Using Coarse Label Constraint for Fine-Grained Visual Classification

Chaohao Lu[1] and Yuexian Zou[1,2(✉)]

[1] ADSPLAB, School of ECE, Peking University, Shenzhen, China
zouyx@pkusz.edu.cn
[2] Peng Cheng Laboratory, Shenzhen, China

Abstract. Recognizing fine-grained categories (e.g., dog species) relies on part localization and fine-grained feature learning. However, these classification methods use fine labels and ignore the structural information between different classes. In contrast, we take into account the structural information and use it to improve fine-grained visual classification performance. In this paper, we propose a novel coarse label representation and the corresponding cost function. The new coarse label representation idea comes from the category representation in the multi-label classification. This kind of coarse label representation can well express the structural information embedded in the class hierarchy, and the coarse labels are only obtained from suffix names of different category names, or given in advance like CIFAR100 dataset. A new cost function is proposed to guide the fine label convergence with the constraint of coarse labels, so we can make full use of this kind of coarse label supervised information to improve fine-grained visual classification. Our method can be generalized to any fine-tuning task; it does not increase the size of the original model; and adds no overhead to the training time. We conduct comprehensive experiments and show that using coarse label constraint improves major fine-grained classification datasets.

Keywords: Fine-grained classification · Multi-label learning
Coarse label constraint

1 Introduction

Fine-grained Visual categorization (FGVC) aims to distinguish very similar categories, such as species of birds [1, 2], dogs [3] and flowers [4], or models of vehicles [5]. These tasks are different from conventional image classification [6] in that they require expert level knowledge to find subtle differences. FGVC has a wide range of applications in many fields, such as image captioning, image generation, and machine teaching [7].

Most of the prior work in FGVC has focused on handling the variations in pose, lighting, viewpoint using part localization techniques [8, 9], attention mechanism [10–12], fine-grained feature extractors [7], and by adding training dataset with noisy data from web [13]. We observed that prior work in FGVC pays much attention to part localization or neural network architecture, and the supervised information used

I. Kompatsiaris et al. (Eds.): MMM 2019, LNCS 11296, pp. 266–277, 2019.
https://doi.org/10.1007/978-3-030-05716-9_22

includes fine labels, bounding box. We call them flat classification because they use fine label as supervised information and the fine labels do not take into account the structural information embedded in the class hierarchy.

The common taxonomy is hierarchical and structural. We take bird classification as an example, there are order, family, genus and species under the bird class, and the bird species are the specific bird label. In a fine-grained visual classification task, we need to distinguish different species of birds, not the corresponding family and genus. This makes me think about two questions. Firstly, can we use the biological taxonomy to promote fine-grained classification; secondly, how to realize it.

This paper answers the above two questions from a very basic point of view. We have created a new coarse label representation and the corresponding cost function to take advantage of this kind of coarse label supervised information. Coarse label representation method draws on multi-label classification [14, 15]. This coarse label can represent the structural relationship between categories, including the parent-child relationship between coarse label and fine label, parallel relationships between different fine labels that belongs to a same coarse label. The new cost function can make use of structural relationship between coarse labels and fine labels, using coarse label supervised information to constrain the error of fine label classification to a smaller interval and improving classification accuracy. Through our new label representation and cost function, we can improve any existing network and achieve 1%–7% improvement on the existing network. It does not change the size of original model and adds no overhead to the training time.

Our main contribution can be summarized as follows:

- We create a new coarse label representation that can well express the structural information embedded in the class hierarchy.
- We propose a new cost function to take advantage of this kind of coarse label supervised information.
- We conduct comprehensive experiments on four datasets (CUB Birds [1], Stanford Dogs [3], NABirds [2], CIFAR100), and achieve 1%–7% improvement on major fine-grained classification datasets.

The rest of the paper is organized as follows. Section 2 describes the related work. Section 3 introduces the proposed method. Section 4 introduces the datasets and networks. Section 5 provides the results and analysis, followed by the conclusion in Sect. 6.

2 Related Work

2.1 Fine-Grained Visual Classification

The research on fine-grained visual classification (FGVC) relies on part localization and discriminative feature learning. The most difference between FGVC task and conventional classification task is that there are subtle differences between fine-grained categories. For example, it may be the wings of birds are different in color. We use local information of the image to assist in classification, such as by extra processing of the bird's head and torso, to improve the overall classification performance [8–12, 16, 17].

Using discriminative feature extractors is also crucial for FGVC. Due to the success of convolutional neural network in conventional image classification, we can fine tune the model that pre-trained on conventional image datasets. Moreover, a bilinear structure [18, 19] is proposed to compute the pairwise feature interactions, and a boosted Deep Convolutional Neural Networks [20] is proposed to combine the merits of boosting and modern neural networks. These prior work can also be potentially combined with our method for future work.

2.2 Transfer Learning

Conventional Neural Networks trained on ImageNet [6] have been widely used for transfer learning [7]. The pre-trained network can be used as a feature extractor, or fine-tuned with the whole network. Compared with conventional image classification, the fine-grained classification datasets are much smaller. Additionally for fine-grained wildlife data collection, some species are harder to photograph, resulting in long-tails data distribution. Recently, some works using large noisy web data [13] to fine tune the network or use large fine-grained datasets [21] to fine tune the small dataset, and they have got incredible results.

2.3 Multi-label Learning

Multi-label learning [14] studies the problem where each example is represented by a single instance while associated with a set of labels simultaneously, whereas traditional multi-class learning studies the problem where each example is represented by a single instance while associated with a single labels. In a way, multi-class learning can be seen as a special case of multi-label learning. There are two main differences between our approach and multi-label learning. First, in multi-label classification, each dimension of the category vector represents whether the category appears. Assuming that there are N categories, a category representation of a multi-label category has 2^N possibilities. We use the representation rule of the multi-label category to represent the coarse label, but the amount of all coarse labels are smaller than N. Second, in multi-label learning, the output of a network is a multi-label vector; our method uses coarse labels as a kind of supervised information, and the final output is a single fine label.

3 Method

We create a novel coarse label representation that can well express the structural information embedded in the class hierarchy. Moreover, a new cost function is proposed to take advantage of this kind of coarse label supervised information.

3.1 Coarse Label Representation

The concept of coarse label is opposite to fine label. For an instance, a fine label represents the specific category it belongs to, and a coarse label is often an abstract label of several similar fine labels. We usually use extra label to describe the coarse label of an instance. This will bring extra overhead on the storage, and it is difficult to make the coarse and fine labels merge with each other during training.

CIFAR-100 dataset provides us with fine label and coarse label for each category. CIFAR100 has 100 classes containing 600 images each. The 100 classes in the CIFAR-100 are grouped into 20 super classes. Each image comes with a "fine" label (the class to which it belongs) and a "coarse" label (the superclass to which it belongs). For example, a super class called fish has 5 subcategories: aquarium fish, flatfish, ray, shark and trout. In this case, we use extra labels "fish" to represent coarse labels. Table 1 shows examples of CIFAR-100 fine labels and corresponding coarse labels.

Table 1. Example labels of CIFAR100 classes.

Super classes	Fine classes
Aquatic mammals	beaver, dolphin, otter, seal, whale
Fish	aquarium fish, flatfish, ray, shark, trout
Flowers	orchids, poppies, roses, sunflowers, tulips
Food containers	bottles, bowls, cans, cups, plates
Fruit and vegetables	apples, mushrooms, oranges, pears, sweet peppers

In multi-label learning, we use a category vector to represent an instance. Multi-label learning studies the problem where each example is represented by a single instance while associated with a set of labels simultaneously. Let's assume that there are a total of N categories, the position i of a multi-label vector is 1, indicating that the class i belongs to this instance. A N-dimensional multi-label vector that represent an instance looks like this:

$$[0, 0, 1, 0, 0 \ldots 1, 0, 0, 1, 0, 0] \tag{1}$$

In conventional fine-grained visual classification, an instance is associated with a single label. Let's assume that there are a total of N categories, the position i of a multi-class vector is 1, indicating that the class i belongs to this instance. A N-dimensional multi-class vector that represent an instance looks like this:

$$[0, 0, 0 \ldots 0, 1, 0, 0, 0, 0] \tag{2}$$

Each fine label has only one corresponding coarse label, while each coarse label has at least one fine label. We assume that there are a total of N fine labels. For a coarse label, we assume that there are n fine labels corresponding to the coarse label. The n fine labels are a_1, a_2 ... a_n, respectively. We use a multi-label vector to represent a fine-grained label while the position i of the vector is 1 indicating that it belongs to the class i. And the final coarse label can be a union of the label vectors of all corresponding fine labels. A N-dimensional coarse label vector that represent an instance looks like this:

$$[1, 1, 0, 0\ldots 0, 0, 1, 0, 0] \tag{3}$$

All the N-dimensional fine label vectors corresponding to this coarse label are given as follows:

$$\begin{aligned}
&[1, 0, 0, 0\ldots 0, 0, 0, 0, 0]\\
&[0, 1, 0, 0\ldots 0, 0, 0, 0, 0]\\
&\ldots\\
&[0, 0, 0, 0\ldots 0, 0, 1, 0, 0]
\end{aligned} \tag{4}$$

In taxonomy, the relationship of biological categories are often represented by parent-child nodes, which require a multi-layer tree structure for storage. The tree structure can represent many relationships such as parent-child relationship between categories. But this kind of category information is difficult to be effectively utilized in machine learning because of the tree structure. In machine learning, the supervised information is often a simple category tag rather than a complex data structure. Instead, our proposed coarse label representation approach is able to make use of the structural relationships between categories. Specifically, our proposed new coarse label representation contains the structural information between the fine labels. The structural information here includes not only the parent-child relationship of the fine labels corresponding to the coarse label, but also the parallel relationship between fine labels that belonging to the same coarse label.

3.2 Cost Function

A new cost function is proposed to take advantage of this kind of coarse label supervised information. This cost function combines sigmoid cross entropy with softmax cross entropy, which makes good use of coarse labels to improve fine label classification. The cost function is an important indicator for evaluating the training effect, and the adjustment of the network parameters minimizes the cost function. In the training of convolutional neural networks, commonly used cost functions include softmax cross entropy, sigmoid cross entropy and so on.

For a convolutional neural network with parameters θ that produces the conditional probability distribution $p_\theta(x)$ over N classes for input image x. For softmax cross entropy, the ground truth we use is fine label y, a multi-class vector representation, then we compute softmax cross entropy between conditional probability distribution $p_\theta(x)$ and ground truth y, where

$$L_{softmax}(x, y) = -\sum\nolimits_{i=1}^{N} y_i * \log p_\theta(x)[i] \tag{5}$$

The sigmoid cross entropy measures the probability error in discrete classification tasks in which each class is independent and not mutually exclusive. For instance, one could perform multi-label classification where a picture can contain both a house and a tree at the same time. For sigmoid cross entropy, the ground truth we use is a coarse label z, the new proposed coarse label representation, then we compute sigmoid cross entropy between conditional probability distribution $p_\theta(x)$ and ground truth z, where

$$L_{sigmoid}(x,z) = \sum_{i=1}^{N} \max(p_\theta(x)[i], 0) - p_\theta(x)[i] * z_i + log(1 + exp(-abs(p_\theta(x)[i]))) \quad (6)$$

We formulate the final cost function for an input image x with fine label y and coarse label z as:

$$L_{final} = a * L_{softmax}(x,y) + b * L_{sigmoid}(x,z) \quad (7)$$

The final cost function in (7) consists of two parts, the first part is $L_{softmax}$ and the second part is $L_{sigmoid}$. Obviously, in conventional image classification, we usually use $L_{softmax}$ as the cost function. We minimize $L_{softmax}$ with fine labels and minimize $L_{sigmoid}$ with coarse labels. The coarse label contains the parallel relationships between different fine labels that belongs to the same coarse label. And in the process of minimizing the cost function, we use $L_{sigmoid}$ for making the errors constrained in similar categories and use $L_{softmax}$ for making the model learn how to correctly classify fine labels.

The parameters (a and b) in (7) are two super parameters which are the controlling parameter in measuring the effect of $L_{softmax}$ and $L_{sigmoid}$ on L_{final}. In this study, we usually set a to 1 and vary b.

4 Experimental Details

We use open-source TensorFlow [22] and Pytorch frameworks to implement and train all the models on Multiple NVIDIA TITAN X GPUs. We will have a brief introduction of three fine-grained classification datasets and one standard image classification dataset used in our paper, we will also briefly introduce the neural network used for fine tuning in this paper.

4.1 Datasets

Fine-Grained Visual Classification Datasets. We evaluate our method using three standard Fine-grained Visual Classification (FGVC) datasets.

The Caltech-UCSD Birds (CUB200) dataset has 5,994 training and 5,794 test images across 200 fine classes of birds. We only observe whether the suffixes of the category names are the same, and then divide them into 70 super classes. So for Caltech-UCSD Birds dataset, there are totally 200 fine labels and 70 coarse labels. The NABirds dataset contains 23,929 training and 24,633 test images across 555 bird categories, and we divide them into 156 super classes using the same method. The Stanford Dogs dataset has 12,000 training and 8,580 test images across 120 classes (dog breeds), and we divide them into 72 super classes using the same method. Labels of each dataset are in Table 2.

Table 2. Labels of each dataset.

Dataset	Fine labels	Coarse labels	Division method
CUB200 [1]	200	70	Same class name suffix
NABirds [2]	555	156	Same class name suffix
Stanford Dogs [3]	120	72	Same class name suffix
CIFAR100	100	20	Official division

Standard Image Classification Datasets. We also utilize a standard image classification dataset CIFAR-100 for study. The CIFAR-100 dataset has 100 classes containing 600 images each. There are 500 training images and 100 testing images per class. The 100 classes in the CIFAR-100 are grouped into 20 super classes. Each image comes with a "fine" label (the class to which it belongs) and a "coarse" label (the superclass to which it belongs). We use the official division as our division.

4.2 Network Architectures

We fine tune three types of network architecture for fine-grained visual classification datasets: VGG19 [23], Resnet50 [24] and Inception-V3 [25]. We fine tune VGG19 and Wide Residual Network [26] for the standard image classification dataset.

VGG. In fine-grained visual classification, VGG is a very common network, including Bilinear CNN, which uses VGG as feature extractor. VGG uses a deeper network structure than AlexNet [27], it won the first and second place respectively in the 2014 ILSVRC localization and classification. The VGG network is very deep, usually with 16–19 layers and a convolution kernel size of 3×3. We use a 19-layer VGG network.

Residual Network. Residual Network has residual connections that reduce the optimization difficulties and enable the network to be much deeper. We use the ResNet with 50 layers as representative for Residual Networks in our experiments.

Inception-V3. The Inception module was firstly proposed as GoogleNet that was designed to be very efficient. Inception module was then further optimized by using Batch Normalization, residual connections and so on. We use Inception-V3 as representative for Inception Networks in our experiments.

Wide Residual Network. Because ResNets is too deep, many residual blocks can only provide a small amount of information, or only a small number of blocks can learn important information. The author thinks that ResNet's main ability comes from the Residual block, and the depth increase is only an aid. He decreased depth and increased width of residual networks. The proposed 16-layer Wide Residual Network can be similar to the 1000-layer ResNet.

5 Results and Analysis

5.1 Experiments on Fine-Grained Visual Classification Datasets

We first describe our results on three fine-grained visual classification datasets. We fine tune three network models that pre-trained on the ImageNet. Our experiment is divided into two steps, the first step is to use only the fine label for fine-tuning, and the second step we use coarse label to constrain error. In the second step, we set the final loss parameters $a = 1$, $b = 1$, and training epochs is the same as the first step. We observe that our approach improves performance for any dataset, any pre-trained network. The results are in Tables 3, 4 and 5.

Table 3. CUB200 accuracies.

Network	Only fine labels	With coarse labels as supervision
VGG19	72.80%	79.67%
ResNet50	77.67%	79.31%
Inception-V3	80.64%	**81.72%**

Taking CUB200 dataset as an example, if VGG19 is used as the pre-trained model, the accuracy rate is increased by nearly 7 percentage points after using the coarse label constraint, and 2 percentage points is increased by using ResNet50 or Inception-V3. The result of VGG19 on the ImageNet is also worse than ResNet50 or Inception-V3, which indicates that VGG's feature extraction capability is not as strong as ResNet50 or Inception-V3, and we have greatly improved this by using the coarse label constraint. Through the improvement of our method, VGG19 can achieve the same effect as Resnet50.

In (7), the final loss is composed of $a * L_{softmax}$ and $b * L_{sigmoid}$, and parameters a, b affect the speed ratio of back propagation during training. We usually set a to 1, and change b. If b is greater than a, then the effect of the sigmoid cross entropy cost function is greater. In our experiment, we find that the value of b is usually larger than a, which makes the network get a better result. This will lead to a final increase of nearly one percentage point. For example, when we use Inception-V3 to fine tune the CUB200 dataset with coarse label constraint, we set $b = 2$ and the final result is 0.6% higher than $b = 1$. However, how to choose the values of parameters a, b still need to be manually adjusted. To get a better performance, the parameter values are not the same while we fine tune different datasets with different models, so we select $a = 1$, $b = 1$ in previous experiments.

5.2 Experiments on Standard Image Classification Dataset

We evaluate the performance of our approach on standard image classification dataset CIFAR-100 using two convolutional neural networks VGG19 and Wide Residual Network. CIFAR-100 has 100 fine classes and 20 super classes, with each super class containing five finer divisions. The results are in Table 6.

Table 4. NABirds accuracies.

Network	Only fine labels	With coarse labels as supervision
VGG19	73.54%	75.10%
ResNet50	77.20%	77.93%
Inception-V3	75.29%	**78.49%**

Table 5. Stanford Dogs accuracies.

Network	Only fine labels	With coarse labels as supervision
VGG19	76.90%	79.15%
ResNet50	79.61%	80.27%
Inception-V3	77.28%	**81.90%**

Table 6. CIFAR-100 accuracies.

Network	Only fine labels	With coarse labels
VGG19	71.95%	73.25%
Wide residual network	80.75%	81.82%

As shown in Fig. 1, after the introduction of the constraint mechanism, the test set accuracy of our network is steadily higher than original Wide Residual Network (WRN), which indicates that this constraint mechanism does improve performance of original WRN.

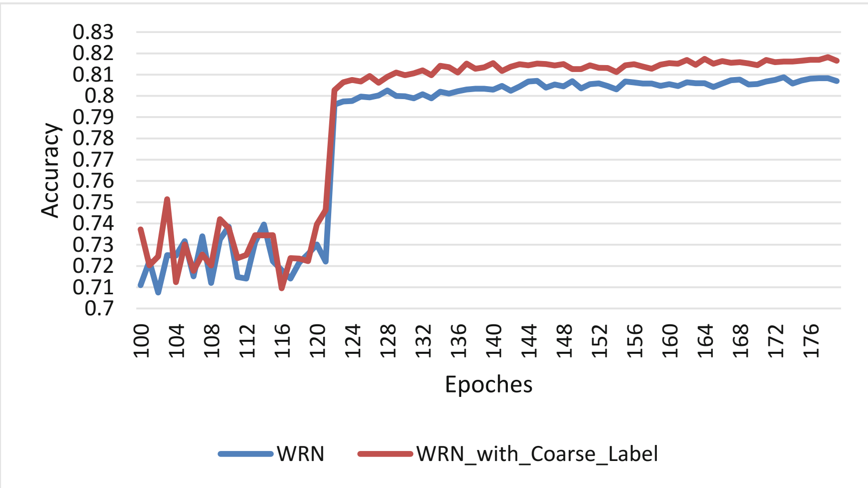

Fig. 1. Test accuracies after 100 epochs, training CIFAR-100 dataset from scratch, using Wide Residual Network (WRN) and WRN with coarse label constraint.

In our experiments, we set the same learning rate and total epoch for original networks and networks with coarse label constraint. We observe that the accuracy curve is very consistent. This shows that after the introduction of the constraint mechanism, there is no variability in the convergence of the network. Moreover, In the initial training phase, the network with coarse label constraint converges faster and the accuracy increases faster. We can see that the network with coarse label constraints can significantly accelerate convergence and promote the convergence of the entire network in the right direction. Comparison with existing methods in Table 7.

Table 7. CIFAR-100 comparison.

Methods	Test error
NIN [28]	35.67%
ResNet [24]	22.71%
WRN [26]	19.25%
WRN with coarse label	**18.18%**

6 Conclusion

In this work, we create a novel coarse label representation that can well express the structural information embedded in the class hierarchy. And we propose the corresponding cost function that take advantage of this kind of coarse label supervised information by guiding the fine label convergence with the constraint of coarse labels. We conduct comprehensive experiments in three fine-grained visual classification datasets and a standard image classification dataset, experimental results show that our method can accelerate network convergence and stably improve the original network performance.

Using coarse label constraint is easy to implement and can be generalized to any fine-tuning task; it does not increase the size of the original model and adds no overhead to training time. Therefore, our method should be beneficial to a wide range of pre-trained CNN models. In the future, we plan to combine our approach with existing methods to reduce the classification error.

Acknowledgment. This paper was partially supported by the Shenzhen Science & Technology Fundamental Research Program (No: JCYJ20160330095814461) & Shenzhen Key Laboratory for Intelligent Multimedia and Virtual Reality (ZDSYS201703031405467).

References

1. Wah, C., Branson, S., Welinder, P., Perona, P., Belongie, S.: The Caltech-UCSD Birds 200-2011 Dataset. California Institute of Technology (2011)
2. Horn, G.V., et al.: Building a bird recognition app and large scale dataset with citizen scientists: the fine print in fine-grained dataset collection. In: Computer Vision and Pattern Recognition, pp. 595–604 (2015)

3. Khosla, A., Jayadevaprakash, N., Yao, B., Li, F.: Novel dataset for fine-grained image categorization (2013)
4. Nilsback, M.E., Zisserman, A.: Automated flower classification over a large number of classes. In: Sixth Indian Conference on Computer Vision, Graphics and Image Processing, ICVGIP 2008, pp. 722–729 (2009)
5. Yang, L., Luo, P., Chen, C.L., Tang, X.: A large-scale car dataset for fine-grained categorization and verification. In: Computer Vision and Pattern Recognition, pp. 3973–3981 (2015)
6. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Li, F.F.: ImageNet: a large-scale hierarchical image database. In: IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009, pp. 248–255 (2009)
7. Yin, C., Yang, S., Chen, S., Howard, A., Belongie, S.: Large scale fine-grained categorization and domain-specific transfer learning (2018)
8. Wei, X.S., Xie, C.W., Wu, J.: Mask-CNN: localizing parts and selecting descriptors for fine-grained image recognition (2016)
9. Huang, S., Xu, Z., Tao, D., Zhang, Y.: Part-stacked CNN for fine-grained visual categorization. In: Computer Vision and Pattern Recognition, pp. 1173–1182 (2016)
10. Sun, M., Yuan, Y., Zhou, F., Ding, E.: Multi-attention multi-class constraint for fine-grained image recognition (2018)
11. Zheng, H., Fu, J., Mei, T., Luo, J.: Learning multi-attention convolutional neural network for fine-grained image recognition. In: IEEE International Conference on Computer Vision, pp. 5219–5227 (2017)
12. Fu, J., Zheng, H., Mei, T.: Look closer to see better: recurrent attention convolutional neural network for fine-grained image recognition. In: Computer Vision and Pattern Recognition, pp. 4476–4484 (2017)
13. Krause, J., et al.: The unreasonable effectiveness of noisy data for fine-grained recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9907, pp. 301–320. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46487-9_19
14. Zhang, M.L., Zhou, Z.H.: A review on multi-label learning algorithms. IEEE Trans. Knowl. Data Eng. **26**, 1819–1837 (2014)
15. Gibaja, E., Ventura, S.: A tutorial on multilabel learning. ACM Comput. Surv. **47**, 1–38 (2015)
16. Zhang, N., Donahue, J., Girshick, R., Darrell, T.: Part-based R-CNNs for fine-grained category detection. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8689, pp. 834–849. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_54
17. Xiao, T., Xu, Y., Yang, K., Zhang, J., Peng, Y., Zhang, Z.: The application of two-level attention models in deep convolutional neural network for fine-grained image classification. In: Computer Vision and Pattern Recognition, pp. 842–850 (2015)
18. Lin, T.Y., Roychowdhury, A., Maji, S.: Bilinear CNN models for fine-grained visual recognition, pp. 1449–1457 (2015)
19. Cui, Y., Zhou, F., Wang, J., Liu, X., Lin, Y., Belongie, S.: Kernel pooling for convolutional neural networks. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 3049–3058 (2017)
20. Moghimi, M., Belongie, S., Saberian, M., Yang, J., Vasconcelos, N., Li, L.J.: Boosted convolutional neural networks. In: British Machine Vision Conference, pp. 24.21–24.13 (2016)
21. Horn, G.V., et al.: The iNaturalist species classification and detection dataset (2018)
22. Abadi, M., et al.: TensorFlow: large-scale machine learning on heterogeneous distributed systems (2016)

23. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. Computer Science (2014)
24. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition, pp. 770–778 (2015)
25. Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., Wojna, Z.: Rethinking the inception architecture for computer vision, pp. 2818–2826 (2015)
26. Zagoruyko, S., Komodakis, N.: Wide residual networks (2016)
27. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: International Conference on Neural Information Processing Systems, pp. 1097–1105 (2012)
28. Lin, M., Chen, Q., Yan, S.: Network in Network. Computer Science (2014)