Journal of Technological Sciences (JTS) Volume 1, Issue 2, 2023

Chinhoyi University of Technology
Journal of Technological Sciences
http://journals.cut.ac.zw/index.php/jts

COVID-19 spread regulator robot - An automatic robot that detects face mask, check human body temperature, and hand sanitizes people at entrance point

Martin Masheka[1]*, Hosia Muchingami[2], Andrew Sama[3], Kudakwashe Mapfumo[4], Shakemore Chinofunga[5]

mashekamartin@gmail.com[1*]

[1]Department of Engineering Research, Zimbabwe Centre for High Performance Computing, University of Zimbabwe, 630 Churchill Avenue, Mount Pleasant, Harare

[2,3,4,5]School of Engineering Sciences and Technology, Chinhoyi University of Technology, Private Bag 7724, Chinhoyi, Zimbabwe

Abstract

Aim - From 3 January 2020 to 6:19pm CEST on April 21, 2022, there were 247,336 confirmed cases of COVID-19 in Zimbabwe, with 5,466 deaths, according to WHO. Despite the fact that there are remedies for illnesses and that our civilization has progressed by great strides, the most powerful and effective weapon society has against this virus, which affects not only health but also economy, politics, and social order, is the prevention of its spread. This research aims at developing an automatic robot system that enforce minimum spread of the virus by checking body temperature, hand sanitizing people, and check proper wearing of face masks before giving people access to restricted premises.

Subjects and methods – The research made use of ultrasonic sensor for hand object detection, MLX90614 sensor for contactless temperature checking, submissible pump to dispense hand sanitizer, and solenoid lock for automatic locking and unlocking of the entrance. This research has built a python model that detects the presents and proper wearing of face masks using MobileNetV2 algorithm and implement the model using OpenCV. The model was implemented in the bigger hardware system. Temperature sensor was calibrated using normal contactless temperature sensors used in public.

Results – Hand object detection was successfully achieved with 100%, temperature checking was successful with a variance of +/-5 degrees Celsius, sanitizer dispenser worked 100% correct, and auto un/locking was also 100% accurately functional while the face mask detection model had 98% for accuracy, F1-score, precision and recall. In actual functionality of the system, the model could correctly detect proper wearing of face masks for 90 people in a total of 100 people.

Conclusion – Therefore, the model proved to be fully functional with high accuracy score as compared to others that make use of Convolutional Neural Networks (CNN). The model could also be implemented for full functionality in a robotic system that works with zero human intervention. The system benefit communities and organizations attempting to limit the spread of COVID-19 because the control will be done by machines, putting personnel at a lower risk than personally inspecting people at access points.

Introduction

Oumina et al. (2020) reported that the COVID 19 (Coronavirus) pandemic has impacted nearly every country and has had a considerable impact on healthcare facilities and treatment systems. Governments regard public health to be a primary concern. As a result, the adoption of many innovative technologies to address multiple challenges associated to this viral pandemic is required. The spread of the coronavirus could be slowed with the use of masks (Das et al., 2021), hand sanitizing or washing hands (Singh et al., 2020), and frequent temperature checking. Persons should use face masks if they have respiratory symptoms or are caring for people who have symptoms, according to the World Health Organization (Feng et al., 2020). Furthermore, several public service providers insist that clients only use their services if they wear masks, have normal temperature and have sanitized their hands.

In this paper, we proposed a system based on TensorFlow, keras, applications, MobileNetV2 algorithm to classify whether face image has a face mask on correct position or not. An ultrasonic sensor was also used to detect human hands while a wireless temperature sensor was used to check body temperature. A submissive pump was used to dispense liquid hand sanitizer to detected hands. Therefore, the system developed could automatically detect human hands, check their temperature before sanitizing them. Once, normal temperature is detected, the system triggers a camera to switch on and check the presents of a human face with a face mask on the correct position. If all conditions are satisfied, the system will automatically unlock the entrance for ten seconds to allow user to pass through, otherwise the system will notify about the anomaly audibly.

Related work

March 2020, most of the governments rolled out movement control order leading to closure of businesses except those providing essential service (Tang, 2022; Nyanga & Zirima, 2020).

Self-isolation, international travel bans, and quarantine of close contacts and people who test COVID-19 positive are among the other measures taken to combat the spread of COVID19. These measures resulted in the temporary closure of schools, colleges, and institutions around the world, affecting about 1,048,817,181 students (Feng et al., 2020). Such measures seem to have a positive impact in controlling the spread of the virus, however, the measures contribute to negative effects on the economy and welfare of many residents.

Face mask detection

Wearing face masks is the most common technique of preventing the spread of corona virus, according to Singh et al. (2021), Hussain et al. (2021), Nowrin et al. (2021), Melnick et al., 2020), and (Yadav, 2020). Object identification algorithms utilizing deep models (Ma & Chen, 2020) have made amazing development in computer vision in recent years, and they are potentially more capable than shallow models in handling complicated problems (Goodfellow et al., 2016). Deep learning object detection models (Farfade et al., 2015) are presently classified into two groups: I two-stage detectors like R-CNN (Liu & Stathaki, 2018), Fast R-CNN (Girshick, 2015), and its variations, and (ii) one-stage detectors like YOLO (Liu & Stathaki, 2018) and SSD. Two-stage detectors execute detection in phases, with the first step computing proposals and the second stage classifying them into object categories. Some approaches, like YOLO and SSD MultiBox, regard detection to be a regression issue and just look at the image once for detection.

In the proposed system, a face mask detection model was trained using TensorFlow based algorithm called MobileNetV2. 22130 images of human faces with face masks put on properly and 23463 human face images without properly put on face masks were used to build the model. The model seems to produce good results in face masks detection when implemented using OpenCV on a live video stream.

Methods and tools

Object detection

Figure 1: Ultrasonic sensor

An ultrasonic sensor is a device that uses ultrasonic sound waves to determine the distance to an item. A transducer is used in an ultrasonic sensor to emit and receive ultrasonic pulses that relay information about the proximity of an item. Boundaries reflect high-frequency sound waves, creating different echo patterns. The ultrasonic sensor was used to detect the presents of hands below hand sanitiser dispenser and MLX90614 wireless temperature sensor. In this project, its main function is to trigger detection devices for temperature checking and hand sanitization whenever an object is detected 20cm away from the detection devices.

Temperature checking

Figure 2: MLX90614 wireless temperature sensor

In this research, an MLX90614 wireless temperature sensor measures human body temperature through the detected hand object. Therefore, an ultrasonic sensor detects a human hand object, it tells the controller that there is an object detected and the controller triggers MLX90614 contactless temperature sensor and the pump. This then allows the system to check human body temperature before hand sanitisation. The MLX90614 is a Contactless Infrared (IR) Digital Temperature Sensor that measures the temperature of a specific object between -70° C to 382.2°C. The sensor measures the temperature of the object using infrared rays and interacts with the microcontroller using the I2C protocol.

Hand sanitization

Figure 3: Submissible 5V water pump

A submissible 5v powered water pump was used to pump liquid hand sanitizer from a container through a tube with a small nozzle point. This is being triggered after an object is detected close to the nozzle, and the objects are assumed to be human hands. The pump works while dipped in liquid material.

Face mask detection

Model building

MobileNetV2, a TensorFlow-based technique, was used to train a face mask identification model. The model was created using 22130 photos of human faces with properly applied face masks and 23463 photographs of human faces without properly applied face masks. Datasets of images were downloaded from different sources on the internet and the images were categorized into two groups (i.e., with mask category and without-mask category). Images were picked from different sources and were of different people in terms of skin complexion and different face shapes, different amount and shape of face hair or whiskers. Figure 1 shows a sample of images used in training the model.

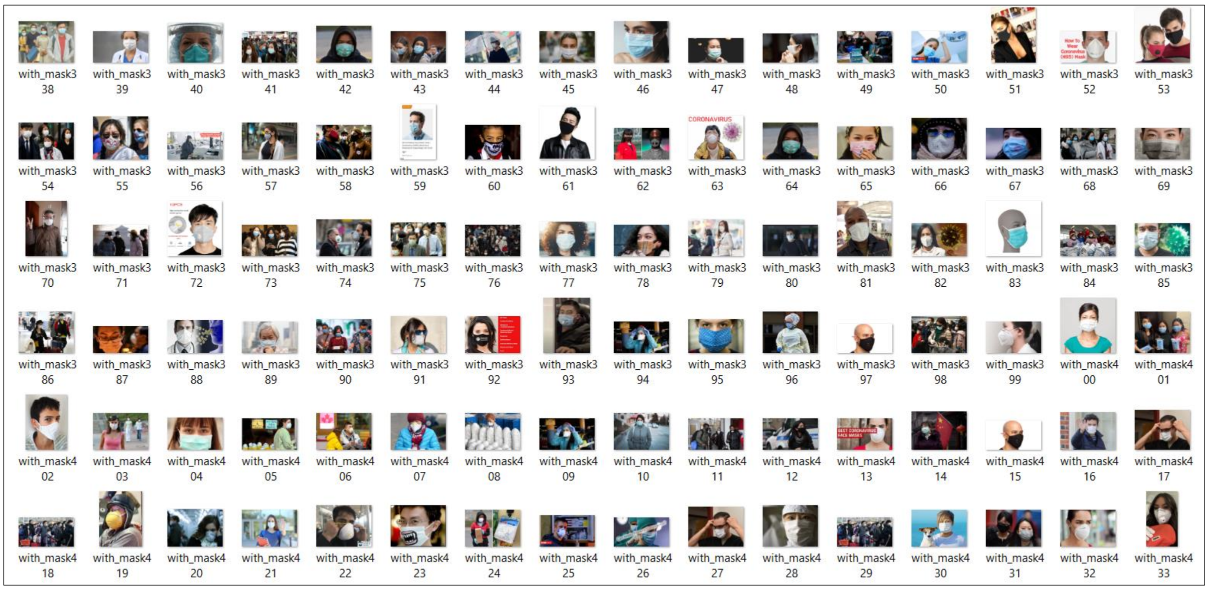

Figure 4: Sample images of people with face masks properly put on

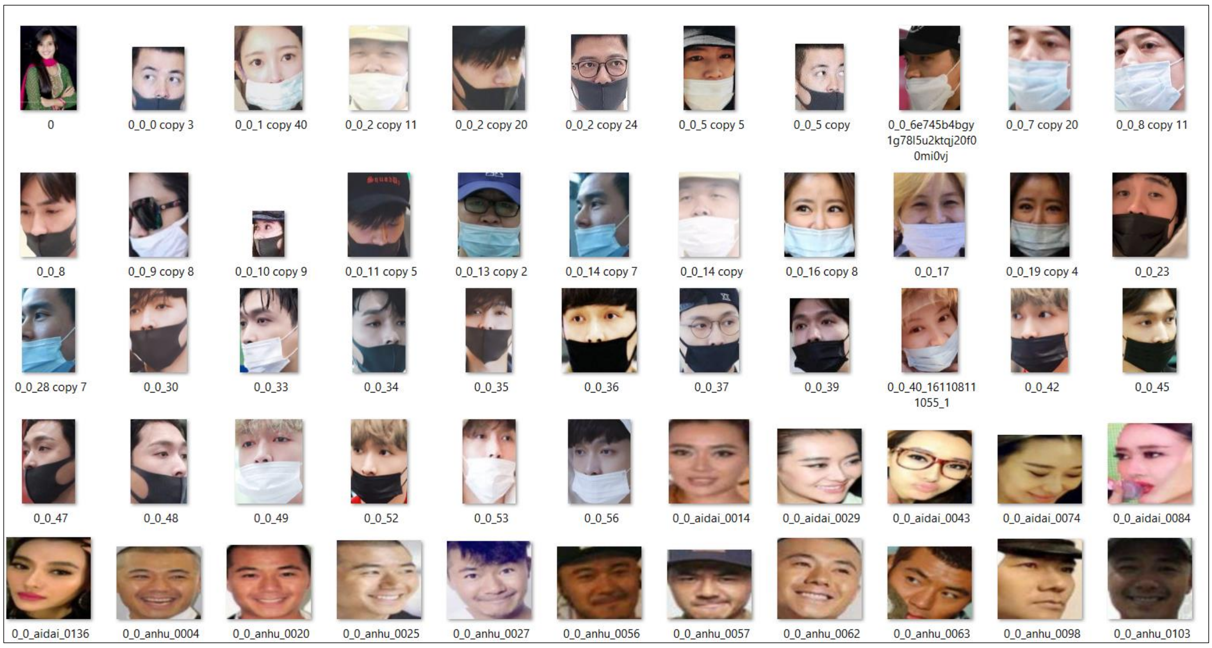

Figure 5: Sample images of people without face masks put on properly

```
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.applications import MobileNetV2
from tensorflow.keras.layers import AveragePooling2D
from tensorflow.keras.layers import Dropout
from tensorflow.keras.layers import Flatten
from tensorflow.keras.layers import Dense
from tensorflow.keras.layers import Input
from tensorflow.keras.models import Model
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.preprocessing.image import load_img
from tensorflow.keras.utils import to_categorical
from sklearn.preprocessing import LabelBinarizer
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from imutils import paths
import matplotlib.pyplot as plt
import numpy as np
import os
```

Figure 6: Imported libraries for face mask detection model building

Figure 3 shows Python libraries involved in building the model that detects proper wearing of face masks. The learning rate was initialized at 1e^{-4}, the number of epochs was 20 and batch size was initialized to 32. The Python program was written to do the following:

i. grab the list of images in the dataset directory,
ii. then initialize the list of data (i.e., images) and class images.
iii. perform one-hot encoding on the labels and construct the training image generator for data augmentation
iv. load the MobileNetV2 network, ensuring the head Fully Connected (FC) layer sets are left off.
v. construct the head of the model that will be placed on top of the base model
vi. place the head FC model on top of the base model (this will become the actual model we will train).
vii. loop over all layers in the base model and freeze them so they will not be updated during the first training process
viii. compile the model
ix. train the head of the network
x. make predictions on the testing set for each image in the testing set we need to find the index of the label with corresponding largest predicted probability
xi. design and show a nicely formatted classification report
xii. serialize the model to disk
xiii. plot the training loss and accuracy

Figure below shows the training epochs and accuracy levels at each epoch.

```
Epoch 1/20
1139/1139 [==============================] - ETA: 0s - loss:
0.2321 - accuracy: 0.9122     WARNING:tensorflow:Your input ran
out of data; interrupting training. Make sure that your dataset
or generator can generate at least `steps_per_epoch * epochs`
batches (in this case, 284 batches). You may need to use the
repeat() function when building your dataset.
1139/1139 [==============================] - 1411s 1s/step -
loss: 0.2321 - accuracy: 0.9122 - val_loss: 0.1281 -
val_accuracy: 0.9554
Epoch 2/20
1139/1139 [==============================] - 1166s 1s/step -
loss: 0.1212 - accuracy: 0.9565
Epoch 3/20
1139/1139 [==============================] - 1116s 980ms/step -
loss: 0.0977 - accuracy: 0.9644
Epoch 4/20
1139/1139 [==============================] - 1105s 970ms/step -
loss: 0.0875 - accuracy: 0.9690
Epoch 5/20
1139/1139 [==============================] - 1136s 997ms/step -
loss: 0.0789 - accuracy: 0.9712
Epoch 6/20
1139/1139 [==============================] - 1221s 1s/step -
loss: 0.0775 - accuracy: 0.9734
Epoch 7/20
1139/1139 [==============================] - 1263s 1s/step -
loss: 0.0739 - accuracy: 0.9731
Epoch 8/20
1139/1139 [==============================] - 1251s 1s/step -
loss: 0.0687 - accuracy: 0.9749
Epoch 9/20
1139/1139 [==============================] - 1478s 1s/step -
loss: 0.0653 - accuracy: 0.9758
Epoch 10/20
1139/1139 [==============================] - 1513s 1s/step -
loss: 0.0645 - accuracy: 0.9766
Epoch 11/20
1139/1139 [==============================] - 1311s 1s/step -
loss: 0.0609 - accuracy: 0.9764
Epoch 12/20
1139/1139 [==============================] - 1338s 1s/step -
loss: 0.0593 - accuracy: 0.9785
Epoch 13/20
1139/1139 [==============================] - 1578s 1s/step -
loss: 0.0575 - accuracy: 0.9792
Epoch 14/20
1139/1139 [==============================] - 1437s 1s/step -
loss: 0.0557 - accuracy: 0.9795
Epoch 15/20
1139/1139 [==============================] - 1609s 1s/step -
loss: 0.0576 - accuracy: 0.9796
Epoch 16/20
1139/1139 [==============================] - 1618s 1s/step -
loss: 0.0550 - accuracy: 0.9802
Epoch 17/20
1139/1139 [==============================] - 1255s 1s/step -
loss: 0.0530 - accuracy: 0.9806
Epoch 18/20
1139/1139 [==============================] - 1428s 1s/step -
loss: 0.0558 - accuracy: 0.9793
Epoch 19/20
1139/1139 [==============================] - 1452s 1s/step -
loss: 0.0539 - accuracy: 0.9800
Epoch 20/20
1139/1139 [==============================] - 1400s 1s/step -
loss: 0.0508 - accuracy: 0.9814
[INFO] evaluating network...
```

Figure 7: Model training epochs with respective epoch evaluations

Actual face mask checking

At this stage, the model trained to detect correct wearing of human face masks was implemented using Python programming language and respective Python libraries as shown in Figure below. Libraries shown here are used only for checking proper wearing of face mask.

```
from tensorflow.keras.applications.mobilenet_v2 import preprocess_input
from tensorflow.keras.preprocessing.image import img_to_array
from tensorflow.keras.models import load_model
from imutils.video import VideoStream
import numpy as np
import imutils
import time
import cv2
import os
```

Figure 8: Libraries used for checking proper wearing face mask

After importing the relevant Python libraries, python code was written to follow the following steps:

i. grab the dimensions of the frame and then construct a blob from it
ii. pass the blob through the network and obtain the face detections
iii. initialize the list of faces, their corresponding locations, and the list of predictions from the face mask network
iv. loop over the detections
v. extract the confidence (i.e., probability) associated with the detection
vi. filter out weak detections by ensuring the confidence is greater than the minimum confidence
vii. compute the (x, y)-coordinates of the bounding box for the object

viii. ensure the bounding boxes fall within the dimensions of the frame
ix. extract the face ROI, convert it from BGR to RGB channel ordering, resize it to 224x224, and preprocess it
x. add the face and bounding boxes to their respective lists
xi. only make a prediction if at least one face was detected
xii. for faster inference make batch predictions on all faces at the same time rather than one-by-one predictions in the above `for` loop
xiii. return a 2-tuple of the face locations and their corresponding locations return a 2-tuple of the face locations and their corresponding locations
xiv. load our serialized face detector model from disk
xv. load the face mask detector model from disk
xvi. initialize the video stream
xvii. loop over the frames from the video stream
xviii. grab the frame from the threaded video stream and resize it to have a maximum width of 400 pixels
xix. detect faces in the frame and determine if they are wearing a face mask or not
xx. loop over the detected face locations and their corresponding locations
xxi. unpack the bounding box and predictions
xxii. determine the class label and color we'll use to draw the bounding box and text
xxiii. include the probability in the label
xxiv. display the label and bounding box rectangle on the output frame
xxv. show the output frame

Automatic lock

A 12V solenoid lock was used for automatic door locking and unlocking. In this project, the solenoid lock was triggered to unlock the door when successful hand sanitization was done on people with normal body temperature and properly wearing their face masks. A solenoid lock is essentially an electronic lock for a standard cabinet, safe, or door. The slug on this 12V solenoid lock has a slanted cut and a nice mounting bracket. The slug draws in when 9-12VDC is applied, so it doesn't stick out and the door can be opened. In this state, it does not utilize any electricity. Automatic door lock systems, such as electronic door locks, are simple to install with the mounting board. This specific solenoid is beautiful and strong.

Program flow diagram

The system starts by initializing hardware detection devices and checks for the presents of a hand object within twenty (20) centimeters away from hand sanitizer dispenser's nozzle where the wireless temperature sensor is also attached. Once a hand is detected, the system controller triggers the temperature sensor to detect body temperature before hand sanitizing the hand. Temperature checking starts first before hands are sanitized; this is because liquid hand sanitizer may have an effect on the temperature of the hands used to check body temperature. The system then checks if temperature is normal (i.e., below 37 according to Hussain et al. (2021)). If body temperature is above 37 degrees Celsius, the system notifies the client and the system administrator and if it is normal, the system proceeds to face mask checking. If the mask is put on properly, the system proceeds to unlock the entrance, otherwise it notifies the client and the client and keep checking for proper face mask wearing. This is demonstrated in the data flow diagram depicted in Figure below.

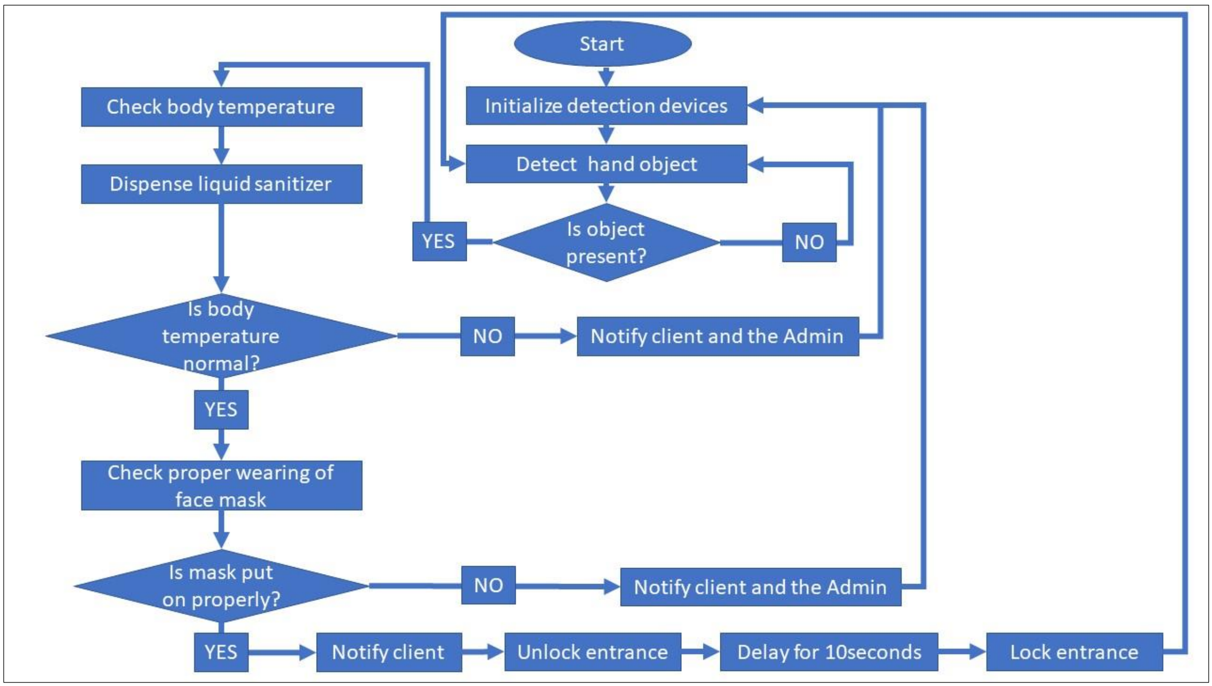

Figure 9: Data flow diagram

System block diagram

The first sensor that triggers the whole system is the ultrasonic sensor that detect objects and sends the signal to the controller. The controller then triggers the temperature sensor for body temperature and the temperature sensor reply with temperature level to complete a two-way communication. Once this is done, the controller triggers the water pump to release some hand sanitizer for a single second. Once this is done and all conditions are satisfied (i.e., the temperature is normal), the controller passes control to the computer CPU that will then run the model that check the presents of a face mask. If the face mask is detected and is put on properly, the system proceeds to open unlock the solenoid lock. The controller and computer CPU communicate serially through a USB cable. The computer CPU and the computer controller have a bidirectional communication channel. The computer's display unit was then used to display temperature levels and human faces. Figure below depicts how data is moving from one device to the other. Directional arrows showing the direction of data flow.

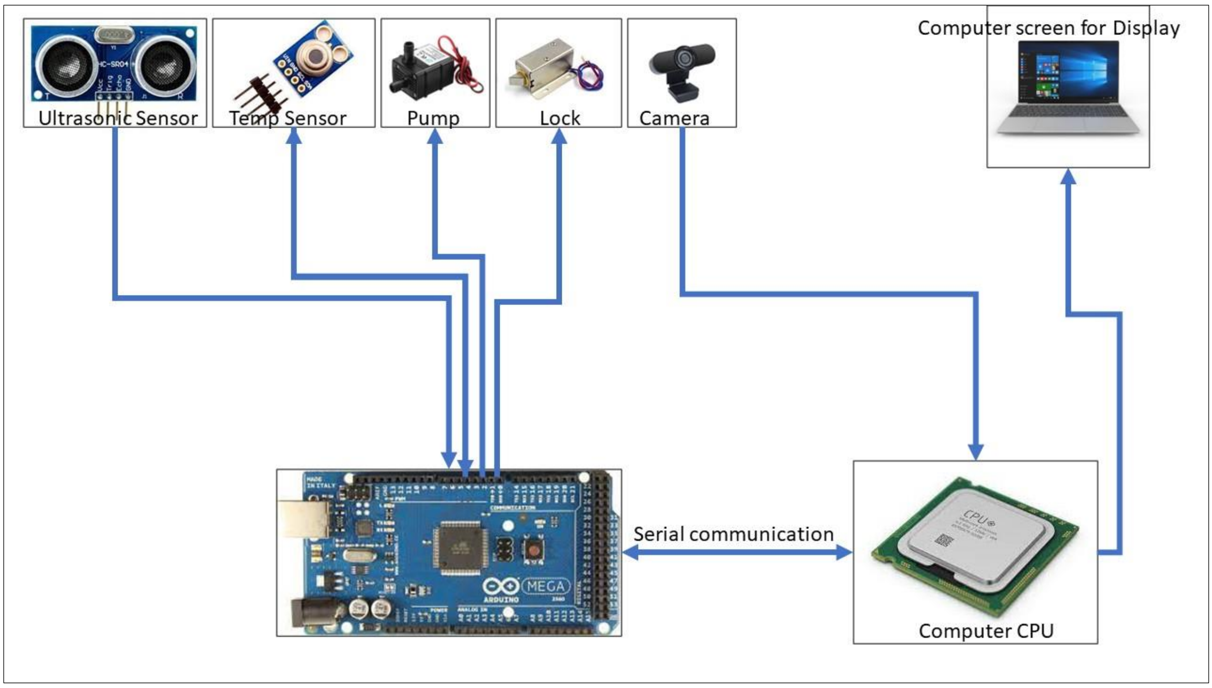

Figure 10: System block diagram

Results

This research shows that, detecting proper wearing of face masks using a python-based model is actually feasible. The model had high evaluation score rates above 98%. The Neural network was evaluated using precision score, recall score, and f1-score. The model has a precision score of 99% for human faces with face mask and 98% for human faces without properly put on face masks. It was the opposite when the network was evaluated using recall score, where 98% was the score for human faces with properly put on face masks and 99% was for properly put on face masks. F1-score came out with the same score for both categories. However, accuracy and the average weight were both 98%, which is a good prediction score.

```
[INFO] evaluating network...
              precision    recall  f1-score   support

   with_mask       0.99      0.98      0.98      4426
without_mask       0.98      0.99      0.98      4693

    accuracy                           0.98      9119
   macro avg       0.98      0.98      0.98      9119
weighted avg       0.98      0.98      0.98      9119

[INFO] saving mask detector model...
```

Figure 11: Network evaluation results

Overly, the system fully worked well as planned by the researchers. Hand objects were detected successfully, liquid hand sanitizer was pumped out successfully, temperature checking was working well, and face mask detection and checking proper wearing was also working as expected. However, the system may not properly detect proper wearing of face masks of 3 people if tested with 100 people. It was also found that the model may fail to detect proper wearing of face masks on human faces that are not clearly visible to the camera. This is depicted in Figure 12 image A, B, F, and G. In this case human faces are not detected, hence no face mask checking was done. The faces are covered with a hat plus a face mask. Additionally, faces are clearly check for proper wearing of face masks when they are directly looking at the camera,and this is shown in Figure 12 images C, and G when one's face was not detected because it's not facing the camera directly. However, such cases does not occur frequently and can be controlled.

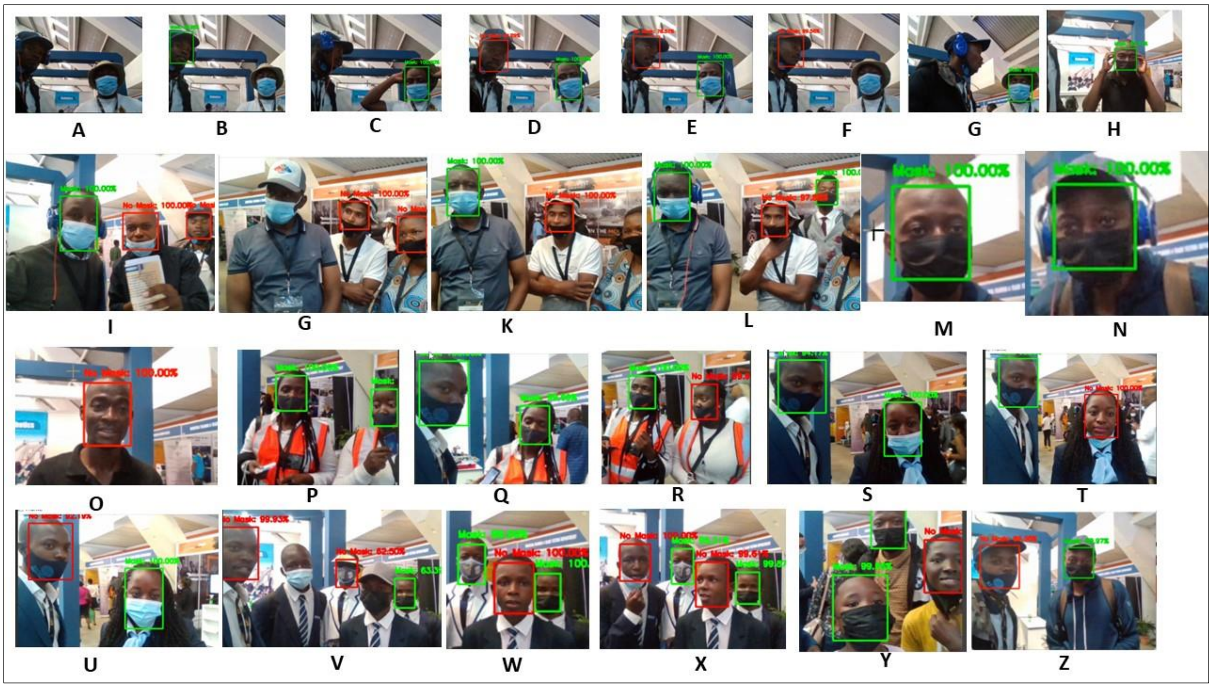

Figure 12: Sample output for face mask checking

Figure 13 below shows training loss and accuracy of the face mask detection model. It's clearly shown that training loss was decreasing as training accuracy was increasing. At the end of the training, training loss was at its minimum, and training accuracy was at its maximum.

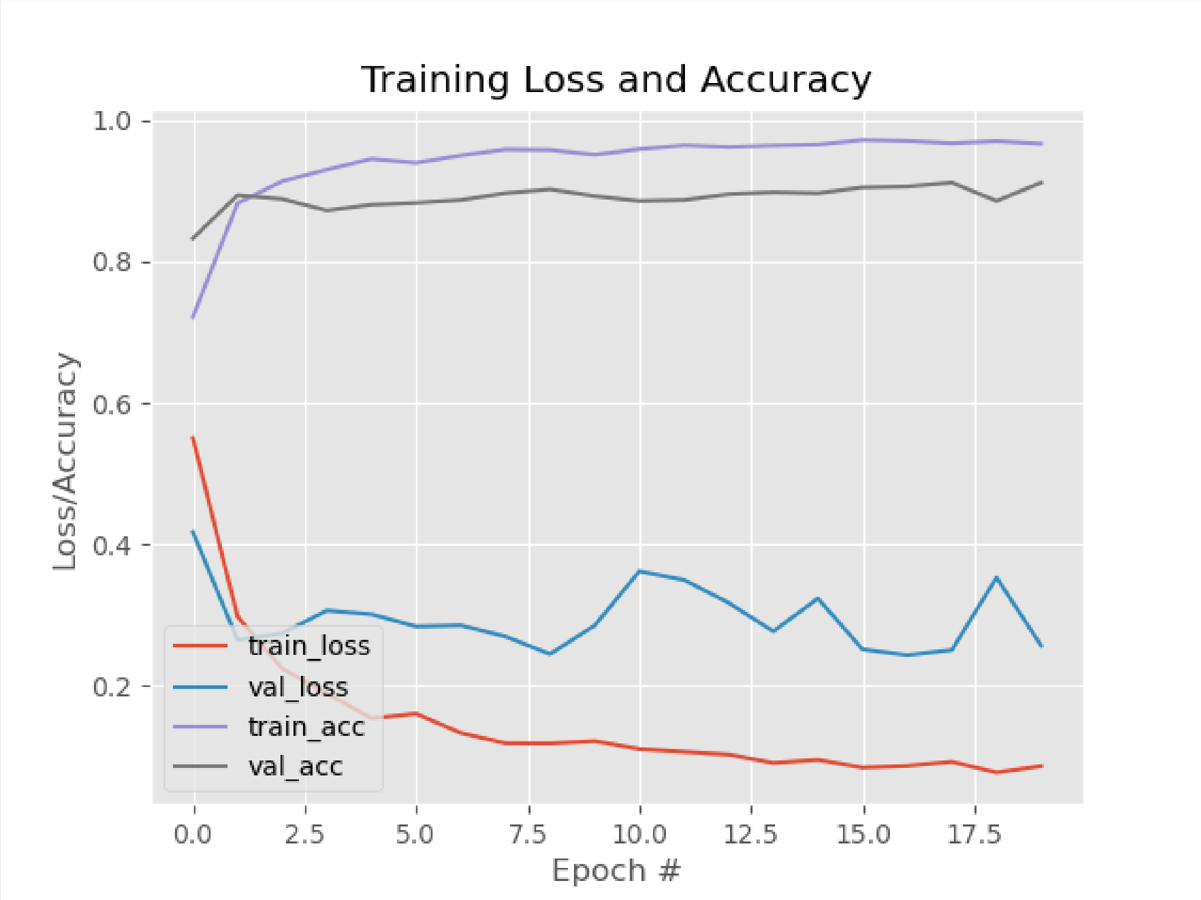

Figure 13: Plot of the training loss and accuracy

Conclusion

In conclusion, the proposed system proved to be fully functional as all objectives were achieved. The system prototype can detect hand objects, check body temperature, sanitize human hands, check proper wearing of face masks at an accuracy rate of 97%. We can conclude that, building an object detection model using MobileNetV2 algorithm can produce high accuracy models that can be implemented in solutions for real-world problems. However, there is need for further research on building good models that detect human faces even if the faces are not directly looking at the camera. Faces hidden in hats and face masks also needs to be detected and checked for proper wearing of face masks.

References

Das, A., Ansari, M. W., & Basak, R. (2021). *Covid-19 Face Mask Detection Using*.

Feng, S., Shen, C., Xia, N., Song, W., Fan, M., & Cowling, B. J. (2020). Rational use of face masks in the COVID-19 pandemic. *The Lancet Respiratory Medicine*, *8*(5), 434–436. https://doi.org/10.1016/S2213-2600(20)30134-X

Girshick, R. (2015). Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, *2015 Inter*, 1440–1448. https://doi.org/10.1109/ICCV.2015.169

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

Hussain, S., Yu, Y., Ayoub, M., Khan, A., Rehman, R., Wahid, J. A., & Hou, W. (2021). IoT and deep learning based approach for rapid screening and face mask detection for infection spread control of COVID-19. *Applied Sciences (Switzerland)*, *11*(8). https://doi.org/10.3390/app11083495

Liu, T., & Stathaki, T. (2018). Faster R-CNn for robust pedestrian detection using semantic segmentation network. *Frontiers in Neurorobotics*, *12*(October), 1–10. https://doi.org/10.3389/fnbot.2018.00064

Ma, J., & Chen, B. (2020). *Dual Refinement Feature Pyramid Networks for Object Detection*. http://arxiv.org/abs/2012.01733

Melnick, H., Darling-Hammond, L., Leung, M., Yun, C., Schachner, A., Plasencia, S., & Ondrasek, N. (2020). Reopening schools in the context of COVID-19: Health and safety guidelines from other countries. *Learning Policy Institute*, *May*, 1–13. https://eric.ed.gov/?id=ED606555

Nowrin, A., Afroz, S., Rahman, M. S., Mahmud, I., & Cho, Y. Z. (2021). Comprehensive Review on Facemask Detection Techniques in the Context of Covid-19. *IEEE Access*, *9*, 106839–106864. https://doi.org/10.1109/ACCESS.2021.3100070

Nyanga, T., & Zirima, H. (2020). Reactions of Small To Medium Enterprises in Masvingo, Zimbabwe To Covid 19: Implications on Productivity. *Business Excellence and Management*, *S.I.*(1), 22–32. https://doi.org/10.24818/beman/2020.s.i.1-02

Oumina, A., El Makhfi, N., & Hamdi, M. (2020). Control the COVID-19 Pandemic: Face Mask Detection Using Transfer Learning. *2020 IEEE 2nd International Conference on Electronics, Control, Optimization and Computer Science, ICECOCS 2020*. https://doi.org/10.1109/ICECOCS50124.2020.9314511

Singh, P., Potlia, I., Malhotra, S., Dubey, H., & Chauhan, H. (2020). Hand Sanitizer an Alternative to Hand Washing—A Review of Literature. *Journal of Advanced Oral Research*, *11*(2), 137–142. https://doi.org/10.1177/2320206820939403

Singh, S., Ahuja, U., Kumar, M., Kumar, K., & Sachdeva, M. (2021). Face mask detection using YOLOv3 and faster R-CNN models: COVID-19 environment. *Multimedia Tools and Applications*, *80*(13), 19753–19768. https://doi.org/10.1007/s11042-021-10711-8

Tang, K. H. D. (2022). Movement control as an effective measure against Covid-19 spread in Malaysia: an overview. *Journal of Public Health (Germany)*, *30*(3), 583–586. https://doi.org/10.1007/s10389-020-01316-w

Yadav, S. (2020). Deep Learning based Safe Social Distancing and Face Mask Detection in Public Areas for COVID-19 Safety Guidelines Adherence. *International Journal for Research in Applied Science and Engineering Technology*, *8*(7), 1368–1375. https://doi.org/10.22214/ijraset.2020.30560