Image Classification with Transfer Learning

Lucen Zhao
MATH4432 Project 3
Student ID: 20256435
lzhaoaj@connect.ust.hk

Abstract

This project tried several models for transfer learning with the pre-trained VGG19 models. It turns out that LDA, Logistic regression and SVM all reached quite good results with the transfer learning feature vectors. I used the models to predict the authentication of Raphael's paintings and it was evaluated by majority voting.

1 Introduction

Transfer learning is a useful method for extracting high-level features from images. By using the pre-trained tranfer learning model, we can extract abstract feature vectors in high dimension from our input images, and then it can be used to solve different problems, such as image classification, image generation and image captioning.

In this project, we used the VGG19 model to classify images with transfer learning method. Firstly, we extracted feature vectors with VGG19 model for 3 datasets we used: MNIST dataset, fashion-MNIST dataset and Raphael's Painting dataset. Then we visualized the features with PCA and clustering, and classified the features with different supervised learning models. The best models were chosen with cross-validation.

2 Data Preprocessing

In this project, 3 datasets are used: MNIST dataset, fashion-MNIST dataset and Raphael's Painting Dataset. In general, the MNIST dataset and fashion-MNIST dataset are in large scale, each with around 70,000 images, while the Raphael's Painting dataset only have dozens of paintings that need to be carefully pre-processed to enlarge the size of training and testing samples. Samples of images are shown in Figure 1, Figure 2. Because the Raphael's Painting dataset are read from .jpg files, the sample data are not included in the report.

2.1 MNIST and Fashion-MNIST Dataset

Because of the lack of computational power, I ran the VGG19 model on the CPU of my laptop and it took me much time extracting features. Hence for the MNIST dataset and fashion-MNIST dataset only part of them were used for training (MNIST: 15,000 for training and 3,000 for testing; fashion-MNIST: 10,000 for training and 2,000 for testing).

Each of the images are enlarged to size 224 x 224 to fit the input size of VGG19.

2.2 Raphael's Painting Dataset

For the Raphael's Painting dataset, because there were a limited number of paintings, I randomly selected different "patches" from each of the paintings for training and testing. Each of the square patches are enlarged to 224 x 224 to fit the input size of VGG19.

30th Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain.

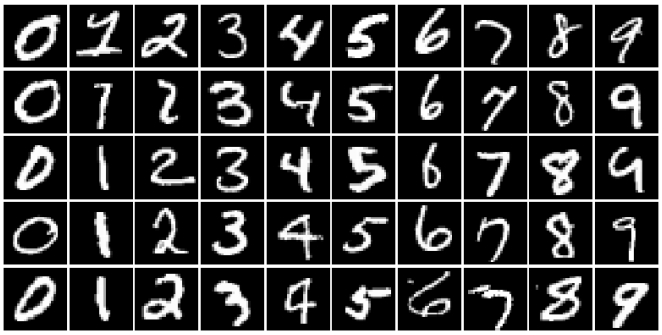

Figure 1: Samples from MNIST database

Figure 2: Samples from fashion-MNIST database

In the training set, each patch of the painting was assigned the label of the original painting, while for testing, the final label assigned to each painting is the majority of labels of patches (e.g. for 100 patches of one image, if 51 or more patches are assigned with label "Raphael", then it is classified as an authentic Raphael's painting). For training, 50 patches are randomly extracted from each image, while for testing, 100 patches are randomly extracted from each image.

3 Transfer Learning

In this project, I chose VGG19 developed by the Visual Geometry Group of Oxford University as our transfer learning model.

3.1 Feature Extraction with VGG19

According to the layer configuration of VGG19, the VGG19 model consists of 5 groups of convolutional layers and 3 fully-connected layers. Each group of the convolutional layer is combined with a set of convolutional and pooling layers, making the total number of layers of the VGG model 19.

The output from different layers can all be used as features vectors. In this project, I chose the output from fully-connected layer "fc7" as the feature vector with 4096 dimension, which is a common choice in other research works.

Another common choice of feature vector is the one from the last convolutional layer, with size of 512. Because I did not save this feature vector when I ran VGG19 model and I did not have enough time to run the model again, I did not use this feature vector for classification of MNIST and fashion-MNIST

datasets. However, it is used for the Raphael's painting dataset, and in future works I might apply it to the other two datasets as well.

3.2 Feature Visualization

The features are visualized with PCA in 3D, with each axis representing one component obtained from PCA. The number of total components in our PCA model is 20, and the results of visualization with 3 components ranking highest with PCA are shown in Figure 3, Figure 4 and Figure 5. To make the plot clearer to see, 20 samples from each class are extracted from MNIST and fashion-MNIST dataset, while 50 samples from each class are extracted from Raphael's painting dataset.

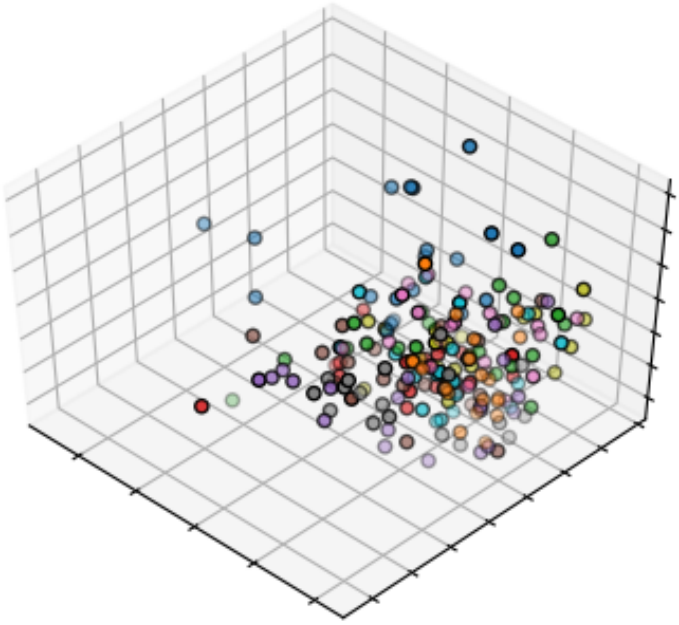

Figure 3: PCA visualization of MNIST database

According to the visualized 3D plots, for MNIST and fashion-MNIST dataset, because there are 10 classes in total, hey are not distinctive enough according to the plots. However, if we look into each color (i.e. each class), they are still tightly clustered together. This shows that a good classification result can be obtained if more dimentions are added.

4 Classification

Because the dimension of original features vectors was too high, for MNIST and fashion-MNIST datasets, I used PCA to reduce the dimension of features to 200.

4.1 Classifiers

According to my results in project 1, LDA, logistic regression and KNN with n=3 has satisfying performance in MNIST classification. Hence, there 3 models are still used as classification methods in this project.

Besides, non-linear models are used as well in this project, such as random forest (RF), gradient boosting and support vector machine (SVM).

Logistic Regression There is a list of optimizers can be chosen with logistic regression. Because in project 1, the solvers were compared with cross-validation with MNIST dataset, this time I used the best one got in project 1: Newton-CG solver. Because in project 1, multi-class logistic regression

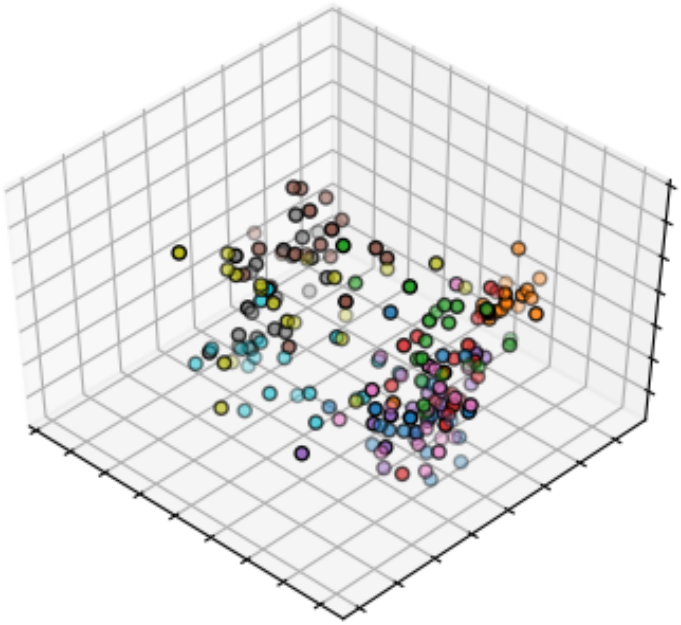

Figure 4: PCA visualization of fashion-MNIST database

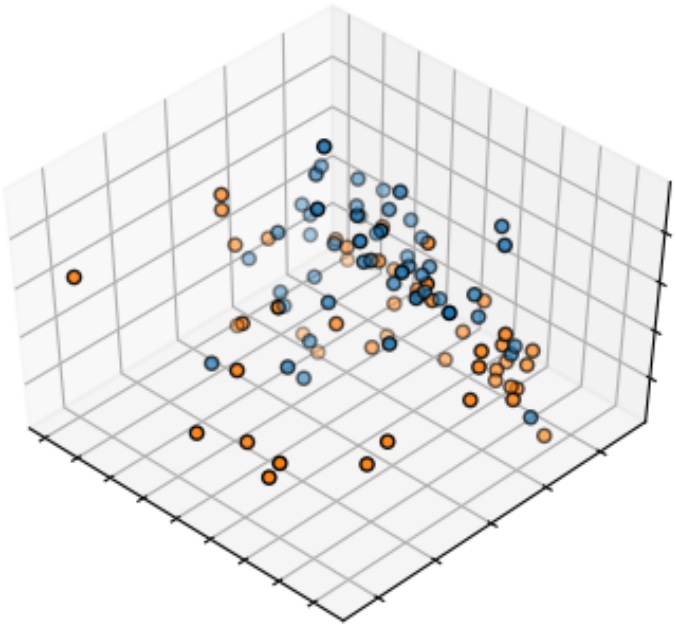

Figure 5: PCA visualization of Raphael's Painting database

model slightly under-performed the traditional one, this time I directed used the traditional 2-class logistic regression model.

LDA LDA model can classify the input data based on a distinct linear function for each class. Compared with logistic regression, much time can be saved by using LDA.

KNN KNN with $k = 3$ is used in this project because it reached the best performance in project 1. However, because the choice of k largely influence the performance of the model, other choices of k (10, 20) will also be compared with this one with cross-validation.

Random Forest The random forest classifier is used for classification. The number of estimators and the max number of features are determined by cross-validation.

Gradient Boosting Gradient boosting classifier is also a tree-based method for classification. The number of estimators and shrinkage are determined by cross-validation.

SVM Support vector machine classifier for multi-class is handled with one-to-one scheme. Different types of kernels are tried, while the final type is determined by cross-validation.

4.2 Validation and Evaluation

For logistic regression and LDA, there are no additional parameters to be decided. However, for KNN, RF, Boosting and SVM, the parameters of the models are chosen with cross-validation with k=3 for MNIST and fashion-MNIST datasets, while for Raphael's Painting dataset we chose the leave-one-out validation by leaving one painting out (instead of leaving one patch out). In this model selection phase, the accuracy is used as the evaluator of a model.

We used classification on original images as the baseline to compare with the results obtained from the feature vectors of transfer learning. Moreover, for the MNIST dataset, the results from project 1 can also be used to compare with our transfer learning results.

The error rate (accuracy), F1-score and confusion matrix of classification is used to evaluate our classification results.

5 Experiment

5.1 Model Selection

The results of model selection are based on the accuracy of classification on the transfer learned features of each dataset.

Because there are three datasets in all, it is not efficient to try all these models on all three datasets for model selection. Hence, the models are selected with our final dataset: Raphael's painting dataset. The models are tested on other datasets just to show that transfer learning can out-perform the models with original images.

The results of leave-one-out cross-validation for model selection are listed in Table 1. LDA, logistic regression and SVM (with polynomial kernel) have achieved the best performance among all models.

Besides, since these models does not cost great computational power, I trained these models with the full feature vector (size=4096) as well with more choices of parameters, and the results are shown in Table 2. In general, these results out-performed the ones with PCA, hence I chose to use the 4096-dimension feature vectors directly to generate final results. According to the cross-validation results, LDA, logistic regression and SVM reached best performance, with the SVM achieved best performance with polynomial kernel of degree 3.

5.2 Final Results

Finally, 3 models are chosen to generate the final results:

- LDA model.
- Logistic Regression model with Newton-CG solver.
- SVM with polynomial kernel of degree 3.

5.2.1 MNIST and Fashion-MNIST Datasets

To test the performance of these models, I trained these models with MNIST and fashion-MNIST dataset, both with the feature vectors obtained from transfer learning, and with original images (baseline). The results are shown in Table 4 and Table **??**.

Table 1: Model Selection via CV

Model	Score
LDA	0.7427
Logistic Regression	0.7364
K-nearest neighbour (k=3)	0.6482
K-nearest neighbour (k=10)	0.6445
K-nearest neighbour (k=20)	0.6455
Random forest (ntree=500, m=p)	0.4827
Random forest (ntree=500, m=$\log p$)	0.4582
Random forest (ntree=500, m=$\sqrt{p}$)	0.4836
Random forest (ntree=1000, m=p)	0.4736
Random forest (ntree=1000, m=$\log p$)	0.45
Random forest (ntree=1000, m=$\sqrt{p}$)	0.4727
Gradient boosting (ntree=500, $\lambda = 0.01$)	0.6355
Gradient boosting (ntree=2000, $\lambda = 0.01$)	0.6482
Gradient boosting (ntree=5000, $\lambda = 0.01$)	0.6554
Gradient boosting (ntree=2000, $\lambda = 0.001$)	0.5936
Gradient boosting (ntree=5000, $\lambda = 0.001$)	0.6218
SVM (kernel=linear)	0.7191
SVM (kernel=poly)	0.72
SVM (kernel=rbf)	0.54
SVM (kernel=sigmoid)	0.6227

Table 2: Model Selection via CV with 4096-dimention feature vector

Model	Score
LDA	0.7236
Logistic Regression	0.74
SVM (kernel=linear)	0.7327
SVM (kernel=poly, degree=2)	0.7336
SVM (kernel=poly, degree=3)	0.7473
SVM (kernel=poly, degree=4)	0.74
SVM (kernel=poly, degree=5)	0.7245
SVM (kernel=rbf)	0.7318
SVM (kernel=sigmoid)	0.4482

According to the tables, for both datasets and all models, the results of transfer learning are largely better than the baseline results. Hence we can say that transfer learning is exceptionally effective for this image classification problem. Specifically, for MNIST dataset, the LDA model achieved best results (2959 correct ones out of 3000 test data). The confusion matrices of best models are shown in Figure 6.

```
[[286   0   0   0   0   0   0   1   0   0]
 [  0 359   1   0   0   0   0   1   0   0]
 [  0   0 275   0   0   0   1   2   0   0]
 [  0   1   2 332   0   0   0   0   0   0]
 [  0   0   0   0 278   0   0   0   0   1]
 [  0   1   0   0   0 265   1   0   1   1]
 [  2   0   1   0   0   0 275   0   3   0]
 [  0   0   2   0   4   0   0 301   0   0]
 [  0   0   3   0   1   2   0   0 292   2]
 [  1   1   0   0   0   1   0   2   2 296]]
```

Figure 6: Confusion matrix of LDA model on MNIST dataset.

Table 3: Test results of MNIST datasets

Model	Accuracy	Precision	Recall	F1-Score
LDA (Baseline)	0.84	0.85	0.84	0.84
Logistic Regression (Baseline)	0.90	0.90	0.90	0.90
SVM (Baseline)	0.19	0.54	0.19	0.13
LDA (Transfer Learning)	0.99	0.99	0.99	0.99
Logistic Regression (Transfer Learning)	0.98	0.98	0.98	0.98
SVM (Transfer Learning)	0.97	0.97	0.97	0.97

5.2.2 Raphael's Painting Dataset

To test the performance of this dataset, because there is no separate testing set with labels, I still used leave-one-out cross-validation to evaluate the model. Here the leave-one-out means leaving all patches of a painting instead of one patch. For each painting, there are 6 models used: LDA, SVM, Regression and their corresponding baseline models (with original image as input). The results of prediction on each image are shown in Figure 7 and Figure 8. It is clear that all models with transfer learning greatly outperformed the ones with original image. For transfer learning, those 3 models reached similar performance.

Painting + Label	2 R	3 R	4 R	5 R	6 R	8 R	9 R	11 N	12 N	13 N	14 N	15 N	16 N	17 N	18 N	19 N	20 N	21 R	22 R	24 R	27 R	28 R	Average	Average R	Average N
Transfer Learning LDA: Predict as Raphael	47	34	34	28	46	18	40	7	21	6	4	14	18	26	10	16	10	49	41	44	16	31			
Transfer Learning LDA: Correct Percentage	94	68	68	56	92	36	80	86	58	88	92	72	64	48	80	68	80	98	82	88	32	62	72.3636364	71.3333333	73.6
Baseline LDA: Predict as Raphael	42	39	11	37	42	6	47	22	31	38	28	30	29	43	10	11	22	42	33	19	30	1			
Baseline LDA: Correct Percentage	84	78	22	74	84	12	94	56	38	24	44	40	42	14	80	78	56	84	66	38	60	2	53.1818182	58.1666667	47.2
Transfer Learning SVM: Predict as Raphael	48	39	43	37	44	16	44	0	20	9	27	5	17	30	1	10	7	48	44	36	11	38			
Transfer Learning SVM: Correct Percentage	96	78	86	74	88	32	88	100	60	82	46	90	66	40	98	80	86	96	88	72	22	76	74.7272727	74.6666667	74.8
Baseline SVM: Predict as Raphael	43	46	12	42	47	11	46	5	17	36	33	20	22	38	5	1	1	41	42	17	30	0			
Baseline SVM: Correct Percentage	86	92	24	84	94	22	92	90	66	28	34	60	56	24	90	98	98	82	84	34	60	0	63.5454545	62.8333333	64.4
Transfer Learning Regression: Predict as Raphael	42	37	43	29	46	20	47	2	20	7	23	4	16	27	1	23	6	50	45	38	8	38			
Transfer Learning Regression: Correct Percentage	84	74	86	58	92	40	94	96	60	86	54	92	68	46	98	54	88	100	90	76	16	76	74	73.8333333	74.2
Baseline Regression: Predict as Raphael	41	37	37	37	36	32	48	44	28	43	38	22	24	31	23	36	16	31	25	26	27	8			
Baseline Regression: Correct Percentage	82	74	74	74	72	64	96	12	44	14	24	56	52	38	54	28	68	62	50	52	54	16	52.7272727	64.1666667	39

Figure 7: Table of prediction for each painting.

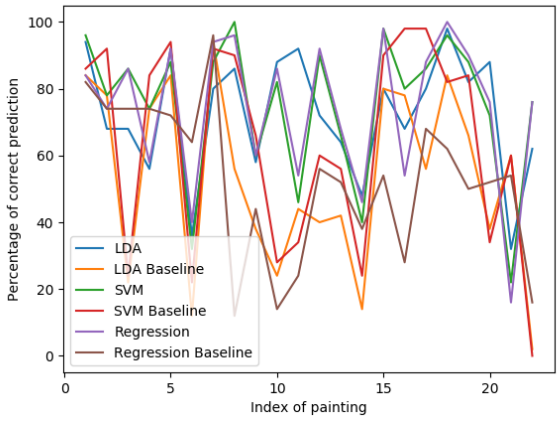

Figure 8: Percentage of correct prediction for each painting.

Moreover, Instead of calculating the four scores in terms of patches, I used the results of threshold voting as the evaluation score. For different thresholds, the accuracy are plotted in Figure 9. According to this result, I used as my final threshold of evaluating the authentication of paintings. It shows that when the threshold is 30 (out of 50 patches), i.e. images with 30 or more patches classified as Raphael's will be regarded as an authentic painting, all 3 models reached good predicition result.

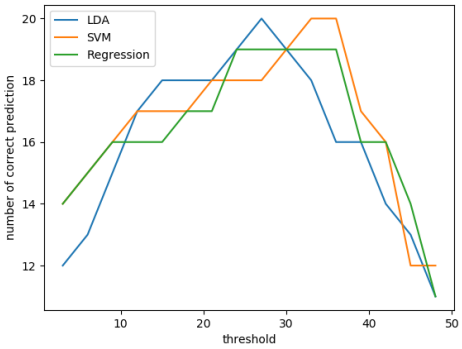

Figure 9: Number of correct prediction with different threshold.

5.3 Authentication of Raphael's Paintings

For the authentication of Raphael's paintings, I extracted 100 patches from each of the 6 painting, and the classification is based on voting. Because in previous section, it was showed by leave-one-out cross validation that 30 out of 50 patches (i.e. 60%) is a reasonable threshold for all models, here I just used this value as my threshold. That is, a painting with 60% or more patches classified as authentic will be labeled as authentic Raphael's painting.

The results of prediction are shown in the table below. According to the results, all 3 models have the same prediction (for SVM model, it got 59% on the painting 23, but it is so close to the threshold hence I just classified it as a true Raphael's painting): 1, 7, 23, 25 are Raphael's authentic paintings, while 10 and 26 are not.

Table 4: Test results of MNIST datasets

Painting	LDA	SVM	Logistic Regression	Final Result
1. Maybe Raphael	73%	81%	67%	Raphael
7. Maybe Raphael	61%	74%	79%	Raphael
10. Maybe Raphael	31%	28%	38%	Not Raphael
23. Maybe Raphael	62%	59%	74%	Raphael
25. Maybe Raphael	76%	82%	78%	Raphael
26. Maybe Raphael	46%	42%	42%	Not Raphael

6 Conclusion

To conclude, this project tried several models for transfer learning with the pre-trained VGG19 models. It turns out that LDA, Logistic regression and SVM all reached quite good results with the transfer learning feature vectors. I used the models to predict the authentication of Raphael's paintings and it was evaluated by majority voting.