Go-selfies: A Fast Selfies Background Removal Method Using ResU-Net Deep Learning

Yunan Wu
Department of Biomedical Engineering
Northwestern University
Evanston, IL, U.S.
yunanwu2020@u.northwestern.edu

***Abstract*—The selfies play an important role in recording meaningful moment in human's daily life. In most cases, before sharing photos, people often synthesis attractive images on some phone applications, such as Photoshop. While these kinds of software have reached good performance nowadays, they are too complex for simple life usage. In this work, we proposed an automatic segmentation model unique to segment human selfies photos. We first constructed a large photo segmentation database and built 8 different models based on resolution, image size and whether or not to use transfer learning and picked the best one among them. We then applied cyclical learning rate method and pre-trained encoder network to fine tune our models. Finally, our best model tested on Google images demonstrated satisfying promising results on both accuracy scores and losses, which will be the precondition in real-time segmentation. We named this lovely web product as "Go Selfies".**

Keywords—background removal, deep learning, selfies, ResU-Net.

I. INTRODUCTION

With the development of the virtual communication, more people get involved in sharing photos on public platform. For example, Facebook, the broadest and fastest growing social platform, has about two billion photos being shared everyday [1]. Although people try to show their lifeway by sharing the photos, most of the photos are "fake". Before posting the photos, people often do image retouching, which relies on Photoshop or other apps to make their ideal photos. For example, in Fig. 1, the girl seems to jump across the cliff, but the truth turns out that she is just jumping on the grass. The thing is that software is often too professional to operate, especially for children and the olds. For example, think about these situations. You need ID photo with blue background but you only have the red one, or you want to keep privacy, i.e. hiding the background, when video chatting with strangers. Therefore, in simple terms, all you need is a real-time background removal platform.

The value of image segmentation cannot be ignored since it appears in every aspect of daily life, such as medical detection [2, 3], automatic driving [4] and video segmentation [5, 6]. However, today's segmentation technique is not mature enough to satisfy all these expectations because each image is specific. Threshold segmentation, like mean shift technique [7], edge maximization technique (EMT) [8] were first proposed based on image intensity, but it required large time and computation consumption, which was unrealistic for real-time applications. The emergence of the convolutional neural networks (CNNs) makes things easier nowadays. Features are not extracted manually anymore but trained and updated by networks. However, only few attempts [9, 10] have made to photo segmentation mainly because there is no specific segmentation dataset aimed at photos and the problem of CNNs architectures losing spatial information is not fit for segmentation task.

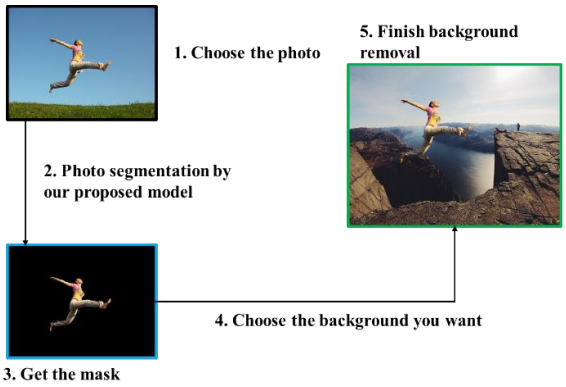

Fig. 1. The framework of our method.
Given one image, our method can extract the figure and get the mask. Then choose whatever background and finish the background removal.

In our work, we first created a huge selfies database with manual pixel labels and then proposed a robust method for photo segmentation by combining deep encoder networks with conventional U-Net. We trained 8 models in total to compare the results of different encoder networks, input resolutions and having pre-trained operation or not on final results. Our proposed method focused not only on the portraits, but also on human photos in different angles and sizes. Furthermore, it was tested to have good results in daily life photos, Google images and outperformed state-of-the-art methods. Finally, our method has promising prospects in video segmentation and daily applications in the future. Fig. 1 shows the overflow of our basic idea.

II. RELATED WORKS

Over the years, many methods have given efforts in the field of image segmentation. In this section, we discuss the recent related work on image segmentation and instance segmentation.

A. Image Segmentation

A lot of studies have been conducted on image segmentation based on deep CNNs. From the very beginning, CNNs are used for image classification. Krizhevsky et al. [11] built up the famous AlexNet and won the ImageNet Large-Scale Visual Recognition Challenge (ILSVRC). Inspired by AlexNet, Simonyan et al. [12] proposed a deeper model and achieved better performance. However, with the trend of neural network going deeper and deeper, the training process also became more difficult. He et al. [13] proposed residual learning framework to ease training computations of network.

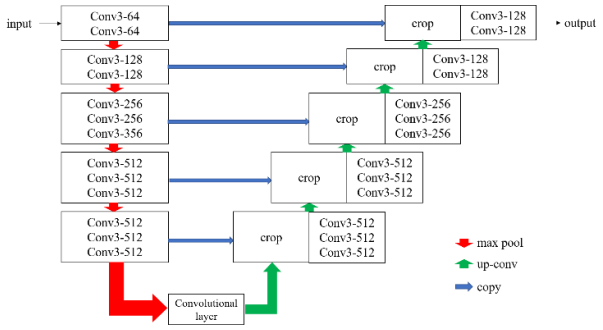

Fig.2. The model architecture
It is divided into two parts, the encoder part (red arrow) and decoder part (green arrow).

Long et al. [14] introduced fully convolutional networks which allowed the models to be trained end-to-end and pixels-to-pixels for image segmentation. Following this framework, Badrinarayanan et al. [15] proposed SegNet that used pooling indices in the decoder part of their model. Ronneberger et al. [16] presented a "U" shape network, especially for biomedical images that had limited data size, so it is called "U-net".

B. Instance Segmentation

Based on the concept of CNNs, Girshick *et al*. [17] presented regions with CNN features (R-CNN) for accurate object detection. Later on, Girshick [18] extended his previous work and created fast R-CNN by using RoIPool. Ren *et al*. [19] advanced fast R-CNN and presented faster R-CNN by learning the attention mechanism with a Region Proposal Network. He *et al*. [20] created Mask R-CNN which detected objects while generated a high-quality segmentation mask simultaneously for single instance. DeepMask, proposed by Pinheiro *et al*. [21], learned to propose segment candidates and then were classified by Fast R-CNN. Likewise, Dai *et al*. [22] proposed a complex multiple-stage cascade that predicted segment proposals from bounding-box proposals, followed by classification.

III. METHODOLOGY

In this section, we mainly introduce our method in selfies segmentation. First, we show the model architecture based on encoder and decoder part. We then introduce the useful method in finding the best parameters and fine tuning. Last, we apply our evaluation metrics while training the model.

A. Model Architecture

Rather than simply using U-Net, two different classification models, VGG16 and ResNet34 are combined with it as the encoder parts. The decoder part consists of upsampling feature maps from the last layers and corresponding feature maps from corresponding decoder parts. Two parts together make up a similar U-Net model, called ResU-Net and VGU-Net, as shown in Fig. 2.

For the encoder part, simply by changing the last fully connected layers into convolutional layers, the feature maps can contain local information from the raw images. The two types of encoder models have some differences. VGG16 contains 13 convolutional layers with pooling layers inserted after each block to decrease the spatial dimensions. ResNet34 has unique skip connections between different layers, making networks deeper without enlarging parameters, as shown in Fig. 3.

The decoder part performs like a symmetric expanding path with the encoder part, which gets precise localization information. At the end of each unit, the channel size is half reduced while the feature map dimensions are twice increased. A 1×1 convolutional layer appends to the last layer, followed by a sigmoid function, which computes the probability of each pixel in that image location. Final output mask is obtained based on each pixel class.

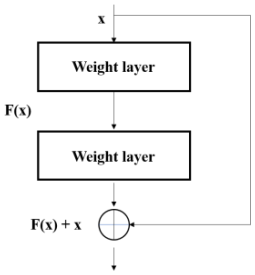

Fig.3. ResNet building block. Short skip connections are used。

B. Cyclical Learning Rate

A useful method, called Cyclical Learning, was proposed by Leslie [23] to find the optimal learning rate easily without the need to repeatedly change the parameters. First, we do learning rate range test to get the range of both minimal and maximal learning rate. As the model runs, we increase the learning rate linearly between random picked low and high learning rate values and plot the loss changes versus different learning rates. As shown Fig. 4, two points should be paid attention to, in which the left red dot presents the position where the loss begins to fall while the right black dot is the place the loss begins to increase. This is the training process result from the model VGU-Net16 with images resolution 128 ×128. 10^{-3} and 10^{-1} are set to be the minimal and maximal of the learning rate. The optimal learning rate is suggested to appear 1/3 or 1/4 of the maximum learning rate [23], i.e. 0.2 $\times 10^{-1}$ in our work.

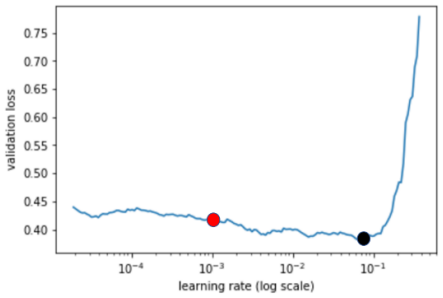

Fig. 4. Optimal learning rate. Left red dot presents the loss begins to fall and the right dark refers the loss begins to increase.

Next, instead of using a fixed or exponentially decreasing learning rate, we apply cyclical learning rate that changes values in different stages of the model. For example, in Fig. 5, we change the learning rate in triangular wave-like format. This is of great importance that (1) improves the model performance with fewer epochs since it updates faster in first convolutional layers than later ones; (2) helps the model get out of any local saddle points, which was proved by Dauphin

et al. [24] that decreasing learning rate would never work when facing local saddle point problems.

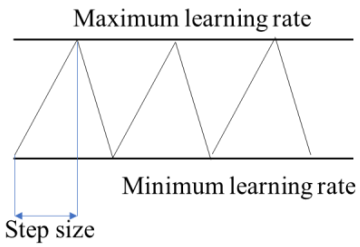

Fig. 5. Using triangular wave in cyclical learning rates. The learning rate changes between the maximum and minimum during each epoch.

C. Fine Tuning

Transfer learning is a simple way to build robust model and promotes fast convergence. Following the idea of transfer learning, we apply the weights that pre-trained on ImageNet dataset to some layers of our model, instead of training the whole model from scratch. ImageNet is a large visual database aiming for the use in visual recognition. Using the pre-trained weights would show excellent result especially when our dataset is small but similar compared to the ImageNet dataset. Moreover, it helps to alleviate the problem of overfitting. Our method freezes the convolutional layers by using the pre-trained weights and only trains the last layers of encoder work with our new dataset. The big influence of pre-trained models on the final results would be illustrated in the results.

D. Evaluation Metrics

Dice coefficient: given two sets, i.e., predicted mask image and ground truth mask image in this paper, dice is defined in (1), where |X| and |Y| refers to the number of pixels in the foreground on these two sets respectively. Dice, rather than intersection of union (IoU) is chosen for the consideration of minor computations. The Dice closer to 1, the better the performance.

$$Dice = \frac{2|X \cap Y|}{|X| + |Y|} \quad (1)$$

Binary cross entropy loss: loss function is defined as in (2). This is a binary classification problem, where y_p is the prediction and y_t is the label of that pixel.

$$loss = (-y_t log(y_p) - (1 - y_t) log(1 - y_p)) \quad (2)$$

Confusion Matrix: a table describes the performance of classification on validation dataset, where accuracy, sensitivity, specificity can be calculated. This paper focuses more on accuracy, which defined as in (3), where N is the number of images.

$$Accuracy = (\sum_N \frac{\text{Correct predicted pixels}}{\text{All pixels in each image}}) / N \quad (3)$$

IV. EXPERIMENT

As there is no photo segmentation dataset, we create a new dataset labeled by ourselves using in training and validation. In this section, we first show the process of data preparation and training details. We then compare it with other state-of-the-art methods in photo segmentation.

A. Data Collection and Preprocessing

We collect in total 2000 images from three datasets, 1000 in COCOA dataset, 700 in Pascal VOC dataset and 300 manually labeled data.

COCO and Pascal VOC data: COCO dataset includes 80000 images with more than 90 classes and Pascal VOC dataset includes 11000 images in 20 classes. We do selections in these datasets, keeping images with person labels. Then, drop the images that contain too many humans, keeping just 1 or 2 people in one image. Finally, make sure the foreground covers at least 30% of the image area. The foreground focuses mainly on person and sometimes includes the things people hold in hand, such as snowboard and pizza. Finally, the background is labeled as black and foreground as white in the masks. The image pick-up standardization shows in Fig. 6.

Manually labeled data: To test the robustness of the approach, we evaluate the model on different dataset. We search 500 images from Google and label them with Matlab Image labeler. Different types of photos are selected, including the front view images, the rear view images, half-body photos and full-body photos. The foreground also includes the person and the things with them.

1700 images are split to be the training set, leaving the rest 300 data to be the separate testing set. Then we do 5-fold cross-validation on the training set. In order to alleviate the

TABLE I. EIGHT MODEL COMPARISONS. THE BEST ARE IN BOLD

Model	Training loss	Testing loss	Testing Accuracy	Testing Dice
Vgg16-128	0.2281	0.2683	0.8862	0.7759
Vgg16-128-pretrain	0.1744	0.2324	0.8936	0.7814
Vgg16-512	0.1461	0.2211	0.9000	0.7855
Vgg16-512-pretrain	0.1302	0.2073	0.9213	0.8468
Resnet34-128	0.2208	0.2559	0.8880	0.7787
Resnet34-128-pretrain	0.1004	0.1826	0.9346	0.8795
Resnet34-512	0.1375	0.2142	0.9200	0.8449
Resnet34-512-pretrain	**0.0888**	**0.1406**	**0.9442**	**0.8977**

(a) (b) (c)

Fig. 6. The standard of chosen images. (a) is the good image chosen in dataset; (b) and (c) are not chosen because (b) has so many people and (c) is no intact person.

TABLE II. COMPARISON WITH STARE-OF-THE-ART

Methods	Dice
FCN (Person Class) [25]	0.7309
Graph-cut [26]	0.8002
Portrait FCN [10]	0.8972
Resnet34-512-pretrain	**0.8977**

overfitting problem, the training set rather than the validation or the testing set is augmented with three transformations, rotation, flipping and lightening. The training set adds four new rotations in -60°, -30°, 30°, 60° and flips horizontally and vertically. The intensity of the image changes from dark to light in four variations: 0.2, 0.4, 0.6, 0.8. To test the influence of different resolution, we train the model in two resolutions, 128 × 128 and 512 ×512.

B. Implementation Details

8 models were trained in total based on two resolutions and two different encoder networks, VGG16 and ResNet34. Pre-training or not is another comparison in this paper. To train our proposed model, we use Adam optimizer and choose 2×10^{-1} to be the first learning rate as described in section 3.2. Batch size is set to be 64 in small resolution 128 and 16 in large resolution 512. The training epoch is 50 in all models. The training and testing process are conducted in Python 3.6 Pytorch with a single GPU, NVIDIA Tesla K80 rent on the platform AWS.

C. Result

In this section, we first compare results in validation set among eight models. We then provide the result on test set using our best proposed model. Finally, we compare our model with other state-of-the-art methods in photo segmentation.

We trained 8 models in total, of which the names are in the format of "encoder model-resolution-pretrained or not". For example, Resnet34-128-pretrain refers to the U-net model using the encoder Resnet34, the input resolution 128 × 128 and pretrained on ImageNet dataset, while Vgg16-512 means the model using the encoder Vgg16, the input resolution 512 × 512 but without pretrained weights. Detailed results are shown in Table 1.

From the experimental results, we report the Dice, the global accuracy and the loss. First, ResNet34 performs better than VGG16. The later one has overfitting problem at the end of the training time partly because VGG16 contains more parameters than ResNet34. Another reason is due to the specific structure of ResNet34, so that it covers more detailed edge information and the boundaries of result become smoother. Next, larger resolution input reaches better results since they have more specific features, but with more time. Pre-training operation has a big impact on the final results, guaranteeing higher accuracy and faster convergence. Therefore, we choose the model of ResNet34-512-pretrain as the best model to do the test.

The test images are chosen randomly from the Google images, which look to be common photos in the daily life. No size limits since the models contain all convolutional layers. As shown in Fig. 7, different types of images are segmented, which proves the model's good generalization.

Finally, as shown in Table 2, we compare the performance of our model with other portrait segmentation methods. Our results outperform the state-of-the-art in this segmentation. Note that, other methods focus merely on portraits, which requires explicit faces and half-body photo, but our method has no limitations. As shown in Fig. 7. We can segment human in any views, which promises more practical uses in daily life.

V. CONCLUSION

In this paper, we proposed a deep learning method in photo segmentation for daily life usage. We first created a big photo segmentation dataset to train and evaluate our model, ResU-Net. It combines the encoder work with decoder work based on basic U-Net architecture. The cyclical learning rate promises fast convergence and higher accuracy. We also verified the importance of the pre-trained encoder work in photo segmentation. Finally, we tested out model in Google image and get satisfying results. In the future, since the ground-truth used in our model lack the details of edge information, we would apply GAN to better improve our models. The code is available via this GitHub link: https://github.com/YunanWu2168/Background-removal-using-deep-learning

REFERENCES

[1] K. Harrison and V. Hefner, "Virtually Perfect: Image Retouching and Adolescent Body Image," *Media Psychol.*, vol. 17, no. 2, pp. 134–153, Apr. 2014.

[2] M. Hameed, M. Sharif, M. Raza, S. W. Haider, and M. Iqbal, "Framework for the Comparison of Classifiers for Medical Image Segmentation with Transform and Moment based features," vol. 2, p. 11, 2013.

[3] Z. Ma, J. M. R. S. Tavares, R. N. Jorge, and T. Mascarenhas, "A review of algorithms for medical image segmentation and their applications to the femalez pelvic cavity," Comput. Methods Biomech. Biomed. Engin., vol. 13, no. 2, pp. 235–246, Apr. 2010.

[4] H. Oliveira and P. L. Correia, "AUTOMATIC ROAD CRACK SEGMENTATION USING ENTROPY AND IMAGE DYNAMIC THRESHOLDING," p. 5.

[5] M. Grundmann, V. Kwatra, M. Han, and I. Essa, "Efficient hierarchical graph-based video segmentation," in 2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2010, pp. 2141–2148.

[6] C. Xu, C. Xiong, and J. J. Corso, "Streaming Hierarchical Video Segmentation," in Computer Vision – ECCV 2012, vol. 7577, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 626–639.

[7] J. Wang, B. Thiesson, Y. Xu, and M. Cohen, "Image and Video Segmentation by Anisotropic Kernel Mean Shift," in Computer Vision - ECCV 2004, vol. 3022, T. Pajdla and J. Matas, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 238–249.

[8] Image Segmentation by Using Edge Detection. .

[9] J. P. Gozali, M. Kan, and H. Sundaram, "Hidden Markov Model for Event Photo Stream Segmentation," in 2012 IEEE International Conference on Multimedia and Expo Workshops, 2012, pp. 25–30.

[10] X. Shen et al., "Automatic Portrait Segmentation for Image Stylization," Comput. Graph. Forum, vol. 35, no. 2, pp. 93–102, May 2016.

[11] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in Advances in Neural Information Processing Systems 25, F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Curran Associates, Inc., 2012, pp. 1097–1105.z

[12] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," ArXiv14091556 Cs, Sep. 2014.

[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in 2016 IEEE Conference on Computer Vision

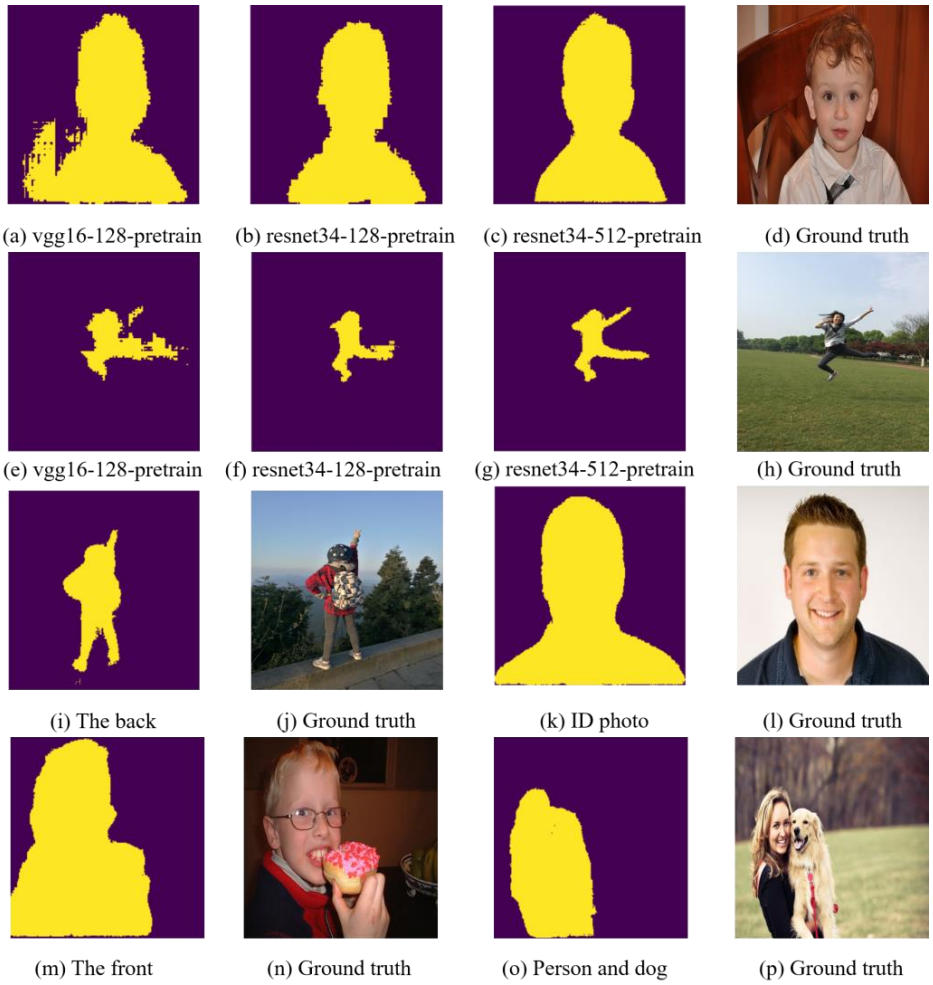

Fig. 7. The results tested on the best model. (a-d) are the results from the validation data, which have better performance from left to right. (e-h) are the results from the test data. (i) and (j) are the test results of back figure. (k) and (l) are the results of ID photo. (m) and (n) are the results of the front photo in daily life. (o) and (p) are results with other things in the foreground. All images are from Google Image.

and Pattern Recognition (CVPR), Las Vegas, NV, USA, 2016, pp. 770–778.

[14] J. Long, E. Shelhamer, and T. Darrell, "Fully Convolutional Networks for Semantic Segmentation," p. 10.

[15] V. Badrinarayanan, A. Kendall, and R. Cipolla, "SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation," IEEE Trans. Pattern Anal. Mach. Intell., vol. 39, no. 12, pp. 2481–2495, Dec. 2017.

[16] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional Networks for Biomedical Image Segmentation," ArXiv150504597 Cs, May 2015.

[17] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," in 2014 IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 2014, pp. 580–587.

[18] R. Girshick, "Fast R-CNN," in 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440–1448.

[19] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," in Advances in Neural Information Processing Systems 28, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 91–99.

[20] K. He, G. Gkioxari, P. Dollar, and R. Girshick, "Mask R-CNN," IEEE Trans. Pattern Anal. Mach. Intell., pp. 1–1, 2018.

[21] P. O. Pinheiro, R. Collobert, and P. Dollar, "Learning to Segment Object Candidates," in Advances in Neural Information Processing Systems 28, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 1990–1998.

[22] J. Dai, K. He, and J. Sun, "Instance-Aware Semantic Segmentation via Multi-task Network Cascades," in 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, pp. 3150–3158.

[23] L. N. Smith, "Cyclical Learning Rates for Training Neural Networks," ArXiv150601186 Cs, Jun. 2015.

[24] Y. Dauphin, H. de Vries, and Y. Bengio, "Equilibrated adaptive learning rates for non-convex optimization," in Advances in Neural Information Processing Systems 28, C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, Eds. Curran Associates, Inc., 2015, pp. 1504–1512.

[25] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: Object Detection via Region-based Fully Convolutional Networks," in Advances in Neural Information Processing Systems 29, D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, Eds. Curran Associates, Inc., 2016, pp. 379–387.

[26] S. Li, H. Lu and X. Shao, "Human Body Segmentation via Data-Driven Graph Cut," in *IEEE Transactions on Cybernetics*, vol. 44, no. 11, pp. 2099-2108, Nov. 2014.