Fast-Moving Coin Recognition Using Deep Learning

Yufeng XIANG

A report submitted to Auckland University of Technology
in partial fulfilment of the requirements for the degree of
Master of Computer and Information Sciences (MCIS)

2020

School of Engineering, Computer and Mathematical Sciences

Abstract

It is challenging to identify fast-moving small objects in motion pictures. Recurrent neural networks have also developed rapidly in recent years. RNN is more suitable for dealing with time series-related problems, such as language recognition and text recognition. Video as a kind of time series, is about how to use RNN for video prediction and improve the identification of small objects in the video.

In this report, we will discuss the rapid recognition of coins through deep learning and verify coins recognized during movement at a speed that our human eyes cannot see clearly. A method to improve the recognition rate through the combination of recurrent neural network and convolutional neural network has been proposed in this report, whose performance is excellent.

Keywords: Small object recognition, Deep learning, Image classification, Faster R-CNN, Long short-term memory (LSTM)

Table of Contents

List of Figures

List of Tables

List of Algorithms

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Signature: Yufeng Xiang Date: 05 June 2020

Acknowledgment

This research work was completed as the part of the Master of Computer and Information Sciences (MCIS) course at the School of Computer and Mathematical Sciences (SCMS) in the Faculty of Design and Creative Technologies (DCT) at the Auckland University of Technology in New Zealand. I would like to deeply thank my parents for the financial support they provided during my entire time of academic study in Auckland.

My sincerest thanks are to my primary supervisor Dr. Wei Qi Yan, who has provided me with much appreciated technological guidance and support. I believe that I could not have been able to achieve my master's degree without his invaluable help and supervision. Besides, I would like to appreciate my secondary supervisor Dr. Parma Nand and school administrators in our school for their support and guidance through the MCIS in the past years.

In addition, I would like to thank my friends Zhao Kun, Anna and Sun Shouming. appreciating them for giving me much help and guidance. By discussing with them, I solved the problems and difficulties I encountered in the experiment.

Finally, I hope to thank my parents again for their encouragement and financial support to help me successfully complete my studies.

Yufeng XIANG

Auckland, New Zealand

June 2020

Chapter 1

Introduction

There are five parts included in this chapter. The content consists of background and motivation, research questions, contribution, the objective of the research project and structure. Realizing the recognition of fast-moving coins is a challenging problem. And the recognition of fast-moving coins can be applied to the application of other small objects and has a wide range of uses. In this chapter, research questions and research objectives will be discussed. The structure of this report will be summarized at the end of this chapter.

1.1 Background and Motivation

Along with the accelerated development of technology, computer vision has also developed rapidly. In many applications, the recognition ability of computer vision has surpassed human vision system (Fossati, Gall, Grabner, Ren, & Konolige, 2012 Ren, & Konolige, 2012). Today, intelligent monitoring equipment is almost everywhere. Because each camera has a different functionalities, they are installed at every corner of our society (Welsh & Farrington, 2009), e.g., the cameras for protecting our security in banks or shopping malls, the cameras for monitoring traffic flow on highways, the cameras for shooting illegal vehicles at intersections, etc. At the same time, the shooting functions of devices such as mobile phones are becoming more and more powerful, the camera of a mobile phone may have more functions (Chesher, 2012). Moreover, with the development of autonomous driving technology, many cars also have video shooting devices with various features.

The extensive use of monitoring equipment has also brought challenges to the recognition of screen content. Usually, the objects that occupy most of the screen are more natural to be recognised (Filonenko & Jo, 2016). Since the monitoring device regularly collects a 2D image, usually, the scene captured by a camera is a 2D image. Objects far away from the shooting device in the picture occupy a smaller area in the image, and objects closer to the camera device will look like more prominent in the image (Kawakita et al., 2000). Therefore, how to identify small objects moving fast on the screen has become one of the problems.

The recognition of small objects will improve recognition efficiency. This means that many objects can be identified in the same picture. Usually, smaller items are challenging to be identified, or the accuracy of identification is relatively low. The recognition technology of small objects can be applied to the fields of security monitoring, face recognition, and automatic driving (Konoplich, Putin, & Filchenkov, 2016). Coins are small objects. In this report, we will discuss how to achieve fast-moving coin recognition through deep learning.

1.2 Research Question

This report is based on deep learning to achieve fast recognition of coins. When a small-sized object moves quickly on a screen, the captured image will be blurred, which make it challengeable to identify the object. Realising the rapid recognition of small objects helps to improve the recognition efficiency and recognition range. Therefore, the main research questions of this report are:

Question:

How to use deep learning to identify fast moving coins?

From this question, we can extend the following questions:

Why use deep learning to complete the recognition of coin?

How to further improve recognition accuracy?

The purpose of this report is to identify fast-moving coins through deep learning. To verify the recognition of fast-moving small objects, our coins in life will be used as experimental objects. We also use this method to test the experimental results.

1.3 Contributions

The identification of small objects is essential for intelligent monitoring. The main goal of this report is to achieve the detection of fast-moving object through deep learning as well as improve the detection speed and accuracy. The experiments in this report are mainly carried out in MATLAB software. The recognition of coin is done in four parts, including data collection, data pre-processing, training network, and tests.

In the next chapter of this report, we will introduce application of fast-moving coin recognition, MATLAB software, detection methods and the reasons for using deep learning to detect small objects. At the end of this report, the experimental results will be analysed and summarised.

1.4 The objective of the research project

The primary purpose of this report is to introduce the recognition of fast-moving coin and increase the speed of recognition. The entire research project will be based on deep learning in which report will demonstrate and evaluate the theories and principles needed for recognition.

To achieve the goal of coin recognition, the specific research process is divided into data collection, video pre-processing, training network, and result testing.

By the end of this report, we will compare the various methods to find a suitable method. Then we verify how to improve the recognition accuracy of small objects.

1.5 Report Structure

The structure of this report is as follows:

In Chapter 2, the literature will be reviewed. Fast-moving object recognition methods are surveyed in related fields. We explain the applications and significance of coin recognition. At the same time, relevant knowledge will be introduced.

In Chapter 3, we will introduce the research method and explain the detailed experimental process. A total of three different experiments were carried out in this chapter. Relevant results were obtained through tests.

In Chapter 4, the results obtained are discussed and analysed. Several network structures are compared. At the end of the chapter, we explain the limitations.

In Chapter 5, we summarize the report and introduce our future work.

Chapter 2 Literature Review

The focus of this report is to review the rapid recognition of fast-moving coins. In this chapter, the existing image recognition technology will be reviewed. We analyse the advantages and disadvantages of different methods as well as the development of object recognition technology.

2.1 Introduction

Intelligent surveillance has been developed rapidly. More and more cameras with multiple functions (B. Sun & Velastin, 2003) have been produced. For example, various public places are installed with security cameras, highways are accountable for monitoring road traffic, intersections are responsible for recording vehicles that violate regulations, multiple cameras are responsible for environmental detection, etc. (Ide, Jackson, & McGonnigle, 2006 2006). More and more functions can be performed by utilizing intelligent monitoring equipment, not just for recording videos. However, most of these functions need to be implemented by detecting objects from videos (Kwak & Song, 2013). For example, face recognition needs to recognize a person in a video, the automated driving function needs to distinguish roads, traffic signs and cars (Hsu & Huang, 2001).

In image recognition, it was mainly based on manual differentiation, which is necessary to distinguish each object manually. In 2012, with ILSVRC (ImageNet Large Scale Visual Recognition Challenge) competition, AlexNet reduced the error rate to 15% for the first time (Pathak, Pandey, & Rautaray, 2018 2018). Since then, convolutional neural networks have been used in image segmentation and recognition. At present, the use of deep convolutional neural networks for detection is the most accurate way, a high recognition accuracy can be obtained (Russakovsky et al., 2015).

In the process of image recognition, the ability to accurately recognize objects is essential. How to use convolutional neural networks to identify small objects is a challenge. For example, we use videos to identify coins moving in a video.

Coin recognition can be used in many scenarios. For instance, in a bank, we can quickly count the amount of money by identifying the coins in the video screen. For example, it can replace the traditional mechanical coin identification device. The identification of small objects can also be used in the field of security. Today, there are many ways to recognize faces in an image, but through the recognition of small objects, threats can be discovered earlier. For example, through monitoring in the bank, we can

find in time that the person is holding a wallet, cash or a knife (Pérez-Hernández et al., 2020).

Figure 2.1: Small object recognition example

This report will use coins as experimental objects to verify the use of deep learning to identify fast-moving coins. We test the experimental results by identifying the fast-moving coins in a video.

2.2 Recognition methods

Firstly, the report will introduce how to identify coin. Through our survey, it is found that there are three ways to recognize coins. They are divided into traditional recognition methods, machine vision, and deep learning recognition. We will introduce three methods below.

2.2.1 Traditional Recognition Methods

The most common method in life is to identify coins by using manual or equipment methods. For example, in some financial sectors, sorting coins manually will take a lot of time (Chavan, Fernandes, Dumane, & Varma, 2020).

The traditional method is to determine the size and weight of the coin. This is also the most common recognition method in daily life. The vending machine was invented in 1941, at that time, it was mainly identified by the size of the coin (Segrave, 2015). Our conventional vending machines, street toll collection devices, and coin-operated public telephones all recognize coins by using the size and weight (Khashman, Sekeroglu, &

Dimililer, 2006a 2006b). Therefore, a significant drawback of the traditional identification device is that it cannot distinguish the authenticity of coins.

Using metal objects of the same size and weight is naturally to deceive the conventional identification device (Dabic, 2013). In advanced vending machines, electromagnetic induction devices will also be added. We identify the magnetism and material of coins by using electromagnetic induction devices. This method is used to prevent fraudulent identification devices. Different currencies are often made of different materials, so the magnetic fields generated by different elements are various. The device judges the authenticity of coins by using pre-recorded magnetic field information (Khashman, Sekeroglu, & Dimililer, 2006b 2006a). Stability is the main advantage of traditional identification methods. After years of use and improvement of this method, this method has a relatively mature technology (Wells & McGlone, 2002).

We proposed the way of using the device to identify coins. How to realize the identification of coins or objects of similar size on the screen is still very difficult. It was mainly through manual observation to identify them (H. Kim, Kim, & Kim, 2016). The main problem of manual recognition is that the recognition efficiency is low and it is difficult to recognize fast moving small objects. In the case of concentration, the average response time of the human eye is 250ms-420ms. When recognizing an object far away or passing a device such as a monitor, the recognition rate will drop significantly (Kirchner & Thorpe, 2006). This is the reason why most industrial environments or dangerous areas use machine vision instead of human eye recognition.

2.2.2 Machine Vision

Machine vision refers to the use of sensors such as cameras and camcorders to cooperate with the machine to learn related algorithms so that the device can realize the function similar to the human eye. Using machine vision can recognize the features of object recognition, measurement and detection (Jain, Kasturi, & Schunck, 1995). Machine vision is a part of computer vision. Machine vision has a history of more than 40 years.

Early machine vision is usually used in industrial environments because machine vision can replace human eyes for recognition, consequently machine recognition is more used in dangerous industrial situations. The machine recognition can find details that are difficult for human eyes to find (Sonka, Hlavac, & Boyle, 2014). For example, machine identification is often used in industrial welding, automobile manufacturing, medical diagnosis and other fields. Machine vision matches the relevant information according to the pixel distribution, brightness, colour and additional information captured by the shooting device to identify objects (Davies, 2004).

Machine vision recognition can often recognize beyond human eyes, which is the reason why it is widely used in industrial testing. Therefore, machine vision can also be used to identify small objects, such as coins. The process of machine recognition is usually first to capture the picture through the shooting device, then locating the object through the edge detection machine or shape. Finally, match the present features to complete the recognition. Machine vision recognition is usually divided into five parts: data collection, image pre-processing, feature extraction, feature screening and inference (Sonka et al., 2014).

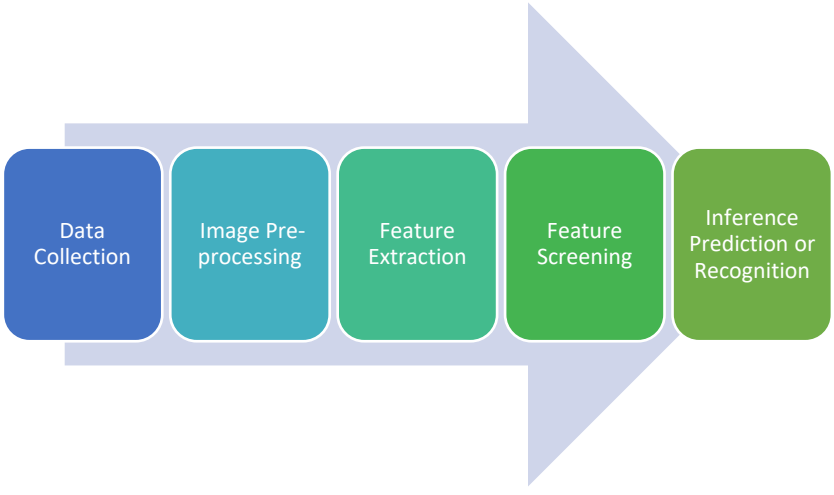


Figure 2.2: The pipeline of machine vision recognition

The Hamburg University of Technology used machine vision to identify coins. They used three colours of LED lights to illuminate the coin surface, then judging the metal

material by using different light reflected by different metal materials. This method can decide not only the authenticity of coins but also improve the accuracy of recognition (Hossfeld, Adameck, & Eich, 2003).

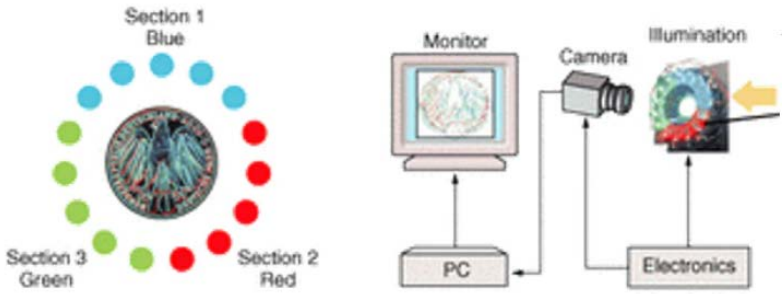


Figure 2.3: Machine vision recognition through three colours of lights

There are coins of various sizes and denominations circulating on the market in India. Coins of the same value all have different sizes and patterns. A device using machine learning to recognize currency symbols and numbers was invented. This device moves the coins to a specified position through a conveyor belt, and then uses multiple cameras to identify the coins according to the amount of coins under the specified lighting conditions. The recognition rate can reach 95% (Joshi, Surgenor, & Chauhan, 2016).

Although machine recognition has good accuracy, machine recognition is relatively slow. Therefore, machine identification usually repeats a single item inspection at a fixed location. On the other hand, machine recognition equipment is often relatively large. If it is used to recognize multiple objects, it needs to calculate a massive amount of data. Until 1999, after the NVIDIA GeForce 256 chip, the chip introduced GPU-related concepts. GPU can perform complex mathematical and logical operations, significantly improving the speed of transactions. This also allows machine recognition to be used in more fields. However, machine vision is difficult to identify small objects in motion.

2.2.3 Deep Learning

Although machine vision can obtain relatively high accuracy, the recognition computation is often relatively large. It is difficult to identify moving coins. Recognizing coins needs to be in a designated area and lighting environment. Therefore, a method of identifying coins by using deep learning is proposed (Schlag & Arandjelovic, 2017).

Deep learning is a part of artificial intelligence (Nilsson, 2014). The concept of artificial intelligence was firstly proposed in 1956, at the time to give computers running sophisticated programs related to human thinking (Russell & Norvig, 2002). Artificial intelligence is proposed to let the computer solve some problems by itself. Therefore, artificial intelligence is a broad concept (Genesereth & Nilsson, 2012).

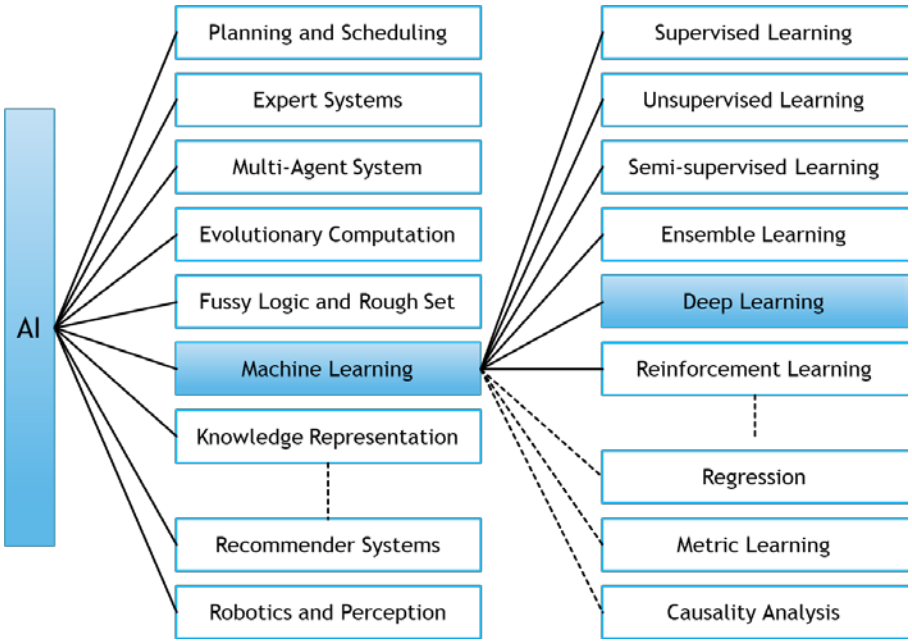


Figure 2.4: Relationship between deep learning and AI

Machine learning is a method to realize artificial intelligence. Machine learning usually uses algorithms and functions to solve relevant problems. For example, the most spam detections are from machine learning. The first spam detection is a method of filtering by using keywords, but the technique of filtering email does not adequately filter spam. Regular emails are also filtered because of keywords. In 2002, Paul Graham proposed to use a Bayesian algorithm to filter spam, the success rate of spam filtering was as high as 80% (Kågström, 2005).

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \tag{2.1}$$

Deep learning is usually based on neural networks and the features are extracted through multilayer neural networks. With the rapid development of computer hardware in recent years, deep learning has gradually become the primary processing method for complex data such as images, texts, languages, and signals (LeCun, Bengio, & Hinton, 2015). For example, the intelligent input method, voice recognition equipment, and automatic translation are all realized through deep learning technology (Goodfellow, Bengio, & Courville, 2016).

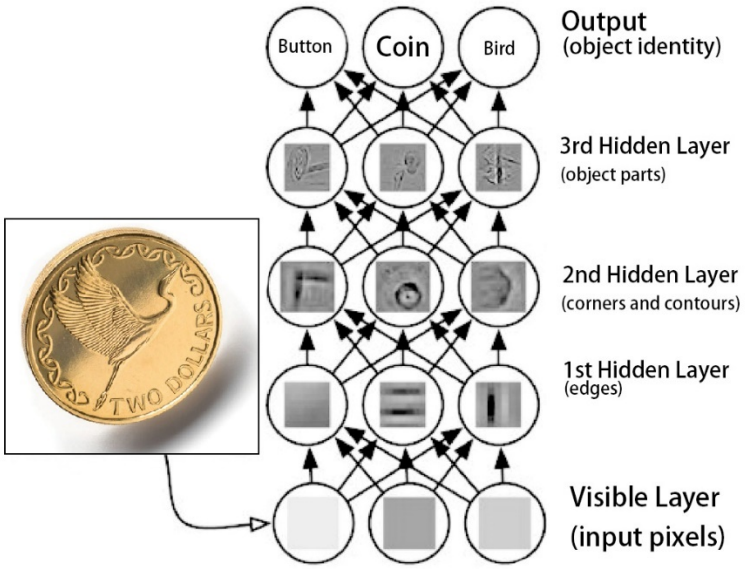


Figure 2.5: The layout of a deep learning neural network

Deep learning extracts features through deep neural networks and finally recognizes visual objects. The original prototype of the deep learning network LeNet was proposed in 1998 (El-Sawy, Hazem, & Loey, 2016). It uses a convolutional neural network. After inputting a letter picture, the picture passes through the convolution layer, the pooling layer and three fully connected layers, and finally outputs the result. The computer can recognize handwritten letters (Shi, Bai, & Yao, 2016). Today, deep learning is widely used

in the field of computer vision. Face recognition, image recognition, pedestrian detection and other scenes use deep learning methods. At the same time, deep learning has the characteristics of optimized maintenance and low cost (Goodfellow et al., 2016).

The characteristics of deep learning technology are very suitable for image detection. Compared with other methods, deep learning has the advantages of low cost and easy deployment (McAllister et al., 2017).

2.3 Image Classification

Image classification is significant to use deep learning to distinguish objects. Image classification is also the basis for image detection or tracking. The essence of image classification is to identify different types of images by various features (Wu, Yu, Huang, & Yu, 2015). Image classification methods based on deep learning can be performed in a supervised or unsupervised learning (Schmidhuber, 2015). In 2012, Alex and his mentor Hinton used AlexNet to achieve excellent results in the ILSVRC competition. Their AlexNet network ultimately won with a Top5 accuracy rate of 83.6%. Top5 accuracy rate means that after giving a picture, the five answers provided by the model include the correct answer as correct. The convolutional neural network also developed rapidly after this year. In 2013, the accuracy of AlexNet reached 88.8%; in 2014, VGG and GoogLeNet achieved an accuracy of 92.7% and 93.3%, respectively. ResNet has reached an accuracy rate of 96.43% in 2015, which has exceeded the average human accuracy of 94.9% (Gysel, Motamedi, & Ghiasi, 2016).

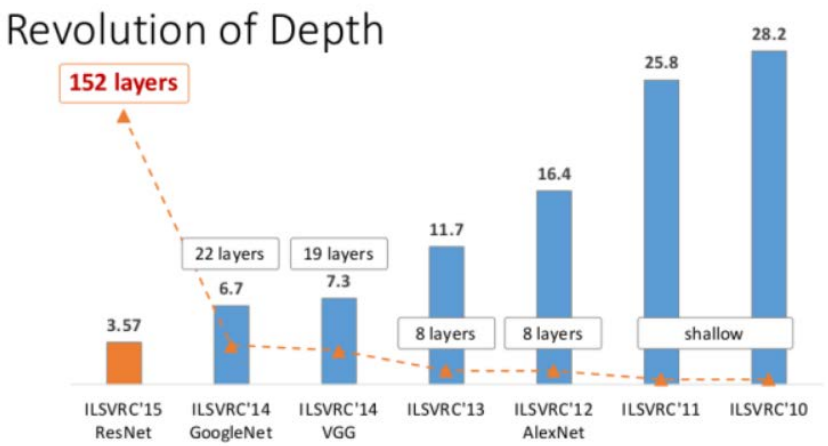


Figure 2.6: 2010-2015 ILSVRC ImageNet Classification top-5 error rate

2.3.1 CNN

A convolutional neural network contains convolutional layer, pooling layer and fully connected layer (Nguyen, Fookes, Ross, & Sridharan, 2017). Convolutional layer: perform convolution operations and extract the features from the bottom to the high level and explore the nature of the local association of the picture (Ye, Ni, & Yi, 2017). Pooling layer is used for downsampling operations. The max pooling or average pooling of the local block in the feature map is output by taking the convolution. The pooling layer can filter out unimportant high-frequency information (Sun, Liang, Wang, & Tang, 2015). Fully connected layer is the neurons in the input layer to the hidden layer are all connected. The fully connected layer can integrate local information with class distinction in the convolutional layer or the pooling layer (Sainath, Vinyals, Senior, & Sak, 2015).

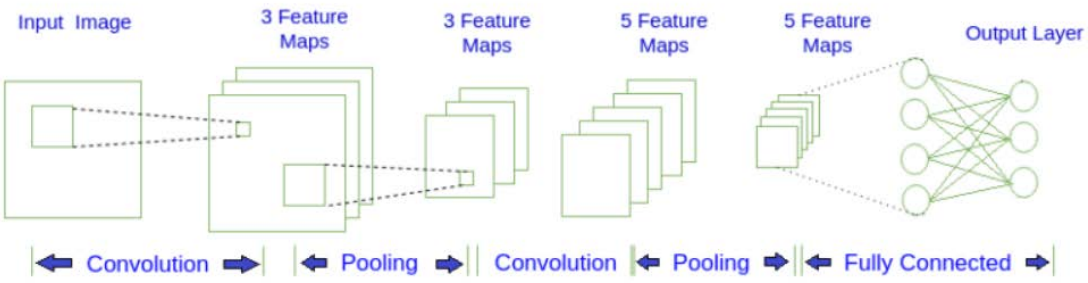


Figure 2.7: CNN model

Convolutional neural networks usually consist of the following types of layers. They include input layer, convolutional layer, activation layer, pooling layer, and fully connected layer. The active layer is also called Rectified Linear Units layer (ReLU) layer (Yang, Choi, & Lin, 2016). By superimposing different layers, a complete convolutional neural network can be constructed (Guo et al., 2016).

2.3.2 Convolutional Layer

In practical applications, the convolutional layer and the Rectified Linear Units layer are usually collectively called the convolutional layer (El-Sawy, et al., 2016). The convolutional layer is the core layer for constructing a convolutional neural network. The convolutional layer generates most of the calculation in the network (El-Sawy, et al., 2016). The role of the convolutional layer is to perform convolution, and each convolutional layer is regarded as a neuron. Because convolution has the feature of weight sharing, the convolution layer can reduce the number of parameters and prevent overfitting due to too many parameters (Ciaburro & Venkateswaran, 2017). The input layer is usually the first layer of the convolutional neural network, and the next layer is about the convolutional layer. In the convolutional layer, the characteristics of the previous layer are felt by a fixed-size field of view, which is called local receptive fields. The length of local receptive fields is usually 3, 5 or 7. It can be also specified by the user (Liu, Shen, & van den Hengel, 2015), when a colour image is an input to the convolutional layer, then three channels of RGB colours are generated. To generate the matrix of the next layer of the network, scanning at the perceptual field of view scans at uniform stride intervals becomes a feature map. The stride is usually one.

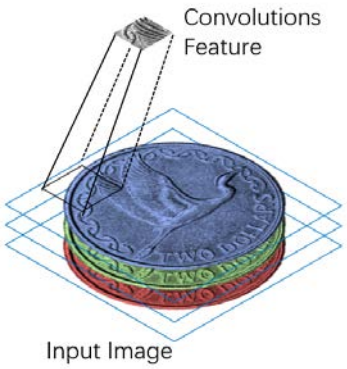


Figure 2.8: Convolutional Layer schematic

At the same time, the role of Rectified Linear Units layer is also very important in this layer. It can enhance the nonlinear function of the decision function and the entire neural network and will not change the spatial arrangement of the convolutional layer. ReLU can also solve the vanishing gradient problem (Dahl, Sainath, & Hinton, 2013). The formula of ReLU is: $f(x) = max(0, x)$. In each time, a convolution operation is performed, CNN applies a ReLU transformation to the convolution features in order to introduce nonlinear laws into the model. The ReLU function returns *x* for all values of *x*> 0, and 0 for all values of x $\leqslant$ 0 (Agarap, 2018).

2.3.3 pooling layer

The pooling layer is also significant in convolutional neural networks. It is a nonlinear form of downsampling. In convolutional neural networks, the pooling layer is usually added periodically. The role of the pooling layer is to reduce the size of the data space, the number of parameters and the amount of calculation required will also be reduced (Yu, Wang, Chen, & Wei, 2014). Common pooling layers are max pooling, average pooling, and L_2-norm pooling. Among them, max pooling and average pooling are more common. The pooling layer can also control overfitting and reduce the sensitivity of the convolutional layer to edges (Giusti, Cireşan, Masci, Gambardella, & Schmidhuber, 2013).

The pooling layer will calculate the output on each depth slice and move the pooling window through stride. For example, in Figure 2.7, the pooling window is a 2x2 pooling layer. By selecting the maximum value in each of the four groups of numbers, the maximum pooling method is used to reduce the amount of data by 75%.

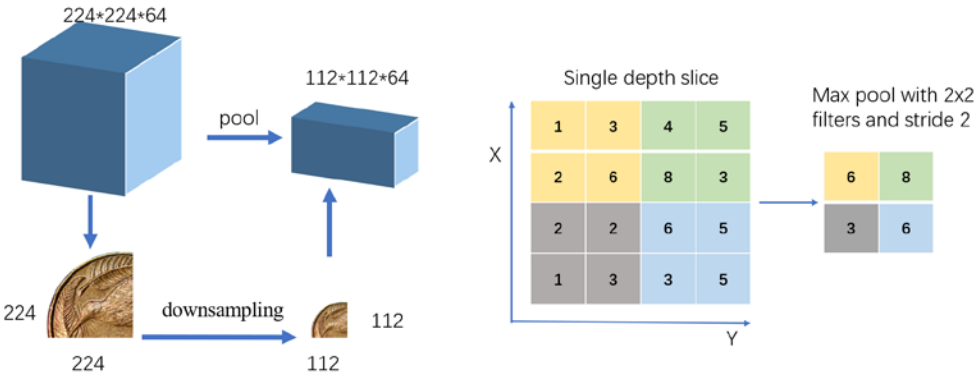


Figure 2.9: Max pooling

The use of neural networks for image detection will generate a massive amount of data. Reasonable use of the pooling layer can significantly reduce the amount of calculation and increase the processing speed (Giusti et al., 2013).

Max pooling is the most common way in neural networks, which means to retain the maximum value in the relevant field and discard other values. The advantage of max pooling is to save the strongest feature, which can leave more features, and reduce the problem of overfitting. The disadvantage is that the location information will be completely discarded (Schmidhuber, 2015).

Average pooling can reduce the increase in variance caused by using the large difference in neighbourhood differences. Average pooling can retain more of the overall features, which is more conducive to information transfer to the next module to extract features while reducing the dimensions (Schmidhuber, 2015).

2.3.3 Fully Connected Layer

There will usually be one or more fully connected layers at the end of the convolutional neural network. The convolutional layer extracts all local features, and the fully connected layer recomposes the previous local features to form a complete feature through the matrix (Sainath et al., 2015). The fully connected layer mainly refits the features to reduce the loss of feature information (Chen, Papandreou, Kokkinos, Murphy, & Yuille, 2014).

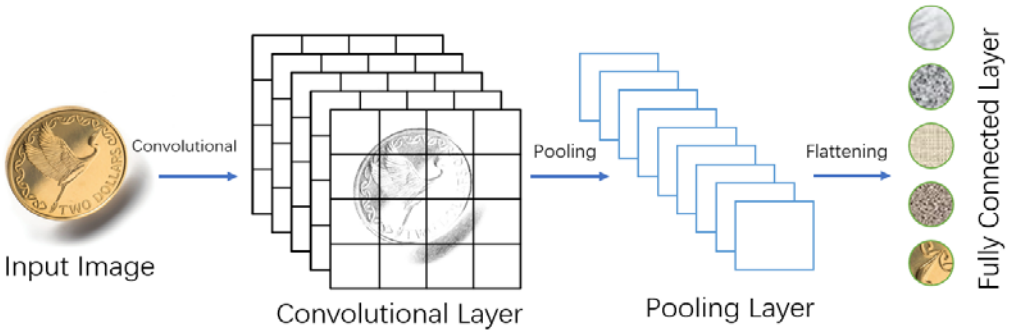


Figure 2.10: Fully connected layer location

2.3.4 CNN network structure

AlexNet was proposed by Alex Krizhevs, which is a deep network applied earlier to ImageNet (Alom et al., 2018). AlexNet has an eight-layer network with five convolutional layers and 3 fully connected layers. And the ReLU function is used. AlexNet has the advantages of fast training speed and high accuracy. AlexNet proved that the neural network can also use higher accuracy.

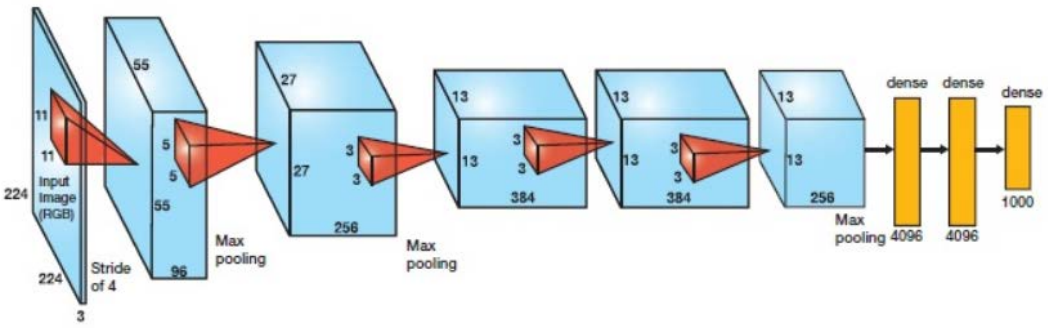


Figure 2.11: AlexNet structure diagram

VGGNet is improved on the basis of AlexNet. VGGNet was proposed by Oxford's Visual Geometry Group. VGGNet has a deeper network. VGGNet consists of a convolutional layer, a pooling layer, a fully connected layer and a softmax layer. The VGGNet network structure is consistent, and multiple convolutional layers can better extract features. The disadvantage of VGGNet is that it requires a lot of resources to calculate. It takes more time to train the network (Alom, et al., 2018). VGGNet proved that increasing the depth of the network can affect the final performance of the network to a certain extent (Schmidhuber, 2015).

GoogLeNet also has a good effect in image classification, GoogLeNet does not use VGG and AlexNet to deepen the network. GoogLeNet uses the Inception structure, which takes use of three convolutional layers and a pooling layer superimposed together to expand the width of the network, which can reduce the amount of calculation while ensuring the accuracy of the network (Ballester & Araujo, 2016).

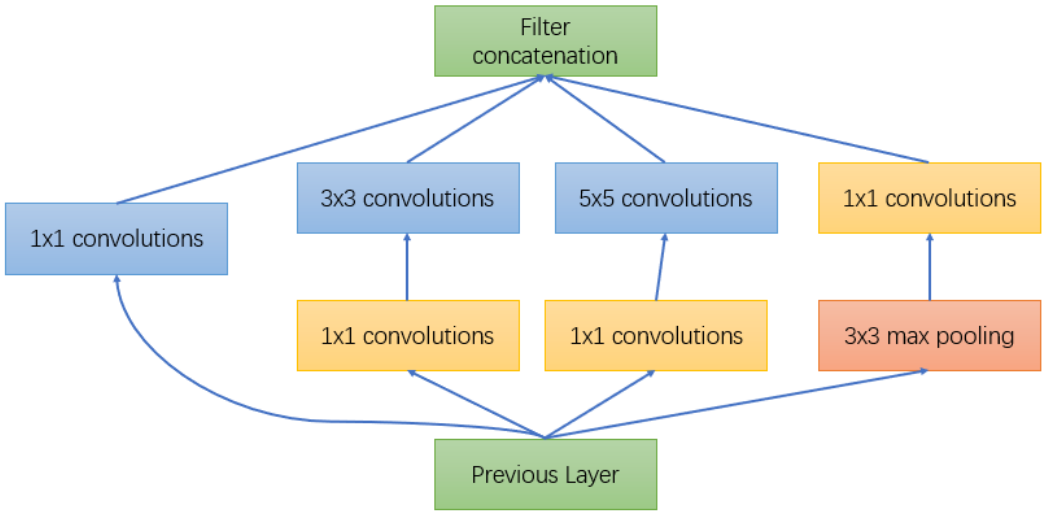


Figure 2.12: GoogLeNet Inception structure

If the network is too deep, it will become difficult to train, at the same time, there will be the problem of the gradient disappearing. However, when the network is deepened to a certain degree, the accuracy rate will be saturated, and continue to increase the

network depth will reduce the accuracy. To solve this problem, a residual network was proposed in 2015 (Targ, Almeida, & Lyman, 2016).

ResNet's residual block formula can be expressed as $\mathcal{Y}1 = \hbar(x_\ell) + \mathcal{F}(x_\ell w_\ell)$, $x_{\ell+1} = \not{f}(\mathcal{Y}_1)$, where x_ℓ and $x_{\ell+1}$ represent the input and output of the ℓ residual unit, respectively, each residual unit generally contains a multi-layer structure. F is the residual function, which represents the learned residual, $\not{f}$ is the ReLU activation function. The learning characteristics from shallow layer ℓ to deep layer L are

$$x_{\mathcal{L}} = x_\ell + \sum_{i=\ell}^{\mathcal{L}-1} F(x_i, w_i)$$

$$\frac{\partial_\varepsilon}{\partial_{x\ell}} = \frac{\partial_\varepsilon}{\partial_{x\mathcal{L}}}\frac{\partial_{x\mathcal{L}}}{\partial_{x\ell}} = \frac{\partial_\varepsilon}{\partial_{x\mathcal{L}}}(1 + \frac{\partial_{x\mathcal{L}}}{\partial_{x\ell}}\sum_{i=\ell}^{\mathcal{L}-1} F(x_i, w_i)) \tag{2.2}$$

ResNet usually uses a 3x3 convolution kernel to solve the problem of network degradation and ensure the complexity of the network. The residual module establishes a direct link between input and output, and the newly added layer can learn new features directly on the original basis (Targ et al., 2016).

2.4 Image Detection

Another essential part of item identification is image detection. Image classification is to classify objects, while image detection is to obtain the category information of objects while obtaining position information. This separates the target item from the background (Ouyang & Wang, 2013).

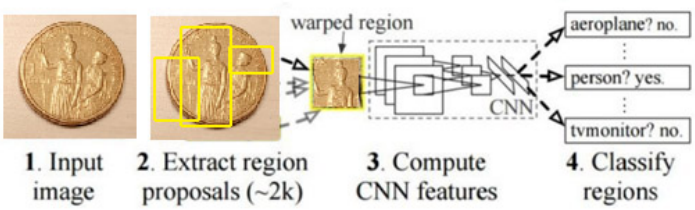


Figure 2.13: R-CNN structure

The appearance of R-CNN solves the problem of image detection very well. R-CNN abstracts detection into two processes. First, it proposes several regions that may contain objects based on pictures, called Region Proposals. Second, we run the classification network on these proposed regions to get the category of objects in each area (Tajane, et al., 2018). The R-CNN algorithm has a simple structure and directly converts the detection task to the classification task. But R-CNN has three serious shortcomings. The first is that it takes up a lot of disk space because the image corresponding to each selection needs to be extracted in advance. The second is that only fixed-size input images can be passed in. The third is the need for repeated calculations, resulting in backward performance (Girshick, Donahue, Darrell, & Malik, 2014).

2.4.1 Fast R-CNN and Faster R-CNN

R-CNN has the shortcomings of a long time and a large amount of calculation because it needs to run CNN separately on each proposal for classification. However, Fast R-CNN integrates *bbox* regression into the neural network and performs classification and regression at the same time after obtaining the features so that the recognition time can be shortened (Girshick, 2015).

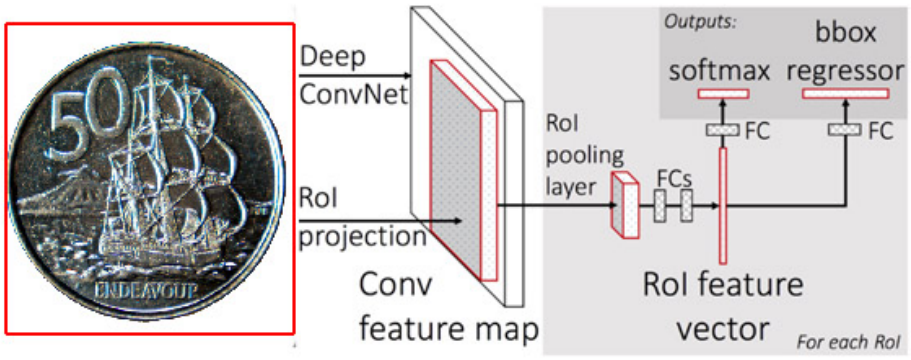


Figure2.14: Fast R-CNN structure

The main advantage of Fast R-CNN is that it accelerates R-CNN and simplifies the ROI pooling layer. It can improve efficiency while getting better training results.

The disadvantage of Fast R-CNN is that it is not efficient enough. Faster R-CNN adds the function of extracting edge neural network based on it. Faster R-CNN proposes to use RPN network instead of Selective Search, and the neural network also does the edge extraction. This also means that Faster R-CNN can further improve the efficiency of recognition (Ren, He, Girshick, & Sun, 2015). Through the steps of image detection, we can find the main differences between R-CNN, Fast R-CNN and Faster R-CNN.

<table>
<tr><td></td><td>R-CNN</td><td>Fast R-CNN</td><td>Faster R-CNN</td></tr>
<tr><td>Extract region proposals</td><td>Selective Search</td><td>Selective Search</td><td>RPN network</td></tr>
<tr><td>Extract features</td><td>CNN</td><td colspan="2" rowspan="2">CNN+ROI pooling</td></tr>
<tr><td>Feature classification</td><td>SVM</td></tr>
</table>

Table 2.1：Differences between R-CNN, Fast R-CNN and Faster R-CNN.

With the continuous advancement of algorithms, the process of deep learning detection targets is getting more straightforward, and the accuracy and speed of detection are gradually improving. To the date, Faster R-CNN can still do a lot of work, it has good accuracy (Jiang & Learned-Miller, 2017).

2.5 Recurrent Neural Network

The recurrent neural network is referred to as RNN for short, RNN is often used to process sequence data, such as text, sound and other data. Therefore, RNN is more used in translation, language recognition, text recognition and other fields. RNN is not as good as CNN in the area of image detection. A detection recurrent neural network consists of an input layer, a hidden layer and an output layer (Jain, Zamir, Savarese, & Saxena, 2016).

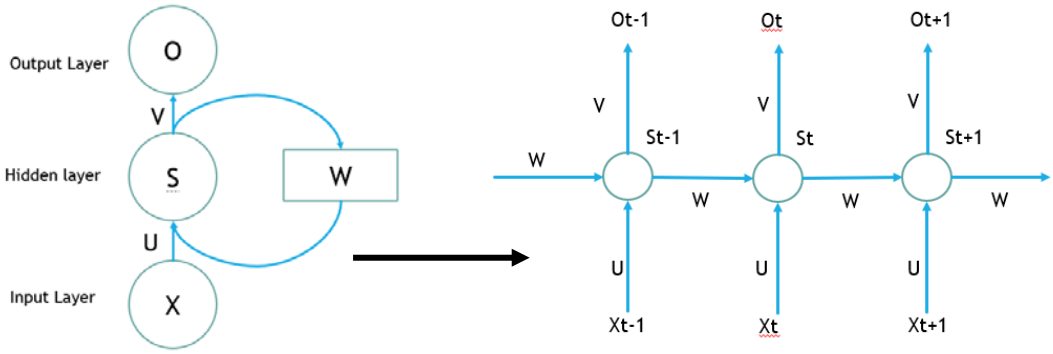


Figure 2.15：RNN structure

The formula for RNN is $O_t = g(V \cdot S_t)$ $S_t = f(U \cdot X_t + W \cdot S_{t-1})$ This means that the result of St does not only depend on Xt, also has an important relationship with St-1 in the previous second (Ciaburro & Venkateswaran, 2017). RNN is affected by timing, which can make it perform well in continuous data.

LSTM stands for Long short-term memory. LSTM is an artificial recurrent neural network (RNN) architecture in the field of deep learning. LSTM has feedback connections. It can deal with not only single data points like images, but also entire sequences of data such as speech or video (Gensler, Henze, Sick, & Raabe, 2016). In LSTM have four important part, Input Gate, Output gate, Forget gate and memory cell. The improvement of RNN by LSTM is mainly reflected in the increase of the weight control of the mind at different times through the gate controller, and the cross-layer connection is added to reduce the effect of the gradient disappearance problem. Long-term memory is retained on the original short-term memory of RNN. LSTM combines direct regards to the unique structure, instead of superimposing purely non-linear connections, to achieve better information dissemination (Gers, Schmidhuber, & Cummins, 1999).

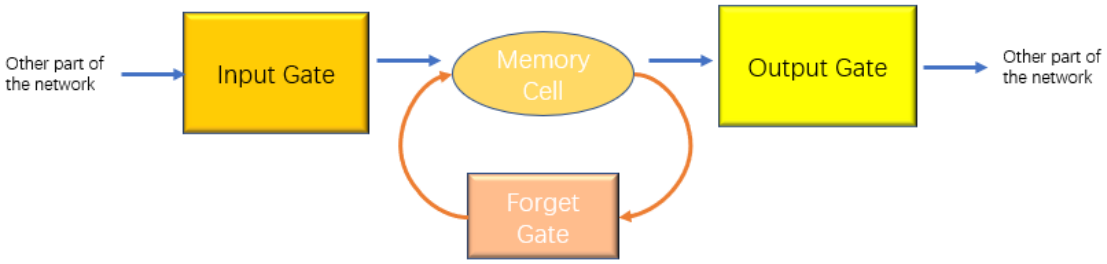


Figure 2.16: LSTM structure

Convolutional neural networks and recurrent neural networks are the same in that each layer of neural network can coexist with multiple neurons, there can be multiple layers of neural network links. The main difference between convolutional neural networks and recurrent neural networks is that CNN can have a deeper depth, CNN is better at feature extraction. The RNN can describe the continuous state in time and has a memory function (Wang et al., 2016).

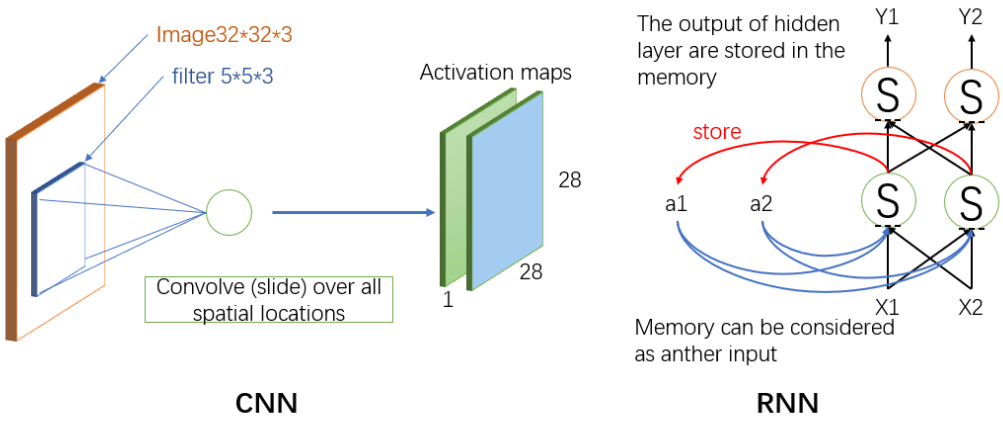


Figure 2.17: The difference between CNN and RNN

Chapter 3 Methodology

This chapter will introduce the experimental method. The purpose is to study the rapid recognition of fast-moving coin through deep learning. A total of three experiments have been conducted in this research project. This chapter will introduce the experimental process in the order of examinations.

The research problem in this project is the recognition of fast-moving coin. The purpose is to realize the recognition and detection of fast-moving small objects through deep learning. The experiment uses two-dollar coins as an example. CNN has excellent accuracy in image classification, but CNN can only be used for classification (Vedaldi & Lenc, 2015). The experiment will be conducted with other models. The experimental process is shown in the figure, including data collection, data pre-processing, model training, testing and results. The experiment is completed in MATLAB, the graphical operation interface is provided by MATLAB. Adding different kinds of toolboxes can achieve a variety of different functions (Higham & Higham, 2016).

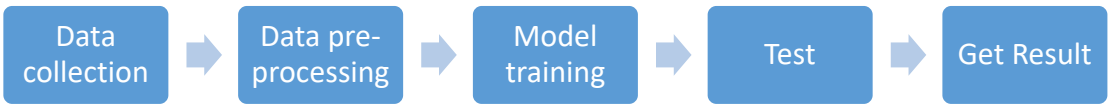


Figure 3.1: Experiment procedure

3.1 Initial experiment

The significance of the initial experiment is to be used as a comparative experiment for subsequent experiments. To verify the recognition of fast-moving coins through ordinary deep learning recognition methods, we use a coin with a face value of $2 as the identification sample.

3.1.1 Data Collection

Because the purpose is to achieve rapid recognition of coin, in the initial experiment, the fast-moving coin video was used as the data set for training. We use mobile phones as the leading shooting equipment. We collect videos of coins moving fast in different directions and currencies moving fast in different light environments. Through MATLAB, the video is divided into pictures. An average of one second of video can be divided into 30 images. A total of 1200 images were collected.

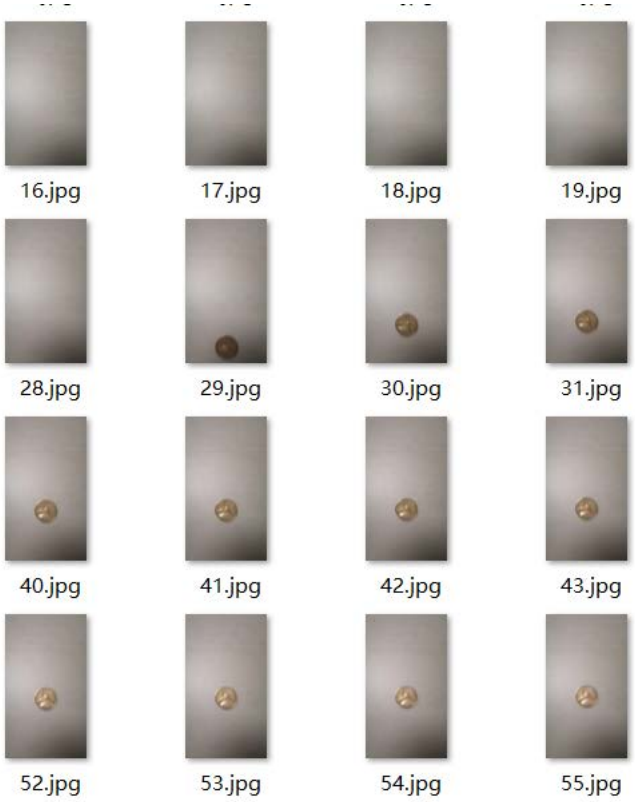


Figure 3.2: Data set sample

3.1.2 Data Pre-processing

In MATLAB, we use the image labeller plugin to label each image. In order to get better training results and reduce the impact of noise on the experiment, we manually mark the position of the coins in the picture.

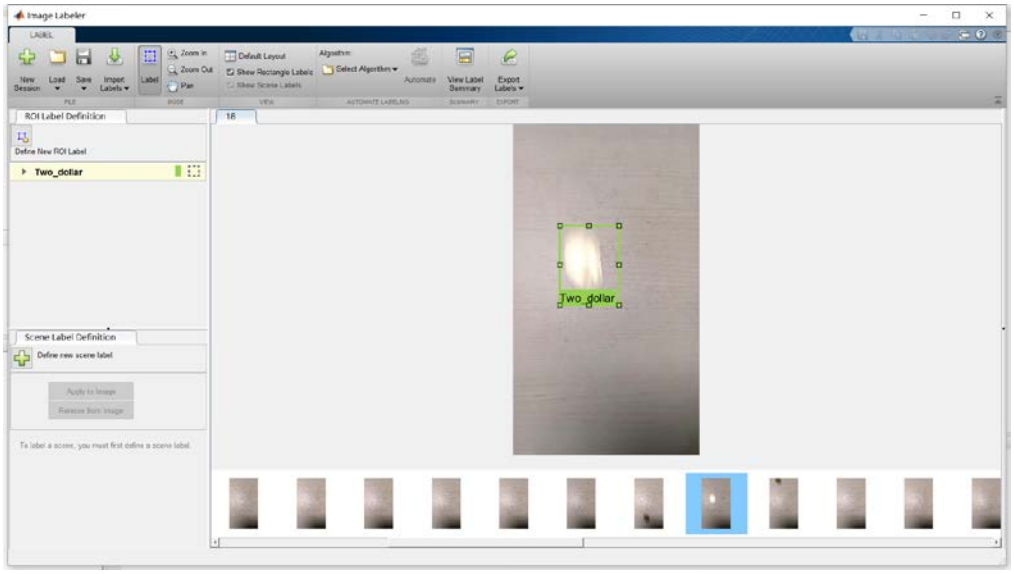


Figure 3.3：Label every picture

3.1.3 Sampling method and result

In the data set, a total of 70% is used for training, and 30% is used for testing. But the experimental results are very unsatisfactory. There are very few opportunities to recognize fast-moving coins. We used Faster R-CNN, YOLO got similar results. The recognition rate does not exceed 1%.

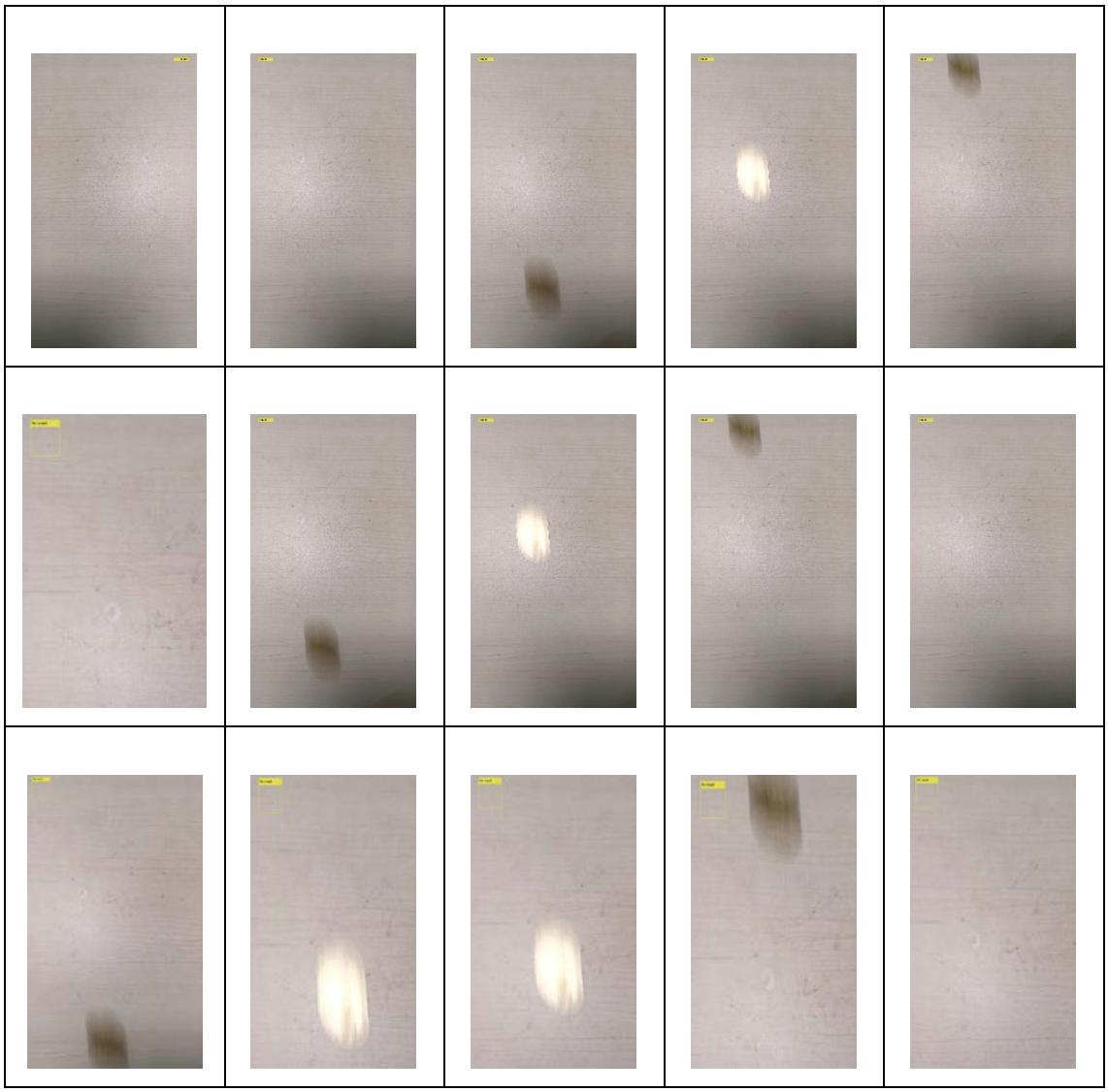


Figure 3.4: Unable to detect fast moving coins

During the experiment, the coin was moved from the bottom of the screen into the image by clicking the coin, sliding all the way to the top of the image, and finally moved out of the image. The movement speed of coins is breakneck, and the entire moving process of coins does not exceed one second. Figure 3.2 shows the graph of the coin test

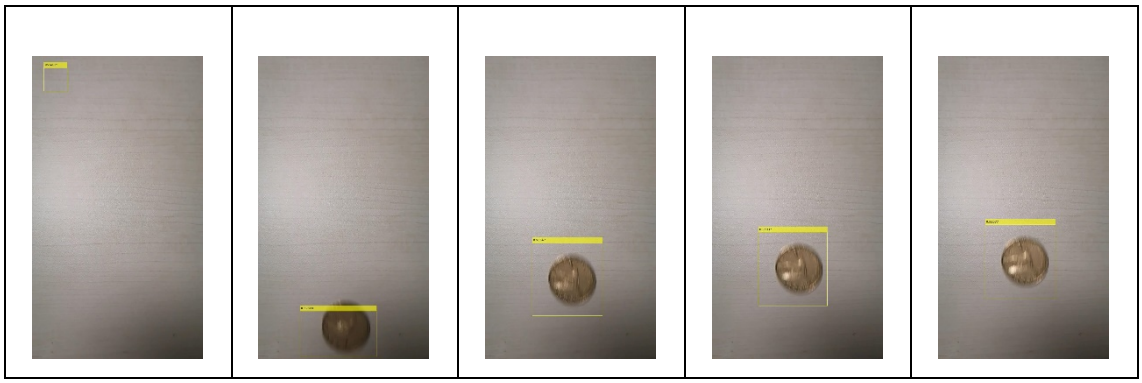
result. The coin slides through the screen three times. It is entirely impossible to recognize moving coins. This movement speed of the coin has exceeded the range that human eyes and cameras can recognize.

3.2 The Second Experiment：Slow-motion

According to the results of the first experiment, it is difficult to identify fast-moving coin through deep learning alone. Standard video can shoot 25 to 30 frames per second. When the moving speed of the object is too fast, a dangerous smear phenomenon will occur in the picture, which will affect object recognition.

In the second experiment, the experiment took the same steps as the first time. The difference lies in the use of the slow-motion camera to capture video. In the experiment, the video was shot through the slow function of the phone. Through the mobile phone's slow-motion video function, we can get 360 frames per second. But each video is limited to 10 seconds. Therefore, we took multiple video and change video to images. After that, we selected 3000 frames from the videos as a data set. We use the same percentage, 70% for training and 30% for testing. Using the same method as the first experiment, we manually mark each image. After that, the network was trained, and the following results were obtained.

Figure 3.3 is a screenshot of part of the verification video. We find that the results have greatly improved compared to the first implementation. The coin used as the test sample can be recognized when it moves quickly. Fig 3.3 has a yellow border around the coin indicates that the coin is recognized.



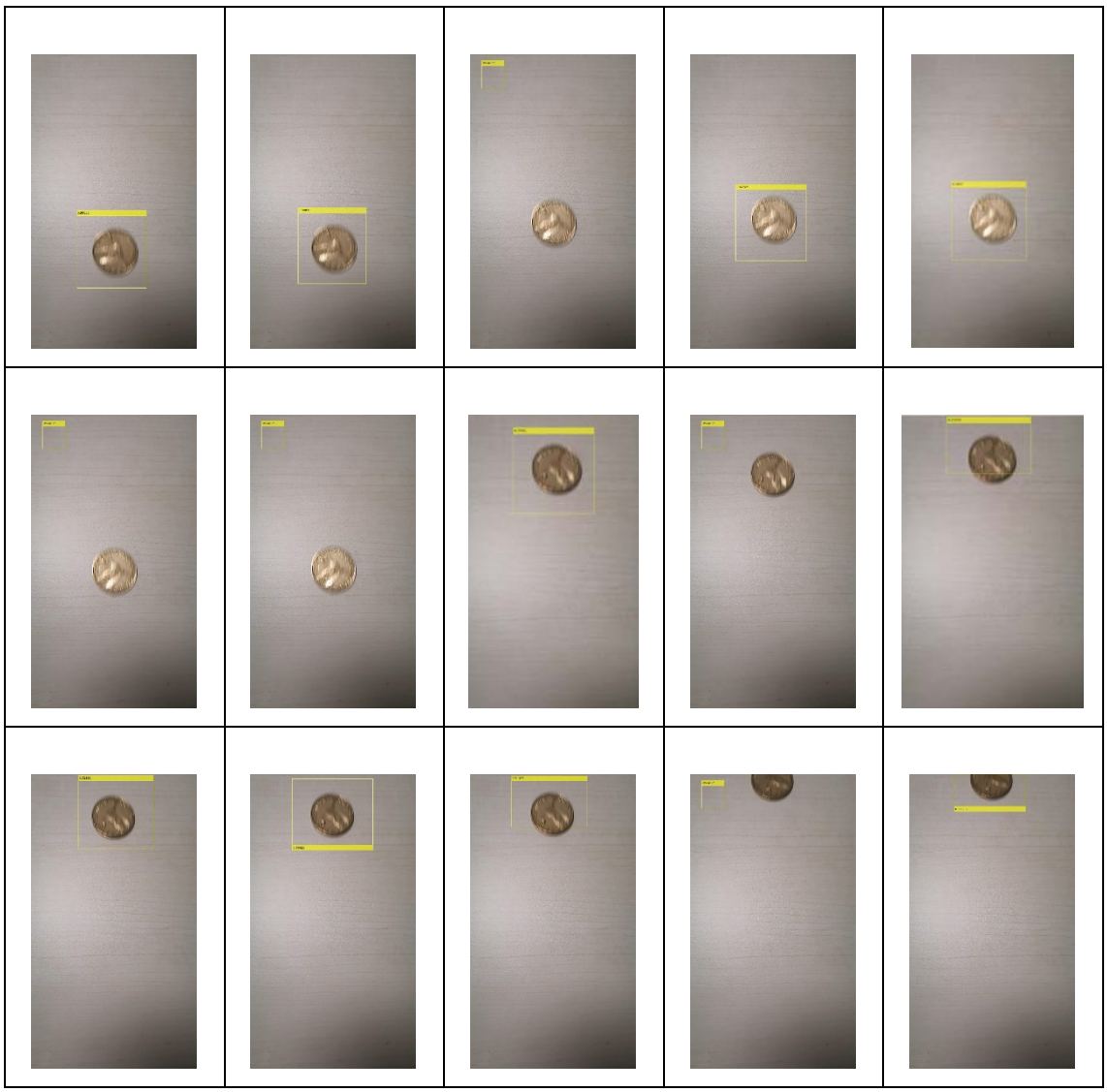


Figure 3.5: Slow motion result verification

By combining the slow-motion video, the experimental results must be improved compared to the first experiment. At the moment of recognition, it can be recognized as a coin with about 80% accuracy. According to the overall test data, the recognition rate is still not high. By calculating $Accuracy = {}^{NR}/_{TN}$, NR means the total number of pictures recognized, TN means The total number of pictures used to test accuracy. In the verification process using Faster R-CNN, the total number of pictures used for verification is 900, of which only 720 pictures can be accurately identified. Therefore, according to the proportion of the recognized picture in the overall picture, the above data is calculated.

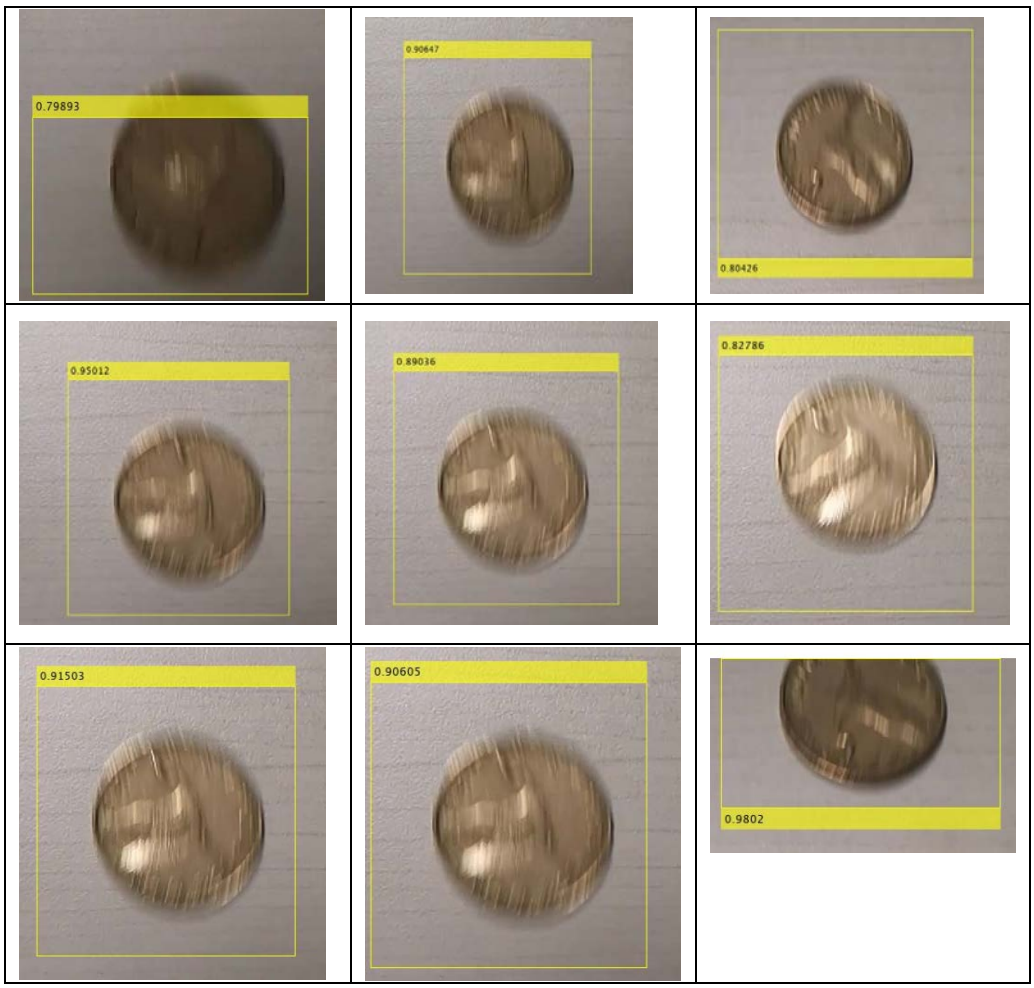


Figure 3.6: Coin recognition accuracy display

On the other hand, Figure 3.4 is a part of the picture that has been identified. We find that the recognized pictures have relatively high accuracy. This proves that the combination of slow shooting and depth recognition can be used for fast moving small object recognition.

3.3 The Third Experiment LSTM

In April 2020, MATLAB updated the new version of software 2020a. This version supports simultaneous use of LSTM and CNN and classification of videos. LSTM is Long short-term memory. Therefore, in the third experiment, LSTM and CNN are used in combination to quickly identify small objects. Try to improve the accuracy of continuous recognition.

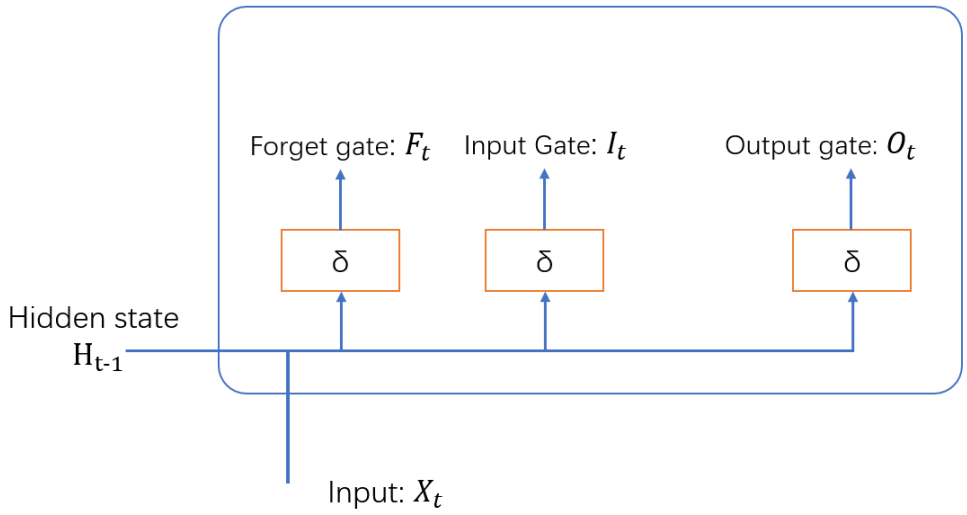


Figure 3.7: LSTM structure

LSTM input is the current time X_t, the hidden state at the last time is H_{t-1}, $W_{xi}, W_{xf}, W_{xo} \in \mathbb{R}^{h\times h}$ and $W_{hi}, W_{hf}, W_{ho} \in \mathbb{R}^{h\times h}$ is weight parameter, $b_i, b_f, b_o \in \mathbb{R}^{1\times h}$ is deviation parameter.

$$I_t = \sigma(X_t W_{xi} + H_{t-1} W_{hi} + b_i)$$

$$F_t = \sigma(X_t W_{xf} + H_{t-1} W_{hf} + b_f)$$

$$OI_t = \sigma(X_t W_{xo} + H_{t-1} W_{ho} + b_{io}) \quad (3.1)$$

LSTM can add direct links on the original basis so that the information can be better transmitted (Greff, Srivastava, Koutník, Steunebrink, & Schmidhuber, 2016). Each picture in the video belongs to a continuous sequence, so it is possible to predict the position of the next frame through LSTM.

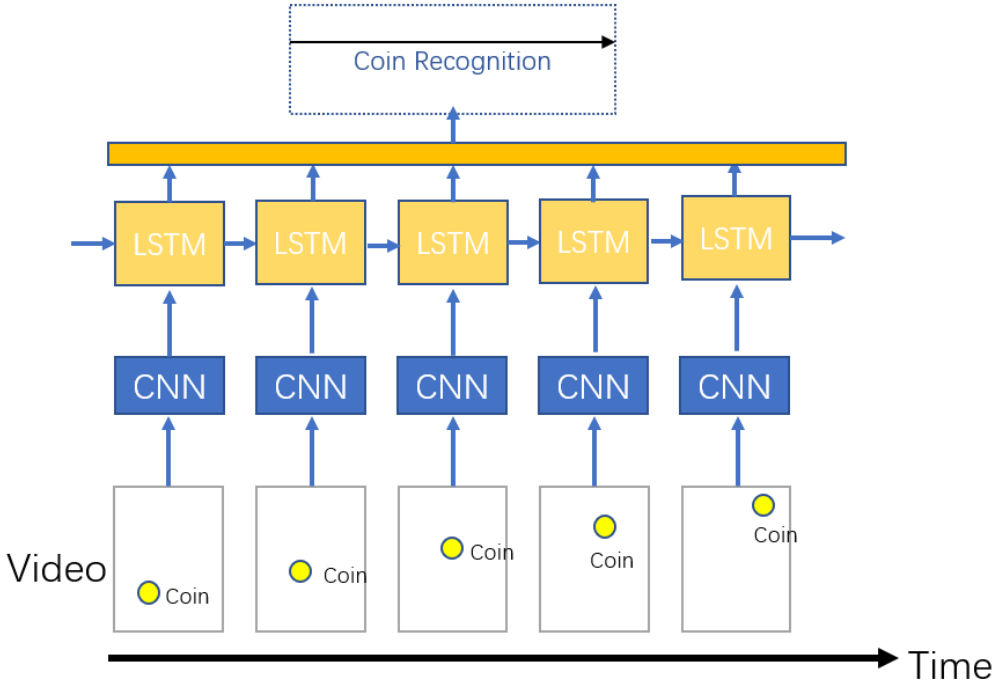


Figure 3.8: LSTM+CNN structure

The motion of objects in the video is also time-sequential. Through the functions provided by the latest version of MATLAB, CNN combined with LSTM is used to improve the recognition rate of small objects when moving fast. Also use coins as samples for training and testing.

First, we convert the video into a sequence of feature vectors and extract features from each frame. Then, we train the LSTM network to predict the video label. The trained network has the ability to predict the position of the next coin based on the direction of the coins in the video. The video processing method is as the following flowchart.

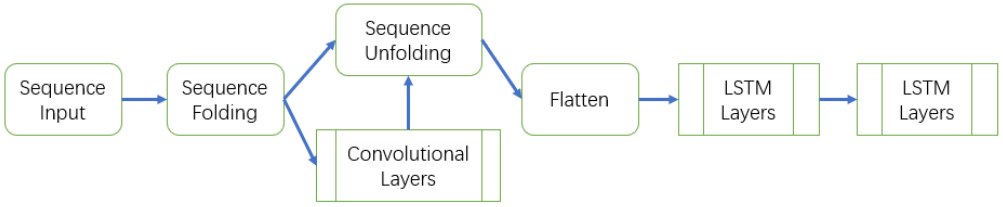


Figure 3.9: LSTM + CNN flowchart

The verification video is a coin spinning on the desktop. We can find that the

recognition ability is improved compared to the previous two experiments. Occasionally, the shadow of coins will be recognized in the recognition. However, the combined use of LSTM and CNN can achieve better tracking of fast-moving coins, and the recognition rate is higher than the classification accuracy using CNN network alone.

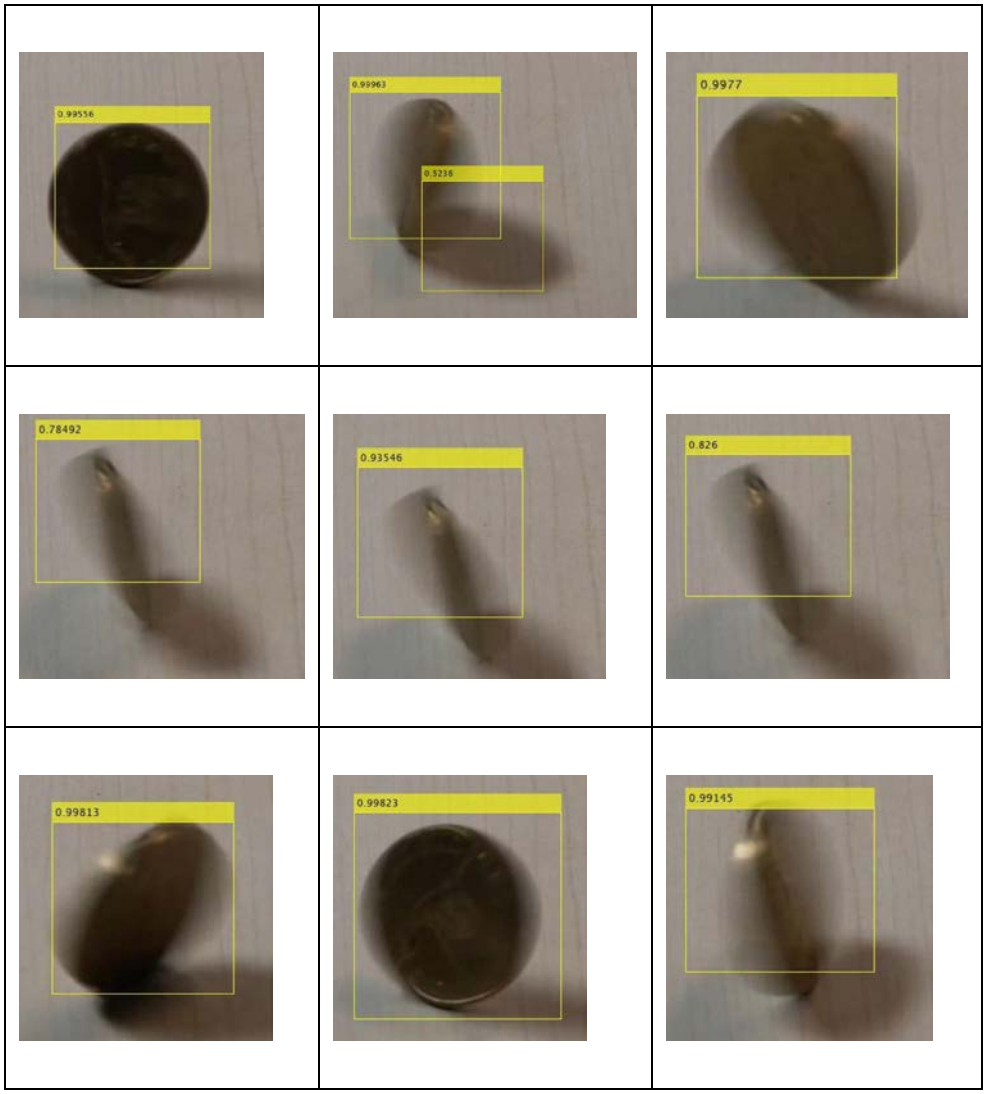


Figure 3.10: Verification results

Figure 3.10 is a partial screenshot of the third verification video. We found that the results have further improved compared to the previous two results. Therefore, we try to use a more difficult test method, we let the coin rotate at a high speed while moving on the table. The recognition rate of the test can still reach about 95%. In Figure 3.10, we can find that the identified coins have higher recognition accuracy than the previous two experiments through the numbers on the yellow border. This proves that LSTM combined with CNN can enhance the recognition rate of fast-moving objects.

Chapter 4 Results and Analysis

The main content of this chapter is to show the results of coin recognition. Through our experiments, we find that a suitable method can implement the rapid identification of fast-moving coin, and improve the accuracy of recognition to a certain extent. This chapter will identify the results and problems. In the end, the limitations of the experiment are explained.

4.1 Experimental Results

The purpose of this research project is to implement the recognition of coin through deep learning. Experiments show that deep learning can be used to identify fast-moving small objects. Using appropriate methods can effectively improve the accuracy of recognition. We use coins as experimental objects, a total of three experiments were conducted in this project. The results of the investigation have been improved through specific improvements. In the preliminary analysis, Faster R-CNN and YOLO were used to identify the coins. But the results are not satisfactory. Through our calculations, the number of pictures identified by using the two methods is less than 1%.

In the final experiment, the video was shot in slow motion, in which the combined use of LSTM + CNN method can improve the accuracy to more than 95%. This verifies that the method can be used for the rapid detection of small objects. At the same time, this result also has certain limitations. Only one coin moves in the video used for training and testing. When more interference items appear in an image, the accuracy will decrease.

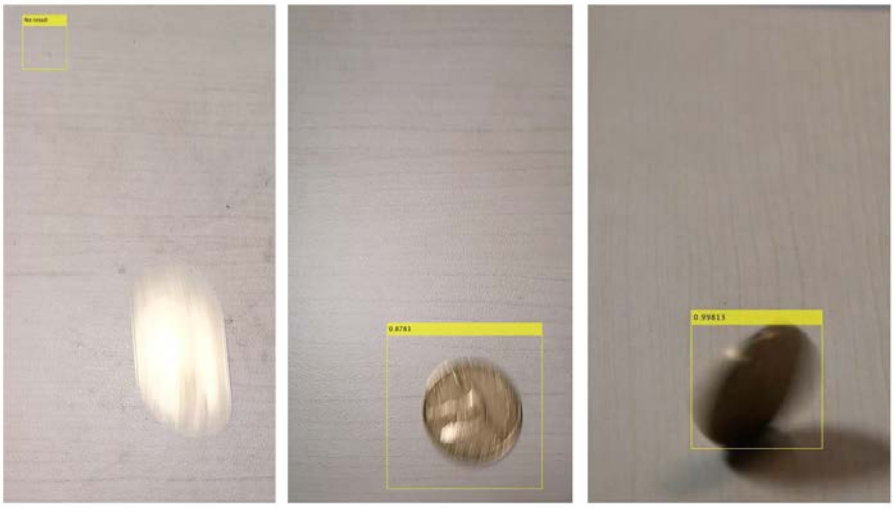


Figure 4.1: Comparison of experimental results

4.2 Result Analysis

Deep learning can accurately identify coins that move slowly. Also, it has more than 90% accuracy. After training the network, it can be recognized by ordinary computer cameras.

Figure 4.2: Coin recognition

But when the coin moves too fast, it becomes complicated to identify. When the moving object speed exceeds the shutter speed of the shooting device, only a blurred image can be obtained. When shooting in slow motion, part of the clear image can be captured for recognition. When the object moves faster, the video image captured by the camera will have a smear phenomenon, which not only makes the details of the coin lost and difficult to identify but also changes the shape of the object in the screen. For example, coins moving at high speed are displayed as elliptical blocks on the screen.

Figure 4.3: Normal shooting and slow-motion video

The picture taken with the slow-motion video of the mobile phone is still not clear enough, but some of the features that can be collected are used to identify the object. The slow-motion video shooting of the mobile phone is 360FPS, which means that 360 pictures can be obtained in one second. However, there is a 10-second time limit for shooting slow motion on mobile phones. In our project, it is found that the slow-motion video of the mobile phone cannot be obtained by shooting at a speed of 360 frames per second. The mobile phone uses an algorithm to add the transition image to the next picture so as to achieve the slow-motion effect (Narang, Agarwal, & Sanu, 2015).

Experiments also show that after the coin moves faster, our human eyes can recognize it. Through the combination of deep learning methods and shooting equipment, moving coins can still be identified. Through the screenshot, we find that though the state of the coin is blurred, the machine still has the ability to be identified.

Through our experiments, it is found that the resolution of the image will also have a specific impact on the recognition accuracy of small objects. We use the same model at the same time, the same number of data sets and training methods. Separately, we use different shooting equipment for data collection. Finally, we found that high-quality pictures can get higher accuracy (Huang et al., 2017).

Machine vision has exceeded the recognition range of human eyes (Mennel et al., 2020). For example, the following three images are very similar. But after zooming in on the image, you see that high-quality images have more details. This also means that more detailed features can be captured through deep neural networks (J. Kim et al., 2017).

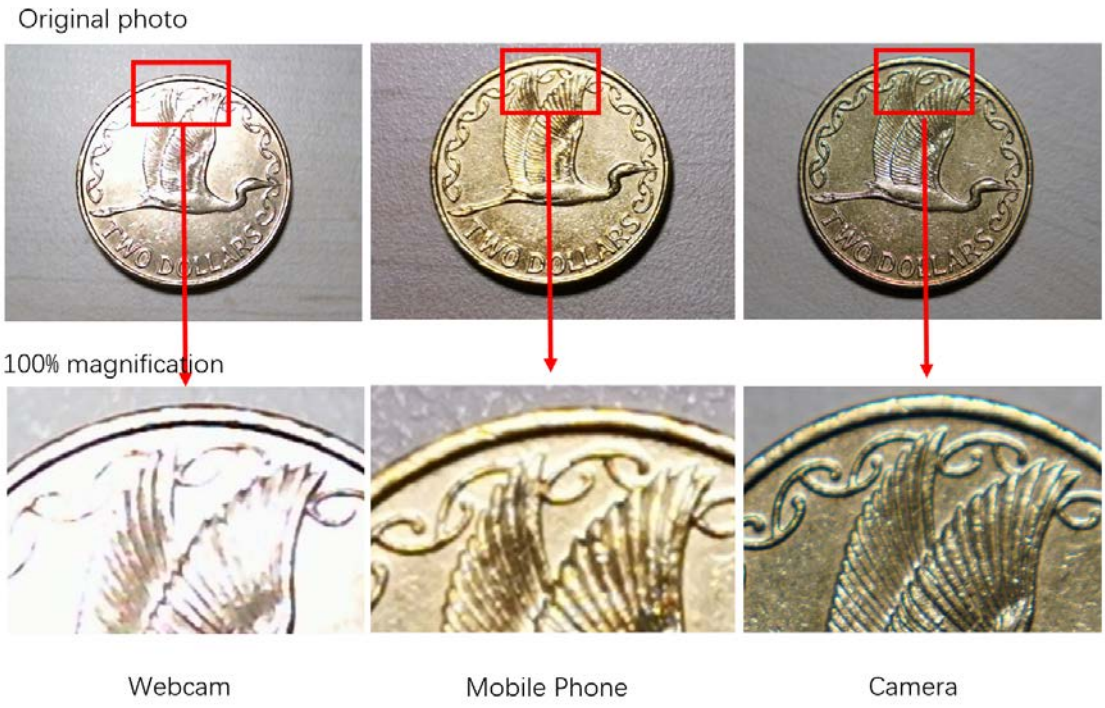


Figure 4.4: Comparison of different device resolutions

A clearer image of the data source can first bring more details, which also means that more detailed features can be extracted from it. On the other hand, a clear image can be expanded by rotating, blocking, blurring, changing the lightness and darkness. Larger data sets can also improve some accuracy.

4.3 Comparison of Training Networks

In the second part of the experiment, the three different neural networks Faster R-CNN, YOLO and SSD. Faster R-CNN have higher accuracy. Compared to Faster R-CNN, YOLO and SSD are one-shot detection models, which have a faster detection speed. Among them, the quickest recognition speed of YOLO can reach 45FPS, the rate of SSD can reach 59FPS (W. Liu et al., 2016). The shooting equipment in life is usually 30FPS, which means that YOLO and SSD can almost meet the daily real-time recognition effect.

Among them, the main disadvantage of YOLO is that it only supports 448x448 resolution. At the same time, the output feature map of the last layer of YOLO is 7x7 fixed size. Therefore, though YOLO has a high recognition speed, it is not suitable for the recognition of small objects (Noman, Stankovic, & Tawfik, 2019).

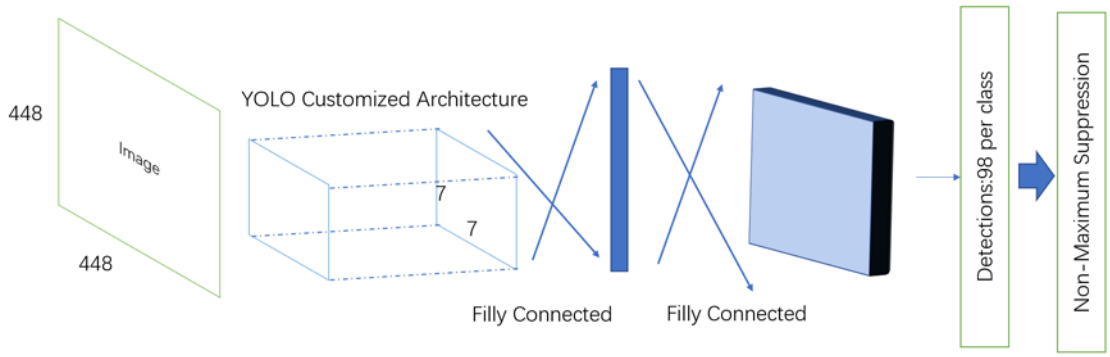


Figure 4.5: YOLO Structure

The defect of YOLO is improved on the SSD. If the detected object is less than 7x7 on the screen, it is difficult to be recognized. SSD takes into consideration of the detection range of objects of various sizes. Therefore, the real-time small object recognition SSD has the highest accuracy meanwhile ensures the recognition speed (Ning, Zhou, Song, & Tang, 2017).

Without considering the recognition speed, Faster R-CNN has the highest accuracy in recognition of small objects, the recognition speed of Faster R-CNN averages 0.2 pictures per second. There is a massive gap between YOLO and SSD (Carlet & Abayowa, 2017).

By dividing the slow-motion video taken by the camera into pictures, they used Faster R-CNN to be verified. In the experiment, the coins rotating on the table were photographed, showing that the single recognition rate was higher than before. But the highest recognition rate is LSTM combined with CNN.

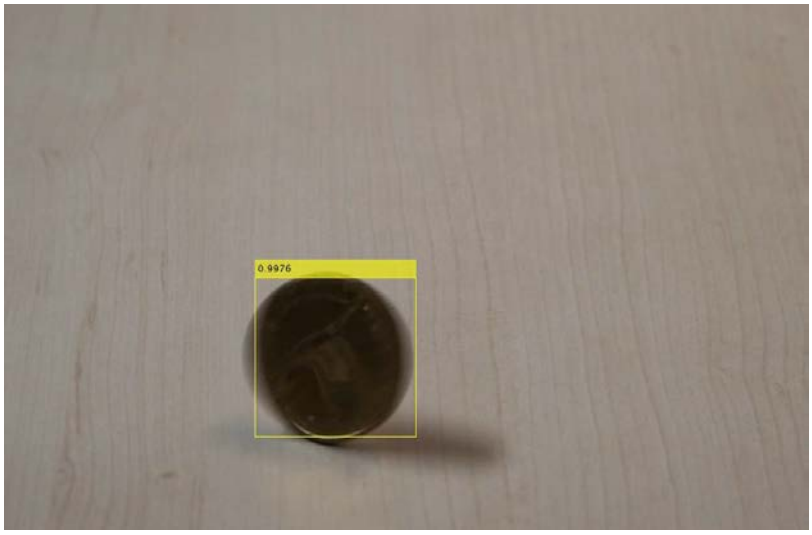


Figure 4.6: Rotating coin recognition

4.4 Limitations of the Research

Although in this paper we implemented the recognition of fast-moving coin, there are still many limitations. These restrictions need to be improved in the future. The restrictions include:

(1) We verify that there is only one fast-moving object in the video. If there are multiple objects, the accuracy may decrease further.

(2) One of the difficulties in shooting slow-motion videos is the difficulty of focusing. The video is fixed by the distance between the shooting device and the object to ensure a clear video.

(3) The structure of LTSM+CNN uses the default GoogLeNet network. It is difficult to replace the network in MATLAB. Using other platforms may result in better results.

(4) Coin recognition has certain limitations. Coins can only be identified by a pattern on one side. The back of the coin has the same pattern of Queen Elizabeth's head. It cannot be recognized effectively on the side of the coin's avatar pattern.

Figure 4.7: Same image on the back of the coin

Chapter 5 Conclusion and Future Work

This chapter will summarize this project. We will comprehensively summarize the identification of small objects and the methods we used in this project. At the end of this chapter, we will point out our future work.

5.1 Conclusion

It is proved through our experiments that the use of deep learning can realize the rapid recognition of coin. Convolutional neural networks are also very suitable for image classification. In recent years, the accuracy of image classification has been gradually improved, but the accuracy of small object recognition is still not high. The combination of deep learning and slow shooting can realize the recognition of fast-moving coins. On this basis, joining the LSTM network can greatly improve the recognition rate.

By using existing networks, fast-moving coin can be quickly identified. For example, Faster R-CNN, YOLO and SSD can quickly identify small objects, but the accuracy still needs to be improved. At the same time, it also proves that the quality of the images has a significant influence on accuracy. High-quality images can often extract more feature details. In the one-stage detection model, compared with SSD and YOLO, SSD is more suitable for small object recognition. Because SSD can identify any size of items in the picture, YOLO cannot accurately recognize when the object features are less than 7x7.

The combination of shooting equipment and deep learning can also achieve high-speed recognition of small objects. In the experiment, the function of slow-motion video shooting using the mobile phone camera can obtain the rapid recognition of small objects. By comparing the pictures of normal speed, we find that the speed that human eyes cannot see at this time. But through deep learning, there is a possibility of giving accurate recognition. If a professional high-speed camera is used, the accuracy of recognition may be further improved.

Finally, it is verified that a suitable network model can achieve higher accuracy. Through the LSTM set CNN model, we can get about 95%’s accuracy. At the speed of this recognition rate test, moving coins can no longer be recognized by our human eyes.

Using deep learning can identify fast-moving coins, which also means that we can use transfer learning to recognize other visual objects. We can capture and identify fast-

moving objects through a camera by a wide range of uses—for example, fast-moving car recognition or face recognition. Yann LeCun also put forward the method of using predictive learning instead of unsupervised learning. He pointed out that video prediction will have more development, for example, in the field of smart cars, the position of the vehicle can be predicted in the next second to predict in advance collision. The vehicle position in advance can improve the accuracy of car identification (Carlet & Abayowa, 2017).

5.2 Future Work

Recognition of small objects has many applications, such as security inspection in financial places, recognition of faces on highways, recognition of distant road signs or cars in automatic driving, etc. Our future work mainly includes:

(1) Future improvements are first of all further improvements inaccuracy. Only with higher accuracy, we can the high-speed recognition of small objects have more application scenarios.

(2) We will consider using a high-speed camera to capture more details to improve accuracy further.

(3) We will optimize the network structure, according to experiments, LSTM combined with CNN can improve the accuracy of recognition. The experiment uses the GoogLeNet that comes with MATLAB. If we use LSTM in combination with SSD or Faster R-CNN, it may further improve the recognition rate.

(4) We improve the experimental data set to make the recognition of small objects closer to life. In this research project, coins are mainly selected as our experimental objects, and the innovative environment is single. It is more practical to identify fast-moving objects in complex environments.

References

Agarap, A. F. (2018). Deep learning using rectified linear units (ReLu). *arXiv:1803.08375*.

Alom, M. Z., Taha, T. M., Yakopcic, C., Westberg, S., Sidike, P., Nasrin, M. S., Asari, V. K. (2018). The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv:1803.01164*.

Ballester, P., & Araujo, R. M. (2016). On the performance of GoogLeNet and AlexNet applied to sketches. The Thirtieth AAAI Conference on Artificial Intelligence

Carlet, J., & Abayowa, B. (2017). Fast vehicle detection in aerial imagery. *arXiv:1709.08666*.

Chavan, S. S., Fernandes, C., Dumane, P. R., & Varma, S. L. (2020). Design and Implementation of Automatic Coin Dispensing Machine. In *ICCCE 2019* (pp. 379-385): Springer.

Chen, L.-C., Papandreou, G., Kokkinos, I., Murphy, K., & Yuille, A. L. (2014). Semantic image segmentation with deep convolutional nets and fully connected CRFs. *arXiv:1412.7062*.

Chesher, C. (2012). *Between image and information: The iPhone camera in the history of photography*: na.

Ciaburro, G., & Venkateswaran, B. (2017). *Neural Networks with R: Smart models using CNN, RNN, deep learning, and artificial intelligence principles*: Packt Publishing Ltd.

Dabic, S. (2013). Coin identification method and apparatus: Google Patents.

Dahl, G. E., Sainath, T. N., & Hinton, G. E. (2013). Improving deep neural networks for LVCSR using rectified linear units and dropout. IEEE International Conference on Acoustics, Speech and Signal Processing

Davies, E. R. (2004). *Machine Vision: Theory, Algorithms, Practicalities*. Elsevier.

El-Sawy, A., Hazem, E.-B., & Loey, M. (2016). CNN for handwritten Arabic digits

recognition based on LeNet-5*Springer.* The International Conference on Advanced Intelligent Systems and Informatics

Filonenko, A., & Jo, K.-H. (2016). Unattended object identification for intelligent surveillance systems using sequence of dual background difference. *IEEE Transactions on Industrial Informatics, 12*(6), 2247-2255.

Fossati, A., Gall, J., Grabner, H., Ren, X., & Konolige, K. (2012). *Consumer depth cameras for computer vision: research topics and applications*: Springer Science & Business Media.

Genesereth, M. R., & Nilsson, N. J. (2012). *Logical foundations of artificial intelligence*: Morgan Kaufmann.

Gensler, A., Henze, J., Sick, B., & Raabe, N. (2016). Deep Learning for solar power forecasting—An approach using AutoEncoder and LSTM Neural Networks. IEEE International Conference on Systems, Man, and Cybernetics (SMC)

Gers, F. A., Schmidhuber, J., & Cummins, F. (1999). Learning to forget: Continual prediction with LSTM.

Girshick, R. (2015). Fast R-CNN. IEEE International Conference on Computer Vision

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. IEEE Conference on Computer Vision and Pattern Recognition

Giusti, A., Cireşan, D. C., Masci, J., Gambardella, L. M., & Schmidhuber, J. (2013). Fast image scanning with deep max-pooling convolutional neural networks. IEEE International Conference on Image Processing

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*: MIT press.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2016). LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems, 28*(10), 2222-2232.

Guo, K., Sui, L., Qiu, J., Yao, S., Han, S., Wang, Y., & Yang, H. (2016). Angel-eye: A complete design flow for mapping cnn onto customized hardware. IEEE Computer Society Annual Symposium on VLSI (ISVLSI)

Gysel, P., Motamedi, M., & Ghiasi, S. (2016). Hardware-oriented approximation of convolutional neural networks. *arXiv:1604.03168*.

Higham, D. J., & Higham, N. J. (2016). *MATLAB guide*: SIAM.

Hossfeld, M., Adameck, M., & Eich, M. (2003). Machine vision detects conterfeit coins. *Laser Focus World, 39*(6), 99-103.

Hsu, S.-H., & Huang, C.-L. (2001). Road sign detection and recognition using matching pursuit method. *Image and Vision Computing, 19*(3), 119-129.

Huang, J., Rathod, V., Sun, C., Zhu, M., Korattikara, A., Fathi, A., . . . Guadarrama, S. (2017). Speed/accuracy trade-offs for modern convolutional object detectors. IEEE Conference on Computer Vision and Pattern Recognition

Ide, C., Jackson, J., & McGonnigle, G. (2006). Environmentally aware, intelligent surveillance device: Google Patents.

Jain, A., Zamir, A. R., Savarese, S., & Saxena, A. (2016). Structural-RNN: Deep learning on spatio-temporal graphs. IEEE Conference on Computer Vision and Pattern Recognition

Jain, R., Kasturi, R., & Schunck, B. G. (1995). *Machine Vision* (Vol. 5): McGraw-Hill New York.

Jiang, H., & Learned-Miller, E. (2017). Face detection with the faster R-CNN. IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)

Joshi, K. D., Surgenor, B. W., & Chauhan, V. D. (2016). Analysis of methods for the recognition of Indian coins: A challenging application of machine vision to automated inspection. International Conference on Mechatronics and Machine Vision in Practice (M2VIP)

Kågström, J. (2005). Improving naive bayesian spam filtering. *Mid Sweden University, Sweden*.

Kawakita, M., Iizuka, K., Aida, T., Kikuchi, H., Fujikake, H., Yonai, J., & Takizawa, K. (2000). Axi-Vision Camera (real-time distance-mapping camera). *Applied Optics, 39*(22), 3931-3939.

Khashman, A., Sekeroglu, B., & Dimililer, K. (2006a). ICIS: A novel coin identification system. In *Intelligent Computing in Signal Processing and Pattern Recognition* (pp. 913-918): Springer.

Khashman, A., Sekeroglu, B., & Dimililer, K. (2006b). Intelligent coin identification system. IEEE International Symposium on Intelligent Control

Kim, H., Kim, K., & Kim, H. (2016). Data-driven scene parsing method for recognizing construction site objects in the whole image. *Automation in Construction, 71*, 271-282.

Kim, J., Zeng, H., Ghadiyaram, D., Lee, S., Zhang, L., & Bovik, A. C. (2017). Deep convolutional neural models for picture-quality prediction: Challenges and solutions to data-driven image quality assessment. *IEEE Signal processing magazine, 34*(6), 130-141.

Kirchner, H., & Thorpe, S. J. (2006). Ultra-rapid object detection with saccadic eye movements: Visual processing speed revisited. *Vision Research, 46*(11), 1762-1776.

Konoplich, G. V., Putin, E. O., & Filchenkov, A. A. (2016). Application of deep learning to the problem of vehicle detection in UAV images. IEEE International Conference on Soft Computing and Measurements (SCM)

Kwak, N. J., & Song, T.-S. (2013). Human action classification and unusual action recognition algorithm for intelligent surveillance system. In *IT Convergence and Security 2012* (pp. 797-804): Springer.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature, 521*(7553), 436-444.

Liu, L., Shen, C., & van den Hengel, A. (2015). The treasure beneath convolutional layers: Cross-convolutional-layer pooling for image classification. IEEE Conference on Computer Vision and Pattern Recognition

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., & Berg, A. C. (2016). SSD: Single shot multibox detector. The European Conference on Computer Vision.

McAllister, R., Gal, Y., Kendall, A., Van Der Wilk, M., Shah, A., Cipolla, R., & Weller, A. (2017). Concrete problems for autonomous vehicle safety: Advantages of bayesian deep learning*International Joint Conferences on Artificial Intelligence, Inc.*

Mennel, L., Symonowicz, J., Wachter, S., Polyushkin, D. K., Molina-Mendoza, A. J., & Mueller, T. (2020). Ultrafast machine vision with 2D material neural network image sensors. *Nature, 579*(7797), 62-66.

Narang, P., Agarwal, A., & Sanu, A. S. (2015). Detecting subtle intraocular movements: Enhanced frames per second recording (slow motion) using smartphones. *Journal of Cataract & Refractive Surgery, 41*(6), 1321-1323.

Nguyen, K., Fookes, C., Ross, A., & Sridharan, S. (2017). Iris recognition with off-the-shelf CNN features: A deep learning perspective. *IEEE Access, 6*, 18848-18855.

Nilsson, N. J. (2014). *Principles of artificial intelligence*: Morgan Kaufmann.

Ning, C., Zhou, H., Song, Y., & Tang, J. (2017). Inception single shot multibox detector for object detection. IEEE International Conference on Multimedia & Expo Workshops (ICMEW)

Noman, M., Stankovic, V., & Tawfik, A. (2019). Object detection techniques: Overview and performance comparison. IEEE International Symposium on Signal Processing and Information Technology

Ouyang, W., & Wang, X. (2013). Joint deep learning for pedestrian detection. IEEE International Conference on Computer Vision

Pathak, A. R., Pandey, M., & Rautaray, S. (2018). Application of deep learning for object detection. *Procedia computer science*, *132*, 1706-1717.

Pérez-Hernández, F., Tabik, S., Lamas, A., Olmos, R., Fujita, H., & Herrera, F. (2020). Object detection binary classifiers methodology based on deep learning to identify small objects handled similarly: Application in video surveillance. *Knowledge-Based Systems*, 105590.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. The Advances in Neural Information Processing Systems

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Bernstein, M. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, *115*(3), 211-252.

Russell, S., & Norvig, P. (2002). Artificial Intelligence: A Modern Approach. MIT Press.

Sainath, T. N., Vinyals, O., Senior, A., & Sak, H. (2015). Convolutional, long short-term memory, fully connected deep neural networks. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)

Schlag, I., & Arandjelovic, O. (2017). Ancient Roman coin recognition in the wild using deep learning based recognition of artistically depicted face profiles. IEEE International Conference on Computer Vision Workshops

Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*, 85-117.

Segrave, K. (2015). *Vending Machines: An American Social History.* McFarland.

Shi, B., Bai, X., & Yao, C. (2016). An end-to-end trainable neural network for image-based sequence recognition and its application to scene text recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *39*(11), 2298-2304.

Sonka, M., Hlavac, V., & Boyle, R. (2014). *Image processing, analysis, and machine*

vision: Cengage Learning.

Sun, B., & Velastin, S. (2003). Fusing visual and audio information in a distributed intelligent surveillance system for public transport systems. *Acta Autom. Sin, 20*(3), 393-407.

Sun, Y., Liang, D., Wang, X., & Tang, X. (2015). DeepID3: Face recognition with very deep neural networks. *arXiv:1502.00873*.

Tajane, A., Patil, J., Shahane, A., Dhulekar, P., Gandhe, S., & Phade, G. (2018). Deep Learning Based Indian Currency Coin Recognition. International Conference On Advances in Communication and Computing Technology (ICACCT)

Targ, S., Almeida, D., & Lyman, K. (2016). Resnet in resnet: Generalizing residual architectures. *arXiv:1603.08029*.

Vedaldi, A., & Lenc, K. (2015). Matconvnet: Convolutional neural networks for MATLAB. ACM international conference on Multimedia

Wang, J., Yang, Y., Mao, J., Huang, Z., Huang, C., & Xu, W. (2016). CNN-RNN: A unified framework for multi-label image classification. IEEE Conference on Computer Vision and Pattern Recognition

Wells, B., & McGlone, J. T. (2002). Gaming device identification method and apparatus: Google Patents.

Welsh, B. C., & Farrington, D. P. (2009). *Making public places safer: Surveillance and crime prevention*: Oxford University Press.

Wu, J., Yu, Y., Huang, C., & Yu, K. (2015). Deep multiple instance learning for image classification and auto-annotation. IEEE Conference on Computer Vision and Pattern Recognition

Yang, F., Choi, W., & Lin, Y. (2016). Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers. IEEE Conference on Computer Vision and Pattern Recognition

Ye, J., Ni, J., & Yi, Y. (2017). Deep learning hierarchical representations for image steganalysis. *IEEE Transactions on Information Forensics and Security, 12*(11), 2545-2557.

Yu, D., Wang, H., Chen, P., & Wei, Z. (2014). Mixed pooling for convolutional neural networks*Springer.* International Conference on Rough Sets and Knowledge Technology