2021 IEEE 3rd International Conference on Communications, Information System and Computer Engineering (CISCE 2021)

A KNN Classifier for Face Recognition

Xinyu Guo*
dept. of Mathematics Beijing Normal University (Zhuhai)
Zhuhai, China
1802030081@mail.bnuz.edu.cn

2021 International Conference on Communications, Information System and Computer Engineering (CISCE) | 978-1-6654-0352-8/21/$31.00 ©2021 IEEE | DOI: 10.1109/CISCE52179.2021.9445908

***Abstract*—Face Recognition has always been a hot topic, especially when the prevalence of Covid-19 calls for ways involving less physical contact in places where personnel identification is critical. However, although there are various algorithms for face recognition, there is limited research in determining the performance of these algorithms using scientific evidence. To address this issue, this paper evaluates the performance of K-Nearest Neighbors (KNN) for face recognition under different situations. To make the result of this study more applicable, this paper aims to train and test the model using photographs taken in profile and partially covered faces to simulate the situation in which the object needs to be identified does not face the camera at a right angle or wears masks. The experimental results demonstrate that K-Nearest Neighbors (KNN) achieved superior performance in recognizing uncovered frontal faces, with a success probability of 95.0%. Nevertheless, the model has a less satisfactory result when classifying profile or masked faces, and the corresponding success probability for the former is 22.2%, the latter 2.22%. It is worth remarking that the accuracy of KNN classifier when used in face recognition is 100% for uncovered frontal faces and 74.7% for covered ones.**

Keywords—face recognition, KNN, masks

I. INTRODUCTION

KNN, Principal component analysis (PCA), and neural networks have always been applied in the field of face recognition [1]. More advanced face recognition algorithms, including genetic algorithms [2] and convolutional neural networks [3], have been constantly utilized to achieve various functionalities in multiple domains, such as computer vision and security.

Traditional face recognition algorithms often face several problems. For instance, they are unlikely to identify people, once given the profile photos of the object to be recognized. Such algorithms also suffer from the low probability of recognizing faces that are partially covered. This paper aims to quantify these issues in KNN, using scientific and effective methodologies.

Furthermore, another active area in the field of face recognition is detecting and addressing races in algorithms. Researchers discussed possible factors leading to this issue [4] and proposed new algorithms to more accurately recognize faces of minorities, including those of Asians [5] and African Americans [6]. The KNN classifier is a traditional way for face recognition, but it has less racial bias compared to some of the modern algorithms, if trained properly. This paper utilized face images of Asians along with that of white people to achieve diversity and inclusivity in the field of machine learning and data science.

In addition to the background information of the question under investigation mentioned above, this section also provides an overview of the experiments conducted. When using the KNN classifier for face recognition, on the one hand, the model demonstrates good performance in identifying uncovered frontal faces, where all facial features are available. On the other hand, the algorithm suffers a significant probability decrease when detecting and recognizing faces that are partially covered or do not face the camera head-on, thus evoked the team's interest in testing the performance of the KNN classifier for recognizing faces. In order to improve the performance of a said model, photos with facial coverings, which are added manually using image editing tools, are introduced in the training dataset. In the previous experiments, five photos of a given person with all facial features identifiable are used as training materials. In order to carry out the experiment more effectively, a pre-experiment is conducted, aiming to determine the relationship between the number of photos in the training dataset and the probability of the model to identify the person correctly. The pre-experiment shows that three photos with facial coverings could slightly improve the performance of the KNN algorithm. Therefore, a new model trained on three photos of the subject need to be identified, based on which three trials are run to determine the probability of the KNN model in recognizing faces with nose and mouth covered. The result and evaluation will be discussed in section 3.

To summarize the aforementioned information related to this paper, the main contributions of this work can be summarized as follows:

1. The successful use of KNN classifier to identify a person whose face is in the model training dataset.
2. The construction of a dataset, which includes photos of faces with face coverings added manually, using a web crawler and image editing tools.
3. The training and test datasets use images of people from different races and diverse cultural backgrounds, including Asian and African.
4. The comparative analysis of the performance of the model when identifying frontal covered faces (with all facial features available), frontal uncovered faces, profile covered faces, and profile uncovered faces.
5. The comparative analysis of the ROC curves for frontal uncovered faces and frontal covered faces, respectively.
6. Discussion of the feasibility of applying KNN classifier to identify fugitives whose faces have been recorded.

978-0-7381-1215-2/21/$31.00 ©2021 IEEE

The rest of this paper is organized in the following way. In Sect. 2, the methodologies used in this paper are presented, and the KNN classifier is discussed in detail. The dataset, evaluation metrics, and the corresponding experimental results are introduced and analyzed in Sect. 3. Finally, Sect. 4 presents a proposal of a potentially feasible application of the model and concludes this paper.

II. Model Formulation

K-Nearest Neighbors (KNN) classification is one of the most fundamental and simple classification methods and should be one of the first choices for a classification study when there is little or no prior knowledge about the distribution of the data. Therefore, we evaluate the performance of K-Nearest Neighbors (KNN) for face recognition under different situations.

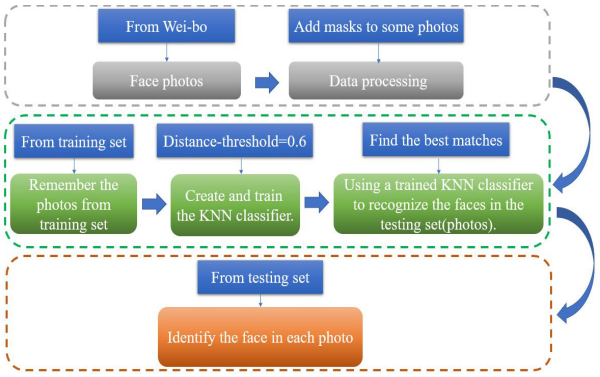

Figure.1 Flow chart of the model

A. Introduction to the KNN Model

1) Principle of model

It is assumed that the red, green and blue circles are distributed in a two-dimensional space where the same color represents the same category. The KNN algorithm will calculate the distance between the hollow circle and all training samples. Then, the KNN classifier selects the smallest distance of k samples (set k = 4 here) which are connected with the hollow circle by thick black lines in Fig. 2. Clearly, the hollow circle is presumed to be in the red category since these four samples are all in the red category.

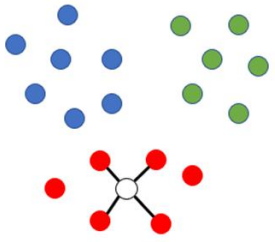

Figure.2 KNN interpretation chart1

If the hollow circle is in the middle of the location (as shown in Fig. 3), the KNN algorithm also selects the most nearby four samples which include three categories (1 red, 1 blue, 2 green). According to such a situation, the KNN algorithm usually uses a voting method to predict the category, finding a category with the most frequent occurrences among the categories of those four samples. Therefore, the hollow circle is presumed to belong to the green category.

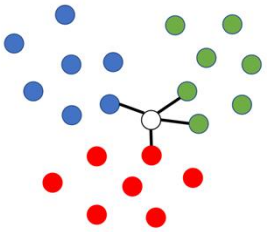

Figure.3 KNN Intrepretation chart 2

2) Introduction to KNN algorithm

(1)Calculate the distance between the sample to be classified and all the training samples.

In KNN, the distance between different samples can be calculated by the non-similarity measure to avoid the matching problem between samples. Manhattan distance or Euclidean distance is normally adopted to measure the similarity of different neighbors, where Euclidean distance is a special case of Minkowski distance p=2 and Manhattan distance is a special case of Minkowski distance p=1.

$$D_{(x,y)} = (\sum_{k=1}^{K}|x_k - y_k|)^p)^{\frac{1}{p}} \quad (1)$$

European distance and Manhattan distance:

$$D_{(x,y)} = \sqrt{\sum_{k=1}^{K}(x_k - y_k)^2} \quad (2)$$

$$D_{(x,y)} = \sum_{k=1}^{K}|x_k - y_k| \quad (3)$$

When p→∞, the equation above can be rewritten as the following form:

$$D_{(x,y)} = max|x_i - y_i|(i = 1,2,3,\dots,n) \quad (4)$$

(2) Sort distances in order from smallest to largest.

(3) Select K training samples with the smallest distance.

If the K value is small, it is equivalent to predict the sample in the smaller neighborhood, therefore the training error will be small, while the generalization error (the testing error) will increase at the same time. Thus, the decrease of K value means/indicates that the model may become complex and prone to over-fitting. If the K value is greater, the generalization error will also decrease. On the contrary, the training error will increase, which seems to show that the model may become simple and the fitting effect may become poor.

(4) Count the occurrence probability of categories of the K training samples.

Using the weighted algorithm to calculate the probability of the sample, the closer the sample is to the testing sample, the higher proportion of the sample should be.

(5) The category with the highest probability of occurrence among the K samples can be/is taken as the category of the testing samples.

B. KNN algorithm and face recognition

KNN classifier is first trained on a set of labeled (known) faces and can then predict the person in an unknown image by finding the K most similar faces (images with closet face-features under Euclidean distance) in its training set and performing a majority vote (weighted) on their label.

We set six operating situations with different training sets and testing sets, then calculate the accuracy of KNN face recognition under the six situations.

TABLE I TRAIN AND TEST CONDITIONS

Train	Test
frontal face without mask	frontal face without mask
	frontal face with mask
	side face without mask
	side face with mask
frontal face with mask	frontal face with mask
	side face with mask

1) Data processing

The selected samples are all photos of the user "1927305954", "3669102477" and "6269329742" from www.weibo.cn. Then we delete the non-face photos, keep the valid frontal and side face photos, and set masks on the needed photos. In the training set, there are five training photos for a frontal face without a mask and a frontal face with a mask, while in the testing set in Table1, all the situations are including 15 photos. To further boost the performance, we add other 60 testing photos, in the case of the frontal face without a mask in the training and testing set.

2) Modeling

(1) Firstly，Let the model remember photos from a training set.
View in source code to see train_dir example tree structure:

```
<train_dir>/
├── <person1>/
│   ├── <somename1>.jpeg
│   ├── <somename2>.jpeg
│   └── ...
├── <person2>/
│   ├── <somename1>.jpeg
│   └── <somename2>.jpeg
└── ...
```

Follow the structure, Let the model loop through each training image for each person. The model adds the face encoding for a current image in the training set when there exist no people (or too many people) in a training image

(2) The modeling and training of the KNN classifier
We should determine how many neighbors to use for weighting in the KNN classifier and set distance_threshold (distance threshold for face classification) equal to 0.6 in this model. After that, we can save the trained KNN classifier in the model.

(3) Using a trained KNN classifier to recognize the faces in the testing set.
Firstly, we load a photo file and find face locations, if no faces are found in the image, return an empty result. Then, the KNN classifier finds encodings for faces in the test image and uses the KNN model to find the best matches for the testing face. Finally, the KNN classifier predicts classes and remove classifications that aren't within the threshold.

(4) We draw a box around the face using the Pillow module and a label with a name below the face.

III. EXPERIMENT RESULT, DISCUSSION, AND EVALUATION

A. Datasets

TABLE II DATASETS UTILIZED IN THE EXPERIEMNT

Dataset	Number	Person number
frontal face with mask	20	1927305954
frontal face without mask	80	
side face with mask	15	
side face without mask	15	
frontal face with mask	20	6269329742
frontal face without mask	80	
side face with mask	15	
side face without mask	15	
frontal face with mask	20	3669102477
frontal face without mask	80	
side face with mask	15	
side face without mask	15	

All experiments are conducted on a computer with a Intel(R) Core (TM) i5-8250U CPU @ 1.60GHz 1.80 GHz operating system. The program codes of data preprocessing and graphs modeling are written by Python 3.7.2.

B. Result

TABLE III PROBABILITY OF KNN IDENTIFYING CORRECTLY (TRAINED ON UUNCOVERED FRONTAL FACES)

Frontal faces	Facial Coverings	Person 1	Person 2	Person 3	Average
Yes	No	96.7%	93.3%	95.0%	95.0%
Yes	Yes	33.3%	33.3%	33.3%	33.3%
No	No	20.0%	20.0%	26.7%	22.2%
No	Yes	0.00%	6.67%	0.00%	2.22%

Table 3 above shows the probability of the KNN classifier identifying the right person when trained on five photos uncovered frontal faces. The probability of correctly classifying frontal faces with no facial coverings is tested in a sample of 60 photos for each individual, and the other three types samples of 15 images.

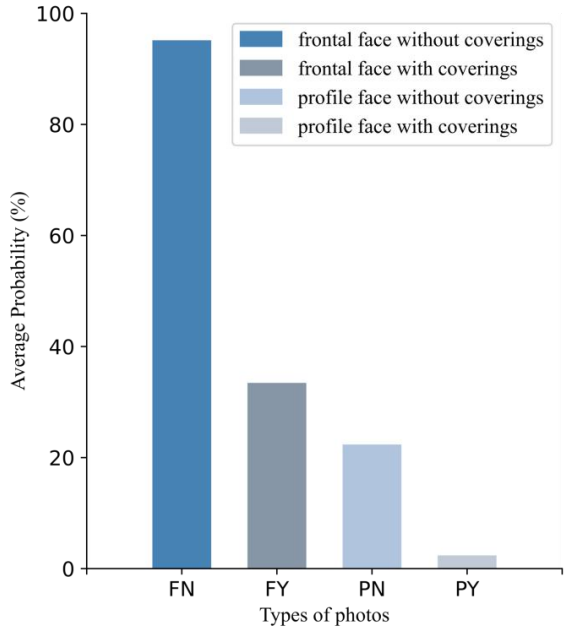

Figure.4 Probability of KNN Model on correctly identifying faces with different facial features

Figure 4 demonstrates that KNN achieves the best performance when identifying frontal faces with no coverings (FN), shown by an average probability of 95%. It is less successful when it comes to frontal faces with coverings (FY) and performs more poorly when identifying profile faces without coverings (PN). Furthermore, the model has merely a 2.22% probability of producing correct results when recognizing profile face with coverings (PY).

In order to improve the performance of the algorithm, training samples have been changed to faces that are partially covered with manually added coverings. Before the experiment, it is noticed that the size of the training set will affect the running time of the program. Therefore, it is ideal to determine the relationship between the number of photos in the training set and the probability of the KNN model.

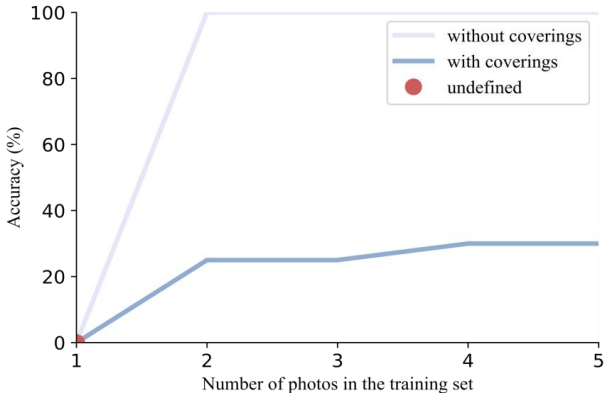

Figure.5 The relationship between number of photos and success probability

The figure 5 shows that when the model is trained on photos with coverings, the number of photos in the training set could be set to either 4 or 5, since each number gives the same probability. Therefore, the following experiment uses 4 training photos for each trial.

TABLE IV SUCCESS PROBABILITY OF KNN TRAINED ON PARTIALLY COVERED FRONTAL FACES

Frontal faces	Facial Coverings	Person 1	Person 2	Person 3	Average
Yes	Yes	46.7%	33.3%	46.7%	42.2%
No	Yes	0.00%	0.00%	6.67%	2.22%

Table 4 demonstrates the accuracy of KNN model trained on 4 photos of covered faces. The test set includes 15 photos of either frontal covered or profile covered faces.

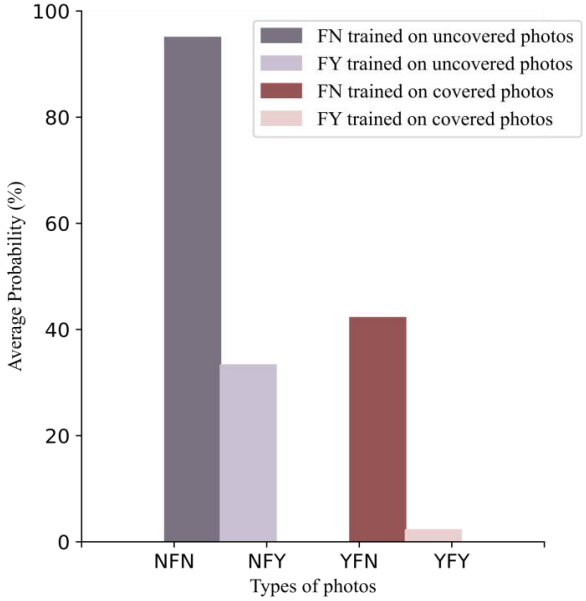

Figure.6 Comparison of success probability of the model trained on photos of covered and uncovered faces

According to figure 6, the KNN model experiences a significant decrease in average probability when trained on

covered photos. This might be because of the limited facial features available when identifying faces.

C. Evaluation Metrics

1) False Acceptance Rate (FAR) and False Rejection Rate (FRR)

TABLE V FAR AND FRR OF KNN (TRAINED ON UNCOVERED FRONTAL FACES)

Frontal faces	Facial Coverings	FAR	FRR
Yes	No	0.00%	0.09%
Yes	Yes	0.00%	4.76%
No	No	0.00%	5.56%
No	Yes	0.00%	6.98%

Table 5 shows that KNN did not falsely recognize people when trained on uncovered frontal faces, but it could falsely reject people.

TABLE IV FAR AND FRR OF KNN (TRAINED ON COVERED FRONTAL FACES)

Frontal faces	Facial Coverings	FAR	FRR
Yes	No	0.00%	4.13%
Yes	Yes	0.00%	6.98%

Tables 5 and 6 show that the KNN algorithm has relatively high FRR when identifying faces without all the facial features available, whether trained on uncovered or covered faces.

2) Confusion Matrix

TABLE VII CONFUSION MATRIC OF UNCOVERED FRONTAL FACES (TRAINED ON UNCOVERED FRONTAL FACES)

	Positive	Negative
Positive	TP: 1	FP: 0
Negative	FN: 0	TN: 1

TABLE VIII CONFUSION MATRIC OF COVERED FRONTAL FACES (TRAINED ON COVERED FRONTAL FACES)

	Positive	Negative
Positive	TP: 0.333	FP: 0.022
Negative	FN: 0.022	TN: 0.978

Table 8 shows that KNN has a high "true negative (TN)" value, i.e., it has a high success rate when identifying photos that do not belong to a certain subject. On the other hand, it performs poorly in recognizing the person whose photo is in the training set.

3) Accuracy (ACC)

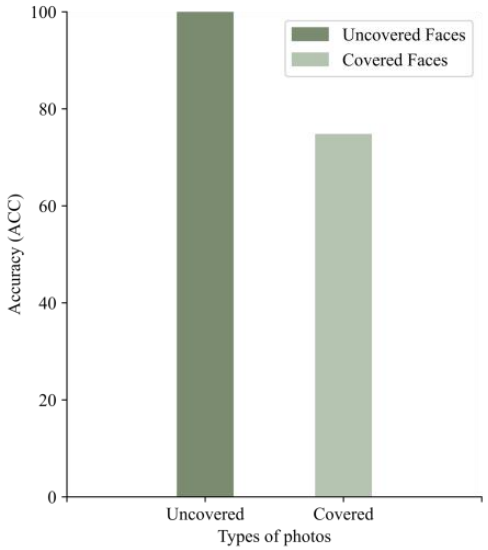

Figure.7 Comparison of accuracy of the model on photos of covered and uncovered faces

Figure 7 shows that there is a significant loss in accuracy when the model is used to identify covered faces.

IV. CONCLUSION

This paper tests the performance of KNN classifier in recognizing faces with and without coverings. In Conclusion, the experimental results demonstrate that the KNN classifier is a satisfactory tool when identifying uncovered frontal faces. Nonetheless, it fails to achieve acceptable performance when face coverings or profile photos are taken into consideration. Furthermore, when images with limited facial features, such as partially covered and profile photos, are included in the training set, the corresponding model demonstrates a slight increase in the probability of identifying the object correctly. This paper introduced the usage of this algorithm in the field of security.

In the future, besides possible usage in security, the KNN face recognition algorithm has the potential to be applied in other areas, where only photos of uncovered frontal faces are considered. Moreover, when recognizing faces with coverings like masks with machine learning algorithms, facial features above the nose, such as eyes and foreheads, should take up more weight. By the same token, photos with different facial coverings, including sunglasses and baseball caps, should be processed by distinct face recognition algorithms.

REFERENCES

[1] Ali, W., Tian, W., Din, S.U. et al. Classical and modern face recognition approaches: a complete review. Multimed Tools Appl 80, pp. 4825–4880, 2021.

[2] Z. Li-Hong, L. Fei and W. Yong-Jun, "Face recognition based on LBP and genetic algorithm," 2016 Chinese Control and Decision Conference (CCDC), Yinchuan, pp. 1582-1587, 2016.

[3] C. Ding and D. Tao, "Trunk-Branch Ensemble Convolutional Neural Networks for Video-Based Face Recognition," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 40, no. 4, pp. 1002-1014, 1 April 2018.

[4] J. G. Cavazos, P. J. Phillips, C. D. Castillo and A. J. O'Toole, "Accuracy comparison across face recognition algorithms: Where are we on measuring race bias?," IEEE Transactions on Biometrics, Behavior, and Identity Science, 2020.

[5] Bon-Woo Hwang, Myung-Cheol Roh and Seong-Whan Lee, "Performance evaluation of face recognition algorithms on Asian face database," Sixth IEEE International Conference on Automatic Face and Gesture Recognition, 2004. Proceedings, Seoul, South Korea, pp. 278-283, 2004.

[6] A. Z. Abd Aziz and H. Wei, "Polarization Imaging for Face Spoofing Detection: Identification of Black Ethnical Group," 2018 International Conference on Computational Approach in Smart Systems Design and Applications (ICASSDA), Kuching, pp. 1-6, 2018.

[7] Warda M. Shaban et al, "A new COVID-19 Patients Detection Strategy (CPDS) based on hybrid feature selection and enhanced KNN classifier", 2020.

[8] N. Mukahar and B. A. Rosdi, "Performance Comparison of Prototype Selection Based on Edition Search for Nearest Neighbor Classification," Proceedings of 2018 7th International Conference on Software and Computer Applications(ICSCA), pp.144-147, 2018.

[9] G. Zhang, Zhongshuai Zhao and F. Chen, "Classification algorithm based on multiple hyper-spheres classifier and KNN method," Proceedings of 2014 International Conference on Industrial Electronics and Engineering(ICIEE), pp.415-422, 2014.

[10] H. Jiang and Wenqiang Li, "Feedback learning classifier based on TFIDF," Proceedings of 2011 13th IEEE Joint International Computer Science and Information Technology Conference(JICSIT 2011), vol. 01, pp. 312-315, 2011.

[11] S. Dong and X. Wang, "Research on Network Intrusion Data Based on KNN and Feature Extraction Algorithm," Abstracts of the 4th International Conference of Pioneering Computer Scientists,Engineers and Educators(ICPCSEE), Springer 2018.

[12] B. Al-Helali, Qi Chen, Bing Xue, Mengjie Zhang, "A new imputation method based on genetic programming and weighted KNN for symbolic regression with incomplete data," pp.1-20,2021.