Real-Time Masked Face Recognition Using FaceNet and Supervised Machine Learning

By Faisal Dharma Adhinata

Real-Time Masked Face Recognition Using FaceNet and Supervised Machine Learning

Faisal Dharma Adhinata, Nia Annisa Ferani Tanjung, Widi Widayat, Gracia Rizka Pasfica, and Fadlan Raka Satura

Abstract The coronavirus pandemic has led to the implementation of health protocols such as the use of masks worldwide. Without exception, work activities also require the wearing of masks. This condition makes it difficult to recognize an individual's identity because the mask covers half of the face, especially when the employee is present. The attendance system recognizes a face without a mask more accurately, in contrast, a masked face makes identity recognition inaccurate. Therefore, this study proposes a combination of facial feature extraction using FaceNet and several classification methods. The three supervised machine learning methods were evaluated, namely multiclass Support Vector Machine (SVM), K-Nearest Neighbor, and Random Forest. Furthermore, the masked face recognition system was evaluated using real-time video data to assess the accuracy and processing time of the video frame. The accuracy result on real-time video data using a combination of FaceNet with K-NN, multiclass SVM, or Random Forest of 96.03%, 96.15%, and 54.04% are obtained respectively and in processing time per frame of 0.056 s, 0.055 s, and 0.061 are obtained respectively. The results show that the combination of the FaceNet feature extraction method with multiclass SVM produces the best accuracy and data processing speed. In other words, this combination can reach 18 fps at real-time video processing. Based on these results, the proposed combined method is suitable for real-time masked face recognition. This study provides an overview of the masked face recognition method so that it can be a reference for the contactless attendance system in this pandemic era.

Keywords Coronavirus pandemic · FaceNet · Masked face recognition · Multiclass SVM · Real-time

F. D. Adhinata (✉) · N. A. F. Tanjung · G. R. Pasfica · F. R. Satura
Department of Software Engineering, Faculty of Informatics, Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia
e-mail: faisal@ittelkom-pwt.ac.id

W. Widayat
Department of Information Engineering, Faculty of Informatics, Institut Teknologi Telkom Purwokerto, Purwokerto, Indonesia

T. Triwiyanto et al. (eds.), *Proceedings of the 2nd International Conference on Electronics, Biomedical Engineering, and Health Informatics*, Lecture Notes in Electrical Engineering 898, https://doi.org/10.1007/978-981-19-1804-9_15

1 Introduction

The coronavirus pandemic has led to the implementation of health protocols worldwide, such as the use of masks. This is important to suppress the escalation of the virus which spreads through direct contact with sufferers or indirectly by touching infected surface areas [1]. Furthermore, in the workplace, attendance activities are usually carried out using fingerprints. This condition promotes the spread of coronavirus via indirect contact. Meanwhile, one of the safe attendance systems is the use of facial biometric data [2, 3]. The human face generally consists of two eyes, as well as the area above and below the nose [4]. An individual is usually recognized through facial features such as shape, hair, nose, mouth, eyes, and eyebrows. However, the use of masks makes the area under the nose covered, therefore, special techniques are needed for masked face recognition at segmentation, facial feature extraction, and classification stages.

Masked face recognition has recently become a new case study due to the difficulties associated with identifying faces covered with masks [5]. Aswal et al. [6], using a combination of RetinaFace and VGGFace2 for the masked face identification process, produced an accuracy of 94.5% but failed to run on real-time data. Therefore, a variety of other methods is needed to accurately run in real-time. The first stage of masked face recognition is segmentation for face detection. This process captures the entire face area from hair to chin, meanwhile, various types of mask motifs used need to be ignored, thereby limiting the segmentation process of the facial area to the upper part of the nose. Therefore, the facial features used in this study are the area above the nose which includes hair, eyes, and eyebrows.

The next step is the main stage of masked face recognition, namely feature extraction, and classification of identities based on facial features. One technique that is often used to extract facial image features is the deep convolutional neural network [7]. FaceNet uses a deep convolutional neural network to extract facial features, It maps each face image to a Euclidean space in which the distances between faces are proportional to the similarity [8]. Furthermore, several studies have used FaceNet for facial data classification [9–11]. Zhao et al. [9] used this technique for recognition at low resolution and obtained an accuracy value of 100% on the identities of 21 people. Also, Pranoto and Kusumawardani [10] used FaceNet for a student attendance system and produced an accuracy above 95%. This technique is not only used for identity recognition, but also to recognize facial expressions [11]. A previous study produced an accuracy of 94.68% and recognized three facial expressions, namely focused, unfocused, and fatigue. Therefore, this study aims to use FaceNet for facial feature extraction. The key to the accuracy of masked face recognition is the classification stage. Several supervised machine learning techniques have been used in masked face recognition studies. The Random Forest was used to classify 40 people's identities with an accuracy of 97.17% [12]. Furthermore, the K-Nearest Neighbor produced an accuracy of 81% with a value of K = 1 [13]. The multiclass SVM was also applied to face recognition systems and produced an accuracy of 86.76% [14]. Therefore, this study analyzed the combination of FaceNet feature extraction with K-Nearest

Neighbor, multiclass SVM, or Random Forest methods. In addition, the processing speed of real-time video data and its accuracy were also evaluated.

This paper is organized as follows. In Sect. 2, the design of the proposed system contains a flowchart and an explanation of the methods. Section 3 contains experiments at the training stage, which are used for testing using real-time video data, and ends with a discussion related to previous research. Section 4 contains conclusions from the results of masked face recognition research.

2 Materials and Method

The masked face recognition system consists of training and testing stages using real-time video. Figure 1 shows the proposed system architecture. Figure 1 shows that the training phase begins with the acquisition of facial data and the respective identity. Furthermore, the facial data used is a cropped face in the area above the nose, the face area was then resized to match the size of the pre-trained FaceNet model. The pre-processing stage results were stored in an array form and further

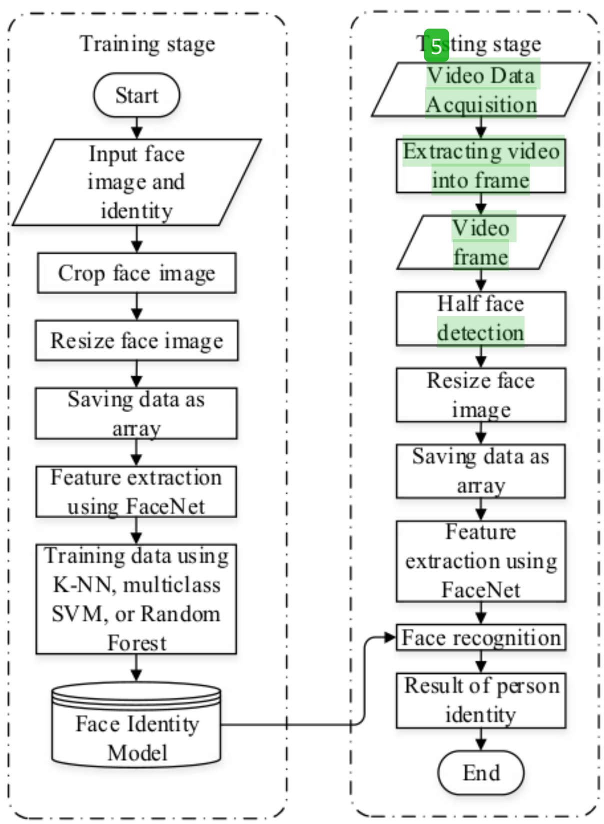


Fig. 1 The proposed system of real-time masked face recognition

Fig. 2 Example of training data and its identity

extracted using FaceNet's facial features. The facial feature extraction results were trained in the form of a model using several supervised machine learning methods, namely K-NN, multiclass SVM, or Random Forest. Moreover, the parameters of each supervised learning method were then evaluated to obtain the best value for training and testing accuracy. The best facial identity model results were stored for use in the testing phase with real-time video data.

In the testing stage, real-time video data were used to test the speed of the proposed system in processing each video frame. The video data was first extracted into video frames, while the frame obtained was segmented to obtain the face area above the nose. Furthermore, the obtained facial area was also resized for feature extraction processing and the results were stored in an array. The testing phase feature extraction also used FaceNet, while the results were matched with the facial identity model. The output of this system is a face bounding box that contains the identity of the person, meanwhile, the accuracy and video data processing speed in real-time were evaluated.

2.1 *Data Acquisition*

The masked face recognition system uses an image as training data and real-time video as testing data. The training data used include 24 identities, with each identity totaling 10 images, 8 as training, and 2 as testing data. Meanwhile, the subject data used in this study varied in age, ranging from 3 to 41 years old. This age variation is such that the research results apply to all ages. Furthermore, for real-time video data, a 2 MP camera with 15 fps was used. The faces used were all towards the front of the camera. Figure 2 shows an example of training data used in this study.

2.2 *Data Segmentation*

The facial segmentation results contained masks with various motifs, therefore, the detected face was cropped only on the face, namely the area above the nose. Figure 3

Fig. 3 Face segmentation results

shows an example of cropped training data. Furthermore, the segmented facial data were then resized to 160 × 160. This is to adjust the size of the pre-trained FaceNet model at the feature extraction stage. The resize results were stored in the form of an array and processed in the feature extraction stage.

2.3 *Feature Extraction Using FaceNet*

FaceNet is a face recognition system developed in 2015 by Google researchers which obtained leading results based on a variety of benchmark datasets [8]. It achieved state-of-the-art accuracy by combining deep convolutional networks and triplet loss. Figure 4 shows the feature extraction of the FaceNet pre-trained model using the Triplet Loss function. Furthermore, FaceNet has a significant advantage over previous systems because it learns the mapping from the photos and generates embeddings without relying on a bottleneck layer for recognition or verification. Following the creation of the embeddings, all subsequent actions such as recognition or verification are carried out using domain-specific standard methods, with the newly formed embeddings serving as the feature vector. The network is trained such that the squared L2 distance between embeddings is proportional to the degree of similarity between faces. In addition, scaled and altered images are used for training as well as severely cropped images around the facial area.

The triplet loss function is used to encode the faces of three images namely an anchor, a positive, and a negative image. It is believed that vectors with the same identity get increasingly similar (have a lower distance), meanwhile, vectors

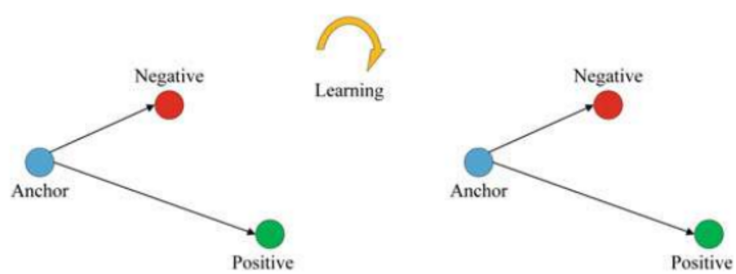


Fig. 4 The triplet loss function [6]

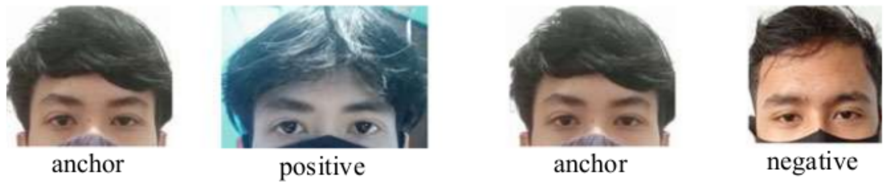


Fig. 5 The illustration of triplet loss function

with different identities become less similar (have a more significant distance) [15]. Figure 5 shows that the anchor and positive images belong to the same person, while the negative image belongs to another individual. The focus on training a model to generate embeddings directly, rather than via an intermediate layer was significantly emphasized in this study.

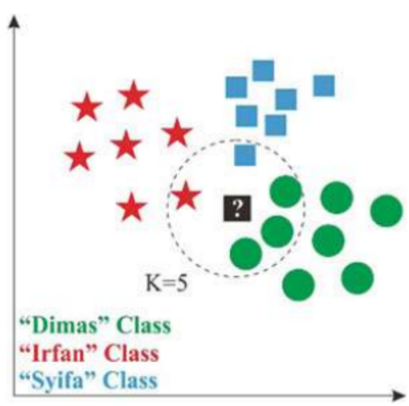
2.4 *Classification Using Supervised Machine Learning*

Supervised Learning is a process of grouping data based on a label. In this study, the image data were grouped based on the respective identity. The label used is the identity of each person, meanwhile, the classification stage used the three classifier methods, namely K-Nearest Neighbor, SVM multiclass, and Random Forest.

K-Nearest Neighbor

The K-Nearest Neighbor (KNN) method of data classification is utilized for face recognition [16, 17]. Each pixel in the face conveys a different piece of information, therefore, face identity was detected based on each pixel classification. The face was selected by the most common class. Figure 6 shows the illustration of K-NN in this study.

Fig. 6 The illustration of K-NN



The training data consist of vectors in a multidimensional feature space, each data with a label. In the training stage, the feature vectors and class labels associated with the training samples are stored. Meanwhile, in the classification stage, K is a user-defined, and an unlabeled vector (test face image) and is classified by assigning the label that appears most frequently among the K training samples closest to that particular test face. A test image is recognized by connecting it with the label of the nearest face in the training set, allowing for the calculation of the distance between both points [18]. The most common search technique is via the use of the Euclidean and Manhattan distance formulas demonstrated in Eqs. 1 and 2.

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{1}$$

$$d(x, y) = \sum_{i=1}^{n} |x_i - y_i| \tag{2}$$

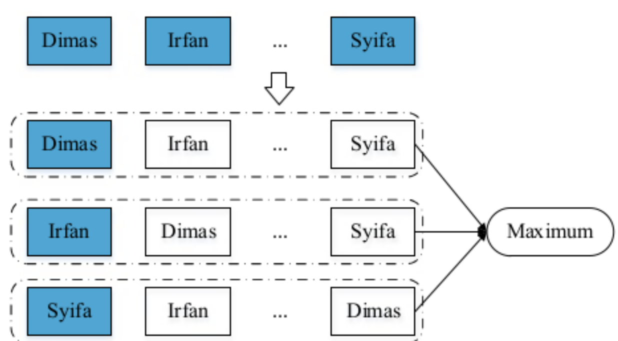
In this study, two distance algorithms were tested, namely Euclidean and Manhattan distance. Furthermore, the value of K in each distance algorithm was also evaluated.

Multiclass SVM

Support Vector Machines (SVM) [19] are a type of machine learning technique invented by Vapnik and colleagues [20]. The earliest methods of implementing SVM to solve multiclass classification problems are the One-vs-Rest (OvR) SVM algorithms [21]. In the OvR stage, SVM was trained for each class k that is an expert at classifying k (one) versus non-k (the rest), hence, a binary classifier was created for each k class. Figure 7 shows the illustration of multiclass SVM in this study.

The model formed was indicated with the blue color as the positive class, while the white color is negative. Furthermore, the maximum value of each class comparison is calculated in the prediction process. The inputs for the model formation are 24 class labels (person identities) and image features on each label, meanwhile, experiments

Fig. 7 The illustration of SVM multiclass



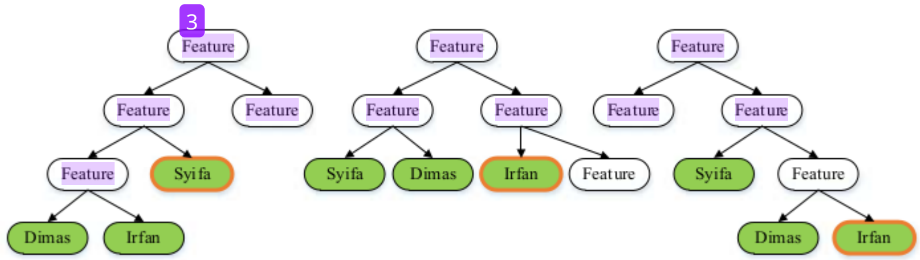


Fig. 8 The illustration of random forest

were carried out using three multiclass SVM kernels, namely sigmoid, polynomial, and linear.

Random Forest

Random forest [22] is a widely known technique for developing predictive models for classification and regulation [23]. The random forest algorithm is used to construct a randomized decision tree. Furthermore, this algorithm frequently produces excellent predictors for each iteration. This method aims to create numerous predictors before aggregating the diverse predictions rather than acquiring an optimum technique at once. The features were then used to categorize or regress a sample of qualitative and/or quantitative variables. Figure 8 illustrates the Random Forest categorization results.

Random forest classification is accomplished using training sample data and then creating a tree. The classification is based on the most decisions observed from the constructed trees. Meanwhile, the construction of trees uses randomly chosen variables and then classifies all the trees produced [24]. For instance, a Random Forest contains three decision trees, when the classification results from two trees are "Irfan" and one tree is "Syifa", then the most votes are used, namely "Irfan" class. In the classification stage using the random forest method, the criterion parameter and the number of trees were evaluated.

2.5 System Evaluation

The accuracy value at the training and testing stages was measured as well as data processing speed using real-time video. The Python language library, scikit-learn was used to measure the accuracy of the training phase, meanwhile, the metric used was accuracy_score. This value was then used to calculate the correct prediction of the system. Equation 3 shows the accuracy formula.

$$Accuracy\left(y, y'\right) = \frac{1}{n_{samples}} \sum_{i=0}^{n_{samples}-1} 1(y_i' = y_i) \tag{3}$$

where y_i' is the predicted value of the i-th sample and y_i is the actual value, then the fraction of true predictions is $n_{samples}$. To determine the accuracy value at the testing stage, the correct rate (CR) value was calculated by dividing the number of correct video frame data (C) by the total number of frames used for testing (A). The correct rate equation is shown in Formula 4 [25].

$$CR = \frac{C}{A} \tag{4}$$

Meanwhile, the computer configuration used to run the program significantly affects the speed of real-time video data processing. Therefore, a desktop computer configuration with an Intel Core i3-9100F CPU @ 3.60 GHz and 8192 MB DDR4 RAM was used to train face data. The operating system is Windows 10 Pro 64-bit. In addition, a 2.0 MP camera with 15 fps was also used.

3 Result and Discussion

The masked face recognition system uses the accuracy and speed of video processing to measure the system's robustness. This study has several experiments at the face data training stage using the K-NN, SVM multiclass, and Random Forest methods. Then the best results from each training process will be used for testing on real-time video data.

3.1 Training Face Data Using K-NN

The masked face identities were classified via the K-NN method by calculating the closest distance using the Manhattan and Euclidean distance algorithms. The result was used to compare the similarity of the test and training feature database. Figure 9 shows the experimental results of variations in the value of K for each K-NN distance algorithm.

The experiment on the K-NN method used an odd K value because the number of classes studied was even, namely 24. Furthermore, the results showed that the K value had a significant effect on training and testing accuracy. The best results were obtained using the value of K = 1, meaning that only one nearest neighbor is needed to classify people's identities. The image used as input for training data makes the features obtained very close to each identity. Therefore, higher K values decrease accuracy as shown in Fig. 9. Moreover, based on the variation of the distance algorithm, the use of Manhattan and Euclidean distance does not significantly affect the accuracy results. Therefore, the value of K = 1 with the Euclidean distance algorithm was used as the test in real-time video data.

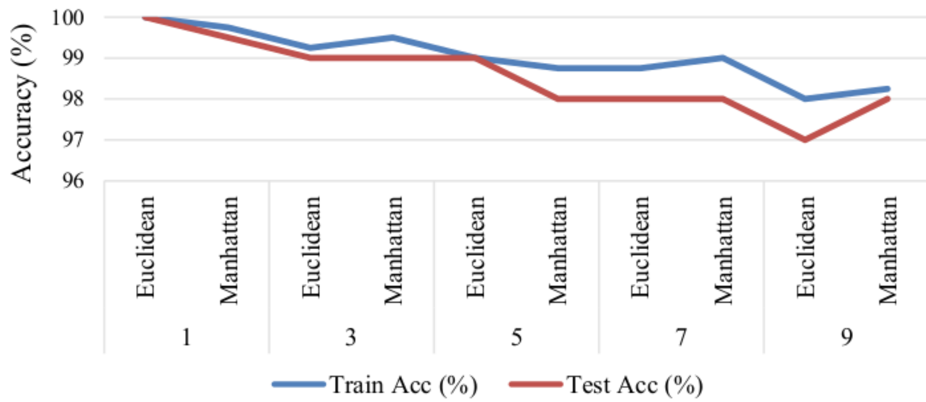


Fig. 9 The graph of classification result using K-NN method

3.2 Training Face Data Using Multiclass SVM

The multiclass SVM algorithms kernel is a collection of mathematical functions. It transforms input data into the desired format. Meanwhile, the model output is poor when the transformation is incorrect. Different SVM algorithms use various kernel functions. In this study, three experiments were conducted with the sigmoid, linear, and polynomial kernels to classify masked facial identities. Figure 10 shows the results of kernel variations using multiclass SVM.

In the learning process using SVM multiclass, image features were transformed into feature space using a kernel trick. The best results were obtained using a polynomial kernel with 100% training and testing accuracy. Furthermore, the image training data used produced numerous and scattered features, which were difficult to separate using straight-line kernels such as sigmoid and linear. However, the polynomial kernel was able to separate the scattered identity classes. Therefore, testing on real-time video was carried out via a multiclass SVM with a polynomial kernel.

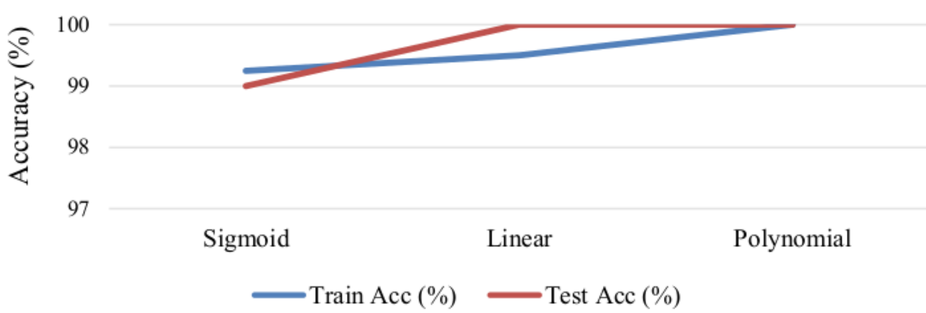


Fig. 10 The graph of classification result using multiclass SVM method

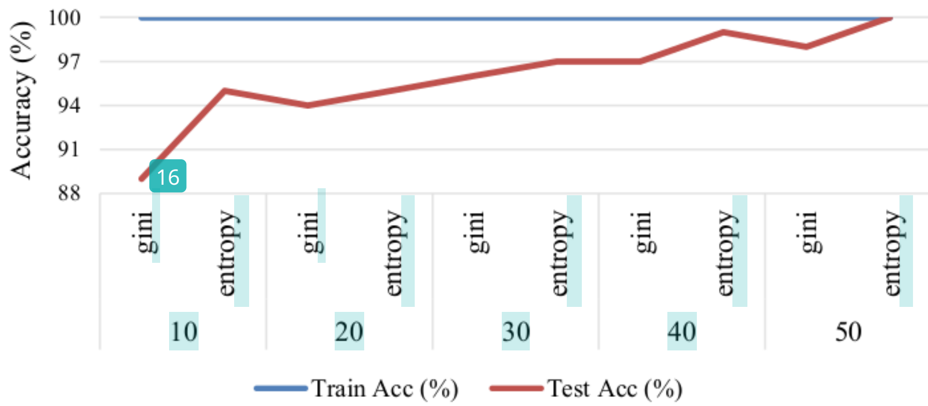


Fig. 11 The graph of classification result using random forest method

3.3 *Training Face Data Using Random Forest*

The classification of masked face identity using random forest produced several decision trees which were used for making predictions. From all the decision trees obtained, voting was carried out to produce the final prediction. Furthermore, an experiment was conducted on the effect of the criterion parameters and number of trees on accuracy results. Figure 11 shows the experimental results of variations in the number of trees for each criterion function.

The experimental training data using random forest obtained accuracy values in variations of the criterion and number of tree parameters. However, for 10 trees, the accuracy value obtained was overfitting, where the test accuracy was smaller than the training. Although the training time was shorter, 0.1051 and 0.4268 s for 10 and 50 trees respectively, overfitting indicates that the model built is not good. The results showed that the use of criterion and 50 trees produced a training and testing accuracy of 100%. Therefore, the model with these results was used for testing the real-time video data.

3.4 *Testing Using Real-Time Video Data*

The real-time video was used for testing data, meanwhile, the camera resolution used was Full HD (1920 × 1080). Furthermore, the best model from the classification results was obtained using K-NN, multiclass SVM, and Random Forest. Then, accuracy was evaluated by observing the face in the video frames detected whether the identity is true or false based on Eq. 4. Moreover, the video data processing speed was evaluated by calculating the average of all detected frames. Table 1 shows the results of accuracy and speed using real-time video data.

Table 1 The accuracy result of testing on real-time video data

Model building	Accuracy (%)	Processing time (s)
FaceNet + K-NN	96.03	0.056
FaceNet + multiclass SVM	96.15	0.055
FaceNet + random forest	54.04	0.061

Fig. 12 The result of real-time video data testing

Based on Table 1, the combination of FaceNet with Multiclass SVM produced the best results with an accuracy of 96.15%. These results are not significantly different from the combination of FaceNet with K-NN. However, in terms of processing speed, FaceNet and Multiclass SVM combination was the fastest, with 0.055 s or 18 fps. Therefore, this combination runs in real-time and is entirely accurate. An example of test results using video data is shown in Fig. 12.

3.5 Discussion

The experimental results of three supervised learning methods show that the parameters of the K-NN, multiclass SVM, and Random Forest methods significantly affect accuracy. In the K-NN method, the K value significantly influenced the accuracy as shown in Fig. 9. In addition, the type of kernel used also affects the accuracy. Meanwhile, in the Random Forest method, the criterion parameter and the number of trees significantly affect the accuracy. The best results were obtained using FaceNet with multiclass SVM and a polynomial kernel. This result was better than the previous study [6], with an elevated accuracy from 94.5 to 96.15% and run in real-time at 18 fps. These results can be used as a reference regarding a safer contactless attendance system during the pandemic era. The world community has started using special techniques to break the spread of the coronavirus, one of which is the development of a contactless masked face recognition method.

There were certain limitations in this study particularly when the hairstyle and face expression are changed. Therefore, the shape of the hair was not changed and used normal expression during the experiments. Future studies are recommended to use a combination of other feature extraction techniques to increase the accuracy of results.

4 Conclusion

The coronavirus pandemic health protocol requires the use of face masks when performing daily activities worldwide. This condition makes face recognition inaccurate because the face is partially covered. Furthermore, the combination of deep learning methods used in previous studies produced quite a good accuracy but failed to run in real-time data. Therefore, we propose combining pre-trained models and supervised learning classification methods in a masked face recognition system. The combination of FaceNet and multiclass SVM methods used in this study increased the accuracy by 1.65% (from 94.5 to 96.15%) and run in real-time at 18 fps. These results prove that the system built has good accuracy and is relatively fast in processing real-time data. Future research can use a feature extraction algorithm that can extract the changing features of the human hair so that the results are more accurate for masked face recognition.

Acknowledgements Thanks to LPPM Institut Teknologi Telkom Purwokerto through the Internal Research Grant for funding in the publication process of this journal.

References

1. Leung NHL (2021) Transmissibility and transmission of respiratory viruses. Nat Rev Microbiol 19:528–545
2. Li L, Mu X, Li S, Peng H (2020) A review of face recognition technology. IEEE Access 8:139110–139120
3. Sunaryono D, Siswantoro J, Anggoro R (2021) An android based course attendance system using face recognition. J King Saud Univ Comput Inf Sci 33(3):304–312
4. Logan AJ, Gordon GE, Loffler G (2017) Contributions of individual face features to face discrimination. Vision Res 137:29–39
5. Li Y, Guo K, Lu Y, Liu L (2021) Cropping and attention based approach for masked face recognition. Appl Intell 51(5):3012–3025
6. Aswal V, Tupe O, Shaikh S, Charniya NN (2020) Single camera masked face identification. In: Proceedings—19th IEEE international conference on machine learning and applications. IEEE, pp 57–60
7. Maity S, Das P, Jha KK, Dutta HS (2021) Face mask detection using deep learning. Appl Artif Intell Mach Learn 495–509
8. Schroff F, Kalenichenko D, Philbin J (2015) FaceNet: A unified embedding for face recognition and clustering. In: 2015 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 815–823

9. Zhao Y, Yu AP, Xu DT (2020) Person recognition based on facenet under simulated prosthetic vision. J Phys Conf Ser 1437(1)
10. Pranoto H, Kusumawardani O (2021) Real-time triplet loss embedding face recognition for authentication student attendance records system framework. JOIV Int J Inform Vis 5(2)
11. Adhinata FD (2021) Fatigue detection on face image using Facenet algorithm and K-nearest neighbor classifier. J Inf Syst Eng Bus Intell 7(1):22–30
12. Kremic E, Subasi A (2016) Performance of random forest and SVM in face recognition. Int Arab J Inf Technol 13(2):287–293
13. Wirdiani NKA, Hridayami P, Widiari NPA, Rismawan KD, Candradinata PB, Jayantha IPD (2019) Face identification based on K-nearest neighbor. Sci J Inform 6(2):150–159
14. Arafah M, Achmad A, Indrabayu, Areni IS (2019). Face recognition system using Viola Jones, histograms of oriented gradients and multi-class support vector machine. J Phys Conf Ser 1341(4)
15. Goel R, Mehmood I, Ugail H (2021) A study of deep learning-based face recognition models for sibling identification. Sensors 21(15)
16. Setiawan E, Muttaqin A (2015) Implementation of K-nearest neightbors face recognition on low-power processor. TELKOMNIKA (Telecommunication Computing Electronics and Control) 13(3)
17. Septiana N, Suciati N (2020) Combination of fast hybrid classification and K value optimization in K-NN for video face recognition. Register Jurnal Ilmiah Teknologi Sistem Informasi 6(1):65–73
18. Beli ILK, Guo C (2017) Enhancing face identification using local binary patterns and K-nearest neighbors. J Imag 3(3)
19. Bhardwaj A, Srivastava P (2021) A machine learning approach to sentiment analysis on web based feedback. Appl Artif Intell Mach Learn 127–139
20. Boser BE, Guyon IM, Vapnik VN A training algorithm for optimal margin classifiers. In: Proceedings of the 5th annual ACM workshop on computational learning theory, pp 144–152
21. Duan KB, Rajapakse JC, Nguyen MN (2007) One-versus-one and one-versus-all multiclass SVM-RFE for gene selection in cancer classification. Lect Notes Comput Sci 4447:47–56
22. Feroz N, Ahad MA, Doja F (2021) Machine learning techniques for improved breast cancer detection and prognosis—a comparative analysis. In: Applications of artificial intelligence and machine learning, pp 441–455
23. Tyralis H, Papacharalampous G, Langousis A (2019) A Brief review of random forests for water scientists and practitioners and their recent history in water resources. Water 11(5)
24. Zhai Y, Zheng X (2018) Random forest based traffic classification method in SDN. In: 2018 international conference on cloud computing, big data and blockchain (ICCBB), pp 1–5
25. Zeng XD, Chao S, Wong F (2010) Optimization of bagging classifiers based on SBCB algorithm. In: 2010 international conference on machine learning and cybernetics, vol 1, pp 262–267

Real-Time Masked Face Recognition Using FaceNet and Supervised Machine Learning

ORIGINALITY REPORT

13%
SIMILARITY INDEX

PRIMARY SOURCES

1	www.mdpi.com Internet	75 words — 2%
2	Agung Triayudi, Iskandar Fitri, Wahyu Oktri Widyarto, Sumiati. "Chapter 3 Improving the Quality and Education Systems Through Integration's Approach of Data Mining Clustering in E-Learning", Springer Science and Business Media LLC, 2022 Crossref	48 words — 1%
3	ebin.pub Internet	37 words — 1%
4	"Proceedings of the 1st International Conference on Electronics, Biomedical Engineering, and Health Informatics", Springer Science and Business Media LLC, 2021 Crossref	32 words — 1%
5	journals.usm.ac.id Internet	31 words — 1%
6	Janakiramaiah Bonam, Lakshmi Ramani Burra, Roopasri Sai Varshitha Godavarthi, Divya Jagabattula, Sowmya Eda, Soumya Gogulamudi. "Chapter 32 Multiple Face Recognition System Using OpenFace", Springer Science and Business Media LLC, 2023 Crossref	29 words — 1%

7 Y. Keerthi, A. Sneha, P. Vani Manasa, N. Neelima. "Enhanced Biometric Attendance Management System using IoT & Cloud", 2022 International Conference on Electronics and Renewable Systems (ICEARS), 2022 — 25 words — 1%
Crossref

8 ijeepse.id — 21 words — < 1%
Internet

9 Rifqi Akmal Saputra, Faisal Dharma Adhinata, Nur Ghaniaviyanto Ramadhan, Rifqi Alfinnur Charisma. "Chapter 10 A Model Convolutional Neural Network for Early Detection of Chili Plant Diseases in Small Datasets", Springer Science and Business Media LLC, 2023 — 20 words — < 1%
Crossref

10 dokumen.pub — 20 words — < 1%
Internet

11 www.hindawi.com — 18 words — < 1%
Internet

12 www.eltech.ru — 16 words — < 1%
Internet

13 R. Rameswari, S. Naveen Kumar, M. Abishek Aananth, C. Deepak. "Automated access control system using face recognition", Materials Today: Proceedings, 2020 — 15 words — < 1%
Crossref

14 Lucas Chong Wei-Jie, Siew-Chin Chong, Thian-Song Ong. "Masked face recognition with principal random forest convolutional neural network (PRFCNN)", Journal of Intelligent & Fuzzy Systems, 2022 — 13 words — < 1%
Crossref

15	Zhi-Yi Duan, Li-Min Wang, Musa Mammadov, Hua Lou, Ming-Hui Sun. "Discriminatory Target Learning: Mining Significant Dependence Relationships from Labeled and Unlabeled Data", Entropy, 2019 Crossref	13 words — < 1%
16	Pranav S. Pandit, Deniece R. Williams, Paul Rossitto, John M. Adaska et al. " Dairy management practices associated with multi-drug resistant fecal commensals and in cull cows: a machine learning approach ", PeerJ, 2021 Crossref	11 words — < 1%
17	www.science.gov Internet	11 words — < 1%
18	Rafael Pino-Mejías. "A comparison of classification models to identify the Fragile X Syndrome", Journal of Applied Statistics, 2008 Crossref	9 words — < 1%
19	Yassine Himeur, Somaya Al-Maadeed, Iraklis Varlamis, Noor Al-Maadeed, Khalid Abualsaud, Amr Mohamed. "Face Mask Detection in Smart Cities Using Deep and Transfer Learning: Lessons Learned from the COVID-19 Pandemic", Systems, 2023 Crossref	9 words — < 1%
20	doaj.org Internet	9 words — < 1%
21	Govind Jeevan, Geevar C. Zacharias, Madhu S. Nair, Jeny Rajan. "An Empirical Study of the Impact of Masks on Face Recognition", Pattern Recognition, 2021 Crossref	8 words — < 1%

22 Ismail Oztel, Gozde Yolcu Oztel, Devrim Akgun. "A hybrid LBP-DCNN based feature extraction method in YOLO: An application for masked face and social distance detection", Multimedia Tools and Applications, 2022
Crossref
8 words — < 1%

23 M Arafah, A Achmad, Indrabayu, I S Areni. "Face recognition system using Viola Jones, histograms of oriented gradients and multi-class support vector machine", Journal of Physics: Conference Series, 2019
Crossref
8 words — < 1%

24 Wahyono, Agus Harjoko, Andi Dharmawan, Faisal Dharma Adhinata, Gamma Kosala, Kang-Hyun Jo. "Real-Time Forest Fire Detection Framework Based on Artificial Intelligence Using Color Probability Model and Motion Feature Analysis", Fire, 2022
Crossref
8 words — < 1%

25 core.ac.uk
Internet
8 words — < 1%

26 "Blockchain and Trustworthy Systems", Springer Science and Business Media LLC, 2020
Crossref
7 words — < 1%

27 Lecture Notes in Computer Science, 2015.
Crossref
7 words — < 1%

28 "Information and Communication Technology and Applications", Springer Science and Business Media LLC, 2021
Crossref
6 words — < 1%

29 Busra Kocacinar, Bilal Tas, Fatma Patlar Akbulut, Cagatay Catal, Deepti Mishra. "A Real-Time CNN-
6 words — < 1%

based Lightweight Mobile Masked Face Recognition System", IEEE Access, 2022
Crossref

30	Hafiz Naderi, Julia Ramírez, Stefan van Duijvenboden, Esmeralda Ruiz Pujadas et al. "Predicting left ventricular hypertrophy from the 12-lead electrocardiogram in the UK Biobank imaging study using machine learning", European Heart Journal - Digital Health, 2023 Crossref	6 words — < 1%
31	Suman Deb, S. Dandapat. "Emotion classification using residual sinusoidal peak amplitude", 2016 International Conference on Signal Processing and Communications (SPCOM), 2016 Crossref	6 words — < 1%

EXCLUDE QUOTES	OFF	EXCLUDE SOURCES	OFF
EXCLUDE BIBLIOGRAPHY	ON	EXCLUDE MATCHES	OFF