Face Mask Detection using Convolutional Neural Network

1. Summary

1.1. Motivation

In 2019, the world witnessed the pandemic Coronavirus Disease also known as COVID-19. To prevent the spread of this infectious disease, the governing body promotes public awareness of the need to reduce the risk of infection by adhering to and putting into practice the 3M movement, which calls for mask wear, keeping distance, and hand washing. Before COVID-19, wearing a mask for a long time was advised to stop the spread of saliva (droplets), particularly for those suffering from respiratory illnesses or the flu. So to detect a large portion of the population, an application is needed. Solutions for mask detection are not limited to managing the application of health protocols. Numerous comparable apps had been created before the COVID-19 outbreak to track employees within an organization for security-related reasons. The technology can transmit alerts to relevant security officers via telegrams in addition to using deep learning for mask detection. The Raspberry Pi 3 B+ used in this study has limited capacity and can only operate video streaming at a frame rate of 20 to 32 frames per second. This is insufficient for usage as a CCTV or streaming device. Since the notification uses Telegram, which only distributes text message data and photos linearly, the system is unable to store history for quarterly reporting.

1.2. Contribution

The goal of this research is to create a monitoring application that can track and identify mask wearers in real-time. Additionally, this technology notifies the concerned security officer using a mobile application that they can access. Thus, it is envisaged that security personnel can keep an eye out for any violations of health in real-time.

1.3. Methodology

This study will use the Convolutional Neural Network (CNN) approach in conjunction with the MobileNet V2 algorithm to detect masks. The idea is to create an application that can not only recognize masks but also alert the user to potential threats. The suggested solution is designed to address the issues that currently exist. Additionally, since the system will be constructed in this research employs a storage database, it is expected that the application will have one so that in the future, should the need arise to issue data in the form of reports or statistical data, the system will be able to display the data. After identifying faces, the system will compare them with the face mask classifier. The image of the face will be taken and kept in the database. The database's data status will be updated by data detection. When data is detected, the cron agent will check the data and send a notification. The alert originates from Firebase Cloud Messaging and uses data detection to gather information. The system notifies users that the data is unavailable if it cannot be located. The system will show a list of detected data if any are found. The system will then show comprehensive data.

1.4. Conclusion

The study's findings on object detection can be used to draw the following conclusions: the deep learning model with MobileNetV2 obtained an accuracy of 99% and data loss of 3.8% with 1000 datasets. The detection tool tested well at a distance of 50 cm to 1 m with multiple objects. It functioned well and had a good degree of accuracy, but because it uses a low-resolution camera, it is unable to detect objects well at distances greater than 1 meter. The tool's 24-hour test results indicate that using a cooler or fan is preferable to not using one and the notification agent system's test results show that everything is operating as intended, with the scheduling agent/cron scheduler scheduled to run every 5 seconds to review the database and deliver notifications to Firebase Cloud Messaging. The application's black box testing also demonstrates that everything is operating as intended.

2. Limitations

2.1. First limitation

Even at one meter distance, the tool is still quite good at detecting objects. Because the camera used in this study had a low resolution and a low frame rate, the tool lost its ability to detect objects at distances greater than one meter and began to produce inaccurate results.

2.2. Second limitation

Given that the test set is still a subset of the training set, accuracy approaches 99%. This accuracy level may therefore differ for real-time data.

3. Synthesis

The goal of future research can be to train the MobileNet model to achieve high accuracy and low loss values by expanding the number of mask datasets with different types and use positions. To prevent notifications from being sent with the same object, a face recognition function can be added. It can recognize repeated object images. Objects can be detected for accurate long-distance recognition using a CCTV camera with high resolution and frames per second. For tool management and user management settings, an application for a monitoring management system can be created. Additionally, a dashboard application that uses the device filter or date filter methods to generate reports regularly can be developed. The scheduling/cron technology that is currently in use can be developed into real-time gateway socket technology for push notifications.