IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received May 8, 2020, accepted June 1, 2020, date of publication June 4, 2020, date of current version June 23, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2999942

Image Object Extraction Based on Semantic Segmentation and Label Loss

XIAORU WANG[1], PEIRONG XU[1], ZHIHONG YU[2], AND FU LI[3], (Senior Member, IEEE)

[1]Beijing Key Laboratory of Network System and Network Culture, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2]Intel China Research Center, Beijing 100190, China
[3]Department of Electrical and Computer Engineering, Portland State University, Portland, OR 97207-0751, USA

Corresponding author: Xiaoru Wang (wxr@bupt.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 61672108 and Grant 61976025.

ABSTRACT Object extraction refers to the operation of obtaining an object area from an image based on a small amount of mark information given by users, which is a key step in image processing. In order to obtain a complete object profile, current methods usually require a large number of manual annotations, especially for objects with irregular contours. Since traditional algorithms rely on low-level pixel features without semantic information, and are based on obvious mathematical assumptions (ie, strong inductive bias), it is difficult to completely identify objects. At present, in order to improve the integrity of object extraction, semantic segmentation-based methods increase the complexity and latancy by adding more pre-processing and post-processing steps. In this paper, we propose a novel model named IOEBSS, which includes a fast binary plane pre-processing, an improved Deeplab v3+ semantic segmentation model, and an auxiliary loss function named Label Loss. Through the fast binary plane pre-processing, the model can accelerate the transformation of interactive inputs. The improved semantic segmentation model makes the extracted results more semantically complete, and Label Loss is more conducive to gradient flow and accelerates training convergence. For the above reasons, IOEBSS can accurately and quickly identify objects with complex contours and colors. On Pascal VOC and COCO datasets, compared to current methods, IOEBSS has a significant improvement in accuracy, inference speed, and convergence speed.

INDEX TERMS Label loss, object extraction, semantic segmentation.

I. INTRODUCTION

Object extraction is a key operation in image processing. It determines the area to be reserved and discarded based on users' interactive inputs containing a small amount of foreground and background information, enabling users to perform subsequent image processing operations such as image fusion, shape and position editing, etc. MagicWand [1] is one of the most commonly used object extraction tools of PhotoShop. The tool is based on the Region Growing algorithm, which exploits a point given by users as a seed point to expand into a larger area. During the region merging process, if the similarities between the adjacent points and the points of the edge of the region are less than the threshold, the adjacent points will be merged into the selected region until no points satisfies the condition. Although Region Growing does not allow background points to be used as seed points, it allows multiple seed points to be merged in parallel, speeding up the extraction. Region Growing relies on the merging of the low-level features among pixels. When the similarities in the low-level features of two pixels are large, although they belong to the same semantic class, they will not be merged into the same region. Eventually, the algorithm has a problem that the extraction of the object area is incomplete.

Poisson Matting [2] is an image fusion tool. It combines the gradient fields of the ROI and the background image, solves the divergence according to the gradient field, and combines the calculated boundary constraints to calculate the fused pixel values. When the background image is blank, the image formed by the fused pixel values is the result of object extraction. Therefore, Poisson Matting is equivalent to the object extraction algorithm that specifies the background image. This algorithm needs Poisson reconstruction. In the reconstruction process, a large number of Poisson equations need to be solved, which makes the algorithm time intensive and difficult to achieve real-time interactive object extraction.

The associate editor coordinating the review of this manuscript and approving it for publication was Eduardo Rosa-Molinar.

GraphCut [3] is a classic interactive segmentation algorithm. It regards the image as a graph model. Correspondingly, the nodes in the graph model are the pixels in the image, and the edge between every two nodes is defined by the feature similarity between the two pixels. The source node and the terminal node are used to transform the problem before the solution to a minimum cut problem. It can be solved according to the maximum flow minimum cut theorem. When the similarities in the low-level pixel features are large, GraphCut extracts better and its principle is clearer than other traditional algorithms. However, the reason why the traditional object extraction algorithms are not widely used is mainly because of the semantic gap. These algorithms typically rely on low-level pixel features. Since different combinations of the same low-level features may form different semantic information, and the same semantic information may be composed of different low-level features, it is difficult for the traditional object extraction algorithms to obtain the object region completely and accurately. For example, for GraphCut, when the object area belongs to the same semantic class but the pixel features are different, it is often not divided into the same area, which usually leads to incomplete extraction of the object area. This problem only turned around after the emergence of semantic segmentation.

In order to step over the semantic gap, the industry has proposed a large number of algorithms based on semantic segmentation [1], [4]–[10]. Depending on the type of interactive inputs, object extraction can be divided into trimaps and strokes. The former divides the image into three areas including foreground, background, and to be divided areas. The latter uses casual graffiti to mark the foreground and background on the image, and the rest is the area to be divided. These semantic segmentation based algorithms have two disadvantages. On the one hand, some semantic segmentation-based algorithms [4], [5], [10] use trimaps as the interactive input method, which requires users to provide more priori information and get more limited in real scenarios. On the other hand, in order to improve the integrity of the object extraction, these algorithms [1], [6]–[9] often require complex pre-processing and post-processing steps, which are not only complicated but also inefficient.

In this paper, our model consists mainly of a fast binary plane preprocessing, an improved Deeplab v3+ semantic segmentation model, and an auxiliary loss function (Label Loss).

First we consider the complexities of the pre-processing and post-processing procedures mentioned above. The previous method [11] converts the interactive inputs into two multi-valued planes according to the Euclidean distance, and the traversal brings a lot of time overhead. At the same time, post-processing with traditional algorithms further increases overhead. Our fast binary plane pre-processing converts users' inputs directly into two 0-1 binary planes without traversing the image pixels, and the high precision of the model is sufficient to replace the post-processing. Therefore, the model structure is simple and the running efficiency is high.

In order to improve the semantic integrity of the object extraction region, we have improved the semantic segmentation model, DeepLab v3+, on the VOC dataset. We fine-tune the number of input channels and output channels for IOEBSS to apply to the object extraction task. At the same time, in order to more accurately understand the interactive input information and realize the migration from semantic segmentation to object extraction, IOEBSS replaces the backbone network with the more versatile ResNet-101. This design enhances the ability to extract semantic information and significantly improves the accuracy of the model.

Since most semantic segmentation models have large capacity and many parameters, training is very time consuming, especially starting from scratch. This is a waste of computing resources and time. In response to this problem, considering the specificity of the object extraction with interactive inputs, we designed an auxiliary loss function named Label Loss, by requiring the distance between the network output and the interactive inputs to be as small as possible, which is more favorable for gradient flowing. It greatly speeds up the convergence of the network convergence during training.

The rest of this paper is organized as follows. Sect 2. reviews the related works, Sect 3. introduces our algorithm principles, Sect 4. describes our experimental process and results, and finally we summarizes our work in Sect 5.

II. RELATED WORK

Object extraction has a long-term research. The earliest research focused on the mapping algorithm, their goal was to get the transparency α corresponding to each pixel. Poisson Matting [2] transforms it into a problem that solves the Poisson equation. Bayes Matting [12] expresses the problem as a Bayesian form and obtains a transparency matrix by solving the maximum posterior probability. Closed Form Matting [13] assumes that the color of a partial window can be represented as a linear combination of two colors, which can be solved in a closed manner without explicitly estimating the front background. KNN Matting [14] is a non-local map that assumes that the transparency of non-local pixels can be obtained by a linear combination of the transparency of pixels that are close together.

Superpixels group perceptually similar pixels to create visually meaningful entities while heavily reducing the number of primitives for subsequent processing steps [15]. Among these methods, Simple Linear Iterative Clustering, or SLIC [16], is the most widely used one in image processing. Based on k-means with weighted distance metrics, SLIC limits search space to areas proportional to superpixel size to reduce complexity. However, the local area association is unable to catch long-range and global context information, which leads to inaccurate object extraction.

Watershed Segmentation [17], [18] is a morphological segmentation method imitating the map immersion process. It combines edge detection and region growing to achieve neighbourhood-based segmentation and get multiple

connected areas. Therefore, for many disconnected object areas(e.g. partially occluded parts), Watershed Segmentation usually cannot get complete segmentation results.

Active Contour [19], [20], or Snake, is a object segmentation method based on object contours. It takes an image and an initial contour as input and minimizes internal force and external force to make the contour close to the edge of the object by iterations. However, due to the energy equation is not intrinsic, this method encounters difficulties when the object area consists of multiple parts.

The above methods are based on certain mathematical assumptions, such as independence assumptions, linear assumptions, etc. In many cases the assumptions are not true and the interactive input processing is not sufficient.

A direct study of the object extraction algorithm began with interactive segmentation. GraphCut [3] and GrabCut [21] are representatives based on graph theory. GraphCut [3] views this problem as the separation between two point sets containing the source and terminal points. GrabCut is an iterative method to further improve the accuracy of the segmentation. Lazy Snapping [22] uses clustering and precomputation based on GraphCut to provide real-time feedback to users. Normalized Cut [23] uses the cluster grouping technique to calculate similar pixel feature regions based on the graph. These algorithms are based on fewer mathematical assumptions than matting-like algorithms and have a significant improvement in effect. However, due to the semantic gap, these algorithms cannot identify complex objects, so their application is limited in real scenes. At the same time, graph-based algorithms require a lot of calculations, especially as the image resolution increases, the time spent will grows at a geometric rate.

In recent years, since FCN [24] established a basic framework for semantic segmentation, the use of codec structure, pre-trained model, abandoning fully connected layers, and feature fusion, have become the main technologies of semantic segmentation. Later studies proposed more techniques, such as dilated convolution [25], [26], conditional random fields [27], [28], pyramid feature modules [25], [29], and neural architecture searches [30], [31]. Based on FCN, a model of object extraction is designed in [11], and post-processing with GraphCut greatly exceeds the performance of the previous methods. But the model contains complex pre-processing and post-processing, which is too cumbersome and inefficient. In addition, there are methods such as AlphaGan [4] and Deep Image Matting [5], but they require the trimap planes as inputs, which needs more prompts from users and does not conform to the application in real scenes.

The above methods based on semantic segmentation often require trimaps as inputs, which leads to users having to give enough prompt information through cumbersome operations, and the application of in real scenes is limited. Moreover, the complicated pre-processing and post-processing brought by the improvement of precision leads to excessive overhead, which greatly reduces the running efficiency.

III. OBJECT EXTRACTION MODEL BASED ON SEMANTIC SEGMENTATION AND LABEL LOSS

As shown in Fig.1, the overall structure of our model is divided into three stages. In the first stage, to transform the users' inputs into two binary planes, we propose the fast binary plane pre-processing, which significantly accelerate processing. In the second stage, to get accurate and complete object areas, we exploit the improved semantic segmentation model, which transforms the 5-channel tensor into an object area mask with better semantic integrity. In the third stage, to accelerate gradient flow, we design an auxiliary loss function named Label Loss, which makes the network converge much faster during training.

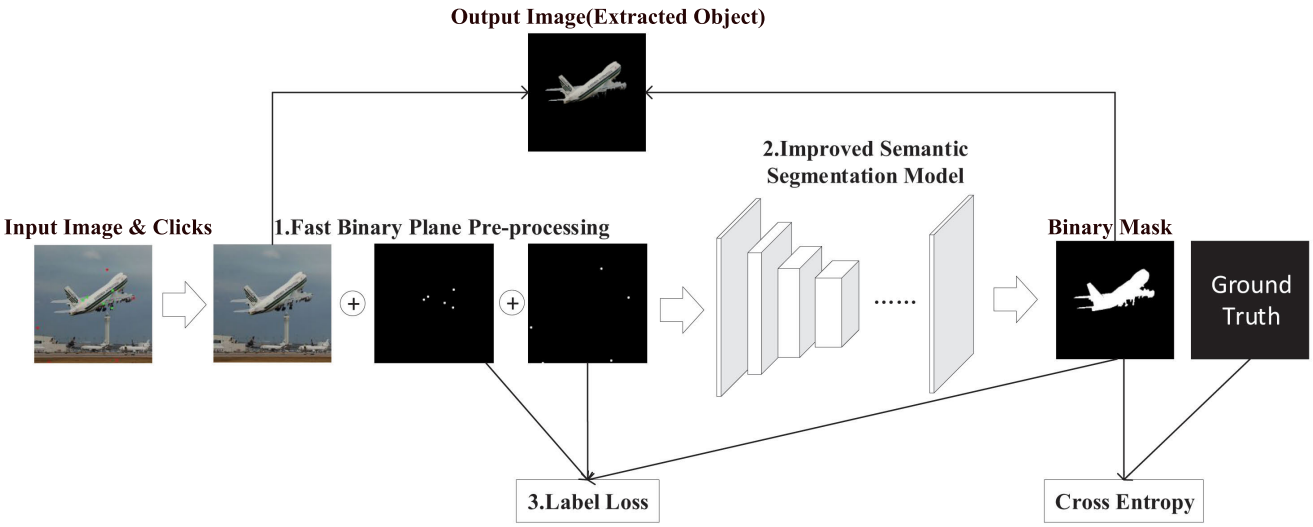


FIGURE 1. The framework of our model.

A. FAST BINARY PLANE PRE-PROCESSING

The following quads is a basic unit for training the model.

$$\{I, P, N, BM\} \tag{1}$$

where I is the original input image, P(Positive Plane) represents the foreground plane, N(Negative Plane) represents the background plane, and BM(Binary Mask) represents the mask of the object area, which is a binary plane and can be transformed from masks of the senmantic segmentation dataset. Since semantic segmentation is a multi-class task, a mask in a dataset typically contains multiple classes. In order to get precise class label as ground truth of each class from a mask, we separate all classes of a mask into multiple planes, which are BinaryMasks.

In order to get PositivePlane and NegativePlane, collecting manual labels is usually adopted. However, this method is not suitable for the task of this paper. On one hand, the cost is unbearable because it takes a lot of time to mark every image. On the other hand, this method is not adaptive. Once the dataset is changed, it needs to be marked again. So we take the method of simulating users' inputs. But due to the different habits of different users, we can't simulate interactive inputs of each person, so a random point selection method is necessary. By randomly simulating users' points as inputs, the foreground point set S^1 and the background point set S^0 are obtained, and the two planes are generated as follow.

$$P^t_{i,j} = \frac{sgn(r - min_{\forall k} d((i,j), S^t_k) + 1)}{2}, t \in \{0, 1\} \tag{2}$$

where $d(A, B) = \sqrt{(A_x - B_x)^2 + (A_y - B_y)^2}$ represents the Euclidean distance, sgn represents the sign function, S^t_k represents the kth point in the point set, and r represents the radius of the point. This method has two advantages. The first is that the conversion from point sets to planes is fast, because there is no need to calculate the values of all pixels, the model only needs to calculate the values of the points near the point set generated by users. The second is that models trained in this way are adaptive to trimap-style data directly.

Reference [11] sets restrictions to random points selection. we believe that these restrictions will lead to tendency of the model. There will be a deviation of extraction results when the interactive inputs violates these restrictions. The following experiment also proves this. In this paper, we eliminate these restrictions to achieve better generalization.

B. IMPROVED SEMANTIC SEGMENTATION MODEL

Typically, the number of channels in the output layer of a semantic segmentation network is equal to the number of semantic classes. In the semantic segmentation processing, using different datasets result in different numbers of output channels of the network. For example, the number of output channels of a network trained on the Pascal VOC dataset is 21, because there are 20 object classes and 1 background class. In the object extraction processing, we need to distinguish only two classes: foreground and background. Therefore, setting the number of channels to 2 in the output layer is necessary. This segmentation network predicts two probabilities for each pixel, which are the probabilities being a foreground point and a background point. When the current is larger than the latter, this pixel is regarded as a foreground point, otherwise a background point.

Nowadays, Deeplab v3+ is one of the best performing semantic segmentation models on the Pascal VOC 2012 dataset. Our work shows that it can be migrated to the object extraction task with slightly modification. In our model, we replace the backbone network with ResNet-101(other resnet variants achieve similar performance but increase complexity) on the basis of the original model. This mainly becasue ResNet is more generalized and adaptable to different tasks with the residual structure. Moreover, the object extraction task with interactive inputs is less complex than general semantic segmentation while Xception-65 is a network with a large parameter space and more likely to fall into local optimumis. So it is not suitable for the task in this paper. The following experiment also proves it.

The semantic segmentation task often takes 3-channel RGB images as inputs. And because of the integration of interactive inputs(Positive Plane and Negative Plane), the number of channels has increased to 5. Therefore the model is difficult to optimize with the pre-trained ResNet model. However, the work of [32] shows that it is not necessary although the pre-trained model can help the network converge faster in some degree. Even if the pre-trained model is not used, the same accuracy can be achieved. Our experiments also show that the precision of our model exceeds other methods with pre-trained models with only a few clicks.

C. LABEL LOSS

Semantic segmentation is a multi-class task, so it is reasonable to take cross entropy as the loss function. However, simply updating the parameters through cross entropy in object extraction process is quite slow because the network needs to understand the connection between the interactive inputs and the object regions. But it is easy to let the output be exactly the same as inputs (Positive Plane) or completely different(Negative Plane). This way is equivalent to learning the mapping.

$$f(x) = x \ or \ f(x) = \overline{x} \tag{3}$$

Therefore, our design comes from the idea that learning the local identity mapping and then spreading to the object area to achieve faster training speed.

In object extraction process, points with a value of 1 in the foreground plane must be predicted to be 1 by the network, and points with a value of 1 in the background plane must be predicted to be 0 by the network. This is because the two planes are the priori information given by users and the network output must be consistent with it.

All cases about network output and two planes are listed in Table 1. The loss function should be 0 in the correct and unknown cases, and maximum in the wrong cases. So that

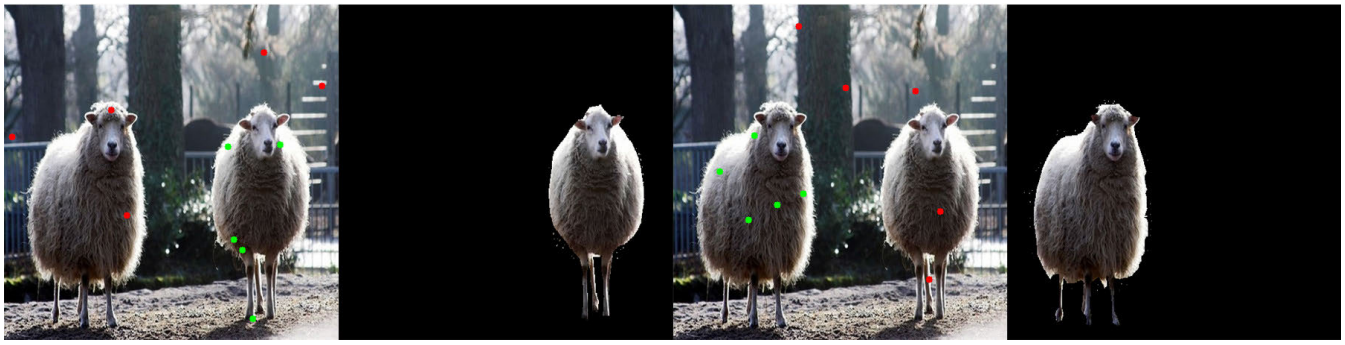


FIGURE 2. Our model extracts different objects based on interactive inputs.

TABLE 1. The relationship between interactive inputs and network output.

Plane	Value	Logit	Judge
Positive	0	0	UNK
Positive	0	1	UNK
Positive	1	1	√
Positive	1	0	×
Negative	0	0	UNK
Negative	0	1	UNK
Negative	1	1	×
Negative	1	0	√

it penalizes network output that is inconsistent with users' inputs. Moreover, the loss function should be mapped to the real number field. According to this design, our proposed label loss expression is as follow.

$$Loss = \sigma\{\frac{\sum[S^1(S^1 - L)^2 - S^0(S^0 - L)^2]}{W \times H}\} \tag{4}$$

where S^1 and S^0 represent the foreground plane and the background plane respectively. The square and multiplication are the operations of the corresponding element of the planes. W and H represent the width and height of the two-dimensional plane, respectively. And $\sigma(x) = \frac{1}{1+e^{-x}}$ is used to map the value to (0, 1). Our experiments below prove that Label Loss can significantly increase the converge speed of network training.

IV. EXPERIMENTS

The IOEBSS model proposed in this paper includes an efficient binary plane pre-processing, a high precision finetuning semantic segmentation model, and an auxiliary loss function that is more conducive to gradient flow. In order to show the role of these stages, we introduced the measurement experiments of accuracy, inference speed and training speed in 4.3.1, 4.3.2, and 4.3.3 respectively. The experiment about how the accuracy and the training time of the model are affected by backbones and Label Loss is introduced in 4.3.4.

A. DATASETS AND METRICS

The Pascal VOC 2012 dataset [33] is one of the popular datasets and benchmarks for semantic segmentation currently. It contains 17125 JPEG images, only 1464 semantic and instance segmentation labels are used as training sets, and 1449 semantic and instance segmentations labels are used as validation sets. There are 20 classes for semantic segmentation, numbered 1-20, and the background number is 0. Reference [34] increased the number of images of the dataset to 10,582, which is a commonly used training set for semantic segmentation models. Similarly, we used the augmented semantic segmentation dataset for training and an instance segmentation validation set for testing.

The COCO dataset is called MS COCO and is a generic image dataset proposed by Microsoft [35]. COCO2014 and COCO2017 are the two main versions used, the former with instance segmentation labels and the latter with material segmentation and panoramic segmentation, so the latter is not suitable for object extraction. Therefore, we tested the trained model with the COCO2014 validation set. Since the COCO dataset contains a total of 80 classes, 20 of which are identical to the 20 classes of Pascal VOC 2012, 60 are unique. Therefore, we tested according to the 20 seen classes of data during model training and the 60 unseen classes of data.

Our next experiment mainly used Pascal VOC 2012 as the measurement data for various indicators. For a better comparison with the previous state-of-the-art, we exploit the COCO dataset as an aid in measuring accuracy and mIoU.

To evaluate our model, we used a range of metrics. In terms of the accuracy of object extraction, we calculate Pixel Accuracy (PA) and mean Intersection over Union(mIoU) of the segmentation results. The latter, as our main evaluation criterion, is also adopted by most semantic segmentation researchers.

B. SETTINGS

In the stage of training the model, the Pascal VOC 2012 semantic segmentation augmented dataset is adopted and resized to the size of 513 x 513. After pre-processing, each RGB image can generally correspond to multiple masks with object areas. We simulate users' inputs completely randomly in the manner mentioned above, dynamically generating positive and negative planes during training. During training, we simulate users' inputs by creating 15 front-background

pairs of points for each image. We have found that too many clicks make it difficult for the network to recognize the connection between the front-background planes and the object mask, and too few clicks causes the network to focus only on the front-background planes, ignoring the information of the RGB image itself. In addition, we take the Adam optimizer, setting the size of the mini batch to 4, the learning rate to 0.0001, and the radius r of the front-background points to 5.

In the test and comparison stage, we first test the accuracy and mIoU of the model on the Pascal VOC 2012 and COCO2014 instance segmentation validation sets. The former exploits the entire dataset while the latter exploits 100 images in each class. In general, the training set and test set should be 1:1 in size. However, in practice, there is often a lack of accurately labeled datasets, so we still use one of the common datasets in semantic segmentation, Pascal VOC 2012 (20 classes), for training. At the same time, in order to verify the generalization ability of the model, we test the model on 20 seen classes and 60 unseen classes in COCO2014. High performance on unseen classes means strong generalization ability. Then we measure the inference time of a series of methods including our model on different resolution images. 100 images at each resolution are measured. The images used for these measurements are from the Pascal VOC 2012 dataset. Then we measure and compare the training time of the model according to the data provided in [11]. Finally, our experiments explore the effects of different semantic segmentation methods.

In order to verify the effectiveness and advancement of the proposed method, we choose a series of common methods in the following experiments, such as MagicWand, Poisson Matting, and GraphCut, as well as state of the art [11], which serves as the baseline.

C. RESULTS AND ANALYSIS

As the number of front-background points increases, the ''tips'' obtained by the model increase, and the mIoU of the extracted object area increases accordingly. In this case, we plot the mIoU as a function of the number of points. In order to compare with [11], and their work is not publicly implemented, we draw and compare based on the data they provide.

The result of Fig.3 shows that our model leads the algorithm and model including [11] on the Pascal VOC 2012 instance segmentation verification set. In the 20 seen classes of COCO2014, our model performance is comparable to [11]. And in the 60 unseen classes of COCO2014, our model surpasses other methods.

1) ACCURACY AND MIOU

Fig.4 shows the results of various methods including our model on Pascal VOC and COCO datasets. The leftmost column represents the RGB image and users' interactive inputs, green for the front points and red for the background points. Each of the other columns presents the output of a method. MagicWand tends to pick out areas with close colors. The GraphCut algorithm often has large areas of false segmentation, and Poisson Matting destroys the pixel distribution of the original image. Baseline here refers to the [11] model, we reproduce it according to the principle, and find that their model is more accurate than the traditional algorithms, but still in some areas with complex deformation, and the edge processing is not perfect. Our model is very close to Ground Truth, with subtle differences on only a few edges.

The mIoU performance of several object extraction methods on the Pascal VOC dataset is shown in Table 2. We test the mIoU achieved after 20 clicks, the number of clicks required to reach 85% mIoU, and the accuracy achieved after 10 clicks. Since MagicWand, Poisson Matting, and GraphCut still can't reach 85% after 20 clicks, we record it as 20+. It can be seen that in the extreme case our model is significantly ahead of other models, and achieving the same effect with fewer clicks.

TABLE 2. Comparison of other common object extraction methods.

Methods	mIoU 20 clicks	clicks 85% mIoU	accuracy
Baseline	91%	6.88	95.4%
MagicWand	63.6%	20+	82.4%
SLIC	55.1%	20+	77.1%
Watershed	64.4%	20+	63.9%
Active Contour	61.7%	20+	80.8%
Poisson Matting	57.9	20+	79.0%
GraphCut	73.2%	20+	88.3%
Ours	**94.1%**	**4.72**	**98.2%**

2) INFERENCE TIME

We measure the time required for different methods to inference images with the same size. As shown in Table 3 since we can't get the implementation of [11], we can only estimate the lower running limit. The time of their models is mainly spent on the generation of the front-background planes, as well as the GraphCut post-processing.

TABLE 3. Comparison of different methods of inference time.

Methods	Inference 256×256	Inference 512 × 512	Inference 1024 × 1024
Baseline	3.76s	12.55s	37.19s
MagicWand	0.152s	0.369s	0.608s
SLIC	0.085s	0.361s	1.45s
Watershed	**0.004s**	0.027s	0.114s
Active Contour	0.608s	0.620s	0.663s
Poisson Matting	0.12s	0.66s	3.28s
GraphCut	1.72s	7.73s	28.7s
Ours(CPU)	0.24s	0.86s	3.50s
Ours(GPU)	0.013s	**0.024s**	**0.072s**

Our model greatly simplifies pre-processing compared to other deep learning models, making slower CPU-based serial computations as less as possible, resulting in a much faster inference.

3) TRAINING TIME

According to the results of [11] and our experiments, Fig.5 shows the comparison of training time. We find that the

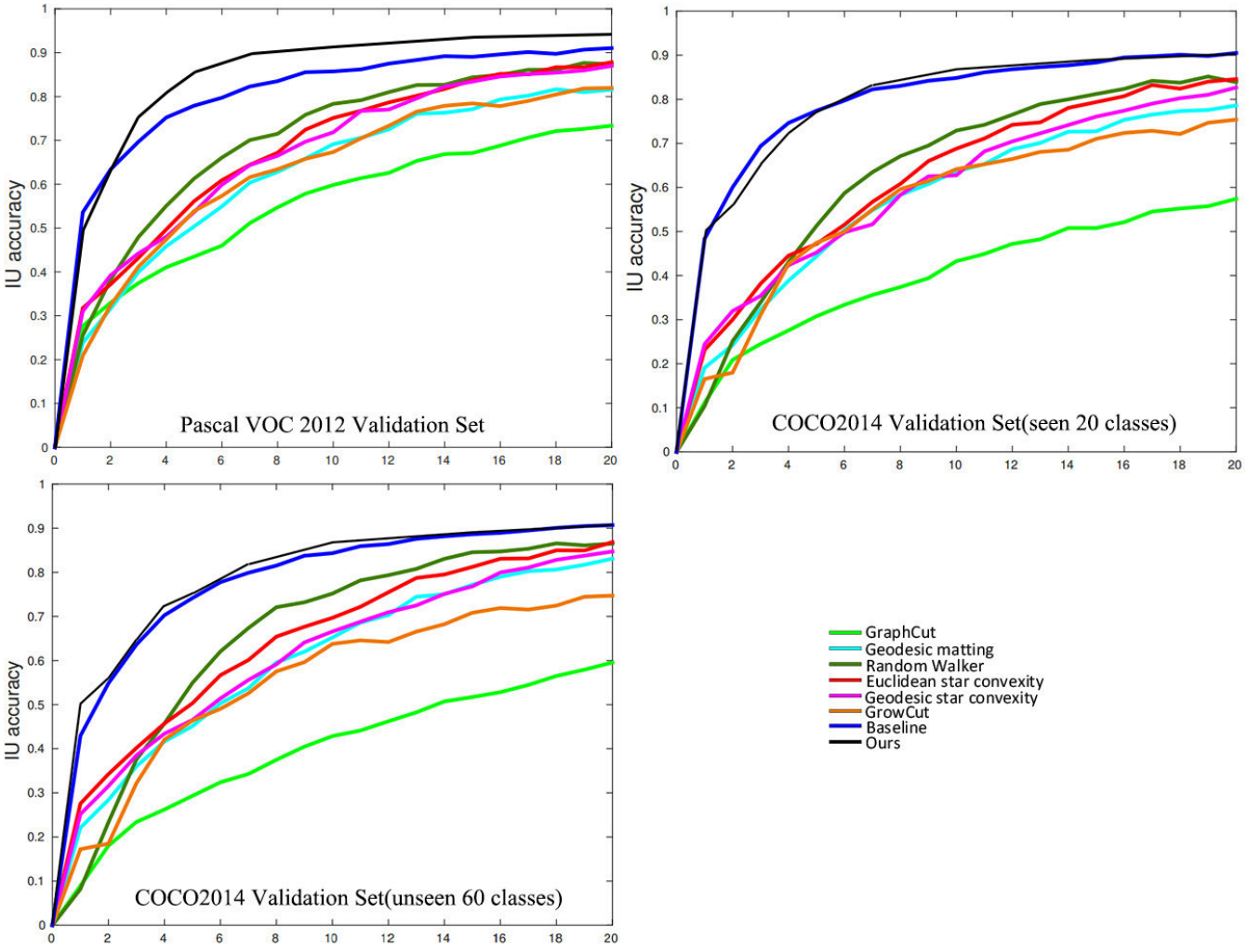


FIGURE 3. Performance of common methods and our model on COCO2014 and Pascal VOC 2012 instance segmentation verification set.

process of FCN model from 32s to 16s to 8s is very lengthy and unnecessary. It is efficient to train end-to-end systems directly. Our model can be trained in just 10 hours on a single GPU without any pretrained model.

In practical applications,users' experience is related to the speed of object extraction. In large scale commercial scenarios, too slow extraction speed will affect the commercial interests of the product, which is why many traditional object extraction algorithms are still used. Our model exhibits extremely high performance at three different resolutions of 256 × 256, 512 × 512, and 1024 × 1024 (Table 3). Even at a large resolution of 1024 × 1024, our algorithm has still about 14 fps. The most deadly problem with other methods is that they rely heavily on CPUs rather than GPUs. Since CPUs perform serial calculations, there is still a high latency even in the case of a small total computation. While [11] also takes GPUs for inference, its pre-processing needs to calculate the value of every pixel, and GraphCut post-processing further slows down the speed.

4) EFFECTS OF BACKBONES AND LABEL LOSS

In order to explore the impact of different segmentation models on object extraction accuracy, we try a variety of models. Table 4 shows that Label Loss can reduce the time to train the maximum mIoU to approximately half without affecting accuracy. Noting that in Deeplab v3+ (ResNet-101), the addition of label loss can even improve performance in the case of several clicks (4-10 clicks).

TABLE 4. Comparison of accuracy and iteration times of different backbones.

Methods	Label Loss	mIoU 20 clicks	clicks 85% mIoU	accuracy 10 clicks	iterations max mIoU
Baseline	No	91%	6.88	-	-
FCN-8s	No	84.9%	20+	96.0%	50k
FCN-8s	Yes	85.0%	20+	96.0%	**30k**
Deeplab v3+(X-65)	No	91.8%	5.91	97.5%	175k
Deeplab v3+(X-65)	Yes	91.8%	5.95	97.6%	**80k**
Deeplabv3+(R-101)	No	**94.1%**	4.89	98.1%	180k
Deeplabv3+(R-101)	Yes	**94.1%**	**4.72**	**98.2%**	**95k**

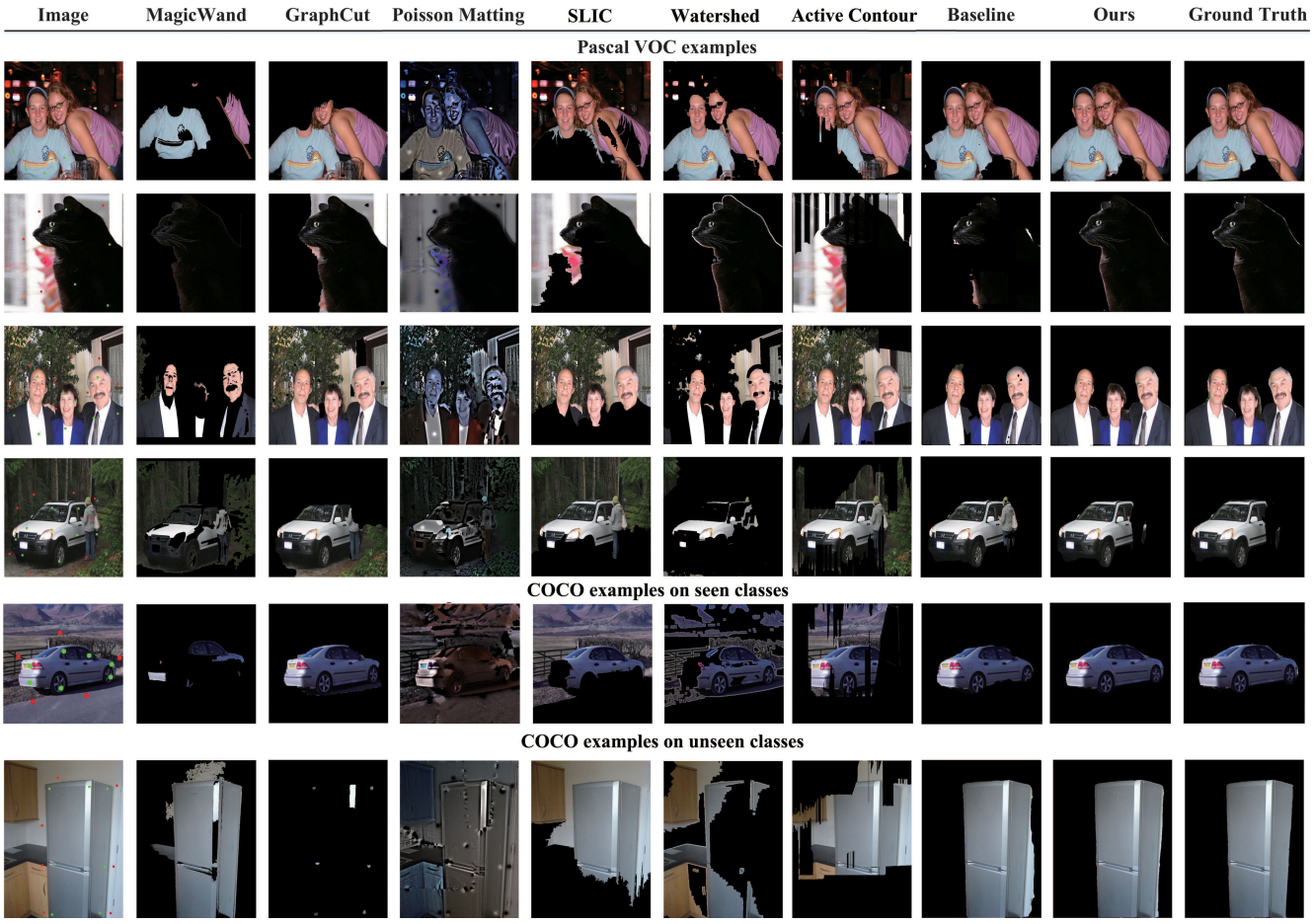


FIGURE 4. Results of object extraction given by different methods.

5) ANALYSIS ON STRENGTHS AND WEAKNESSES

The advantages of our proposed method are that previous object extraction methods can achieve faster inference speed, higher accuracy, and less training time. For traditional methods, the main advantage of our proposed method is that it can identify regions with semantic features. This is because our method is based on data learning, and semantic knowledge is contained in labeled data, while traditional methods are not. For deep learning-based methods, the main advantages of our proposed method are that it reduces unnecessary post-processing, reduces the complexity of preprocessing, and also improves the training speed of the model.

As far as the baseline is concerned, they used a gray value of 0-255 in the preprocessing to represent the area clicked by the user, which will cause a large number of traversals and greatly reduce the inference speed. Our method only performs a binary representation according to the user's click, and only needs to calculate the points near the clicked position, thus reducing the time consumption. In addition, they performed GraphCuts on the output of the semantic segmentation model, which will greatly increase the time overhead, and our method does not need to use post-processing.

Finally, our proposed label loss is a new loss function for interactive tasks. It turns out that most methods based on deep learning only use cross entropy, L1, L2 loss, and do not directly use the prior information given by the user to calculate the loss. This requires the model to take a long time to understand the prior information. Relationship with object area. And our label loss achieves fast model training by directly minimizing the difference between the model output and these prior information.

The disadvantage of our proposed method is that, because the semantic segmentation model attaches importance to the understanding of the overall semantics of the image, the ability to extract regions with weak semantics is insufficient. For example, it is difficult for IOEBSS to extract some strange subsea areas, because these areas may have weak semantics. Since there are similar problems in semantic segmentation (for example, there is only one chair leg, most existing methods are difficult to segment correctly), we believe that this is a deficiency of the semantic segmentation model itself. However, this deficiency may be gradually resolved with the development of semantic segmentation research. At this point, if these regions have high local similarities (brightness, texture, etc.), traditional methods that rely solely on pixel

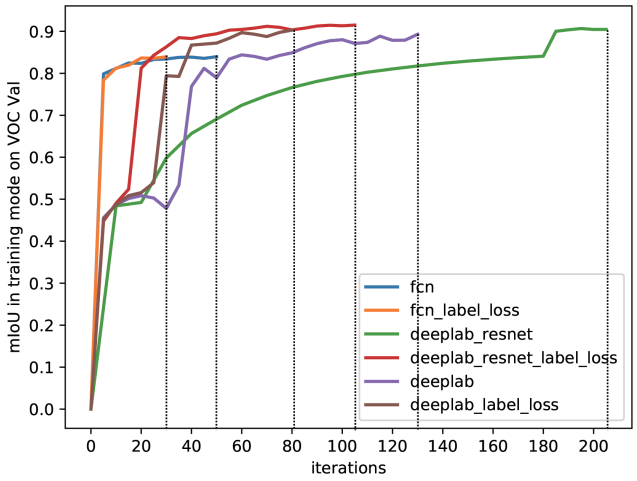


FIGURE 5. Training time spent by different methods.

features may perform slightly better. However, once these areas with weak semantics show complex local features, the traditional method often loses effect.

V. CONCLUSIONS AND FUTURE WORK

The traditional object extraction algorithms have low accuracy, and they are difficult to identify the object regions with large difference in low-level pixel features and strong semantic relevance. Methods based semantic segmentation add more pre-processing and post-processing, which leads to slow training and inference. In this paper, we propose a new object extraction model to deal with these problems. With the help of the fast binary plane pre-processing, a stronger semantic segmentation network, and Label Loss, our model achieves higher accuracy, faster model convergence speed and inference speed on Pascal VOC and COCO datasets.

In the future, we will explore whether high precision object extraction can be used to improve the precision of semantic segmentation. Simulating interactive inputs to optimize the results of semantic segmentation may be a feasible approach. In addition, we hope to find other applications and extensions of Label Loss in interactive tasks.

Besises, using a stronger semantic segmentation model is definitely a way to improve accuracy, but it cannot fundamentally solve the problem of extracting regions with weak semantics. This problem may need to start with the amount of data. However, the data for semantic segmentation is scarce and expensive. We recommend using unsupervised semantic segmentation and training a large number of image data to achieve a more robust object extraction method.

REFERENCES

[1] R. Adams and L. Bischof, "Seeded region growing," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 6, pp. 641–647, Jun. 1994.

[2] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Trans. Graph.*, vol. 22, no. 3, pp. 313–318, Jul. 2003.

[3] Y. Y. Boykov and M.-P. Jolly, "Interactive graph cuts for optimal boundary & region segmentation of objects in N-D images," in *Proc. 8th IEEE Int. Conf. Comput. Vis. (ICCV)*, vol. 1, Jul. 2001, pp. 105–112.

[4] S. Lutz, K. Amplianitis, and A. Smolic, "AlphaGAN: Generative adversarial networks for natural image matting," in *Proc. BMVC*. London, U.K.: BMVA Press, 2018, p. 259. [Online]. Available: http://dblp.uni-trier.de/db/conf/bmvc/bmvc2018.html#LutzAS18

[5] N. Xu, B. Price, S. Cohen, and T. Huang, "Deep image matting," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2970–2979.

[6] G. Wang, M. A. Zuluaga, W. Li, R. Pratt, P. A. Patel, M. Aertsen, T. Doel, A. L. David, J. Deprest, S. Ourselin, and T. Vercauteren, "DeepIGeoS: A deep interactive geodesic framework for medical image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 7, pp. 1559–1572, Jul. 2019.

[7] G. Wang, W. Li, M. A. Zuluaga, R. Pratt, P. A. Patel, M. Aertsen, T. Doel, A. L. David, J. Deprest, S. Ourselin, and T. Vercauteren, "Interactive medical image segmentation using deep learning with image-specific fine tuning," *IEEE Trans. Med. Imag.*, vol. 37, no. 7, pp. 1562–1573, Jul. 2018.

[8] D. Cho, Y.-W. Tai, and I. Kweon, "Natural image matting using deep convolutional neural networks," in *Computer Vision—ECCV 2016*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham, Switzerland: Springer, 2016, pp. 626–643.

[9] D. Mirkovic and L. Johnsson, "Automatic performance tuning in the UHFFT library," *Comput. Graph. Forum*, vol. 35, no. 2, pp. 93–102, 2001.

[10] Y. Zheng and C. Kambhamettu, "Learning based digital matting," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 889–896.

[11] N. Xu, B. Price, S. Cohen, J. Yang, and T. Huang, "Deep interactive object selection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 373–381.

[12] Y.-Y. Chuang, B. Curless, D. H. Salesin, and R. Szeliski, "A Bayesian approach to digital matting," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Dec. 2001, p. 2.

[13] A. Levin, D. Lischinski, and Y. Weiss, "A closed-form solution to natural image matting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 2, pp. 228–242, Feb. 2008.

[14] Q. Chen, D. Li, and C.-K. Tang, "KNN matting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 9, pp. 2175–2188, Sep. 2013.

[15] D. Stutz, A. Hermans, and B. Leibe, "Superpixels: An evaluation of the state-of-the-art," *Comput. Vis. Image Understand.*, vol. 166, pp. 1–27, Jan. 2018. [Online]. Available: http://dblp.uni-trier.de/db/journals/cviu/cviu166.html

[16] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk, "SLIC superpixels compared to State-of-the-Art superpixel methods," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2274–2282, Nov. 2012. [Online]. Available: http://dblp.uni-trier.de/db/journals/pami/pami34.html

[17] R. Gaetano, G. Masi, G. Poggi, L. Verdoliva, and G. Scarpa, "Marker-controlled watershed-based segmentation of multiresolution remote sensing images," *IEEE Trans. Geosci. Remote Sens.*, vol. 53, no. 6, pp. 2987–3004, Jun. 2015. [Online]. Available: http://dblp.uni-trier.de/db/journals/tgrs/tgrs53.html

[18] J. Cousty, G. Bertrand, L. Najman, and M. Couprie, "Watershed cuts: Thinnings, shortest path forests, and topological watersheds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 925–939, May 2010. [Online]. Available: http://dblp.uni-trier.de/db/journals/pami/pami32.html

[19] M. Ciecholewski, "Malignant and benign mass segmentation in mammograms using active contour methods," *Symmetry*, vol. 9, no. 11, p. 277, Nov. 2017. [Online]. Available: http://dblp.uni-trier.de/db/journals/symmetry/symmetry9.html

[20] K. Ding, L. Xiao, and G. Weng, "Active contours driven by region-scalable fitting and optimized Laplacian of Gaussian energy for image segmentation," *Signal Process.*, vol. 134, pp. 224–233, May 2017. [Online]. Available: http://dblp.uni-trier.de/db/journals/sigpro/sigpro134.html

[21] N. Xu, B. L. Price, S. Cohen, J. Yang, and T. S. Huang, "Deep GrabCut for object selection," in *Proc. BMVC*. London, U.K.: BMVA Press, 2017. [Online]. Available: http://dblp.uni-trier.de/db/conf/bmvc/bmvc2017.html#XuPCYH17

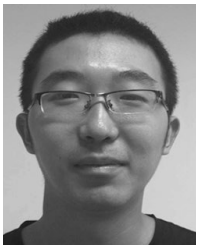
[22] Y. Li, J. Sun, C. Tang, and H. Shum, "Lazy snapping," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 303–308, Aug. 2004. [Online]. Available: http://doi.acm.org/10.1145/1015706.1015719

[23] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, Aug. 2000.

[24] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 3431–3440.
[25] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Apr. 2018.
[26] F. Yu and V. Koltun, "Multi-scale context aggregation by dilated convolutions," in *Proc. ICLR (Poster)*, Y. Bengio and Y. LeCun, Eds. 2016. [Online]. Available: http://dblp.uni-trier.de/db/conf/iclr/iclr2016.html#YuK15
[27] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille, "Semantic image segmentation with deep convolutional nets and fully connected CRFs," in *Proc. Int. Conf. Learn. Represent. (Poster)*, Y. Bengio and Y. LeCun, Eds. 2015.
[28] 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 *Proc. ICCV*. IEEE Computer Society, 2015, pp. 1529–1537. [Online]. Available: http://dblp.uni-trier.de/db/conf/iccv/iccv2015.html#0001JRVSDHT15
[29] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 2881–2890.
[30] L.-C. Chen, M. Collins, Y. Zhu, G. Papandreou, B. Zoph, F. Schroff, H. Adam, and J. Shlens, "Searching for efficient multi-scale architectures for dense image prediction," in *Proc. Adv. Neural Inf. Process. Syst.*, S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, Eds. Red Hook, NY, USA: Curran Associates, 2018, pp. 8699–8710. [Online]. Available: http://papers.nips.cc/paper/8087-searching-for-efficient-multi-scale-architectures-for-dense-image-prediction.pdf
[31] B. Zoph and Q. V. Le, "Neural architecture search with reinforcement learning," in *Proc. ICLR*. OpenReview.net, 2017. [Online]. Available: http://dblp.uni-trier.de/db/conf/iclr/iclr2017.html#ZophL17
[32] K. He, R. B. Girshick, and P. Dollár, "Rethinking ImageNet pre-training," in *Proc. ICCV*, 2019, pp. 4917–4926. [Online]. Available: http://dblp.uni-trier.de/db/conf/iccv/iccv2019.html#HeGD19
[33] M. Everingham, A. Zisserman, C. K. I. Williams, L. V. Gool, M. Allan, C. M. Bishop, O. Chapelle, N. Dalal, T. Deselaers, and G. Dorkó, "The 2005 PASCAL visual object classes challenge," *Lect. Notes Comput. Sci.*, vol. 111, no. 1, pp. 98–136, 2007.
[34] B. Hariharan, P. Arbelaez, L. Bourdev, S. Maji, and J. Malik, "Semantic contours from inverse detectors," in *Proc. Int. Conf. Comput. Vis.*, Nov. 2011, pp. 991–998.
[35] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft COCO: Common objects in context," in *Computer Vision—ECCV 2014*, D. Fleet, T. Pajdla, B. Schiele, and T. Tuytelaars, Eds. Cham, Switzerland: Springer, 2014, pp. 740–755.

XIAORU WANG received the M.S. and Ph.D. degrees in computer science and technology from the Beijing University of Posts and Telecommunications, in 2001 and 2015, respectively.

She is currently an Associate Professor and a Ph.D. Tutor with the School of Computer Science and Technology, Beijing University of Posts and Telecommunications. She is also the Director of the Big Data Center, Beijing University of Posts and Telecommunications University. Her research interests include image processing and understanding, computer vision, and pattern recognition.



PEIRONG XU was born in Shenyang, Liaoning, China, in 1996. He received the B.S. degree in information security from the Beijing University of Posts and Telecommunications, Beijing, China, in 2019, where he is currently pursuing the M.S. degree.

He has published an article named *Feedback LSTM Network Based on Attention for Image Description Generator*. His research interests include semantic segmentation, image object extraction, and image caption.

ZHIHONG YU received the B.S. and Ph.D. degrees in information engineering from the Beijing University of Posts and Telecommunications.

He is currently as a Solution Architect in Intel China Research Center. His research interests include visual cloud, heterogenous computing, accelerators, and so on.

FU LI (Senior Member, IEEE) received the B.S. and M.S. degrees in physics from Sichuan University, China, in 1982 and 1985, respectively, and the Ph.D. degree in electrical engineering from the University of Rhode Island, in 1990.

Since 1990, he has been with Portland State University, where he is currently a Full Professor of electrical and computer engineering. His research interests include signal, image, and video processing, as well as wireless, networks, and multimedia communications.