International Journal of Automation and Computing 14(2), April 2017, 119-135
DOI: 10.1007/s11633-017-1053-3

A Survey on Deep Learning-based Fine-grained Object Classification and Semantic Segmentation

Bo Zhao[1,2] Jiashi Feng[2] Xiao Wu[1] Shuicheng Yan[2]

[1]School of Information Science and Technology, Southwest Jiaotong University, Chengdu 613000, China

[2]Department of Electrical and Computer Engineering, National University of Singapore, 117583, Singapore

Abstract: The deep learning technology has shown impressive performance in various vision tasks such as image classification, object detection and semantic segmentation. In particular, recent advances of deep learning techniques bring encouraging performance to fine-grained image classification which aims to distinguish subordinate-level categories, such as bird species or dog breeds. This task is extremely challenging due to high intra-class and low inter-class variance. In this paper, we review four types of deep learning based fine-grained image classification approaches, including the general convolutional neural networks (CNNs), part detection based, ensemble of networks based and visual attention based fine-grained image classification approaches. Besides, the deep learning based semantic segmentation approaches are also covered in this paper. The region proposal based and fully convolutional networks based approaches for semantic segmentation are introduced respectively.

Keywords: Deep learning, fine-grained image classification, semantic segmentation, convolutional neural network (CNN), recurrent neural network (RNN)

1 Introduction

Deep learning has recently achieved superior performance on many tasks such as image classification, object detection and neural language processing. The core of the deep learning technology is that the layers of the features are not designed by human engineers and instead learned from data using a general-purpose learning procedure. There are a huge number of variants of the deep learning architecture. Most of them are branched from some original parent architectures. In this survey, we mainly focus on the convolutional neural network (CNN) and recurrent neural network (RNN) based approaches.

CNN is a type of feed-forward artificial neural network consisting of one or more convolutional layers which are then followed by one or more fully connected layers as in a standard MultiLayer perceptron (MLP). The convolutional layer is the core building block of a CNN. The layer′s parameters comprise a set of learnable filters (or kernels), which have a small receptive field, but extend through the full depth of the input volume. CNN has wide applications in image classification, object detection and image retrieval systems. Fully convolutional network (FCN) is a special convolutional neural network which replaces all the fully connected layers in CNNs with convolutional layers. FCN can be trained end-to-end, pixels-to-pixels, which is very suitable for the task of semantic segmentation.

RNN is a kind of neural network where connections between units form a directed cycle, thus the activations can flow round in a loop. Unlike feedforward neural networks, RNNs can use their internal memory to process arbitrary sequences of inputs. This makes them applicable to tasks such as unsegmented connected handwriting recognition[1] or speech recognition[2]. One of the most popular RNNs is the long-short term memory (LSTM)[3] which can remember a value for an arbitrary length of time. An LSTM unit contains multiple gates that determine when the input is significant enough to be remembered, when it should continue to remember or forget the value, and when it should output the value. Other RNN models include GNU[4], MGU[5].

Most deep learning networks can be trained end-to-end efficiently using backpropagation. It is a common method of training artificial neural networks used in conjunction with an optimization method such as gradient descent. The method calculates the gradient of a loss function with respect to all the weights in the network. The gradient is fed to the optimization method which in turn uses it to update the weights, in order to minimize the loss function.

Different from backpropagation, reinforcement learning is another kind of technology that lets the networks learn what to do – how to map situations to actions – so as to maximize a numerical reward signal. The networks are not told which actions to take, as in most forms of deep learning, but instead must discover which actions yield the most reward by trying them. In the most interesting and challenging

Review
Manuscript received July 1, 2016; accepted September 30, 2016; published online January 18, 2017
This work was supported by the National Natural Science Foundation of China (Nos. 61373121 and 61328205), Program for Sichuan Provincial Science Fund for Distinguished Young Scholars (No. 13QNJJ0149), the Fundamental Research Funds for the Central Universities, and China Scholarship Council (No. 201507000032).
Recommended by Associate Editor Nazim Mir-Nasiri

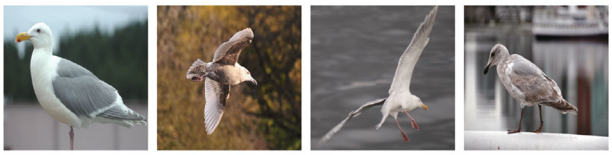
cases, actions may affect not only the immediate reward but also the next situation and, through that, all subsequent rewards.

In this survey, we introduce the deep learning based approaches using the backpropagation or reinforcement learning. More concretely, the deep learning based fine-grained object classification will be firstly elaborated and then the deep learning based image semantic segmentation.

2 Deep fine-grained image classification

With the advancement of deep learning, fine-grained image classification received considerable attention. Many deep learning based approaches have been proposed in recent years. Fine-grained object classification aims to distinguish objects from different subordinate-level categories within a general category, e.g., different species of birds, dogs or different classes of cars. However, fine-grained classification is a very challenging task, because objects from similar subordinate categories may have marginal visual differences that are even difficult for humans to recognize. In addition, objects within the same subordinate category may present large appearance variations due to changes of scales or viewpoints, complex backgrounds and occlusions. Fig. 1 demonstrates three different species of gulls with high intra-class variance and small inter-class variance.

(a) Calilfornia gull



(b) Glaucous gull

Fig. 1 Two species of gulls from CUB 200 dataset illustrate the difficulty of fine-grained object classification: large intra-class variance and small inter-class variance. The pose, background and viewpoint of the gull within the same species vary largely, and different specie of gulls display high visual similarity. The discriminative differences only exist in some subtle regions, e.g., the beak or wings.

Existing deep learning based fine-grained image classification approaches can be classified into the following four groups according to the use of additional information or human inference: 1) those approaches that directly use the general deep neural networks (mostly the CNNs) to classify the fine-grained images, 2) those using the deep neural networks as the feature extractor to better localize different parts of the fine-grained object and do alignment, 3) those using multiple deep neural networks to better differentiate highly visually-similar fine-grained images, and 4) those using the visual attention mechanism to find the most discriminative regions of the fine-grained images.

In this section, we will first introduce several convolutional neural networks which are mostly used for fine-grained image classification. Then, part detection and alignment based approaches and ensemble of networks based approaches will be elaborated respectively. The last part of this section will review the attention based approaches.

2.1 General CNN for fine-grained image classification

CNN has a long history in computer vision. It was firstly introduced by LeCun et al.[6] and has consistently been competitive with other methods for recognition tasks. Recently, with the advent of large-scale category-level training data, e.g., ImageNet[7], CNN exhibits superior performance in large-scale visual recognition. The impressive performance of CNN[8] also motivates researchers to adapt CNNs pre-trained on ImageNet to other domains and datasets, such as the fine-grained image datasets. Besides, CNN usually is able to yield more discriminative representation of the image, which is essential for fine-grained image classification. Most of the current state-of-the-art CNNs can be adopted for fine-grained image classification.

AlexNet[6] is a deep convolutional neural network which is a winner of the ILSVRC-2012 competition with top-5 test error rate of 15.3 %, compared to 26.2 % achieved by the second-best entry. It contains eight learnable layers. The first five are convolutional and the remaining three are fully-connected. Fig. 2 illustrates the architecture of AlexNet.

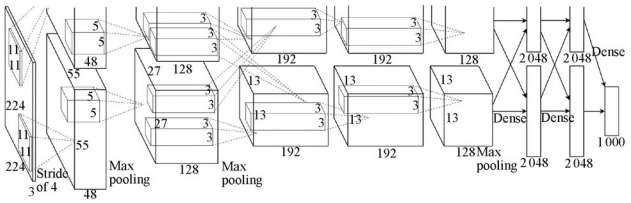


Fig. 2 Framework of AlexNet. This figure is from the original paper[8].

The VGG net[9] increases the depth of the neural networks, which not only achieves the state-of-the-art accuracy on ILSVRC classification and localization tasks, but also is applicable to other image recognition datasets. The VGG-16 has 13 convolutional layers with 3 fully connected layers, while the VGG-19 has 3 more convolutional layers than a VGG-16 model. They use filters with a very small receptive field: 3×3 (which is the smallest size to capture the notion of left/right, up/down, center). All hidden layers are equipped with the rectification non-linearity.

The GoogLeNet[10] sets the new state-of-the-art for classification and detection in the ImageNet large-scale visual recognition challenge 2014 (ILSVRC14). The main hall-

mark of this architecture is the improved utilization of the computing resources inside the network. This is achieved by a carefully crafted design called "inception module" that allows for increasing the depth and width of the network while keeping the computational budget constant. GoogLeNet is 22 layers deep when counting only layers with parameters (or 27 layers if we also count pooling). By stacking the inception modules, it uses 12 times fewer parameters than the winning architecture of Krizhevsky et al.[8] The inception module is depicted as Fig. 3. The 1×1 convolutions are used to compute reductions before the expensive 3×3 and 5×5 convolutions. Besides being used as reductions, they also include the use of rectified linear activation which makes them dual-purpose.

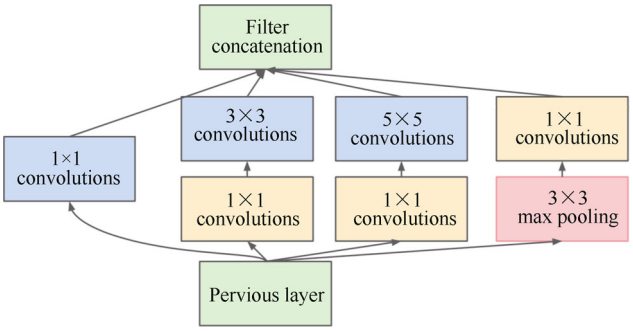


Fig. 3 Inception module of GoogLeNet. This figure is from the original paper [10].

Some other general deep convolutional feature extractors for image classification include CNN features off-the-shelf[11], ONE[12] and InterActive[13]. When using these CNNs for fine-grained image classification, the last fully-connected layer will be set as the class number of the fine-grained images such as 200 for the CUB-Bird-2011 dataset. The classification results indicate that the generic descriptors extracted from the convolutional neural networks are very powerful.

2.2 Part detection and alignment based approaches

Semantic part localization can facilitate fine-grained categorization by explicitly isolating subtle appearance differences associated with specific object parts. Localizing the parts in an object is therefore important for establishing correspondence between object instances and discounting object pose variations and camera view position changes. Many traditional approaches follow the pipeline illustrated in Fig. 4. The parts of the fine-grained object are first localized, such as head and torso for bird species classification, then the part alignment is done and the last is the classification using the feature extracted on the aligned parts. POOF[14] learns a set of intermediate features using data mining techniques. Each of these features specializes in discrimination between two particular classes based on the appearance at a particular part. To find accurate part such as face and eyes of dogs, Liu et al.[15] build exemplar-based geometric and appearance models of dog breeds and their face parts. Yang et al.[16] propose a template model to discover the common geometric patterns of object parts and the co-occurrence statistics of the patterns. Features are extracted within the aligned cooccurred patterns for fine-grained recognition. Similarly, Gavves et al.[17] and Chai et al.[18] segment images and align the image segments in an unsupervised fashion. The alignments are then used to transfer part annotations from training images to test images and extract features for classification. In this subsection, we will introduce the part detection methods based on deep learning.

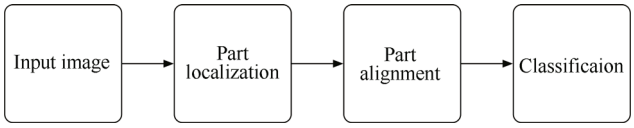


Fig. 4 General framework for part detection and alignment based approaches.

2.2.1 Part-based R-CNN

With deep convolutional features, the part detector which is widely used in many approaches improves its performance. Therefore, the Part-based R-CNN[19] learns the part detectors by leveraging deep convolutional features computed on bottom-up region proposals. It extends R-CNN[20] to detect objects and localize their parts under a geometric prior. The whole process is illustrated in Fig. 5. Starting from several region proposals using selective search[21], both object and part detectors are trained based on the deep convolutional features. During testing, all proposals are scored by all detectors, and non-parametric geometric constraints are applied to rescore the proposals and choose the best object and part detections. The final step is to extract features on the localized semantic parts for fine-grained recognition for a pose-normalized representation and then train a classifier for the final categorization.

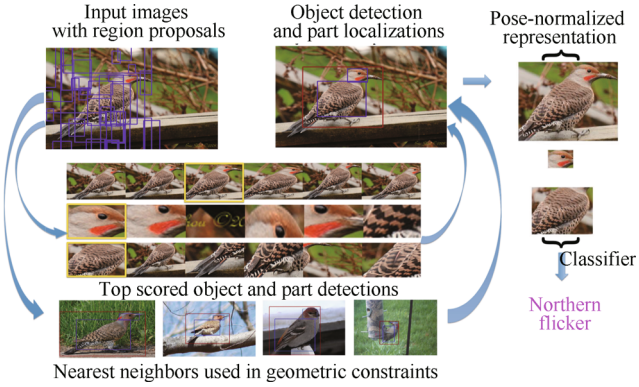


Fig. 5 Part-based R-CNN for fine-grained image classification. This figure is from the original paper [19].

In order to make the deep CNN derived features more discriminative for the target task of fine-grained bird classifi-

cation, ImageNet pre-trained CNN is first fine-tuned for the 200-way bird classification task from ground truth bounding box crops of the original CUB images. In particular, the original 1 000-way fc8 classification layer in CaffeNet, which is almost identical as the AlexNet, is replaced with a new 200-way fc8 layer. Both the full objects bounding box annotations and a fixed set of semantic parts annotations are used to train multiple detectors. All objects and each of their parts are initially treated as independent object categories: A one-versus-all linear SVM is trained on the convolutional feature descriptors extracted over region proposals. Because the individual part detectors are less than perfect, the window with highest individual part detector scores is not always correct, especially when there are occlusions. A geometric constraint over the layout of the parts relative to the object location is considered to filter out incorrect detections.

In testing, the bottom-up region proposals are scored by all detectors, and the non-parametric geometric constraints are imposed to rescore the windows and choose the best object and part detections. At the final step, features for the predicted whole object or part region are extracted and concatenated using the network fine-tuned for that particular whole object or part. Then, a one-versus-all linear SVM classifier is trained using the final feature representation.

2.2.2 Part localization using multi-proposal consensus

Different from the part-based R-CNN, which uses the geometric constraint to better locate the parts, multi-proposal consensus[22] predicts the keypoint and region (head, torso, body) using a single neural network based on the AlexNet[8] architecture.

The multi-proposal consensus modifies the AlexNet to simultaneously predict all keypoint locations and their visibilities for any given image patch. The final fc8 layer is replaced with two separate output layers for keypoint localization and visibility, respectively. The network is trained on edge box crops[23] extracted from each image and is initialized with a pre-trained AlexNet trained on the ImageNet[7] dataset. After getting the keypoint predictions and their visibilities, the ones with low visibility confidences will be removed. The remaining predictions will have a peaky distribution around the ground truth. Therefore, medoid is used as a robust estimator for this peak. Fig. 6 demonstrates the process of finding the right eye keypoint. The best location of the right eye is determined by performing confidence thresholding and finding the medoid. Black edge boxes without associated dots make predictions with confidences below the set threshold, and green denotes an outlier with a high confidence score.

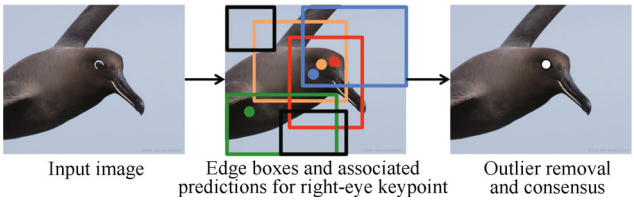


Fig. 6 Part localization using multi-proposal consensus. This figure is from the original paper[22].

Using the keypoints, three regions are identified from each bird: head, torso, and whole body. The head is defined as the tightest box surrounding the beak, crown, forehead, eyes, nape, and throat. Similarly, the torso is the box around the back, breast, wings, tail, throat, belly, and legs. The whole body bounding box is the object bounding box provided in the annotations. To perform classification, fc6 features of AlexNet are extracted from these localized regions. These CNN features are then concatenated into a feature vector of length $4\,096 \times 3$, and used for 200-way linear one-vs-all SVM classification.

2.2.3 Pose normalized nets

The pose normalized net[24] first computes an estimate of the object′s pose which is used to compute local image features. These local features in turn are used for classification. The features are computed by applying deep convolutional networks to image patches that are located and normalized by the pose. The pose normalized net integrates lower-level feature layers (conv5, fc6) with pose-normalized extraction routines and higher-level feature layers (fc8) with unaligned image features as shown in Fig. 7.

In training, the pose normalized net uses the DPM[25] to predict 2D locations and the visibility of 13 semantic part keypoints or directly uses the pre-provided object bounding box and part annotations to learn the pose prototypes. Then, different parts of the object will be warped and fed

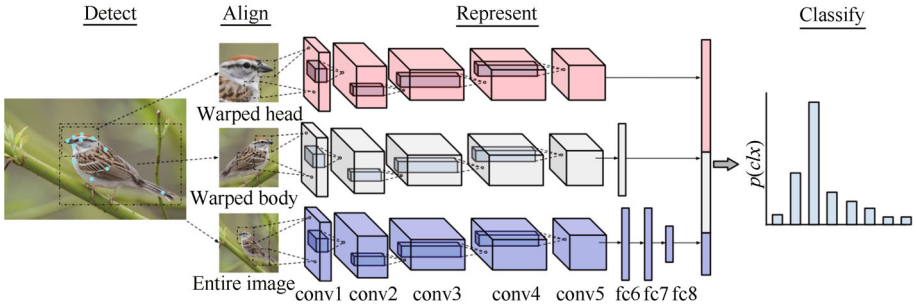


Fig. 7 Pose normalized nets. This figure is from the original paper [24].

into different deep neural networks (AlexNet) to extract the features. Finally, the classifier is trained with the concatenated convolutional features extracted from each prototype region and the entire image.

In testing, given a test image, groups of detected keypoints or the oracle parts annotations are used to compute multiple warped image regions that are aligned with prototypical models. Multiple warped image regions are then aligned with prototypical models. Then, the pose-warped image regions are each fed into a feature extractor, which is a deep convolutional neural network[8]. It is proved that a model that integrates lower-level feature layers with pose-normalized extraction routines and higher-level feature layers with unaligned image features works best.

2.2.4 Part-stack CNN

Based on manually-labeled strong part annotations, the part-stacked CNN (PS-CNN) model[26] consists of a fully convolutional network to locate multiple object parts and a two-stream classification network that encodes object-level and part-level cues simultaneously, as shown in Fig. 8.

An FCN is achieved by replacing the parameter-rich fully connected layers in standard CNN architectures by convolutional layers with 1×1 kernels. Given an input RGB image, the output of an FCN is a feature map in the reduced dimension compared to the input. The computation of each unit in the feature map only corresponds to pixels inside a region with fixed size in the input image, which is called its receptive field. FCN is preferred in PS-CNN due to the following three reasons: 1) Feature maps generated by FCN can be directly utilized as the part localization results in the classification network. 2) Results of multiple object parts can be obtained simultaneously using an FCN. 3) FCN is very efficient in both learning and inference.

Using M keypoints annotated at the center of each object part, the localization network, which is a fully convolutional network[27], is trained to generate dense output feature maps for locating object parts. A Gaussian kernel is used to remove isolated noise in the feature maps. The final output of the localization network is M locations in the conv5 feature map, each of which is computed as the location with the maximum response for one object part. Then, the part locations are fed into the classification network, in which a two-level architecture is adopted to analyze images at both object-level (bounding boxes) and part-level (part landmarks).

At the part level, the computation of multiple parts is first conducted via a shared feature extraction route, and then separated through a part crop layer. The input for the part crop layer is a set of feature maps, e.g., the output of the conv5 layer, and the predicted part locations from the previous localization network, which also reside in conv5 feature maps. For each part, the part crop layer extracts a local neighborhood region centered at the detected part location. At the object level, bounding-box supervision is used to extract object-level CNN features, i.e., pool5 features. Three fully connected layers achieve the final classification results based on a concatenated feature map containing information from all parts and the bounding box.

2.2.5 Deep LAC

The deep LAC[28] incorporates part localization, alignment, and classification in one deep neural network. Its framework is demonstrated in Fig. 9. A valve linkage function (VLF) is proposed for back-propagation chaining, and to form the deep localization, alignment and classification (LAC) system. The VLF can adaptively compromise the errors of classification and alignment when training the LAC model. It in turn helps update localization.

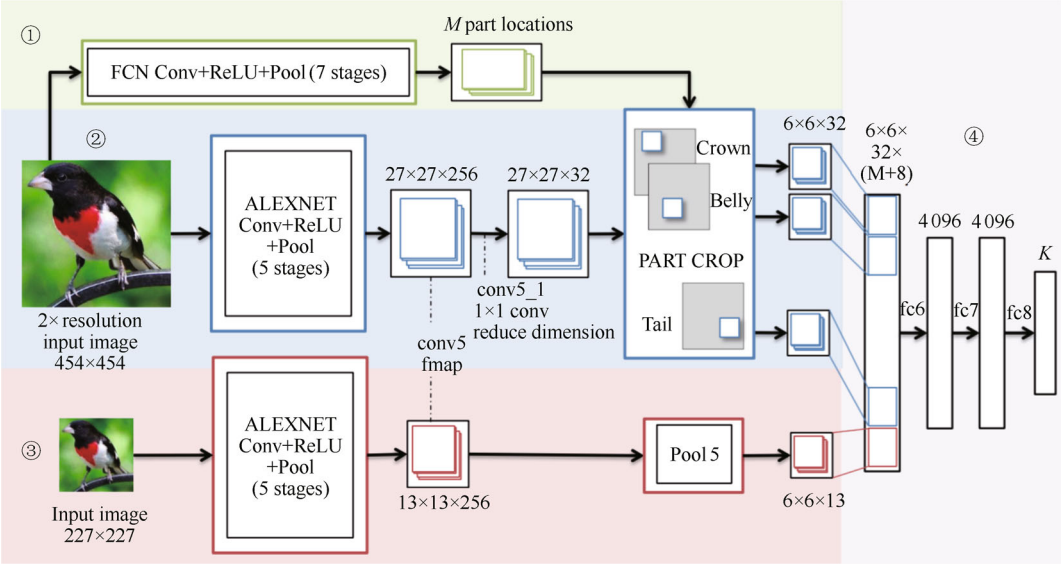


Fig. 8 Part-stack CNN. This figure is from the original paper [26].

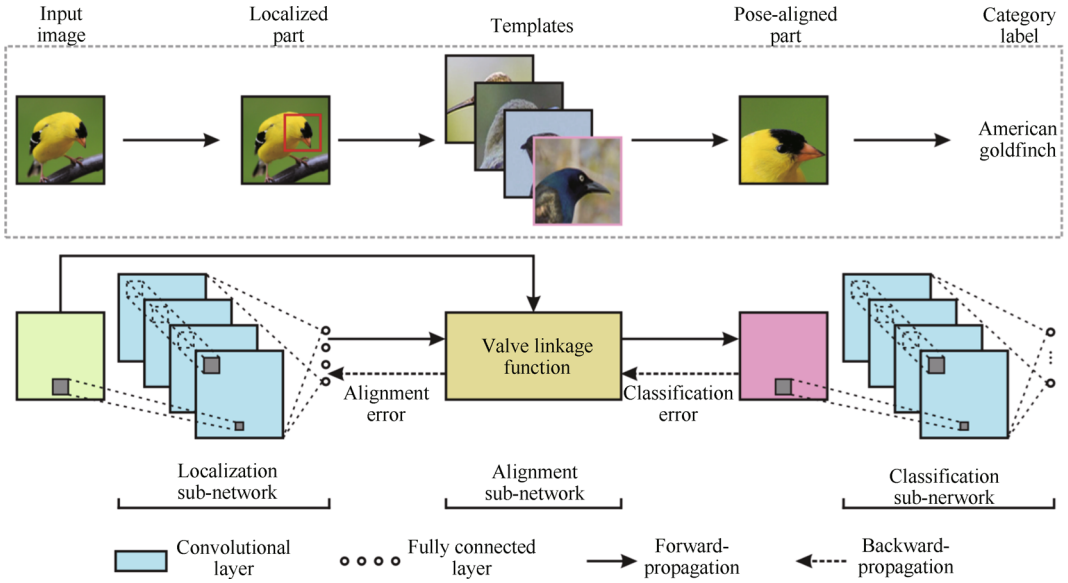


Fig. 9 Framework of deep LAC. This figure is from the original paper[28].

The part localization sub-network consists of 5 convolutional layers and 3 fully connected ones. It outputs the commonly used coordinates for the top-left and bottom-right bounding-box corners, given an input natural image for fine-grained recognition. In the training phase, deep LAC regresses bounding boxes of part regions. Ground truth bounding boxes are generated with part annotations.

The alignment sub-network receives part locations (i.e., bounding box) from the localization sub-network, performs template alignment[29] and feeds a pose-aligned part image to classification. The alignment sub-network offsets translation, scaling, and rotation for pose-aligned part region generation, which is important for accurate classification. Apart from pose aligning, this sub-network plays a crucial role in bridging the backward-propagation (BP) stage of the whole LAC model, which helps utilize the classification and alignment results to refine localization.

In the deep LAC framework, the VLF in the alignment sub-network is the most essential part which optimally connects the localization and classification modules. It not only connects all sub-networks, but also functions as information valve to compromise classification and alignment errors. If the alignment is good enough in the forward propagation stage, VLF guarantees corresponding accurate classification. Otherwise, errors propagated from classification finely tune the previous modules. These effects make the whole network reach a stable state.

2.3 Ensemble of networks based approaches

Dividing the fine-grained dataset into multiple visually similar subsets or directly using multiple neural networks to improve the performance of classification is another widely used method in many deep learning based fine-grained image classification systems. We will introduce these methods in this subsection.

2.3.1 Subset feature learning networks

As shown in Fig. 10, the subset feature learning networks[30] consist of two main parts: a domain-generic convolution neural network and several specific convolutional neural networks. The domain-generic convolution neural network is first pre-trained on a large-scale dataset of the same domain as the target dataset and then fine-tuned on the target dataset. Using the fc6 feature with linear discriminant analysis (LDA) to reduce its dimensionality, visually similar species are clustered into K subsets to train multiple specific CNNs in the second part.

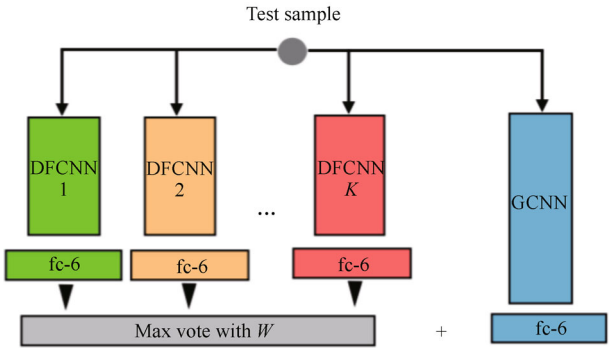


Fig. 10 Framework of subset feature learning networks. This figure is from the original paper [30].

A separate CNN is learned for each of the K pre-clustered subsets. The aim is to learn features for each subset which can more easily differentiate visually similar species. The fc6 feature of each individual CNNs is used as the learned subset feature for each subset. Each specific convolutional neural networks is suitable for a subset of the images. Therefore, how to choose the best specific convolutional neural networks for an image is the core problem of the

subset feature learning networks.

A subset selector CNN (SCNN) is utilized to select the most relevant CNNs to make prediction for a given image. Using the output from the pre-clustering as the class labels, SCNN is trained by changing the softmax layer fc8 to make it have K outputs. The softmax layer predicts the probability of the test image belonging to a specific subset, and then max voting is applied to this prediction to choose the most likely subset. As with the previously trained CNNs, the weights of SCNN are trained via backpropagation and stochastic gradient descent (SGD) using the AlexNet[8] as the starting point.

2.3.2 Mixture of deep CNN

Similar to subset feature learning networks, the MixDCNN[31] system will also learn K specific CNNs. However, it does not require pre-dividing images into K subsets of similar images. The image will be fed into all the K CNNs and the outputs from each CNN are combined to form a single classification decision. In contrast to subset feature learning networks, MixDCNN adopts the occupation probabilities equation to perform joint end-to-end training of the K CNNs simultaneously.

The occupation probability is defined as

$$\alpha_k = \frac{e^{C_k}}{\sum_{c=1}^{K} e^{C_c}} \tag{1}$$

where C_k is the best classification result for the k-th CNN. The occupation probability gives a higher weight to components that are confident about their prediction. The overall structure of this network is shown in Fig. 11.

The occupation probability of each subset is based on the classification confidence from each component, which makes it possible to jointly train the K DCNNs (components) without having to estimate a separate label vector y or train a separate gating network as in subset feature learning networks. Classification is performed by multiplying the output of the final layer from each component by the occupation probability and then summing over the K components. This mixes the network outputs together and the probability for each class is then produced by applying the softmax function.

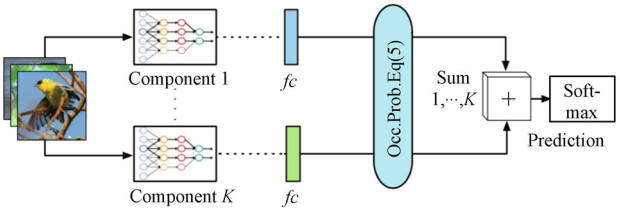


Fig. 11 Framework of MixDCNN. This figure is from the original paper [31].

2.3.3 CNN tree

Motivated by the observation that a class is usually confused by a few number of other classes, which are called the confusion set, in multi-class classification, more discriminative features could be learned by a specific CNN to distinguish the classes only in this set. Based on this observation, CNN tree[32] is used to progressively learn fine-grained features for different confusion sets.

Given a node of the tree, a CNN model is first trained on its class set. Next, the confusion set of each class is estimated by using the trained model. Then, these confusion sets are packed into several confusion supersets and each of them is assigned to a new child node for further learning. This procedure repeats until it reaches the maximal depth. The tree structure is shown in Fig. 12. The CNN tree progressively learns fine-grained features to distinguish a subset of classes, by learning features only among these classes. Such features are expected to be more discriminative, compared to features learned for all the classes. Besides, test images that are misclassified by the root CNN model might be the correctly classified by its descendent.

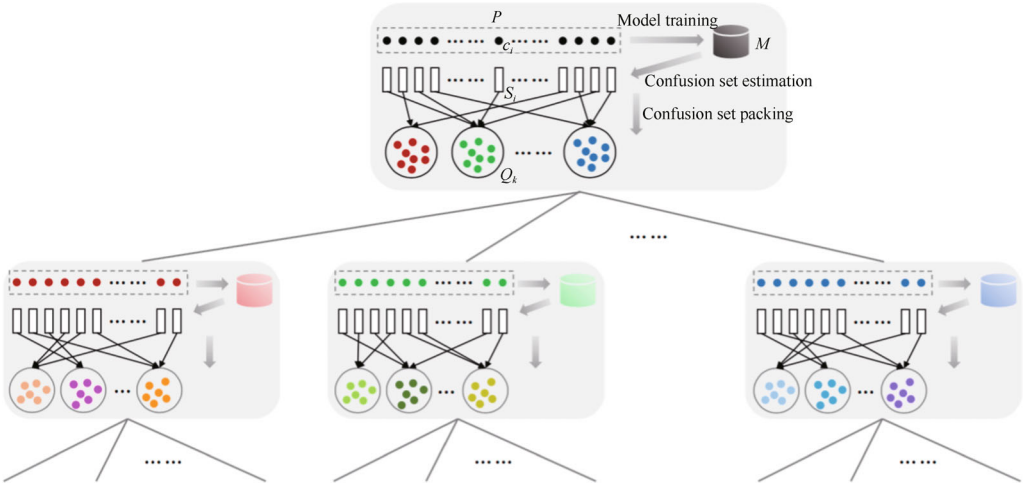


Fig. 12 Framework of CNN tree. This figure is from the original paper [32].

2.3.4 Multiple granularity CNN

A core observation is that a subordinate-level label carries an implied hierarchy of labels, each corresponding to a level in the domain ontology. For instance, melanerpes formicivorus, also known as acorn woodpecker, can also be called melanerpes at genus level, or picidae at family level. These labels are free for extracting their corresponding discriminative patches and features. These free labels can be used to train a series of CNN-based classifiers, each specialized at one grain level. The internal representations of these networks have different regions of interest, allowing the construction of multi-grained descriptors that encode informative and discriminative features covering all the grain levels.

Based on this idea, the multiple granularity CNN[33] contains a parallel set of deep convolutional neural networks as shown in Fig. 13, each optimized to classify at a given granularity. In other words, the multiple granularity CNN is composed of a set of single-grained descriptors. Saliency in their hidden layers guides the selection of regions of interest (ROI) from a common pool of bottom-up proposed image patches. ROI selection is therefore by definition granularity-dependent, in the sense that selected patches are results of the associated classifier of a given granularity. Meanwhile, ROI selections are also cross-granularity dependent: The ROIs of a more detailed granularity are typically sampled from those at the coarser granularities. Finally, per-granularity ROIs are fed into the second stage of the framework to extract per-granularity descriptors, which are then merged to give classification results.

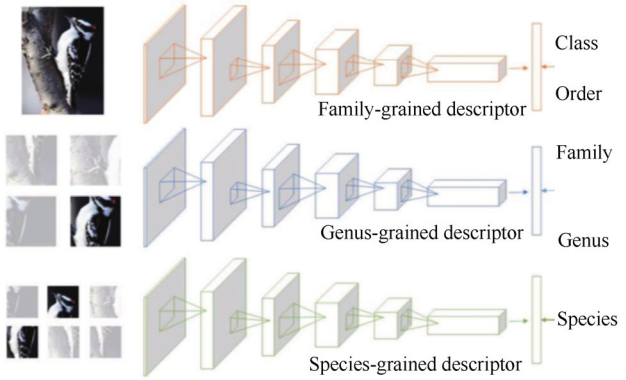


Fig. 13 Framework of Multiple Granularity NN. This figure is from the original paper [33].

2.3.5 Bilinear deep network models

The bilinear models[34] are a recognition architecture that consists of two feature extractors whose outputs are multiplied using outer product at each location of the image and pooled to obtain an image descriptor. This architecture, as shown in Fig. 14, can model local pairwise feature interactions in a translationally invariant manner which is particularly useful for fine-grained categorization.

A bilinear model for image classification consists of a quadruple $\mathcal{B} = (f_A, f_B, \mathcal{P}, \mathcal{C})$. Here f_A and f_B are feature functions, $\mathcal{P}$ is a pooling function and C is a classification function. A feature function is a mapping $f : \mathcal{L} \times \mathcal{I} \rightarrow \mathbf{R}^{c \times D}$ that takes as input an image I and a location L and outputs a feature of size $c \times D$. The locations generally can include position and scale. The feature outputs are combined at each location using the matrix outer product, i.e., the bilinear feature combination of f_A and f_B at a location l is given by bilinear $(l, \mathcal{I}, f_A, f_B) = f_A(l, \mathcal{I})^\top f_B(l, \mathcal{I})$. Both f_A and f_B must have the feature dimension c to be compatible. To obtain an image descriptor, the pooling function $\mathcal{P}$ aggregates the bilinear features across all locations in the image. One choice of pooling is to simply sum all the bilinear features, i.e., $\phi(\mathcal{I}) = \sum_{l \in \mathcal{L}}$ bilinear $(l, \mathcal{I}, f_A, f_B)$. An alternative is max-pooling. Both ignore the locations of the features and are hence orderless. If f_A and f_B extract features of size $C \times M$ and $C \times N$ respectively, then $\phi(\mathcal{I})$ is of size $M \times N$. The bilinear vector obtained by reshaping $\phi(\mathcal{I})$ to size $MN \times 1$ is a general-purpose image descriptor that can be used with a classification function $\mathcal{C}$. Intuitively, the bilinear form allows the outputs of the feature extractors f_A and f_B to be conditioned on each other by considering all their pairwise interactions similar to a quadratic kernel expansion.

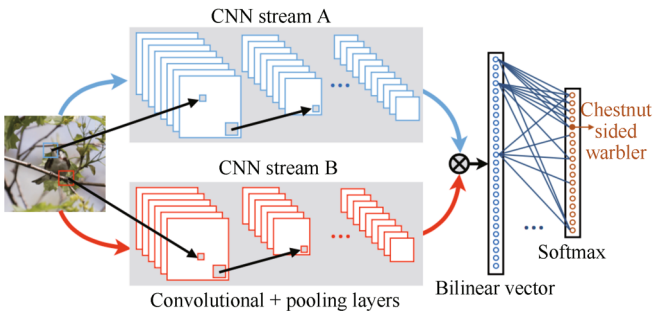


Fig. 14 Framework of bilinear CNN model. This figure is from the original paper [34].

A natural candidate for the feature function f is a CNN consisting of a hierarchy of convolutional and pooling layers. In this paper, the authors use two different CNNs pre-trained on the ImageNet dataset[7] truncated at a convolutional layer including non-linearities as feature extractors. By pre-training, bilinear deep network model will benefit from additional training data in the cases of domain specific data scarcity. This has been shown to be beneficial for a number of recognition tasks ranging from object detection, texture recognition, to fine-grained classification[20, 35–37]. Another advantage of using only the convolutional layers is that the resulting CNN can process images of an arbitrary size in a single forward-propagation step and produce outputs indexed by the location in the image and the feature channel.

2.4 Visual attention based approaches

One of the most curious facets of the human visual system is the presence of attention. Rather than compressing

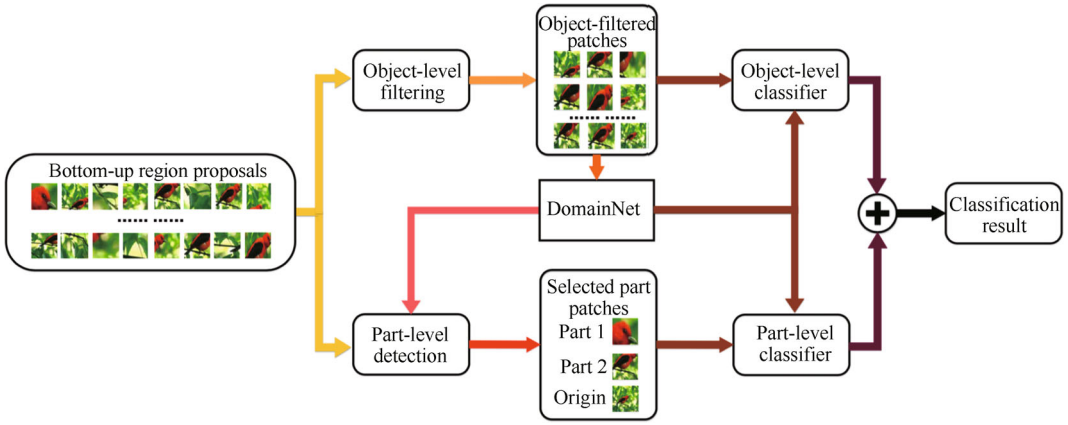


Fig. 15 Framework of two-level attention. This figure is from the original paper [38].

an entire image into a static representation, the attention system allows for salient features to dynamically come to the forefront as needed. This is especially important when there are many clutters in an image. A visual attention mechanism is also used in many fine-grained image classification systems.

2.4.1 Two-level attention

The two-level attention model[38], illustrated in Fig. 15 integrates three types of attention: the bottom-up attention that proposes candidate patches, the object-level top-down attention that selects relevant patches to a certain object, and the part-level top-down attention that localizes discriminative parts. These attention types are combined to train domain-specific deep nets, and then used to find foreground object or object parts to extract discriminative features. The model is easy to generalize, since it does not require the object bounding box or part annotation.

Then, a DomainNet is trained with the patches selected by the FilterNet. Essentially, spectral clustering is performed on the similarity matrix S to partition the filters in a middle layer into k groups. Each cluster acts as a part detector. The patches selected by the part detector are then wrapped back to the input size of DomainNet to generate activations. The activations of different parts and the original image are concatenated and used to train an SVM as the part-based classifier. Finally, the prediction results of the object-level attention and the part-level attention are merged to utilize the advantage of the two level attention.

2.4.2 Attention for fine-grained categorization

Inspired from the way how humans perform visual sequence recognition, such as reading by continually moving the fovea to the next relevant object or character, recognizing the individual object, and adding the recognized object to the internal representation of the sequence, the attention for fine-grained categorization (AFGC) system is proposed[39]. It is a deep recurrent neural network that at each step, it processes a multi-resolution crop of the input image, called a glimpse. The network uses information from the glimpse to update its internal representation of the input, and outputs the next glimpse location and possibly the next object in the sequence.

Fig. 16 shows the framework of the attention module in AFGC. It uses an RNN and a powerful visual network (GoogLeNet) to perform fine-grained classification. The system as a whole takes as input an image of any size and outputs N-way classification scores using a softmax classifier, which is a task similar to find digits and digit addition[40]. The model is a recurrent neural network, with N steps that correlate with N "glimpses" into the input image. At step n, the model receives row and column coordinates l_n, which describe a point in the input image. The network extracts a multi-resolution patch from the input image at those coordinates, passes the pixels through fully-connected layers which combine with activations from the previous glimpse step, and either outputs coordinates $\hat{l}_n$ for the next glimpse or a final classification y_s.

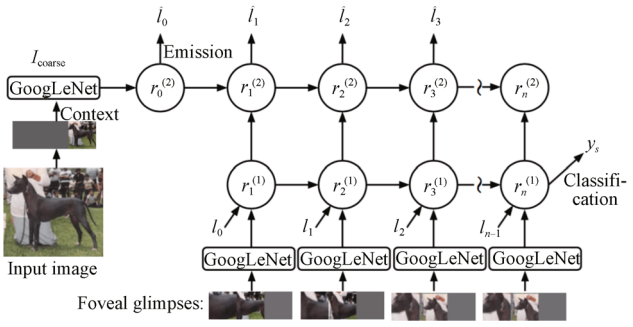


Fig. 16 Framework of attention for fine-grained categorization. This figure is from the original paper [39].

2.4.3 FCN attention

FCN attention[41] is a reinforcement learning-based fully convolutional attention localization network to adaptively select multiple task-driven visual attention regions. Compared to previous reinforcement learning-based models[39, 40, 42], the proposed approach is noticeably more computationally efficient during both training and testing because of its fully-convolutional architecture, and it is capable of simultaneous focusing its glimpse on multiple visual attention regions.

Fig. 17 illustrates the architecture of the fully convolutional attention localization network. It can localize multiple object parts using the attention mechanism. Different parts can have different pre-defined sizes. The network contains two components: part localization component and classification component.

The part-localization component uses a fully-convolutional neural network to locate part locations. Given an input image, the basis convolutional feature maps are extracted using the VGG 16 model[9] pre-trained on ImageNet dataset[7] and fine-tuned for the target fine-grained dataset. The attention localization network localizes multiple parts by generating a score map for each part using the basis convolutional feature map. Each score map is generated using two stacked convolutional layers and one spatial softmax layer. The first convolutional layer uses sixty-four 3×3 kernels, and the second one uses one 3×3 kernel to output a single-channel confidence map. The spatial softmax layer is applied to the confidence map to convert the confidence score into probability. The attention region with highest probability is selected as the part location. The same process is applied for a fixed number of time steps for multiple part locations. Each time step generates the location for a particular part.

The classification component contains one deep CNN classifier for each part as well as the whole image. Different parts might have different sizes, and a local image region is cropped around each part location according to its size. An image classifier for each local image region is trained as well as the whole image separately. The final classification result is the average of all the classification results from the individual classifiers. In order to discriminate the subtle visual differences, each local image region is resized to high resolution. A deep convolutional neural network is trained for each part for classification separately.

2.4.4 Diversified visual attention

A diversified visual attention network (DVAN)[43] is proposed to pursue the diversity of attention and is able to gather discriminative information to the maximal extent. The architecture of the proposed DVAN model is described in Fig. 18, which includes four components: attention canvas generation, CNN feature learning, diversified visual attention and classification. DVAN first localizes several regions of the input image at different scales and takes them as the "canvas" for following visual attention. A convolutional neural network (i.e., VGG-16) is then adopted to learn convolutional features from each canvas of attention. To localize important parts or components of the object within each canvas, a diversified visual attention component is introduced to predict the attention maps, so that important locations within each canvas are highlighted and information gain across multiple attention canvases is maximized. Different from traditional attention models focusing on a single discriminative location, DVAN jointly identifies diverse locations with the help of a well designed diversity promoting loss function. According to the generated attention maps, the convolutional features will be dynamically pooled and accumulated into the diversified attention model. Meanwhile, the attention model will predict the object class at each time step. All the predictions will be averaged to obtain the final classification results.

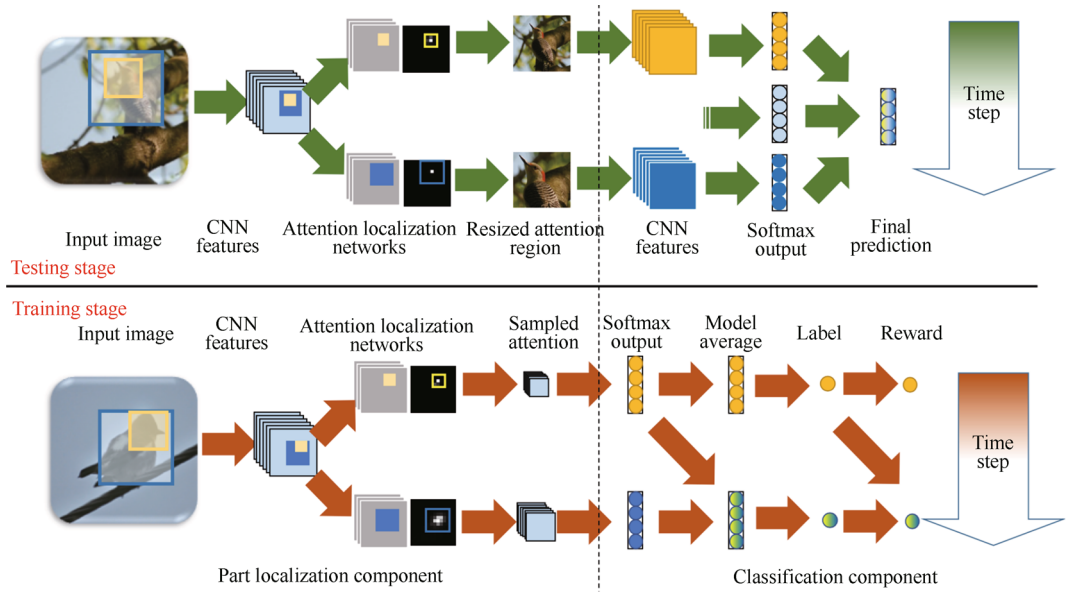


Fig. 17 Framework of FCN attention. This figure is from the original paper [41].

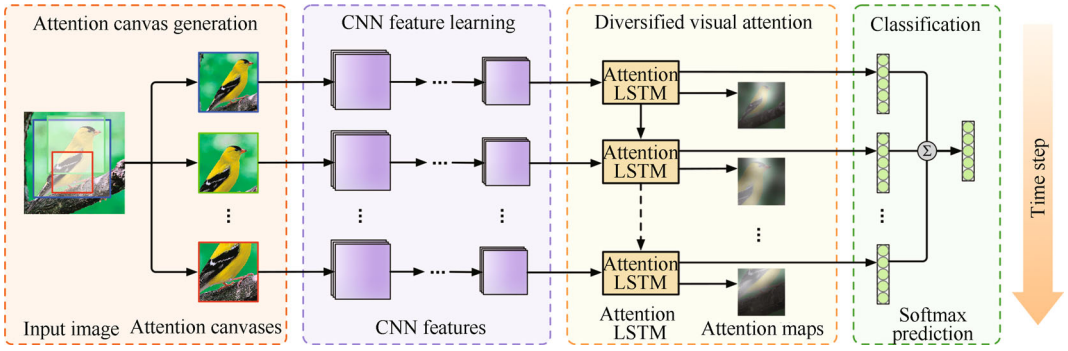


Fig. 18 Framework of diversified visual attention networks. This figure is from the original paper [43].

The adopted visual attention component in DVAN consists of two modules: attentive feature integration and attention map prediction, as demonstrated in the top and bottom panels of Fig. 19. To diversify the attention regions at each time step, a diversified loss function and attention canvas generation methods are proposed in DVAN.

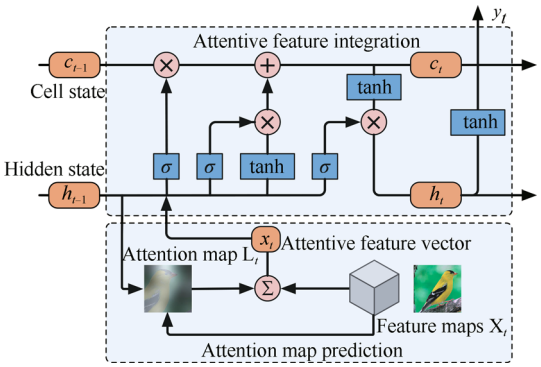


Fig. 19 DVAN attention component. This figure is from the original paper[43].

2.5 Performance comparison and analysis

We list the classification accuracy of CUB200-2011 dataset[44] using the above mentioned deep learning approaches in Table 1. The classification accuracy is defined as the average of class classification accuracy. CUB-200-2011 dataset consists of 11 778 images from 200 bird categories. It provides rich annotations, including image-level labels, object bounding boxes, attribute annotations and part landmarks. There are 5 994 images for training and 5 794 images for testing. Note that some approaches are omitted since they did not report the result on this dataset.

From Table 1, it can be seen that the approaches are grouped into three groups. The approaches in first group are based on the part detection and alignment and the approaches in second group ensemble multiple neural networks to boost the classification performance. While the visual attention based models in the third group simulate the observation process of human beings and usually do not need the bounding box information or part annotation. These approaches are based on different neural networks such as AlexNet[8], VGGNet[9] or GoogLeNet[10]. Some approaches may use the bounding box and part annotations in training and testing, while some only use the category label to train the networks.

In general, part localization-based fine-grained recognition algorithms can localize important regions using a set of predefined parts. However, detailed part annotations are usually difficult to obtain. Recent fine-grained object classification methods are capable of learning discriminative region localizers only from category labels with reinforcement learning. However, they cannot accurately find multiple distinctive regions without utilizing any explicit part information. Comparing to the labor and time consuming part annotation for fine-grained object classification, the attribute labeling is more amenable. Therefore, attribute label information can be used as a weak supervision to the part localization to further improve the classification accuracy.

Meanwhile, the recurrent visual attention models are effective in localizing the parts and learn their discriminative representations in an end-to-end way. Many recurrent visual attention models are proposed in recent years. Existing visual attention models can be classified as soft or hard attention. Soft attention models[45, 46] predict the attention regions in a deterministic way. As a consequence, it is differentiable and can be trained using back-propagation. Hard attention models[39−42, 47] predict the attention points of an image, which are stochastic. They are usually trained by reinforcement learning[48] or maximizing an approximate variational lower bound. In general, soft attention models are more efficient than hard attention models, since hard attention models require sampling for training while soft attention models can be trained end-to-end. However, the visual attention models also suffer from several drawbacks in practice. Firstly, by far the soft attention models only result in small performance improvement. More powerful visual attention model is expected to improve the classification accuracy. Secondly, the hard attention methods using reinforcement learning techniques usually are not as efficient as the soft attention methods. Methods which can improve the efficiency of the hard attention model should be explored further.

Table 1 Performance comparison with different approaches

Method	Architecture	Train annotation	Test annotation	Accuracy(%)
Part detection and alignment based approaches				
Part-based R-CNN[19]	AlexNet	BBox + Parts	BBox	76.4
Part-based R-CNN[19]	AlexNet	BBox + Parts	–	73.9
Multi-proposal consensus[22]	AlexNet	BBox	BBox	80.3
PoseNorm[24]	Alexnet	BBox + Parts	–	75.7
PS-CNN[26]	AlexNet	BBox + Parts	BBox	76.2
Deep LAC[28]	AlexNet	BBox	BBox	80.3
Ensemble of networks based approaches				
Subset FL[30]	AlexNet	–	–	77.5
MixDCNN[31]	AlexNet	BBox	BBox	74.1
Multiple granularity CNN[33]	VGGNet	BBox	–	83.0
Multiple granularity CNN[33]	VGGNet	–	–	81.7
Bilinear CNN[34]	VGGNet	BBox	BBox	77.2
Bilinear CNN[34]	VGGNet	–	–	72.5
Visual attention based approaches				
Two-level attention[38]	AlexNet	–	–	69.7
FCN attention[41]	GoogLeNet	BBox	–	84.3
FCN attention[41]	GoogLeNet	–	–	82.0
DVAN[43]	VGGNet	–	–	79.0

3 Deep image semantic segmentation

Deep learning based image semantic segmentation aims to predict a category label for every image pixel, which is an important yet challenging task for image understanding. Recent approaches have applied convolutional neural network (CNNs)[49−51] to this pixel-level labeling task and achieved remarkable success. A number of these CNN-based methods for segmentation are region-proposal-based methods[20, 52], which first generate region proposals and then assign category labels to each. Very recently, FCN[50, 52, 53] has become a popular choice for semantic segmentation, because of its effective feature generation and end-to-end training.

3.1 Region proposal based approaches

In R-CNN[20], semantic image segmentation is performed based on the object detection results. The detection system is demonstrated in Fig. 20. Taking an input image, selective search[21] is used to extract around 2 000 bottom-up region proposals. A convolutional neural network takes the affine warped regions as the input to generate a fixed-size CNN feature, regardless of the region′s shape. Then, several class-specific linear SVMs are used to classify different regions. Finally, the category-specific mask of the surviving candidate is predicted using the features from the CNN.

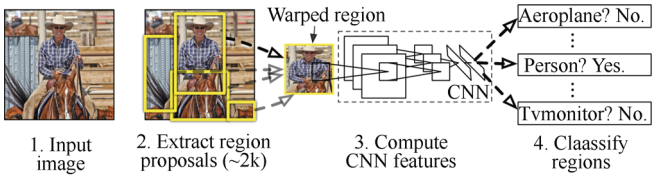


Fig. 20 Framework of region CNN. This figure is from the original paper [20].

Similar to R-CNN, simultaneous detection and segmentation (SDS)[52] starts the semantic segmentation with category-independent bottom-up object proposals. The framework of SDS is shown in Fig. 21. Multisale combinatorial grouping (MCG)[54] is chosen in the paper to generate 2 000 region candidates per image. Two CNNs (bBox CNN and region CNN) are trained to extract the features from the bounding box of the region and the cropped, warped region with the background of the region masked out (with the mean image). Compared to using the same CNN for both inputs (image windows and region masks), using separate networks where each network is fine tuned for its respective role dramatically improves the performance.

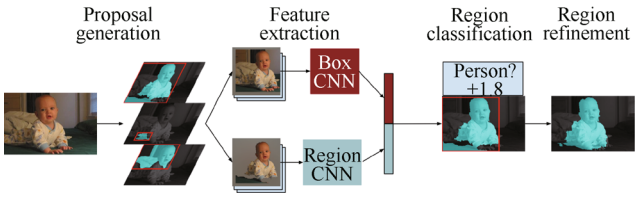


Fig. 21 Framework of simultaneous detection and segmentation. This figure is from the original paper [52].

A region classifier is then trained using the CNN features to assign a score for each category to each candidate. To generate the final semantic segmentation, SDS first learns to predict a coarse, top-down figure-ground mask for each region. The final segmentation is generated by projecting the coarse mask to superpixels by assigning to each superpixel the average value of the coarse mask in the superpixel.

3.2 FCN based approaches

DAG[51] is a fully convolutional network (FCN) trained end-to-end, pixels-to-pixels on semantic segmentation. Fig. 22 demonstrates framework of DAG net. It transforms

the fully connected layers into convolutional layers and enables a classification net to output a heatmap. A spatial loss is used to train the FCN end-to-end efficiently.

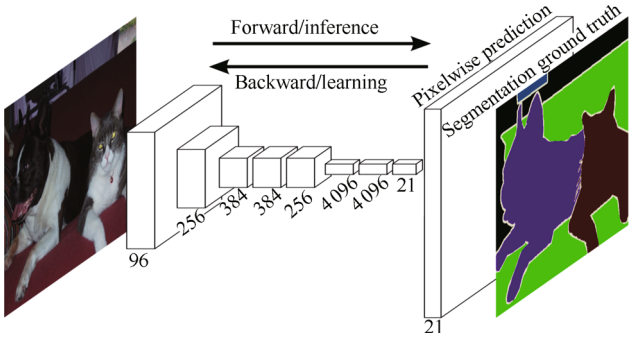


Fig. 22 Framework of DAG nets. This figure is from the original paper [51].

This approach does not make use of pre-processing and post-processing complications, including superpixels[49, 52], proposals[52, 55], or post-hoc refinement by random fields or local classifiers[49, 55].

The deconvolution networks as shown in Fig. 23 are composed of two parts: convolution and deconvolution networks. The convolution network corresponds to the feature extractor that transforms the input image to multi-dimensional feature representation, whereas the deconvolution network is a shape generator that produces object segmentation from the features extracted from the convolution network. The final output of the network is a probability map with the same size as the input image, indicating the probability of each pixel that belongs to one of the predefined classes.

VGG 16-layer net[9] is employed for the convolutional part with its last classification layer removed. The deconvolution network has 13 convolutional layers altogether, where rectification and pooling operations are sometimes performed between convolutions, and 2 fully connected layers are augmented at the end to impose class-specific projection. The deconvolution network is a mirrored version of the convolution network, and has multiple series of unpooling, deconvolution, and rectification layers. In contrary to the convolution network that reduces the size of activations through feed-forwarding, deconvolution network enlarges the activations through the combination of unpooling and deconvolution operations. Finally, the dense pixel-wise class prediction map is constructed through multiple series of unpooling, deconvolution and rectification operations.

DeepLab[57] employed the convolution with upsampled filters or "atrous convolution" as a powerful tool for image segmentation. Atrous convolution explicitly controls the resolution at which feature responses are computed within deep convolutional neural networks. It also effectively enlarges the field of view of filters to incorporate larger context without increasing the number of parameters or the amount of computation. The atrous spatial pyramid pooling (ASPP) is used to segment objects at multiple scales. ASPP probes an incoming convolutional feature layer with filters at multiple sampling rates and effective fields-of-views, thus capturing objects as well as image context at multiple scales. Another contribution of DeepLab is that it improves the localization of object boundaries by combining methods from DCNNs and a fully connected conditional random field (CRF), which is shown both qualitatively and quantitatively to improve localization performance. The framework of DeepLab model is illustrated as Fig. 24.

4 Conclusions

The paper surveys some recent progress in deep learning based fine-grained image classification and semantic segmentation. Several general convolutional neural networks are first introduced including the AlexNet, VGG net and GoogLeNet. They can be directly adapted to find-grained image classification. Since the subtle differences of visually similar fine-grained objects usually exist in some common parts, many approaches resort to deep learning technology to boost the performance of part localization, while some approaches integrate the part localization into the deep learning framework and can be trained end-to-end. Some fine-grained classification approaches also combine multiple neural networks to gain more classification capability for fine-grained images. By integrating the attention mechanism, some visual attention based approaches can automatically localize the most discriminative regions of the fine-grained images without using any bounding box or part

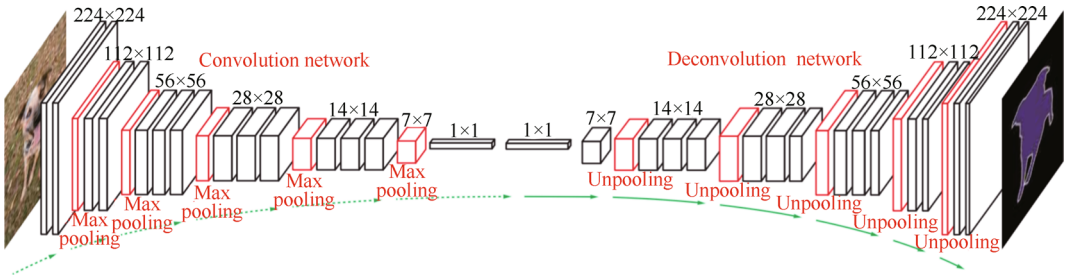


Fig. 23 Framework of deconvolution networks. This figure is from the original paper [56].

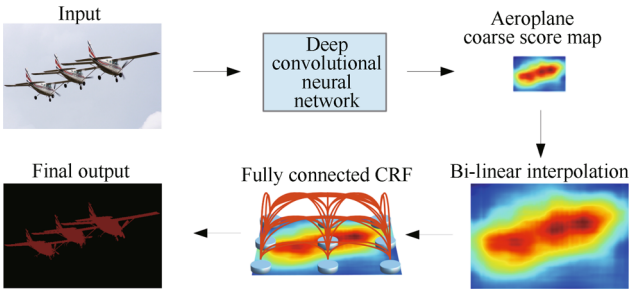


Fig. 24 Framework of DeepLab. This figure is from the original paper [57].

annotation. The approaches for possible growth of the fine-grained image classification include: 1) Using deeper neural networks to boost the performance. Residual Networks are one of the recently proposed deep neural networks with a depth of up to 152 layers which is 8 times deeper than VGG nets but still having lower complexity. An ensemble of these residual nets achieves 3.57 % error on the ImageNet test set. Using these deeper networks as the feature extractor or base networks will certainly improve the accuracy. 2) Using reinforcement learning technique[48, 58] to learn task-specific policies will also benefit the fine-grained object classification, because they can learn the part localization and discriminative representation in an end-to-end way, and they do not require manually labeled object. For the semantic segmentation, region proposal based approaches and FCN based approaches are introduced respectively.

Appendix

Links to codes

- AlexNet
  `https://github.com/BVLC/caffe/tree/master/models/bvlc\_alexnet`
- VGGNet
  `http://www.robots.ox.ac.uk/~vgg/research/very_deep`
- GoogLeNet
  `https://github.com/BVLC/caffe/tree/master/models/bvlc_googlenet`
- Deep residual networks
  `https://github.com/KaimingHe/deep-residual-networks`
- Part-based RCNNs for fine-grained category detection
  `https://github.com/n-zhang/part-based-RCNN`
- Fine-grained classification via mixture of deep convolutional neural networks
  `https://github.com/zongyuange/MixDCNN`
- Bilinear CNN models for fine-grained visual recognition
  `https://bitbucket.org/tsungyu/bcnn.git`
- Fully convolutional networks for semantic segmentation
  `https://github.com/shelhamer/fcn.berkeley-urlvision.org`
- Simultaneous detection and segmentation
  `https://github.com/bharath272/sds_eccv2014`
- DeepLab
  `https://bitbucket.org/deeplab/deeplab-public`

References

[1] A. Graves, M. Liwicki, S. Fernández, R. Bertolami, H. Bunke, J. Schmidhuber. A novel connectionist system for unconstrained handwriting recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 5, pp. 855–868, 2009.

[2] H. Sak, A. W. Senior, F. Beaufays. Long short-term memory recurrent neural network architectures for large scale acoustic modeling. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association*, ISCA, Singapore, pp. 338–342, 2014.

[3] W. Zaremba, I. Sutskever, O. Vinyals. Recurrent neural network regularization. arXiv:1409.2329, 2014.

[4] K. Cho, B. van Merrienboer, D. Bahdanau, Y. Bengio. On the properties of neural machine translation: Encoder-decoder approaches. arXiv:1409.1259, 2014.

[5] G. B. Zhou, J. X. Wu, C. L. Zhang, Z. H. Zhou. Minimal gated unit for recurrent neural networks. *International Journal of Automation and Computing*, vol. 13, no. 3, pp. 226–234, 2016.

[6] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Computation*, vol. 1, no. 4, pp. 541–551, 1989.

[7] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li, F. F. Li. ImageNet: A large-scale hierarchical image database. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Miami, USA, pp. 248–255, 2009.

[8] A. Krizhevsky, I. Sutskever, G. E. Hinton. ImageNet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems 25*, NIPS, Lake Tahoe, USA, pp. 1097–1105, 2012.

[9] K. Simonyan, A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556, 2014.

[10] C. Szegedy, W. Liu, Y. Q. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich. Going deeper with convolutions. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Boston, USA, pp. 1–9, 2014.

[11] A. S. Razavian, H. Azizpour, J. Sullivan, S. Carlsson. CNN features off-the-shelf: An astounding baseline for recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, Columbus, USA, pp. 512–519, 2014.

[12] L. X. Xie, R. C. Hong, B. Zhang, Q. Tian. Image classification and retrieval are ONE. In *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, ACM, New York, USA, pp. 3–10, 2015.

[13] L. X. Xie, L. Zheng, J. D. Wang, A. Yuille, Q. Tian. Interactive: Inter-layer activeness propagation. In *Proceedings of IEEE International Conference on Computer Vision and Pattern Recognition*, IEEE, Las Vegas, USA, pp. 270–279, 2016.

[14] T. Berg, P. N. Belhumeur. POOF: Part-based one-vs-one features for fine-grained categorization, face verification, and attribute estimation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Portland, USA, pp. 955–962, 2013.

[15] J. X. Liu, A. Kanazawa, D. Jacobs, P. Belhumeur. Dog breed classification using part localization. In *Proceedings of the 12th European Conference on Computer Vision*, Springer, Florence, Italy, vol. 7572, pp. 172–185, 2012.

[16] S. L. Yang, L. F. Bo, J.Wang, L. G. Shapiro. Unsupervised template learning for fine-grained object recognition. *Advances in Neural Information Processing Systems 25*, NIPS, Lake Tahoe, USA, pp. 3122–3130, 2012.

[17] E. Gavves, B. Fernando, C. G. M. Snoek, A. W. M. Smeulders, T. Tuytelaars. Fine-grained categorization by alignments. In *Proceedings of IEEE International Conference on Computer Vision*, IEEE, Sydney, Australia, pp. 1713–1720, 2013.

[18] Y. N. Chai, V. Lempitsky, A. Zisserman. BiCoS: A Bi-level co-segmentation method for image classification. In *Proceedings of IEEE International Conference on Computer Vision*, IEEE, Barcelona, Spain, pp. 2579–2586, 2011.

[19] N. Zhang, J. Donahue, R. Girshick, T. Darrell. Part-based R-CNNs for fine-grained category detection. In *Proceedings of the 13th European Conference on Computer Vision*, Springer, Zurich, Switzerland, vol. 8689, pp. 834–849, 2014.

[20] R. Girshick, J. Donahue, T. Darrell, J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Columbus, USA, pp. 580–587, 2014.

[21] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, A. W. M. Smeulders. Selective search for object recognition. *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[22] K. J. Shih, A. Mallya, S. Singh, D. Hoiem. Part localization using multi-proposal consensus for fine-grained categorization. arXiv:1507.06332, 2015.

[23] C. L. Zitnick, P. Dollár. Edge boxes: Locating object proposals from edges. In *Proceedings of the 13th European Conference on Computer Vision*, Springer, Zurich, Switzerland, pp. 391–405, vol. 8693, 2014.

[24] S. Branson, G. Van Horn, S. Belongie, P. Perona. Bird species categorization using pose normalized deep convolutional nets. arXiv:1406.2952, 2014.

[25] S. Branson, O. Beijbom, S. Belongie. Efficient large-scale structured learning. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Portland, USA, pp. 1806–1813, 2013.

[26] S. L. Huang, Z. Xu, D. C. Tao, Y. Zhang. Part-stacked CNN for fine-grained visual categorization. arXiv:1512.08086, 2015.

[27] O. Matan, C. J. C. Burges, Y. LeCun, J. S. Denker. Multi-digit recognition using a space displacement neural network. *Advances in Neural Information Processing Systems 4*, NIPS, San Mateo, USA, pp. 488–495, 1992.

[28] D. Lin, X. Y. Shen, C. W. Lu, J. Y. Jia. Deep LAC: Deep localization, alignment and classification for fine-grained recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Boston, USA, pp. 1666–1674, 2015.

[29] J. P. W. Pluim, J. B. A. Maintz, M. A. Viergever. Mutual-information-based registration of medical images: A survey. *IEEE Transactions on Medical Imaging*, vol. 22, no. 8, pp. 986–1004, 2003.

[30] Z. Y. Ge, C. McCool, C. Sanderson, P. Corke. Subset feature learning for fine-grained category classification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, Boston, USA, pp. 46–52, 2015.

[31] Z. Y. Ge, A. Bewley, C. McCool, P. Corke, B. Upcroft, C. Sanderson. Fine-grained classification via mixture of deep convolutional neural networks. In *Proceedings of IEEE Winter Conference on Applications of Computer Vision*, IEEE, Lake Placid, USA, pp. 1–6, 2016.

[32] Z. H. Wang, X. X. Wang, G. Wang. Learning fine-grained features via a CNN tree for large-scale classification. arXiv:1511.04534, 2015.

[33] D. Q. Wang, Z. Q. Shen, J. Shao, W. Zhang, X. Y. Xue, Z. Zhang. Multiple granularity descriptors for fine-grained categorization. In *Proceedings of IEEE International Conference on Computer Vision*, IEEE, Santiago, Chile, pp. 2399–2406, 2015.

[34] T. Y. Lin, A. RoyChowdhury, S. Maji. Bilinear CNN models for fine-grained visual recognition. In *Proceedings of IEEE International Conference on Computer Vision*, IEEE, Santiago, Chile, pp. 1449–1457, 2015.

[35] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, A. Vedaldi. Describing textures in the wild. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Columbus, USA, pp. 3606–3613, 2014.

[36] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzen, T. Darrel. DeCAF: A deep convolutional activation feature for generic visual recognition. arXiv:1310.1531, 2013.

[37] A. S. Razavian, H. Azizpour, J. Sullivan, S. Carlsson. CNN features off-the-shelf: An astounding baseline for recognition. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops*, IEEE, Columbus, USA, pp. 512–519, 2014.

[38] T. J. Xiao, Y. C. Xu, K. Y. Yang, J. X. Zhang, Y. X. Peng, Z. Zhang. The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Boston, USA, pp. 842–850, 2015.

[39] P. Sermanet, A. Frome, E. Real. Attention for fine-grained categorization. arXiv:1412.7054, 2014.

[40] J. Ba, V. Mnih, K. Kavukcuoglu. Multiple object recognition with visual attention. arXiv:1412.7755, 2014.

[41] X. Liu, T. Xia, J. Wang, Y. Q. Lin. Fully convolutional attention localization networks: Efficient attention localization for fine-grained recognition. arXiv:1603.06765, 2016.

[42] V. Mnih, N. Heess, A. Graves, K. kavukcuoglu. Recurrent models of visual attention. *Advances in Neural Information Processing Systems 27*, Montréal, Canada, pp. 2204–2212, 2014.

[43] B. Zhao, X. Wu, J. S. Feng, Q. Peng, S. C. Yan. Diversified visual attention networks for fine-grained object classification. arXiv:1606.08572, 2016.

[44] C. Wah, S. Branson, P. Welinder, P. Perona, S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset, Computation & Neural Systems, Technical Report, CNS-TR, California Institute of Technology, USA, 2011.

[45] S. Sharma, R. Kiros, R. Salakhutdinov. Action recognition using visual attention. arXiv:1511.04119, 2015.

[46] M. Jaderberg, K. Simonyan, A. Zisserman, K. Kavukcuoglu. Spatial transformer networks. *Advances in Neural Information Processing Systems 28*, Montréal, Canada,pp. 2017–2025, 2015.

[47] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhutdinov, R. Zemel, Y. Bengio. Show, attend and tell: Neural image caption generation with visual attention. arXiv:1502.03044, 2015.

[48] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, vol. 8, no. 3–4, pp. 229–256, 1992.

[49] C. Farabet, C. Couprie, L. Najman, Y. LeCun. Learning hierarchical features for scene labeling. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–1929, 2013.

[50] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected CRFs. arXiv:1412.7062, 2014.

[51] J. Long, E. Shelhamer, T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Boston, USA, pp. 3431–3440, 2015.

[52] B. Hariharan, P. Arbeláez, R. Girshick, J. Malik. Simultaneous detection and segmentation. In *Proceedings of the 13th European Conference on Computer Vision*, Springer, Zurich, Switzerland, vol. 8695, pp. 297–312, 2014.

[53] J. F. Dai, K. M. He, J. Sun. Boxsup: Exploiting bounding boxes to supervise convolutional networks for semantic segmentation. In *Proceedings of IEEE International Conference on Computer Vision*, IEEE, Santiago, Chile, pp. 1635–1643, 2015.

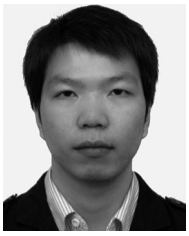
[54] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, J. Malik. Multiscale combinatorial grouping. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, IEEE, Columbus, USA, pp. 328–335, 2014.

[55] S. Gupta, R. Girshick, P. Arbeláez, J. Malik. Learning rich features from RGB-D images for object detection and segmentation. In *Proceedings of the 13th European Conference Computer Vision*, Springer, Zurich, Switzerland, vol. 8695, pp. 345–360, 2014.

[56] H. Noh, S. Hong, B. Han. Learning deconvolution network for semantic segmentation. In *Proceedings of IEEE International Conference on Computer Vision*, IEEE, Santiago, Chile, pp. 1520–1528, 2015.

[57] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A. L. Yuille. DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs. arXiv:1606.00915, 2016.

[58] D. R. Liu, Hong-Liang Li, L. D. Wang. Feature selection and feature learning for high-dimensional batch reinforcement learning: A survey. *International Journal of Automation and Computing*, vol. 12, no. 3, pp. 229–242, 2015.



Bo Zhao received the B. Sc. degree in networking engineering from Southwest Jiaotong University in 2010. He is a Ph. D. degree candidate at School of Information Science and Technology, Southwest Jiaotong University, China. Currently, he is at the Department of Electrical and Computer Engineering, National University of Singapore, Singapore as a visiting scholar.

His research interests include multimedia, computer vision and machine learning.

E-mail: zhaobo@my.swjtu.edu.cn
ORCID iD: 0000-0002-2120-2571

Jiashi Feng received the B. Eng. degree from University of Science and Technology, China in 2007, and the Ph. D. degree from National University of Singapore, Singapore in 2014. He was a postdoc researcher at University of California, USA from 2014 to 2015. He is currently an assistant professor at Department of Electrical and Computer Engineering, National University of Singapore, Singapore.

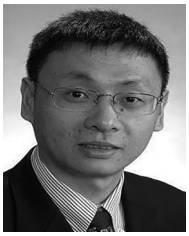
His research interests include machine learning and computer vision techniques for large-scale data analysis. Specifically, he has done work in object recognition, deep learning, machine learning, high-dimensional statistics and big data analysis.

E-mail: elefjia@nus.edu.sg

Xiao Wu received the B. Eng. and M. Sc. degrees in computer science from Yunnan University, China in 1999 and 2002, respectively, and the Ph. D. degree in computer science from City University of Hong Kong, China in 2008. He is an associate professor at Southwest Jiaotong University, China. He is the assistant dean of School of Information Science and Technology, and the head of Department of Computer Science and Technology. Currently, he is at School of Information and Computer Science, University of California, USA as a visiting associate professor. He was a research assistant and a senior research associate at the City University of Hong Kong, China from 2003 to 2004, and 2007 to 2009, respectively. From 2006 to 2007, he was with the School of Computer Science, Carnegie Mellon University, USA as a visiting scholar. He was with the Institute of Software, Chinese Academy of Sciences, China, from 2001 to 2002. He received the second prize of Natural Science Award of the Ministry of Education, China in 2015.

His research interests include multimedia information retrieval, image/video computing and data mining.

E-mail: wuxiaohk@gmail.com (Corresponding author)
ORCID iD: 0000-0002-8322-8558



Shuicheng Yan is currently an associate professor at the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, the founding lead of the Learning and Vision Research Group (http://www.lv-nus.org). He has authored/co-authored nearly 400 technical papers over a wide range of research topics, with Google Scholar citation>12 000 times. He is ISI highly-cited researcher 2014, and IAPR Fellow 2014. He has been serving as an associate editor of *IEEE Transactions on Knowledge and Data Engineering*, *Computer Vision and Image Understanding* and *IEEE Transactions on Circuits and Systems for Video Technology*. He received the Best Paper Awards from ACM MM′13 (Best paper and Best student paper), ACM MM′12 (Best demo), PCM′11, ACM MM′10, ICME′10 and ICIMCS′09, the runnerup prize of ILSVRC′13, the winner prizes of the classification task in PASCAL VOC 2010–2012, the winner prize of the segmentation task in PASCAL VOC 2012, the honorable mention prize of the detection task in PASCAL VOC′10, 2010 TCSVT Best Associate Editor (BAE) Award, 2010 Young Faculty Research Award, 2011 Singapore Young Scientist Award, and 2012 NUS Young Researcher Award.

His research interests include machine learning, computer vision and multimedia.

E-mail: eleyans@nus.edu.sg