[Pattern Recognition 76 (2018) 704–714](https://doi.org/10.1016/j.patcog.2017.10.002)

Contents lists available at [ScienceDirect](http://www.ScienceDirect.com/)

Pattern Recognition

journal homepage: [www.elsevier.com/locate/patcog](http://www.elsevier.com/locate/patcog)

Mask-CNN: Localizing parts and selecting descriptors for ﬁne-grained bird species categorization



Xiu-Shen Wei [a,](#_bookmark0)[1,](#_bookmark2) Chen-Wei Xie [a,](#_bookmark0)[1,](#_bookmark2) Jianxin Wu [a,](#_bookmark0)[∗,](#_bookmark3) Chunhua Shen [b](#_bookmark1)

a *National Key Laboratory for Novel Software Technology, Nanjing University, China*

b *The University of Adelaide, Adelaide, Australia*

## a r t i c l e i n f o a b s t r a c t

*Article history:*

Received 17 June 2017

Revised 3 September 2017

Accepted 6 October 2017

Available online 13 October 2017

*Keywords:*

Fine-grained image recognition Deep descriptor selection

Part localization

Fine-grained image recognition is a challenging computer vision problem, due to the small inter-class variations caused by highly similar subordinate categories, and the large intra-class variations in poses, scales and rotations. In this paper, we prove that selecting useful deep descriptors contributes well to ﬁne-grained image recognition. Speciﬁcally, a novel Mask-CNN model without the fully connected layers is proposed. Based on the part annotations, the proposed model consists of a fully convolutional network to both locate the discriminative parts (*e.g.*, head and torso), and more importantly generate weighted ob- ject/part masks for selecting useful and meaningful convolutional descriptors. After that, a three-stream Mask-CNN model is built for aggregating the selected object- and part-level descriptors simultaneously. Thanks to discarding the parameter redundant fully connected layers, our Mask-CNN has a small feature dimensionality and eﬃcient inference speed by comparing with other ﬁne-grained approaches. Further- more, we obtain a new state-of-the-art accuracy on two challenging ﬁne-grained bird species categoriza- tion datasets, which validates the effectiveness of both the descriptor selection scheme and the proposed Mask-CNN model.

© 2017 Elsevier Ltd. All rights reserved.

### Introduction

Fine-grained recognition tasks such as identifying the species of birds [[1,2]](#_bookmark29), ﬂowers [[3,4]](#_bookmark30) and cars [[5]](#_bookmark31), have been popular in appli- cations of computer vision and pattern recognition. Since the cate- gories are all similar to each other, different categories can only be distinguished by slight and subtle differences, which makes ﬁne-grained recognition a challenging problem. Compared to the general object recognition tasks, ﬁne-grained recognition beneﬁts more from learning critical parts of the objects, which helps dis- criminate different subclasses and align objects of the same class [[6–13]](#_bookmark33).

A straightforward way to represent parts is to use the deep con- volutional features/descriptors. The convolutional descriptors con- tain more localized (*i.e.*, parts) information compared to the fea- ture of the fully connected layers (*i.e.*, whole image). In addition, these deep descriptors are known to correspond to mid-level in-

formation, *e.g.*, object parts [[14]](#_bookmark38). All the previous part-based ﬁne- grained approaches, *e.g.*, [[7,8,10,11]](#_bookmark34), directly used the deep convo- lutional descriptors and encoded them into a single representa-

∗ Corresponding author.

*E-mail address:* [wujx2001@nju.edu.cn](mailto:wujx2001@nju.edu.cn) (J. Wu).

1 The ﬁrst two authors contributed equally to this work.

tion, without evaluating the usefulness of the obtained object/part deep descriptors. By using powerful convolutional neural networks [[15]](#_bookmark39), we may not need to select useful dimensions inside feature vectors, as what we do for hand-crafted features [[16,17]](#_bookmark40). However, since most deep descriptors are not useful or meaningful for ﬁne- grained recognition, it is necessary to select useful deep convo- lutional descriptors. Recently, selecting deep descriptors sheds its light on the ﬁne-grained image retrieval task [[18]](#_bookmark42). Moreover, it is also beneﬁcial to ﬁne-grained image recognition.

In this paper, by developing a novel deep part detection and descriptor selection scheme, we propose an end-to-end Mask-CNN (M-CNN) model which discards the fully connected layers for ﬁne- grained bird species categorization. We only require the part an- notations and image-level labels during the training time. In M- CNN, given the part annotations, we ﬁrstly separate them into two point sets. One set corresponds to the head part of the ﬁne- grained bird image, and the other is for the torso. Then, the small- est rectangles that cover each point set are returned as the ground- truth mask, as shown in [Fig. 1](#_bookmark8). The other pixels are background. By treating part localization as a three-class segmentation task, we leverage fully convolutional networks (FCN) [[19]](#_bookmark43) to generate weighted masks in the testing time for both localizing parts and selecting useful deep descriptors, *which does not use any annota-*

*tion during testing*. After getting these two part masks, the seg-

<https://doi.org/10.1016/j.patcog.2017.10.002>

0031-3203/© 2017 Elsevier Ltd. All rights reserved.

*X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714* 705

**Table 1**

Comparison of classiﬁcation accuracy on *CUB200-2011* with state-of-the-art methods. Note that, “Model” describes the deep models used in these methods. The inference speeds (frames/sec) of M-CNNs on a K40 GPU are also reported.

Method Train phase Test phase Model Dim. Acc.

Part-Stacked CNN [[7]](#_bookmark34) ✓ ✓ ✓ Part-Stacked CNN × 1 4,096 76.6%

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | BBox | Parts |  | BBox Parts |  | | |
| PB R-CNN with BBox [[10]](#_bookmark35) | ✓ | ✓ |  | ✓ | Alex-Net × 3 | 12,288 | 76.4% |

Deep LAC [[8]](#_bookmark36) ✓ ✓ ✓ Alex-Net × 3 12,288 80.3%

PB R-CNN [[10]](#_bookmark35) ✓ ✓ Alex-Net × 3 12,288 73.9%

Pose Normalized CNNs [[30]](#_bookmark54) ✓ ✓ Alex-Net × 3 13,512 75.7%

MixDCNN [[45]](#_bookmark52) ✓ GoogLeNet × 1 6144 81.1%

Co-Segmentation [[38]](#_bookmark46) ✓ VGG-19 × 2 126,976 82.0%

Multi-grained [[46]](#_bookmark53) ✓ VGG-19 × 3 12,288 83.0%

Two-Level [[34]](#_bookmark41) VGG-16 × 1 16,384 77.9%

Weakly supervised FG [[11]](#_bookmark37) VGG-16 × 1 262,144 79.3%

Constellations [[33]](#_bookmark57) VGG-19 × 1 208,896 81.0%

Multi-grained [[46]](#_bookmark53) VGG-19 × 3 12,288 81.7%

Bilinear [[27]](#_bookmark51) VGG-16 and VGG-M 262,144 84.1%

Spatial Transformer CNN [[26]](#_bookmark50) ST-CNN (inception) × 4 4,096 84.1%

PDFS [[35]](#_bookmark44) VGG-16 × 1 69,632 84.5%

Our 3-stream M-CNN (Alex-Net)[a](#_bookmark5) ✓ Alex-Net (w/o FCs) × 3 **1,536 78.6%**

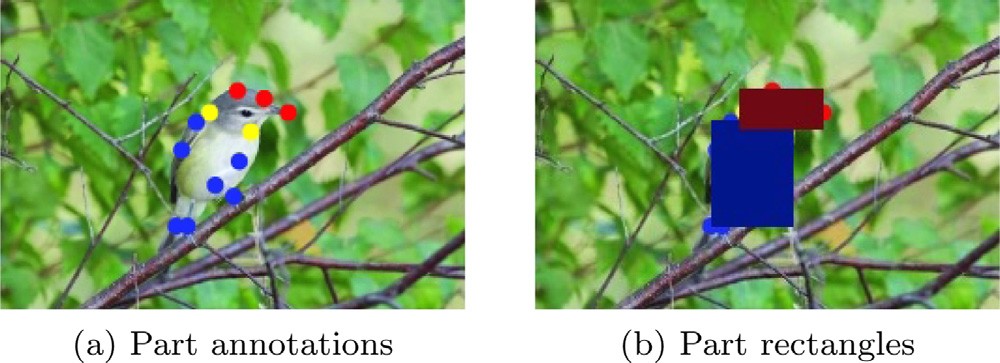
Our 3-stream M-CNN (VGG-16)[b](#_bookmark6) ✓ VGG-16 (w/o FCs) × 3 **3,072 85.7%**

Our 3-stream M-CNN (ResNet-50)[c](#_bookmark7) ✓ ResNet-50 × 3 **12,288 87.3%**

a The inference speed of M-CNN (Alex-Net) is 33.9 frames/sec.

b The inference speed of M-CNN (VGG-16) is 8.3 frames/sec.

c The inference speed of M-CNN (ResNet) is 11.8 frames/sec. (All the speeds here contain both part masks prediction and the ﬁnal classiﬁcation.)



**Fig. 1.** We generate the rectangles (in [Fig. 1](#_bookmark8)b) for the bird’s head and torso based on the part annotations (red, blue and yellow dots in [Fig. 1](#_bookmark8)a). Other pixels are treated as background. The two yellow part key points (*i.e.*, nape and throat) are included in both head and torso. (Best if viewed in color.). (For interpretation of the references to color in this ﬁgure legend, the reader is referred to the web ver- sion of this article.)

mentation class scores are treated as the automatically learned weights for aggregating descriptors. Meanwhile, we combine these two part masks to form the weighted object mask. Based on these object/part masks, a three-stream Mask-CNN (image, head, torso) is built for joint training and aggregating the object-level and part- level cues simultaneously. The architecture of the proposed three- stream M-CNN is shown in [Fig. 2](#_bookmark11). In each stream of M-CNN, we discard the fully connected layers. In the last convolutional layer,  an input image is represented by multiple deep descriptors. In or- der to select useful descriptors to keep only those corresponding to the object, the pre-learned object/part masks by FCN are used. Af- ter that, the selected descriptors of each stream are both average and max pooled into two 512-d feature vectors, respectively. The standard *£*2 -normalization is followed. Finally, the feature vectors of these three streams are concatenated, and then a classiﬁcation (fc+softmax) layer is added for end-to-end joint training.

We validate the proposed three-stream M-CNN on two bench- mark ﬁne-grained image recognition datasets, *i.e.*, the Caltech- UCSD Birds (CUB) 200–2011 [[1]](#_bookmark29) and *Birdsnap* [[20]](#_bookmark45) dataset. On *CUB200-2011*, we achieved 85.7% classiﬁcation accuracy based on VGG models [[21]](#_bookmark46) and 87.3% on Residual Nets [[22]](#_bookmark48). On *Bird- snap*, our proposed M-CNN obtained 77.3% accuracy based on VGG

* 1. and 80.2% based on Residual Nets [[22]](#_bookmark48). The classiﬁcation ac- curacy of our M-CNN is new state-of-the-art on both two ﬁne-

grained datasets. Moreover, we also get accurate part localization (cf. [Section 4.3](#_bookmark22)). The key advantages and major contributions of the proposed M-CNN model are:

* + - To the best of our knowledge, Mask-CNN is the ﬁrst model that selects deep convolutional descriptors for object recogni- tion, especially for ﬁne-grained image recognition.
    - We present a novel and eﬃcient part-based three-stream model for ﬁne-grained recognition. By discarding the fully connected layers, the proposed M-CNN is computationally eﬃcient (cf. [Table 1](#_bookmark4) and [Table 4](#_bookmark23)). Additionally, comparing with state-of-the- art methods, M-CNN has smaller feature dimensionality. Be- yond those, it achieves the highest classiﬁcation accuracy on *CUB200-2011* and *Birdsnap* among published methods.[2](#_bookmark10)
    - The part localization performance of the proposed model out-

performs other part-based ﬁne-grained approaches which re- quires additional bounding boxes. In particular, M-CNN is 12.76% higher than state-of-the-art for head localization on *CUB200-2011*.

The rest of the paper is organized as follows. [Section 2](#_bookmark9) summa- rizes related work. The proposed Mask-CNN model including the object/part masks learning procedure and the classiﬁcation train- ing process is described in [Section 3](#_bookmark13). Detailed performance studies and analyses are conducted in [Section 4](#_bookmark18). [Section 5](#_bookmark25) concludes the paper.

### Related work

In this section, we ﬁrst review ﬁne-grained image recognition, and then, give a brief recap about the researches of deep descriptor selection.

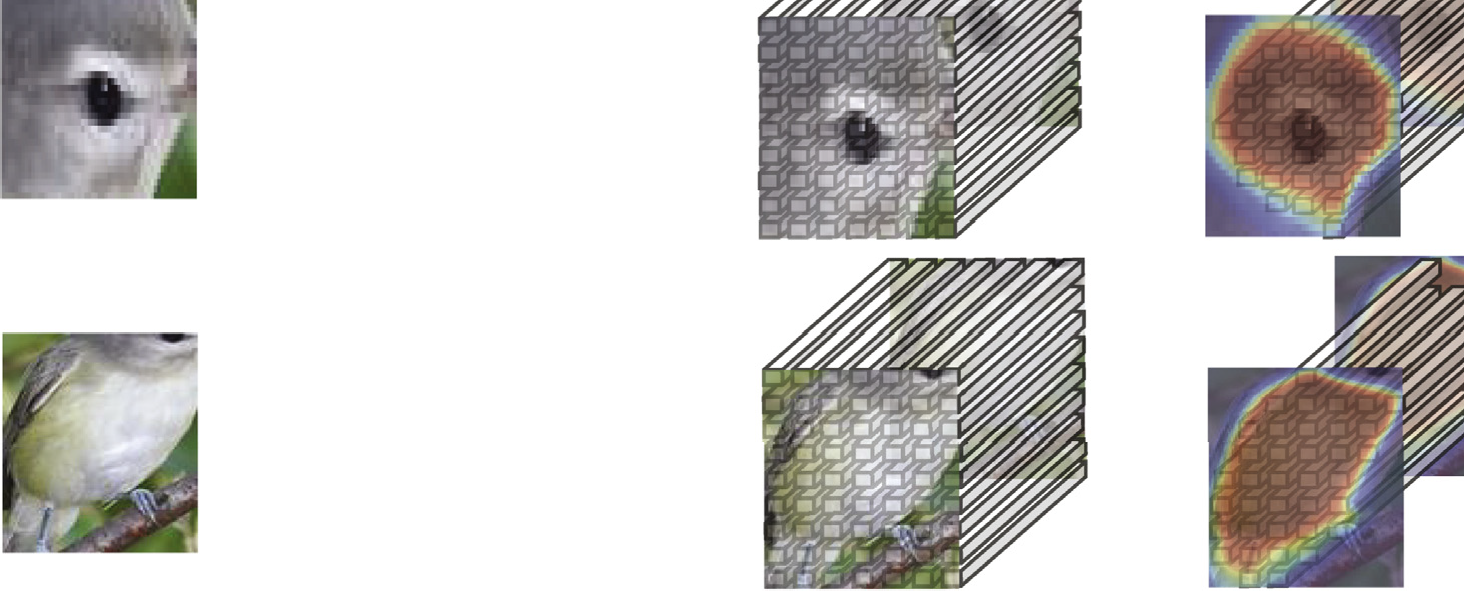
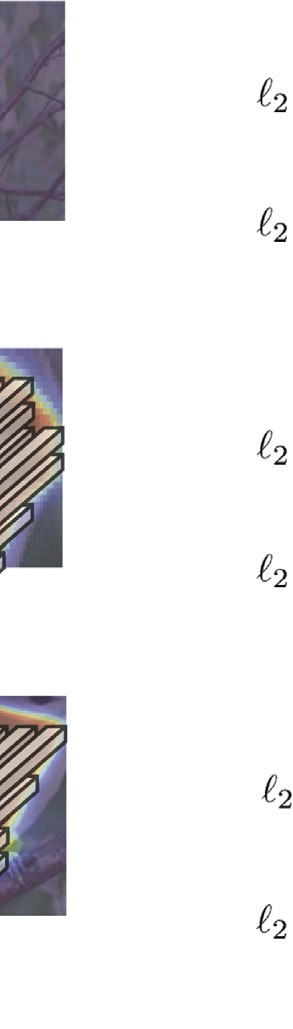
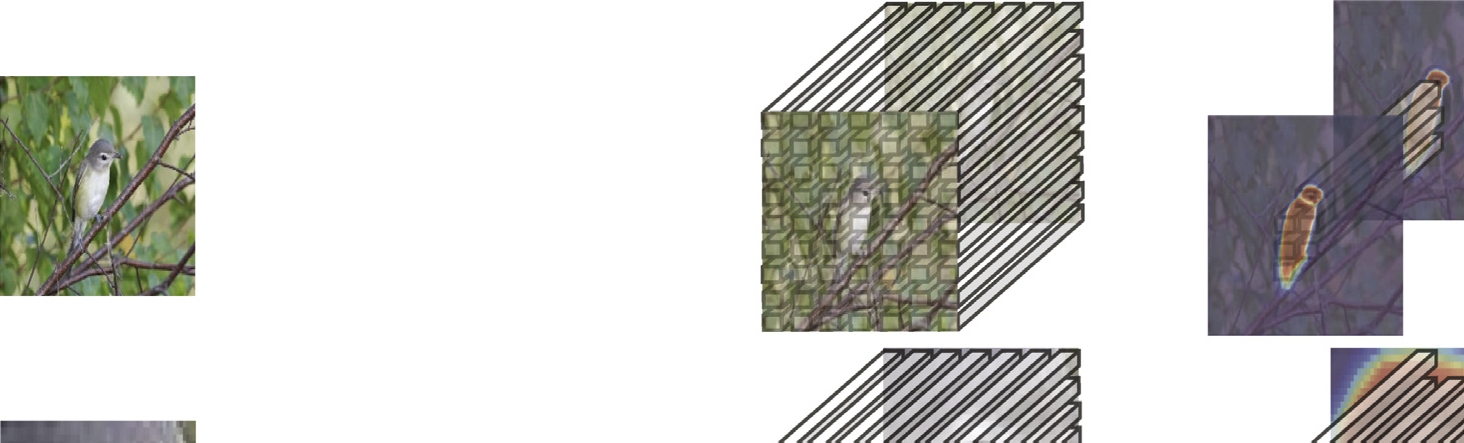
* 1. *Fine-grained image recognition*

Fine-grained recognition is a challenging problem and has re- cently emerged as a hot topic [[1,3,5,23–25]](#_bookmark29). During the past few years, a number of effective ﬁne-grained recognition methods have

2 In this comparison, we do not consider methods that use large amounts of ex- ternal images collected from the web.

706 *X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714*

1024-d



3072-d

conv1\_1 relu1\_1

...

pool1

...

conv5\_3 relu5\_3 pool5

...

conv5\_3 relu5\_3 pool5

Avg Pool Max Pool

|  |
| --- |
| Image feature |
| Head feature |
| Torso feature |

Whole image 448×448

conv1\_1 relu1\_1

...

pool1

Head 224×224

1024-d

1024-d

Avg Pool Max Pool

Avg Pool Max Pool

K

# (f) Classification

fc+softmax

Torso 224×224

conv1\_1 relu1\_1

...

pool1

...

conv5\_3 relu5\_3 pool5

# Inputs

1. CNN w/o FCs
2. Convolutional activation tensor
3. Descriptor selection
4. Weighted aggregation and concatenation

**Fig. 2.** Architecture of the proposed three-stream Mask-CNN model. The three streams correspond to the whole image, head and torso image patches, respectively. In each stream, we employ the learned part/object masks to select the useful deep descriptors, and then aggregate these selected descriptors by weights (presented by different colors in Fig. 2d) to form the ﬁnal image representation. As shown, thanks to the descriptor selection scheme, a large number of descriptors corresponding to background can be discarded by M-CNN, which is beneﬁcial to ﬁne-grained recognition. (This ﬁgure is best viewed in color.)

been developed in the literature [[7,8,10–13,26–29]](#_bookmark34). We can roughly categorize these methods into three groups. The ﬁrst group, *e.g.*, [[26,27]](#_bookmark50), attempted to learn a more discriminative feature repre- sentation by developing powerful deep models for classifying ﬁne- grained images. The second group aligned the objects in ﬁne- grained images to eliminate pose variations and the inﬂuence of camera position, *e.g.*, [[8,30,31]](#_bookmark36). The last group focused on part- based representations, because it is widely acknowledged that the subtle difference between ﬁne-grained images mostly resides in the unique properties of object parts.

For the part-based ﬁne-grained recognition methods, [[8,10,32]](#_bookmark36) used both bounding boxes of the birds and part an- notations during training to learn an accurate part localization model. Then, based on these detected parts, different CNNs are ﬁne-tuned using the detected parts separately. To ensure satisfac- tory localization results, they even used bounding boxes in the testing phase. In contrast, our method only need part annotations for training, and do not need any supervision during testing. Moreover, our three-stream M-CNN is a uniﬁed framework for capturing object- and part-level information simultaneously.

Some other part-based methods considered a weakly super- vised setting, in which they categorize ﬁne-grained images with only image-level labels, *e.g.*, [[11,33–35]](#_bookmark37). As will be shown by our experiments, classiﬁcation accuracy of M-CNN is signiﬁcantly higher than these weakly supervised methods. Meanwhile, M-CNN discards the parameter redundant fully connected layers, which makes it eﬃcient to train/inference. Besides, the dimensionality of image representations in M-CNN is quite low, cf. [Table 1](#_bookmark4). Therefore, M-CNN can be scalable to large-scale ﬁne-grained datasets.

Moreover, these part-based methods, *e.g.*, [[10,11,33–36]](#_bookmark35), usually require to ﬁrstly produce object/part proposals by selective search [[37]](#_bookmark47). By comparing with that, the proposed M-CNN is more con-

cise, which can accurately localize ﬁne-grained parts *without utiliz- ing bounding boxes and redundant object proposals*.

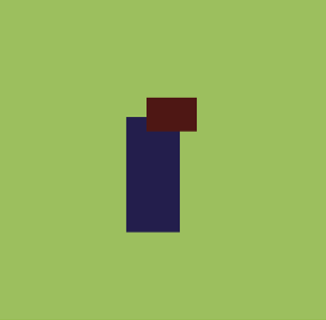
In addition, there are also ﬁne-grained recognition methods based on segmentation, *e.g.*, [[7,38]](#_bookmark34). The most signiﬁcant difference between them and M-CNN is: these methods only use segmenta- tion to localize the whole object [[38]](#_bookmark46) or parts [[7]](#_bookmark34), while we fur- ther select useful deep convolutional descriptors using the masks obtained from segmentation. Among them, the part-stacked CNN model [[7]](#_bookmark34) is the most related work to ours. In [[7]](#_bookmark34), part-stacked CNN requires both bounding boxes and part annotations in train- ing, and even needed the bounding boxes during testing. Within the image patch cropped using the bounding box, [[7]](#_bookmark34) treated the image crop around each of the ﬁfteen part key points as 15 seg- mentation foreground classes, and used FCN to solve the 16-classes segmentation task. After obtaining the trained FCN, it localized these part point positions in the last convolutional layer. Then, deep activations corresponding to the ﬁfteen parts and the whole object were stacked together. Fully connected layers were used for classiﬁcation. Comparing with part-stacked CNN, M-CNN only needs to localize two main parts (head and torso), which makes the segmentation problem much easier and more accurate. M-CNN achieves high localization accuracy, as will be shown in [Table 3](#_bookmark20). Meanwhile, as demonstrated in [[7]](#_bookmark34), using all the ﬁfteen part ac- tivations cannot lead to better classiﬁcation accuracy. Besides, M- CNN’s accuracy on *CUB200-2011* is 2.0% higher than that of [[7]](#_bookmark34) us- ing the same baseline network, although we use less annotations in training and do not use any annotation in testing. More detailed empirical comparisons can be found in [Section 4.2](#_bookmark21).

* 1. *Deep descriptor selection*

As aforementioned, in the deep learning scenario, we might no longer need to select useful dimensions inside the learnt deep

*X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714* 707

## The input (b) FCN 3 (c) The mask g.t.



3

96

**Fig. 3.** Demonstration of the mask learning procedure by fully convolutional network (FCN) [[19]](#_bookmark43). (Best viewed if in color.)

features. While, useful deep descriptors are necessary to be se- lected and noisy descriptors should be discarded, especially for ﬁne-grained images. The so called “descriptor” here indicates the *d*-dimensional component vector of activations in a convolutional layer.

In the line of deep descriptor selection for ﬁne-grained im- ages, SCDA [[18]](#_bookmark42) (Selective Convolutional Descriptor Aggregation) was proposed recently for dealing with the ﬁne-grained image *re- trieval* problem. In SCDA, it employs pre-trained models to ﬁrst lo- calize the main object in ﬁne-grained images unsupervisedly. Then, based on the results of localization, it treats these deep descriptors

corresponding to the object localization as the useful descriptors, and regards the others as background and noises. Thanks to the descriptor selection scheme, SCDA achieves the best retrieval per- formance in the content-based ﬁne-grained image retrieval task. Comparing with SCDA, our proposed method can not only localize whole objects, but also localize ﬁne-grained parts in a supervised manner, which achieves much more accurate part localization per- formance. Besides, our M-CNN is *the ﬁrst work* to demonstrate that

deep descriptor selection is beneﬁcial to ﬁne-grained image *recog-*

*nition*.

### The Mask-CNN model

In this section, we present the proposed three-stream Mask- CNN (M-CNN) model. Firstly, we adopt a fully convolutional net- work (FCN) [[19]](#_bookmark43) to generate the object/part masks for locating ob- ject/parts, and more importantly selecting deep descriptors. Then, based on these masks, the three-stream M-CNN is built for joint training and capturing both object- and part-level information.

* 1. *Learning object and part masks*

The fully convolutional network (FCN) [[19]](#_bookmark43) is designed for pixel- wise labeling. FCN can take an input image with any resolution and produce an output of the same size. In our method, we use FCN to not only localize the object and parts in ﬁne-grained images, but also treat the segmentation predictions as the object and parts masks for the later descriptor selection process.

Each ﬁne-grained image in the *CUB200-2011* [[1]](#_bookmark29) and *Birdsnap*

[[20]](#_bookmark45) dataset is equipped with part annotations. *CUB200-2011* has ﬁfteen part key points for each image, and *Birdsnap* has seven- teen part key points for each. While, the other ﬁne-grained image datasets (*e.g.*, [[4,5,23]](#_bookmark32)) have no such part annotations. As shown in [Fig. 1](#_bookmark8), we split these key points into two sets, including the head key points (*i.e.*, the beak, forehead, crown, left eye, right eye, nape and throat for *CUB200-2011*; the beak, forehead, crown, left eye, right eye, left cheek, right cheek, nape and throat for *Birdsnap*) and torso key points (*i.e.*, the back, breast, belly, left leg, right leg, left wing, nape, right wing, tail and throat for both *CUB200-2011* and *Birdsnap*). Based on the key points, two ground-truth of part masks are generated. One is the *head mask*, which corresponds to

the smallest rectangle covering all the head key points. The other is the *torso mask*, which is the smallest rectangle covering all the torso key points. The overlapping part of the two rectangles is re- garded as the head mask. As shown in [Fig. 1](#_bookmark8)b, the red rectangle is the head mask, and the blue one is for torso. The rest of the image is background. Similar to [[39,40]](#_bookmark48), these bounding-box-like part masks are treated as the segmentation ground-truth. Thus, we model the part mask learning procedure as a three-class segmen- tation problem. For effective training, all the training and testing ﬁne-grained images remain at their original resolutions. Then, we crop a 384 × 384 image patch in the middle of the original image as the inputs to FCN. The mask learning network architecture is shown in [Fig. 3](#_bookmark12). In our experiments, we adopted FCN-8s [[19]](#_bookmark43) for learning and predicting part masks.

During the FCN inference, without using any annotation, three class heat maps (in the same size as the original input image) are returned for every image. Moreover, the predicted segmenta- tion class scores are regarded as the learned part weights for the later descriptors aggregation process. We randomly choose some qualitative examples of the predicted part masks, and show them in [Fig. 4](#_bookmark16). In these ﬁgures, the learned masks are overlaid onto the original images. The head part is highlighted in red, and the torso is in blue. The predicted background pixels are in black. As can be seen from these ﬁgures, even though the ground-truth part masks are not very accurate, the learned FCN model is able to re- turn more accurate part masks. Meanwhile, these part masks can also localize the part positions by ﬁnding their enclosing rectan- gles. Moreover, comparing with the segmentation ground-truth (in the third row of [Fig. 4](#_bookmark16)), head masks combining with torso masks can be generally able to segment the foreground object well, even though no post-processing (*e.g.*, conditional random ﬁelds [[41,42]](#_bookmark49)) is used. Quantitative results of part localization and object segmen- tation will be reported in [Section 4.3](#_bookmark22) and [Section 4.4](#_bookmark24), respectively. Also, there are several failure cases, *e.g.*, the ﬁgures shown in the right side of [Fig. 4](#_bookmark16). In some cases, it will treat the branch as the bird’s torso. Some ones will also detect the head’s and torso’s reﬂections in water. In other cases, due to the scale of the main object or the complicated background, the torso masks can not be

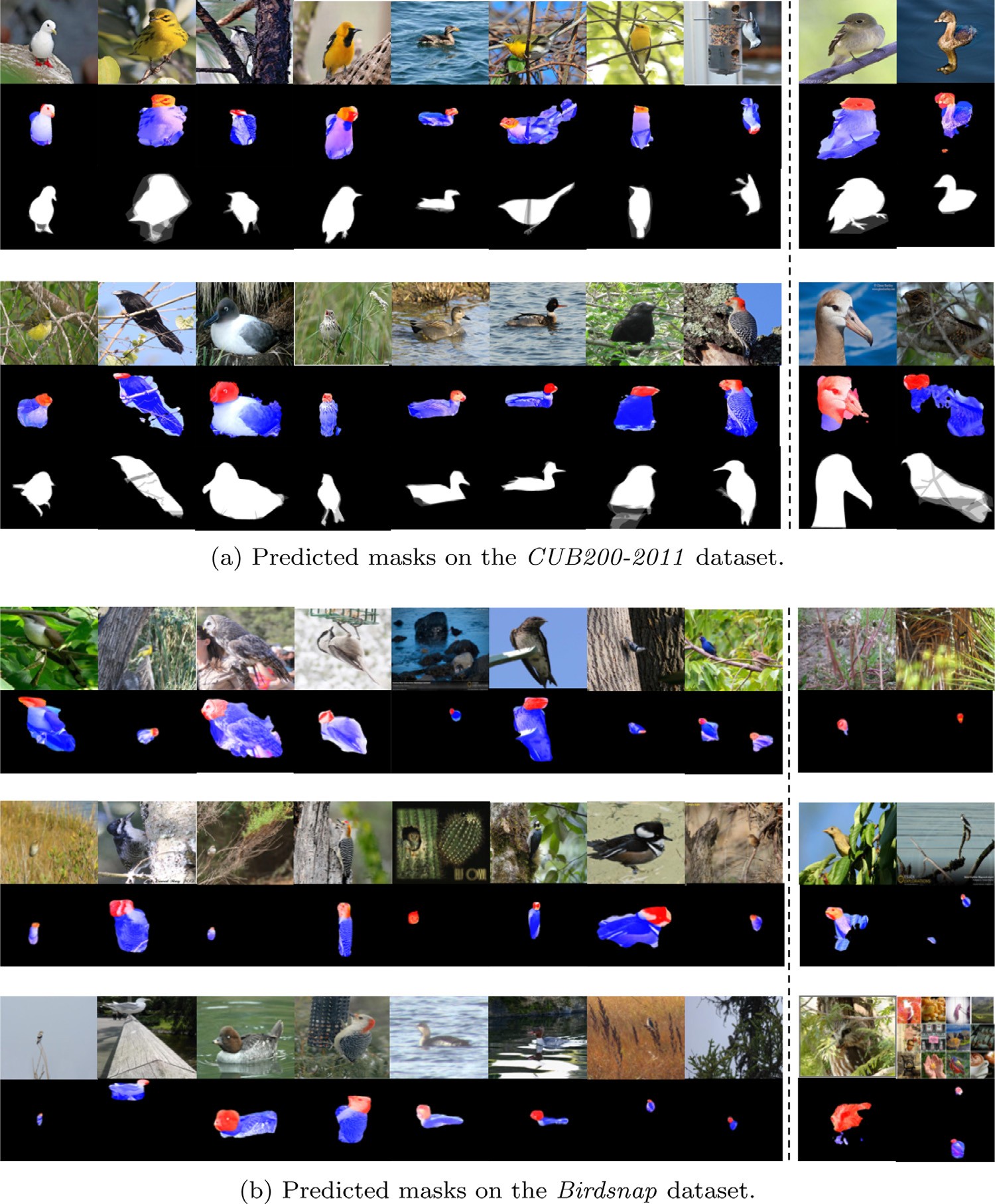
intactly predicted.

For the ﬁnal recognition performance, both part masks, if ac- curately predicted, will beneﬁt the later deep descriptor selection process and the ﬁnal ﬁne-grained classiﬁcation. Therefore, during both the training and testing phases, we will use the predicted masks for both part localization and descriptor selection in M-CNN. We also combine the two masks to form a mask for the whole ob- ject, which is called the *object mask*.

* 1. *Training Mask-CNN*

After obtaining the object and part masks, we build the three- stream M-CNN for joint training. The overall architecture of the

708 *X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714*



**Fig. 4.** Random samples of successfully predicted part masks (on the left side) and four failure cases (on the right side) from the testing set on the *CUB200-2011* [[1]](#_bookmark29) and *Birdsnap* [[20]](#_bookmark45) dataset, respectively. The ﬁrst row of each subﬁgure contains input ﬁne-grained images. The second row are the part masks predictions. In these ﬁgures, we overlay the part mask predicted by FCN (the head highlighted in red and the torso in blue) onto the original images. The pixels predicted as background are in black. The third row in (a) is the corresponding segmentation ground-truth provided in the *CUB200-2011* dataset. The *Birdsnap* dataset does not supply the segmentation ground-truth for its ﬁne-grained images. (The ﬁgures are best viewed in color.)

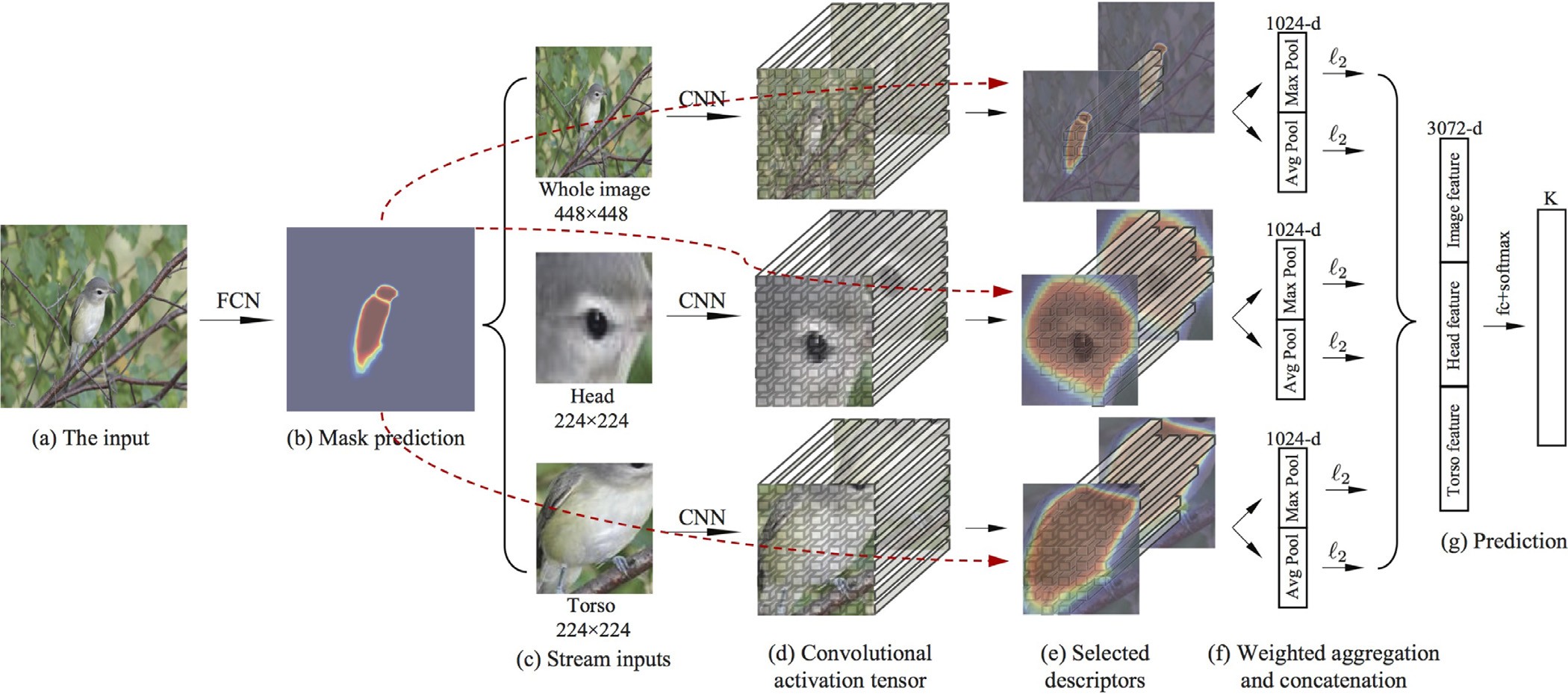
proposed model is presented in [Fig. 2](#_bookmark11). We take the whole image stream as an example to illustrate the pipeline of each stream.

The inputs of the whole image stream are the original im- ages resized to *h* × *h*. In our experiments, we report the results for *h* = 224 and *h* = 448*,* respectively. The input images are fed into a traditional convolutional neural network, but the fully connected

layers are discarded. That is to say, the CNN model used in our pro- posed M-CNN only contains convolutional, ReLU and pooling lay- ers, which greatly brings down the M-CNN model size. Speciﬁcally, we use VGG-16 [[21]](#_bookmark46) as the baseline model, and the layers before pool5 are kept (including pool5 ). We obtain a 7 × 7 × 512 activation tensor in pool5 if the input image is 224 × 224. Therefore, we have 49 deep convolutional descriptors of 512-d, which also correspond to 7 × 7 spatial positions in the input images. Then, the learned ob- ject mask (cf. [Section 3.1](#_bookmark14)) is ﬁrstly resized to 7 × 7 by the bilinear interpolation, and then used for selecting useful and meaningful deep descriptors.

As illustrated in [Fig. 2](#_bookmark11)c and [Fig. 2](#_bookmark11)d, the descriptor should be kept by weights when it locates in the object region. If it locates in the background region, that descriptor will be discarded. In our im- plementation, the mask contains the learned part/object segmenta- tion scores, which is a real matrix whose elements are in the range of [0, 1]. Correspondingly, 1 stands for absolutely keeping and 0 is for absolutely discarding. We implement the selection process as an element-wise product operation between the convolutional ac- tivation tensor and the mask matrix. Therefore, the descriptors lo- cated in the object region will remain by weights, while the other descriptors will become zero vectors. Concretely, if the pixels are predicted as head/torso by FCN, the real values of the mask are kept. Otherwise, if the pixels indicate the regions are background, the value of these background regions in the mask are reset to the zero value. Then, the processed masks are used for selecting de- scriptors and the rest processing.

*X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714* 709



**Fig. 5.** Testing stage of Mask-CNN. (Best viewed if in color.)

For these selected descriptors, in the end-to-end M-CNN learn- ing process, we both average and max pool them into two 512-d feature vectors, respectively. Then, the *£*2 -normalization is followed for each of them. After that, we concatenate them into an 1024-d feature as the ﬁnal representation of the whole image stream.

The streams for head and torso have similar processing steps as the whole image one. However, different from the inputs of the whole image stream, we generate the input images of the head and torso streams as follows. After obtaining the two part masks, we use the part masks as the part detectors to localize the head part and torso part in the input images. For each part, we return the smallest rectangle bounding box which contains the part mask regions. Based on the rectangle bounding box, we crop the image patch which acts as the inputs of the part stream. The last two

streams of [Fig. 2](#_bookmark11) show the head and torso streams in M-CNN. The inputs of these two streams are all resized into 224 × 224 in our experiments.

In the classiﬁcation step shown in [Fig. 2](#_bookmark11)f, the ﬁnal 3,072-d im- age representation is the concatenation of the whole image, the head and the torso features. The last layer of M-CNN is a 200- way classiﬁcation (fc+softmax) layer for recognition on *CUB200- 2011* and a 500-way classiﬁcation layer on *Birdsnap*, respectively. The three-stream M-CNN is learned end-to-end, with the parame- ters of three CNNs learned simultaneously. During training M-CNN, the parameters of the learned FCN segmentation network are ﬁxed.

* 1. *Testing stage of Mask-CNN*

During inference, when facing with a testing image, the learned FCN model ﬁrstly returns the corresponding mask predictions for both head and torso. Then, based on the masks, we use them as the part detectors to localize the head part and torso part in the input images. The extracted head and torso image patches are re- garded as the inputs for the head and torso streams in Mask-CNN. After obtaining the convolutional descriptors through the convo- lution layers of three-stream Mask-CNN, the predicted masks are employed again. While, at this time, the masks are utilized for se- lecting descriptors (cf. [Fig. 5](#_bookmark17)(e)). At last, the selected descriptors are aggregated following the strategy in the training stage, and then we can get the predicted label based on the 3,072-d ﬁnal im-

age representation. The whole testing stage of Mask-CNN is shown in [Fig. 5](#_bookmark17).

### Experiments

In this section, we ﬁrstly describe the experimental settings and implementation details. Then, we report the classiﬁcation accuracy. The performance of part localization and object segmentation will also be provided. Finally, we present some discussions about the proposed M-CNN model.

* 1. *Dataset and implementation details*

Following the other published part-based ﬁne-grained methods [[7,8,10]](#_bookmark34), we perform the empirical evaluation on the widely-used ﬁne-grained benchmark Caltech-UCSD 2011 bird dataset [[1]](#_bookmark29) and Birdsnap [[20]](#_bookmark45) dataset. The *CUB200-2011* dataset contains 200 bird categories, and each category has roughly 30 training images. To- tally, there are 5994 training images and 5794 test images in that dataset. For the *Birdsnap* dataset, it contains 500 North Ameri- can bird species whose images having 47,386 images for training and 2443 images for test, which is much larger and challenging than *CUB200-2011*. We follow the training and testing splitting in- cluded with these two datasets. In the training phase, the ﬁfteen part annotations of each dataset are adopted for generating the part masks’ ground-truth, and meanwhile the image-level labels are used for the end-to-end M-CNN joint training. We need no su- pervision signals (*e.g.*, part annotations or bounding boxes) when testing.

The proposed Mask-CNN model and FCN used for generating masks are implemented using the open-source library MatConvNet [[43]](#_bookmark50). In our experiments, after getting the learned part masks, we ﬁrstly generate the image patches of birds’ head and torso as de- scribed in [Section 3.2](#_bookmark15). Then, to facilitate the convergence of three- stream CNNs, each single stream corresponding to the whole im- age, head and torso is ﬁne-tuned on its input images separately. The CNNs used in each stream is initialized by the popular VGG- 16 model [[21]](#_bookmark46) pre-trained on ImageNet. The loss function in each stream is the popular used cross-entropy loss function. For fair comparisons with other methods (*e.g.*, [[7,8,10]](#_bookmark34)), we also implement our three-stream M-CNN model based on the Alex-Net model [[15]](#_bookmark39).

710 *X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714*

**Table 2**

Comparison of classiﬁcation accuracy on *Birdsnap* with state-of-the-art methods. Note that, “Model” describes the deep models used in these methods. We do not list the inference speeds on this dataset, because the inference speeds on *Birdsnap* is similar to the speeds on *CUB200-2011*.

Method Train phase Test phase Model Dim. Acc.

BBox Parts BBox Parts

MixDCNN [[45]](#_bookmark52) ✓ GoogLeNet × 1 6144 74.1%

Multi-grained [[46]](#_bookmark53) ✓ VGG-19 × 3 12,288 74.8%

Multi-grained [[46]](#_bookmark53) VGG-19 × 3 12,288 65.9%

Our 3-stream M-CNN (Alex-Net) ✓ Alex-Net (w/o FCs) × 3 **1,536 64.8%**

Our 3-stream M-CNN (VGG-16) ✓ VGG-16 (w/o FCs) × 3 **3,072 77.3%**

Our 3-stream M-CNN (ResNet-50) ✓ ResNet-50 × 3 **12,288 80.2%**

**Table 3**

Comparison of part localization performance on the *CUB200-2011* dataset.

Method Train phase Test phase Head Torso

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | BBox | Parts |  | BBox | Parts |  | |
| Strong DPM [[32]](#_bookmark56) | ✓ | ✓ |  | ✓ |  | 43.49% | 75.15% |
| Part-based R-CNN with BBox [[10]](#_bookmark35) | ✓ | ✓ |  | ✓ |  | 68.19% | 79.82% |
| Deep LAC [[8]](#_bookmark36) | ✓ ✓ ✓ | | | | 74.00% | | **96.00%** |
| Part-based R-CNN [[10]](#_bookmark35) | ✓ ✓ | | | | 61.42% | | 70.68% |
| Ours (Alex-Net based FCN) | ✓ | | | | 81.22% | | 91.72% |
| Ours (VGG-16 based FCN) | ✓ | | | | **86.76%** | | 91.87% |

In addition, we double the training data by a horizontal ﬂipping for all the three streams. After ﬁne-tuning on each stream, as shown in [Fig. 2](#_bookmark11), the joint training of three-stream M-CNN is performed. Dropout is not used in M-CNN. At the test time, we average the predictions of the image and its ﬂipped copy, and output the class with the highest score as the prediction for a test image. In addi- tion, directly using the softmax predictions results is a slight drop in accuracy compared to logistic regression (LR), which is con- sistent with the observations in [[27]](#_bookmark51). Therefore, in the following, the reported results of M-CNN are all achieved by one-vs-all lo- gistic regression [[44]](#_bookmark51) on the extracted features of three-stream M- CNNs with the default hyper-parameter *C*LR = 1. Upon acceptance, we will release our source code and trained models, so that all re- sults in the paper can be reproduced. All the experiments are run on a computer with Intel Xeon E5-2660 v3, 64G main memory, and an Nvidia Tesla K40 GPU.

* 1. *Comparisons with state-of-the-art methods*

In this section, we compare our proposed M-CNN with state-of- the-arts on *CUB200-2011* and *Birdsnap*, respectively.

* + 1. *Results on CUB200-2011*

We report the classiﬁcation accuracy on the *CUB200-2011* dataset of the proposed three-stream M-CNN model, and compare with the baseline methods and state-of-the-art methods in the lit- erature. The classiﬁcation results are presented in [Table 1](#_bookmark4). For fair comparison, we only report the results when they do not use part annotations in testing.

At ﬁrst, the input images of the three streams are all resized to 224 × 224. The classiﬁcation accuracy of M-CNN is 84.2%. Fol- lowing [[7,27]](#_bookmark34), we change the input images of the whole image stream to 448 × 448 pixels. It improves the classiﬁcation perfor- mance by 1.5%, which achieves the best classiﬁcation accuracy 85.7% on *CUB200-2011*. Moreover, when using the Residual Net-50

[[22]](#_bookmark48) architecture, three-stream M-CNN obtains 87.3% accuracy.

For comparisons of the ﬁnal classiﬁcation accuracy on *CUB200- 2011*, since there is no previous work in the same experimental setting (*i.e.*, only using the part annotation in training) as ours, we divide the previous work into two kinds of ﬁne-grained meth- ods: the ﬁrst one are the methods using the part annotations (*e.g.*,

[[7,8,10,30]](#_bookmark34)), and the second one are the methods using only image- level labels (*e.g.*, [[11,26,27,33–35,38,45,46]](#_bookmark37)).

On one hand, comparing with the methods using the part anno- tations, part-stacked CNN [[7]](#_bookmark34) was one of state-of-the-arts, which is a strong baseline of Mask-CNN. Speciﬁcally, because part-stacked CNN used the Alex-Net model [[15]](#_bookmark39), we also build another three- stream M-CNN based on Alex-Net. The accuracy of our three- stream M-CNN (Alex-Net) is 78.6%. It is 2.0% higher than that of [[7]](#_bookmark34). Meanwhile, the inference speed of our three-stream M-CNN (Alex-Net) is 33.9 FPS, which is much faster than 20 FPS reported in [[7]](#_bookmark34). Moreover, in the Alex-Net based three-stream M-CNN, the ﬁnal feature vector is only 1,536-dimensional.

On the other hand, for the methods using only image-level la- bels, such as PDFS [[35]](#_bookmark44), Spatial Transformer CNN [[26]](#_bookmark50) and Bilinear [[27]](#_bookmark51), they are two outstanding ﬁne-grained methods using only the image-level supervisions. The classiﬁcation accuracy of our Mask- CNN is 1.2% and 1.6% higher than PDFS and STCNN, respectively. It also validates the effectiveness of our proposed method. Moreover, the image representation of M-CNN has lower feature dimensions than that of Bilinear [[27]](#_bookmark51) and PDFS [[35]](#_bookmark44).

* + 1. *Results on birdsnap*

The classiﬁcation accuracy on *Birdsnap* is reported in [Table 2](#_bookmark19). The input images of the whole image stream are of the 448 × 448 image resolution. The input images of the other streams are of 224 × 224. Comparing with the previous methods conducted on *Birdsnap*, our proposed M-CNN outperforms them by a large mar- gin. Meanwhile, the small dimensionality bring it scalability and

eﬃciency in large-scale datasets.

* 1. *Part localization results*

In addition to the qualitative part localization results shown in [Section 3.1](#_bookmark14), in this section, we quantitatively assess the localization correctness using the Percentage of Correctly Localized Parts (PCP) metric.

As reported in [Table 3](#_bookmark20), the metrics of *CUB200-2011* are the per-

centage of parts (*i.e.*, the head and torso) that are correctly lo- calized with a *>* 50% IOU with the ground-truth part bounding boxes as generated in [[8,10]](#_bookmark36). Comparing the results of PCP for torso, our method (no matter based on VGG-16 or Alex-Net) out- performs part-based R-CNN [[10]](#_bookmark35) and strong DPM [[32]](#_bookmark56) by a large

*X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714* 711

**Table 4**

Comparison with the baseline methods on *CUB200-2011* and *Birdsnap*. For all the models, the inputs of the whole image stream are 224 × 224 for fair comparisons. The inference speed contains both part masks predictions and the ﬁnal classiﬁcation process.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | 3-stream FCs | 3-stream Pooling | The proposed 3-stream M-CNN |
| Descriptor selection | ✗ | ✗ | ✓ |
| Accuracy on *CUB200-2011* | 81.4% | 82.8% | **84.2%** |
| Accuracy on *Birdsnap* | 73.0% | 74.3% | **76.0%** |
| Inference speed | 9.2 FPS | **13.0 FPS** | 12.9 FPS |

margin. However, because we do not use any annotation in test- ing, the localization performance is lower than the one of Deep LAC [[8]](#_bookmark36) which used the bounding boxes during testing. In addi- tion, for the head localization task which is more challenging than the torso one, even though our method just uses part annotations in training, the head localization performance (86.76% for VGG-16 based, and 81.22% for Alex-Net based) is still signiﬁcantly higher than the other methods.

Additionally, the head and torso localization accuracy on *Bird-*

*snap* of our method are 67.40% and 78.87% based on Alex-Net, and 74.51% and 84.45% based on VGG-16, respectively. Since there is no previous results on part localization on *Birdsnap* in the literature, we further conducted experiments using the released source codes of Part-based R-CNN [[10]](#_bookmark35) for comparisons. When utilizing part an- notations and bounding boxes of the *Birdsnap* dataset, the part lo- calization accuracy of Part-based R-CNN are 49.97% and 74.86% for head and torso, respectively. Particularly, the head localization ac- curacy of our method (no matter based on Alex-Net or VGG-16) is signiﬁcantly higher than the accuracy of Part-based R-CNN. The observations of *Birdsnap* are consistent with the localization results

of *CUB200-2011*.

* 1. *Object segmentation performance*

Because the *CUB200-2011* dataset also supplies the object seg- mentation ground-truth, we can directly test the learned object masks on the segmentation metric. The ﬁgures in the second row of [Fig. 4](#_bookmark16) show qualitative segmentation results. Our method based on FCN is generally able to segment the foreground object well, but understandably has trouble to segment the birds’ ﬁner de- tails, *e.g.*, claws and beak. Since our goal is not to segment ob- jects, we do not perform any reﬁnement as pre-processing or post- processing. We evaluate the segmentation performance quantita- tively by the common semantic segmentation metric mean IU (re- gion intersection over union) between the ground truth foreground object and the predicted object masks. It is 74.59% on the testing set. In fact, a better segmentation result will lead to better pre- dicted object/part masks, and also beneﬁt the ﬁnal classiﬁcation. To further improve the classiﬁcation accuracy, some pre-processing methods, *e.g.*, GrabCut [[47]](#_bookmark54), are worth trying to obtain better mask ground-truth than the rectangles in [Fig. 3](#_bookmark12)c.

* 1. *Ablation and diagnostic experiments*

In this section, we conduct ablation experiments on both the *CUB200-2011* and *Birdsnap* dataset, and present discussions of the proposed three-stream M-CNN model. The experimental results are all based on M-CNN with the VGG-16 architecture.

* + 1. *Is descriptor selection effective?*

In order to clearly validate the effectiveness of the descriptor selection process in M-CNN, we perform two baseline methods which are also based on the proposed three-stream architecture. Different from our M-CNN, these two baseline methods do not contain the descriptor selection part, *i.e.*, the processing shown in [Fig. 2](#_bookmark11)d.

The ﬁrst baseline method employ the traditional fully con- nected layers to conduct classiﬁcation for each stream, which is called “3-stream FCs”. In “3-stream FCs”, we replace the (b) to (e) parts of each stream in [Fig. 2](#_bookmark11) with a CNN containing fully con- nected layers (*i.e.*, VGG-16 with only fc8 removed). Thus, the gen- erated feature in the last layer of each stream is a 4,096-d single vector. The rest procedure is also to concatenate the three 4,096-d features into the ﬁnal one with 12,288-d, and to learn a 200-way (or 500-way) classiﬁcation (fc+softmax) layer on the 12,288-d im- age representation.

The second baseline is similar to the proposed M-CNN. The most prominent difference is that it discards the descriptor se- lection part, *i.e.*, the processing in [Fig. 2](#_bookmark11)d. Thus, the convolutional deep descriptors of pool5 in each stream are directly average and max pooled, and then *£*2 -normalized, respectively. Therefore, we call it the “3-stream Pooling”. The remaining procedures are the same as the proposed M-CNN.

[Table 4](#_bookmark23) presents the comparison of classiﬁcation accuracy and inference speed on the *CUB200-2011* and *Birdsnap* dataset. Be- cause CNN models with fully connected layers require the inputs of 224 × 224, the input images of these three compared methods

in [Table 4](#_bookmark23) are all 224 × 224. In that table, the proposed M-CNN

achieves the best classiﬁcation accuracy rate. Due to the missing of descriptor selection, “3-stream Pooling” is about 1.4% lower than M-CNN on *CUB200-2011* and 1.7% on *Birdsnap*. The “3-stream FCs” baseline method has the lowest accuracy. Its lower accuracy might be caused by the fully connected layers, which may have caused overﬁtting.

In addition, the feature extraction speeds (frames/sec) on a Tesla K40 GPU for these methods using our MatConvNet based im- plementation are shown on the bottom of [Table 4](#_bookmark23). The speeds are conducted on the *CUB200-2011* dataset, and the *Birdsnap* dataset has the similar inference speeds. In addition, please note that, for these streams models, the speeds are the serial computing speeds. That is to say, a GPU is used for inference one stream by one stream (whole image → head → torso). The inference speed of the

proposed M-CNN is almost the same as the “3-stream Pooling”

baseline model without selecting descriptor, and is 3.7 FPS faster than the baseline with fully connected layers. Besides, as the input images in the whole image stream are of 224 × 224, the inference speed (12.9 FPS) is faster than the one whose inputs are 448 × 448 (8.3 FPS).

For further validating the effectiveness of selecting descriptors, we also change the inputs of the whole image stream in “3-stream Pooling” to 448 × 448, which can get 84.5% on *CUB200-2011* (76.0%

on *Birdsnap*). It is still 1.2% (1.3%) lower than the classiﬁcation ac-

curacy of our three-stream M-CNN (cf. 85.7% in [Table 1](#_bookmark4) and 77.3% in [Table 2](#_bookmark19)).

* + 1. *How important are different streams?*

We here investigate what different streams contribute to the ﬁ- nal recognition performance. [Table 5](#_bookmark26) reports the classiﬁcation ac- curacy of M-CNN containing different streams. When it only has the whole image stream, on *CUB200-2011*, the accuracy is 80.5%. By incorporating the head and torso stream, after joint training, the

712 *X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714*

**Table 5**

Comparison of M-CNN with different streams on the *CUB200-2011* and *Birdsnap*

dataset.

Dataset Stream Accuracy

Image Head Torso

*CUB200-2011* ✓ 80.5%

✓ ✓ 84.2%

✓ ✓ 82.1%

✓ ✓ ✓ **85.7%**

*Birdsnap* ✓ 72.7%

✓ ✓ 76.2%

✓ ✓ 73.6%

✓ ✓ ✓ **77.3%**

**Table 6**

Comparison of the three-stream M-CNN model with different pooling strategies on

the *CUB200-2011* and *Birdsnap* dataset.

### Conclusion

In this paper, we presented the beneﬁts of selecting deep con- volutional descriptor in object recognition, especially ﬁne-grained image recognition. By developing the descriptor selection scheme, we proposed a novel end-to-end Mask-CNN (M-CNN) model with- out the fully connected layers to not only accurately localize ob- ject/parts, but also generate weighted object/part masks for se- lecting deep convolutional descriptors. After aggregating the se- lected descriptors, the object-level and part-level cues were en- coded by the proposed three-stream M-CNN model. Mask-CNN not only achieved a new state-of-the-art bird species classiﬁcation ac- curacy on *CUB200-2011* and *Birdsnap*, but also had the lowest di- mensional feature representations.

In the future, we plan to solve the part detection problem of M-CNN in the weakly supervised setting, in which we only require

the image-level labels. Thus, it will require far less labeling effort

Accuracy Pooling Accuracy Ave.-pool Max-pool

*CUB200-2011* ✓ 85.1%

* 85.4%

✓ ✓ **85.7%**

*Birdsnap* ✓ 77.1%

* 76.9%

✓ ✓ **77.3%**

accuracy increases to 85.7% until containing both two part streams. From that table, we can ﬁnd the head stream could be more im- portant/discriminative than the torso stream. After incorporating the head stream, the original whole image stream can improve 3.7% (80.5% → 84.2%). However, incorporating the torso stream, it just increases 1.6% accuracy (80.5% → 82.1%). Additionally, compar- ing with the results of two-stream (*i.e.*, the second and third row) and three-stream (*i.e.*, the last row) in [Table 5](#_bookmark26) separately, we can see that: adding the torso stream improves the accuracy by 1.5% (84.2% → 85.7%), while adding the head one can improve it by 3.6% (82.1% → 85.7%). Besides, similar observations can be found on *Birdsnap*.

* + 1. *Are different pooling strategies necessary?*

In M-CNN, we propose to concatenate both the average- and max-pooled features in each stream as the ﬁnal representations. In the following, the diagnostic experiments on different pooling strategies are presented. As shown in [Table 6](#_bookmark27), on *CUB200-2011*,  the proposed M-CNN (average-pooling+max-pooling, 85.7%) out- performs the ones with only average-pooled features (85.1%) or only max-pooled features (85.4%). For *Birdsnap*, different pooling ensemble can also improve the classiﬁcation accuracy up to 77.3%. Therefore, different pooling strategies used in M-CNN is necessary for the ﬁnal classiﬁcation accuracy.

* + 1. *Can M-CNN share the previous layers between different streams?*

Because the early layers of CNNs usually correspond to low- level visual atoms (*e.g.*, orientated edges, bars or blobs) [[48]](#_bookmark55), we at- tempt to combine the ﬁrst ten layers (from “conv1\_1 with relu1\_1 ” to “conv2\_1 *,* relu2\_1 with pool2 ”) in VGG-16 as layer sharing, and jointly train the new three-stream M-CNN. However, the accuracy of the layer sharing M-CNN is only 82.1% on *CUB200-2011* (73.6% on *Birdsnap*), which are signiﬁcantly worse than 85.7% (77.3%) of the proposed M-CNN. But, the result of layer sharing justiﬁes the separate design of the three-stream M-CNN’s architecture.

to achieve comparable classiﬁcation accuracy. In addition, another interesting direction is to explore the beneﬁts of descriptor selec- tion for generic object categorization.

### Acknowledgement

[This research was supported by the National Natural Science Foundation of China under Grant](https://doi.org/10.13039/501100001809) [6](#_bookmark28)[1772256 and Grant](https://doi.org/10.13039/501100001809) [6](#_bookmark28)[1422203,](https://doi.org/10.13039/501100001809) the Collaborative Innovation Center of Novel Software Technology and Industrialization. The authors would like to thank Qichang Hu, Jian-Hao Luo for reading the draft, and thank the anonymous re- viewers, whose comments have helped improving this paper.

### References

1. [C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie, The caltech-UCSD birds-200-2011 dataset, Tech. Report CNS-TR-2011-001 (2011).](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0001)
2. [P. Welinder, S. Branson, T. Mita, C. Wah, F. Schroff, S. Belongie, P. Perona, Cal- tech-UCSD birds 200, Technical Report, California Institute of Technology, 2010.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0002)
3. [A. Angelova, S. Zhu, Y. Lin, Image segmentation for large-scale subcategory ﬂower recognition, in: Proceedings of Applications of Computer Vision, Clear- water Beach, FL, Jan. 2013, pp. 39–45.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0003)
4. [M.-E. Nilsback, A. Zisserman, Automated ﬂower classiﬁcation over a large number of classes, in: Proceedings of Indian Conference on Computer Vision, Graphics and Image Processing, Bhubaneswar, India, Dec. 2008, pp. 722–729.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0004)
5. [J. Krause, M. Stark, J. Deng, L. Fei-Fei, 3D object representations for ﬁne-grained categorization, in: Proceedings of IEEE International Conference on Computer Vision Workshop on 3D Representation and Recognition, Sydney, Australia, Dec. 2013.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0005)
6. [T. Berg, P. Belhumeur, POOF: Part-based one-vs.-one features for ﬁne-grained categorization, face veriﬁcation and attribute estimation, in: Proceedings of IEEE International Conference on Computer Vision, Portland, Oregon, Jun. 2013, pp. 955–962.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0006)
7. [S. Huang, Z. Xu, D. Tao, Y. Zhang, Part-stacked CNN for ﬁne-grained visual cate- gorization, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, Jun. 2016, pp. 1173–1182.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0007)
8. [D. Lin, X. Shen, C. Lu, J. Jia, Deep LAC: Deep localization, alignment and classiﬁcation for ﬁne-grained recognition, in: Proceedings of IEEE Confer- ence on Computer Vision and Pattern Recognition, Boston, MA, Jun. 2015, pp. 1666–1674.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0008)
9. [S. Maji, G. Shakhnarovich, Part and attribute discovery from relative annota- tions, Int. J. Comput. Vis. 108 (1–2) (2014) 82–96.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0009)
10. [N. Zhang, J. Donahue, R. Girshick, T. Darrell, Part-based R-CNNs for ﬁne-grained category detection, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Euro- pean Conference on Computer Vision, Part I, LNCS 8689, Springer, Switzerland, Zürich, Switzerland, Sept. 2014, pp. 834–849.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0010)
11. [Y. Zhang, X.-S. Wei, J. Wu, J. Cai, J. Lu, V.-A. Nguyen, M.N. Do., Weakly super- vised ﬁne-grained categorization with part-based image representation, IEEE Trans. Image Process. 25 (4) (2016) 1713–1725.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0011)
12. [G.-S. Xie, X.-Y. Zhang, W. Yang, M. Xu, S. Yan, C.-L. Liu, LG-CNN: From local parts to global discrimination for ﬁne-grained recognition, Pattern Recognit. 71 (2017) 118–131.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0012)
13. [S.H. Lee, C.S. Chan, S.J. Mayo, P. Remagnino, How deep learning extracts and learns leaf features for plant classiﬁcation, Pattern Recognit. 71 (2017) 1–13.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0013)
14. [M.D. Zeiler, G.W. Taylor, R. Fergus, Adaptive deconvolutional networks for mid and high level feature learning, in: Proceedings of IEEE International Confer- ence on Computer Vision, Sydney, Australia, Dec. 2013, pp. 2018–2025.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0014)
15. [A. Krizhevsky, I. Sutskever, G.E. Hinton, ImageNet classiﬁcation with deep con- volutional neural networks, in: Advances in Neural Information Processing Sys- tems, Lake Tahoe, NV, Dec. 2012, pp. 1097–1105.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0015)

*X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714* 713

1. [A. Eigenstetter, B. Ommer, Visual recognition using embedded feature selec- tion for curvature self-similarity, in: Advances in Neural Information Process- ing Systems, Lake Tahoe, NV, Dec. 2012, pp. 377–385.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0016)
2. [Y. Zhang, J. Wu, J. Cai, Compact representation for image classiﬁcation: To  choose or to compress? in: Proceedings of IEEE Conference on Computer Vi- sion and Pattern Recognition, Columbus, OH, Jun. 2014, pp. 907–914.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0017)
3. [X.-S. Wei, J.-H. Luo, J. Wu, Z.-H. Zhou, Selective convolutional descriptor aggre- gation for ﬁne-grained image retrieval, IEEE Trans. Image Process. 26 (6) (2017) 2868–2881.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0018)
4. [J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic seg- mentation, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, Jun. 2015, pp. 3431–3440.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0019)
5. [T. Berg, J. Liu, S.W. Lee, M.L. Alexander, D.W. Jacobs, P.N. Belhumeur, Birdsnap: Large-scale ﬁne-grained visual categorization of birds, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, Jun. 2014, pp. 2019–2026.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0020)
6. [K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, in: Proceedings of International Conference on Learning Representations, San Diego, CA, May. 2015, pp. 1–14.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0021)
7. [K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, Jun. 2016, pp. 770–778.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0022)
8. [A. Khosla, N. Jayadevaprakash, B. Yao, L. Fei-Fei, Novel dataset for ﬁne-grained image categorization, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshop on Fine-Grained Visual Categorization, Col- orado Springs, CO, Jun. 2011, pp. 806–813.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0023)
9. [O.M. Parkhi, A. Vedaldi, A. Zisserman, C.V. Jawahar, Cats and dogs, in: Pro- ceedings of IEEE Conference on Computer Vision and Pattern Recognition, Jun. 2012, pp. 3498–3505.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0024) [Providence, RI](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0024)
10. [E. Rodner, M. Simon, G. Brehm, S. Pietsch, J.W. Wägele, J. Denzler, Fine-grained recognition datasets for biodiversity analysis, in: Proceedings of IEEE Confer-  ence on Computer Vision and Pattern Recognition Workshop on Fine-grained Visual Classiﬁcation, Boston, MA, Jun. 2015.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0025)
11. [M. Jaderberg, K. Simonyan, A. Zisserman, K. Kavukcuoglu, Spatial transformer networks, in: Advances in Neural Information Processing Systems, Montréal, Canada, Dec. 2015, pp. 2008–2016.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0026)
12. [T.-Y. Lin, A. RoyChowdhury, S. Maji, Bilinear CNN models for ﬁne-grained visual recognition, in: Proceedings of IEEE International Conference on Computer Vi- sion, Sandiago, Chile, Dec. 2015, pp. 1449–1457.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0027)
13. [S. Gao, I.W.-H. Tsang, Y. Ma, Learning category-speciﬁc dictionary and shared dictionary for ﬁne-grained image categorization, IEEE Trans. Image Process. 23 (2) (2014) 623–634.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0028)
14. [H. Yao, S. Zhang, Y. Zhang, J. Li, Q. Tian, Coarse-to-ﬁne description for ﬁne-grained visual categorization, IEEE Trans. Image Process. 25 (10) (2016) 4858–4872.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0029)
15. [S. Branson, G.V. Horn, S. Belongie, P. Perona, Bird species categorization using pose normalized deep convolutional nets, in: British Machine Vision Confer- ence, Nottingham, England, Sept. 2014, pp. 1–14.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0030)
16. [E. Gavves, B. Fernando, C.G. Snoek, A.W. Smeulders, T. Tuytelaars, Local align- ments for ﬁne-grained categorization, Int. J. Comput. Vis. 111 (2) (2014) 191–212.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0031)
17. [H. Azizpour, I. Laptev, Object detection using strongly-supervised deformable part models, in: A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, C. Schmid (Eds.), European Conference on Computer Vision, Part I, LNCS 7572, Springer, Heidel- berg, Firenze, Italy, Oct. 2012, pp. 836–849.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0032)
18. [M. Simon, E. Rodner, Neural activation constellations: Unsupervised part model discovery with convolutional networks, in: Proceedings of IEEE In- ternational Conference on Computer Vision, Sandiago, Chile, Dec. 2015, pp. 1143–1151.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0033)
19. [T. Xiao, Y. Xu, K. Yang, J. Zhang, Y. Peng, Z. Zhang, The application of two-level attention models in deep convolutional neural network for ﬁne-grained im- age classiﬁcation, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, Jun. 2015, pp. 842–850.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0034)
20. [X. Zhang, H. Xiong, W. Zhou, W. Lin, Q. Tian, Picking deep ﬁlter responses for ﬁne-grained image recognition, in: Proceedings of IEEE Conference on Com- puter Vision and Pattern Recognition, Las Vegas, NV, Jun. 2016, pp. 1134–1142.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0035)
21. [L. Zhang, Y. Yang, M. Wang, R. Hong, L. Nie, X. Li, Detecting densely distributed graph patterns for ﬁne-grained image categorization, IEEE Trans. Image Pro- cess. 25 (2) (2016) 553–565.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0036)
22. [J.R.R. Uijlings, K.E.A. van de Sande, T. Gevers, A.W.M. Smeulders, Selective search for object recognition, Int. J. Comput. Vis. 104 (2) (2013) 154–171.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0037)
23. [J. Krause, H. Jin, J. Yang, L. Fei-Fei, Fine-grained recognition without part an- notations, in: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, Jun. 2015, pp. 5546–5555.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0038)
24. [J. Dai, K. He, J. Sun, Boxsup: Exploiting bounding boxes to supervise con- volutional networks for semantic segmentation, in: Proceedings of IEEE International Conference on Computer Vision, Sandiago, Chile, Dec. 2015, pp. 1635–1643.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0039)
25. [G. Papandreou, L.-C. Chen, K. Murphy, A. Yuille, Weakly- and semi-supervised learning of a deep convolutional network for semantic image segmentation, in: Proceedings of IEEE International Conference on Computer Vision, Sandiago, Chile, Dec. 2015, pp. 1742–1750.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0040)
26. [P. Krähenbühl, V. Koltun, Eﬃcient inference in fully connected CRF with Guas- sian edge potentials, in: Advances in Neural Information Processing Systems, Granada, Spain, Dec. 2011, pp. 109–117.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0041)
27. [L. Ladicky, C. Russell, P. Kohli, P.H. Torr, Associate hierarchical crfs for object class image segmentation, in: Proceedings of IEEE International Conference on Computer Vision, Kyoto, Japan, Sept. 2009, pp. 739–746.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0042)
28. A. Vedaldi, K. Lenc, MatConvNet – Convolutional Neural Networks for MAT- LAB, in: Proceeding of ACM International Conference on Multimedia, Brisbane, Australia, Oct. 2015, pp. 689–692. <http://www.vlfeat.org/matconvnet/>.
29. [R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, C.-J. Lin, LIBLINEAR: A library for large linear classiﬁcation, J. Mach. Learn. Res. 9 (2008) 1871–1874.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0044)
30. [Z.Y. Ge, A. Bewley, C. McCool, P. Corke, B. Upcroft, C. Sanderson, Fine-grained classiﬁcation via mixture of deep convolutional neural networks, in: Proceed- ings of IEEE Winter Applications of Computer Vision, Lake Placid, NY, Mar. 2016, pp. 1–6.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0045)
31. [D. Wang, Z. Shen, J. Shao, W. Zhang, X. Xue, Z. Zhang, Multiple granularity de- scriptors for ﬁne-grained categorization, in: Proceedings of IEEE International Conference on Computer Vision, Sandiago, Chile, Dec. 2015, pp. 2399–2406.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0046)
32. [C. Rother, V. Kolmogorov, A. Blake, GrabCut: Interactive foreground extraction using iterated graph cuts, ACM Trans. Graph. 23 (2004) 309–314.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0047)
33. [M. Zeiler, R. Fergus, Visualizing and Understanding Convolutional Neural Net- works, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), European Con- ference on Computer Vision, Part I, LNCS 8689, Springer, Switzerland, Zürich, Switzerland, Sept. 2014, pp. 818–833.](http://refhub.elsevier.com/S0031-3203(17)30399-0/sbref0048)

714 *X.-S. Wei et al. / Pattern Recognition 76 (2018) 704–714*

**Xiu-Shen Wei** received the B.S. degree in computer science and technology in 2012. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Nanjing University, China. His research interests are computer vision and machine learning. He received the ﬁrst prize in the Apparent Personality Analysis competition (in association with ECCV 2016) and the ﬁrst RunnerUp Award from the Cultural Event Recognition Competition (in association with ICCV 2015) as the Team Director.

**Chen-Wei Xie** received his BS degree from Southeast University, China, in 2015. He is currently a postgraduate student in the Department of Computer Science and Technol- ogy, Nanjing University, China. His research interests include computer vision and machine learning.

**Jianxin Wu** received his BS and MS degrees in computer science from Nanjing University, and his PhD degree in computer science from the Georgia Institute of Technology. He is currently a professor in the Department of Computer Science and Technology at Nanjing University, China, and is associated with the National Key Laboratory for Novel Software Technology, China. He was an assistant professor in the Nanyang Technological University, Singapore, and has served as an area chair for CVPR 2017, ICCV 2015, senior PC member for AAAI 2017, AAAI 2016 and an associate editor for Pattern Recognition Journal. His research interests are computer vision and machine learning.

**Chunhua Shen** is a Professor at School of Computer Science, University of Adelaide. He is a Project Leader and Chief Investigator at the Australian Research Council Centre of Excellence for Robotic Vision (ACRV), for which he leads the project on machine learning for robotic vision. Before he moved to Adelaide as a Senior Lecturer, he was with the computer vision program at NICTA (National ICT Australia), Canberra Research Laboratory for about six years. His research interests are in the intersection of computer vision and statistical machine learning. He studied at Nanjing University, at Australian National University, and received his PhD degree from the University of Adelaide. From 2012 to 2016 he held an Australian Research Council Future Fellowship.