MLP Coursework 3 : Project Interim Report

Team G22 : s1717961 s1742237

Abstract

In reality, identifying or comparing art works has become a popular topic from the point of view of computer vision researchers who have tried to achieve such image classification on the basis of varies increasingly developed machine learning approaches. In particular, deep learning has become more crucial in helping to solve complex classification problems using convolutional network and deep neural network. This paper is trying to compare random two paintings and judge whether these two paintings belong to the same painter in deep systems and convolutional schemes. All these experimental results were obtained through training a similar network in terms of image data sets coming from one of Kaggle competition, "painter-by-numbers". Some hyperparameter setting and approaches that used to discourage overfit are focused to be explored in the baseline system. Shallower convolutinal network and lower learning rate is suitable for selected subset of raw data set.

1. Introduction

1.1. Motivation

The identification of art works has long been regarded as a difficult problem to regardless of professional artists or ordinary art enthusiasts. For example, it is difficult for a connoisseur who equips with resourceful inspection knowledge even in relation to a chemical or physical analysis of materials in order to distinguish the authorities of various paintings, not to mention for people who lack of such experience like common gallery visitors. It is natural to be wondering that whether there are some convenient tools or methods that can be used to classify these paintings with very distinct styles according to different painters, which can be regarded as a computer vision problem. In fact, computer vision has become a powerful tool in solving a wider range of practical questions mainly because it focuses on how to obtain high-level understanding like human visual system from digital images, audios or videos, even though they share a lot in common, like image processing software and it is able to deal with vast amounts of data in images (AIA).

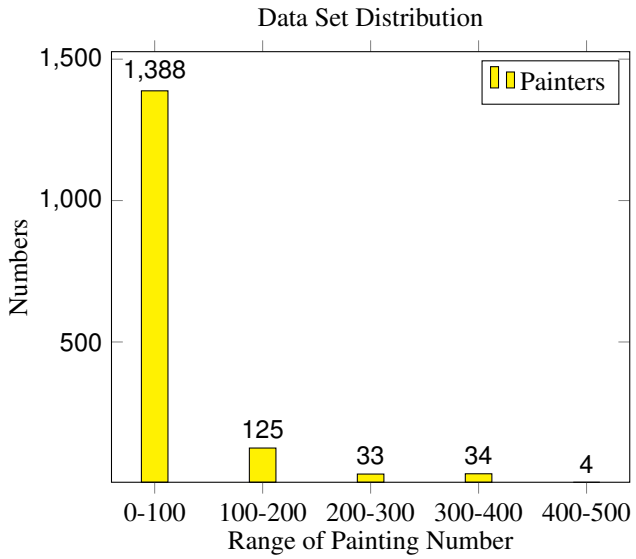
Also, various experiments and methods in the context of image analysis will be shown in the experiment sections of this paper, which further convinces of that computer vision provides more efficient ways in solving difficult problems in practice that people have a difficulty in dealing with. With the rapid development of machine learning, Kaggle opened an interesting competition called Painter by Numbers in the hope that identifying the authorities of two different paintings picked from the data set including 79433 training examples with labels about the artist, title, style, genre and date, and 23817 test images. Obviously, this kaggle competition question with respect to painting classification that is closely related to the application of computer vision using deep learning. What this paper want to do is to explore better approaches or network settings in order to deal with such artistic similarity detection. Thus, all experiments are basically based on this competition's topic and data set, taking advantage of computer vision techniques. To be clear, we try to achieve better performance on such vision-related classification problem through comparing experiment data of different algorithms and models in deep neural network.

1.2. Objective and Research questions

However, it is obvious that many computer vision techniques that is not limited in simulating human visual perception, most importantly, these advanced techniques are expected to realize robust image analysis regarding various of real questions. The main objective of this paper is to achieve higher accuracy with respect to artistic similarity detection through setting up better models and investigating model hyper-parameter choices. Specially, section 3 will introduce a baseline system in detail with reference to the first place's model on the basis of deep convolutional neural network. Interestingly, the proposed deep neural network classify paintings on the basis of authors in comparison with classical classification of digital images is likely to be based on artwork styles. At the same time, some specific research questions will be explored in deep. Firstly, we want to discuss how a couple of hyper-parameter settings like learning rate, batch size and depth of deep neural network that may affect the experiment results. Secondly, the potential impact on model performance regarding using or failing to use a few strategies to discourage over-fitting such as L2 Regularization, Batch Normalization and dropout. Lat but not least, whether there are more suitable network or algorithms that are more likely to obtain satisfying classification validation accuracy.

1.3. Data set and task

Painter by numbers contains a lot of images created by 1584 artists, and the number of each image of per artist is not evenly distributed. All images are divided into two parts.



One is the training part contains 79433 images with label about the artist, title, style, genre and date. The other(test set) contains 23817 images which divided into 13 groups. The number of images in every group is between 1400-2000. It need to notice that the image of an artist may only appear in the training set or in the test set. Plus, there are almost 3.1 billion different pairs of paintings that can be formed using 79433 training examples, which is too large combinations given our current condition. First of all, we select 60 painters randomly from 125 painters (See figure 1, the second mode other than 1388 in range of [100, 200)) whose number of paintings in original data set is from 100 to 200 and picked up 100 pictures for each painters in this subset randomly so as to convert training data into balanced data set. Secondly, we resized the raw data set with different size into the same 64*64 pixel cropped from the center of each painting in order to create the same fixed input size. By doing this we can get an image with smaller size containing the same information. Although this pre-processing method will cause inevitable loss of information, it reduces running cost significantly, especially for limited running time and memory. Similarly, test set has been cropped into the same size and reduced its data size to 1500, one fourth of the new training data size. After this preprocessing phase, images in the training part are divided into training set and validation set in a ratio of 9 to 1 according to painter classes.

Five main sections will consist of the main content of this paper. Firstly, main methods applied in our baseline system will be explain in section 2. In particular, t-SNE dimensionality reduction technique and Siamese network will be discussed in detail. Then section 3 will show that how to build a baseline system. Although the structure of this system is involved with a wider range of choices, section3 is supposed to introduce the first place's network with little changes. Thirdly, conclusion and related outcomes analysis of experiment part are summarized in section 4. In the end of this report, further plan of this large project will be discussed in terms of deeper researches and predicted results and risks.

2. Methodology

In our project, we aim to establish the deep learning model to map the fixed-size inputs to fixed-size output. Commonly, convolutional neural network is selected to resolve this complex image detection problem. Convolution layer we used here can be regarded as moving fixed-size filter along the two dimensions of every image, the filter moves constantly and compute with every covered region until all the areas have been calculated. Finally, the layer will produce a fixed-size output using supervised learning method. The final output is a dot product of the two class distribution vectors which are the outputs of softmax layer. Also, we built up several convolution layers to extract the image feature information. A fixed-size input and a fixed-size filter produce an output we want. In our experiment, 3 plus 3 convolutional filters with stride 1 are used to produce feature maps. Zero padding is adopted here as well that refers to inserting zeros around the image dimension to keep the origin shape after convolution operation. Also, we chose PReLU as the activation function partly because it adds only a small number of parameters, which means that the computational complexity of the network and the risk of over-fitting may be reduced to some extent. In particular, there are fewer parameters when different channels use the same ai. This advantage is important for our likely complex network. PRelU is introduced in the paper of K.He et.al and its algorithm is shown as follows. PReLU can be seen as an improvement of ReLU, as a non-linear activation function. If α is set as a small and fixed value, PReLU turns to LReLU we have explored during previous coursework.α in PReLU is introduced as a trainable parameter in the paper of K.He et.al. This nature allows every feature map can correspond different values of α, and the parameters are optimized at the process of batch normalization. PReLU (Parametric Rectified Linear Unit)((He et al., 2015))

$$PReLU(x) = \begin{cases} \alpha * x & \text{if } x \leq 0 \\ x & \text{if } x > 0. \end{cases} \quad (1)$$

Before making a dot product on softmax layer, the front part of model used in our experiment can be regarded as an image classification problem, which is sort of similar with the MNIST classification problem we have researched last semester. According to the competition champion's model, Adam is selected as his learning rule. In practice, this learning rule does help to improve experiment performance in some cases in comparison with other common learning rules like AdaGrad. Adam is essentially an RMSprop with momentum terms that dynamically adjusts the learning rate of each parameter using the first-order and second-order moment estimates of the gradient. The main advantage of Adam is that after offset correction, the learning rate of each iteration has a certain range, making the parameters relatively stable. Besides, this straightforward method is easily to implement because it requires little memory, which play a vital role in our baseline system. Thus, we follow the choice of winner's network without comparing with other

learning rules, but to investigate potential effect of other network setting decisions in section 3.

In theory, we need to build a non-linear neural network to handle such complex painting detection problem, so training procedure is likely to result in over-fitting. That is why we introduced a wide range of means to discourage over-fitting. At first, batch normalization aims to achieve zero mean version for each training mini-batch. When training a certain layer, Batch Normalization performs normalization on each mini-batch data to make the output normalized to the distribution of N (0,1), reducing internal neuronal distribution change. The traditional deep neural network is training, the distribution of the input changes in each layer are changing, so training difficult, can only choose to use a very small learning rate, but each layer with the BN, you can effectively solve this problem , The learning rate can increase many times. So we apply batch normalization in the model. In addition to this, we add Dropout method in the experiment in order to prevent over-fitting in training set, dropout prevents over-fitting by randomly deleting half the hidden neurons in the network. Previous experiments prove dropout indeed avoiding the over-fitting behavior and obtaining smaller gaps. Then max-pooling is also used in the model to reduce the error taken by convolution layer parameters through picking the maximum value of a fixed-size region. The region of max-pooling is 2 plus 2 with stride 2. By doing this the number of neurons on two dimensions of every image is halved. Max-pooling can reduce noise in the specify small area. It can also realize reduce dimension without any impact on data accuracy because the relative position of feature hasn't changed. Flatten layer also be used as the transition from convolution layer to fully-connected layer to reduce the matrix dimension into one. An interesting algorithm used in the model is t-SNE dimensionality reduction technique. According to the introduction of the paper of Laurens et.al, the main idea of t-SNE dimensionality reduction technique is to regard high-dimensional data as point xi in high-dimensional space, and map xi to yi in low-dimensional space. The whole process must maintain all the points keeping relative the spatial distance((Maaten & Hinton, 2008)).

3. Baseline experiment

3.1. Structure

Figure 1 shows the basic architecture of our baseline system that is mostly refer to the model of champion in Painter by Numbers competition.
This section will explain the whole structure of our baseline system, first entered images have been processed with fixed-size of 3*64*64 dimensions. Then these images go through three convolution networks, and each convolution network contains three convolution layers and a max-pooling layer. During each convolution network, 3 plus 3 filter is applied in the network with stride 1 to figure out feature maps during every convolution layer, after this, picking out the maximum value in the 2 plus 2 max-pooling region which moves along the width and height of images so as to reduce half of the number of neurons. following three convolution networks, the output dimensions come into 64*8*8. Then flatten layer is used to convert the vector into one dimension. The fully-connected layers are formed in the dense layers, and a dropout layer is inserted before every dense layer to prevent the model over-fitting the training data. So we get the output with 60 dimensions in the end. Softmax layer is the final part of the model which aims to calculate the doc product of two vectors and get the result vector with 60 dimensions. The final result helps us to decide wither two images belong to one artist.

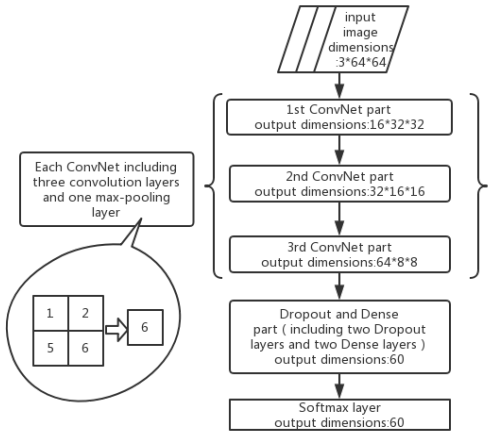


Figure 1. The structure of baseline experiment.

3.2. Experiment procedure

As explained in section 1.3 data set part, we preprocessed raw paintings at first, reducing their dimensions to 64 pixels with the consideration of running cost, even though such operation will lose image information. Meanwhile, during training procedure, label-retaining and random transformations such as zooms, shifts, shears, rotations and flips were adopted in our model. However, 60 classes with 100 paintings each one formed a small data set with regards to this rather complicated image detection problem, especially for our built deep convolutional neural network. This may give rise to possible underfitting during training process, which play a large part in influencing model performance and most setting decisions. The following will discuss some experiment results corresponding to our three research questions. Considering memory and running time constraints, our baseline system chose batch size 96 and 200 epoch consistently to train 5400 training data, at the same time, we are supposed to test its accuracy and loss on the remaining 600 validation examples. It should be noted that the third research question failed to be investigated in this paper. Instead, it will be explored in further work of our project.

3.2.1. HYPER-PARAMETER SETTING

This paper study two setting decisions rather than all of them due to limited time, learning rate and number of convolutional layers. As explained before, Adam is used to as learning rule, it adopts gradient descent method like most learning rules. This means that the learning rate of this basic network should be set carefully which seems to control the stride length of gradient descent. For example, we may miss the optimal data point and the cost function will increase significantly (see Figure 2) if this rate was set to be too large whereas learning may become stuck with a high cost value if learning rate was too small(Bishop, 2014).

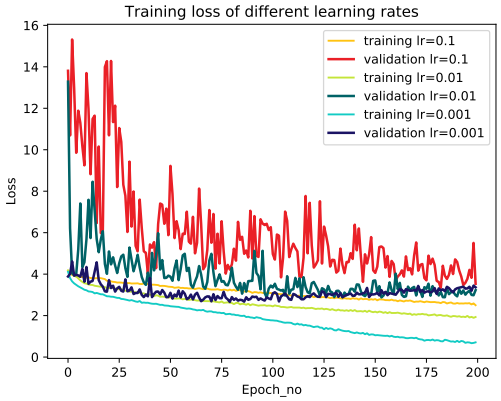


Figure 2. Loss of three learning rates.

Typically, 0.1 is set to be the starting value, then progressively reduce this rate(0.1, 0.01, 0.001). Figure 2 and Figure 3 give information about loss and accuracy of these three learning rates respectively. To be clear, all line figures used in this report, dark (red, green and purple colors here) and thin lines represent corresponding validation data. Obviously, red thin line in Figure 2 shows the highest validation loss when learnig rate is 0.1, and such loss decreased as learning rates decrease. By contrast, this line illustrates the lowest validation accuracy in Figure 3. And validation accuracies appear to rise with the learning rate reducing to 0.001. Thus, this value is used to be the default learning rate of the baseline network and applied in latter all experiments in this report, despite lower learning rate may be further explored in the next report considering the significant fluctuations in line figures.

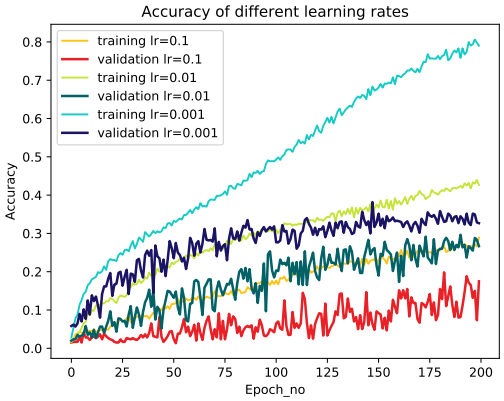


Figure 3. Accuracy of three learning rates.

Although validation accuracy tends to increase as the complexity of network rises, what our experiments show are very different with our previous view. We tries three different depths in terms of convolutional layers ranging from 8 to 14, increasing three layers each time. The lower depth, the higher validation accuracy according to Figure 4, the highest accuracy is obtained by red thin line that represent the network with 8 convolutional layers. The first explanation that we can come up with is largely because the small data set we chose versus original 13 times data set.

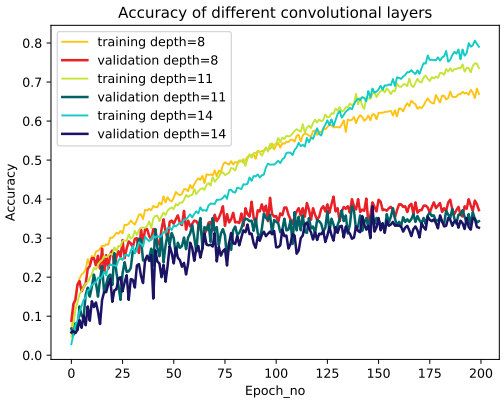


Figure 4. Accuracy of different depths.

This idea is confirmed from Figure 5, this loss graph shows that validation loss reduce during at first but rise afterwards with the increase of epoch number with respect to layer number 11 and 14. Our balanced subset is trained in rather complicated deep network, which cause overfit. As a result, we choose to add eight convolutional layers in our baseline network. Meanwhile, some approaches used to mitigate overfitting will be added as well.

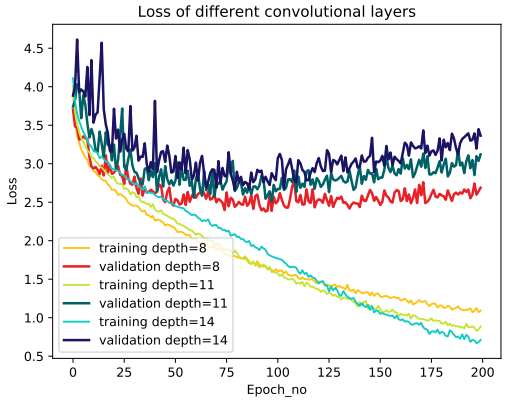


Figure 5. Loss of different depths.

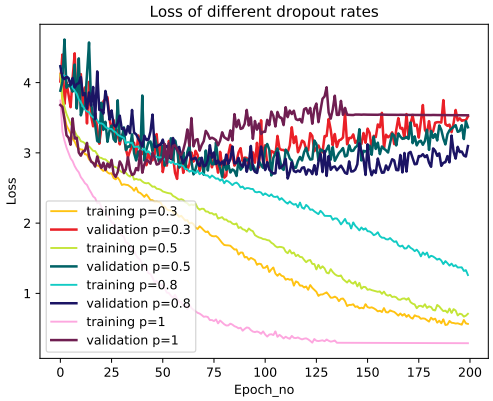


Figure 6. Loss of four dropout rates.

3.2.2. Discourage over-fitting

In terms of such complex image classification question, complicated neural network is expected to be set up in most cases because the more complex a model, higher classification accuracy can be achieved. However, complex network is prone to cause overfit. In order to cope with this, there are various approaches added to our baseline system such as L2 regularization, dropout and batch normalization.

In contrast, regularization is often used to mitigate overfitting behavior through changing cost functions and weights of networks while dropout reduces overfitting by changing network itself. Dropout is added in last layer before the flatten layer in this baseline system through randomly deleting half the hidden neurons in the network by default, whereas leaving the input and output neurons untouched. In practice, it seems like neural networks tend to trained with dropout, and the different networks will overfit in different ways(Nielsen), in the hope that the net effect of dropout will be to reduce overfitting.

However, the dropout rate is required to be considered for different type and amount of data set in such convolutional system. We tries four dropout rate 0.3, 0.5, 0.8 and 1(no effect on removing connections) when l2 regularization fails to take effect and the following two figures (Figure 6 and Figure 7)provide related performance. Apparently, Loss figure sees clear overfit trend from validation loss (dark and thin lines) except for that of p=0.3 and their overfit degree is bigger with the dropout rate increasing from 0.5. On the other hand, accuracy figure shows a rise in validation accuracy, although the accuracies of dropout rate 0.5, 0.8 and 1 level off at almost the same number after 100 epoch. Taking results of these two graphs into account, dropout rate 0.5 seems to achieve decent validation accuracy, despite it suffers with slightly overfit. In addition, we also adopt other means to help dropout to mitigate overfitting behavios, for instance, L2 regularization has been added in our baseline system and their parameter decisions are discussed as below.

On the other hand, accuracy figure shows a rise in validation accuracy, although the accuracies of dropout rate 0.5, 0.8 and 1 level off at almost the same number after 100 epoch. Taking results of these two graphs into account, dropout rate 0.5 seems to achieve decent validation accuracy, despite it suffers with slightly overfit. In addition, we also adopt other means to help dropout to mitigate overfitting behavios, for instance, L2 regularization has been added in our baseline system and their parameter decisions are discussed as below.

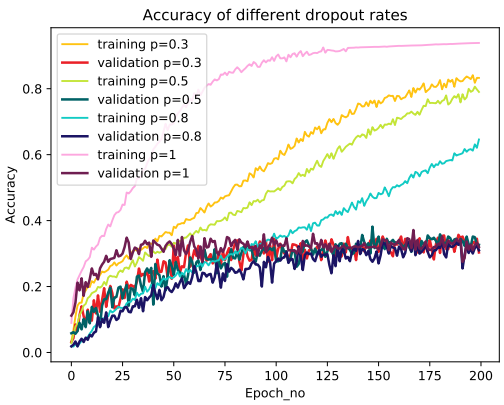


Figure 7. Accuracy of four dropout rates.

Regularization has been well known for most machine learners which introduces a new lamda variable into cost function as well as penalizes the weight. L2 regularization is adopted comparing with similar L1 regularization because L1 tends to shrink some weights to zero, leaving a few large important connections. In other words, L1 courages sparsity, which is not our expectation, in spite that both of them have the effect of penalizing larger weights. In this trail, baseline system is used when keeping learning rate, learning rule, hidden units number, depth and activation function so on as before chosen.

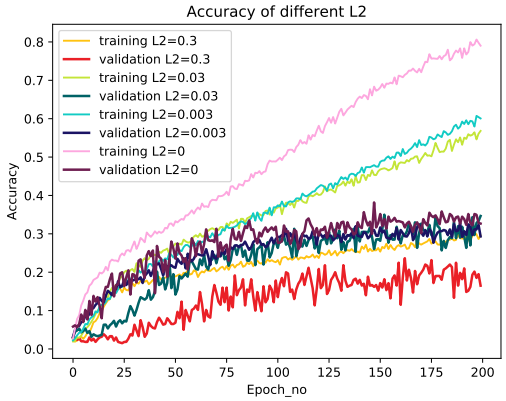


Figure 8. Accuracy of four L2 parameters.

Four L2 parameters are chosen in order to study the positive or negative effect of L2 regularization and their accuracies can be seen from Figure 8. In this experiment, dropout rate is set to be 0.5 when other parameter choice remain unchanged. The influence of this regularization will become larger as this parameter increase from 0 to 0.3. As we can see from figure 6, validation accuracies increase substantially according to red and other thin lines. Plus, model perform better if we chose not to add L2 regularization. But the gap between training and validation accuracy widens with the impact of regularization reducing to zero, which may indicate possible overfitting behavior as we see from the result of the above dropout trails. Since the accuracies are very close in terms of lines of 0.003, 0.03 and 0.3, we decided to use L2 parameter 0.03 to help our baseline system to curb overfitting behavior for our small data set in such deep network.

3.2.3. Algorithm and network choices

Although we know supervised means are likely to gain better performance than unsupervised ones, we chose unsupervised method that attempts to judge one author from the total 60 classes based on the product of two paintings' class distribution vectors in our baseline since it is easy to carry out. We plan to try suitable supervised learning methods in our next step and this part will be explained in detail in the last section future work.

4. Interim conclusion and evaluation

In conclusion, lower learning rate tend to achieve better performance especially for deep convolutinal network. Regarding small dataset, the basic idea that the more complex the network, the higher validation accuracies in most cases fail to be confirmed. In our baseline system, it even obtains an opposite view. That is why we chose shallower network. Also, we attempt to take a wide range of methods that can discourage overfit into account and decide their regularization impact through experiments. Selected small data set play a large part in influencing the performance of network, which results in different behavior o

5. Plan for future work

As the third research question indicated, we hope to try some effective algorithms and network in order to improve performance. For example, we plan to replace the final unsupervised method into supervised method like Siamese network. According to the paper of S.Chopra et.al, in Siamese network the weight of two inputs are equal and during the back propagation the gradients are contributed by two models and equal the sum of two models(Chopra et al., 2005). The inputs of Siamese network have two channels, they are picked up feature maps separately but sharing weight during this process. These two channels are connected at the last fully-connected layer which is different from the unsupervised network which combines two inputs at the beginning using a simple dot product(Bell & Bala, 2015). In most cases, Siamese network, such supervised learning performs better in contrast to used unsupervised method in out baseline system regarding increasing validation accuracy.

However, the Siamese network's parameters are nearly doubled compared with that of unsupervised network, so its training process tend to require longer running time and memory. Taking these resources constraints into consideration, we failed to use Siamese network in this baseline experiment. However, we plan to use Siamese network to detect artist classes in the next step of our project.

So far, most existing approaches depend on convolutional neural network(CNNs) for image detection using a shallow channel which leads to loss of detailed information. But it is easy to conclude that training effect will better if we can obtain more information. Thus, we searched and found that there are several means that can remain full information of images. For instance, a two-channel convolutional neural network (including one shallow and one deep channel can be considered as the deep layers constantly extract and iterate the underlying characteristics while the shallow channel can reserve original images' rough information(Li et al., 2018) .

Besides, although we change our raw unbalanced data set to a small balanced data set, which mitigate limited memory and running time problem, it cause overfit easily in our deep network. Thus, we may try to use a larger data set in our next coursework.

References

Bell, Sean and Bala, Kavita. Learning visual similarity for product design with convolutional neural networks. *ACM Trans. Graph.*, 34:98:1–98:10, 2015.

Bishop, Christopher. *Machine Learning and Pattern Recognition*. Springer-Verlag New York, 2014.

Chopra, Sumit, Hadsell, Raia, and LeCun, Yann. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pp. 539–546. IEEE, 2005.

He, Kaiming, Zhang, Xiangyu, Ren, Shaoqing, and Sun, Jian. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *Proceedings of the IEEE international conference on computer vision*, pp. 1026–1034, 2015.

Li, Sumei, Fan, Ru, Lei, Guoqing, Yue, Guanghui, and Hou, Chunping. A two-channel convolutional neural network for image super-resolution. *Neurocomputing*, 275:267–277, 2018.

Maaten, Laurens van der and Hinton, Geoffrey. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008.

Nielsen, Michael. *Neural network and Deep Learning*. MIT Press.