Group Report For Project 1

Donghao Li
Department of Mathematics
HKUST
dlibf@ust.hk

Jiamin Wu
Department of Mathematics
HKUST
jwubz@ust.hk

Wenqi Zeng
Department of Mathematics
HKUST
wzengad@ust.hk

Yang Cao
Department of Mathematics
HKUST
ycaoau@ust.hk

Abstract

In this project, we conduct feature extraction of the MNIST data through different neural networks, such as Scattering Net, pre-trained VGG19 and ResNet50. For each feature extraction method, we perform image classification with logistic regression, random forest, LDA, and SVM based on respective features. We also compare the performances of the classifier after feature extraction with the performance of the classifier based on raw data. In addition, we employ two different models to identify the Raphael's paintings from the forgeries and compare their accuracies.

1 Introduction

Convolutional neural networks have been widely used to settle the problem of image classification[5]. In this project, we intuitively decompose a Convolutional Network into two parts, feature extractor and classifier.

Specially, we conduct feature extraction by Scattering Net and two pre-trained neural networks, VGG19 and RESNET50. Scateering Net captures translation invariant features which is stable to deformations [2]. The filter in Scattering Net is carefully designed and fixed rather than learned from data. While, VGG [11] and ResNet [3] uses 3×3 convolution filters, which are learned from data, to increase the depth of the network. In addition, ResNet considers "shortcut connections", which are the connections skipping one or more layers [8]. We use the methods of logistic regression, random forest, LDA, and SVM on the extracted feature and raw data as classifiers. We also try the fine-tuning with VGG19.

We mainly conducted classification on the MNIST database[12]. We found that although feature extraction do help improve performance of classifiers using logistic regression, LDA or SVM, it could actually worsen the performance of the classifier using random forest. We provide a possible explanation to this interesting phenomenon and conduct experiments to validate our explanation. In addition, we train two different models to identify Raphael's paintings from the forgeries. The first one is Scattering Net combined with fully connected neural network, and the second one is a convenlutional neural network with similar structure of Scattering net, whose parameters are all learned from data. By comparing the performance of these two methods, we show the effectiveness of the Scattering Net.

2 Experiment

2.1 Overview

In this section, we would conduct experiments on MNIST database[12]. The MNIST database of hand written digits has a training set of 60,000 examples, and a test set of 10,000 examples. Each examples are size-normalized and centered in a 28×28 image. We use Scattering Net, pre-trained VGG19 and RESNET50 as our feature extractor, and logistic regression, random forest, LDA, and SVM as our classifier.

2.2 Feature extraction

We do feature extraction by Scattering Net and two pre-trained neural networks VGG19 and RESNET50. The meaning behind the latter one, which is known as transfer learning, is to use models trained on one big database as a starting point to another related smaller one. In our case both models are pre-trained on ImageNet database.

2.2.1 Scattering Net

Scattering Net can be regarded as a feature extractor with a structure similar to a convolutional network. It is composed of three convolutional layers, and every output of the convolution operation would be collected as output of the network. The detailed structure is shown in Figure 1. We can see that it behaves like a convolutional network without fully connected layers. However, the main difference between them is that the filter in Scattering Net is fixed and carefully designed by wavelet transform (or other methods), rather than learned from data using gradient descent like what deep convolutional networks do. Therefore, there are some properties can be guaranteed such as its invariant under some transformation.

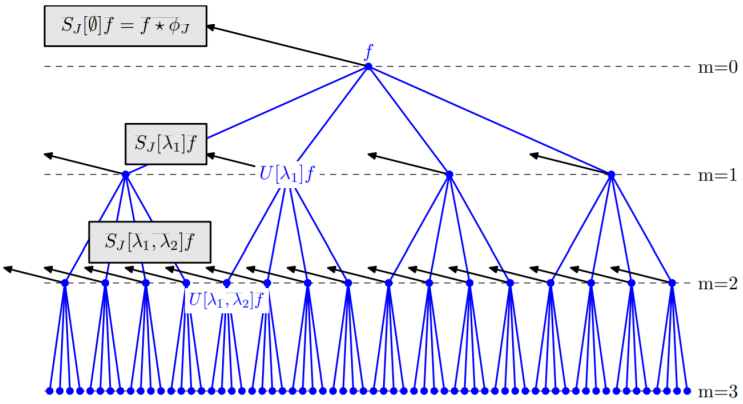

Figure 1: Structure of Scattering Net

Luckily, there is a mature python package called Kymatio [1] which offers wavelet scattering transform in Python and can use GPUs. The implementation process is quite easy. All we need is putting the one channel digital hand written images with size 28×28 in to the model and then collecting the output as 3969 dimensions features.

2.2.2 VGG19

One improvement of VGG19 over AlexNet is the replacement of large convolution kernels (11×11, 7×7, 5×5) in AlexNet with several consecutive 3×3 convolution kernels. For a given receptive field (the local size of the input picture associated with the output), the use of stacked small convolution kernels is superior to the use of large convolution kernels. Because multiple layers of nonlinear layers can increase network depth to ensure learning of more complexity with still relatively small cost

(less parameters). On the other hand VGG19 can consume huge computing resources and uses more parameters due to its 3 full connection layers, resulting in more memory usage of 140M.

2.2.3 ResNet

The Deep residual network (ResNet) born from a simple observation why very deep networks perform worse when adding more layers is a milestone in the history of CNN images. ResNet refers to VGG19 and has been modified based on it. The residual unit has been added through the short-circuit mechanism, as shown in the following Figure 2. Even if ResNet50 is deeper than VGG19, the size of it is actually quite small. Replacing the fully connected layer with global average pooling can reduce the size of the model to 102MB.

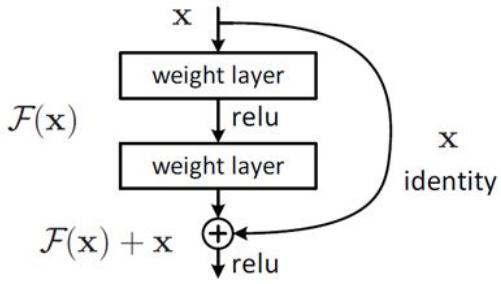

Figure 2: Structure of ResNet

2.3 Image classification with traditional methods

Image classifications with traditional supervised learning methods based on the features extracted may give the identification accuracy straightforward. As [7] indicated, LDA is a linear classifier with the main purpose of selecting the best projection direction. After determine the considerable direction, thresholds of different classes can be dissolved. It is easy and straightforward. However, it may cause the problem of overfitting sometimes and the performance may be poor when dealing with the high-dimensional data. Random forest is a decision tree algorithm. Some samples are selected from the whole database to generate a cart tree and then be put back in the initial database. After some random sampling process, it derive the final result from the cart trees. Random forest is easy to understand and can be fast in learning process although it can be overfit sometimes [9]. SVM is nonlinear method aiming to find a maximum margin in even higher dimensions. It can solve nonlinear and high-dimensional problems but the cost is running time [10]. According to [4], Logistic regression is a typical method with regression method to predict the results. It is easy to understand but can be under-fitting sometimes.

In order to compare the results among different feature extraction methods and classification methods, we use the methods of logistic regression, random forest, LDA, and SVM on the feature extracted data with VGG19, ResNet50 and Scattering Net respectively. To show the effect of feature extraction, we also get the accuracy scores on the raw data of MNIST database with the four classification methods. The implement of the methods is based on the package scikit-learn in python3 with default parameters. Our experiment on the MNIST database give the accuracy scores on these supervised classification methods. The comparison and analysis can be found in Section 2.5.

2.4 Training of the last layer

Despite the traditional methods, we could also train thelast layer ourself. Fine-tuning the last layers of a network is a transfer learning method. The motivation is very intuitive. We can decompose a Deep Convolutional Network into two parts, the first parts is a feature extractor, the second part is a classifier. Then a network trained on a huge database (like ImageNet) have learned a good feature extractor, which can extract semantic information and can be used on other tasks. When the target database is small, if we simply train a new network on it, it can easily overfit the train set. So, we just fine-tune the last layers (can be regarded as classifier) of the trained network, then we can avoid overfitting and make use of the information in the original database.

However, we think that MNIST database is quite simple and we can get a good classification results by training a much smaller network like LeNet5. And it seems not reasonable to use such large network trained on ImageNet to do fine tuning on MNIST. So we just try the fine-tuning with VGG19 network.

The implementation is also very easy, since Keras provide detailed examples on how to do transfer learning. However, there is a problem that we need to change the gray scale image in to RGB image, and also need to resize the input image to match the network requirements. We tried to resize the figure to 32×32, 64×64 and 224×224 to see which size can be better.

2.5 Comparison

Since we can't intuitively feel the high-dimensional space, we take PCA and t-SNE to reduce dimensionality on the original MNIST database as well as three databases extracted from networks and visualize them.

PCA transforms a set of possible correlation variables into a set of linear uncorrelated variables by orthogonal transformation, which in another way converts multiple indicators into a few comprehensive indicators. The converted set of variables is called principal components and can reflect most of the information of the original variables, while information contained in them do not repeat to achieve the purpose of dimensionality reduction. The results is shown in Figure 3.

Figure 3: Comparison of figures on 4 datasets after PCA

The t-SNE algorithm maps data points to probability distributions and uses conditional probability to measure the similarity between data points by making it as high as possible between high and low dimensions. Dimension reduction by t-SNE not only maintains the data difference, but also maintains the local structure of the data well. The results is shown in Figure 4.

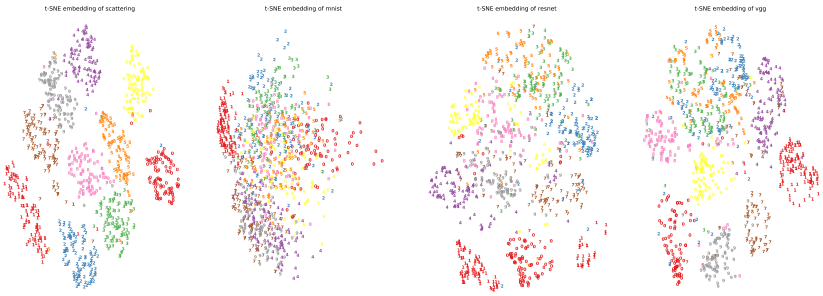

Figure 4: Comparison of figures on 4 datasets after t-SNE

0 and 4 are the easiest ones to distinguish on original MNIST data set while 0 and 1 are most obvious ones after feature extraction. The data was almost smashed and clusters was completely invisible due to the loss of too much information in the process under the linear dimensionality reduction of PCA.

t-SNE is more capable of exhibiting data-reduced-dimensional clusters than PCA, which may be related to PCA's attempts to preserve linear structure and t-SNE's attempts to preserve topology (neighborhood) structure. But t-SNE is much more computationally expensive than PCA correspondingly.

In four databases, the original MNIST data is worse than the data after feature extraction. If the selected features lack certain representations to distinguish each category, the accuracy of the model is greatly reduced, no matter what classification strategy to use. However, the features extracted by neural networks (Scattering Net, VGG19, ResNet50) are adjusted according to the image and its label which can be somehow considered as supervised feature extraction. At the same time the convolution kernels in neural networks have included the connections between pixels.

On the three databases after feature extraction, clusters of feature extraction by Scattering Net have the most clear boundaries. One possible reason is that VGG19 and ResNet50 are trained on 1000 categories on the color picture database ImageNet, and there is no fine tune for the last layer on gray picture database MNIST. Another conjecture is that features are extracted from the penultimate layer of VGG19, after which the layers are fully connected layers. The features maybe only contains category information. Thus, in MNIST case, it might be like mapping the feature of numbers onto pandas or lions.

Feature extraction directly affects the classification results. For example, in the features extracted by VGG19, number 2, 3, 5 are not well distinguished so that the accuracy of classification on 2, 3, 5 is lower than that of others.

The comparison between different feature extraction methods, shown in Table 1, shows that scattering net method has the best performance on the four methods. Among different classification methods, SVM seems to be the worst.

Table 1: Performance comparison between different feature extractors and classifiers

	Scaterring Net	Resnet	VGG	Raw
Logistic	0.9894	0.9854	0.9757	0.9181
Random Forest	0.9823	0.9337	0.9500	0.9700
LDA	0.9910	0.9761	0.9657	0.8730
SVM	0.9531	0.9536	0.9476	0.1135

2.6 Discussion

We can see an interesting phenomenon from Table 1. That is, when we compare using random forest as a classifier with others, we could find their performance diametrically opposed. Note that random forest randomly selects one pixel (variable) of the image as its filter. Suppose that, after feature extraction, the pixels that affect the classification is fewer, but they could be more effective. Then we could see that the chance for random forest to make a mistake could actually be higher then before. But other classification methods would somehow take the weights of pixels, which measure the impact of the pixels, into consideration. So they may establish better results after feature extraction. Then we perform a lasso regression of the extracted feature to validate our thoughts. The Lasso Path is shown in Figure 5, there are 8192 possible variables in total.

We can see from Figure 5 that the effective variables are sparse. Only a few variables in 8192 variables would actually contribute to the classification. This result could somehow verify our hypothesis. The theoretic analysis could be discussed in our future work.

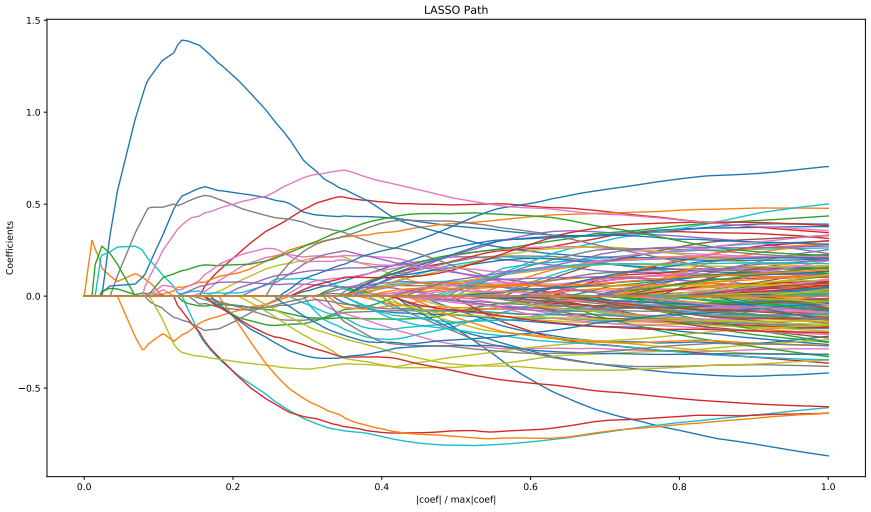

Figure 5: Lasso path of feature extracted by ResNet. In total, the dimension of feature is 8192. From this figure, we can see that feature is highly sparse.

3 Identification of Raphael's paintings from the forgeries

3.1 Task introduction

For some famous artists like Raphael, there are many imitations which would disturb the market and art research. In this section, we want to use statistic methods to identify if the painting belongs to Rapheal.

The database contains 28 digital images of painting. The resolution of the pictures range from 1192×748 to 6326×4457 and their physical size differs from each other. There are labels for some of these paintings. 12 of the paintings are from Raphael, while 9 of them are from other artists. And the remaining 6 of them are contentious. (Thanks for Prof. Wang Yang in HKUST providing the database.)

There are also some other work focusing on identification like [6], [12]. In their work, they propose a feature extractor called Geometric Tight Frame. Its core idea is use some carefully defined filters and do convolution operation on the image, then calculate some statistic like quantiles as feature. We would compare this methods later.

3.2 Models

We employ two models to solve this task. The first one is the Scattering Net combined with fully connected neural network (SNN). We choose this one because Session 2.5 has shown the effectiveness of this method. The second one we choose is training a convolutional neural network (CNN) with similar structure with Scattering Net, but all the parameters are learned from the data. We design this model to show the effectiveness of the Scattering Net.

3.3 Experiment

Since the database only contain 21 images with label, (Among them, the 28th painting can not be loaded into python, therefore we just ignore it, so we only have 20 training samples.) we should carefully design data augmentation to fully use these data.

First, we choose to turn RGB/RGBA images into gray scale images. This preprocessing method is also used in [6] and [12], because the color feature might not be as useful as shape feature or texture feature. Also, each picture might have different background and color, which might cause the model overfit more easily. Then, normalization is also applied to the gray scale images.

The most important part of data augmentation processing is random crop. To be more specific, we choose a patch with size 64×64 randomly as input each time. In the training process, we define the label of that random patch the same as the the original one. In the validation and test process, we randomly sample 500 patches and average the predicted label as final predicted label. Since one painting always have the similar pattern in every patch, we can fully make use of each high resolution painting in this way.

Moreover, to avoid overfitting, we use Leave One Out Cross Validation. It could also make fully use of the training data and get validation results correctly.

Results of the two models are showed in Table 2. We also compared the results from Table 1 in [12], Where GTF+OD means using Geometric Tight Frame to do feature extraction and using outlier detection to distinguish the artist. Similarly, GTF+NN means using neural network as classifier after feature extraction.

Table 2: Train and validation accuracy of different methods

Model	CNN	SNN	GTF+OD	GTF+NN
Train Accuracy	84.21%	92.82%	–	–
Validation Accuracy	30.00%	85.00%	81.00%	81.00%

We can see that the CNN model performs even worse then random guess. This should be because CNN have too many parameters to fit and it would overfit easily. Then we can see that SNN gives a better performance than CNN, GTF+OD and GTF+NN. Because the GTF based feature extraactor only consider the statistic of the feature get from convolution transform, but SNN would make use of the whole feature and then using average of all patches.

Based on Table 2, we choose SNN to predict the unknown paintings. We also compare the results from [6].The results is shown in Table 3. Though the three results differ from each other, we can see that there is a trend and some kind of consistency. The GTF+OD is the most strict one, and it reject all the unknown paintings to be Raphael. Then is GTF+NN, it thinks No.1 and No.26 belong to Raphael. Finally, our methods regards only No.7 and No.10 to be fake painting. Therefore, we can say that the No.7 and No.10 is least likely ones to be Raphael's painting since all the three methods thinks it is Fake. And No.1 and No.26 is very likely to be Rapheal's painting.

Table 3: Identification results of differents methods

Painting ID	1	7	10	20	23	25	26
GTF+OD	Fake	Fake	Fake	Fake	Fake	Fake	Fake
GTF+NN	Raphael	Fake	Fake	Fake	Fake	Fake	Raphael
SNN	Raphael	Fake	Fake	Raphael	Raphael	Raphael	Raphael

4 Conclusion

In this work, we use Scattering Net, VGG19 and ResNet50 to perform feature extraction on the MNIST database. Then, we use logistic regression, random forest, LDA and SVM to do classification. Through comparing the performances of all combinations of feature extractors and classifiers, we found a interesting phenomenon. That is, the performance of random forest opposes performances of other classifiers. We give an explanation, and conduct experiments to validate our hypothesis.

In addition, we use two different methods to identify Raphael's paintings from the forgeries. We compared our results with others' and provide an identification.

5 Contribution

- Donghao Li: Feature extraction; Feature visualization; Fine-tuning deep neural networks; Identification of Raphael's paintings; Performance comparison.
- Jiamin Wu: Image classification; Performance comparison.
- Wenqi Zeng: Feature extraction; Feature visualization; Performance comparison.
- Yang Cao: Discussions; Summarization.

References

[1] Mathieu Andreux, Tomás Angles, Georgios Exarchakis, Roberto Leonarduzzi, Gaspar Rochette, Louis Thiry, John Zarka, Stéphane Mallat, Joakim andén, Eugene Belilovsky, Joan Bruna, Vincent Lostanlen, Matthew J. Hirn, Edouard Oyallon, Sixin Zhang, Carmine Cella, and Michael Eickenberg. Kymatio: Scattering transforms in python, 2018.

[2] Joan Bruna and Stéphane Mallat. Invariant scattering convolution networks. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1872–1886, 2013.

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[4] David W Hosmer and Stanley Lemesbow. Goodness of fit tests for the multiple logistic regression model. *Communications in statistics-Theory and Methods*, 9(10):1043–1069, 1980.

[5] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[6] Haixia Liu, Raymond H Chan, and Yuan Yao. Geometric tight frame based stylometry for art authentication of van gogh paintings. *Applied And Computational Harmonic Analysis*, 41(2):590–602, 2016.

[7] Sebastian Mika, Gunnar Ratsch, Jason Weston, Bernhard Scholkopf, and Klaus-Robert Mullers. Fisher discriminant analysis with kernels. In *Neural networks for signal processing IX: Proceedings of the 1999 IEEE signal processing society workshop (cat. no. 98th8468)*, pages 41–48. Ieee, 1999.

[8] Yohhan Pao. Adaptive pattern recognition and neural networks. 1989.

[9] Juan José Rodriguez, Ludmila I Kuncheva, and Carlos J Alonso. Rotation forest: A new classifier ensemble method. *IEEE transactions on pattern analysis and machine intelligence*, 28(10):1619–1630, 2006.

[10] Bernhard Schölkopf, Alexander J Smola, Francis Bach, et al. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2002.

[11] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[12] Christopher J.C. Burges Yann LeCun, Corinna Cortes. The mnist database of handwritten digits.