Takeki *et al. IPSJ Transactions on Computer Vision and Applications* (2016) 8:5
DOI 10.1186/s41074-016-0006-z

IPSJ Transactions on Computer Vision and Applications

EXPRESS PAPER Open Access

Combining deep features for object detection at various scales: finding small birds in landscape images

Akito Takeki*, Tu Tuan Trinh, Ryota Yoshihashi, Rei Kawakami, Makoto Iida and Takeshi Naemura

Abstract

Demand for automatic bird ecology investigation rises rapidly along with the widespread installation of wind energy plants to estimate their adverse environmental effect. While significant advance in general image recognition has been made by deep convolutional neural networks (CNNs), automatically recognizing birds at small scale together with large background regions is still an open problem in computer vision. To tackle object detection at various scales, we combine a deep detector with semantic segmentation methods; namely, we train a deep CNN detector, fully convolutional networks (FCNs), and the variant of FCNs, and integrate their results by the support vector machines to achieve high detection performance. Through experimental results on a bird image dataset, we show the effectiveness of the method for scale-aware object detection.

1 Introduction

Wind turbines, one of the mainstream technologies for cultivating renewable energy sources, are yet at the same time considered serious threats to endangered bird species [1]. Assessments of bird habitats around planned sites are now required for the operators [2], whereas the surveys rely on experts who conduct manual observations. Automatic bird detection has hence drawn the attention of industry, as it can reduce the cost and increase the accuracy of investigations. It may also assist automatic systems that decelerate the blades or sound an alarm at the approach of birds.

When conducting bird surveillance with fixed-point cameras, however, three issues occur related to resolution and precision.

First, finding various scales of objects in large images has been addressed as a difficult problem because of the large differences in resolution. Second, images of surveillance cameras have different characteristic from those in general image recognition datasets, as objects captured by wide-field-of-view cameras are often ambiguous due to low resolution.

Finally, the number of flying birds is irregular and there are many scenes without any birds; thus, the detector is required to reduce false detections of backgrounds as few as possible for practical use.

To solve these problems, this paper presents a scale-aware bird detection method with practically high precision. Following the idea of scene parsing (e.g., [3]), we carefully select the combination of methods, each of which are suited for objects at different scales; specifically, a successor [4] of convolutional neural networks (CNNs) [5] for small birds and two kinds of fully convolutional networks (FCNs) for larger areas: the original FCNs [6] and DeepLab [7]. FCN-based methods can recognize both birds and backgrounds, while FCNs is more suited for middle-size birds, and DeepLab is good at backgrounds. Linear SVMs [8] are used to merge all the features for final results. This paper is based on our previous work [9] but improved so that features in the selected methods are all based on deep learning.

The proposed method was experimentally evaluated with a bird dataset especially constructed for ecological investigations around wind farms, showing that combining deep features from a detector and semantic segmentation is effective for scale-aware object detection. It achieved precision of 97 % in the bird detection task with 80 % recall rate.

*Correspondence: takeki@hc.ic.i.u-tokyo.ac.jp
The University of Tokyo, 7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan

1.1 Related work

The advances in CNNs and the growing availability of large-scale image datasets have brought outstanding improvements in image recognition. In particular, stronger learning models [10, 11] as well as effective techniques for suppressing overfitting [12] and avoiding the vanishing gradient problem [13] have significantly improved the performance of CNNs.

Many new detection methods have been proposed along with the advances in CNNs. In popular region-based CNN methods (R-CNN) [14], a selective search [15] is first used to identify potentially salient object regions (referred to as region proposal), from which image features are extracted by CNNs and classified by SVMs. We utilize ResNet [4], one of the most successful networks in detection, while we leave the region proposals as future work and use background subtraction for candidate region selection in this study.

Significant progress has also been made in semantic segmentation. There has been much debate about how to parse both object categories (*things*) and background categories (*stuff*), each of which account for smaller and larger parts of images. Various methods parse stuff and things separately with region-based and detector-based methods [3, 16].

Recently, a number of semantic segmentation methods have been proposed that are based on FCNs [6, 7].

FCNs can obtain a coarse object label map from the networks by combining the final prediction layer with lower layers (skip layer) [17], where the context and localization information are available for pixel-wise labeling.

DeepLab use the hole algorithm [18], which convolutes every other pixel. This approach can grasp the feature map more sparsely, which improves the ability to recognize background.

2 Method

An overview of the proposed method is illustrated in Fig. 1.

An input image is fed into three pipelines: (1) ResNet-based CNNs as a detector for small birds after a background subtraction pre-processing, (2) FCNs as a method that works as a detector but also as a semantic segmentation, and (3) DeepLab as a method that works as a semantic segmentation. SVMs combine the class likelihoods and scores derived from three pipelines. The outcomes of the method are regions estimated to be birds.

2.1 CNNs for bird detection

We designed the CNN network model using ResNet [4], which achieved the best results in the detection and classification of ILSVRC 2015. In ResNet, the input of a convolutional (conv) layer bypasses one or more layers and is added to the outputs of the stacked layers. Compared with previous net structures, ResNet learns so-called residual mappings, which make the learning easier even with deeper structures.

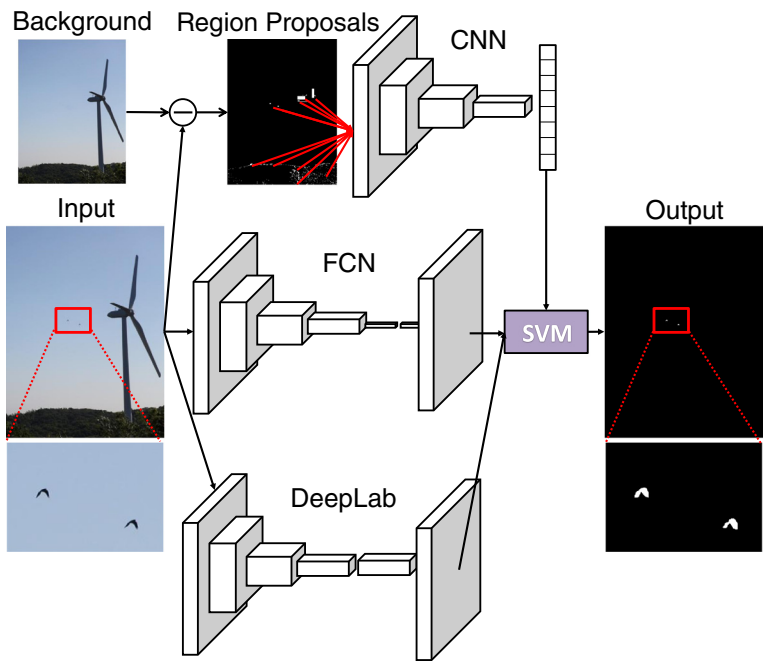

Fig. 1 Overview of the proposed method

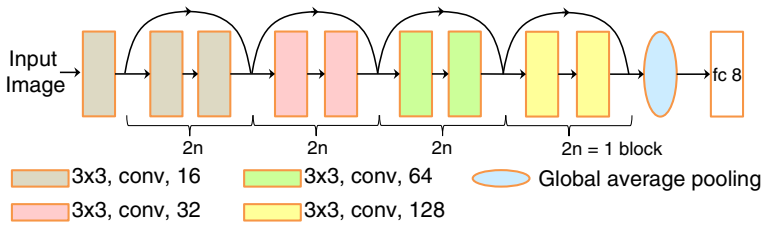

Fig. 2 Example CNN architecture for small bird image detection

Figure 2 shows our network architecture based on ResNet. We assume the sizes of the bird images ranges from 10 to 200 pixels square; thus, we design the networks to take 64×64 images as inputs, doubled the size of the original. Any size of detected bounding boxes will be fitted to 64×64 and fed into the networks. Because of this, one more block (the layers in yellow) is added before the global average pooling to capture features effectively with more hierarchies. Experimentally, the combination of four blocks with $n = 2$ produces the best results; four blocks with $n = 3$ produce similar results but require a longer training time, fewer blocks have less accuracy even with larger n, and more blocks cause overfitting even with fewer n.

The rest of the networks follows [4]; here, we briefly explain it for completeness. In every conv layer, the size of the kernels is 3×3. The very first conv layer has 16 kernels. Subsequently, there are four blocks, each of which includes four ($2n$ with $n = 2$) conv layers. The number of kernels is 16, 32, 64, and 128 in each block, respectively. When the dimensions increase by shortcut connections, we use 1×1 convolutions with a stride of 2 to equalize the input and output dimensions.

The first of four conv layers in the second and later blocks includes a stride of two subsamples, and this reduces the feature map size into half. Thus, the feature map size (64×64) becomes 64, 32, 16, and 8, after the process of each respective block. Finally, the ends of convolutions are connected using global average pooling, an eight-way fully connected layer (fc 8) and softmax. We use 18 stacked weighted layers in total.

2.2 Combining class likelihoods by SVM

We modified FCNs and DeepLab to have four classes (i.e., bird, sky, forest, and wind turbine), and CNNs have eight classes from its architecture, which we selected them as follows: bird, blade, tower, anemometer, nacelle, hub, forest, and other. The implementation details of FCNs and DeepLab are provided in the training section.

Each of the three pipelines yields a class-wise likelihood or score: FCNs and DeepLab generate pixel-wise likelihoods of classes, whereas CNNs generate a bounding box-wise score of the likelihoods of classes. For SVM training, we use only the pixels at the center of the bounding boxes of candidate regions proposed by the inter-frame difference method in order to reduce calculation time, so that it finishes within a reasonable amount of time. After the first training, we use hard negative mining to reduce false positives and to improve the overall performance. Specifically, image regions of anemometers, night lights, the lower parts of nacelles, in which the FCNs often produce false detections, are added for SVM training. The pixels collected by the inter-frame difference have statistical difference from the true pixel distribution. Because of this, when CNNs are simply combined with semantic segmentation-based methods, the whole framework inclines to include many misdetections by CNNs; thus, we add the background regions (sky, cloud, forest, and wind turbine) inside the candidate bounding boxes in the training.

3 Experimental results

We implemented CNNs, FCNs, and DeepLab, as well as AdaBoost with Haar-like feature [19, 20] and SuperParsing [21] as baselines. Then, we also trained several combinations of methods with our proposed framework and evaluated their performance using a wide-area surveillance dataset of wild birds [22], which contains a set

Table 1 F-measure of various methods

Method	Precision	Recall	F-measure
HA	0.064	0.514	0.114
SP	*1.000*	0.366	0.536
FCN	0.684	0.519	0.590
FCN*	0.709	0.585	0.641
SP*	0.989	0.508	0.672
DL	*1.000*	0.557	0.716
CNN	0.598	*0.902*	*0.719*
FCN+DL	0.979	0.527	0.664
CNN+DL	0.799	0.628	0.703
CNN+FCN	0.924	0.798	0.856
CNN+FCN+DL	*0.974*	*0.803*	*0.880*

*represents the method combined with SVMs

of images with 2806×3744 pixels taken nearby a wind turbine.

3.1 Data

For training of SuperParsing, FCNs, and DeepLab, we picked out 82 images with different weather conditions from the dataset and manually annotated them into four classes: bird, wind turbine, sky, and forest, which are all classes included in [22]. Finally, 77 images out of 82 were used and 5 were omitted since they were too dark due to stormy weather. Except for SuperParsing, the images were cropped to 500×500 pixels because the original images were too large to process with FCNs and DeepLab on our GPU memory. Cropping the entire image randomly causes many frames only tagged with the sky labels because more than a half of each image was occupied by sky. With this in mind, we performed cropping around the wind turbine area more intensively, and obtained 70 frames from each image by shifting a 500×500 pixel window through the area. Eventually, we had 77×70 = 5390 frames for training FCNs and DeepLab.

The training images for ResNet were acquired as candidate regions of moving objects with background subtraction from the entire dataset. The training images include bird and non-bird regions, and we prepared a class of bird and seven background classes.

These extra classes help training the networks because they are frequently included in the candidate regions and likely to cause misdetection. We categorized candidate

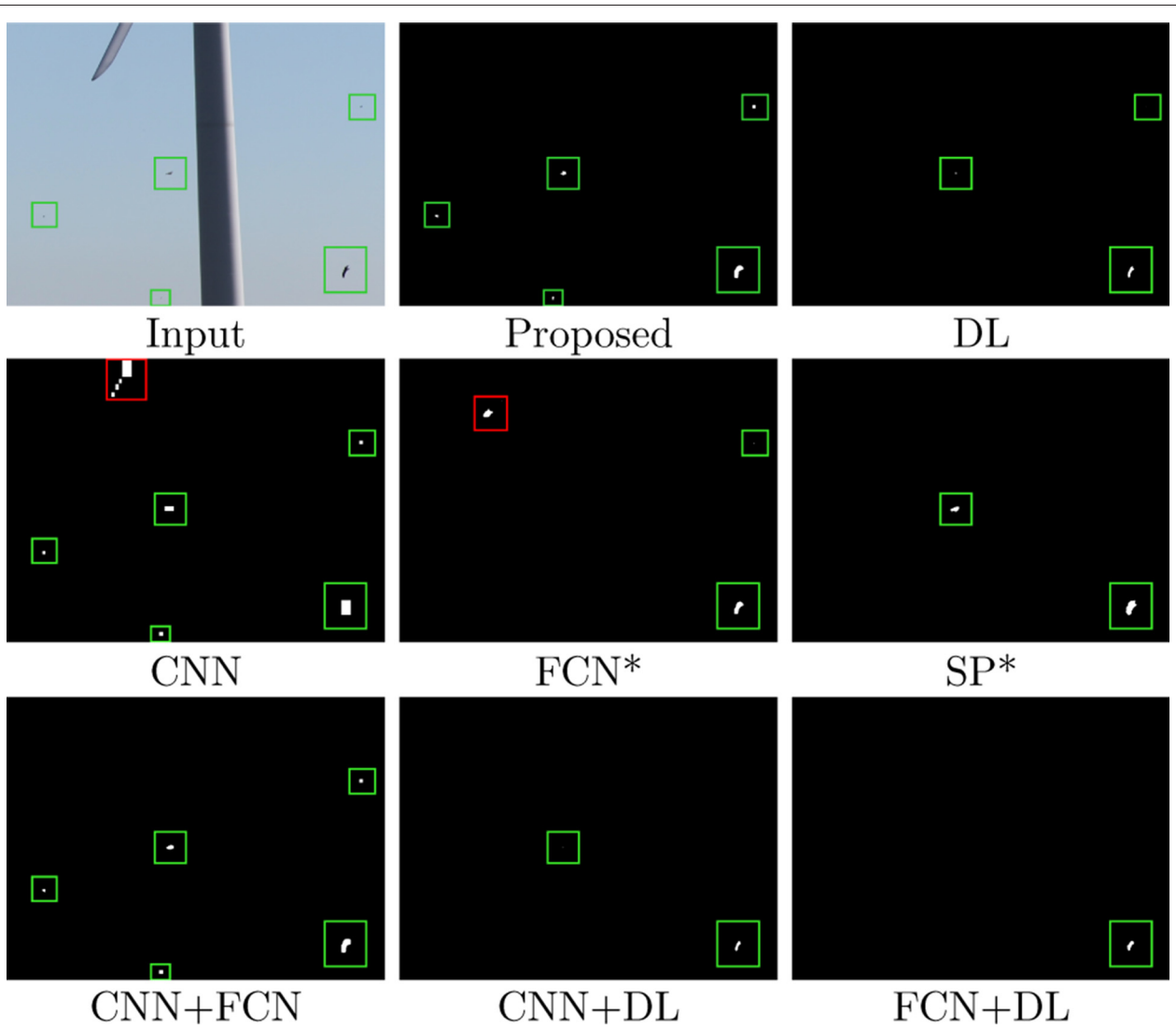

Fig. 3 Examples of detection results on the bird image dataset intended for ecological investigations. The *green squares* mean TP. The *red squares* mean FP

regions into those eight classes manually. To train the AdaBoost with Haar-like features, we used 15,705 bird images and 18,688 non-bird images similarly collected to train ResNet.

3.2 Training

3.2.1 FCNs

We used an FCN-8s model [6] pretrained on PASCAL-Context [23], which contains 59 category (+ background) segmentations. We then fine-tuned the model with the images we prepared for training by using twofold cross validation.

3.2.2 DeepLab

We used an DeepLab-MSc-LargeFOV model [7] pretrained on PASCAL VOC 2012 [24], which contains 20 category (+ background) segmentations. We modified the layer "fc8" from 21 outputs to 4: bird, forest, sky, and wind turbine. As FCNs, we then fine-tuned the model with the prepared images by using twofold cross validation.

3.2.3 CNNs

We trained the ResNet-based model with eight-class training images from scratch. In the same way as [4], we used the method described in [25] for weight initialization. In addition, we used batch normalization [13] to reduce the internal covariate shift and accelerate learning.

3.2.4 Haar+AdaBoost

AdaBoost with Haar-like features was trained following [22]. Moving object regions were chosen by the inter-frame difference. Then, the proposed regions were marked with square bounding boxes and then trained the detector with the bird and non-bird labels.

3.2.5 SVMs

We combined the class likelihoods and scores by using pixel-wise SVM training and evaluated the performances of the individual methods and their combinations.

3.3 Evaluation

We used 44 of the 77 labeled images that included more birds (183 in total) than the others for the evaluation. The performance of the method is ranked by using the F-measure, i.e., the harmonic mean of precision and recall.

In the evaluation, we regarded detected bounding boxes that had any overlap with ground-truth boxes as correct detections and boxes with no overlap as misdetection.

Similarly, in segmentation-based methods, we regarded the outputs that had any region of overlap with the ground truth as correct detections and those without overlap as misdetections.

3.4 Results

We counted the true positives (TP) and false positives (FP) of birds and calculated the precision, recall, and F-measure. The results are summarized in Table 1.

AdaBoost with Haar-like features, SuperParsing, and DeepLab are denoted as HA, SP, and DL, respectively. In addition, SP* and FCN* represent the method combined with SVMs. Usually, SP or FCNs output class label with the highest likelihood, while SVMs consider all of the class likelihoods for the output through training.

The upper part of Table 1 shows the results of individual methods. SP and DL achieved the highest precision, while CNNs achieved the best recall rate. FCNs achieved the intermediate score between SP and CNNs. As expected, CNNs highly outperform HA. DL performed similarly to SP, but with much higher recall rate. SP* and FCN* performed better than the ones without SVMs.

The lower part of Table 1 shows the results of combination of methods, where most combinations exceed each single method in terms of F-measure. Particularly, combinations with DL have higher precision, suggesting that DL can suppress false positives because it can recognize backgrounds well. The CNN+FCN result shows FCNs also can recognize backgrounds. The CNN+DL did not achieve a good score in spite of the combination of the best detector and semantic segmentation, and it shows that FCN is also necessary for better performance. Figure 3 shows typical examples of detection results of each method. More results can be found in the Additional file 1.

To show the robustness of our method to the size of the bird images, Table 2 summarizes the results according to image size. The three image sizes, tiny ($\leq 15\times15$), small ($\leq 45\times45$), and normal ($> 45\times45$) are determined according to [26].

Table 2 F-measure of various methods by size

Size	Method	Precision	Recall	F-measure
Tiny	FCN+DL	0.333	0.149	0.029
	CNN+DL	0.432	0.239	0.308
	CNN+FCN	0.808	0.627	0.706
	CNN+FCN+DL	*0.915*	*0.642*	*0.754*
Small	FCN+DL	*1.000*	0.738	0.849
	CNN+DL	0.844	0.813	0.828
	CNN+FCN	0.972	*0.863*	0.914
	CNN+FCN+DL	*1.000*	*0.863*	*0.926*
Normal	FCN+DL	*1.000*	0.890	0.941
	CNN+DL	*1.000*	*0.972*	*0.986*
	CNN+FCN	*1.000*	*0.972*	*0.986*
	CNN+FCN+DL	*1.000*	*0.972*	*0.986*

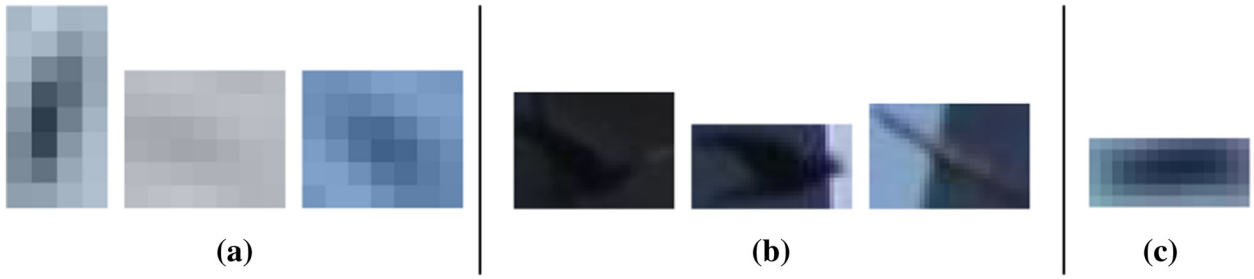

Fig. 4 Examples of bird images which could not be detected by the proposed method: **a** blurred shape, **b** overlap with the wind turbine, and **c** part occlusions

In all image sizes, the proposed method produces the best F-measure. DL is not suited for detecting tiny images of birds, but CNN+FCN detects tiny bird images more effectively. With DL, the performance is more improved particularly in precision. This shows that FCN detects more birds and DL is good at backgrounds.

Regarding the region proposals obtained by background subtraction, the number of them was about 1000 to 2000 per an input image. As shown in the Fig. 2, almost all the region proposals belong to the forest class. CNN succeeded to filter most of them and contributed to precision.

To clarify the limitation, we analyzed bird images which could not be detected by the proposed method, as shown in Fig. 4. Overlooked bird images were classified into three patterns: blurred shape due to extremely low resolution, overlap with other objects (e.g., wind turbine), and part occlusions (e.g., a bird is at the end of the image).

Almost all images with ambiguous shape were either only detected by CNN or not detected by any methods. In detail, FCN and DeepLab showed too weak reaction to very small birds to detect them. A few bird images over the wind turbine were detected by FCN and DeepLab, but when combined with CNN, they were missed because of low likelihood of birds. There was only one bird image whose parts were occluded; thus, it was hard to train such pattern of bird images.

4 Conclusion

We combined different types of deep features from a CNN-based detector and fully convolutional networks by using support vector machines to achieve high performance in detecting objects at various scale in large images.

Experiments on a bird image dataset intended for ecological investigations showed that our method detects birds with high precision.

We showed combination of multiple deep convolutional features are effective for scale-aware detection.

Additional file

Additional file 1: More experimental results including extreme cases. (PDF 854 kb)

Competing interests
The authors declare that they have no competing interests.

Authors' contributions
AT designed and executed the experiments and wrote the manuscript. TT contributed to the concept and wrote the manuscript. RY collected the experimental data, helped the implementation, and wrote the manuscript. RK advised AT on the concept and experiments and wrote the manuscript. MI and TN supervised the work and edited the manuscript. All authors reviewed and approved the final manuscript.

Acknowledgements
This work is in part entrusted by the Ministry of the Environment, JAPAN (MOEJ), the project of which is to examine effective measures for preventing birds, especially sea eagles, from colliding with wind turbines, and by JSPS KAKENHI Grant Number JP16K16083.

Received: 21 April 2016 Accepted: 23 June 2016
Published online: 02 August 2016

References

1. Smallwood KS, Rugge L, Morrison ML (2009) Influence of behavior on bird mortality in wind energy developments. J Wildl Manage 73(7):1082–1098
2. Bassi S, Bowen A, Fankhauser S (2012) The case for and against onshore wind energy in the UK. Grantham Res. Inst. on Climate Change and Env. Policy Brief
3. Tighe J, Lazebnik S (2013) Finding things: image parsing with regions and per-exemplar detectors. In: CVPR. IEEE. pp 3001–3008
4. He K, Zhang X, Ren S, Sun J (2016) Deep residual learning for image recognition. In: Proc. of Computer Vision and Pattern Recognition. IEEE. pp 770–778
5. Krizhevsky A, Sutskever I, Hinton GE (2012) Imagenet classification with deep convolutional neural networks. In: NIPS. pp 1097–1105
6. Long J, Shelhamer E, Darrell T (2015) Fully convolutional networks for semantic segmentation. In: CVPR. IEEE
7. Chen LC, Papandreou G, Kokkinos I, Murphy K, Yuille AL (2015) Semantic image segmentation with deep convolutional nets and fully connected CRFs. In: ICLR. http://arxiv.org/abs/1412.7062
8. Cortes C, Vapnik V (1995) Support-vector networks. Mach Learn 20(3):421–436
9. Takeki A, Tuan Trinh T, Yoshihashi R, Kawakami R, Iida M, Naemura T (2016) Detection of small birds in large images by combining a deep detector with semantic segmentation. In: ICIP. IEEE
10. Simonyan K, Zisserman A (2015) Very deep convolutional networks for large-scale image recognition. In: ICLR. IEEE

11. Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, Erhan D, Vanhoucke V, Rabinovich A (2015) Going deeper with convolutions. In: CVPR. IEEE
12. Srivastava N, Hinton G, Krizhevsky A, Sutskever I, Salakhutdinov R (2014) Dropout: a simple way to prevent neural networks from overfitting. JMLR 15(1):1929–1958
13. Ioffe S, Szegedy C (2015) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: ICML
14. Girshick R, Donahue J, Darrell T, Malik J (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR
15. Uijlings JR, van de Sande KE, Gevers T, Smeulders AW (2013) Select search object recognition. IJCV 104(2):154–171
16. Dong J, Chen Q, Yan S, Yuille A (2014) Towards unified object detection and semantic segmentation. In: ECCV. Springer. pp 299–314
17. Hariharan B, Arbeláez P, Girshick R, Malik J (2015) Hypercolumns for object segmentation and fine-grained localization. In: CVPR
18. Mallat S (1999) A wavelet tour of signal processing. Academic press
19. Viola P, Jones M (2001) Rapid object detection using a boosted cascade of simple features. In: CVPR. IEEE Vol. 1. p 511
20. Freund Y, Schapire RE (1997) A decision-theoretic generalization of on-line learning and an application to boosting. J Comput Syst Sci 55(1):119–139
21. Tighe J, Lazebnik S (2013) Superparsing. IJCV 101(2):329–349
22. Yoshihashi R, Kawakami R, Iida M, Naemura T (2015) Construction of a bird image dataset for ecological investigations. In: ICIP. IEEE. pp 4248–4252
23. Mottaghi R, Chen X, Liu X, Cho NG, Lee SW, Fidler S, Urtasun R, Yuille A (2014) The role of context for object detection and semantic segmentation in the wild. In: CVPR. IEEE. pp 891–898
24. Everingham M, Van Gool L, Williams CK, Winn J, Zisserman A (2010) Pascal vis object class (VOC) challenge. IJCV 88(2):303–338
25. He K, Zhang X, Ren S, Sun J (2015) Delving deep into rectifiers: surpassing human-level performance on imagenet classification. In: ICCV. IEEE
26. Pepik B, Benenson R, Ritschel T, Schiele B (2015) What is holding back convnets for detection? In: Patt. Recog. Springer. pp 517–528