

A Self-adaptive Cascade ConvNets Model Based on Three-Way Decision Theory

Wen Shen, Zhihua Wei^(✉), Cairong Zhao, and Duoqian Miao

Department of Computer Science and Technology, Tongji University,
Shanghai 201804, China
zhihua.wei@tongji.edu.cn

Abstract. Convolutional Neural Networks (ConvNets) have a great improvement on the classification performance compared to traditional image classification technologies and become one of the leaders in computer vision. In this paper, we present a Correcting Reliability Level (CRL) supervised three-way decision (3WD) cascade model to implement image classification of mass commodity data. Our model simulates the human decision process by using 3WD to determine “accepted” or “unsure” for the classification result. When judged as “unsure”, CRL will supervise the 3WD and learn more information to make the final prediction. In addition, we introduce a Class Grouping algorithm based on feedback to learn the similarity between classes, which help us to train several expert ConvNets for different types of commodity images. Experimental results show that our model can effectively reduce the classification error rate compared with the base classifier.

Keywords: Three-way decisions · Cascade model
Convolutional Neural Networks · Image classification
Correcting Reliability Level

1 Introduction

Conventional image classification techniques were limited in their ability to process natural image data in their raw form. It required careful engineering and considerable domain expertise to design a feature extractor that transformed the raw data (the pixel values of an image) into a suitable internal representation or feature vector from which the classifier could classify patterns in the input [6]. On the contrary, deep-learning methods like ConvNets can learn representation directly from the data.

ConvNets are now the most commonly used large-scale image classification models. As early as 1990, ConvNets were trained for the task of classifying low-resolution images of handwritten digits [1]. However, with limited computation and

W.Shen—student.

training data, ConvNets were not popular until recent years. Krizhevsky et al. [4] trained a large, deep ConvNet (named AlexNet) to win over other contestants in the ILSVRC-2012 competition which was the first time a model performed so well on ImageNet dataset. Zeiler and Fergus [17] explained a lot of the intuition behind ConvNets and showed how to use a multi-layered deconvnet [18] to visualize the filters and weights correctly. Simonyan & Zisserman [12] created a simple and deep (19 layers) VGG Net, which reinforced the notion that ConvNets have to have a deep network of layers. Szegedy et al. [13] introduced the Inception module and built GoogLeNet. Coming up with the Inception module, a creative structuring of layers can lead to improved performance and computationally efficiency. He et al. [3] presented a residual learning framework to ease the training of deep networks and built a 152 layer ResNet, which won ILSVRC 2015 with an incredible error rate of 3.6% (lower than the error rate of humans, around 5–10%). Larsson et al. [5] advanced ResNet and built FractalNet which shows that explicit residual learning is not a requirement for building ultra-deep neural networks. All in all, the development of ConvNets is amazing.

In this paper, we use ConvNets for mass commodity image classification. However, a single ConvNet is limited in its ability of treating all the classes fairly. Combining the predictions of many different models is a very successful way to reduce test errors. Therefore, we propose a Class Grouping algorithm based on feedback to learn the similarity between classes and train several deep ConvNets become experts of different types of commodity images.

Schmidhuber et al. [10] declared that only winner neurons are trained. They trained several deep neural columns become experts on inputs preprocessed in different ways and average their predictions. Simonyan and Zisserman [11] proposed a two-stream ConvNet architecture which incorporates spatial and temporal networks. Each stream was implemented using a deep ConvNet, softmax scores of which were combined by late fusion.

Inspired by works above, we build a cascade ConvNets model to implement large-scale image classification. However, more innovative than those works, our model introduces the theory of three-way decisions to simulate human decision process. We build a 3WD-based cascade model including 3 layers, the first layer is a base ConvNet for all images and the third layer consists of several expert ConvNets trained for similar classes. The most important second layer is a three-way decision layer, which controls whether the data flows from the first layer to the third layer.

The notion of three-way decisions was originally introduced by the needs to explain the three regions of probabilistic rough sets. A theory of three-way decisions is constructed based on the notions of acceptance, rejection and non-commitment [15], whenever it is impossible to make an acceptance or a rejection decision, the third noncommitment decision is made [14]. Three-way decisions play a key role in everyday decision-making and have been widely used in many fields and disciplines. Three-way spam filtering systems [19], for example, add a suspected folder to allow users make further examinations of suspicious emails, thereby reducing the chances of misclassification. Three-way decisions are also

commonly used in medical decision making [8,9]. In the threshold approach to clinical decision making proposed in [9], by comparing the probability of disease with a pair of a “testing” threshold and a “test-treatment” threshold, doctors make one of three decisions: (a) no treatment no further testing; (b) no treatment but further testing; (c) treatment without further testing.

This paper extends the application of three-way decisions to the image classification. When there is doubt about the classification result(the first layer), our model will make a noncommitment decision(the second layer) and learn more information from expert classifiers(the third layer) to make the final prediction.

The structure of this paper is as follows. In Sect. 2 we describe our model. Next, in Sect. 3 we show the experiments and analyze the results. Last, we summarize our work in Sect. 4.

2 Model

We use GoogLeNet model throughout the paper. GoogLeNet, as defined in [13], uses 9 Inception modules in the whole architecture, which is a network in the network structure [7]. By using the Inception models, the network is deeper and wider, and is better than previous models.

In Sect. 2.1, we define some symbols that will be referred later. Next, in Sect. 2.2, we introduce the Class Grouping algorithm that based on feedback. Later, in Sect. 2.3 we introduce the three-way decision cascade model. Last, Sect. 2.4 shows the ultimate form of our model, CRL-supervised 3WD cascade model.

2.1 Symbols Definition

Before going further into our model, we define some symbols.

Img , the input image.

$CAT = \{c_1, c_2, \dots, c_i, \dots, c_C\}$, the class set, including C classes.

$P = \{p_1, p_2, \dots, p_i, \dots, p_C\}$, the classification result of a ConvNet model, where p_i is the probability that Img is classified as class c_i .

$Conf = (n_{ij})_{C \times C}$, the confusion matrix of ConvNet test result, where n_{ij} is the number of images of class c_i being classified as class c_j . The bigger the n_{ij} , the easier that images of class c_i are classified as class c_j .

Threshold of possible classes (Th-pos). Obviously, we do not need to consider the situation that Img belonging to class c_i if p_i is very small. Therefore, we need a threshold to determine the possible classes of Img . If p_i is no less than $Th-pos$, we think that Img may belong to class c_i . We stipulate that $Th-pos$ is no less than $\frac{1}{C}$.

Top-1 class (referred to as c_{top}), the class considered the most probable by the model, the probability is p_{top} .

2.2 Class Grouping Algorithm Based on Feedback

For commodity images of web-based platforms, many commodity classes are similar to each other, which is difficult for both humans and machines to distinguish

them (see Fig. 1). Therefore, a classifier trained for all the classes is not enough for distinguishing those similar classes, we need some more specified classifiers trained for certain similar classes. In this paper, we propose a Class Grouping (CG) algorithm (see Algorithm 1) based on the feedback of the classification results.



Fig. 1. Experimental data, 4 samples each class. Classes c_1, c_5, c_{10} and c_{14} are of similar features, they are all kinds of sweaters. And classes $c_3, c_{18}, c_{29}, c_{30}, c_{31}$ and c_{32} are kinds of trousers.

Algorithm 1. Class Grouping

Input: $CAT; S = (s_{ij})_{C \times C}$; cluster number K ;
Output: K clusters;
 Make each class in CAT a cluster;
 Compute pair-wise distance of all clusters, $d_{mn} = \min(s_{ij}), \forall c_i \in clt_m \ \& \ \forall c_j \in clt_n$;
repeat
 find two clusters that are closest to each other;
 merge the two clusters form a new cluster clt_{new} ;
 compute the distance form clt_{new} to all other clusters.
until there are only K clusters

We define s_{ij} the similarity between class c_i and class c_j (see Formula (1)).

$$s_{ij} = \frac{n_{ij}}{\sum_{t=1}^C n_{it}} \times \frac{n_{ji}}{\sum_{t=1}^C n_{jt}} \tag{1}$$

After running CG algorithm, we get K clusters: $cat-1, cat-2, \dots, cat-K$. We train an expert ConvNet for each cluster(see Sect. 3.2). These ExpConvNets will be used to build the third layer of cascade model.

Based on class grouping experimental results, we introduce *similar-classes*. If class c_i and c_j are belong to the same subset $cat-k$, we call that c_i and c_j are similar-classes. Similar-classes are of high probability to be wrongly classified to each other and therefore require expert judgments.

2.3 Three-Way Decision Based Cascade Model

A theory of 3WD is constructed based on the notions of acceptance, rejection and noncommitment. Inspired by 3WD theory, we no longer directly accept the classification result of the base classifier. Instead, we make one of two decisions: (a) accept it if it is reliable; (b) opt for a noncommitment if it is not reliable. Since this is not a binary-decision problem with two options, but a multiclass classification problem, there is no “reject” option. We judge the classification result is not reliable if meeting two conditions: (i) existing class $c_a (c_a \neq c_{top})$ and $p_a \geq Th-pos$; (ii) c_a and c_{top} belonging to the same subset $cat-k$. The condition (i) guarantees that Img may belong to class c_a , condition (ii) guarantees that c_a and c_{top} are similar-classes. We define these conditions under the hypothesis that meeting these two conditions means that the classifier is confused. The classifier considers that Img can be predicted as c_a or c_{top} . Thus, we need to put the image into the expert classifier $ExpConvNet-k$ for further judgment. Formula (2) is the 3WD process.

$$3WD = \begin{cases} \text{delay,} & \text{satisfying condition (i) and (ii)} \\ \text{accept,} & \text{otherwise} \end{cases} \quad (2)$$

The ability of a single classifier is limited. Therefore, combining several different models is a way to reduce the error rate [4]. Cascade [2] is a special case of ensemble learning. The basic idea of cascade is the connection of multiple classifiers. The information is passed between layers and the output information of the upper classifier is used as the additional information of the next classifier.

Under the guidance of 3WD theory, we establish a self-adaptive cascade ConvNets model, including 3 layers (see Algorithm 2). The first layer is a base classifier (a base ConvNet), the second layer is a 3WD layer and the third layer is expert layer, including several expert classifiers (ExpConvNets). We put Img into the first layer (a base classifier) and send the classification result P^1 into the 3WD layer. Next, the 3WD layer will make one of two decisions: (a) accept P^1 as the final P ; (b) opt for a noncommitment and put Img into $ExpConvNet-k$ (the classification result is P^2). Finally, we calculate probability P based on P^1 and P^2 , see Formula (3).

$$p_i = \begin{cases} p_{c_i}^2, & \text{if } c_i \in cat-k \\ p_{c_i}^1, & \text{otherwise} \end{cases} \quad (3)$$

where $p_{c_i}^1$ represents the probability of the image being predicted as class c_i by the base classifier and $p_{c_i}^2$ represents the probability of the image being predicted as class c_i by the expert classifier.

Algorithm 2. 3WD-CM

Input: image Img , size $n * n$
Output: prediction P , size C

Input Img into the first layer (a base ConvNet), get P^1 ;
 Send the classification result P^1 to the 3WD layer;

if $\exists c_a$, & $p_a \geq Th\text{-}pos$ **then**
 if $c_a \in cat\text{-}k$ & $c_{top} \in cat\text{-}k$ **then**
 put Img into $ExpConvNet\text{-}k$, get P^2 ;
 calculate the final P with Formula (3)
 else
 $P = P^1$
 end if
else
 $P = P^1$
end if

2.4 CRL-supervised 3WD Cascade Model

3WD decides which images may need expert judgments. But the experimental result (see Table 2) tells us that blindly following the decision of 3WD is not a good idea. Suppose that the base classifier considers Img as class c_i while 3WD layer delays this result and after expert judgment, the cascade model finally judge Img as class c_j . Experimental experience tells us that there are two situations: (1) in most cases, c_i is the correct class while c_j is wrong; (2) on the contrary, in most cases, c_i is wrong while c_j is the correct class. Situation (1) tells us that the 3WD makes the classification result from right to wrong; while situation (2) tells us that 3WD makes the classification result from wrong to right. Obviously, we welcome the latter situation. Thus, we need to supervise the 3WD process.

We define *Correcting Reliability Level (CRL)* to supervise the 3WD process. A high CRL means that the 3WD has a high probability of making right decision. We define TF_i (Truth to False) to describe the situation (1) above, and FT_i (False to Truth) to describe the situation (2) above. CRL computing see Formula (4).

$$CRL_i = \begin{cases} \frac{N_{FT_i} - N_{TF_i}}{N_{FT_i} + N_{TF_i}} & \text{while } N_{FT_i} > N_{TF_i} \text{ \& } N_{FT_i} > 0 \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

We use a random function

$$R(p) = \text{binomial}(1, p) \quad (5)$$

to move CRL value to a Boolean value, “True” means following the 3WD and putting Img into the expert classifier, while “False” means “accept” the result of base classifier. p is the probability of return “True”. $R(CRL_i)$, for instance, has a probability of CRL_i to return “True”. Therefore, the larger the CRL value is, the more likely that Img will be passed into the expert classifier.

On the basis of 3WD-CM, we add a CRL table after the 3WD layer. CRL table is used to determine whether Img is worthy of expert judgment (see Algorithm 3).

Algorithm 3. CRL-CM

Input: image Img , size $n * n$
Output: prediction P , size C
 Input Img into the first layer (a base ConvNet), get P^1 ;
 Send the classification result P^1 to the 3WD layer;
if $\exists c_a$, & $p_a \geq Th_pos$ **then**
 if $c_a \in cat-k$ & $c_{top} \in cat-k$ **then**
 Check CRL table and get $crl = CRL_{top}$
 if $R(crl)$ is TRUE **then**
 put Img into $ExpConvNet-k$, get P^2 ;
 calculate the final P with Formula (3)
 else
 $P = P^1$
 end if
 else
 $P = P^1$
 end if
else
 $P = P^1$
end if

3 Experiments

In this section, we show the experiment details. In Sect. 3.1, we introduce the experimental dataset. Next, in Sect. 3.2 we introduce the class grouping process and ExpConvNets training. Then, in Sect. 3.3 we show the results of 3WD-based Cascade Model. At last, in Sect. 3.4 we analyze the results of CRL-supervised 3WD Cascade Model.

3.1 JD Clothing Dataset

The experimental data of this paper is JD clothing dataset, examples see Fig. 1. JD is one of the most famous B2C shopping site in China and the first large-scale integrated business platform to be listed in the United States. JD has a strong market share; therefore, it has accumulated a large number of commodity image data, which provides researchers with a lot of resources. Our experimental dataset has about 400,000 clothing images, including 37 classes. The dataset is divided into training set, validation set and test set at a ratio of 8: 1: 1.

In this paper, we report two error rates: top-1 and top-5, where the top-5 error rate is the fraction of test images for which the correct label is not among the five labels considered most probable by the model [4].

3.2 Class Grouping and ExpConvNets Training

In this paper, we do experiments on a deep learning framework named Caffe. Yosinski et al. [16] pointed out that fine-tuning is better than randomly initialize parameters. Our experimental results also confirm this. We start from a pre-trained model (GoogLeNet on ImageNet LSVRC-2014) and fine-tune it. The top-1 error rate is 44.59% and the top-5 error rate is 9.40%, which are better than randomly initializing (the top-1 error rate is 57.11% and the top-5 error rate is 21.16%). We call the fine-tuned model Base Model (BM) below and the latter experiments will fine-tune models on the basis of it.

Test results confirm that images of many classes are easily to be misclassified to each other, like class c_1 and c_{10} (see Fig. 2). Thus, we group those similar classes into the same subset with CG algorithm introduced in Sect. 2.2.

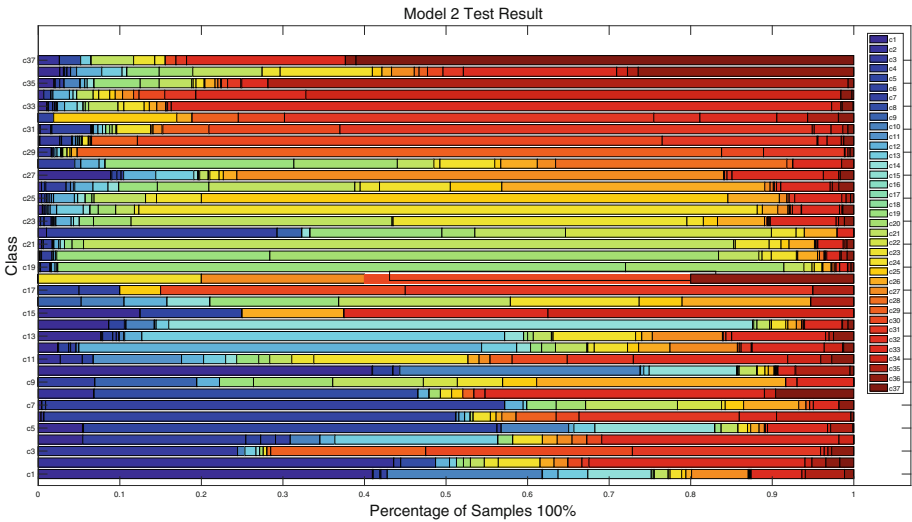


Fig. 2. Stacked bar chart of test result of BM. Take class c_1 as example, there is about 20% of test images misclassified as class c_{10} . Similarly, there is about 40% of test images of class c_{10} misclassified as class c_1 .

We set $K = 5$ and divide CAT into 5 subsets. After class grouping, we train an expert ConvNet for each subset. We fine-tune the BM and adapt most of the architecture (only change the output number of the last layer), and resume training from the BM weights. Table 1 shows the class grouping result and the ExpConvNets error rates.

Table 1. Class grouping results and ExpConvNets error rates.

Subset	Classes	ExpConvNet	Top-1 error rate(%)
cat-1	$c_2, c_8, c_{15}, c_{27}, c_{33}, c_{37}$	ExpConvNet-1	9.16
cat-2	$c_1, c_5, c_{10}, c_{14}, c_{25}, c_{35}$	ExpConvNet-2	41.17
cat-3	$c_3, c_6, c_{17}, c_{18}, c_{21}, c_{29}, c_{30}, c_{31}, c_{32}$	ExpConvNet-3	23.50
cat-4	$c_4, c_{12}, c_{13}, c_{34}$	ExpConvNet-4	16.67
cat-5	$c_7, c_9, c_{11}, c_{16}, c_{19}, c_{20},$ $c_{22}, c_{23}, c_{24}, c_{26}, c_{28}, c_{36}$	cat-5ExpConvNet-5	33.48

3.3 3WD-based Cascade Model

In order to get the CRL table, we first need to test images with 3WD-CM. We set Th-pos 0.1 in this experiment. The top-1 error rate of 3WD-CM is 44.401%, reducing by 0.189% compared with BM; top-5 error rate is 9.475%, increasing by 0.075% compared with BM. The results are bad. The top-1 error rate reduces a little bit, and the top-5 error rate does not drop but increase.

Table 2 shows the counting results of TF and FT. We can see that there are totally 1195 cases of situation (1) and 1,273 cases of situation (2). Therefore, in fact only 78 samples are modified correctly by 3WD-CM, the accuracy increases by only 0.189%. When c_1 is considered as c_{top} by the base classifier, there are totally 613 images considered needing expert judgement by 3WD, wherein, 135 images are modified correctly and 478 images are modified incorrectly. Thus, there are 343 images being modified incorrectly in total. This shows that when c_1 is considered as c_{top} , we should better ignore the decision of 3WD that *Img* needing expert judgment. We should better accept c_1 as the prediction result. On the contrary, if the base classifier considers c_{10} as c_{top} , we would better follow the 3WD and do an expert judgment for *Img*. Because, for c_{10} , the number of images which are modified correctly is greater than the number of images which are modified incorrectly. Therefore, we use CRL to supervise 3WD process.

Table 2. Classification results of 3WD-CM

Class	N_{FT}	N_{TF}	NET	Class	N_{FT}	N_{TF}	NET	Class	N_{FT}	N_{TF}	NET	Class	N_{FT}	N_{TF}	NET
c_1	135	478	-343	c_{11}	1	2	-1	c_{21}	0	22	-22	c_{31}	64	52	12
c_2	4	7	-3	c_{12}	9	9	0	c_{22}	3	8	-5	c_{32}	1	1	0
c_3	11	3	8	c_{13}	3	13	-10	c_{23}	22	10	12	c_{33}	15	42	-27
c_4	1	1	0	c_{14}	73	29	44	c_{24}	37	32	5	c_{34}	0	2	-2
c_5	33	18	15	c_{15}	0	0	0	c_{25}	3	8	-5	c_{35}	17	43	-26
c_6	16	13	3	c_{16}	0	0	0	c_{26}	11	19	-8	c_{36}	12	10	2
c_7	5	2	3	c_{17}	0	0	0	c_{27}	6	20	-14	c_{37}	3	3	0
c_8	5	2	3	c_{18}	0	0	0	c_{28}	9	1	8	Total	1273	1195	78
c_9	3	2	1	c_{19}	60	58	2	c_{29}	21	26	-5				
c_{10}	602	150	452	c_{20}	58	54	4	c_{30}	30	55	-25				

Note: $NET(= N_{FT} - N_{TF})$ means the net number of samples modified correctly by 3WD-CM.

3.4 CRL-supervised 3WD Cascade Model

Now, we calculate CRL of each class, see Table 3. We establish CRL-supervised 3WD cascade model (CRL-CM). Table 4 shows the classification performance of CRL-CM under different $Th-pos$ values (0.1, 0.2, 0.3 and 0.4). We test 30 times for each $Th-pos$ value and take the average error rate. Compared with BM, the top-1 error rate reduces by about 1.09% when $Th-pos = 0.1$. Top-5 error rate does not reduce obviously. With the increasement of $Th-pos$, the error rate reduces less, because the greater the $Th-pos$ is, the more harsh that 3WD determines a sample being “unsure”, thus, those misclassified samples lose the chance of being modified correctly. The experimental results show that the CRL-CM can effectively reduce the classification error rate compared with a single base ConvNet.

Table 3. CRL table.

Class	CRL	Class	CRL	Class	CRL	Class	CRL
c_1	0	c_{11}	0	c_{21}	0	c_{31}	0.103
c_2	0	c_{12}	0	c_{22}	0	c_{32}	0
c_3	0.571	c_{13}	0	c_{23}	0.375	c_{33}	0
c_4	0	c_{14}	0.431	c_{24}	0.072	c_{34}	0
c_5	0.294	c_{15}	0	c_{25}	0	c_{35}	0
c_6	0.103	c_{16}	0	c_{26}	0	c_{36}	0.091
c_7	0.429	c_{17}	0	c_{27}	0	c_{37}	0
c_8	0.429	c_{18}	0	c_{28}	0.8		
c_9	0.2	c_{19}	0.017	c_{29}	0		
c_{10}	0.601	c_{20}	0.036	c_{30}	0		

Table 4. Average error rates of CRL-CM under different $Th-pos$.

$Th-pos$	Top-1 error rate(%)	Top-5 error rate(%)
0.1	43.50 (1.09)	9.386 (0.017)
0.2	43.62 (0.97)	9.398 (0.005)
0.3	43.68 (0.91)	9.415 (-0.013)
0.4	43.77 (0.82)	9.413 (-0.011)

Note: The numbers in parentheses are reduced error rates (%) compared with Model 2.

4 Conclusion

In this paper we integrate several different ConvNets to build a CRL-supervised 3WD cascade model. Experimental results show that our model can effectively

reduce the error rate compared with a single ConvNet. The contributions of this paper are: (i) Simulating the human decision process by using 3WD to construct a cascade model with several ExpConvNets which become experts on inputs preprocessed in different ways; (ii) introducing CRL to supervise 3WD process which reduces error rate effectively. In future work, we will do more experiments with public datasets like ImageNet to prove the validity of our model.

Acknowledgments. The work is partially supported by the National Natural Science Foundation of China (No. 61573259, 61673301, 61573255 and 61673299), the program of Further Accelerating the Development of Chinese Medicine Three Year Action of Shanghai (No. ZY3-CCCX-3-6002) and the Natural Science Foundation of Shanghai (NO. 15ZR1443800).

References

1. Cun, Y.L., Boser, B., Denker, J.S., Howard, R.E., Habbard, W., Jackel, L.D., Henderson, D.: Handwritten digit recognition with a back-propagation network. In: *Advances in Neural Information Processing Systems*, pp. 396–404 (1990)
2. Gama, J., Brazdil, P.: Cascade generalization. *Mach. Learn.* **41**(3), 315–343 (2000)
3. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition, pp. 770–778 (2016)
4. 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)
5. Larsson, G., Maire, M., Shakhnarovich, G.: Fractalnet: Ultra-deep neural networks without residuals (2016)
6. Lecun, Y., Bengio, Y., Hinton, G.: Deep learning. *Nature* **521**(7553), 436–444 (2015)
7. Lin, M., Chen, Q., Yan, S.: Network in network. *Computer Science* (2014)
8. Lurie, J.D., Sox, H.C.: Principles of medical decision making. *Spine* **24**(5), 493 (1999)
9. Pauker, S.G., Kassirer, J.P.: The threshold approach to clinical decision making. *New Engl. J. Med.* **302**(20), 1109 (1980)
10. Schmidhuber, J., Meier, U., Ciresan, D.: Multi-column deep neural networks for image classification. In: *Computer Vision and Pattern Recognition*, pp. 3642–3649 (2012)
11. Simonyan, K., Zisserman, A.: Two-stream convolutional networks for action recognition in videos. *Adv. Neural Inf. Process. Syst.* **1**(4), 568–576 (2014)
12. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. *Computer Science* (2015)
13. Szegedy, C., Liu, W., Jia, Y., Sermanet, P.: Going deeper with convolutions. In: *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–9 (2015)
14. Yao, Y.: Three-way decisions with probabilistic rough sets. *Inf. Sci.* **180**(3), 341–353 (2011)
15. Yao, Y.: An outline of a theory of three-way decisions. In: Yao, J.T., Yang, Y., Słowiński, R., Greco, S., Li, H., Mitra, S., Polkowski, L. (eds.) *RSTC 2012. LNCS (LNAI)*, vol. 7413, pp. 1–17. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-32115-3_1

16. Yosinski, J., Clune, J., Bengio, Y., Lipson, H.: How transferable are features in deep neural networks? *Eprint Arxiv* 27, pp. 3320–3328 (2014)
17. Zeiler, M.D., Fergus, R.: Visualizing and understanding convolutional networks. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) *ECCV 2014*. LNCS, vol. 8689, pp. 818–833. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10590-1_53
18. Zeiler, M.D., Taylor, G.W., Fergus, R.: Adaptive deconvolutional networks for mid and high level feature learning. In: *IEEE International Conference on Computer Vision*, pp. 2018–2025 (2011)
19. Zhou, B., Yao, Y., Luo, J.: A three-way decision approach to email spam filtering. In: Farzindar, A., Kešelj, V. (eds.) *AI 2010*. LNCS (LNAI), vol. 6085, pp. 28–39. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-13059-5_6