Journal of Applied Informatics and Computing (JAIC)
Vol.10, No.1, February 2026, pp. 165~170
e-ISSN: 2548-6861

Performance Evaluation of Face Mask Detection Using Feature Descriptor and Supervised Learning Method

Adi Suheryadi [1]*, Faisal Dharma Adhinata [2], Heru Wijanarko [3]*****
* Department of Informatics, Politeknik Negeri Indramayu, Indramayu, Indonesia
** Software Engineering, Telkom University, Purwokerto, Indonesia
*** Department of Electrical Engineering, Politeknik Negeri Batam, Batam, Indonesia
adisuheryadi@polindra.ac.id [1], faisaldharma@telkomuniversity.ac.id [2], wijanarko@polibatam.ac.id [3]

Article Info

Article history:

Received 2025-12-11
Revised 2025-12-29
Accepted 2026-01-07

Keyword:

Face Mask Detection,
Feature Extraction,
LBP,
Machine Learning,
Random Forest.

ABSTRACT

The use of masks as a measure to prevent the spread of dangerous diseases such as COVID-19 and others has become a social norm. Manual detection is less effective, especially in areas with high mobility. This study develops and evaluates an artificial intelligence (AI)-based face mask detection system using feature description and machine learning models. An optimal and lightweight model can help hospitals implement face mask detection systems in areas prone to disease transmission. Image preprocessing, feature description, supervised learning model studies, and performance evaluation were conducted using accuracy, precision, recall, and F1-score metrics, and a confusion matrix was used to assess the overall model performance. The performance evaluation results show that the combination of the LBP feature description with the random forest model is the best choice, with a relatively high and stable accuracy of around 96.3% with an average value, precision, recall, and F1-score of around 96% using K-Fold Cross-Validation. These findings suggest that this method is helpful in detecting mask use while minimizing error and computation rates. This study contributes to the development of lightweight mask detection systems that can be used in real time.

I. INTRODUCTION

In 2020, the COVID-19 epidemic spread rapidly throughout Indonesia [1]. People were required to wear masks to protect themselves from exposure to the virus. Currently, Indonesians wear masks in their daily activities. Wearing a face mask protects against exposure to airborne infectious illnesses [2]. People in public places, such as offices, schools, and shopping malls, should wear masks correctly to help prevent the transmission of disease. Public compliance with face mask use promotes both personal and collective protection [3], thereby limiting the spread of illness. As a result, monitoring proper face mask use is an important component of preventative efforts to ensure the health of the Indonesian public.

Officers cannot manually monitor all individuals wearing face masks in public areas due to their limited field of view. Manual monitoring is inefficient and can cause weariness from having to remind everyone. As a result, a face mask detection device is required to aid officers. The monitoring process can make use of artificial intelligence (AI) [4], namely computer vision [5]. Video data may be processed to detect everyone wearing a face mask. This technique enables automated, reliable identification of face mask use.

Among numerous AI methods, machine learning is a popular methodology for object classification among academics [6], [7], [8]. Machine learning can identify patterns in image files and swiftly differentiate items. Machine learning is clearly superior to traditional rule-based methods, such as if-else programming, because it can adapt to a wide range of data variations, including lighting effects and mask types. The AI-based classification method is Support Vector Machine (SVM) and Random Forest (RF). SVM is a popular supervised learning method for separating objects with limited datasets [9]. The RF method can also work on limited datasets [10]. In addition to the choice of machine learning method, the selection of object features is also a benchmark for classification success. This research did not implement deep learning due to its high computational time. Uma researcher [27] used MobileNetV2 for object classification. The resulting speed was only 25 FPS, despite using a Graphics Processing Unit (GPU). This is because the deep learning model relies heavily on convolution during data processing.

Machine learning performance also depends on the features used to describe the objects being classified. In this research, the objective was to distinguish between masked and unmasked faces. The shapes and textures of masked and unmasked faces are certainly different. This study will evaluate the use of the Histogram of Oriented Gradients (HOG) [11], [12] and Local Binary Patterns (LBP) [13], [14] as feature descriptors. HOG represents the edge and gradient features of an object, while LBP extracts the object's texture pattern in the image. The evaluation will be conducted on a combination of HOG-SVM, LBP-SVM, HOG-RF, and LBP-RF.

The goal of this research is to develop an automatic masked-face detection system based on video data using machine learning. Model evaluation will use accuracy to assess a machine learning model's performance. The main contribution of this research is the evaluation of using a feature descriptor and a supervised learning method. The use of handcrafted machine learning aims to ensure fast and efficient prediction processing.

II. METHODS

The masked face detection system begins with video input. The video is extracted into frames, which are then resized to speed up computation time. Face detection is performed using the Haar cascade on the video frames. Then, the first detection checks whether a nose is present. If a nose is detected, it will be classified as no mask. Then, if the nose is not detected, it is likely wearing a mask. The detected face is divided into two halves: the upper and lower halves. In this research, the area used to detect mask use is the lower half of the face. The second detection is to determine whether the lower half is indeed wearing a mask. Prediction uses the HOG-SVM, LBP-SVM, HOG-RF, or LBP-RF model. If a mask is detected, it is classified as with mask. Conversely, if a mask is not detected, it is classified as no mask. Figure 1 shows the flow of the masked face detection system.

A. Data Acquisition

This research is based on a publicly available dataset from Deb Chandrika [15]. The dataset was collected from various sources, namely RMFD Datasets[28], Kaggle, and scraping from the searching API, as explained. Actually, This dataset consists of 4095 images belonging to two classes including with_mask 2165 images and without_mask: 1930 images. The dataset has diverse characteristics, including various types of masks such as medical, cloth, and respirator masks, etc., as well as various angles of image capture, predominantly taken from the front of the face, sometimes full face, and sometimes the entire image along with different object backgrounds. This diversity makes the dataset quite representative for training mask detection models to be able to work robustly in real-world conditions. In this study, we used 400 image data consisting of two classes, namely with_mask (200 images) and without_mask (200 images), to reduce processing complexity when the data was trained. The standard image size was 64 x 64 pixels.

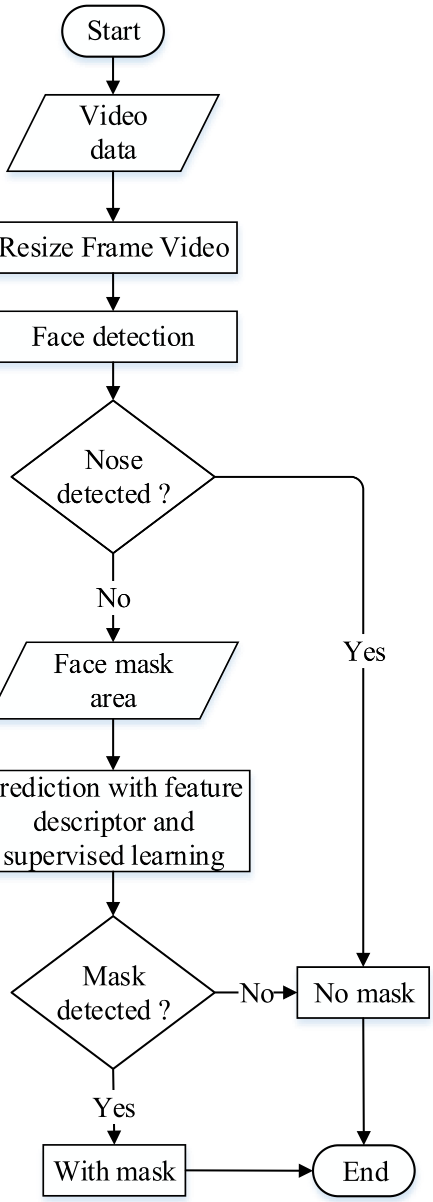

Figure 1. Masked face detection system

B. Face Segmentation

Face detection in this study uses the Haar cascade frontal face. The face to be detected must be facing forward. Then, after the face is detected, the presence or absence of a nose is determined using the Haar cascade nose. This Haar cascade is OpenCV-based and implemented in Python [16]. If the nose area is detected, it will be classified as no mask. Then, if no nose is detected, the face will be divided into two. Half of the area below the nose will be used for feature extraction. An illustration of the use of the lower half of the face is shown in Figure 2.

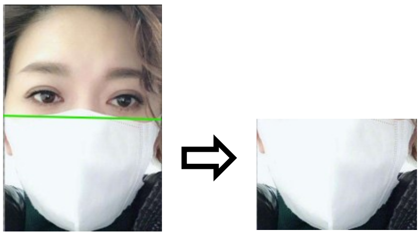

Figure 2. Illustration of the use of the lower half of the face

C. Feature Extraction

Feature extraction in this study uses HOG and LBP methods. HOG and LBP are two widely used feature descriptors in image processing. HOG represents an image based on the distribution of gradient directions (edges) [17], making it excellent for describing the shape and contour of an object. In contrast, LBP defines an image using a binary pattern surrounding each pixel [18], emphasizing local texture information. This descriptor feature is intended to distinguish between masked and unmasked faces. Furthermore, there are numerous varieties of masks. Therefore, the descriptor feature should be able to accurately classify them.

D. Classification Using Supervised Learning

Support Vector Machine (SVM) and Random Forest are widely used machine learning techniques for classification. SVM works by determining the best separating hyperplane in the feature space that maximizes the margin between data from different classes. Thus, SVM is good at separating two or more classes with well-defined boundaries, and it often performs well even with small datasets [9]. Meanwhile, Random Forest is an ensemble method that creates numerous decision trees and aggregates their predictions using voting. Each tree is trained on randomly selected data samples and features, making the model more robust against overfitting [10]. This research will compare the use of these two supervised learning methods. The best model will be used for video data evaluation.

E. System Evaluation

In this research, model evaluation was conducted through two main stages: data splitting and K-Fold-based cross-validation. The dataset was split into training (80%) and test (20%) sets, and balanced class proportions were maintained in both subsets using stratified sampling. Next, during the training stage, the GridSearchCV technique was used with 5-fold cross-validation, which alternately split the training data into 5-folds: one as validation data and the other four as training data. This procedure allowed selection of the optimal parameter combination based on the average F1-macro across five training cycles. The optimal model resulting from the cross-validation process was then tested on a test set not used in training to obtain an objective final accuracy and assess the model's generalization to new data.

F. System Configuration

The search for the best configuration of the classification model was conducted by exploring various hyperparameter combinations with GridSearchCV. For the Support Vector Machine (SVM) algorithm, the combined parameters include the regularization value C = {0.1, 1, 10} to control the trade-off between margin and classification error, the kernel choice {linear, rbf} to determine the transformation of the feature space, and the gamma value = {scale, 0.01, 0.001} which affects the complexity of the decision function in the RBF kernel. Meanwhile, in the Random Forest model, the parameters tested include the number of decision trees n_estimators = {100, 300}, the maximum tree depth max_depth = {None, 20}, the minimum sample size for the formation of new nodes min_samples_split = {2, 5}, the minimum number of samples on a leaf min_samples_leaf = {1, 2}, and max_features = {sqrt, log2} to set the number of features considered at each branch. The configuration of SVM and Random Forest is shown in Table 1.

TABLE 1
SVM AND RANDOM FOREST CONFIGURATION

Model	Parameter	Value
SVM	C	{0.1, 1, 10}
	kernel	{linear, rbf}
	gamma	{scale, 0.01, 0.001}
Random Forest	n_estimators	{100, 300}
	max_depth	{None, 20}
	min_samples_split	{2, 5}
	min_samples_leaf	{1, 2}
	max_features	{sqrt, log2}

TABLE 2
DATA PROCESSING CONFIGURATION

Feature extraction	Setting Code	Parameter Configuration	Characteristics
HOG	A	Orientations = 8, Pixels per Cell = (16×16), Cells per Block = (2×2)	Compact, fast extraction time, low structural detail
HOG	B	Orientations = 9, Pixels per Cell = (8×8), Cells per Block = (2×2)	complexity and precision of features
HOG	C	Orientations=12, Pixels per Cell = (8×8), Cells per Block = (3×3)	Highly detailed, high complexity, large dimensions
LBP	A	P = 8, R = 1, Method= uniform	Simple texture representation
LBP	B	P = 16, R = 2, Method = uniform	Capturing a wider range of texture patterns
LBP	C	P = 24, R = 3, Method = nri_uniform	Very rich texture details, more sensitive to pattern variations

Each feature extraction method was tested with three parameter configurations (Setting A, Setting B, and Setting C) to assess the resulting feature complexity. In HOG feature extraction, Setting A uses a simpler configuration of orientations and cell sizes to produce compact and fast-computing features. Setting B increases the descriptor resolution by reducing the cell size. Setting C adds more orientations and blocks, resulting in richer feature details but higher computational complexity. Meanwhile, in LBP feature extraction, Setting A uses the smallest number of neighbors and a uniform pattern, Setting B increases the radius with a larger number of neighbors, while Setting C uses a non-rotation-invariant uniform pattern to capture texture patterns with more complex details. Comparing these three feature extraction settings allows evaluation of the effect of feature complexity on classification performance and computational requirements. The feature extraction scenario setting table is shown in Table 2.

III. Results And Discussion

The feature extraction results using the HOG descriptor are shown in Figure 3. In the HOG without a mask image, more gradients are produced compared to the HOG with a mask. This effect occurs because the mask's texture tends to be uniform, eliminating gradients and magnitude in the area where the mask is applied.

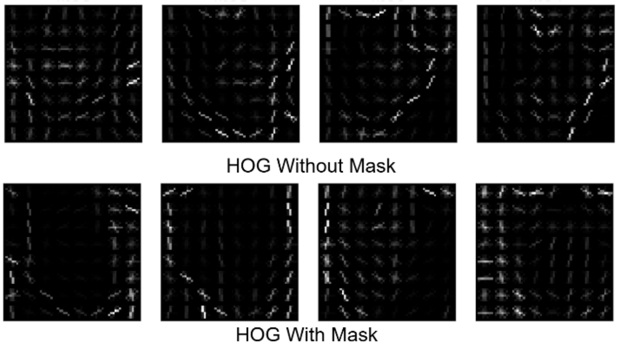

Figure 3. Visualization of HOG Feature Descriptors

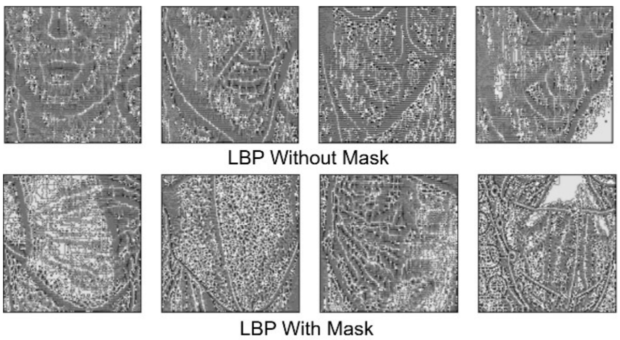

Figure 4. Visualization of LBP Feature Descriptors

On the other hand, feature extraction using LBP descriptors is shown in Figure 4. LBP descriptors contain more detailed information, including more complex textures, as shown by differences between images with and without masks. Therefore, the histograms of masked and unmasked images will appear to have different distributions compared to those obtained from HOG feature extraction, as shown in Figure 5.

A. Results of Training and Validation

Based on the test results in Table 3, the best overall combination in this study was LBP Setting C (P = 24, R = 3, Method = nri_uniform) using Random Forest, with the highest test accuracy of 0.963 and a mean F1-macro of 0.941. This indicates that adding complexity to the texture pattern using 24 neighbors and a larger radius captures image pattern variations better than other LBP configurations.

In the HOG descriptor, the best performance was achieved with Setting C using SVM, with an accuracy of 0.950 and a mean F1-macro of 0.944, outperforming Settings A and B. This demonstrates that increasing the number of gradient orientations and larger block sizes enhances the model's ability to capture the shape and contour details of objects. However, Setting C also results in the highest number of features, approximately 21,168.

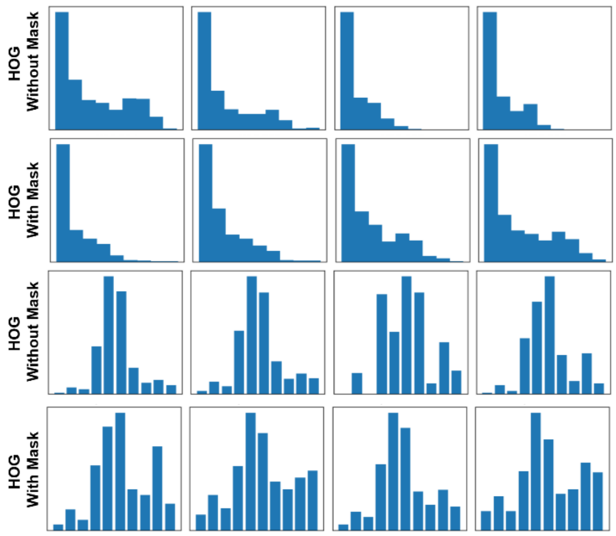

Figure 5. Visualization of the difference between HOG and LBP histograms

TABLE 3
COMPARISON OF PARAMETER EVALUATION

Feature	Set	Classifier	Mean F1-macro	Testing Accuracy	Feature Dimensions
HOG	A	SVM	0.959	0.925	1568
HOG	A	RF	0.937	0.887	1568
HOG	B	SVM	0.953	0.925	8100
HOG	B	RF	0.925	0.887	8100
HOG	C	SVM	0.944	0.950	21168
HOG	C	RF	0.941	0.912	21168
LBP	A	SVM	0.877	0.925	256
LBP	A	RF	0.874	0.912	256
LBP	B	SVM	0.906	0.912	256
LBP	B	RF	0.928	0.900	256
LBP	C	SVM	0.953	0.938	256
LBP	C	RF	0.941	0.963	256

The comparison between classifiers shows that SVM is more effective for HOG. In contrast, Random Forest with configuration setting on n_estimators: 300, min_samples_split: 5, min_samples_leaf: 1, and max_features: sqrt shows the best performance on LBP Setting C (P = 24, R = 3, Method = nri_uniform). This distinction arises because HOG produces high-dimensional, structured features, which are more suitable for linear decision boundaries in SVM, while LBP produces low-dimensional histogram features that are highly sensitive to small texture changes, so the Random Forest method is better at managing variation in the feature distribution. The complexity-efficiency of the descriptor features can be seen in HOG Setting C, which produces fairly high accuracy with a feature size much larger than LBP's, approximately 21,168 features compared to 256. This data shows that LBP is superior in terms of performance-to-complexity ratio.

B. Performance Comparison

Based on the results of comparative research presented in Table 4, the performance of face mask detection is influenced by the methods used and the characteristics of the dataset. Deep learning-based approaches, such as the Cascaded CNN in the research by Wei et al. [19], achieved 86.6% accuracy on a small dataset. In contrast, VGG-16, used by Sammy and Nanette [20], achieved 96% accuracy on a large dataset of 25,000 images. This shows that deep learning models are highly dependent on large datasets and require substantial computational resources to achieve optimal performance. On the other hand, classical method-based approaches such as Viola-Jones [21] and HGL [23] achieve accuracies of 93%-95% but are susceptible to variations in pose and lighting.

From this comparison, it can also be seen that the method proposed in this study, namely the combination of Local Binary Pattern (LBP) with parameter settings C and Random Forest, produces a relatively high accuracy of 96.3% compared to similar studies with relatively small datasets of around 400 images, as shown in Table 4. Some errors occurred due to the failure of the initial detector to detect the object's face and nose in the image due to a small face in the image. In addition, the resolution of the image also affected the characteristics described by the feature. These results indicate that traditional feature extraction methods combined with supervised learning classification algorithms can deliver competitive performance without requiring heavy computation.

TABLE 4
COMPARISON OF PERFORMANCE OF RELATED STUDIES

Research	Methods	Dataset	Performance
Wei et al [19]	Cascaded CNN	Masked Face, 200 images	86.6%
Sammy and Nanette [20]	VGG-16 CNN	No mentioned dataset, 25,000 images	96%
Nieto-Rodriguez et. al [21]	Viola-Jones detector	LFW, BAO [22], 496 faces on images	95%
Li et al [23]	HGL method	MAFA dataset [24], 35.000	93.6%
Das et al. [25]	CNN	FMD dataset [26]	95.7%
Ours	**LBP – RF**	**Deb Chandrika [15], 400 images**	**96.3%**

C. Results of Facemask Classification and Detection

The classification results for images with and without masks are shown in Figure 6, with probability scores of 0.93 and 1. Meanwhile, the results for mask detection in real-time video are presented in Figure 7, with a probability of about 0.97 and a frame rate of 38.8 FPS.

In a real-world face mask detection scenario, this system can be integrated into CCTV systems for real-time video analysis. The developed system only detects mask use, so it does not reveal the identities of offenders who are not wearing masks. Furthermore, facial data is not stored. Therefore, there are no privacy concerns for individuals. However, implementation challenges exist in public environments, such as variations in lighting, camera angles, occlusion, and facial blur. These factors mean that system performance still requires a controlled environment.

Figure 6. Result of face mask classification on a single image

Figure 7. Result of face mask detection on real-time video

IV. CONCLUSION

Based on the research results, a comparison of feature description and supervised learning methods for automatic mask detection on faces shows that the combination of LBP feature description with parameters C (P = 24, R = 3, Method = nri_uniform) and Random Forest with the settings n_estimators: 300, min_samples_split: 5, min_samples_leaf: 1, and max_features: sqrt perform the best. In this study, the model achieved average precision, recall, and F1-score of around 96%, with an accuracy of 96.3%, based on 5-fold cross-validation testing on the Chandrika dataset, which includes 400 images. In addition, this system has the potential to be applied in real-world settings, such as the real-time monitoring of health protocols in public areas with lightweight standard CPU-based computing devices. In the other hand, some opportunities and challenges for further work in this study, include adding more detailed classifications, such as incorrect mask classes, to detect incorrect mask usage. In addition, integration with automatic face detection systems also presents its own challenges and opportunities for more practical implementation.

REFERENCES

[1] Dicky *et al.*, "The trend of paediatric cases during the first year of the COVID-19 pandemic in North Sumatra, Indonesia," *IJID Regions*, vol. 8, pp. S18–S21, Sep. 2023.

[2] R. R. Kalavakonda, N. V. R. Masna, S. Mandal, and S. Bhunia, "A smart mask for active defense against airborne pathogens," *Sci Rep*, vol. 11, no. 1, p. 19910, Oct. 2021.

[3] L. Martinelli *et al.*, "Face Masks During the COVID-19 Pandemic: A Simple Protection Tool With Many Meanings," *Front. Public Health*, vol. 8, Jan. 2021.

[4] S. Asif, Y. Wenhui, Y. Tao, S. Jinhai, and K. Amjad, "Real Time Face Mask Detection System using Transfer Learning with Machine Learning Method in the Era of Covid-19 Pandemic," in *2021 4th International Conference on Artificial Intelligence and Big Data (ICAIBD)*, May 2021, pp. 70–75.

[5] A. Oumina, N. El Makhfi, and M. Hamdi, "Control The COVID-19 Pandemic: Face Mask Detection Using Transfer Learning," in *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science (ICECOCS)*, Dec. 2020, pp. 1–5.

[6] Y. Lin and H. Xie, "Face Gender Recognition based on Face Recognition Feature Vectors," in *2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE)*, Sep. 2020, pp. 162–166.

[7] F. D. Adhinata, N. A. F. Tanjung, W. Widayat, G. R. Pasfica, and F. R. Satura, "Real-Time Masked Face Recognition Using FaceNet and Supervised Machine Learning," in *Proceedings of the 2nd International Conference on Electronics, Biomedical Engineering, and Health Informatics*, T. Triwiyanto, A. Rizal, and W. Caesarendra, Eds., Singapore: Springer Nature, 2022, pp. 189–202.

[8] F. Cahyono, W. Wirawan, and R. Fuad Rachmadi, "Face Recognition System using Facenet Algorithm for Employee Presence," in *2020 4th International Conference on Vocational Education and Training (ICOVET)*, Sep. 2020, pp. 57–62.

[9] S. Pandey and P. Mishra, "An Efficient Image Classification using SVM, CNN," *IJCA*, vol. ICAIDSC2023, no. 1, pp. 5–10, Jan. 2025.

[10] J. Luan, C. Zhang, B. Xu, Y. Xue, and Y. Ren, "The predictive performances of random forest models with limited sample size and different species traits," *Fisheries Research*, vol. 227, p. 105534, Jul. 2020.

[11] M. Arafah, A. Achmad, Indrabayu, and I. S. Areni, "Face recognition system using Viola Jones, histograms of oriented gradients and multi-class support vector machine," *J. Phys.: Conf. Ser.*, vol. 1341, no. 4, p. 042005, Oct. 2019.

[12] M. Yusuf and A. A. Fuad, "Real time implementation of face recognition based automatic institutional attendance system," *WSEAS Transactions on Signal Processing*, vol. 17, pp. 46–56, 2021.

[13] P. A. R. Devi and R. P. N. Budiarti, "Image Classification with Shell Texture Feature Extraction Using Local Binary Pattern (LBP) Method," *Applied Technology and Computing Science Journal*, vol. 3, no. 1, pp. 48–57, Sep. 2020.

[14] V. Nguyen Thanh Le, B. Apopei, and K. Alameh, "Effective plant discrimination based on the combination of local binary pattern operators and multiclass support vector machine methods," *Information Processing in Agriculture*, vol. 6, no. 1, pp. 116–131, Mar. 2019.

[15] C. Deb, "Face Mask Detection.," GitHub. Accessed: Nov. 21, 2025. [Online]. Available: https://github.com/chandrikadeb7/Face-Mask-Detection/tree/master/dataset

[16] OpenCV, *OpenCV (Open-Source Computer Vision), OpenCV modules*. Accessed: Sep. 02, 2025. [Online]. Available: https://docs.opencv.org/4.x/

[17] K. T. Islam, S. Wijewickrema, R. G. Raj, and S. O'Leary, "Street Sign Recognition Using Histogram of Oriented Gradients and Artificial Neural Networks," *Journal of Imaging*, vol. 5, no. 4, p. 44, Apr. 2019.

[18] I. F. Katili, M. A. Soeleman, R. A. Pramunendar, and others, "Character Recognition of Handwriting of Javanese Character Image using Information Gain Based on the Comparison of Classification Method," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 7, no. 1, pp. 193–200, 2023.

[19] W. Bu, J. Xiao, C. Zhou, M. Yang, and C. Peng, "A cascade framework for masked face detection," in *2017 IEEE International Conference on Cybernetics and Intelligent Systems (CIS) and IEEE Conference on Robotics, Automation and Mechatronics (RAM)*, Nov. 2017, pp. 458–462.

[20] S. V. Militante and N. V. Dionisio, "Real-Time Facemask Recognition with Alarm System using Deep Learning," in *2020 11th IEEE Control and System Graduate Research Colloquium (ICSGRC)*, Aug. 2020, pp. 106–110.

[21] A. Nieto-Rodríguez, M. Mucientes, and V. M. Brea, "System for Medical Mask Detection in the Operating Room Through Facial Attributes," in *Pattern Recognition and Image Analysis*, R. Paredes, J. S. Cardoso, and X. M. Pardo, Eds., Cham: Springer International Publishing, 2015, pp. 138–145.

[22] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled Faces in the Wild: A Database forStudying Face Recognition in Unconstrained Environments," in *Workshop on Faces in "Real-Life" Images: Detection, Alignment, and Recognition*, Marseille, France: Erik Learned-Miller and Andras Ferencz and Frédéric Jurie, Oct. 2008. Accessed: Sep. 09, 2025. [Online]. Available: https://inria.hal.science/inria-00321923

[23] S. Li *et al.*, "Multi-angle Head Pose Classification when Wearing the Mask for Face Recognition under the COVID-19 Coronavirus Epidemic," in *2020 International Conference on High Performance Big Data and Intelligent Systems (HPBD&IS)*, May 2020, pp. 1–5.

[24] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting Masked Faces in the Wild With LLE-CNNs," presented at the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2682–2690. Accessed: Dec. 09, 2025. [Online]. Available: https://openaccess.thecvf.com/content_cvpr_2017/html/Ge_Detecting_Masked_Faces_CVPR_2017_paper.html

[25] A. Das, M. Wasif Ansari, and R. Basak, "Covid-19 Face Mask Detection Using TensorFlow, Keras and OpenCV," in *2020 IEEE 17th India Council International Conference (INDICON)*, Dec. 2020, pp. 1–5.

[26] andrewmvd, "Face Mask Detection Keras." Accessed: Sep. 09, 2025. [Online]. Available: https://www.kaggle.com/andrewmvd/face-mask-detection.

[27] Pranav, S., Ma, S., & Vignesh, V. (2025, February). Smart Agriculture: Enhancing Crop and Weed Detection Using MobileNetV2 for Autonomous Farming Systems. In *2025 3rd International Conference on Intelligent Systems, Advanced Computing and Communication (ISACC)* (pp. 1-6). IEEE.

[28] Wang, Z., Huang, B., Wang, G., Yi, P., & Jiang, K. (2023). *Masked face recognition dataset and application.* *IEEE Transactions on Biometrics, Behavior, and Identity Science*, 5(2), 298-304.