# 2.5 Introduction of Deep Learning

Deep learning is part of broader family of machine learning methods based on artificial neural networks with representation learning [1]. Deep learning architectures can be divided into Feed-Forward-Net and Recurrent-Neural-Network. Feed-Forward-Net is an artificial neural network, which consists of input layer, hidden layers and output layer. The information moves in only on direction, from the input node, through the hidden nodes to the output nodes[2]. The basic difference of the Recurrent-Neural-Network from Feed-Forward-Net is that RNN can use intern state (memory) to process the variable length of inputs. The representative architecture is long short-term memory.

In our project, the features extracted from various paper, such as mass, dark gray proportion, are the values with fixed length, not a sequence of inputs. The outputs are the number of the paper class. According to these characters, we selected the architectures, such as Convolutional Neural Network and some pre trained Network, which belong to the Feed-Forward-Network.

## 2.5.1 Convolutional Neural Network

Convolutional neural network is one of the most popular algorithms for deep learning. Most commonly applied to analyzing visual imagery [3]. Figure 1 shows the basic architecture of CNN. Like the most other Neural Network, a CNN is composed of an input layer and an output layer and many hidden layer in between. The typical character of CNN is the mathematical operation called convolution. Convolution is specialized kind of linear operation and it can create feature from input images. In CNN this operation is executed by multiplication of matrix between input image and a series of convolutional filters. The rectified linear unit (ReLU) can transfer the output of each convolutional layer and allow for nonlinearity. The final part in one feature learning is pooling. It can reduce the number of parameters, which should be learned by Network and combine the feature with other pixels together. Normally, these three operation are repeated over tens or hundreds layers in order to detect different features.

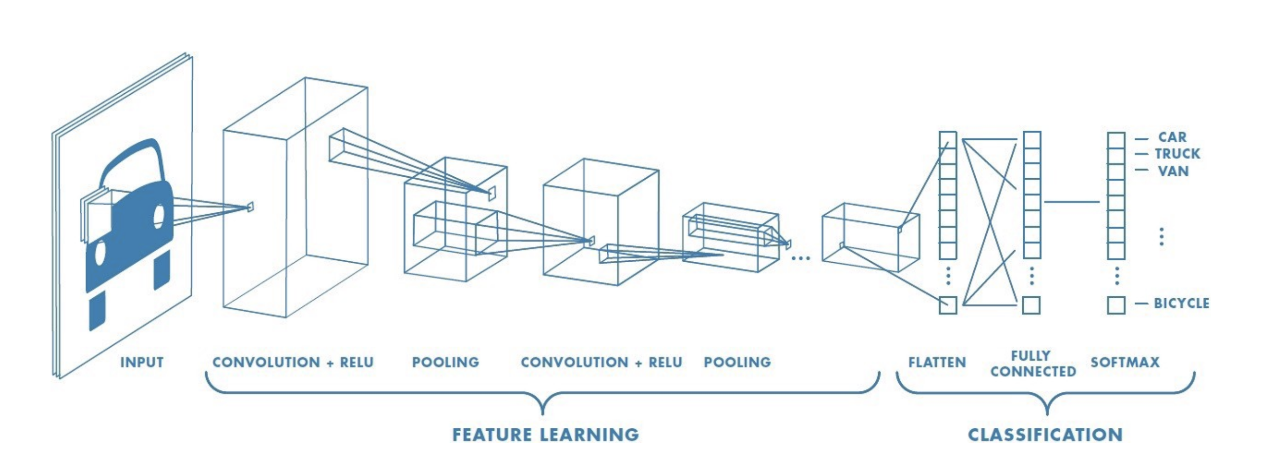


Fig.1 the basic architecture of CNN

After feature detection, the architecture shifts to fully connected layer, which is similar to the traditional multi-layer perceptron neural network (MLP). If the problem is classification, there is also a softmax layer to calculate the probability of the input in each class

In principle CNN is applied in image classification, which belongs to 2-D problem. In our project, the CNN is also planned to apply in NIR spectrum, in order to predict the content of Kaolin. However, the NIR spectrum is a 1-D spectrum. It means that, directly using the CNN is not possible. Chenhao Cui [4]from University College London has proposed a new method about implementation of traditional CNN in spectroscopic analysis by changing the input size of Input layer and build a new series of convolution filters which are suitable for 1-D input data. Malek, Melgani and Bazi [5] also explored 1D-CNN for spectroscopic regressions. In figure 2 shows the converted architecture of a CNN, which can be applied in NIR spectrum. The concrete process included the feature extraction and preprocessing will be introduced in chapter 3.

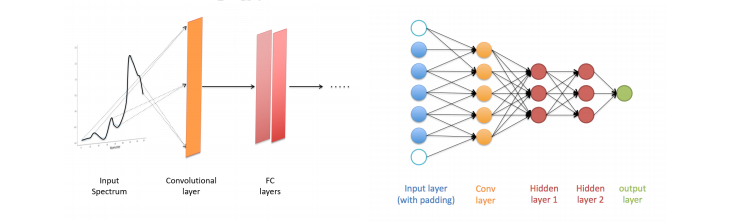


Fig.2 CNN architecture applied NIR spectroscopic analysis [4]

The most significant advantage of CNN compared to traditional machine learning is that it is not necessary for CNN to extract the feature manually. The convolutional filter can detect the feature from computer vision. It can create more reliable features than them from the algorithm written by human.

## 2.5.2 Transfer learning

Although data mining and machine learning technologies have already success in many knowledge engineering areas including classification, regression and clustering[6], for deep learning network, it still needs a large amount of data for training and testing, if a neural network wants to be built for a specify application. Nowadays, in some applications (such as in paper classification), data collection is still a troublesome and expensive work. In such case, Transfer learning is a desirable choice. Instead of building a new neural network, transfer learning can achieve a relative precise result by fine-tuning based on a pre trained network. Nowadays there are various pre trained architectures of neural network. Most of them can be taken in image classification problem.

Since there are totally 3800 objects can be used for paper classification, it is impossible to build a new neural network as classifier. Therefore, it is more property to build a classification neural network by transfer learning. In this chapter, four kinds of typical architecture, which are used in this project will be introduced. They are Alexnet, GoogLenet, VGG-net and ResNet. All the following introduction of neural network are referred to the architecture in Matlab® 2020a

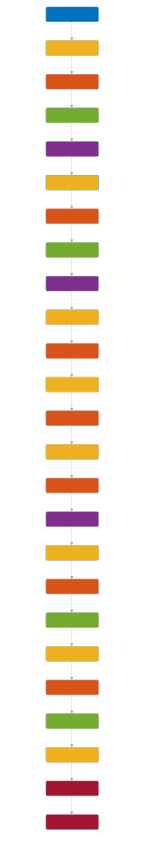
**AlexNet** is a convolutional neural network that is 8 layers deep. Figure 3 shows the overview architecture of AlexNet. It contains 8 leraned layers – five convolutional and three fully-connected.

Fig. 3 Architecture of AlexNet

The output of the last fully connected layer is fed to a 1000-way softmax which produces a distribution over the 1000 class labels [7]. The Outstanding of AlexNet compared to the other Neural Network is that it enlarge the learning capacity of CNNs by increasing the depth and breath of a CNNs. Addtionaly, AlexNet uses ReLU Nonlinearity and Local Response Normalization as depicted in Figure 4, in order to reduce the overfitting problem during the training. Therefore, Alexnet has better robustness and compared to the other popular pre trained network such as GoogLeNet, those are also used in this project, it has also a simpler architecture.

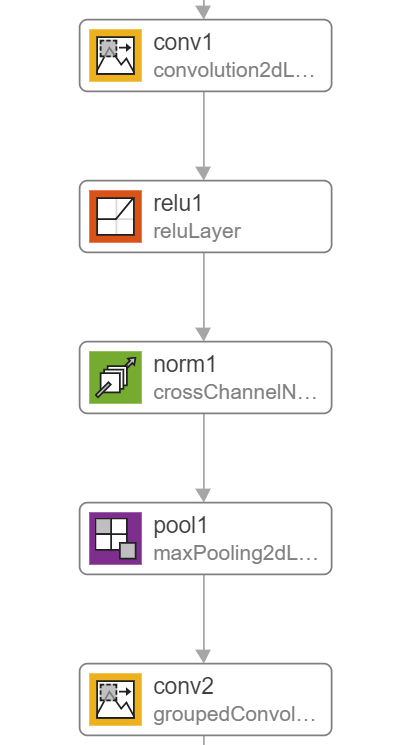


Fig.4 the first feature detection layer in AlexNet

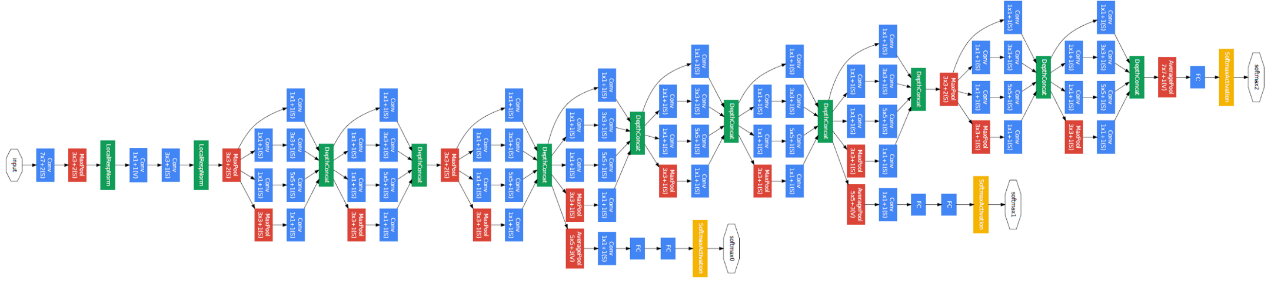
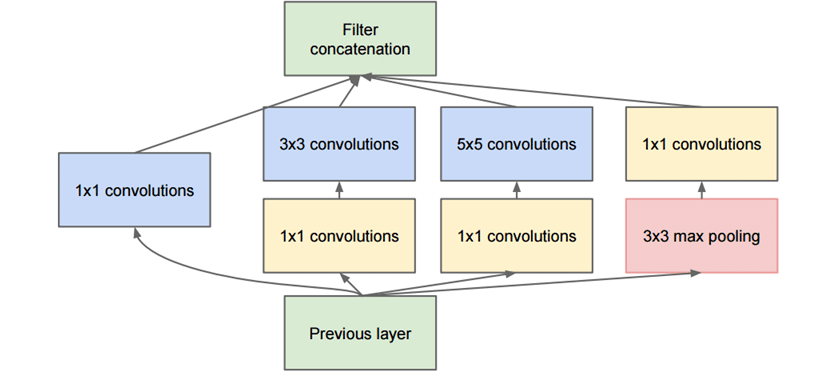
**GoogLeNet** is another convolutional neural network with 22 layers deep. Its architecture is called inception. Figure 5 shows the overview of the GoogLeNet.

Fig 5 Architecture of GoogLeNet

The main idea of the Inception module (Fig. 6) is that it stacks 1x1 convolution for compute reductions before the expensive 3 x 3 and 5 x 5 convolutions [8]. The Function is similar to the PCA, which maintains the height and width of feature map but reduces the depth. The advantage of it is learning efficiency. Besides, compared to Alexnet, GoogLeNet reduced the number of parameters from 60 million to 4 million.

Fig. 6 Inception module

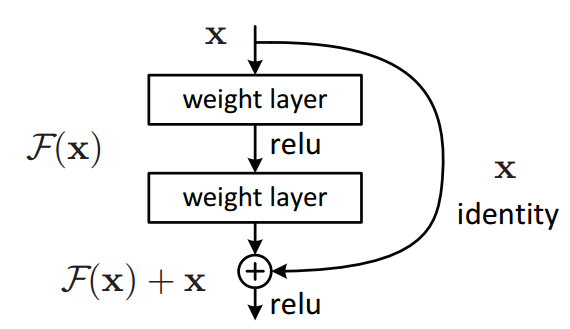
**ResNet** is an artificial neural network and uses residual learning function instead of unreferenced learning function. The hallmark of ResNet is that it can solve the vanish gradient problem in deep layer. The principle of the Residual learning is described in Figure 7. Instead of underlying mapping H(x), Residual mapping F(x) is as Input to fit each layer. In this time, the original feature map H(x) is F(x) + x. This formulation can be realized by feedforward neural network with “shortcut connection” [9].

Fig. 7 Residual learning: a building block

As mentioned before, in AlexNet there are many 11 x 11 and 5 x 5 convolutions. They cause much kernel parameters and decrease the calculation performance. However, a large convolution kernel can enlarge the receptive field and combine more information from image, which is benefited to increase quality of the generated feature maps. GoogLeNet solves this problem by stacking a 1 x 1 convolution to reduce the depth of feature map.

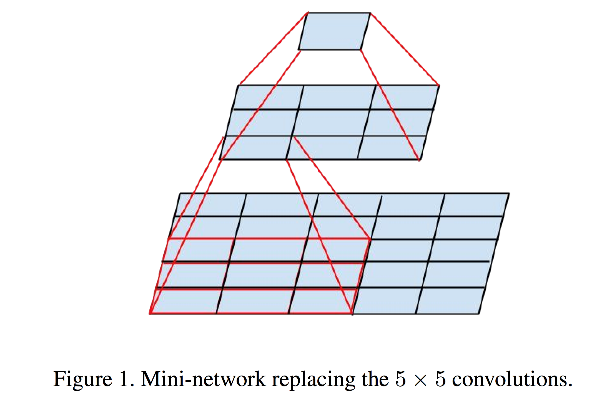
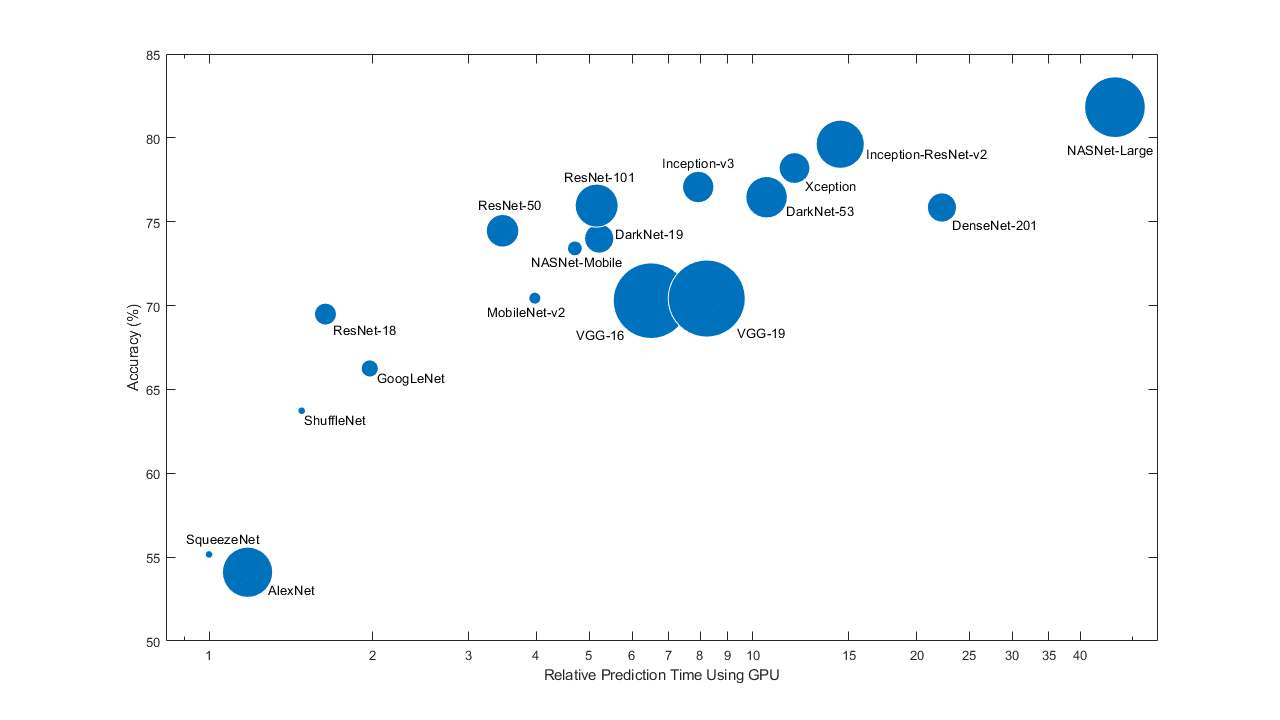
**VGGNet** is also a convolutional neural network, which can also solve this dilemma from another perspective. VGGNet use very small 3 x 3 receptive fields through the whole net [10]. As shown in Figure 8, a 5 x 5 convolution layer can replaced by a stack of two 3 x 3 convolution layers. It confirms that a stack of two 3 x 3 convolution has more effective receptive field than a 5 x 5 convolutions. By analogy three such layers can achieve similar function of a 7 x 7 convolution layer.

Fig. 10 Mini-network replacing the 5 x 5 convolutions

Figure 11 shows the performance of different tranning model according to the accuracy and utilization of computer resource (specially in GPU). The most important characteristics of a model

Fig. 10 Characteristic of different pre trained model

are accuracy, size and speed. It can be clearly concluded that, the increase of accuracy in model comes with sacrificing calculation time. Therefore, choosing a network is generally a tradeoff between these characteristics. Table 1 shows the summary of pre trained model, which are used in this project. According to this table, it is still hard to certain which model is the most property. The results of using different model will be introduced in chapter 4.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Network | Depth | Parameter | Size | Characteristic |
| AlexNet | 8 | 61 Millions | 227 MB | Parallel Working on two GPU |
| SqueezNet | 18 | 1.24 Millions | 4.6 MB |  |
| GoogLeNet | 22 | 7.0 Millions | 27 MB | Inception module |
| ResNet 18 | 18 | 11.7 Millions | 44 MB | Residual learning function |
| VGG16 | 16 | 138 Millions | 515 MB | Stacked convolutions |

Table 1 Overview of different model

# 3.1 Feature from NIR Spectrum

## 3.1.1 Hyperspectral imaging

The hyperspectral camera used in our project is a FX17 from SPECIM Imaging Ltd., which can provide totally 224 wavelength over a range from 900nm to 1700nm [11]. This camera can scan 640 pixels in line and provides spatial resolution of 0.94 nm. The camera can take a picture in every trigger and takes 70 line-scans. All the information is stored in a hypercube, whose length and width represent location and depth represents wavelength.

Since the camera can only capture the intensity of light, it still needs to be corrected, in order to get the reluctant value. The dark background can absorb all the light and then even no photon can reflect back to sensor. The intensity is as *B* recorded. The Intensity of white background can be recorded as *W* in the same way. Using this two parameters can the actual reflectance R of Object with intensity of object S be calculated in following equation 3-1

3-1

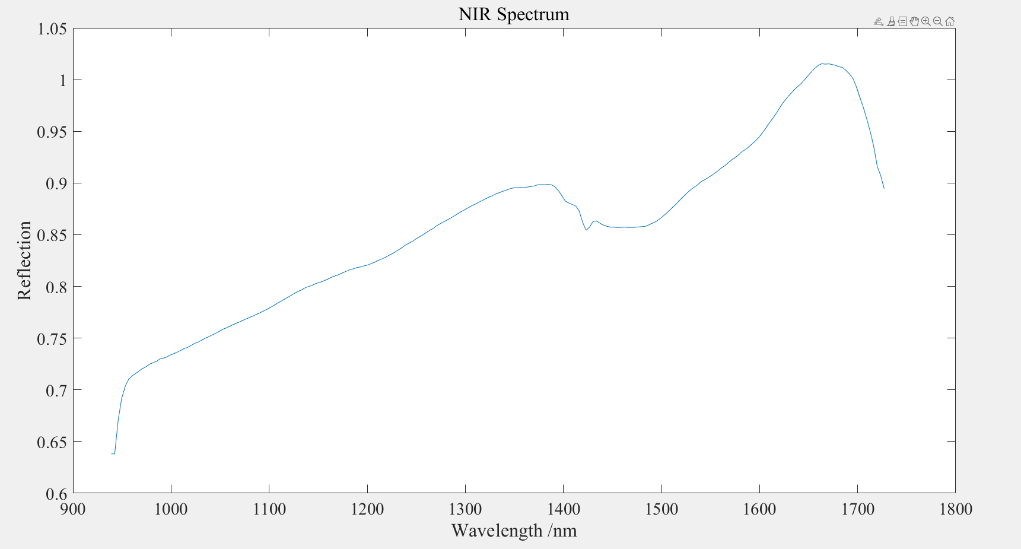
In order to decide the range of interest (ROI), the value of 100th channel is checked to distinguish the conveyor belt and object. Those pixel, where the intensity is very low are labelled to black. Other pixels will labelled to white. In order to avoid the influence of object’s bounding, the pixel on the bounding of the object are also labelled to black. Only the white labelled area is used for the feature project. Since the size of original hypercube is too large for computer resource to store it, only the average value of the every channel is recorded and the whole 224 values are regarded as the final NIR spectrum for an object. A typical Spectrum is shown in Figure 3-1.

Fig. 3-1 NIR Spectrum after process of hyperspectral image

It can be clearly detected that there is a spike between the 1400nm and 1450nm. This spike is regarded as relevant to Kaolin content. The content of Kaolin varies according to the different paper class, especially in grey carton and in white carton. The distinguish of these two class is always hard for computer version.

## 3.1.2 NIR Spectrum pre processing

**Select Region of Interest**

As the Figure 3-1 described, the spectrum a rapid upward trend at the beginning. In this wavelength range the spectrum contains too much noise and these values is also useless for the classification and regression. The spectrum at the end of mess range is also unstable. Therefore, a range of wavelength should be decided, in order to obtain a data set with good quality. In our project, the region of interest (ROI) is from 974 nm to 1680 nm. The cropped spectrum is as shown in Figure

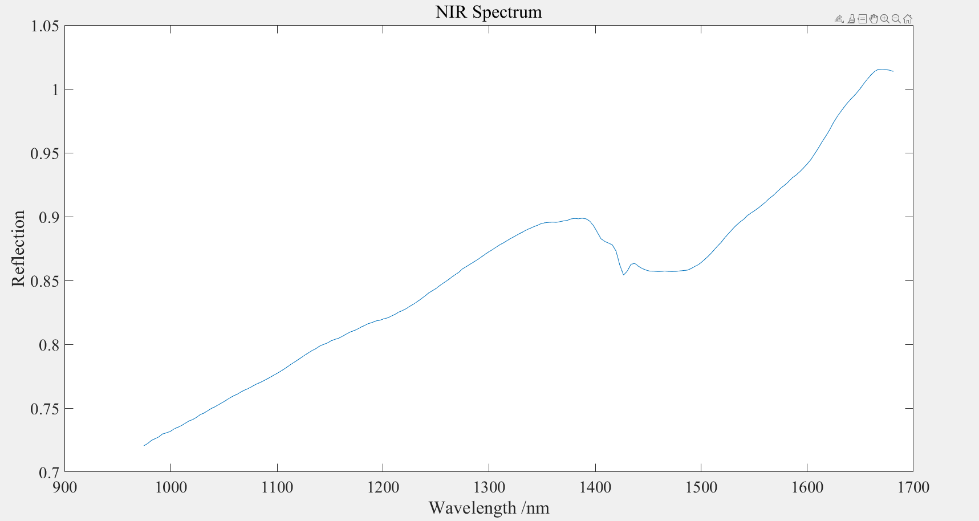
3-2.

Fig. 3-2 NIR Spectrum after ROI process

**Savitzky Golay Filter and Median Filter**

As mentioned in section 3.1.1, the spike appearing in NIR spectrum is a very important feature for classification. Savitzky Golay Filter can increase the spectrum resolution and remove the background. In general, Savitzky Golay Filter are often used with second derivative. Because it can not only remove the constant and linear background, but also be easier to interpretation [12]. However, the drawback of the differentiation is that it also amplifies the noise. Therefore, in our project, before Savitzky Golay Filter, a Median Filter is used to remove the noise. Besides, the the moving window in Savitzky Golay Filter can also smooth the spectrum. In Matlab, the window size and the order of polynomial order in Savitzky Golay Filter function should be carefully selected. If the window size is too big, it will filter out the relevant spike. In our project the window size is 11 and the order of polynomial is 3. For Kaolin content regression the first and second derivatives and for paper classification only the first derivative is used to process the NIR spectrum. Figure 3-3 shows the result with Median filter and SG filter in second order.

(此处有图)

**Standard Normal Variate**

Standard Normal Variate is a common method to handle with NIR spectrum. SNV can remove multiplicative interferences of scatter and particle size [12]. The spectrum after SNV is centered to have a mean value 0 and scaled to have a standard deviation 1. The equation is depicted as 3-2

3-2

μrepresents the mean value of variable x and σis the standard deviate of variable x. In our project the number of row represents the amount of NIR spectrum and every column contains the value of corresponding wavelength in a row. Therefore, SNV is executed along the row dimension. Especially, for Classification it is also necessary to execute SNV method along every column in order to make the feature matrix more property for machine learning.

(此处有图)

**Offset Correction**

Offset Correction is an alternative method to SNV in our project. It is not a popular method and executed only when it causes singular value by using SNV method. In our project, these phenomenon occurs in linear discriminant analysis method that will be introduced in section 3.1.3.

Offset correction is applied to correct for a parallel baseline shift (Figure 3-5). The principle of this method is arbitrary chosen value and Subtraction from each spectrum independently. In our project, the mean reflection of first five value in spectrum is calculated and used to correct.

3-3

Where is the corrected spectrum, is the original spectrum and is the mean value of the first five wavelengths. The processed result is shown in Figure 3-6.

(此处有图)

## 3.1.4 Feature Extraction for paper classification

Some dimension reduction methods have been introduced in chapter 2. In this section, the practical application of these methods in feature extraction for paper classification will be introduced. Although different dimension reduction methods are based on various principles, they follow the same working flow as Figure 3-7

(此处有图)

In pre-processing, the commend pre-processing like Select of ROI, SNV and SG filter are used in PLS, PCA and Autoencoder methods. Especially in our project, if the SNV along the column direction is executed in LDA method, it will cause singular value in extracted feature matrix. These singular value cannot be regarded as feature in machine learning application. Therefore, alternative offset correction method is used here instead of SNV along each wavelength. The spectrum after pre-process is described in Figure 3-8

(此处有图)

Afterwards, the number of the components should be decided. The criteria of component selection is different based on the principle of the methods. In PLS the RMSE between the response calculated from first n components and the original label can suggests the optimal amounts of components. This value means that how completely the reduced predictors describes the observation. Usually the optimal point in the graphic 3-9 is selected, when before this point, the mean square error declines rapidly but afterwards slowly. This point is also regarded as “Elbow Point”

(此处有图)