Received September 29, 2017, accepted October 31, 2017, date of publication November 10, 2017,
date of current version December 22, 2017.

Digital Object Identifier 10.1109/ACCESS.2017.2772269

Segmentation Mask Refinement Using Image Transformations

TUNG DUC NGUYEN[1], AYUMU SHINYA[1], TOMOHIRO HARADA[2], AND RUCK THAWONMAS[2], (Senior Member, IEEE)

[1]Graduate School of Information Science and Engineering, Ritsumeikan University, Shiga 525-8577, Japan
[2]College of Information Science and Engineering, Ritsumeikan University, Shiga 525-8577, Japan

Corresponding author: Ruck Thawonmas (ruck@is.ritsumei.ac.jp)

This work was supported by the Strategic Research Foundation Grant-aided Project for Private Universities, Japan, under Grant S1511026.

ABSTRACT This paper discusses object proposal generation, which is a crucial step of instance-level semantic segmentation (instance segmentation). Known as a challenging computer vision task, the instance segmentation requires jointly detecting and segmenting individual instances of objects in an image. A common approach to this task is first to propose a set of class-agnostic object candidates in the forms of segmentation masks, which represent both object locations and boundaries, and then to perform classification on each object candidate. In this paper, we propose an effective refinement process that employs image transformations and mask matching to increase the accuracy of object segmentation masks. The proposed refinement process is applied to three state-of-the-art object proposal methods (DeepMask, SharpMask, and FastMask), and is evaluated on two standard benchmarks (Microsoft COCO and PASCAL VOC). Both the quantitative and qualitative results show the effectiveness of the process across various experimental settings.

INDEX TERMS Instance segmentation, object proposal, segmentation mask, convolutional neural networks, deep learning.

I. INTRODUCTION

Instance-level semantic segmentation or instance segmentation is the task of jointly detecting and segmenting individual instances of objects in an image. It inherits from two computer vision tasks: object detection and semantic segmentation, but considerably increases the difficulty level. Unlike object detection, its outputs are object segmentation masks representing both object locations and boundaries, rather than just bounding boxes. Unlike semantic segmentation, its goal is to assign not only category label but also instance label to every pixel in the input image. In other words, it requires distinguishing individual instances of the same category besides distinguishing different object categories.

Currently, there are several different approaches to instance segmentation. Dai *et al.* [1] and Hayder *et al.* [2] divide this task into three sub-tasks and solve them sequentially: their methods generate a set of object candidates in the forms of class-agnostic bounding boxes, then predict an object segmentation mask for each bounding box, and finally classify objects into categories. Hariharan *et al.* [3], Dai *et al.* [4], and Zagoruyko *et al.* [5] directly generate object proposals in the forms of class-agnostic segmentation masks, then perform object classification. More recently, Li *et al.* [6] and He *et al.* [7] proposed to use a single convolutional neural network (CNN) for generating bounding boxes, segmentation masks and classifying objects in parallel.

In this paper, we focus on object proposal generation—a crucial step of many instance segmentation methods above, with the aim of improving the accuracy of object segmentation masks. With the rapid advance of deep learning in recent years, methods which utilize deep CNNs to learn a direct mapping from images to segmentation masks dominate the current trend. However, they are far from perfect: they usually produce segmentation masks lacking of pixel-accurate object boundaries. As there is still room for improvement, we propose a simple but effective segmentation mask refinement process that can be easily applied to any object proposal methods. The proposed process uses image transformations, such as image flipping, to generate a new image from the input image, then matches and combines segmentation masks of the original and newly transformed images.

II. RELATED WORK

DeepMask [8] is one of the representative CNN-based object proposal methods, in which a deep CNN learns to generate object segmentation masks from an input image. The lower part (layers near the input layer) of the network is initialized with an existing network that was trained to perform image classification. This pre-trained network, referred to as *body net*, is a CNN with all the fully-connected layers removed. The upper part of DeepMask is then split into two branches: one branch outputs a segmentation mask given a fixed-scale image patch, and the other branch outputs a confidence score that measures how likely a whole object is centered in the patch. Note that since DeepMask follows the class-agnostic approach, it has no notion of the object category. After being trained, the network can be applied densely at multiple scales and locations in the input image to output all possible objects segmentation masks.

While DeepMask is successful in generating masks that roughly represent object locations and boundaries, it is not able to generate pixel-accurate masks. The reason behind this is that DeepMask uses only a feed-forward pass through the network and generates segmentation masks from its upper layers. It has been revealed that early layers in a deep network tend to capture low-level features such as colors and edges, while upper layers tend to capture more abstract and semantically meaningful information such as the presence of an object. In order to solve this problem, SharpMask [9] augments the feed-forward (bottom-up) architecture of DeepMask with a top-down refinement process that combines low-level information from the early layers with high-level information from the upper layers. Given an input image patch, the bottom-up pass outputs a coarse mask with low spatial resolution, which is the input of the top-down pass. Multiple refinement modules in the top-down pass then iteratively refine that mask by combining it with features at successively lower layers in the network. This process results in segmentation masks that better adhere to object boundaries.

Both DeepMask and SharpMask rely on an image pyramid containing images at multiple scales to handle the scale variance of objects during inference. Recently, Hu *et al.* [10] proposed a novel network architecture for object proposal generation, called FastMask, which can segment multi-scale objects in one shot. The network consists of three functional modules: *body*, *neck* and *head*. Similar to DeepMask and SharpMask, the body module is initialized with a pre-trained network and is responsible for extracting feature maps from the input image. The neck module is the key difference; it repeatedly zooms out the extracted feature maps to create a feature pyramid containing feature maps at multiple scales. The feature pyramid is then passed through a 1×1 convolutional layer for reducing dimensions. Next, dense sliding windows are extracted from the output feature maps and fed into the head module for multi-scale inference. The head module is then responsible for generating segmentation masks from those sliding windows.

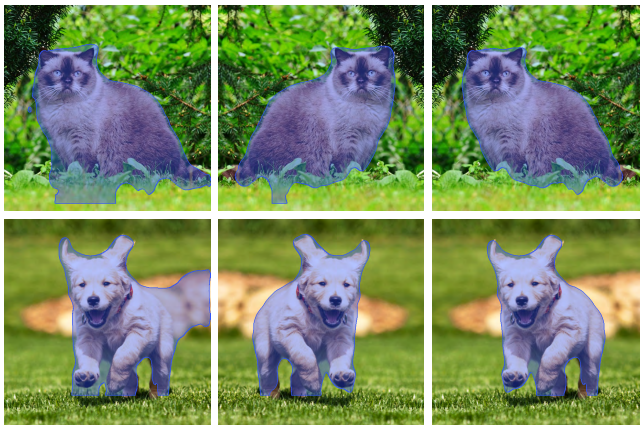

FIGURE 1. Example cases where segmentation masks generated for the original image and the transformed image are different. (a) Masks generated by SharpMask for original images, (b) masks generated by SharpMask for flipped images, (c) masks obtained by combining the masks in (a) and (b).

III. METHODOLOGY

While DeepMask, SharpMask, and FastMask can achieve good results compared to previous object proposal methods, they are still far from perfect. Masks generated by them often have boundaries outside real objects and cover redundant parts of the background, which is not well suited for tasks that require pixel-accurate masks. In order to further improve the accuracy of those masks, we propose a refinement process that employs image transformations and mask matching. This process is based on the observation that when an image is transformed, e.g., horizontally flipped as shown in Fig. 1, the masks of the transformed image may be different from—better or worse than—those of the original image. This also demonstrates that these networks are not transformation-invariant even though they are usually trained with data augmented by various image transformations such as resizing, cropping, flipping, and rotation. As a result, we can potentially combine the masks of the original image and the transformed image in some way to obtain even better masks.

Let us consider an image transformation T that takes as input an image x. The inverse of T, if exists, is a transformation T^{-1} such that $T^{-1}(T(x)) = x$. For example, for the horizontal flip T_{hf} and the vertical flip T_{vf}, $T_{hf}^{-1} = T_{hf}$ and $T_{vf}^{-1} = T_{vf}$. For the rotation $T_{r,\theta}$ which rotates the input image by θ degrees, $T_{r,\theta}^{-1} = T_{r,-\theta}$.

Let $SM()$ denote a segmentation mask generator, H and W denote the height and width of the original image x. The proposed segmentation mask refinement with respect to an image transformation T proceeds as follows:

1) Generate segmentation masks for objects in x:

$$\mathcal{M} = SM(x) = \{m_i\} \quad (i = 1, 2, \ldots, N) \qquad (1)$$

where $m_i \in \mathbb{R}^{H \times W}$ is a binary mask. An element of m_i is 1 if the corresponding pixel belongs to an object and 0 otherwise. N is a predefined number of masks.

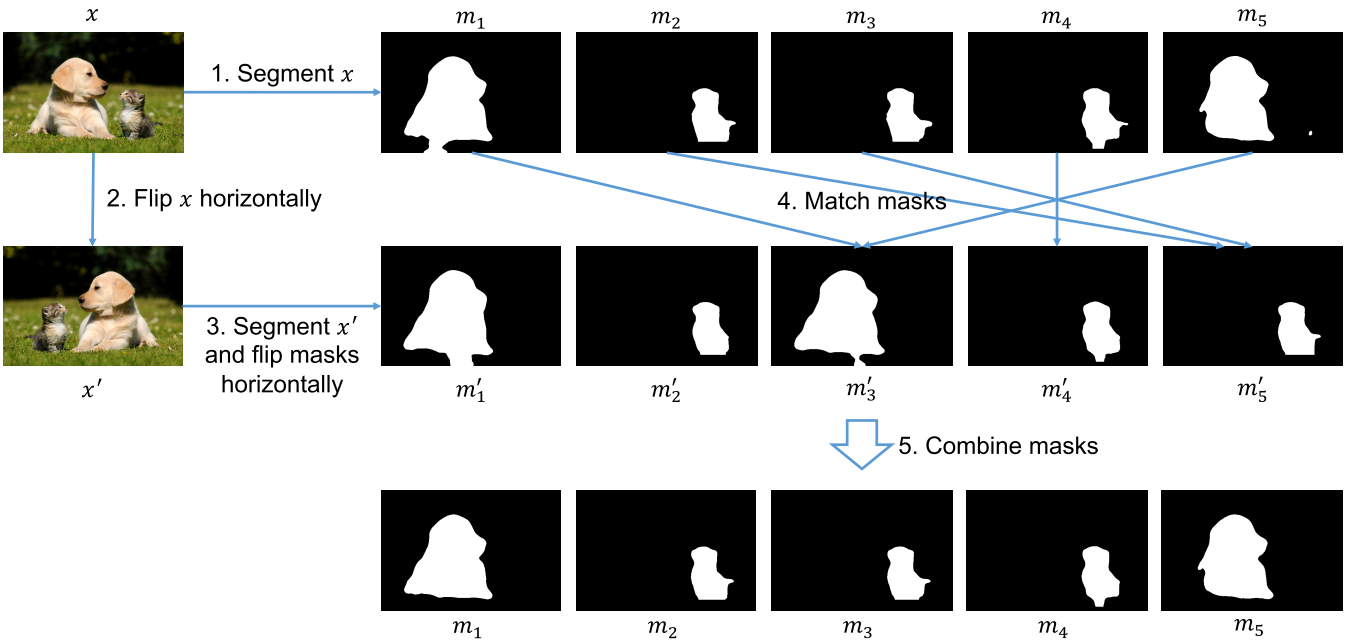

FIGURE 2. An illustration of the proposed refinement process with horizontal flip and $N = 5$.

2) Transform the original image x to form a new image x':

$$x' = T(x) \tag{2}$$

3) Generate segmentation masks for objects in x', then apply the inverse transformation to the generated masks:

$$\mathcal{M}' = T^{-1}\left(SM\left(x'\right)\right) = \left\{m'_j\right\} \quad (j = 1, 2, \ldots, N) \tag{3}$$

where $m'_j \in \mathbb{R}^{H \times W}$ is a binary mask. In general, $\mathcal{M} \neq \mathcal{M}'$ because most networks are not transformation-invariant.

4) Match each mask m_i with a mask $m'_{\phi(i)}$ that has the highest intersection-over-union (IoU) value with it:

$$\begin{aligned} \text{IoU}_{ij} &= \frac{|m_i \wedge m'_j|}{|m_i \vee m'_j|} \\ \phi(i) &= \arg\max_j \text{IoU}_{ij} \end{aligned} \tag{4}$$

where $\wedge$ and $\vee$ denote element-wise minimum and maximum operators, respectively, and $|m|$ denotes the sum of all elements in m. It is worth noting that rather than matching m_1 with m'_1, m_2 with m'_2, etc., this step matches each m_i with the most promising counterpart and thus may benefit from a large value of N.

5) For all i, combine m_i and $m'_{\phi(i)}$ by element-wise minimum operation:

$$m_i = m_i \wedge m'_{\phi(i)} \tag{5}$$

Henceforth, $\mathcal{M} = \{m_i\}$ is the set of refined segmentation masks. The quality of $\mathcal{M}$ depends critically on the image transformation T. While there are several options for T, we propose to use horizontal flip because empirically it results in natural-looking images that usually do not cause troubles, e.g., missing objects or incorrectly locating object boundaries, for mask generator networks. An illustration of the whole process with horizontal flip and $N = 5$ is given in Fig. 2.

TABLE 1. Statistics of the two datasets used in experiments.

	MS COCO	PASCAL VOC
Images	5000	2913
Categories	80	20
Objects	35078	6934
Objects per image	7.0	2.4
Objects per category	438.5	346.7

IV. EVALUATION

A. EXPERIMENTS

The proposed refinement process is analyzed and evaluated using various experimental settings. For reference, some variants of the process that employs vertical flip and rotation are also examined.

1) DATASETS

Two standard datasets for instance segmentation—Microsoft COCO (MS COCO) [11] and PASCAL VOC [12]—are selected as benchmarks for several object proposal methods in our experiments. In particular, we use the first 5000 images in the MS COCO 2014 validation set (following previous studies [8]–[10]), and all images in the PASCAL VOC 2012 training and validation set. Statistics of the two datasets are summarized in Table 1.

TABLE 2. Average recall (%) given different numbers of proposals per image.

Method	MS COCO			PASCAL VOC		
	AR@10	AR@100	AR@1000	AR@10	AR@100	AR@1000
DeepMask	15.1	28.6	37.1	24.5	35.0	40.7
DeepMask+hflip	15.9	30.1	38.9	26.1	36.9	42.4
DeepMask+vflip	15.2	28.7	36.9	25.6	35.7	40.8
DeepMask+rt90	15.1	28.4	36.8	25.5	35.6	40.7
DeepMask+rt180	15.3	28.8	37.1	25.7	36.0	40.9
DeepMask+rt270	15.0	28.3	36.4	25.4	35.4	40.4
SharpMask	16.0	30.3	39.4	25.7	36.8	42.8
SharpMask+hflip	**17.3**	**32.2**	**41.3**	**28.2**	**39.4**	44.9
SharpMask+vflip	16.7	30.9	39.5	27.7	38.2	43.2
SharpMask+rt90	16.3	30.2	38.7	27.1	37.5	42.5
SharpMask+rt180	16.8	31.1	39.7	27.8	38.5	43.4
SharpMask+rt270	16.1	30.0	38.3	27.1	37.4	42.4
FastMask	16.9	31.3	40.9	25.2	36.9	45.1
FastMask+hflip	17.0	31.4	41.0	26.4	38.2	**45.8**
FastMask+vflip	16.8	31.1	40.6	25.4	36.9	44.8
FastMask+rt90	16.8	31.1	40.6	25.4	36.9	44.9
FastMask+rt180	16.8	31.1	40.6	25.5	36.9	44.8
FastMask+rt270	16.8	31.1	40.6	25.4	36.9	44.9

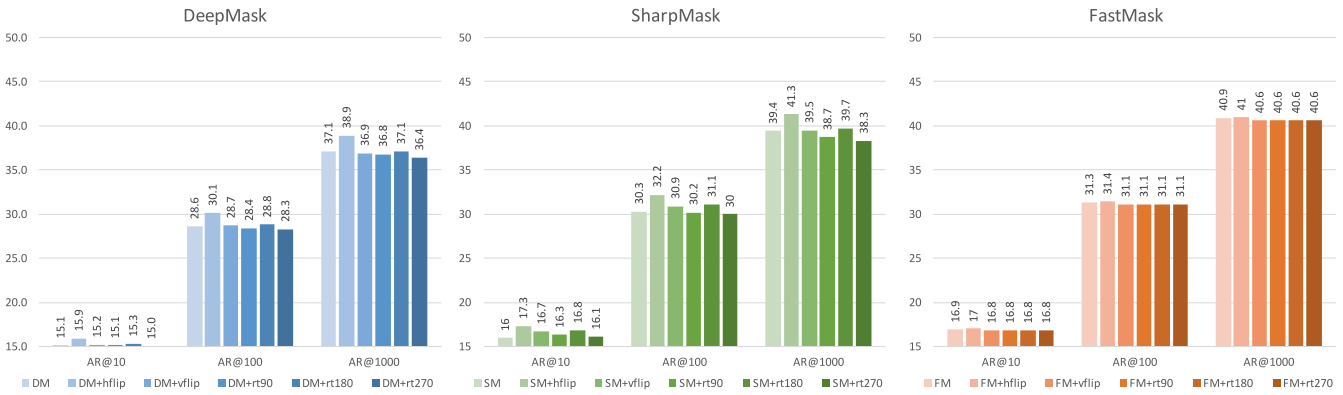

FIGURE 3. Performance comparisons of baseline methods and variants on the MS COCO dataset: DM, SM, FM denote DeepMask, SharpMask, FastMask, respectively.

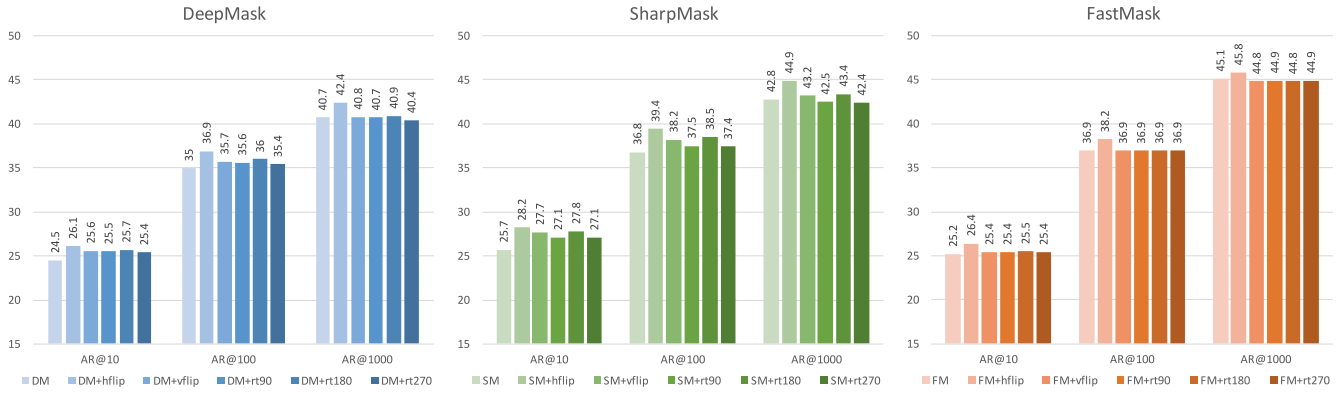

FIGURE 4. Performance comparisons of baseline methods and variants on the PASCAL VOC dataset: DM, SM, FM denote DeepMask, SharpMask, FastMask, respectively.

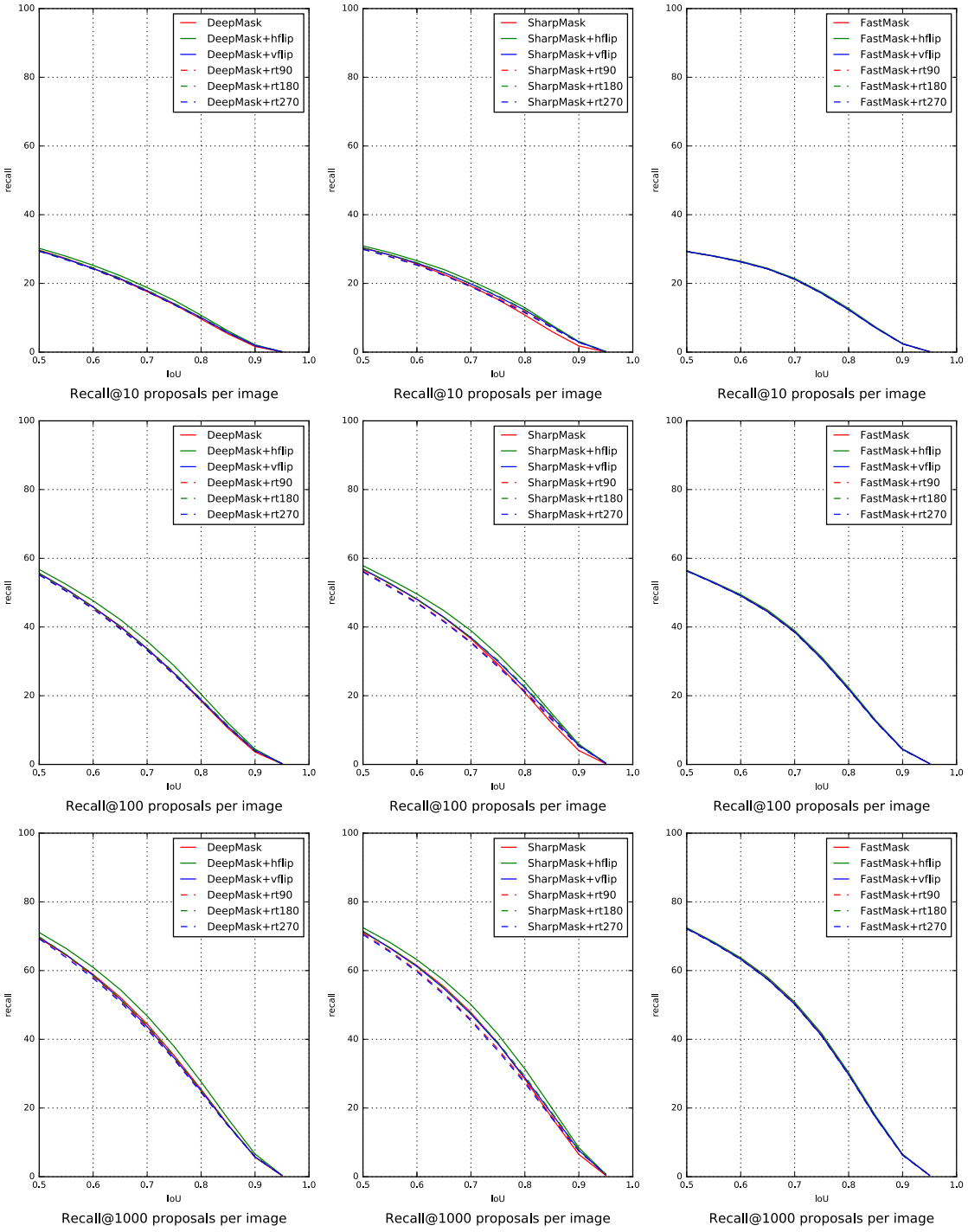

FIGURE 5. Recall (%) vs. IoU threshold on the MS COCO dataset.

2) METHODS

DeepMask, SharpMask, and FastMask are selected as baseline object proposal methods since currently they are state-of-the-art and their trained models have been made publicly available.[1] For fair comparison, all models adopt ResNet-50 [13] as their body networks and were trained on the MS COCO 2014 training set, which contains about 80,000 images having 500,000 segmented objects of 80 common categories. For DeepMask and SharpMask, we use the "zoom" setting (DeepMaskZoom and SharpMaskZoom) for inference as it can boost the performance particularly for small objects. We then combine the three baseline

[1]DeepMask and SharpMask: https://github.com/facebookresearch/deepmask FastMask: https://github.com/voidrank/FastMask

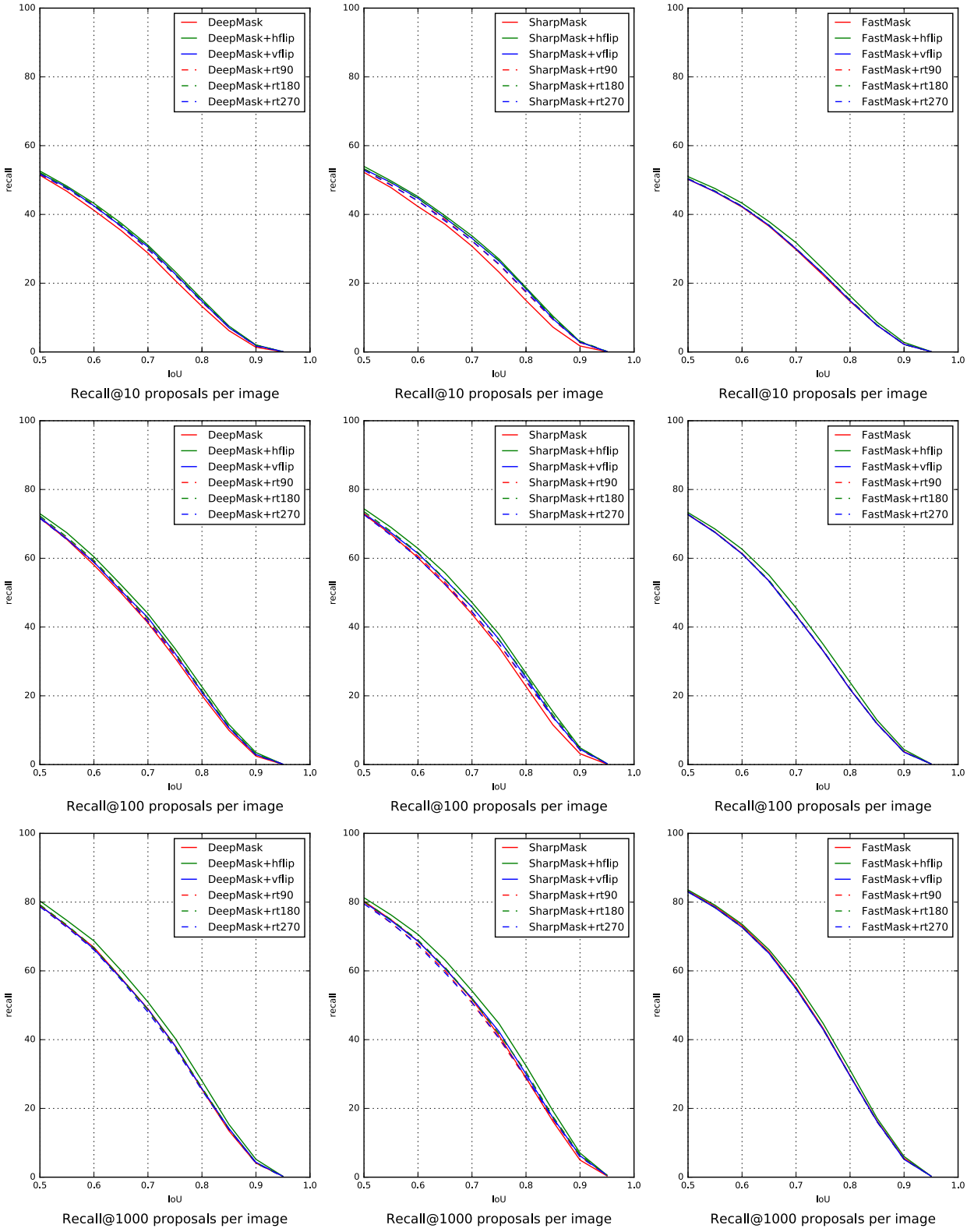

FIGURE 6. Recall (%) vs. IoU threshold on the PASCAL VOC dataset.

methods with five variants of the proposed process that employ horizontal flip (hflip), vertical flip (vflip), counterclockwise rotation of 90 degrees (rt90), counterclockwise rotation of 180 degrees (rt180), and counterclockwise

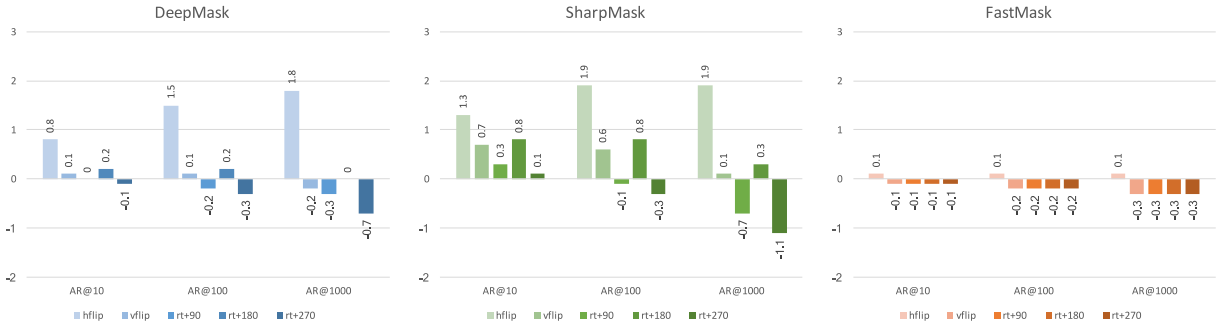

FIGURE 7. Gains in average recall by the proposed refinement process and its variants on the MS COCO dataset.

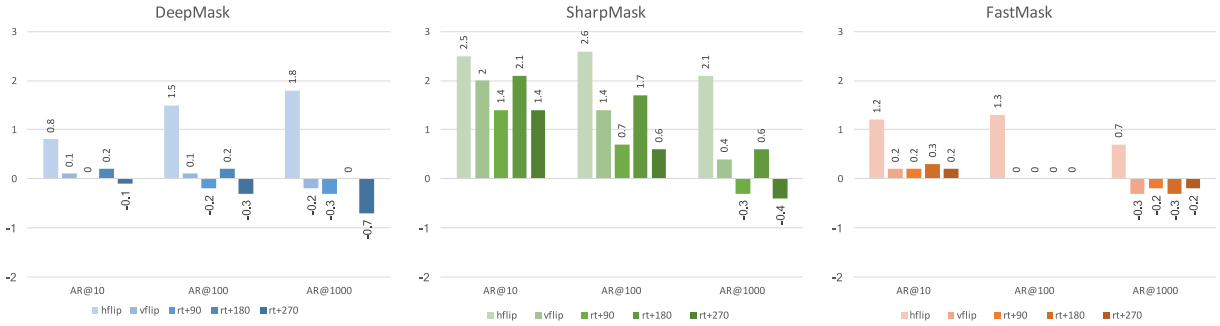

FIGURE 8. Gains in average recall by the proposed refinement process and its variants on the PASCAL VOC dataset.

rotation of 270 degrees (rt270), which results in a total of 18 object proposal methods.

3) METRICS

Segmentation masks generated by the methods above for images in the two datasets are compared to ground truths available in the datasets. For a generated mask and a ground truth mask, if their IoU is above a certain threshold, the generated mask is considered to be "accurate" and the object corresponding to the ground truth mask is considered to be "accurately detected". A common metric for evaluating the performance of an object segment proposal method is *recall*, which is defined as the number of accurately detected objects divided by the total number of objects. Following previous studies, we compute recall at multiple IoU thresholds and report the average value, i.e., *average recall* (AR). AR rewards both high recall and good localization, and has been shown to correlate extremely well with detection performance [14]. In detail, the IoU thresholds in use are 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, and 0.95. In addition, by definition, recall increases when a method generates more segmentation masks. Therefore, AR is computed with regard to only a specific number of proposals per input image, which is 10, 100, and 1000 in this work.

B. QUANTITATIVE RESULTS

Table 2 reports average recall of each method on the two datasets. AR@10, AR@100, and AR@1000 denotes average recall given 10, 100, and 1000 proposals per input image, respectively. Numbers in bold are the highest values in corresponding columns. These results are analyzed from several perspectives in the rest of this subsection.

1) IS THE PROPOSED REFINEMENT EFFECTIVE?

The reported results are divided into groups of six: baseline, baseline+hflip, baseline+vflip, baseline+rt90, baseline+rt180, and baseline+rt270; comparisons within each group is visualized in Figs. 3 and 4. As can be seen, DeepMask+hflip, SharpMask+hflip, and FastMask+hflip outperform their baselines in all experimental settings on both datasets, i.e., the proposed refinement process helps improve DeepMask, SharpMask, and FastMask. In addition, multiple recall-vs-iou curves shown in Figs. 5 and 6 confirm that the proposed refinement is effective not only in terms of average recall, but also in terms of separate recall at different IoU thresholds. The reason behind the effectiveness of the proposed process lies in the mask matching and combination steps, which take advantages of masks of two images, original and flipped, to generate even better masks.

2) WHICH IMAGE TRANSFORMATION IS THE BEST?

The proposed process with horizontal flip results in better improvement than all of its variants. Although this conclusion can be drawn from the figures above, for a clearer comparison, absolute gains in average recall by each of the image transformations are plotted in Figs. 7 and 8. While horizontal

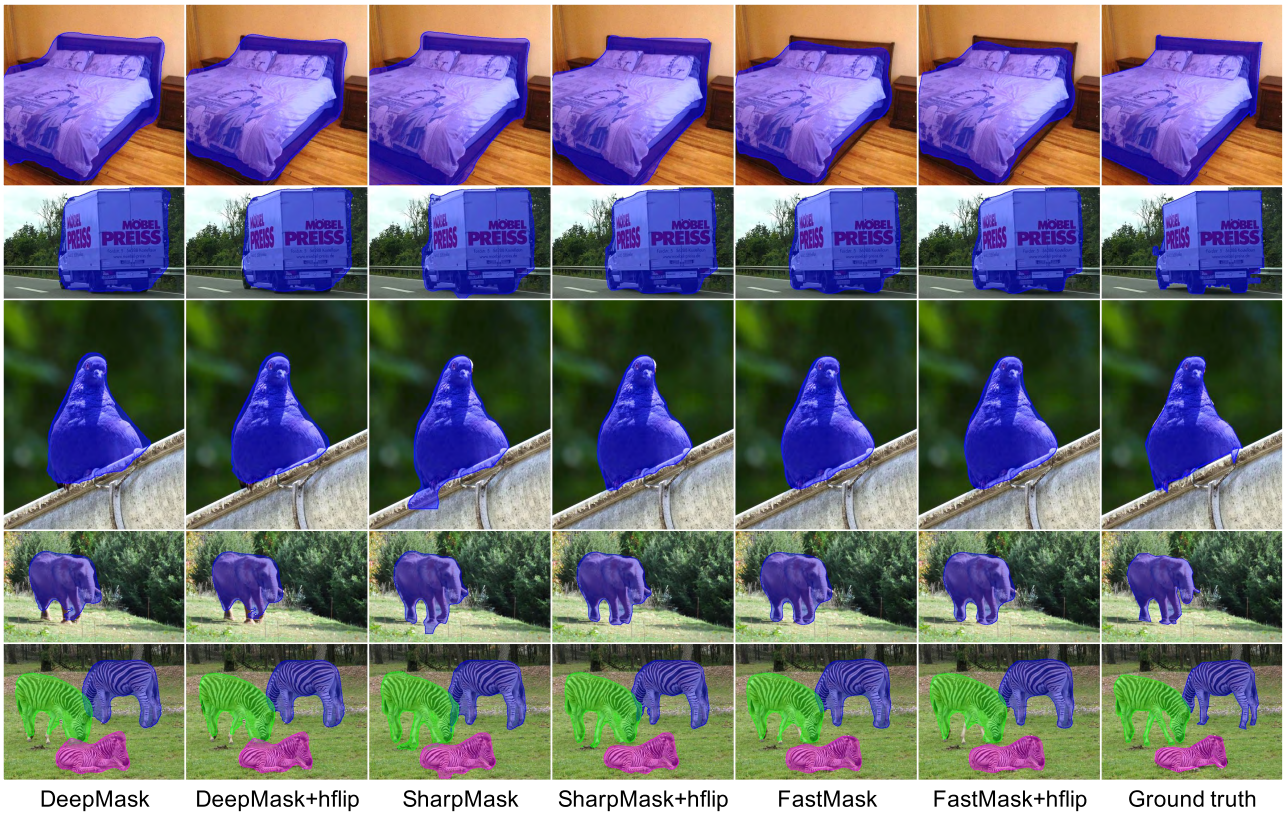

FIGURE 9. Examples of segmentation masks generated by different methods for images in the MS COCO dataset.

flip increases average recall of all the baseline methods, other transformations sometimes even have lower performance. The superiority of horizontal flip over the others can be explained by that many objects appear in horizontally symmetrical shapes and thus flipping an image horizontally still results in a natural-looking image. On the other hand, flipping it vertically or rotating it results in an image containing strange-looking objects. In addition, vertically flipped images and rotated images are usually not present in popular training datasets, thus may cause troubles for neural networks during inference, unless they are specifically trained to be invariant to such transformations.

3) WHICH METHOD BENEFITS THE MOST FROM THE PROPOSED REFINEMENT?

Figures 7 and 8 show that among the three baseline methods, SharpMask benefits the most from the proposed refinement. In addition, as can be seen in Table. 2, SharpMask+hflip surpasses FastMask in almost all experimental settings and becomes the best method in general. One more finding is that the performance of DeepMask+hflip is very close to that of SharpMask, which is encouraging because as mentioned in Section II, SharpMask is an extension of DeepMask with complex refinement modules. Unlike DeepMask and SharpMask, FastMask—the best among the three baselines—does not benefit much from the proposed refinement.

Besides the findings above, it is worth noting that all methods perform better on the PASCAL VOC dataset than on the MS COCO dataset even though they are all trained on the latter dataset. The reason is that the average number of objects per image in the PASCAL VOC is only about one third of that in the MS COCO dataset, as shown in Table 1. During inference, given the same number of proposals for images in both datasets, the dataset with a smaller number of ground truth objects is likely to lead to higher recall.

C. QUALITATIVE RESULTS

This subsection shows examples of segmentation masks generated by different methods, the three baselines and their variants where the proposed refinement process is applied, for images in the MS COCO dataset and the PASCAL VOC dataset. Since each segmentation mask is associcated with a confidence score by a method of interest, only the masks with highest confidence scores are visualized. Alternatively, as all images are provided with ground truth annotations, one can visualize the masks with the highest IoU with ground truths, as done in existing work [8], [9]. However, such an approach is not followed here because it is not suitable for practical usage. In practice, because ground truth annotations are usually not given, one must rely on the aforementioned confidence scores to select the best masks for their purposes. Figures 9 and 10 confirm that the proposed refinement can

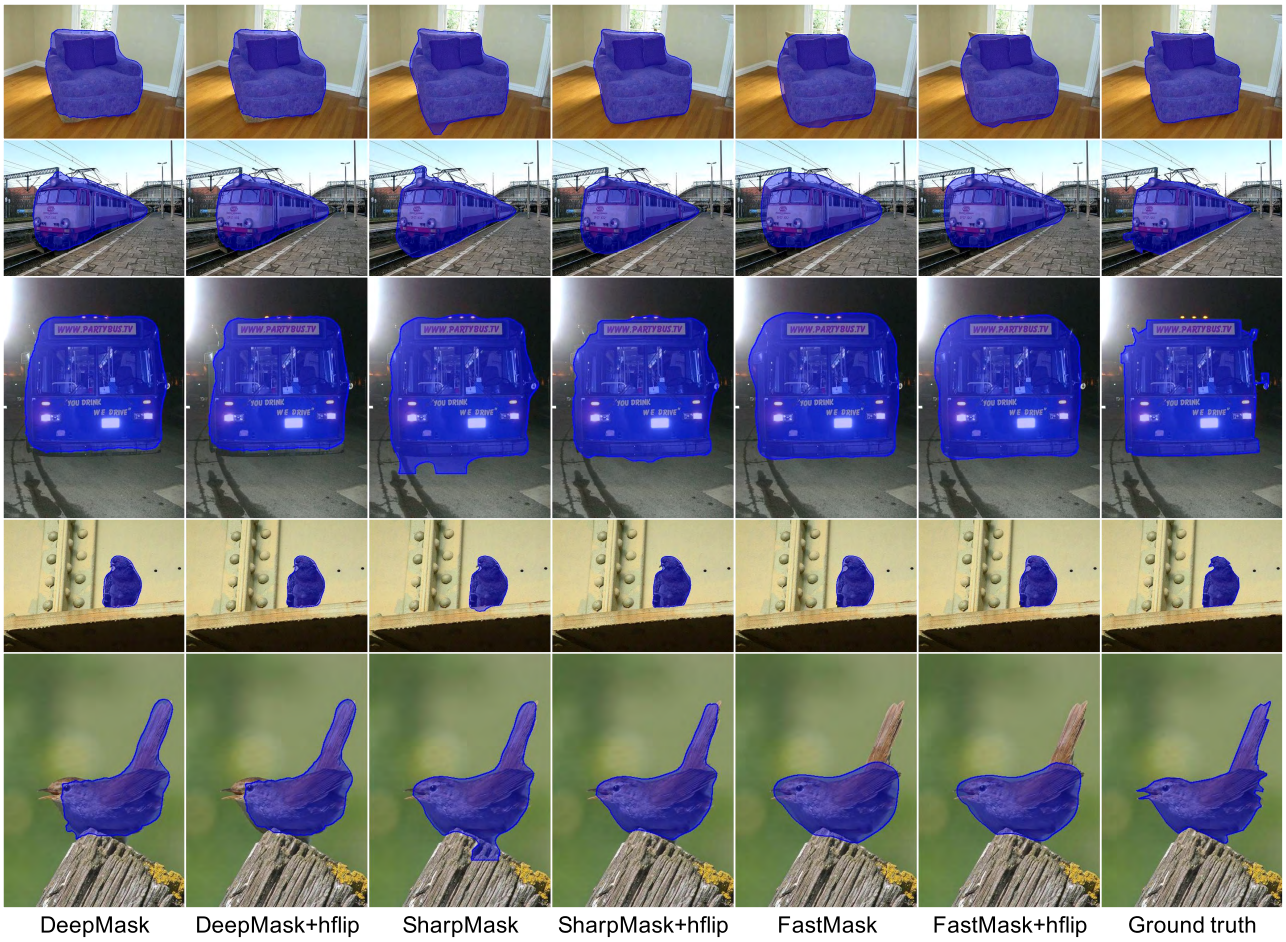

FIGURE 10. Examples of segmentation masks generated by different methods for images in the PASCAL VOC dataset.

help all methods, especially SharpMask, generate segmentation masks with higher accuracy, i.e., masks that delineate object boundaries more precisely.

V. CONCLUSIONS

In this paper, we proposed a simple but effective segmentation mask refinement process that can be applied to any object proposal methods to increase the accuracy of object segmentation masks. The process works by matching and combining the masks generated for an input image and its horizontally flipped image. We applied the process to three current state-of-the-art baseline methods (DeepMask, SharpMask, FastMask), and conducted various experiments on two standard datasets (the MS COCO dataset and the PASCAL VOC datatset). The results confirm that the proposed process is effective, i.e., it improves the performance (average recall) of all baseline methods in various experimental settings on both datasets.

In future work, more image transformations or their combinations, as well as other methods for combining masks will be examined to further improve the proposed refinement process. Moreover, the process may be applied to state-of-the-art instance segmentation methods such as FCIS [6] or Mask R-CNN [7]. Since object proposal generation is a crucial step of instance segmentation, it is expected to improve those methods with regard to both detection and segmentation performances.

REFERENCES

[1] J. Dai, K. He, and J. Sun, "Instance-aware semantic segmentation via multi-task network cascades," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 3150–3158.

[2] Z. Hayder, X. He, and M. Salzmann, "Boundary-aware instance segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 5696–5704.

[3] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik, "Simultaneous detection and segmentation," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 297–312.

[4] J. Dai, K. He, and J. Sun, "Convolutional feature masking for joint object and stuff segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 3992–4000.

[5] S. Zagoruyko *et al.*, "A multipath network for object detection," in *Proc. Brit. Mach. Vis. Conf.*, Sep. 2016, pp. 15.1–15.12, doi: https://dx.doi.org/10.5244/C.30.15.

[6] Y. Li, H. Qi, J. Dai, X. Ji, and Y. Wei, "Fully convolutional instance-aware semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 4438–4446.

[7] K. He, G. Gkioxari, P. Dollár, and R. Girshick. (Apr. 2017). "Mask R-CNN." [Online]. Available: https://arxiv.org/abs/1703.06870

[8] P. O. Pinheiro, R. Collobert, and P. Dollar, "Learning to segment object candidates," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 1990–1998.
[9] P. O. Pinheiro, T.-Y. Lin, R. Collobert, and P. Dollár, "Learning to refine object segments," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 75–91.
[10] H. Hu, S. Lan, Y. Jiang, Z. Cao, and F. Sha. (Apr. 2017). "FastMask: Segment multi-scale object candidates in one shot." [Online]. Available: https://arxiv.org/abs/1612.08843
[11] T.-Y. Lin *et al.*, "Microsoft COCO: Common objects in context," in *Proc. Eur. Conf. Comput. Vis.*, 2014, pp. 740–755.
[12] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The Pascal visual object classes (VOC) challenge," *Int. J. Comput. Vis.*, vol. 88, no. 2, pp. 303–338, Sep. 2009.
[13] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2016, pp. 770–778.
[14] J. Hosang, R. Benenson, P. Dollár, and B. Schiele, "What makes for effective detection proposals?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 4, pp. 814–830, 2016.

TUNG DUC NGUYEN received the B.Eng. degree in human and computer intelligence and the M.Eng. degree in human information science from Ritsumeikan University, Kusatsu, Japan, in 2014 and 2017, respectively. He was with the Framgia Vietnam Company Ltd., Vietnam, from 2014 to 2015, and rejoined in 2017. His research interests include image style-transfer, colorization and segmentation using deep learning, machine learning, and artificial intelligence applied to computer games. He was the principal developer of IceBot winning both CIG 2014 StarCraft AI Competition and AIIDE 2014 StarCraft AI Competition. He was a recipient of the Human Higher Education Development Support Project on ICT Scholarship and the Japanese Government (MEXT) Scholarship, respectively. He was also a co-recipient of the IEEE GCCE 2013 Outstanding Student Paper Award, as the first author, and the IEEE GCCE 2015 Outstanding Student Paper Award, as the second author.

AYUMU SHINYA received the B.Eng. degree from the College of Information Science and Engineering, Ritsumeikan University, in 2017, where he is currently pursuing the Degree with the Graduate School of Information Science and Engineering. He passed the Information Technology Engineers Examination, conducted by the Information-Technology Promotion Agency, Japan, in 2014. He is currently involved in research on image colorization and style-transfer using deep learning, and has a number of peer-reviewed publications in these areas.

TOMOHIRO HARADA received the B.S., M.E., and Ph.D. degrees from the University of Electro-Communications, Japan, in 2010, 2012, and 2015, respectively, all in engineering. He was a Research Fellow of the Japan Society of the Promotion of Science DC1 from 2012 to 2015. He has been an Assistant Professor with the College of Information Science and Engineering, Ritsumeikan University, since 2015. His research interests include evolutionary computation, machine learning, game AI, and sleep measurement. He is a member of JSAI, IPSJ, SICE, and the Japanese Society for Evolutionary Computation. He has been an Editorial Committee Member of the Information Processing Society of Japan Journal since 2017. He received the Journal Paper Award from System and Information, the Society of Instrument and Control Engineering in 2012.

RUCK THAWONMAS (M'97–SM'99) received the B.Eng. degree in electrical engineering from Chulalongkorn University, Bangkok, Thailand, in 1987, the M.Eng. degree in information science from Ibaraki University, Ibaraki, Japan, in 1990, and the D.Eng. degree in information engineering from Tohoku University, Sendai, Japan, in 1994. Since 2002, he has been with the College of Information Science and Engineering, Ritsumeikan University, where he is currently a Professor. He was with various institutions, including Hitachi, Ltd., RIKEN, the University of Aizu, and the Kochi University of Technology. His research interests include artificial intelligence, computational intelligence, and their applications to image processing and computer games. He was a recipient of the MEXT Scholarship during 1987–1993. He has been an Associate Editor of the IEEE TRANSACTIONS ON COMPUTATIONAL INTELLIGENCE AND AI IN GAMES and *Games for Health Journal* since 2014.