ISSN 2616-5775 Vol. 6, Issue 9: 11-16, DOI: 10.25236/AJCIS.2023.060902

LOGO recognition system based on deep learning

Jiaming Liu[1,a,#], Sizhe Chen[1,b,#], Songyang Chen[1,c,#], Hanlun Dong[1,d,#], Haojie Ma[1,e,#], Xinyang Zhang[1,f,#]

[1]Beijing 21st Century International School, Haidian, Beijing, China
[a]andrewjiaming@outlook.com, [b]csz20050526@163.com, [c]cshawny@163.com, [d]Hanlundong@126.com, [e]3563552942@qq.com, [f]zxy070326@sina.com
([#]Co-first author)These authors contributed equally to this work

Abstract: *We used the deep learning architecture designed by ourselves to identify the logo, with good effect and accuracy. Our architecture uses four convolutional neural network architectures, two pooling structures and two fully connected neural network architecture.The characteristic of our architecture is that it is relatively simple. We can use the limited things we learn to create a program that meets our requirements.The results of the test were relatively successful. The logo recognition accuracy for our own data set can reach 95.83%.*

Keywords: *Deep-learning, LOGO recognition, Neutral network architectures*

1. Introduction

Logo represents the special mark on certain company. It appears in several forms such as patterns, characters and both. An enterprise's strong overall strength, perfect management mechanism, quality products and services, are concentrated in the trademark. The company repetitively promote their products by using logo, leaving a profound concept among consumers and stimulating their lust for buying those products. By letting users remember their trademarks, they can increase the advertising efficiency of the company [1]. In order to reach this result, there are several characteristics which show the simplicity and oddity. First, there should have a big contrast between the words and the background colors among color, texture and style. In addition, logo should be simple but contain the diversity of certain product or company, giving potential customers novel feelings. However, as the technology advances by leaps and bounds, sham products start appearing in the market. According to Organisation for Economic Co-operation and Development, counterfeit and pirated goods accounted for 3.3% of global merchandise trade in 2016. It seems not too much but when considering the whole global trade, it is a severe problem. The existence of a large number of fake and shoddy commodities not only affects consumers' shopping guarantee, but also directly affects the trustworthiness of e-commerce platforms. To solve this problem, people design a method of commodity detection to test whether there is a tiny difference which cannot be discovered by naked eyes. This approach increases the quality and quantity of doing authentic identification.

Machine learning can be divided into three kinds according to different learning styles: Supervised learning, Unsupervised Learning and Semi-supervised learning; The main task of supervised learning is to generate an input-to-output mapping function based on the correspondence between labeled input and output data. Unsupervised learning is the direct modeling process of unlabeled data, and the machine itself learns to find out the potential rules between these different categories of data, such as common clustering tasks. The main task of semi-supervised learning is to use the model hypothesis of data distribution to build a learner to label unlabeled samples. The Logo detection and recognition technology to be studied in this paper is the supervised learning in the machine learning method [2]. There are several widespread, simple methods to achieve the goal. The first one is SIFT method. To help match the many viewpoints of an object, SIFT's primary function is to find and describe local characteristics in photos. Key points, or interesting aspects of the thing, are what make up a feature description. Recognizing and identifying the object in a test image among numerous other objects can be made easier by extracting this description from the training image [3]. The arithmetic of SIFT is contained by scale space, Gaussian Blur, gaussian pyramid. It first acquires the absolute maximum and minimum and localize the feature point for recognition, then confirm the main direction, last having a description of the particular logo [1]. The second method is Adaboost method. This method is the combination of Harr feature operator and Adaboost classifier. First, Haar feature description operator is

ISSN 2616-5775 Vol. 6, Issue 9: 11-16, DOI: 10.25236/AJCIS.2023.060902

extracted, and then Adaboost classifier is used for classification. It can be used in the face detection. Later, some researchers applied the Haar feature description operator in the field of Logo detection and recognition [2]. Another machine learning algorithm built on statistical theory called the LS-SVM excels at solving the small-sample learning problem. It can successfully avoid the neural network's local optimal solution and get over the dimensionality curse. The best feature of LS-SVM is to convert the quadratic programming problem into a linear equation problem by changing the inequality constraint in the SVM to the equality constraint and using the training error square in place of the slack variable to significantly increase the speed and accuracy of model parameter estimation [4].

Furthermore, some models are produced for deep learning. The first one is DNS. The core concept behind the DSN architecture is the idea of stacking, which is where small modules of functions or classifiers are first assembled and then "stacked" on top of one another to learn complicated functions or classifiers. The DSN uses supervision information to stack each of the fundamental modules, which is represented simply as a multilayer per- caption. In the fundamental module, the hidden units are nonlinear and the output units are linear [5]. Besides, there is another method called Restricted Boltzmann Machine. The visible layer (input layer) and the hidden layer are the only layers in an RBM, making it structurally similar to a shallow neural network. Through the process of reconstructing the input, it can automatically discover patterns in data. The lack of connections between neurons in each layer and their connections to all other neurons in the other layer is what causes an RBM to be regarded as restricted. symmetrical bidirectional connections exist between neurons in RBM networks. Because the weights are the same in both ways, information flows in both directions during network usage and network training [6]. A type of neural network called a DBN is built from numerous RBMs. Only one layer is trained at a time, and the higher layer receives its input from the lower layer's output. A back-propagation technique is then employed to adjust the network as a whole. Every DBN is used to learn hierarchical feature representations, and all of the DBNs that are thought to be poor learners are linked together by a boosted classifier [7].

Those three methods can recognize the logo pretty well, but there is a problem exist in those methods. Both of them are pretty complex. Even though they can recognize the logo pretty well, it is difficult to achieve the target in low version computer. In order to solve the problem, we use less convolution layer to realize our goals. In this paper, we use 6 convolution layers which strengthen the operation efficiency.

The structure of the essay is as follows. First, we introduced the research background and significance of logo detection, analyzing the problem of existing methods and the solutions toward this problem. Second, we will contend the development of the whole system including the information of convolution layer, fully connected neutral network, and the introduction of pooling layer. In addition, we will introduce the system majorization. The third part is about system testing which proves our predictions and have a brief introduction of data preparation and management. We will exhibit several groups of consequences of our test and show our conclusion. Last but not least, we will make a conclusion of our whole experiments, adding some thought about future experiment, including strengthen the variety models of logo type and the application of those models.

2. Neutral network architecture

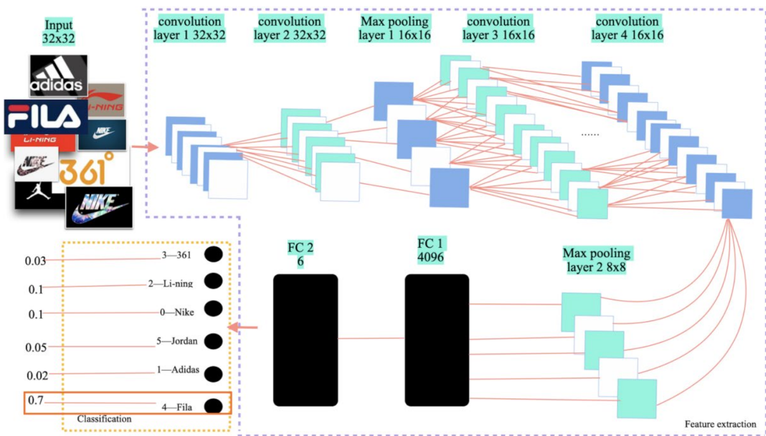

Figure 1: Convolutional layer

ISSN 2616-5775 Vol. 6, Issue 9: 11-16, DOI: 10.25236/AJCIS.2023.060902

In this experiment, a convolutional neural network is designed to help consumers recognizing the manufacture information quickly. After inputting images into the computer, there are three different kinds of layers processing those image again and again. First, two convolutional layers help the computer recognizing special features. Next, max polling layer 1 decreases the amount of data from 32x32 to 16x16. After the pooling layer, there are two convolution layers again to extract features and another pooling layer declined number of data to 8x8. Finally there are two FC layers outputting judgment data. In the experiment we choose six sports brands as sample. The operated picture is belong to that brand which corresponds to the largest resulting (Figure 1).

2.1. Convolutional layer

The convolutional layer realizes automatic feature extraction. The convolutional layer which is closer to the input can be extracted features easier, like some lines, outlines, and so on. In contrast, the convolutional layer which is closer to the output is more complex. The features extracted from those complex convolutional layers are high-order features that are extracted based on the low-order features. The convolution operation in a convolutional Neural Network can be viewed as an operation for samples and the convolution kernel, which reaches feature response plots. By performing a convolution operation on the input samples, we can get feature response plots of the first layer. From the second layer, the input samples should be the feature response plots of the former. After conducting the operation with the convolution kernel of the corresponding layer, feature response plots for each layer can be gained.

2.2. The Rectified Linear Units Layer (ReLU layer)

To make the convolution results can be used as feature plots 'calculation results, calculation provided by Rectified Linear Units layer is needed. Activation function includes sigmoid, tanh, ReLU, Softmax, and so on. Usually, we use The Rectified Linear Units, which has the traits of converging quickly and equations, as an activation function.

2.3. Pooling layer

The pooling layer can be used to decrease the size of the characteristics graph produced by the convolutional layer. It is the process of pooling layers that choosing one area in the original characteristics graph and gets a new characteristics graph which will be smaller. This method not only pulls the characteristic graph to a lower dimension but also make the way of representing traits to be more stable.

2.4. Fully connected layer

The convolutional layer is to learn the local features, the fully connected layer re-adjusts the graph with eigenvalue by performing a dimensionality reduction characteristic graph. The features obtained from the convolutional layer are expanded into column vectors and fed into a fully connected neural network, which calculates the probability, or score, of each category. For example, there is an image. The computer can recognize whether a picture is for a dog or a cat. After the graph is calculated, the traits like a cat are 0.7 and the traits like a dog are 0.3. Two numbers must add up to 1. 0.7 is greater than 0.3, thus the computer knows that this picture is a cat.

3. Optimizing

Overfitting refers to the concept that the algorithm is too close to a specific set of data, so that other data cannot be predicted accurately. It is a problem that frequently occur in machine learning systems. In order to solve this problem in our system, several methods are used in our system.

One way to reduce the effect brought by overfitting is using regularization. Mathematically speaking, regularization adds a regularization term to the error function in order to prevent the coefficients to fit so perfectly to overfit[1]. In our system, L2 regularization is used.

$$C = C_0 + \frac{\lambda}{2n}\sum_w w^2 \tag{1}$$

The function above is the L2 regularization. C_0 represents the original loss function, which can be

ISSN 2616-5775 Vol. 6, Issue 9: 11-16, DOI: 10.25236/AJCIS.2023.060902

replaced. λ is the self-defined parameter. n stands for the number of training samples. The sum ∑ act for all weights in the matrix. The idea to train the smaller weights primarily and retain the greater weights only when C_0 could be reduced significantly is promoted implicitly in L2 regularization.

Another method is dropout. By using dropout, units are dropped from the network temporarily during training. Thus, various "thinned" networks are created, and the effect of averaging the predictions can be easily approximated by using unthinned network with smaller weights. As a result, overfitting is significantly reduced in our system, and problem brought by the slow speed of large network is also solved[2].

3.1. Data set description:

The samples, the brand logo, encompass the Nike, Adidas, Li-Ning, 361, Fila and Jordan. There are 3935 picture for each brand logo we study, and the total amounts of picture are up to 23620. The data were found using python crawler and were downloaded to the database. Then, we refine the database by manually delete the unrelated or vague pictures of the logo presented in the database. (Table 1)

Table 1: Brand logo samples.

Brand logo name	Sample1	Sample2	Sample3	Sample4	Sample5
Nike					
Adidas					
Li-ning					
361					
Fila					
Jordan					

3.2. Data's alternation and transcript:

Table 2: The one-hot encoding for each brand.

Brand name	Label	One-hot encoding
Nike	0	[1, 0, 0, 0, 0, 0]
Adidas	1	[0, 1, 0, 0, 0, 0]
Li-ning	2	[0, 0, 1, 0, 0, 0]
361	3	[0, 0, 0, 1, 0, 0]
Fila	4	[0, 0, 0, 0, 1, 0]
Jordan	5	[0, 0, 0, 0, 0, 1]

Train data and Test data are devised in the database. Train data are used to modify our model and the Test data are used to debug the model. The pictures number ratio for Train data to Test data is 4:1. The Train data and Test data are used in MATLAB form (Table 2).

ISSN 2616-5775 Vol. 6, Issue 9: 11-16, DOI: 10.25236/AJCIS.2023.060902

4. The results of experiment

Table 3: Top 15 results of experiment

dropout_rate	Base_learning_rate	Decay_rate	Iteration_steps	Accuracy(%)
0.97	0.002	0.99	2000	95.83
0.97	0.001	0.97	2000	95.36
0.97	0.001	0.99	2000	94.36
0.96	0.001	0.97	2100	94.1
0.93	0.001	0.99	1900	94.04
0.91	0.001	0.99	1900	93.94
0.95	0.001	0.97	2100	93.65
0.97	0.001	0.99	1900	93.54
0.95	0.001	0.99	1900	93.12
0.97	0.001	0.97	1900	92.89
0.95	0.001	0.99	2000	92.7
0.99	0.001	0.99	2000	92.4
0.97	0.001	0.99	2100	91.72
0.91	0.002	0.99	1900	91.69
0.96	0.002	0.97	2200	91.45

(Note: ranking in decreasing order)

We change four variables, dropout_rate, base_learning_rate, decay_rate, iteration_steps in order to make a best prediction. After 80 rounds of experiments, we left the top 15 accurate results from the experiments. The results suggests that achieved 95.83% accuracy in six logo identification is a great output. This means that the algorithm was able to correctly identify logos 95 out of 100 times, which is a very high success rate. This is especially impressive considering that logos can be very complex and varied, with many different shapes, colors, and fonts.

There are a few factors that may have contributed to the high accuracy of the algorithm. First, the algorithm was trained on a small dataset of logos, which gave it a not really good understanding of the different features that logos can have. From the results, we can see that the higher number of training data is, the accuracy it is. Second, the algorithm was able to learn to identify logos in different contexts, such as different sizes, colors, and orientations.

The high accuracy of the algorithm suggests that it has the potential to be a valuable tool for businesses and organizations. For example, the algorithm could be used to automatically identify logos in images or videos, which could be helpful for tasks such as product recognition, brand protection, and marketing analytics.

Overall, the artificial experiment that achieved 95.83% accuracy in logo identification is a promising development. The algorithm has the potential to be a valuable tool for businesses and organizations, and it could help to improve the efficiency and accuracy of a variety of tasks(Table 3).

5. Conclusion

We use the deep learning architecture designed by ourselves to identify the logo, which can help customers or users quickly identify the corresponding information of the logo manufacturer, and can also quickly identify the products with logo infringement. The structure of this program is input into four convolutional neural network structures and two pooling, a pooling is followed by two convolution, then is the fully connected neural network, and pooling layer is followed by output. Our program can realize the accuracy of logo identification in a short time with 95.83% in the test with our dataset. In the future, our program can optimize the algorithm and model, make the running speed faster, and increase the types and accuracy of logo recognition, so as to make our program more perfect.

References

[1] Wenlong, Zhang (2015). Logo Recognition In Complex Situation (Master Dissertation, Tianjin Normal University)

[2] Zhang, Shuzhou. (2018). Based on the deep learning Logo detection and recognition technology

ISSN 2616-5775 Vol. 6, Issue 9: 11-16, DOI: 10.25236/AJCIS.2023.060902

research (master's degree thesis, university of electronic science and technology). https://kns. cnki.net/ KCMS/detail/detail.aspx?dbname=CMFD201802&filenam e=1018707864.nh

[3] Zahedi, M., & Eslami, S. (2011). Farsi/Arabic optical font recognition using SIFT features. Procedia Computer Science, 3, 1055–1059. https://doi.org/10.1016/j.procs.2010.12.173

[4] Deng, W., Yao, R., Zhao, H., Yang, X., & Li, G. (2017). A novel intelligent diagnosis method using optimal LS-SVM with improved PSO algorithm. Soft Computing, 23(7), 2445–2462. https://doi.org/ 10.1007/s00500-017-2940-9

[5] Deng, L. (2014). Deep Learning: Methods and Applications. Foundations and Trends® in Signal Processing, 7(3-4), 197–387. https://doi.org/10.1561/2000000039

[6] Coşkun, M., Yildirim, Ö., Uçar, A., & Demir, Y. (2017). An Overview of Popular Deep Learning Methods. European Journal of Technique (EJT), 7(2), 165–176. https://dergipark. org.tr/en/ pub/ejt/issue/34562/403498

[7] Du, X., Cai, Y., Wang, S., & Zhang, L. (2016, November 1). Overview of deep learning. IEEE Xplore. https://doi.org/10.1109/YAC.2016.7804882