Random Forests for Face Recognition

Yeong Chee Chiew
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
TP068860@mail.apu.edu.my

Siew Yung Hong
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
TP060743@mail.apu.edu.my

Wong Yen Wei
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
TP063782@mail.apu.edu.my

Chai Cheng Ti
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
TP060723@mail.apu.edu.my

Zailan Arabee bin Abdul Salam
School of Computing
Asia Pacific University of Technology and Innovation (APU)
Kuala Lumpur, Malaysia
zailan@apu.edu.my

***Abstract*— This paper aims to unravel the face recognition issues with the execution of the Random Forest Algorithm. This paper presents a modified Random Forest algorithm to advance the proficiency and viability of understanding the acknowledgment issue. In this paper, a few parameters of the algorithm can be changed and compared to each other during the solving of the face recognition problem, which leads to a better result. The parameters that can be changed are epsilon and number tree, which directly affect the algorithm. Consequently, the best parameter combination was found for a face recognition issue with enhancement.**

***Keywords*— random forest, rf, face recognition, parameter, accuracy score**

I. INTRODUCTION

Face recognition has gradually become common. It may seem like a very easy task, but it needs some artificial intelligence algorithm to achieve the idea. So, in order to understand more deeply, we have chosen the Random Forest algorithm to implement the face recognition problem. Random Forest is a machine learning algorithm that is used for a variety of tasks including regression and classification. This method consists of a large number of small decision trees, called estimators in which each produces their own predictions. This allows the random forest model to combine the predictions of each estimator to create a much more accurate prediction. The study case that we will use that shows off the Random Forest Algorithm is where it will use Random Forest in face recognition on an Olivetti face dataset created in 1990.

As time passes, technology modernizes in conjunction with it. Artificial Intelligence has come a long way too, from when it was just a checkers program created in the 1950s, to now being integrated into better machines to create self-driving cars, facial recognition, and e-currency in the 20th century.

We will be diving more into facial recognition. Facial recognition was first introduced in the 1960s by Woody Bledsoe, Helen Chan Wolf and Charles Bisson. They were the first to ever use computers to recognize a face from a human. Even with the limited resources of technology back in the 60s, they were still able to make the A.I (Artificial Intelligence) recognize the various parts of the face like mouth, eyes among others by manually marking it by themselves. But as time went by, the advancement and accuracy of Facial Recognition was improved because of the modernization of the technology used.

Researchers have experimented with different processes and verification methods that include Random Forest Algorithm in two databases that are FERET and JAFFE (Mehri, 2017). During the part where he compares both Random Forest and Decision Tree to see which has the best accuracy in recognizing faces from both databases, it shows that Random Forest had more classification accuracy compared to Decision Tree. In the end they conclude that these algorithms will get a much better result if they were using a much better classifier like the Support Vector Machine (SVM).

Other researchers also conducted a similar project related to SVM (Kremic & Subasi, 2016). They wanted to test the performance of Random Forest and SVM in Face Recognition. As mentioned, Support Vector Machine (SVM) is a machine learning method that is used for classifying different faces. The database they used consisted of 20 individual photos of people that have different facial expressions. As tested, the SVM only had an accuracy of 93.20% but when combined with other classifiers for this instance Random Forest, the accuracy rate increased to 97.71%. In conclusion, they are able to piece together how this combination will be beneficial in improving facial recognition on mobile phones.

One more similar project that also focuses on facial recognition with Random Forest (Salhi, Kardouchi, & Belacel, n.d.), their main objective was to find out how to achieve a facial recognition system that will have a higher recognition rate, cheaper computational cost and a high efficiency on time. They combined the descriptor called Histograms of Oriented Gradients (HOG) with Random Forest and in the end the result they got was that the system had exceptionally good results in the accuracy section compared to other existing systems. In conclusion, even though the time taken to compute was lower as both descriptor and classifier are speedup, the accuracy will cost a bit of time as it is a bit slower so it could get a good accuracy result.

II. MATERIAL AND METHODS

This section will explain the details of how the materials are obtained and the methods used to solve the problem.

A. *Sources of Material*

First of all, the source code is from an open-source cloud storage GitHub, the source code is obtained from there in python code (Beyeler, 2017).

B. *Algorithm Implementation*

In this algorithm study, the Random Forest algorithm is used to solve the facial recognition problem. Random forest is an ensemble-learning algorithm that builds a set of many individual classifiers known as weak learners to form a unique classification system. Random Forest belongs to an ensemble method category that can respond on different combination of decision tree-type classifier, in the way that per tree that depends on the values of a random vector sampled independently and with the same distribution for all trees in the forest. RF can be seen as combination of two types of ensemble-method, boosting and bagging. In fact, it is built by randomly sampling a feature subset for each decision tree as boosting, and by randomly sampling a training data subset for each decision tree as in Bagging.

Ensemble learning algorithms produce a set of classifiers. In order to achieve a good result in accuracy score to our facial recognition problem, it is a necessary to test in the experiment with different number of trees, record the final results and compare them. There are a few advantages of using Random Forest, some of them are:

- Random Forest is a fast and straightforward learner.
- Random Forest can achieve very accurate classifier in various kinds of dataset given.
- Random Forest can handle a large amount of input variables from the users.
- Random Forest uses random selection, it allows the experiment to perform better especially in the case of many redundant features discrimination are involved.

III. RESULT AND DISCUSSION

This section will explain the hardware and software that were implemented in the experiment, and also explain what parameters were modified, which can bring the best result.

A. *Discussion on Implementation*

In this experiment, we will use a device with the Windows 10 operating system using AMD Ryzen 7 5700G with Radeon Graphics 3.80 GHz as the central processing unit and 32GB of RAM installed. Furthermore, Jupyter Notebook will be used as a software that processes the Random Forest algorithm. Before starting to train the algorithm, you need to simply preprocess the dataset.

Fig. 1. Sample of image were editted

The Olivetti Face dataset is the main dataset that will be used in this experiment. Specifically, the datasets are images, and we need to make sure that all images have the same mean grayscale level. We need to repeat this procedure for every image to make sure the feature values of every data point (that is, a row in X) are centered around zero.

After finish processing the image, we can start our training. We have a few parameters that need modifying.

Here we have two parameters that need to be modified, that is, number tree and epsilon. These two parameters have their own function. The function of a number tree is to have the maximum number of iterations. The function of an epsilon is the change in parameters at which the iterative algorithm stops.

For the size of the number tree, we will implement 50, 100, 300, 500, 1000, 1500 and 3000 number trees. This number affects the accuracy of score given by the system when predicting a new face.

At the same time, all the sizes of the numbers tree will be implemented with different epsilon. It will be 0.01, 0.1, and 1.

B. *Result*

Table I shows that epsilon 0.01 and the number of trees of 1000 generate the accuracy score of 94%, the highest accuracy score in this experiment; epsilon 1 and the number of trees of 50 generate the accuracy score of 25%, the lowest accuracy score in this experiment. As a result of the experiment, we know that increasing the epsilon or number of trees does not always result in the best accuracy score. We will explain our assumption about the number of trees first. When the number of trees increases, the accuracy score will also increase. However, after reaching a certain number of trees, the accuracy score will not change or even decrease. Besides that, epsilon can also not be set too high. This is because it will affect accuracy score too big or too small.

TABLE I. RESULT NUMBERS TREES WITH EPSILON

Number Tree / **Epsilon**	**50**	**100**		**300**	**500**	**1000**	**1500**	**3000**
0.01	91%	92%		94%	92%	94%	93%	93%
0.1	88%	89%		90%	90%	91%	92%	90%
1	25%	34%		36%	38%	38%	39%	40%

IV. CONCLUSION

An integrated framework for the detection, identification, and learning of faces using random forests is presented in our thesis. Although the decision criteria are randomized, random forests are fundamentally similar to traditional decision trees. In order to learn characteristics, random forests sample randomly chosen rectangular patches. We have a version of our framework in use that has been evaluated against cutting-edge methods.

The training time is a crucial aspect of random forests. The random tree grows quickly since only randomly chosen characteristics are used to sample them. The scale of the forest is an important factor. The classification accuracy and training duration improve as more trees are allowed to grow.

As per shown in Table I, the accuracy of face recognition does not only depend on the number of trees or the epsilon, the best result we get is 1000 trees with 0.01 epsilon. Therefore, we know that having the suitable set of setting is important for the recognition process.

REFERENCES

Mehri, A. (2017). Face Recognition Using Random Forest Classifiers Based on PCA, LDA and LBP Features (Master's thesis, Eastern Mediterranean University EMU-Doğu Akdeniz Üniversitesi (DAÜ)).

Kremic, E., & Subasi, A. (2016). Performance of random forest and SVM in face recognition. Int. Arab J. Inf. Technol., 13(2), 287-293.

Salhi, A. I., Kardouchi, M., & Belacel, N. (2012, September). Fast and efficient face recognition system using random forest and histograms of oriented gradients. In 2012 BIOSIG-Proceedings of the International Conference of Biometrics Special Interest Group (BIOSIG) (pp. 1-11). IEEE.

Beyeler, M. (2017, July 22). opencv-machine-learning/10.03-Using-Random-Forests-for-Face-Recognition.ipynb at master · mbeyeler/opencv-machine-learning. GitHub. https://github.com/mbeyeler/opencv-machine-learning/blob/master/notebooks/10.03-Using-Random-Forests-for-Face-Recognition.ipynb [accessed Aug. 1, 2022