Low-Cost Camera-Based Smart Surveillance System for Detecting, Recognizing, and Tracking Masked Human Face

https://doi.org/10.3991/ijim.v15i23.22725

Ervan Adiwijaya Haryadi, Grafika Jati(✉), Ario Yudo Husodo, Wisnu Jatmiko
Universitas Indonesia, Depok, Indonesia
grafikajati@ui.ac.id

Abstract—A surveillance system is still the most exciting and practical security system to prevent crime effectively. Surveillance systems run on edge devices such as the low-cost Raspberry mobile camera with the Internet of Things (IoT). The primary purpose of this system is to recognize the identity of the face caught by the camera. However, it raises the challenge of unstructured image/video where the video contains low quality, blur, and variations of human poses. Moreover, the challenge is increasing because people used to wear a mask during the Covid -19 pandemic. Therefore, we proposed developing an all-in-one surveillance system with face detection, recognition, and face tracking capabilities. The surveillance system integrated three modules: Multi-Task Cascaded Convolutional Network (MTCNN) face detector, VGGFace2 face recognition, and Discriminative Single-Shot Segmentation (D3S) tracker. We train new face mask data for face recognition and tracking. This system utilizes the Raspberry Pi camera and processes the frame on the cloud as a mobile sensor approach. The proposed method was successfully implemented and got competitive detection, recognition, and tracking results under an unconstrained surveillance camera.

Keywords—surveillance system, face detection, face recognition, face tracking, low-cost camera, masked face

1 Introduction

The COVID-19 pandemic, which has spread since 2020, has made humans adapt socially to minimize the virus's infection rate, such as social distancing and Work From Home (WFH). Many companies have been forced to lay off employees during this pandemic to maintain profit and avoid bankruptcy. The high rate of layoffs made people start switching professions to other sectors. Unfortunately, not everyone has an opportunity to get a job again, so that it has the potential to increase crime. Meanwhile, Stickle and Felson said no visible increase in crime rates; we need to anticipate that the pandemic is still ongoing for an indefinite term [1]. One of the crimes that often occurs is theft or home burglary. Usually, before acting, the perpetrator has already staked out the target house. Preventive measures are needed that can help people to keep their homes from becoming targets of crime.

Many surveillance systems are easy to install in homes. Unfortunately, that existing surveillance systems do not yet have intelligent systems. Surveillance systems not only record activity but also act as a prevention tool. The proposed approach is an innovative technology in the smart home concept explained by Wei [2]. A surveillance system equipped with closed-circuit television (CCTV) to identify the suspect who committed the crime in real-time. Thus, surveillance systems must have the capability to perform automatic detection, recognition, tracking, and behavior analysis.

Nowadays, researches still improve intelligent surveillance system performance. Kakarla develops new CNN architecture for face recognition in the attendance system, achieving 99% accuracy [3]. Rai also successfully created real-time face detection and recognition using a smartphone camera [4]. However, both research still deals with the ideal condition of the front face image. While, in an actual situation, CCTV is installed at the top of the view, so it produces a more varying size and angle of the face. The viewing angles vary depending on the person's position when entering the camera region of interest (ROI). Zheng solved those problems by developing face recognition under unconstraint video/images [5]. Zheng utilizes a multi-scale single shot detector combined with an unsupervised subspace learning approach. The research successfully dealt with face recognition in various poses, illumination, occlusion, scene, and blur image.

Currently, during the pandemic, the detection and recognition system is facing a new challenge. People have to use a mask to protect themselves, making the face area's visibility smaller and more obscure. Draughon proposed a surveillance system that can detect a person in a pedestrian park [6]. Draughon utilizes Mask R-CNN as a person detector, then combined with CNN-based face mask classification. However, Draughon only does two-class, either someone using a mask or not. On surveillance systems, we need recognition capability to identify a person's identity under all conditions. One of the latest research on masked targets, Negi also proposed a CNN-based face mask classifier [7]. Negi implemented the classifier in Keras Surgeon with a model pruning mechanism. However, it still deals with the front view image, which is not the natural condition in the surveillance system.

Furthermore, detection and recognition are not enough for surveillance systems. The system also needs tracking capabilities. Tracking helps follow suspicious movement. Face tracking also increases recognition accuracy, especially on masked faces. It is because face tracking makes face recognizers take consecutive frames before determining the identity of a face.

Jain developed CamAspect, an intelligent surveillance system that employs Deep SORT tracker and FaceNet to detect, track, and recognize a person [8]. As a result, Jain obtained accuracy, about 99% for multiple face recognition, while 61.1 MOTA score for detection and tracking. However, these studies have not solved the detection and identification of masked faces.

This paper develops an all-in-one surveillance system framework capable of detecting, recognizing, and tracking people's faces caught in the surveillance area. This system is adapted from a global detection system to detect a target caught on camera. After the face is successfully detected, we recognize the person identifies as the detected face. The system also has long-range tracking capability that can re-detect after loss from camera view. The workflow of the system that we propose can be seen in Figure 1. The system uses the Internet of Things (IoT) framework that utilizes a single board computer and mobile camera from Raspberry. Our system is designed in low-cost infrastructure and can be combined with existing smart-home IoT-based applications [9].

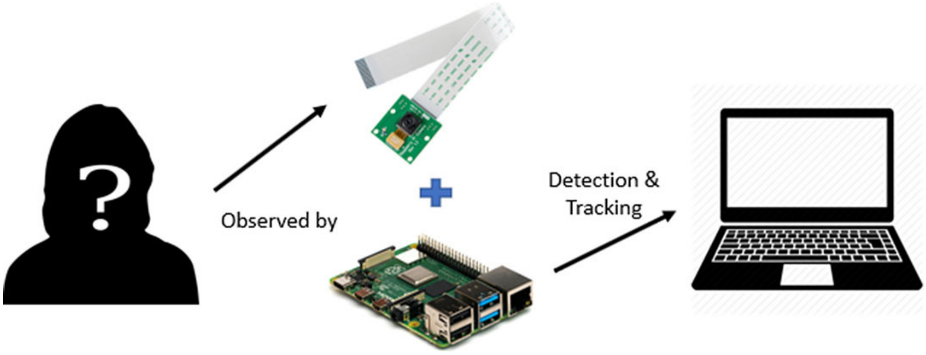


Fig. 1. Workflow overview of the proposed system

This paper aims to integrate three modules: face detection, recognition, and tracking, into a single system. We utilized the latest face detector for face detection, namely Multi-Task Cascaded Convolutional Network (MTCNN) [10]. Furthermore, the proposed system deal with unstructured conditions such as various poses, illumination, and blur images. So, we train new data for unconstrained images under the recognition system's VGG2 [11] algorithm. Finally, when a face is identified, the tracker follows the face movements using a state-of-the-art tracker, namely D3S [11]. Thus, the proposed system contributes to detecting, recognizing, and tracking a face in various conditions, even in masked faces.

This paper is arranged in the following order: the second section defines other work related to this work. The third section describes the research methodology undertaken. The process and result of our experiment, alongside its discussion, are detailed in section four. Finally, section five contains the conclusions drawn.

2 Related works

2.1 Visual face detection and recognition

Face detection is used to recognize human faces in digital images and mark them in a bounding box. In general, face detection techniques work by looking for features or components exclusively on the face, such as a pair of eyes, nose, and mouth. Unfortunately, face detection cannot identify whose face it is detected in the picture [12]. Face recognition is a method that recognizes the identity of the face. Figure 2 illustrates the difference between the output of face detection and the recognition process.

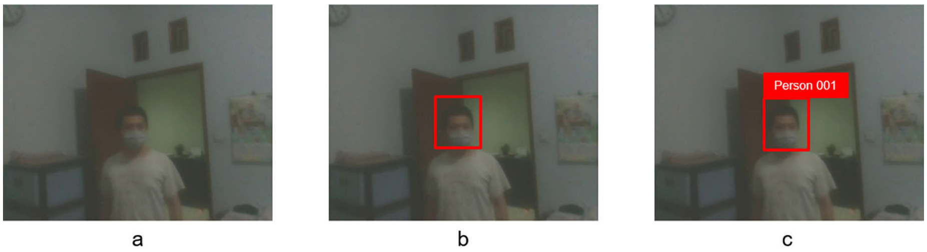


Fig. 2. An illustration of face detection and recognition task.
(a) raw image. (b) face detection. (c) and face recognition

Several studies focus on the detection and recognition of masked faces. Bu introduced triple-cascaded CNN to detect faces wearing full-face masks covering the entire face except for the eyes [13]. Lin proposed the MLeNet, which also conducted full-face masks people detection [14]. Ejaz [15] used the Viola-Jones algorithm and PCA to extract facial features and perform mask and non-mask recognition.

Ge [16] proposed the LLE-CNN method consisting of three modules. The first module is a proposal module for facial feature extraction. The second is an embedding module that forms vectors based on the similarities between masked. The last module is unmasked faces and a verification module that performs bounding box regression and face classification. Both Bu [13], Lin [14], Ge [16], and Ejaz [15] have solved face mask detection only. Ding [17] conducted face mask recognition using the CNN branching method, namely the global learning method and partial Learning on certain parts of the uncovered face area. However, the face data is still in an ideal condition which is the front view face image.

2.2 Face tracking

There are two types of object tracking based on the target presence: short-term and long-term object tracking. Short-term is a tracking problem whose target is assumed to be always in the image during the observation in a short duration. This condition makes the tracking method not require a re-detection mechanism to re-recognize the target [18]. On the contrary, the target can disappear from the image during the observation long-term object tracking [19].

A target can enter a camera area in the surveillance system, then disappear and enter other camera points of view. Therefore, long-term trackers need to be implemented so the surveillance system can identify the same target in many different cameras. This paper upgrading short-term trackers to become long-term trackers by adding a re-detection step.

3 Methodology

The surveillance system consists of three main phases, namely face detection, recognition, and tracking. Surveillance begins with accepting video input from the camera.

The video is split into frames, where each frame becomes the input for the MTCNN face detector. We modified MTCNN so it can detect a masked face. Once the MTCNN detects a face bounding box, the detected face is continued to face recognition. VGGFace2 converts the face feature into a vector feature. Then, it compared with the face database using One-Shot Learning.

Each face is labeled according to the matching face in the database. If a face is not recognized, then it is labeled as Unknown. After recognition, a face is tracked using the D3S tracker, the best tracker at VOT Challenge 2020 [19] for accuracy and speed. If the face detector detects more than one face, the system can prioritize tracking the [Unknown] face. If two or more faces are labeled as Unknown, the first detected face is tracked first. Then, the system back to the face detection phase when the face disappears from the camera. The proposed system can be seen in Figure 3.

3.1 Face detection

The proposed surveillance system utilizes MTCNN as face detectors. The MTCNN is a deep-learn-based detector that detects faces by searching and generating five landmarks: left eye, right eye, nose, and left and right corner of the mouth. An illustration of how MTCNN works in the surveillance system can be seen in Figure 4.

The MTCNN consists of four steps which are pre-processing, Proposal Network (P-Net), Refinement Network (R-Net), and Output Network (O-Net). Pre-processing step changes image size using the pyramid method. P-Net is a Fully Convolutional Network (FCN) that generates a proposal window. The windows are probably overlapped. The R-Net applies Convolutional Neural Network to merges that overlapping window. After R-Net is done, MTCNN can classify the presence of faces in the candidate window. Finally, the Output Network (O-Net) issues a more accurate bounding box result. It describes the face's details using the five facial directions previously mentioned.

3.2 Face recognition and One-Shot Learning

VGGFace is a Visual Geometry Group (VGG) model of two million facial data. The Triplet loss activation function is used to save the feature vector and utilize it in predicting the face identity. This method is also known as the face embedding method.

VGGFace was upgraded into VGGFace2 in 2017. VGGFace2 is trained using a new face dataset consisting of 3.31 million data. VGGFace2 applies ResNet-50 and SE-ResNet-50 architectures. The model also uses 2048 feature vectors for face descriptors. Finally, a face descriptor is utilized to calculate the similarity distance of faces using the Cosine similarity formula.

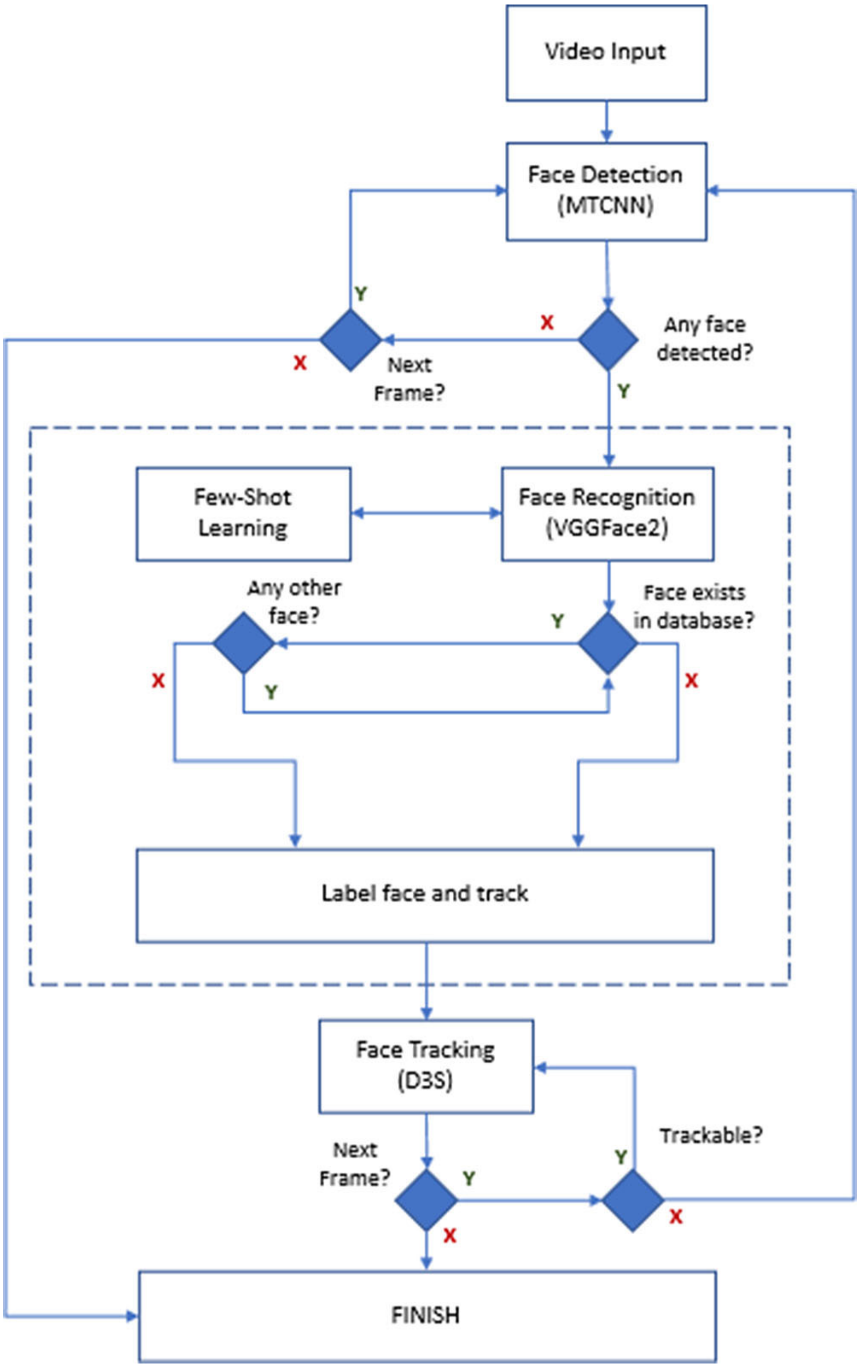


Fig. 3. The proposed surveillance system framework

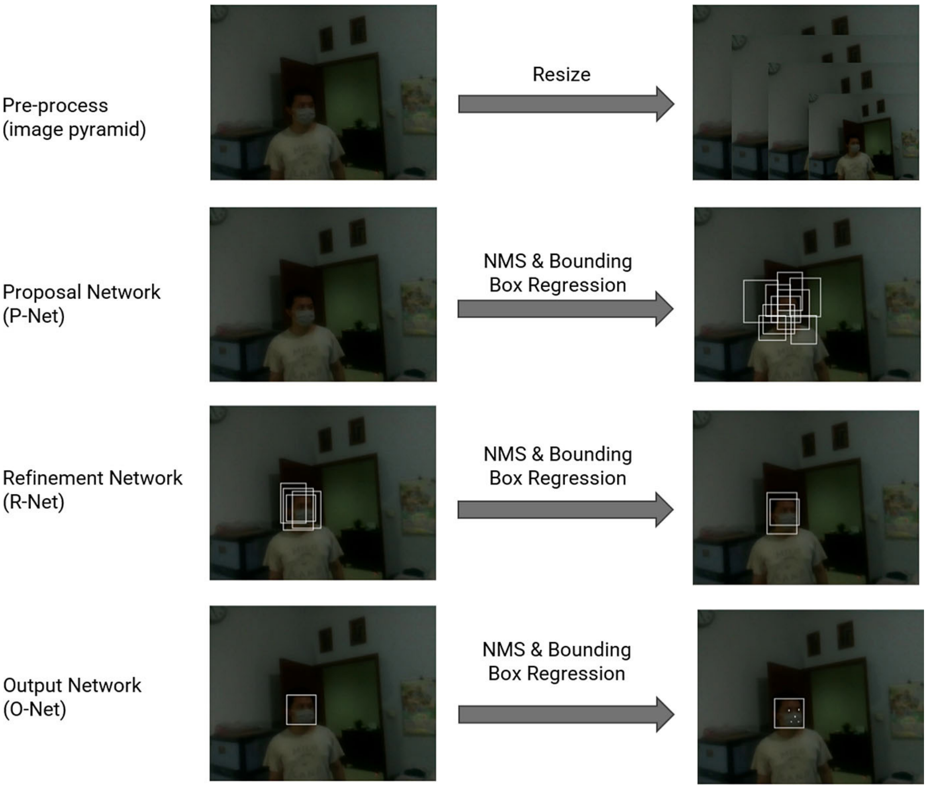


Fig. 4. Face detector MTCNN that used in the proposed surveillance system

The proposed surveillance system distinguishes face by integrating the One-Shot Learning learning method into VGGFace2. One-Shot Learning is a classification model training method with small amounts of training data but tries to classify large amounts of new data. The One-Shot Learning method is suitable for face recognition because the unique human face makes it easy to distinguish.

Figure 5 illustrates the One-Shot Learning mechanism. First, we collect face images of the same person's identity. A collection of faces with the same identity is converted into several chunks of data. Each chunk is proceed using VGGFace2 to produce a batch of feature vectors. The average value of each batch feature vector is used as a distance comparison or referred to as the similarity score. The score is stored with the associated identity in the pickle file. The score is reloaded to be compared with the data during the face inference process.

Figure 6 shows the inference process of face recognition. The One-Shot Learning employ face features value that is given as a group of recognized identities. The feature value is then calculated as the similarity score with the detected face feature value to determine the identity of the detected face in the input image.

3.3 Face tracking

Discriminative Single-Shot Segmentation or D3S is a visual object tracker that combines the segmentation mask model with the correlation filter tracker. The correlation filter tracker is excellent in terms of processing speed. However, it has low robustness when the object changes in shape and size. On the other side, the segmentation mask model can solve deformation and changes in shape or size. So it is suitable for surveillance systems to combined segmentation model and correlation filter. The combination can deal with targets that appear in multiple cameras at different angles dynamically.

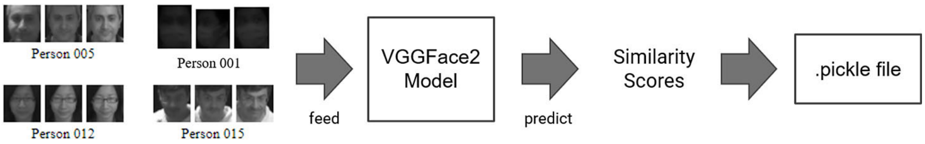


Fig. 5. Integrated VGGFace2 and One-Shot Learning method

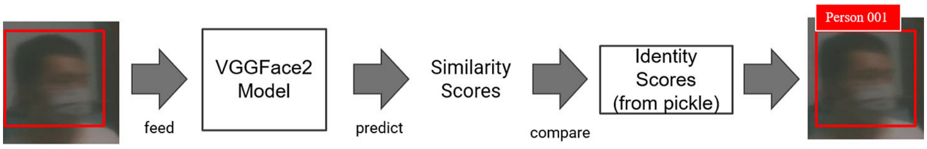


Fig. 6. Inference process of face recognition

3.4 Evaluation method

The proposed surveillance system is evaluated based on face recognition and tracking performance. Face recognition performance is described in confusion matrix variables: True Positive, True Negative, False Positive, and False Negative. The definition of each evaluation for face recognition can be seen in Table 1:

Table 1. Evaluation variable for face recognition

Variables	Value
True Positive (TP)	Face recognized and labeled as the identity in the database
True Negative (TN)	The face is not identified and given as the [Unknown] label
False Positive (FP)	The face is given an incorrect identity label (mismatch)
False Negative (FN)	The face is given the [Unknown] label even though the identity is in the database. The face is labeled, but no actual identity is stored

We calculate several evaluation parameters from the four variables above, namely accuracy, specificity, sensitivity, and precision formulated standardly in [20]. Furthermore, the proposed surveillance system also implements a long-term tracker. So tracker is evaluated using the standard evaluation of the long-term tracker [21]. This method is commonly used in the Visual Object Tracker (VOT) 2020 challenge contest [19]. The evaluation metrics used are shown in equation 1, 2, and 3:

$$\Pr(\tau\theta) = \frac{1}{N_P} \sum_{t\in\{t:At(\theta t)\neq\varnothing\}} \Omega(At(\theta t), Gt), \tag{1}$$

$$\mathrm{Re}(\tau\theta) = \frac{1}{N_g} \sum_{t\in\{t:Gt\neq\varnothing\}} \Omega(At(\theta t), Gt), \tag{2}$$

$$F(\tau\theta) = \frac{2\,\Pr(\tau\theta)\,\mathrm{Re}(\tau\theta)}{\Pr(\tau\theta) + \mathrm{Re}(\tau\theta)} \tag{3}$$

Pr(*τθ*), Re(*τθ*) and F(*τθ*) are variables that represent the Precision, Recall and F-Score respectively. They are influenced by a classification threshold *τθ* which determines the tracking confidence (detection certainty). The *Gt* states the ground truth position, (*θ*) states the tracker's prediction position, influenced by the prediction certainty score *θ* in each frame *t*. *Ng* stated the number of frames that apply *Gt* ≠ ø. There is a bounding box ground truth in the image. *Np* states the number of frames that apply *At* (*θ*) ≠ ø (there is a bounding box prediction results in the image). *Ω* ((*θ*)) states Intersection over Union (IoU) between ground truth and the predicted results, which are formulated in equation 4:

$$\Omega(At(\theta t), Gt) = \frac{At(\theta t) \cap Gt}{At(\theta t) \cup Gt} \tag{4}$$

4 Experiments and result

Section 4 evaluates the performance of the proposed system consist of face detection, recognition, and tracking subsystem. This section describes the dataset and experiment setup followed by experiment results. In the experiment result, our proposed system was compared with ground truth. Performance of face detection and recognition is measured on accuracy, specificity, sensitivity, and precision. Followed by tracking performance is showed in precision, recall, and F-score. Overall performance was analyzed with discussion along with each step result snapshot of the proposed system.

4.1 Datasets and experiment setup

The proposed surveillance system uses public and self-retrieved datasets. We use a Conv2B sequence from the People in Indoor Room with Perspective and Omnidirectional cameras (PIROPO) [22]. The Conv2B represents indoor conditions with various human activities, including walking across the room, standing indoors, and sitting indoors. The Conv2B illustrated our assumption that strangers (non-residents) do suspicious activities like waiting in front of the fence.

We also self-collected the masked person dataset, namely the *SelfTake* sequence. The dataset is taken using the Raspberry Pi 3 and PiCamera tools. This dataset represents the proposed surveillance system's actual conditions which are unconstrained and low-quality surveillance cameras. The Raspberry Pi 3 and Pi Camera specifications can be seen in Table 2. Single masked persons enter and leave a room while occasionally stopping inside the observation area for several seconds. The capture condition is specified in Table 3:

Table 2. Raspberry Pi 3 and Pi Camera spesification

Raspberry Pi 3 Model B		**Pi Camera**	
Processor	Quad Core 64-bit 1.2 GHz	Resolution	5 MP
RAM	1 GB	Video	10 fps
Wifi	Supported	Picture	1280×1024 pixels
PiCamera	Supported	Interface	Serial Camera Interface

Table 3. Camera settings for video capture using Raspberry Pi 3

Exposure	**Night**
ISO	800
Brightness (%)	55
Contrast	10
Sharpness	15
Additional Lighting	Yes

All datasets are self-annotated, including the cropped faces for the face recognition task. Several snapshots of our sequences can be seen in Figure 7. Figure 7a represents the *SelfTake* sequence. The target person wearing the mask is seen on camera. Figure 7b is a sample of Conv2B from the PIROPO dataset. A single person walks into the room, stands idly, and walks out several times for the first half, followed by another person doing the same routine. The proposed surveillance framework runs on Google Colab. The hardware specifications are Intel Xeon CPU 2.20 GHz, 16 GB RAM, and a Tesla T4 graphic card.

4.2 Experimental results

We evaluate face detection along with recognition performance. We count every time the detector detects faces and comparing with the ground truth. Table 4 provides a summary of the face identification:

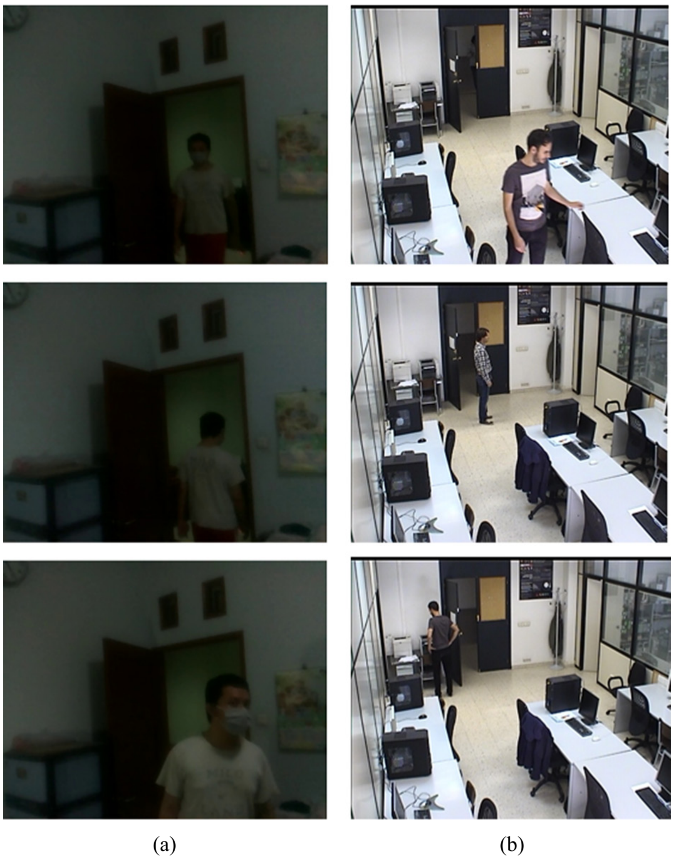


Fig. 7. Samples from sequences used in the experiment

Table 4. Face detection and recognition results

Sequences	Accuracy (%)	Specificity (%)	Sensitivity (%)	Precision (%)
SelfTake	100	–	100	100
Conv2B	57.14	100	0	–

Table 4 shows that the face recognition system can recognize more than half of the faces across sequences, with sequence *SelfTake* displaying a hundred percent (100%) detection and recognition accuracy. Every face is recognized as the correct identity in the *SelfTake* sequence. As a result, the sensitivity and precision obtain 100%. No specificity value can be counted on the *SelfTake* sequence because of no [Unknown] label or mismatched label during the experiment.

On The Conv2B sequence, there are two 'Unknown' labeled persons. In this case, some faces are recognized as existing identities on the database, making the accuracy lower than the *SelfTake* sequence with 57.14%. In addition, some detected faces are given the identity from the database instead of the [Unknown] label, decreasing the sensitivity value to zero (0%). Similar to specificity on the *SelfTake* sequence, the precision on Conv2B can not be counted since no mismatch or correct identification. On the other side, the proposed system obtains good results in specificity, which is 100% cause there is no mismatched face on the Conv2B. Based on the above results, we conclude that the face identification model can recognize and identify faces. The face recognition system's performance is about 64 percent, with a total of 16 faces predicted correctly from a total of 25 detected faces during surveillance.

Table 5. Face tracking result

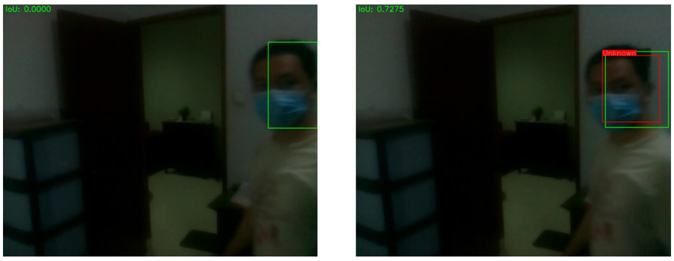
Sequences	Precision	Recall	F-Score
SelfTake	0.656	0.645	0.650
Conv2B	0.547	0.439	0.487

The surveillance system is also evaluated under tracking performance measurement. Table 5 shows the results of long-term tracking evaluations using precision, recall, and F-score evaluation metrics. Table 5 shows that the highest evaluation value belongs to the *SelfTake* sequence, with an F-score of 0,650. The proposed system is considerably competitive in long-term tracking. The state-of-the-art tracker still achieves F-Score ranged from 0.400 to 0.600 [19] even if the results are not run on the same dataset and computing platform. The occlusion and out-of-view problems make long-term tracers give poor results when evaluated using quantitative metric evaluation and ground truth bounding box. It is due to the varying re-detection capabilities for each different system. The earlier detection than ground truth decreases the precision value, while late detection decreases the recall value. Figure 8 shows an example long term tracking of our proposed method.

4.3 Discussion

The proposed surveillance system is successfully implemented, as shown in Figure 9. Face detection generates a bounding box of detected faces. The faces are successfully detected by MTCNN, including faces wearing masks which is a challenging obstacle in detection tasks. Furthermore, the proposed system has an advantage on mobility since observation is done using a low-cost camera. The surveillance system then recognizes and tracks the detected person by displaying a person's name or ID.

Some challenges occur on face detection, recognition, and tracking in the experiments. The first problem is that the ability of the D3S tracker as a general object tracker is too adaptive, making a variety of bounding box sizes. Bounding box size variations affect tracking accuracy. In addition, D3S uses a Discriminative Correlation Filter (DCF), so the resulting bounding box can affect the features studied for object tracking in the next frame. Another problem is the reliability of MTCNN in detecting faces in dim lighting.



(a) (b)

Fig. 8. Example of the long term tracking result

Another challenge is face recognition. The quality of facial features captured by the face detector is not detailed. Lack of detail in the facial features studied makes faces less unique, increasing the risk of facial identification errors. In addition, identifying faces wearing masks is proved challenging because more than half of the face is covered by masks, dominating the features studied by facial identification models. Finally, using the grayscale color feature reduces the number of features retrieved and indirectly generalizes faces with no identity. We also suggest train using a larger number of faces with varying shooting conditions.

The authors recommend using a camera with minimal 1280x1024 pixels to detect and track humans or other objects that do not require a high level of detail. PiCamera already has a reasonably sharp image quality, especially for indoor surveillance systems. But we need higher image quality for future work, especially if you want to make a tracking application outside the home with various conditions.

5 Conclusion

We propose an automatic surveillance system with facial detection, recognition, and face tracking. The proposed system consists of three main modules: MTCNN for face detection, VGGFace2 for facial recognition and identification, and D3S for face tracking. The proposed method detects all faces found in the observation area and tracks when the face is not recognized in the database. The back-end method runs in a cloud platform that making mobile surveillance using an edge device, Raspberry Pi 3 camera is possible. We evaluated the proposed system using two video datasets. The experiment results demonstrate the promising performance of the proposed approach.

For future work, a surveillance system that can deal with mask faces grows explosively. Object detection becomes a key feature before entering the face recognition and tracking phase. Further research can implement object detection methods like YOLO alongside InsightFace for face recognition. Future works must collect more data on masked faces to increase face detection, recognition, and tracking accuracy.

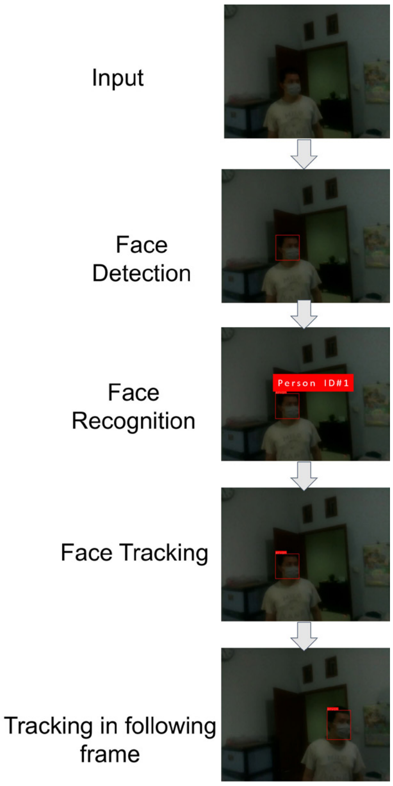


Fig. 9. The overall result of the proposed surveillance system consists of face detection, recognition, and tracking under the unconstrained image

6 Acknowledgment

The author would like to thank Universitas Indonesia for funding this research through PUTI Grant with contract number NKB-3225/UN2.RST/HKP.05.00/2020.

7 References

[1] B. Stickle, "Crime Rates in a Pandemic: The Largest Criminological Experiment in History," *American Journal of Criminal Justice*, vol. 45, pp. 525–536, 2020. https://doi.org/10.1007/s12103-020-09546-0

[2] N. T. Wei, A. S. Baharudin, L. A. Hussein, and M. F. Hilmi, "Factors Affecting User's Intention to Adopt Smart Home in Malaysia," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 13, no. 12, pp. 39–54, 2019. https://doi.org/10.3991/ijim.v13i12.11083

[3] S. Kakarla, P. Gangula, M. S. Rahul, C. S. C. Singh, and T. H. Sarma, "Smart Attendance Management System Based on Face Recognition Using CNN," in *IEEE-HYDCON*, 2020, pp. 1–5. https://doi.org/10.1109/HYDCON48903.2020.9242847

[4] L. Rai and Z. Wang, "Software Development Framework for Real-Time Face Detection and Recognition in Mobile Devices," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 14, no. 4, pp. 103–120, 2020. https://doi.org/10.3991/ijim.v14i04.12077

[5] J. Zheng, R. Ranjan, C. Chen, J. Chen, C. D. Castillo, and R. Chellappa, "An Automatic System for Unconstrained Video-Based Face Recognition," *IEEE Transactions On Biometrics, Behavior, And Identity Science*, vol. 2, no. 3, pp. 194–209, 2020. https://doi.org/10.1109/TBIOM.2020.2973504

[6] G. T. S. Draughon, "Implementation of a Computer Vision Framework for Tracking and Visualizing Face Mask Usage in Urban Environments," in *IEEE International Smart Cities Conference (ISC2)*, 2020. https://doi.org/10.1109/ISC251055.2020.9239012

[7] A. Negi, "Face Mask Detection Classifier and Model Pruning with Keras-Surgeon," in *IEEE International Conference on Recent Advances and Innovations in Engineering—ICRAIE*, 2020. https://doi.org/10.1109/ICRAIE51050.2020.9358337

[8] V. Jain, M. S. Pillai, L. Chandra, R. Kumar, and M. Khari, "Sensor systems CamAspect: An Intelligent Automated Real-Time Surveillance System With Smartphone Indexing," *IEEE Sensors Letter*, vol. 4, no. 10, pp. 3–6, 2020. https://doi.org/10.1109/LSENS.2020.3019172

[9] I. S. Areni, A. Waridi, C. Yohannes, A. Lawi, and A. Bustamin, "IoT-Based of Automatic Electrical Appliance for Smart Home," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 14, no. 18, pp. 204–212, 2020. https://doi.org/10.3991/ijim.v14i18.15649

[10] K. Zhang, Z. Zhang, Z. Li, S. Member, Y. Qiao, and S. Member, "Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks," *IEEE Signal Processing Letter*, vol. 23, no. 10, pp. 1499–1503, 2016. https://doi.org/10.1109/LSP.2016.2603342

[11] A. Lukě, "D3S – A Discriminative Single Shot Segmentation Tracker," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) D3S*, 2020, pp. 7131–7140.

[12] J. Whitehill, G. Littlewort, I. Fasel, M. Bartlett, and J. Movellan, "Toward Practical Smile Detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 2106–2111, 2009. https://doi.org/10.1109/TPAMI.2009.42

[13] W. Bu, "A Cascade Framework for Masked Face Detection," in *IEEE 8th International Conference on CIS & RAM*, 2017, pp. 458–462. https://doi.org/10.1109/ICCIS.2017.8274819

[14] S. Lin, L. Cai, X. Lin, and R. Ji, "Masked face detection via a modified LeNet," *Neurocomputing*, vol. 218, pp. 197–202, 2016. https://doi.org/10.1016/j.neucom.2016.08.056

[15] S. Ejaz, R. Islam, and A. Sarker, "Implementation of Principal Component Analysis on Masked and Non-masked Face Recognition," in *International Conference on Advances in Science, Engineering and Robotics Technology*, 2019. https://doi.org/10.1109/ICASERT.2019.8934543

[16] S. Ge, J. Li, Q. Ye, and Z. Luo, "Detecting Masked Faces in the Wild with LLE-CNNs," in *IEEE Conference on Computer Vision and Pattern Recognition Detecting*, 2017, pp. 426–434. https://doi.org/10.1109/CVPR.2017.53

[17] F. Ding, "Masked Face Recognition with Latent Part Detection," in *The 28th ACM International Conference on Multimedia*, 2020, pp. 2281–2289. https://doi.org/10.1145/3394171.3413731

[18] M. Kristan, J. Matas, G. Nebehay, F. Porikli, and L. Cehovin, "A Novel Performance Evaluation Methodology for Single-Target Trackers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 11, pp. 2137–2155, 2016. https://doi.org/10.1109/TPAMI.2016.2516982

[19] M. Kristan, A. Lukě, O. Drbohlav, L. He, and Y. Zhang, *The Eighth Visual Object Tracking VOT2020 Challenge Results*. Springer, 2020.

[20] A. O. Christiana and B. A. Gyunka, "Optimizing Android Malware Detection Via Ensemble Learning," *International Journal of Interactive Mobile Technologies (iJIM)*, vol. 14, no. 9, pp. 61–78, 2020. https://doi.org/10.3991/ijim.v14i09.11548

[21] A. Lukě and L. Cehovin, “Now You See Me: Evaluating Performance in Long-term Visual Tracking,” *arXiv preprint*, 2018.

[22] R. Carlos, P. Carballeira, F. Jaureguizar, and N. García, “Robust People Indoor localization With Omnidirectional Cameras using a Grid of Spatial-Aware Classifiers,” *Signal Processing: Image Communication*, vol. 93, no. January, p. 116135, 2021. https://doi.org/10.1016/j.image.2021.116135

8 Authors

Ervan Adiwijaya Haryadi received Bachelor’s degree in computer science undergraduate from the University of Indonesia. Currently, he works as a research assistant at the Faculty of Computer Science, Universitas Indonesia. His research interests including computer vision, robotics, and the internet of things (e-mail: erhar23@gmail.com).

Grafika Jati received his BS and MSc in Computer Science from the Universitas Indonesia in 2014 and 2016. Currently, he works as a researcher and lecture at the Faculty of Computer Science, University of Indonesia. His research interest includes visual object tracking and autonomous robot.

Ario Yudo Husodo holds Bachelor’s and Master’s degrees in Informatics from Institut Teknologi Bandung, Indonesia. He works at the University of Mataram, Lombok, Indonesia as a lecturer. While writing this paper, he acts as a Ph.D. student in the Faculty of Computer Science at the Universitas Indonesia. His research interest is Computer Vision and Intelligent systems.

Wisnu Jatmiko received his BS in Electrical Engineering and MSc in Computer Science from the Universitas Indonesia in 1997 and 2000. In 2007, he received his Dr. Eng. Degree from Nagoya University, Japan. In 2020, he was the head of the IEEE Indonesian Section. He is a Full Professor at the Faculty of Computer Science, University of Indonesia. His research interest is autonomous robots, optimization, and traffic monitoring systems.

Article submitted 2021-03-18. Resubmitted 2021-09-03. Final acceptance 2021-09-03. Final version published as submitted by the authors.