

Research on Visual Checkout System based on Convolutional Neural Network

Shiji Zheng^{1, a}, Xuetao Huang^{1, b}

¹School of Information Engineering, Jiangmen Polytechnic, Jiangmen 529000, China

^ashiji_zheng@163.com, ^bhtz_kj@163.com

Abstract

With the arrival of the new retail era, the development of stores is facing more opportunities and challenges. In this paper, the commodity checkout process of the existing offline stores is improved in order to speed up the checkout process and reduce the human error factors in the checkout process. A visual checkout system based on convolution neural network is designed and implemented. The system can automatically identify goods, automatically calculate the quantity of goods and the total price of goods, improve the purchasing experience of users, and reduce the cashiers in stores. It also reduces the probability of miscalculation of prices caused by human factors.

Keywords

Image recognition; Image processing; Convolution neural network; Retail.

1. Introduction

With the arrival of the new retail era, the development of stores is facing more opportunities and challenges. With the continuous improvement of the personalized needs of end users, new retail methods such as experiential consumption and intelligent retail will become an important part of the development opportunities of stores. So far, most retail stores still use the traditional method of checkout: cashiers need to manually scan the bar code of the item to settle the bill. This method leads to some shortcomings such as long checkout time, easy to make mistakes, low efficiency and so on.

With the rapid development of AI technology, a large number of AI technologies that can understand pictures have been realized. In order to solve the above automation requirements of goods identification and improve the efficiency of goods recognition, this paper proposes the design and research of visual checkout system based on convolution neural network.

2. Convolutional Neural Network

Convolutional neural network is the most widely used deep learning model. It can learn the features comprehensively and effectively through training. Compared with the traditional neural network, convolutional neural network takes advantage of the characteristics of convolution algorithm, and greatly reduces the complexity of the network structure by means of partially connected, local receptive field, shared weights and pooling. At present, convolutional neural network has been widely used in natural language processing, climate prediction, artificial intelligence and other fields. Because convolutional neural network can learn the characteristics of different levels from a large number of data, it will have a very broad application prospect in the field of network intrusion detection. The classical convolution neural network model is mainly composed of five parts: input layer, convolution layer, pooling layer, full connection layer and classification layer, as shown in Figure 1.

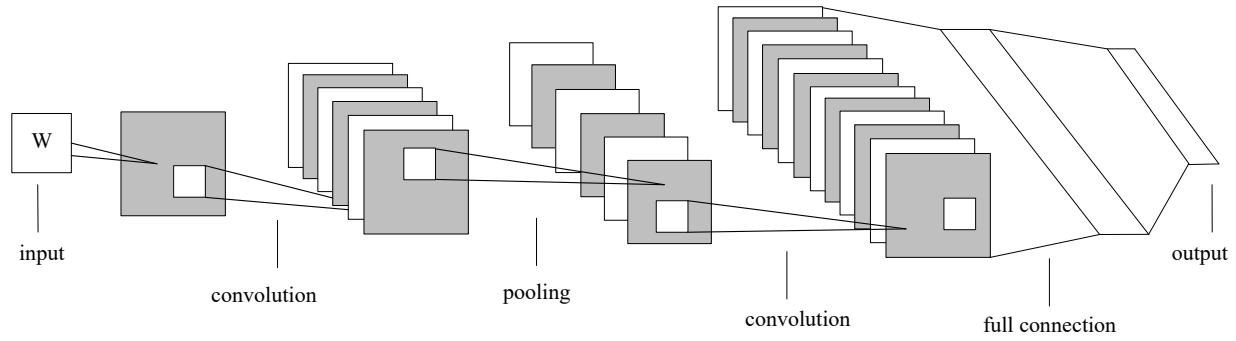


Figure 1. Classical convolution neural network model

2.1. Convolution Layer

The convolution layer is the most important part of convolution neural network. After the feature image is input into the convolution layer, it will convolute with the convolution kernel. The convolution formula is as follows:

$$x_j^l = f \left(\sum_{i \in P_j} x_i^{l-1} * k_{ij}^l + b_j^l \right) \quad (1)$$

$f \left(\sum_{i \in P_j} x_i^{l-1} * k_{ij}^l + b_j^l \right)$ is a tanh function, P_j is a local receptive field, x_i^{l-1} is the value of the $l-1$ feature on the i window, the position on the 1st floor is (i, j) , k_{ij}^l is the weight of the convolution kernel, b_j^l is the offset of the feature.

During convolution operation in convolution layer, the convolution kernel slides on the whole input feature according to the set step size, and simultaneously multiplies and sums the parts corresponding to the local receptive field to perform convolution operation until the convolution kernel slides out of the input feature.

2.2. Pooling Layer

The role of pooling layer is to reduce the dimension of features and filter out redundant features, so as to reduce the computation and improve the generalization ability of the network. According to the different sampling methods, the pooling layer can be divided into maximum pooling and mean pooling. The pooling process can be expressed as follows:

$$x_j^l = f(\text{pool}(x_i^{l-1}) + b_j^l) \quad (2)$$

x_i^{l-1} is the value of the i window in the layer $l-1$ input feature, b_j^l is the offset of the j window in layer l , and pool represents the sampling function.

2.3. Full Connection Layer

The convolution neural network generally configures a full connection layer after the convolution layer and pooling layer. The neurons in the full connection layer are connected to each other in the previous layer. The calculation formula is as follows:

$$x^l = f(u^l) \quad (3)$$

$$u^l = W^l x^{l-1} + b^l \quad (4)$$

$f(u^l)$ is the activation function, W^l is the weight of layers $l-1$ to l , b^l is the offset of layer l , x^{l-1} is the output feature of layer $l-1$.

2.4. Classification Layer

The effect of convolutional neural network model largely depends on the selection of features and classification layer classifier. In general, if you have good features, even if you choose a simple classifier, such as Support Vector Machine (SVM), you can get good results. However, when SVM is used in large-scale detection applications, it is usually constrained by time and space, and its efficiency is not high. In this paper, Softmax is used as classifier [3]. Softmax is suitable for multi classification. Its expression function is as follows:

$$A_0(x^{(i)}) = \begin{bmatrix} p(y^{(i)} = 1 | \langle x^{(i)}; \theta \rangle) \\ p(y^{(i)} = 2 | x^{(i)}; \theta) \\ \dots \\ p(y^{(i)} = k | x^{(i)}; \theta) \end{bmatrix} = \frac{1}{\sum_{i=1}^k e^{\theta_j^T x^{(i)}}} \begin{bmatrix} e^{\theta_1^T x^{(i)}} \\ e^{\theta_2^T x^{(i)}} \\ \dots \\ e^{\theta_k^T x^{(i)}} \end{bmatrix} \quad (5)$$

The output of the function is a k -dimensional vector (the sum of the vector is 1), which is used to represent the probability value of the k estimates. That is to say, for a given input x vector of k dimension, the probability $p(y = i | x)$ of each category j is calculated to achieve the purpose of estimating the probability value of each classification output result of input x .

3. System Structure and Testing

Collect all kinds of goods pictures and label classification, through the TensorFlow training model for the identification of goods in the picture training. The picture of the goods captured by the camera is transmitted to the software system, and the software system identifies the goods by the trained model, finds it through the database, calculates the price and outputs it.

3.1. Collect and Mark Goods

Label the goods and use the LabelImg tool. Each time a picture is marked, a xml file is generated, and these labeled xml files are placed in two directories according to the training set and the verification set.

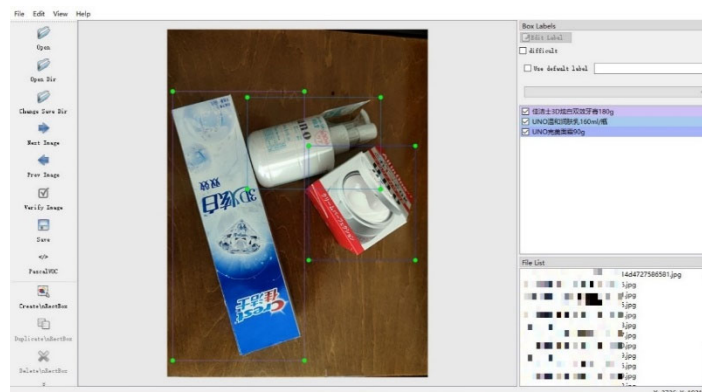


Figure 2. LabelImg software interface

3.2. Convert to Csv File

The xml file is divided into training set and test set, which are converted into csv files respectively.

	A	B	C	D	E	F	G	H
1	filename	width	height	class	xmin	ymin	xmax	ymax
2	93e751779	3258	4344	佳洁士3D炫白牙膏	69	775	1716	4135
3	93e751779	3258	4344	UNO温和洁面乳	1002	849	2669	1995
4	93e751779	3258	4344	UNO完美面霜	1769	1455	3102	2882
5	IMG_20200618_121111	1311	5792	佳洁士3D炫白牙膏	1395	163	2558	5227

Figure 3. Convert to csv file

3.3. Convert CSV files to TFRecord format

TFRecord format is developed by TensorFlow. It is a binary file that combines picture and label data. It can make better use of memory mechanism to achieve fast read, copy, move, storage and other functions.

3.4. Use the Model Garden of TensorFlow for Training

TensorFlow Model Garden is a repository that contains many implementations of the latest (SOTA) models and modeling solutions for TensorFlow users. Use the SSDMobilenet, to detect the product in real time faster.

3.5. Tensorboard Dynamic Monitoring Training.

Using TensorBoard tool can monitor the training process through a graphical interface. TensorBoard has many index curves and has powerful functions. For example, you can look at the overall training through Total_Loss.

F. Practical application effect



Figure 4. Test result

4. Conclusion

The visual checkout system based on convolutional neural network improves the shopping experience of users, solves the problem of low efficiency when cashiers or customers scan goods bar codes manually, and greatly reduces the dependence on hardware and human requirements in human-computer interaction. At the same time, the visual checkout system also gives customers more self-choice when shopping, which satisfies the customer's "pay-and-go" shopping experience. Based on the convolution neural network technology, this paper designs and studies the visual checkout system to help users quickly identify the selected goods in the settlement process and improve the user shopping experience in the intelligent retail process. at the same time, improve the efficiency of user self-settlement in the process of offline retail, which proves that it is possible to apply convolution neural network to automatic checkout in retail industry. It has certain research and engineering application value.

References

- [1] Karen Simonyan, Andrew Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition [J]. Arxiv.org, 2014, arXiv:1409.1556.
- [2] Sonka M, Hlavac V, Ceng R B D M. Image Processing, Analysis and Machine Vision[J]. Journal of Electronic Imaging, 2014, xix(82):685–686.
- [3] Yoo H J. Deep Convolution Neural Networks in Computer Vision[C]. Ieie Transactions on Smart Processing & Computing, 2015, 4(1):35-43.
- [4] Hu J, Shen L, Sun G. Squeeze-and-excitation networks. in: Proceedings of Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, 7132–7141.
- [5] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition. in: Proceedings of Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, 770–778.