

CSE 802 Pattern Recognition & Analysis

Spring 2017, Project Final Report

Deliang Yang
Department of Electrical and
Computer Engineering
A50836814
yangdeli@msu.edu

Mengying Sun
Department of Computer
Science and Engineering
A49087906
sunmeng2@msu.edu

Abstract—In this project, we propose a deep network model to perform unconstrained face recognition task. The performance of the model has been evaluated and compared with traditional methods.

I. INTRODUCTION

Face detection and recognition has been both attracting and challenging in modern society. It becomes possible through a wide variety of images available online and large database system, also achieves higher detection rate via rapidly developed techniques such as deep neural networks. However, it remains challenging due to large feature size and complex noise. More efforts are still to be devoted in pattern recognition community by improving algorithms and method for successful face detection.

In this paper, we propose a deep convolutional network frame to improve detection rate based on CASIA and LFW database [4]. Results demonstrate that the proposed network significantly improves human face recognition rate compared to other traditional methods. Verification rates of over 48% at false accept rate (FAR) of 0.1% are achieved by both PCA and LDA method.

II. PROJECT METHODS AND DESIGN

A. Workflow

The basic workflow is described as follows. First we train the deep network model with CASIA Webface Database [1]. The model includes a fully connected layer at the end of the network to classify the input images to one of the 10,575 classes. Then we feed the network with LFW database images, and use the model intermediate hidden layer output as features. Those features will be extract once again by PCA or LDA. Finally, the features from LFW will be evaluated via 10 trails cross validation in BLUFR protocol [6].

B. Deep Network Model Structure

There are 8 convolutional neural network (CNN) [5] layers in our model. The sequential structure and corresponding unit sizes are shown in Table I. These 8 CNN layers will be grouped into 4 larger layers. At the end of each large layers, ReLU (Rectifier Linear Unit) will be applied as activation function. To reduce the feature dimension, pooling layers are

TABLE I
DEEP NETWORK STRUCTURE IN THE PROJECT

Layer	Output dim	Parameters #
conv2d_1	$110 \times 110 \times 32$	896
conv2d_2	$108 \times 108 \times 64$	18496
activation_1 ReLU		
max_pooling2d_1	$54 \times 54 \times 64$	
conv2d_3	$52 \times 52 \times 64$	36928
conv2d_4	$50 \times 50 \times 128$	73856
activation_2 ReLU		
max_pooling2d_2	$25 \times 25 \times 128$	
conv2d_5	$23 \times 23 \times 128$	147584
conv2d_6	$21 \times 21 \times 256$	295168
activation_3 ReLU		
max_pooling2d_3	$10 \times 10 \times 256$	
conv2d_7	$8 \times 8 \times 256$	590080
conv2d_8	$6 \times 6 \times 512$	1180160
activation_4 ReLU		
average_pooling2d_1	$1 \times 1 \times 512$	
flatten_1	512	
dropout_1 (60%)	512	
dense_1	10575	5424975
activation_5 Softmax		

employed. The first three activation functions will be followed by MaxPooling layers with kernel size 2x2. The final pooling layer is AveragePooling, with kernel 6x6. After the average pooling, the feature dimension is 1x1 for each unit. The final pooling output is flattened and fed into a dropout layer to set random values with a proportion of 60% to 0, which prevents model from overfitting. The fully connected layers and Softmax activation will extract the output label of the input sample.

Compared to the baseline model provided in project guideline, the changes we made are as follows: 1) remove two convolutional layers because we encounter “not enough dimension error” when compiling the model; 2) slightly change the number of the filter size of each CNN layer.

TABLE II
COMPARISON OF THE BLUFR EVALUATION RESULTS OF THE DEEP NETWORK AND TRADITIONAL METHODS

	LE (Learning-based descriptors)		High Dimensional LBP		CNN Model	
Feature size	20736		127440		512	
Method	PCA	LDA	PCA	LDA	PCA	LDA
Verification @ FAR = 0.1%:	8.94%	20.95%	8.20%	30.86%	48.38%	48.46%
Open-set Identification @ Rank = 1	4.93%	13.39%	5.32%	17.46%	4.37%	5.18%

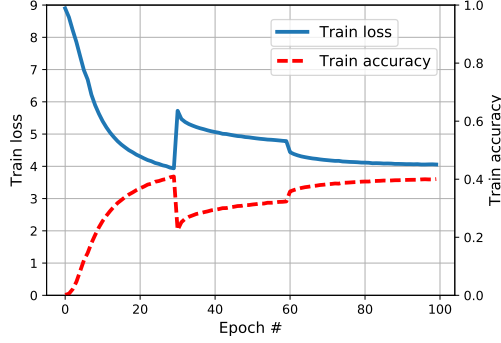


Fig. 1. Train loss and accuracy vs. number of epochs

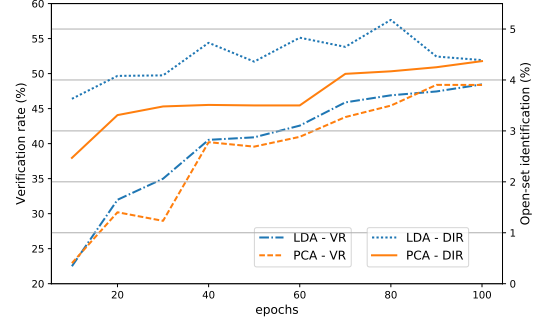


Fig. 2. Train loss and accuracy vs. number of epochs

C. Software Framework

The library we use to build the deep network is Keras [3], which employs the Google open source deep network library, Tensorflow [2] as backend.

In data preprocessing, the keras built-in class ImageDataGenerator is used to randomly selecting samples from the CASIA database, which will prevent the model from overfitting within a small set of recently trained classes. The image preprocessing class also support random rotation, horizontal shift, vertical shift, horizontal flip and so on, which actually generates larger number of training samples than the original database. It is believed this method will eventually improve the robustness and accuracy while applied to unconstrained face recognition task.

D. Hyper Parameters

In our project we use the following hyper parameters settings:

- Optimizer: AdaDelta [7]
- Learning rate: 1.0
- Parameter update decaying control factor ρ : 0.95
- Fuzzy factor ε : $1e-8$
- Batch size: 128
- Epoch: 100
- Loss: categorical cross entropy
- Random rotation degree limit: 20
- Random horizontal shift limit: 0.2
- Random vertical shift limit: 0.2

III. EVALUATION AND RESULT

The training process will be evaluated by training loss and accuracy. The test performance will be benchmarked by veri-

fication rate at FAR = 0.1% (VR) and open-set identification at Rank = 1 (DIR) in BLUFR protocol.

A. Training Performance

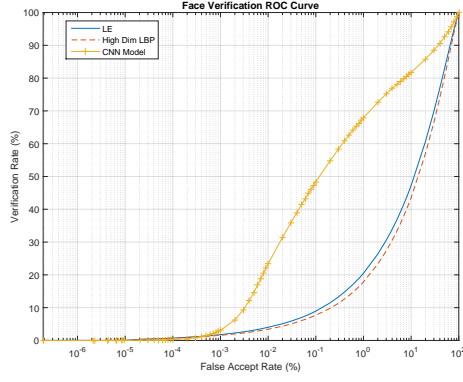
The model training loss and accuracy curve can be seen in Fig. 1. The first 30 epochs are trained without real time data augmentation during preprocessing. As we can see, with augmentation, training process slows down, and the loss increases, accuracy decreases. Because the model hasn't fit the augmented pictures before. Although the train accuracy is lower than 40% in the CASIA dataset, the actually verification rate in LFW can reach up to 48% at FAR 0.1% for both PCA and LDA feature extraction methods, which will be demonstrated in the following section.

The trend of VR and DIR varying with the training epochs is shown in Fig. 2. As we can notice, after introducing data augmentation at preprocessing, the VR increases about 10% for both PCA and LDA method from 30th epoch to 40th one. There is some boost in DIR under LDA method, however DIR under PCA remains almost the same.

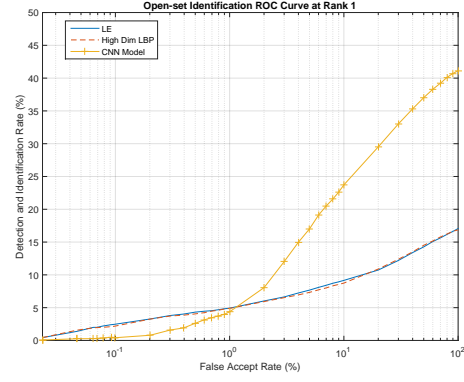
B. LFW Dataset Evaluation

Evaluations are taken on three different types of features via two approaches: i) Learning-based descriptors; ii) High dimensional LBP; iii) CNN extracted features each is evaluated through both PCA and LDA. High dimensional LBP has the largest feature size, which is 127,440, while CNN extracted features has least dimension, which is 512. The results can be found at Table II.

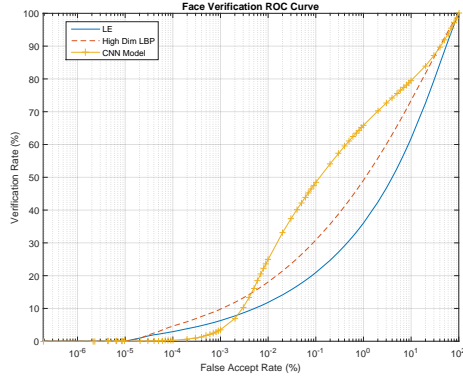
LDA outperforms PCA for all three features in the same feature setting, with high dimensional LBP serves the best



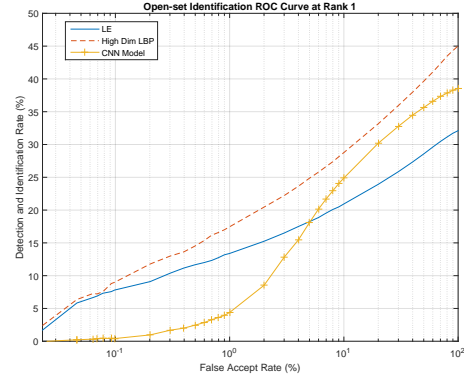
(a) Verification ROC curve with PCA feature extraction



(b) Open-set identification ROC curve at Rank 1 with PCA feature extraction



(c) Verification ROC curve with LDA feature extraction



(d) Open-set identification ROC curve at Rank 1 with LDA feature extraction

Fig. 3. ROC curve of different features and corresponding feature extraction methods.

prediction accuracy for open-set identification among all features, where 17.46% is achieved under LDA method. A highest verification rate of 48.46% is achieved by CNN extracted feature under LDA method. Feature extracted by our proposed CNN achieves open-set identification rate of only 4.37% and 5.18% at Rank 1 for PCA and LDA method respectively.

Figure 3 shows the ROC curve at different false accept rate for each feature under different methods. In verification with PCA, CNN model has highest values across all the FAR. In LDA method, LE and LBP perform better than CNN model at low FAR, which means CNN model may failed in restrict condition. For open-set identification in PCA method, CNN model also has lower value than LE and high dimensional LBP in low FAR range, but it increases dramatically as FAR increases. However in LDA method, the extracted feature from LBP has higher discriminative ability than CNN model across most of the FAR values. This indicate that discriminability of features extracted from CNN may not be augmented by LDA methods.

IV. TEAMWORK

Teamwork is appreciated along with the completion of the project. Basically the job is divided as follows:

- Mengying finishes the BLUFR protocol framework processing functions and run the PCA and LDA results on LE (learning-based descriptors) and high dimensional LBP (local binary patterns) feature extractor.
- Deliang builds the framework of deep network with Keras with tensorflow backend. He also writes the code that converts the raw LFW images to feature matrix.

Both of the two team members work together to run the code on HPCC successfully. We also collaborate to tune parameters and update the network layers.

Progress report, presentation and final report are also finished by teamwork.

V. CONCLUSION

In this project, we use CASIA database to train a deep network model that can extract features for unconstrained face recognition system. The CNN model achieves over 40% verification rate and 4.73% open-set identification rate under LDA method in BLUFR protocol and outperforms traditional method such as LE and LBP in many aspects. It is a promising model that can be applied to face recognition. We also want to point out that there are lot of technical tricks that can be done to further improve the performance.

REFERENCES

- [1] CASIA WebFace Database. <http://www.cbsr.ia.ac.cn/english/CASIA-WebFace-Database.html>, 2014. [Online; accessed 28-April-2017].
- [2] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. *arXiv preprint arXiv:1603.04467*, 2016.
- [3] F. Chollet. Keras, 2015.
- [4] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.
- [5] S. Lawrence, C. L. Giles, A. C. Tsoi, and A. D. Back. Face recognition: A convolutional neural-network approach. *IEEE transactions on neural networks*, 8(1):98–113, 1997.
- [6] D. Yi, Z. Lei, S. Liao, and S. Z. Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014.
- [7] M. D. Zeiler. Adadelata: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.