 + pool	Conv + BN + ReLU × 2	3 × 3 × 128	1	128 × 256 × 256
	MaxPool	2 × 2	2	128 × 128 × 128
Conv3 + pool	Conv +BN + ReLU × 3	3 × 3 × 256	1	256 × 128 × 128
	MaxPool	2 × 2	2	256 × 64 × 64
Conv4 + pool	Conv + BN + ReLU × 3	3 × 3 × 512	1	512 × 64 × 64
	MaxPool	2 × 2	2	512 × 32 × 32
Conv5 + pool	Conv + BN + ReLU × 3	3 × 3 × 512	1	512 × 32 × 32
	MaxPool	2 × 2	2	512 × 16 × 16
Deconv_32x	ConvTranspose	32 × 32 × *C*	32	*C* × 512 × 512
Deconv_16x	ConvTranspose	16 × 16 × *C*	16	*C* × 512 × 512
Deconv_8x	ConvTranspose	8 × 8 × *C*	8	*C* × 512 × 512

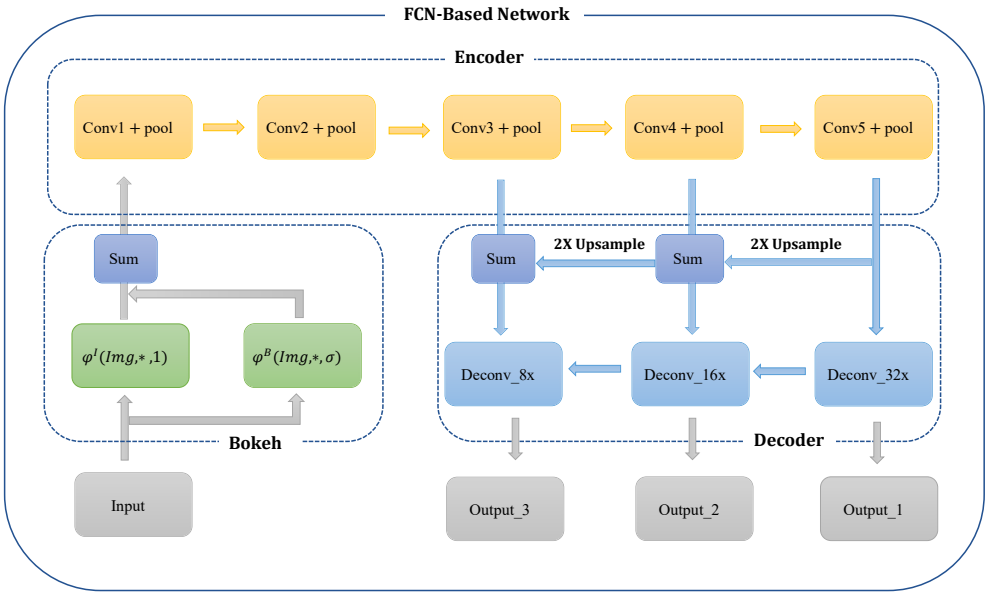


Figure 5. The FCN [9] network architecture after adding bokeh is illustrated in the figure above. The entire network can be subdivided into bokeh, encoder and decoder. Bokeh is described in Section 3. The encoder is composed of 5 "ConvX + pool" downsampling blocks. The value of "*" is GT (ground truth) in the training stage and CP (coarse prediction) in the verification stage.

4. Experimental Results

FCN [9], BiseNet [14], and HyperSeg [21] are selected as our segmentation networks, and relevant experiments are carried out on PASCAL VOC 2012 [19] and CamVid [20] benchmarks. A brief review of the corresponding datasets and metrics will be first presented. Following this, implementation and explanation of experiments will be given.

PASCAL VOC 2012: As one of the rockstars used for semantic segmentation, PASCAL VOC 2012 covers not only indoor and outdoor scenes, but also night-time scenes with a total of 21 semantic categories (20 categories of interest and a class for the background). The whole dataset contains 4369 images, 1464 of which are used for training, 1449 for validation and 1456 for testing. The training set and validation set adopt full annotation, while the test set does not provide labels. The capacity has been later expanded in SBD [30] to reach 10,582 training samples.

CamVid: As a small-scale urban street view dataset, CamVid includes a total of 701 fully annotated images, 367 of which are employed to train, 101 for validation and 233 for testing. The CamVid dataset consists of 11 semantic categories (e.g., cars, buildings, billboards, etc.) and an Unlabelled class. Each image has the same resolution: 720×960.

Metrics: Let n_{ij} be the number of pixels that class i is predicted to be class j, and C be the number of object classes (including the background class). We compute four indices: Pixel Acc, Mean Acc, Mean IOU and F.W IOU, as defined below. Naturally, the higher the values are, the better network performance is.

- Pixel accuracy(Pixel Acc): $\sum_{i=1}^{C} n_{ii} / \sum_{i=1}^{C} \sum_{j=1}^{C} n_{ij}$;
- Mean pixel accuracy(Mean Acc):$(1/C) \sum_{i=1}^{C} n_{ii} / \sum_{i=1}^{C} \sum_{j=1}^{C} n_{ij}$;
- Mean intersection over union(Mean IOU):$(1/C) \sum_{i=1}^{C} (n_{ii} / (\sum_{j=1}^{C} n_{ij} + \sum_{j=1}^{C} n_{ji} - n_{ii}))$;
- Frequency weight intersection over union (F.W IOU): $(1/\sum_{i=1}^{C} \sum_{j=1}^{C} n_{ij}) \sum_{i=1}^{C} ((\sum_{j=1}^{C} n_{ij} n_{ii}) / (\sum_{j=1}^{C} n_{ij} + \sum_{j=1}^{C} n_{ji} - n_{ii}))$.

4.1. Implementation Protocol

We reconstruct the classical FCN [9], BiSeNet [14], and HyperSeg [21] network. In order to more objectively evaluate the impact of background information on segmentation

accuracy, we remove all data augmentation (except for cropping size) in the original paper. The reconstructed networks are represented by (Re)FCN-8s, (Re)FCN-16s, (Re)FCN-32s, (Re)HyperSeg, and (Re)BiSeNet, respectively. Our reconstruction results are a little bit lower than the original results because we did not add a mass of data augmentation. However, our focus is to demonstrate the feasibility of our method, rather than narrowing the gap with the original paper.

Training details: For the CamVid [19] dataset, an Adam optimizer was used, with batch size 8, initial learning rate 1×10^{-4}, and weight decay 1×10^{-4} in training. Similar to Deeplab series[11–13], the "poly" learning rate attenuation strategy was also adopted, and the last learning rate was multiplied by $(1 - \frac{iter}{iter_{max}})^{power}$, where $power = 0.9$, after each iteration. For the PASCAL VOC 2012 [19] dataset, parameters were set with batch size 12, and weight decay 2×10^{-4} in training. After every 50 epochs, the learning rate decayed to half of the last one.

Data augmentation: No additional operations are required except clipping. For Camvid, images processed by SegNet [31] are used as input in this paper, and these images are 360×480. The PASCAL VOC 2012 dataset is clipped to a fixed size as input.

4.2. Ablation for Bokeh

Applying bokeh to multiple segmentation networks on two datasets, comparative experiments were made. The experimental results of three FCN network architectures and HyperSeg on the PASCAl VOC dataset are shown in Table 4. As can be seen, the mean IOU of the four mentioned network architectures ((Re)FCN-32s, (Re)FCN-16s, (Re)FCN-8s, and (Re)HyperSeg) with bokeh are improved by 4.7, 4.6, 4.8, and 5.2 points, respectively. At the same time, the specific precision of FCN-8s before and after adding the bokeh module on the PASCAL VOC 2012 Val dataset is given, as illustrated in Table 5. Note that the segmentation accuracy of an individual category is significantly improved, except for 5 out of 84 comparison items.

Table 4. The segmentation performance of three variants of FCN on the PASCAL VOC 2012 validation set, "(Re) FCN-XXs + bokeh" means the bokeh module is added to "(Re)FCN-XXs".

Method	Pixel Acc	Mean Acc	Mean IOU	F.W IOU
(Re)FCN-32s [9]	80.7	74.7	52.9	78.2
(Re)FCN-16s [9]	81.6	75.4	54.6	79.5
(Re)FCN-8s [9]	82.3	76.5	55.6	80.5
(Re)HyperSeg [21]	93.8	83.4	72.8	89.1
(Re)FCN-32s + bokeh	83.2	77.1	57.6	82.2
(Re)FCN-16s + bokeh	83.9	78.0	59.2	83.1
(Re)FCN-8s + bokeh	84.6	78.0	60.4	83.8
(Re)HyperSeg + bokeh	95.6	86.2	78.0	92.1

Considering the results in Table 2 and Table 5 together, we find relatively small categories, such as bicycle (0.29%), boat (0.58%) and potted plant (0.64%), make a significant contribution to accuracy improvement. Note that the segmentation accuracy of exceptional categories, such as "Cow" and "Dinning Table", decrease inversely. This is because while blurring the background, it depresses the context information to some extent. We considered it from two aspects. One is to employ the fusion of "(Re)FCN-8s" and "(Re)FCN-8s + bokeh", named "(Re)FCN-8s + Fusion". Another is to confine the scope of the background blur field to ensure rich context information is preserved, named "(Re)FCN-8s + Shrink." The experimental results demonstrate, as shown in Table 5, that these two methods can avoid the accuracy decrease in specific categories. However, the improvement of the overall segmentation accuracy is not as good as before. As a result, the accuracy decrease in some individual categories is permitted. Qualitative examples on this dataset are shown in Figure 6.

Table 5. Comparison of detailed accuracy of FCN-8s before and after adding bokeh module on the PASCAL VOC 2012 Val dataset.

Method	Background	Aero Plane	Bicycle	Bird	Boat	Bottle	Bus	Car	Cat	Chair	Cow	Dining Table	Dog	Horse	MotorBike	Person	Potted Plant	Sheep	Sofa	Train	Monitor	Mean IOU
(Re)FCN-8s [9]	85.30	57.04	25.65	57.23	48.19	53.33	70.88	59.84	67.83	31.44	49.49	60.06	59.75	52.87	57.36	59.74	36.11	51.46	48.23	69.54	65.32	55.6
(Re)FCN-8s + Fusion	87.81	59.58	29.98	61.31	49.68	55.17	74.37	61.52	68.40	32.57	52.73	60.86	64.43	56.33	61.52	61.73	40.85	58.89	49.20	74.42	69.21	58.6
Improved	2.51	2.54	4.33	4.08	1.49	1.84	3.49	1.68	0.57	1.13	3.24	0.8	4.68	3.46	4.16	1.99	4.74	7.43	0.97	4.88	3.89	3.0
(Re)FCN-8s + Shrink	88.14	60.38	31.37	59.44	52.52	55.58	76.41	61.98	70.58	33.70	52.66	62.31	63.62	56.01	61.00	62.70	41.43	60.11	50.22	75.00	68.33	59.2
Improved	2.84	3.34	5.72	2.21	4.33	2.25	5.53	2.14	2.75	2.26	3.17	2.25	3.87	3.14	3.64	2.96	5.32	8.65	1.99	5.46	3.01	3.6
(Re)FCN-8s + bokeh	89.05	64.16	33.14	63.87	57.57	54.40	73.63	62.88	73.19	34.54	49.28	59.69	62.20	59.46	61.05	64.01	43.42	61.23	52.94	76.80	71.07	60.4
Improved	3.75	7.12	7.54	6.64	9.38	1.07	2.75	3.04	5.39	3.1	−0.21	−0.37	2.45	6.59	3.69	4.27	7.31	9.77	4.71	7.26	5.75	4.8
(Re)HyperSeg [21]	94.11	86.67	60.39	87.2	75.01	69.05	92.1	78.98	85.85	31.22	78.68	54.94	77.56	75.9	76.9	79.22	47.68	79.84	48.92	82.08	65.76	72.8
(Re)HyperSeg + bokeh	96.81	90.91	68.04	87.29	84.41	82.18	89.31	84.18	85.49	32.98	83.46	68.24	75.01	85.93	79.46	82.22	58.73	80.16	54.38	85.26	84.57	78.0
Improved	2.7	4.24	7.65	0.09	9.4	13.13	−2.79	4.2	−0.36	1.76	4.78	13.3	−2.55	10.03	2.56	3.00	11.05	0.32	5.46	3.18	18.81	5.2

The bokeh module improves the segmentation results on the CamVid val dataset by 3.7 in Mean IOU, as shown in Table 6. This demonstrates that the proposed bokeh module is easily embedded into a real-time network architecture. In view of the consequences of PASCAL VOC 2012, bokeh plays a vital role in the class with a small proportion of the dataset. As for CamVid, the background occupies a relatively small proportion, but the accuracy increase is clear. The proportion of the CamVid training dataset by category is presented in Table 7.

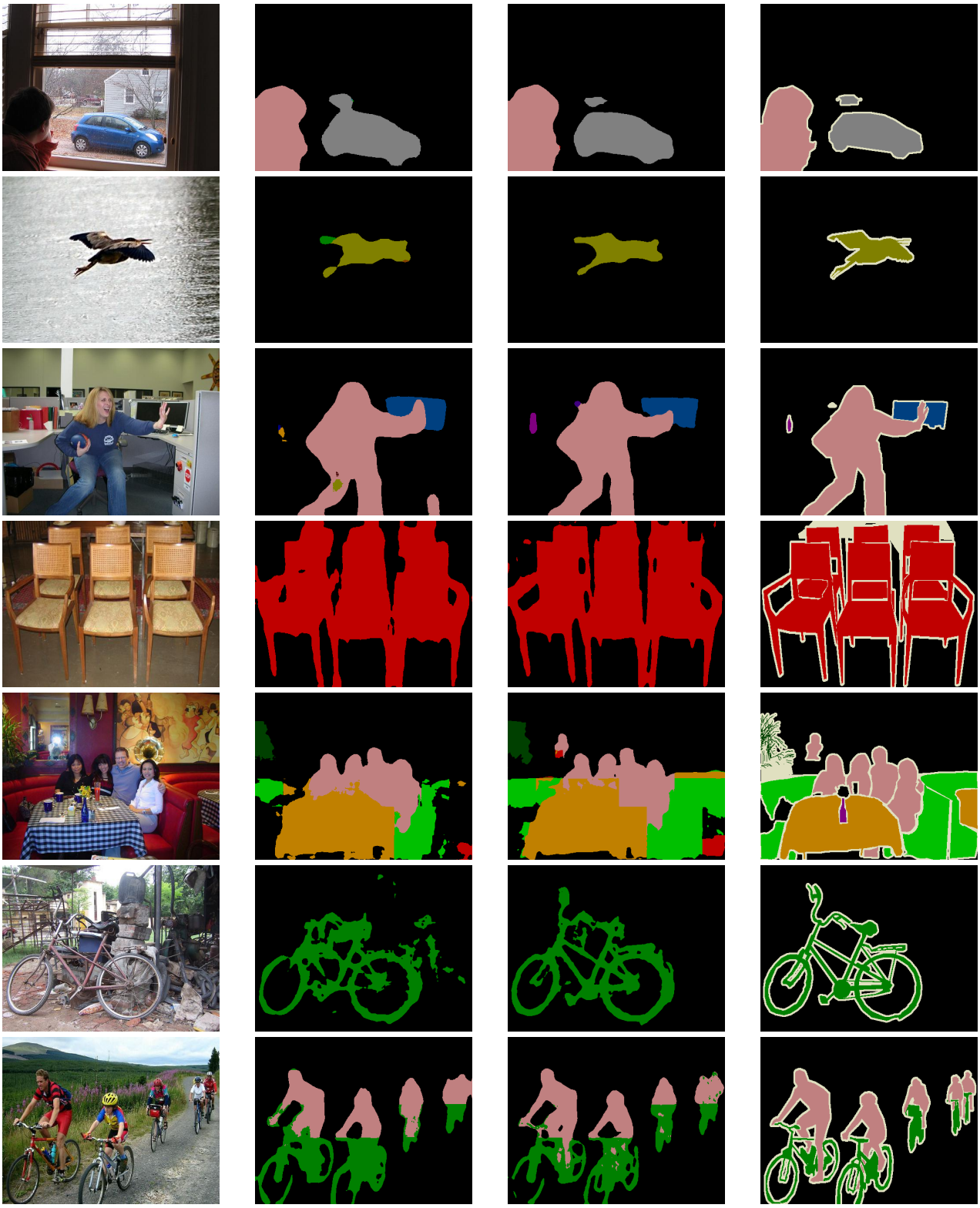
Table 6. Accuracy result on the CamVid val dataset. After adding the bokeh module, the result is improved by 3.7 points.

Method	Pixel Acc	Mean Acc	Mean IOU	F.W IOU
(Re)BiSeNet [14]	80.3	74.0	60.7	84.5
(Re)HyperSeg [21]	94.5	85.8	77.3	89.9
(Re)BiSeNet + bokeh	81.8	76.3	64.4	85.9
(Re)HyperSeg + bokeh	96.0	87.8	80.9	92.6

Table 7. The category proportion of the CamVid training dataset.

Class	Sky	Buliding	Pole	Road	Sidewalk	Tree	SignSymbol	Fence	Car	Pedestrain	Bicyclist	Unlabelled
P(%)	16.85	23.26	0.98	31.66	4.49	9.72	1.17	1.13	5.87	0.64	0.29	3.95

It is clear from Table 8 that the improvement in the Unlabelled category is the most significant. Analyzing the characteristics of CamVid, it can be observed that low background proportion of the dataset itself and diversity of categories in each image leads to this. After adding the bokeh module, the image changes are not noticeable compared to the ones before the fuzzification. However, this slight improvement occurs in almost all of the Unlabelled category. Qualitative examples on this dataset are given in Figure 7.

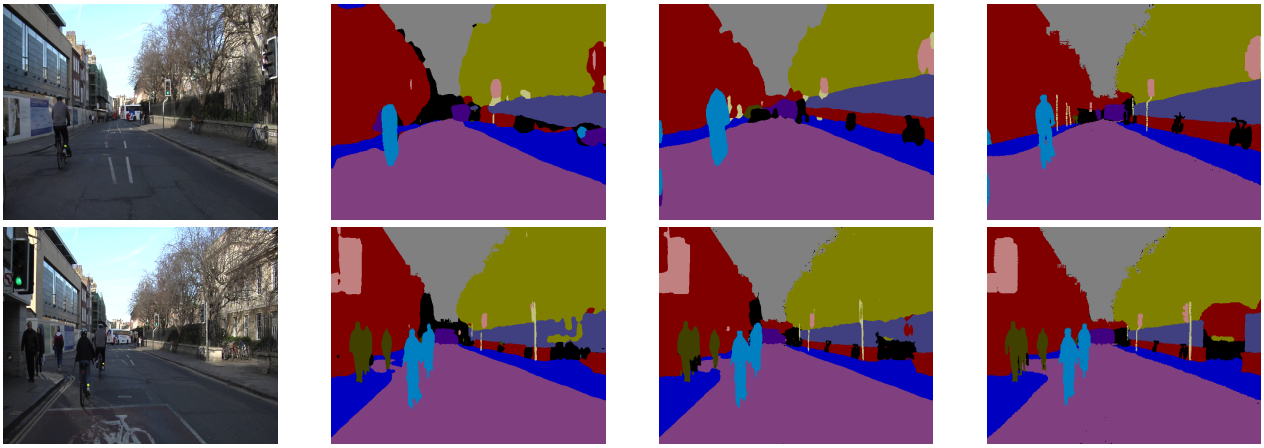


(**a**) (**b**) (**c**) (**d**)

Figure 6. Examples of the output before and after adding the bokeh module on the PASCAL VOC 2012 dataset, our resulting contour is much smoother. The first three rows are the results of experiments on FCN, and the last four rows are the results of experiments on Hyperseg. (**a**) Image; (**b**) (Re)Seg.Network; (**c**) (Re)Seg.Network+bokeh; (**d**) Ground Truth.

Table 8. Accuracy result on the CamVid val dataset.

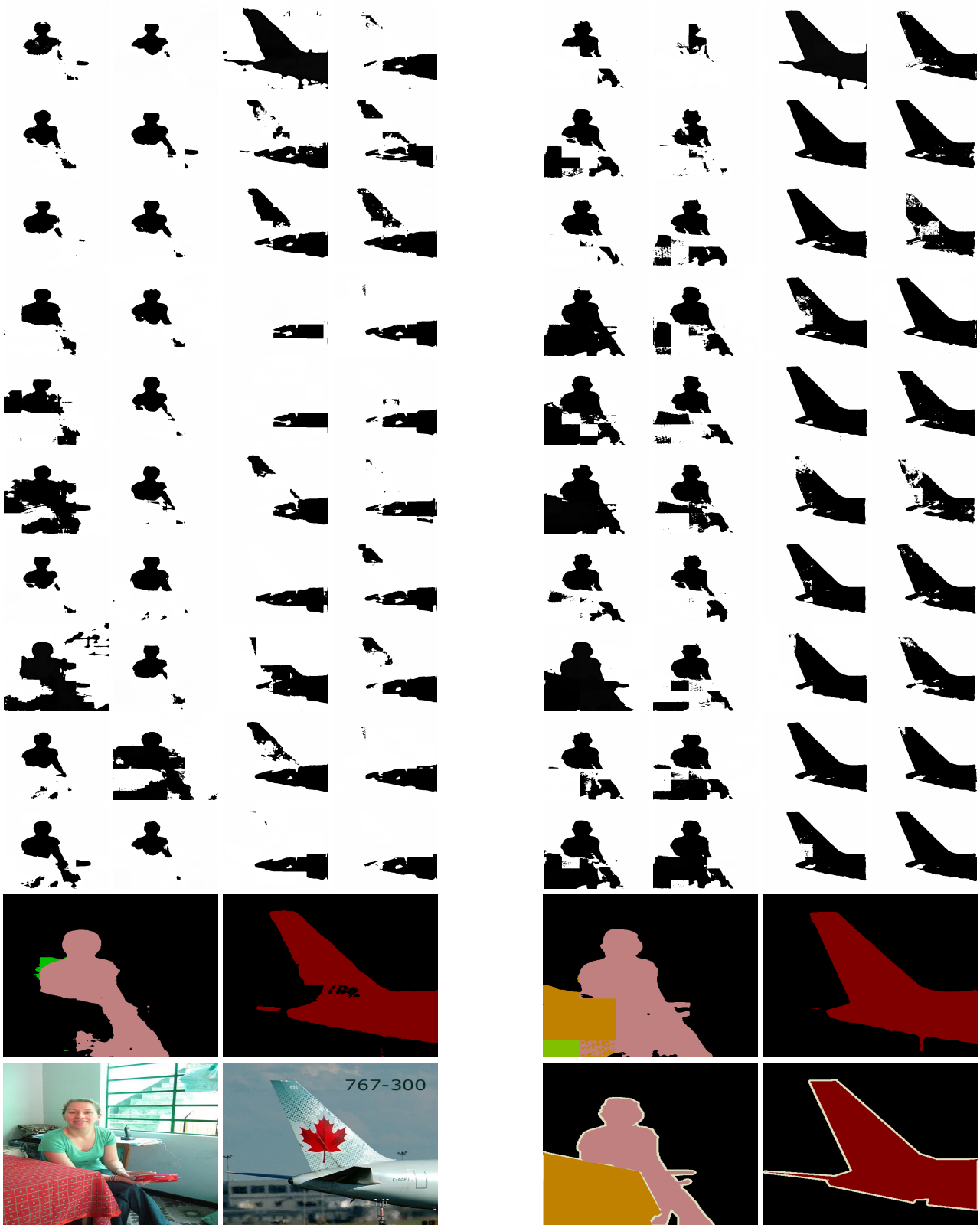
Method	Sky	Buliding	Pole	Road	Sidewalk	Tree	SignSymbol	Fence	Car	Pedestrain	Bicyclist	Unlabelled	Mean IOU
(Re)BiSeNet [14]	92.15	82.11	10.48	94.34	77.81	89.16	39.03	55.23	68.58	31.87	63.59	23.65	60.7
(Re)HyperSeg [21]	94.39	88.66	36.93	97.87	90.96	91.70	74.94	75.68	90.95	66.98	86.66	31.66	77.3
(Re)BiSeNet + bokeh	93.03	85.55	9.15	94.61	78.83	89.58	42.61	52.72	65.74	38.26	59.16	63.12	64.4
Improved	0.88	3.44	−1.33	0.27	1.02	0.4	3.58	−2.51	−2.85	6.39	−4.43	39.47	3.7
(Re)HyperSeg + bokeh	95.53	92.40	31.28	97.96	92.07	92.43	72.85	78.22	91.65	70.33	87.15	68.68	80.9
Improved	1.14	3.74	−5.6	0.09	1.11	0.73	−2.09	2.54	0.70	3.35	0.49	37.02	3.6



(**a**) (**b**) (**c**) (**d**)

Figure 7. Examples of the output before and after adding the bokeh module on the CamVid dataset. It is obvious that the network is sensitive enough to recognize the background (the black part in picture (**b**,**c**)) after adding the bokeh module. The first row is the results of experiments on BiSeNet and the second row on HyperSeg. (**a**) image; (**b**) (Re)Seg.Network; (**c**) (Re)Seg.Network+bokeh; (**d**) ground truth.

We compared the advanced feature maps of 20 channels (excluding background channels) in the last layer of HyperSeg [21] before and after adding bokeh. The sensitivity of the network to the categories of interest is higher after adding bokeh, as shown in Figure 8.



(**a**) (**b**)

Figure 8. Comparison of advanced feature maps and prediction: (**a**) feature map and prediction of the HyperSeg network before adding bokeh, the last row is the input image. (**b**) Feature map and prediction of the HyperSeg network after adding bokeh, the last row is the ground truth. The feature map does not include background channels.

5. Conclusions

In this paper, we propose a semantic segmentation method based on background blurring, which adaptively processes the input image background via the fuzzy factor σ, without changing the original network structure or introducing additional parameters, to expand differences between background and foreground and guide the network segmentation. The selection of σ is determined by the overall background rate R^{B^*} of the dataset and the background rate R^B of the current image. The former determines the approximate range of its value, while the latter determines its specific value. Compared to the attention mechanism in the network layer, bokeh plays the same role in the dataset, by weakening the background information to highlight the features of the foreground. Moreover, our approach can be lightly embedded into the existing segmentation network. As our experiments show, our method achieves competitive performance on PASCAL VOC 2012 and CamVid, with mean IOU increased by 5.2 and 3.7, especially for the small proportion category in the dataset. The main limitation of this study is that our bokeh method relies on the existing segmentation network, and the performance of the existing segmentation network directly determines whether we can accurately trace the background. Different segmentation networks selected may result in diverse results. Therefore, a natural progression of this work is how to efficiently segment the foreground and background without relying on the current network. In addition, adding classical image processing methods and how to encode and decode contour information effectively will be the focus of this paper in the future.

Author Contributions: Writing—original draft, H.L.; Writing—review and editing, C.L. and A.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded in part by Sichuan Science and Technology Program grant number 2019YJ0541, the Open Project of the Key Lab of Enterprise Informationization and Internet of Things of Sichuan Province grant number 2021WZY02, the Open Fund Project of Artificial Intelligence Key Laboratory of Sichuan Province grant number 2018RYJ02, Postgraduate course construction project of Sichuan University of Science and Engineering grant number YZ202103 and Graduate innovation fund of Sichuan University of Science and Engineering grant number Y2021099.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Huang, Z.; Wang, X.; Huang, L.; Huang, C.; Wei, Y.; Shi, H.; Liu, W. CCNet: Criss-Cross Attention for Semantic Segmentation. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27 October–2 November 2019; pp. 603–612.
2. Vu, T.-H.; Jain, H.; Bucher, M.; Cord, M.; Pérez, P. ADVENT: Adversarial entropy minimization for domain adaptation in semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 2512–2521.
3. Yuan, Y.; Chen, X.; Wang, J. Object-Contextual Representations for Semantic Segmentation. *arXiv* **2020**, arXiv:1909.11065.
4. Li, Y.; Yuan, L.; Vasconcelos, N. Bidirectional learning for domain adaptation of semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 6929–6938.
5. Cortinhal, T.; Tzelepis, G.; Aksoy, E. SalsaNext: Fast Semantic Segmentation of LiDAR Point Clouds for Autonomous Driving. *arXiv* **2020**, arXiv:2003.03653.
6. Zhang, X.; Chen, Z.; Wu, Q.M.J.; Cai, L.; Lu, D.; Li, X. Fast semantic segmentation for scene perception. *IEEE Trans. Ind. Inf.* **2019**, *15*, 1183–1192. [CrossRef]
7. Ko, T.; Lee, S. Novel Method of Semantic Segmentation Applicable to Augmented Reality. *Sensors* **2020**, *20*, 1737. [CrossRef] [PubMed]
8. Lin, C.; Yan, B.; Tan, W. Foreground detection in surveillance video with fully convolutional semantic network. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; pp. 4118–4122.

9. Shelhamer, E.; Long, J.; Darrell, T. Fully convolutional networks for semantic segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 640–651. [CrossRef] [PubMed]
10. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *J. Comput. Sci.* **2014**, 4, 357–361.
11. Chen, L.-C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *40*, 834–848. [CrossRef] [PubMed]
12. Chen, L.-C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking atrous convolution for semantic image segmentation. *arXiv* **2017**, arXiv:1706.05587.
13. Chen, L.-C.; Zhu, Y.; Papandreou, G.; Schroff, F.; Adam, H. Encoder-Decoder with Atrous Separable Convolution for Semantic Image Segmentation. *arXiv* **2018**, arXiv:1802.02611.
14. Yu, C.; Wang, J.; Peng, C.; Gao, C.; Yu, G.; Sang, N. BiSeNet: Bilateral segmentation network for real-time semantic segmentation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 334–349.
15. Paszke, A.; Chaurasia, A.; Kim, S.; Culurciello, E. Enet: A deep neural network architecture for real-time semantic segmentation. *arXiv* **2016**, arXiv:1606.02147.
16. Li, H.; Xiong, P.; Fan, H.; Sun, J. DFANet: Deep feature aggregation for real-time semantic segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 9522–9531.
17. Lin, G.; Shen, C.; Hengel, A.V.; Reid, I. Efficient Piecewise Training of Deep Structured Models for Semantic Segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 3194–3203.
18. Lin, Z.; Feng, M.; Santos, C.D.; Yu, M.; Xiang, B.; Zhou, B.; Bengio, Y. A Structured Self-attentive Sentence Embedding. *arXiv* **2017**, arXiv:1703.03130.
19. Everingham, M.; van Gool, L.; Williams, C.K.I.; Winn, J.; Zisserman, A. The Pascal visual object classes (VOC) challenge. *Int. J. Comput. Vis.* **2010**, *88*, 303–338. [CrossRef]
20. Brostow, G.J.; Fauqueur, J.; Cipolla, R. Semantic object classes in video: A high-definition ground truth database. *Pattern Recognit. Lett.* **2009**, *30*, 88–97. [CrossRef]
21. Nirkin, Y.; Wolf, L.; Hassner, T. HyperSeg: Patch-wise Hypernetwork for Real-time Semantic Segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 4061–4070.
22. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention is all you need. In Proceedings of the Advances in Neural Information Processing Systems, Long Beach, CA, USA, 4–9 December 2017; pp. 5998–6008.
23. Fu, J.; Liu, J.; Tian, H.; Li, Y.; Bao, Y.; Fang, Z.; Lu, H. Dual attention network for scene segmentation. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Long Beach, CA, USA, 15–20 June 2019; pp. 1–10.
24. Ignatov, A.D.; Patel, J.; Timofte, R. Rendering Natural Camera bokeh Effect with Deep Learning. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, 14–19 June 2020; pp. 1676–1686.
25. Zheng, S.; Lu, J.; Zhao, H.; Zhu, X.; Luo, Z.; Wang, Y.; Fu, Y.; Feng, J.; Xiang, T.; Torr, P.H.; et al. Rethinking Semantic Segmentation from a Sequence-to-Sequence Perspective with Transformers. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 19–25 June 2021; pp. 6881–6890.
26. Horn, R.; Johnson, J. *Matrix Analysis*; Cambridge University Press: Cambridge, UK, 2012.
27. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* **2012**, *25*, 1097–1105. [CrossRef]
28. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
29. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.E.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
30. Hariharan, B.; Arbelaez, P.; Bourdev, L.; Maji, S.; Malik, J. Semantic contours from inverse detectors. In Proceedings of the 2011 International Conference on Computer Vision, Barcelona, Spain, 6–13 November 2011; pp. 991–998.
31. Badrinarayanan, V.; Kendall, A.; Cipolla, R. SegNet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef] [PubMed]