Modular network for object detection deep neural network optimization

Anonymous Author(s)
Affiliation
Address
email

Abstract

We present a novel modular object detection convolutional neural network that significantly improves the accuracy of object detection. The network consists of two stages in a hierarchical structure. The first stage is a network that detects general classes. The second stage consists of separate networks to refine the classification and localization of each of the general classes objects. Compared to a state of the art object detection networks the classification error in the modular network is improved by approximately 3-5 times, from 12% to 2.5 %-4.5%. This network is easy to implement and has a 0.94 mAP. The network architecture can be a platform to improve the accuracy of widespread state of the art object detection networks and other kinds of deep learning networks. We show that a deep learning network initialized by transfer learning becomes more accurate as the number of classes it later trained to detect becomes smaller.

1 Introduction

In this paper, we present a novel highly accurate deep learning network for computer vision object detection. In particular, for fine grained object detection. There is constant effort to increase the accuracy of deep learning objects detection networks. A major topic in object detection is fine grain object detection objects for detecting differences between similar object classes .

The main principles that guide the building of our network are modularity and hierarchy. Our object detection network denoted as modular network, consists of two stages, the first stage is an object detection network for detecting multi classes objects where the classes are general. The second stage consists of separate object detection networks, each one of them trained to detect only similar and related classes that belong to one of the general classes of the first stage network. Images in the first stage with detected objects that belong to one of the general classes are passed on to the appropriate network in second stage for detailed identification of the object's kind and location. We compared the detection results of our modular network to a state of the art multi class object detection network which was trained to detect the same classes as the modular network. The experiments showed that our modular network has significantly higher accuracy.

Our contributions in this paper are: 1) A simple to implement highly accurate, modular and hierarchical network for fine grained object detection. 2) We show both experimentally and theoretically that a deep learning network designed to detect a small number of classes and initially trained by transfer learning is more accurate than a network trained on more classes.

The modular network architecture suggested in this paper can be used to increase the accuracy of state of the art object detection networks by integrating them as parts of the building blocks of this network and without changing the intensive optimizations carried out on them. Other types of networks can improve their accuracy by inserting them into this modular network platform.

2 Related Work

2.1 Object detection

Notable convolutional neural networks for object detection are [14, 10, 12, 18]. Faster R-CNN [13]that consists of: a classification network, a region proposal network which divides the image into rectangular regions, followed by regression for additional accuracy in classification and location. . Most of the state of the art object detection networks include a core image classification network such as Alexnet

[8], VGG [16] or Resnet [3] these networks use transfer learning based on the training on a large image data set such as Imagenet [15] and Coco [9].

2.2 Hierarchical structures

Hierarchical structures appear in many forms in computer vision, Fukushima [2] and Jarrett et al [7] proposed a neural network for visual pattern recognition based on a hierarchical network.

3 The modular network

3.1 Modular network architecture

We present in this paper a new modular and hierarchical object detection network. The network consists of two stages, the first stage consists of a deep learning object detection network trained to detect predetermined general classes and the second stage consists of several deep learning object detection networks each trained on more fine grained classes belong to the same single general class of the first stage network. All the building blocks networks inside the modular network trained on negative images too.

Each independent deep learning network in the modular network goes independently through complete object detection processes of training and inference. The full input image data set for inference is inserted to the first stage network, if an object in an image is detected to belong to one of this network classes the image is passed to inference by the second stage network trained to detect sub classes of this class. The purpose of the second stage network is to distinguish between objects of similar classes making more detailed classification and more accurate location of the object in the image. Each sub network in the modular network was initialized by transfer learning weights [4, 6, 11, 17, 21] trained on ImageNet database. Figure 1 shows the modular network in our experiment. The building blocks of the modular network are Faster-RCNN network [13]. In the first stage there is a single network trained to detect 5 general classes if a class object is detected in an inference image. This image with no changes as it entered the first stage network is passed to fine grained detection at the appropriate network at the second stage that trained to detect detailed classes belong to the general class detected at the first stage.

One of the main reason that makes the building blocks of our modular networks and the whole modular network are more accurate than a regular multi class network is, each of the building blocks networks inside our modular network is designated to detect fewer classes than a regular multi class network.

A possible modification of the modular network is a modular network that consists of more than two hierarchical stages.

3.2 Algorithm and deep learning network construction

a. To detect multiple classes use an object detection network trained by transfer learning. Merge similar classes labels to a general class label

b. Train this network denoted as the first stage network to detect new general classes C_i and additional negative images with no labels that don't belong to any of these general classes.

c. For each of the general classes C_i , train a second stage network on the same images used to train the detection of the general class and on negative images. This time sort and label the training images

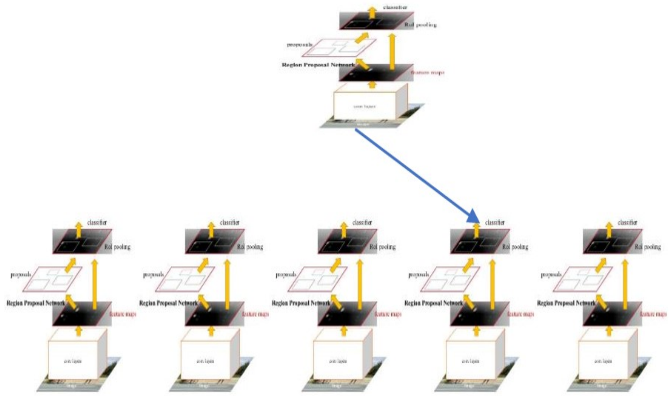

Figure 1: A modular network whose first stage is a single deep learning network trained to detect 5 general classes. Its second stage networks, consist of 5 separate networks each trained to detect 2 sub-classes of one of the general classes.

with fine grained classes all belong to this general class. It is possible to train the network on other images with objects belong to these fine grained classes.

d. Input images for inference into the first stage network. Images with objects detected to belong to a general class are passed to the second stage network dedicated to this class.

e. Input the passed images for inference in the appropriate second stage network for fine grained object classification and location.

3.3 Advantages and risk of the modular network

In each of the sub convolutional neural networks inside the modular network, there are fewer classes than in a regular network designated to detect the same number of classes as the whole modular network. Thus there are more features, filters and network parameters dedicated to detection of each class, result in better accuracy in object detection. A small number of features to identify a class causing less distinction in detection of similar classes and errors in detection of rare class objects of too, since when the amount of features is small features are formed to identify objects types that appear in many images in the training. In addition when there are a few features available to identify each class more features are formed to detect multiple classes this causes errors in fine grained object detection.

Fewer classes in object detection network mean potentially less bounding boxes of detected objects in the image, which gives fewer errors in identifying the objects and finding their locations.

In the modular network training there are less images in the input data set for each of the second stage networks because the training images are distributed over several networks. This results in less parameters and features dilution of each image or object by images and objects that not belong to the designated classes for object detection.

The advantage of the hierarchical structure of the modular network compared to detection by many few classes networks with no connection to each other is the hierarchical structure drastically cuts down the number of required inferences as the inferences are arranged in a tree structure.

The condition the accuracy of the modular network will be better than a multi class network is,

$$a < (a + \Delta_1)(a + \Delta_2) \tag{1}$$

a - the multi-label network accuracy, Δ_1 - the improvement in accuracy of the first stage of the modular network compared to the multi class network accuracy and Δ_2 - the improvement in accuracy of the second stage compared to the multi class network accuracy.

Assuming we use as the building block network of the modular networks the same type of object detection network as the multi class network. If the multi class network has low accuracy then the multi class network is preferred since the building blocks networks inside the modular network should have a very large accuracy improvement compared to the multi-class network accuracy for the whole modular network to be more accurate than the multi class network. For most state of the art object detection networks, their accuracy is high enough to use them as the building block network for the modular network and obtaining a modular network with higher accuracy compared to the selected state of the art object detection network. A risk of the modular network is false negatives defections in the network first stage. This may reduce accuracy as some images with true object may not be included in the input of the network second stage. To deal with this problem we designed a second version of the modular network specified for images sequence where the same object is assumed to appear in more than one image. The network architecture of this version denoted as modular network v2 is the same as modular network first version, v1, the difference is that after inference of all the images sequence in the first stage of the modular network. The entire images sequence is sent for inference to the networks in the second stage whose fine grained classes match the general classes of the objects detected in the first stage. In this way the loss of accuracy due to false negative detection in the first stage is reduced.

4 Convolutional neural network classification error model.

This model describes how reducing the number of classes for detection in a convolutional neural network (CNN) reduce the network classification error. Each of the building block networks inside the modular network has less classes than the regular multi class network. Let x= { $x_1 \ldots x_f$} be the features space. Let c be a set of classes c={$c_0 \ldots c_n$}. Every detection of an object in an image is defined by a set of features that are active if this object appears in an image , for example, the features set {$x_m \ldots x_p$} identify objects belong to class C_1. N - is the total number of features of the designated classes the CNN can identify . L and T are numbers of features of the designated classes the CNN can identify based on transfer learning and fine tuning [21] respectively, where each feature belong to a single class. U- is the number of features the CNN can identify that are common to several classes. N= L+T+U. When each of the designates classes has similar number of training images S- the number of features detecting a designated class, is $S \approx \frac{N}{n} \approx \frac{L+T}{n} + U$. in this approximation the amount of features for detecting a single designated class is inversely related to n the number of the CNN designated classes, the smaller is n there are more features for detecting the designated to class making this class objects detection more accurate. The parameters that determine K-the number of features a CNN can identify are: r- the numbers of parameters in the CNN, a-number of filters, d-sizes of filters, h-number of filters channels and q-number of layers in the CNN, these parameters are constant for each network. In this model every CNN has an upper bound of total number of features *sup* K(r,a,d,h,q) it can identify without increasing the classifications errors. Classification error caused by a larger amount of features than the optimal amount for the network can be for example, from two channels in the same filter where the weights pattern formed in each channel detect feature of different class. The two patterns can have partial overlap in shape and location. M and B are output matrices of the convolution of each channel with the corresponding features map channel. If in martix M there is a feature, part of this feature can appear in Matrice B too and the $\sum_{i,j \in G}(|M|_{i,j} + |B|_{i,j}) > |M|_{i,j}$ G is a set of all the i,j couples, where i and j have the values of raw and column indices of pixels include in this feature area. This Result in deformation of a feature in the filter's features map which is the sum of all the channels features maps and can cause classification error.

We use Bayes error to estimate the classification error [20, 19, 1, 5]. As an example we analysed classification of two fine grained classes C_1 and C_0. According to Bayes error estimation when there is a probability density that a feature x_i is activated, i.e there is a probability that feature x_i appears in the feature map when there is object of class C_0 and another probability density that feature x_i is activated when an object of class C_1 is in the image, the classification error caused by feature x_i is the smallest probability density between these two probabilities densities. The sum of the all the smallest probabilities densities classification errors of all the features is the classification error. Assuming for each of the features in the network the probability densities to be activates by classes C_1 or C_0 are known. The probability for error in classification is describes in equation.2, Where P(C_0), P(C_1) are the prior probability densities of class C_0 and C_1 respectively. P(x_i| C_0), P(x_i| C_1) are the conditional probability densities that feature x_i is active given the class is C_0 or

C_1 respectively. Additional criterion in equation.2 is the significance of the feature feature x_i in the classification.The criterion's weights for classes C_0 and C_1 are denoted by $w_i(C_0)$ and $w_i(C_1$) respectively. The reason is if an active feature does not influence the classification of an object it does not contribute to the classification probability of the object class. The criterion's weights values $w_i(C_0)$ and $w_i(C_1$)is based on how many times feature x_i was essential for the classification of the class from all the time this feature was activated by this class objects.

$$P_{error} = \sum_{i=1}^{N_f} min(P(x_i|C_0)P(C_0)w_i(C_0), P(x_i|C_1)P(C_1)w_i(C_1)) \tag{2}$$

The probabilities densities of the features are presented in discrete values, which we approximate as a continues graph.

In graphs 1,2 the X-axis is the features range denoted as N_f. The Y axis values is the probability density that a feature is activated. In the graph all features with probability of matching a particular class are in the same area on the x axis. Features that have a probabilities of matching the two classes will be displayed in the graph in a shared area for both classes. The classification error of classes C_0 and C_1 defined by Bayes error, is the sum of or integration, on every feature minimal probability density in C_0 and C_1 mutual area, which is the overlapping area of classes C_0 and C_1 curves.

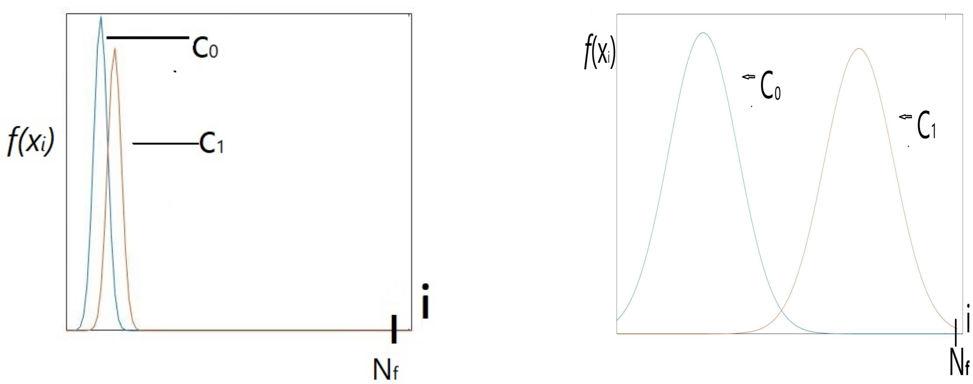

Figure 2: graph.1

Graph.1 illustrates the features probabilities densities of identifying C_0 and C_1 of a network trained to detect ten classes. The active features are about a quarter of the total features in the network. The miss classified features area is significant compared to the total areas of classes C_0 and C_1 features this indicates a large classification error. This is because there are many classes and the number of features dedicated to each class is small, result in shortage of features to identify fine grained features. Since there are many classes the total number of features exceeds the supermum number of filters for this network result in features that give false detections.

Graph.2 illustrates a network trained to detect only two classes and negative images. Most of the features detected by this network are of classes C_0 and C_1. The miss classified features area is small compared to the two classes total areas, indicating the classification error is small. The reason is the number of features for each class is large this able to train features for detecting more detailed features, which reduce the classification error.

In the first stage of the modular network that trained to detect general classes C_0 and C_1 ore both include in the same general class C_g . $C_g = C_0 \cup C_1$ this eliminates the error of miss classification between the two classes result in low classification error .Classification errors in this network are between general classes which require less details and less features do differentiate between them.

5 Experiments

5.1 Implementation

The original training image data set contains 522 images distributed between 10 classes or five couples of similar classes. The images augmented to 46,044 training images by mirroring, sharpness, brightness and contrast augmentations these images used as the training data set to both the nodular network and the multi class network. The size of each of the original images in the data is up to 800*800 pixels. The size of the output images of the network is 800*800 pixels. For the multi-class network and the building blocks networks of the modular network we used the state of art object detection network Faster R-CNN with backbone classification network VGG 16. The Faster R-CNN network is initialized by training on ImageNet 2012 database contained 1.2 million images for training and 50k validation images in 1,000 categories. The sub networks inside the modular network and the multi-class network all have the same hyper-parameters values previously optimized on different classes than the classes the networks trained to detect, to make the comparison between a multi class network and the modular network unbiased. Fine tuning training was made in all the networks inside the modular network and the multi-class network and included all the networks layers. Each of the networks trained for 40 epochs, with learning rates of: 0.001 on the first 10 epochs, 0.0001 on the next 10 epochs and 0.00001 on the last 20 epochs. The test data set contained 125 original images distributes similarly between four classes: two dog species Pekinese and Spaniel and two planets Mars and Saturn. Both the modular network and the multi class network both inferred on this test data. Most of the original images for the training and the test sets were taken from the Caltech 101 image database and the rest randomly from the internet.

5.2 Experiments results

5.2.1 multi-class network

The multi class object detection network was trained to detect ten classes and negative imges, with training loss of 0.0229 , the training loss is defined in Faster RCNN paper [13]. The multiclass network inference results are 0.87 mAP and 12% error.

5.2.2 modular network

The modular network has two stages. The first stage network was trained on the same training data set as the multi class network including the negative images but labeled with five general classes instead of the more detailed 10 classes of the multiclass network. The modular network first stage classes are dog, planet, bike, boat, bird each of these classes is a unification of a couple of similar classes from the 10 classes labeled for training by the multiclass network, the training loss is 0.0216. In the second stage each network trained on two fine grained or similar classes as the multiclass network was trained on and the same negative images. For example, one network trained on two dog species classes Pekinese and Spaniel with training loss of 0.0151 loss, a second network was trained to detect two solar planets; Mars, Saturn with training loss of 0.0170. The network was trained only on images of these classes from the initial training data set. The modular network v1 inference results are 0.94 mAP and4.5% error. The modular network v2 inference results are 0.95 mAP and 2.5% error.

The experimental results indicate the modular network is significantly more accurate than the multi-class network.

Table 1 shows experiments results of the mean average precision, mAP, of the modular networks and the multi class network , tested on the same images.

The modular network v1 AP is calculated by taking into account the images detected as false negative on the first state of the modular network thereby do not appear on the mAP of the second stage, each false negative precision is rated as zero and its part in the calculation of the whole modular network mAP is one divided by the total number of this modular network inference images. For example, in table.1, the AP of Saturn in the modular network v1 is 0.91 but the AP of Saturn in the second stage network is 0.94.

Network	dogs AP		planets AP		mAP
	Spaniel	pekinese	Mars	Saturn	
Modular net v1	0.97	0.90	0.97	0.91	0.94
Modular net v2	0.97	0.90	0.97	0.94	0.95
Multi-class	0.93	0.74	0.84	0.94	0.87
General classes modular	0.93		0.92		0.93

Table 1: Object detection average precision

Table 2 shows the experiments results of the networks classification errors. The modular network error was significantly reduced to 6% and 3% error for dogs and planets compared to 14% and 10% respectively in the multi class network.

Network	Error-dogs	Error-planets	Error-Avg
Modular network v1	6%	3%	4.5%
Modular network v2	5%	0%	2.5%
Multi-class network	14%	10%	12%
General classes Mod	1.5%	3%	2.25%

Table 2: Classification Error

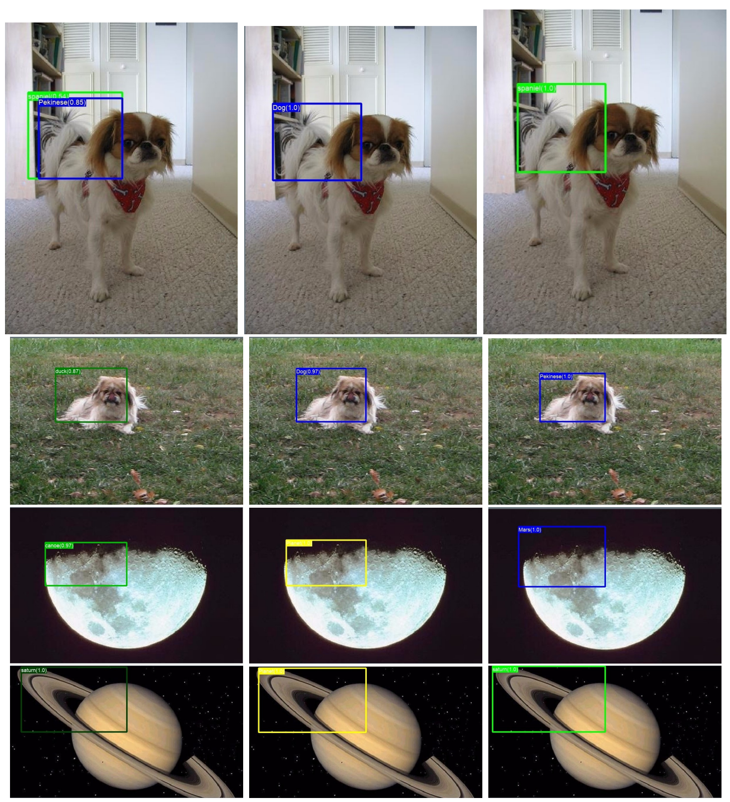

Figure 3: Left column are object detection images by the multi class network, center column are detected images by the general classes network and right column are images detected by fine grained networks

In figure.5 in the first column where the images detected by the multi class network, in the first three rows there are errors in classification. While the general classes network and the fine grained network detected the same objects correctly. It is shown in second raw images that the detection of the object location is more accurate in the right image detected by the fine grained network compared to the object location in the left image detected by the multi class network

6 Discussion

Our experiments obtained that most of the classification errors in the multi class network were between similar classes. The modular network version 1 and 2 accuracy is higher by additional 7.5%and 9.5% respectively compared to the multi-class network. This is a reduction of the classification error by 2.7 and 4.8 times respectively. We obtained that network with fewer classes is more accurate, the accuracy of a network that trained to detect only two similar objects is 9.5% higher in compared to the multi-class network that detects 10 classes. The training results indicate that as the number of classes trained to be detected by a network become smaller the training loss become smaller too. The classification error in the modular network is smaller for planets classes than dogs classes, the planet classes are less similar to each other. Thus we obtain the classification error is smaller if the fine grained classes are less similar.

A fundamental question in machine learning is what kind of learning has higher accuracy. A network that trained to detect only few focused classes or a network that trained to detect many classes of wide range subjects? We obtain that a network that initially trained on a wide range of classes by transfer learning and later trained to detect few classes by fine tuning on all the network layers is more accurate than a network initialized by transfer learning and later trained to detect larger number of classes. Previous works on transfer learning [4, 21] obtained that a network initially trained by transfer learning and later trained to detect the designated classes is more accurate compared this network when only trained to detect the designated classes. From both findings we conclude that a network initially trained by transfer learning and then designated to detect a small number of classes is more accurate than if it were designated to detect larger number of classes.

7 Conclusion

The modular network presented in this paper significantly improves object detection performances in both classification and location. This is true especially for detection require differentiating between similar classes. This modular network improves state of the art deep learning object detection networks even without requiring a change to those networks architecture and hyper-parameters. We found that reducing the number of classes a convolutional neural network is trained to detect increases the network accuracy. This modular network could be a platform for other types of deep learning networks for example, segmentation , improving their accuracy by implementing them as buildings blocks of the modular network. This modular network can be applied for fine grained pattern recognition in artificial intelligence, medical images detection and scientific research.

References

[1] W. Hou B. Juang and C. Lee. Minimum classification error rate methods for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 5, 1997.

[2] K. Fukushima. Neocognitron: A hierarchical neural network capable of visual pattern recognition. *Neural Networks*, 1, 1988.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] Minyoung Huh, Pulkit Agrawal, and Alexei A Efros. What makes imagenet good for transfer learning? *arXiv preprint arXiv:1608.08614*, 2016.

[5] H. Deng J.Zhang. Gene selection for classification of microarray data based on the bayes error. *BMC Bioinformatics*, 8, 2007.

[6] A. Karpathy. Large-scale video classification with convolutional neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[7] J. Kevin, K. Koray, R. Marc'Aurelio, and L. Yann. What is the best multi-stage architecture for object recognition? volume 12, 2009.

[8] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012.

[9] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[10] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. Ssd: Single shot multibox detector. pages 21–37, 2016.

[11] Maxime Oquab, Leon Bottou, Ivan Laptev, and Josef Sivic. Learning and transferring mid-level image representations using convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1717–1724, 2014.

[12] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[13] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39 (6):1137–1149, June 2017. ISSN 1939-3539. doi: 10.1109/TPAMI.2016.2577031.

[14] Girshick Ross. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[15] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vision*, 115(3), 2015.

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

[17] Q. Yang S.Pan. Ieee transactions on knowledge and data engineering. *CoRR*, 22, 2009.

[18] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9, 2015.

[19] Bao Xiaomin and Wang Yaming. Apple image segmentation based on the minimum error bayes decision [j]. *Transactions of the Chinese Society of Agricultural Engineering*, 5, 2006.

[20] S. Yang and B.Hu. Discriminative feature selection by nonparametric bayes error minimization. *IEEE Transactions on Knowledge and Data Engineering - TKDE*, 24, 2012.

[21] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014.