wjert, 2022, Vol. 8, Issue 4, 303-310. *Review Article* ISSN 2454-695X

World Journal of Engineering Research and Technology
WJERT
www.wjert.org SJIF Impact Factor: 5.924

FACE MASK DETECTION USING CNN

***[1]Amogh B. and [2]Dr. Ganesh D**

[1]Department of MCA JAIN (Deemed-to-be-university) Bangalore.

[2]Associate Professor School of CS & IT JAIN (Deemed-to-be-university), Bangalore.

Article Received on 12/02/2022 **Article Revised on 02/03/2022** **Article Accepted on 22/03/2022**

***Corresponding Author**
Amogh B.
Department of MCA JAIN (Deemed-to-be-university) Bangalore.

ABSTRACT

The COVID-19 epidemic has had a dramatic impact on our daily lives, disrupting global trade and transportation. Protecting one's face with a mask has become the new normal. Many public service providers will need clients to wear masks correctly in the near future in order to use their services. As a result, detecting face masks has become a critical responsibility in aiding worldwide society. This project shows how to achieve this goal with the help of several fundamental Machine Learning tools such as TensorFlow, Keras, OpenCV, and Scikit-Learn. The suggested approach successfully detects the face in the image and then determines whether or not it is covered by a mask. It can also detect a face and a mask in motion as a surveillance task performance. The approach achieves a level of accuracy of up to 99%. We investigate optimum parameter values using the Sequential Convolutional Neural Network model to appropriately detect the existence of masks without over-fitting.

KEYWORDS: Coronavirus, Covid-19, Machine Learning, Face Mask Detection, Convolutional Neural Network, TensorFlow.

INTRODUCTION

Coronavirus illness 2019 (COVID-19) has infected over 20 million individuals worldwide, resulting in over 0.7 million deaths, according to the World Health Organization's official Situation Report – 205. COVID-19 patients have reported a wide range of symptoms, ranging from mild signs to major sickness. One of them is respiratory issues such as shortness of breath or difficulty breathing. COVID-19 infection can cause major consequences in elderly patients with lung disease, as they appear to be at a higher risk. 229E, HKU1, OC43, and

NL63 are some of the most prevalent human coronaviruses that infect people all over the world. Viruses like 2019-nCoV, SARS-CoV, and MERS-CoV infect animals and evolve into human coronaviruses before infecting humans. Anyone (who is in close touch with them) can be exposed to people who have respiratory difficulties. The environment of a tainted person can trigger contact transmission because virus-carrying droplets can land on his nearby surfaces.

Wearing a clinical mask is essential for preventing some respiratory viral infections, such as COVID-19. The public should know whether to wear the mask for source control or COVID-19 aversion. The use of masks has the potential to reduce vulnerability to danger from a noxious individual during the "pre-symptomatic" stage, as well as stigmatise individuals who wear masks to prevent the spread of virus. The World Health Organization emphasises the use of medical masks and respirators for health care assistants. Face mask detection has thus become a critical task in today's global society.

Face mask detection is identifying the location of a person's face and then assessing whether or not they are wearing a mask. The problem is related to general object detection, which is used to identify different types of things. Face identification is the process of recognising a specific group of entities, such as people. It has a wide range of applications, including autonomous driving, education, and spying. This project uses basic Machine Learning (ML) tools such as TensorFlow, Keras, OpenCV, and Scikit-Learn to serve the above objective in a simplified manner.

Problem Statement

The COVID-19 epidemic has had a dramatic impact on our daily lives, disrupting global trade and transportation. Protecting one's face with a mask has become the new normal. This real-time system determines whether or not the person is wearing a mask. We'll use Keras and Open CV to train the face mask detector model.

This model recognises and distinguishes those who are not wearing a mask from those who are, and then sends a report to the municipalities and relevant authorities, who will fine the individual and educate them about the necessity of covid-19 safety measures.

Proposed solution

Face mask detection is identifying the location of a person's face and then assessing whether or not they are wearing a mask. The problem is related to general object detection, which is used to identify different types of things. Face identification is the process of recognising a specific group of entities, such as people. It has a wide range of uses, including autonomous driving, education, and spying.[5] The core Machine Learning (ML) packages such as TensorFlow, Keras, OpenCV, and Scikit-Learn are used in this research to propose a simpler solution to serve the aforesaid objective. A cascade classifier and a pre-trained CNN with two 2D convolution layers coupled to layers of dense neurons are used in the suggested technique. The face mask detection algorithm is as follows.

Data Processing: Data preprocessing is converting data from one format to another that is more user-friendly, desirable, and understandable. It can take any shape, including tables, photographs, movies, graphs, and so on. Numpy and OpenCV are used in the suggested method to deal with image and video data.

a) Data Visualization: The technique of translating abstract data into meaningful representations via knowledge exchange and insight finding through encodings is known as data visualisation. It is beneficial to investigate a specific trend in the dataset. The total number of photos in the dataset is represented in both 'with mask' and 'without mask' categories. The number of labels is then calculated by separating those categories.

b) Conversion of RGB image to Gray image: Modern descriptor-based image recognition systems frequently work with grayscale images, without expanding on the color-to-grayscale conversion mechanism. This is because when utilising robust descriptors, the color-to-grayscale approach has little impact. Incorporating non-essential data could increase the amount of training data needed to attain effective results. Grayscale is used for extracting descriptors rather than working on colour photos in real time because it rationalises the approach and reduces computational requirements. A fixed-size input image is required for deep CNNs. As a result, all of the photos in the collection must have the same size. The grayscale image is downsized to 100 × 100 pixels using resize.

c) Image Reshaping: A three-dimensional tensor is used as the input during image relegation, with each channel having a prominent unique pixel. All of the photos must be the same size and belong to the 3D feature tensor. However, neither pictures nor their related feature tensors are usually coextensive. Most CNNs can only accept photos that have been fine-tuned. This causes a slew of issues during data gathering and model implementation. This constraint can be overcome by rearranging the input images before supplementing them into the network. The photos have been adjusted to bring the pixel range between 0 and 1 closer together. The images are then converted to four-dimensional arrays, with one representing the Grayscale image. As the neural network's last layer has two outputs, one with a mask and one without, it has categorical outputs.

Training of Model

Building the model using CNN architecture: CNN has risen to the top of a variety of computer vision challenges. Sequential CNN is used in the current technique.

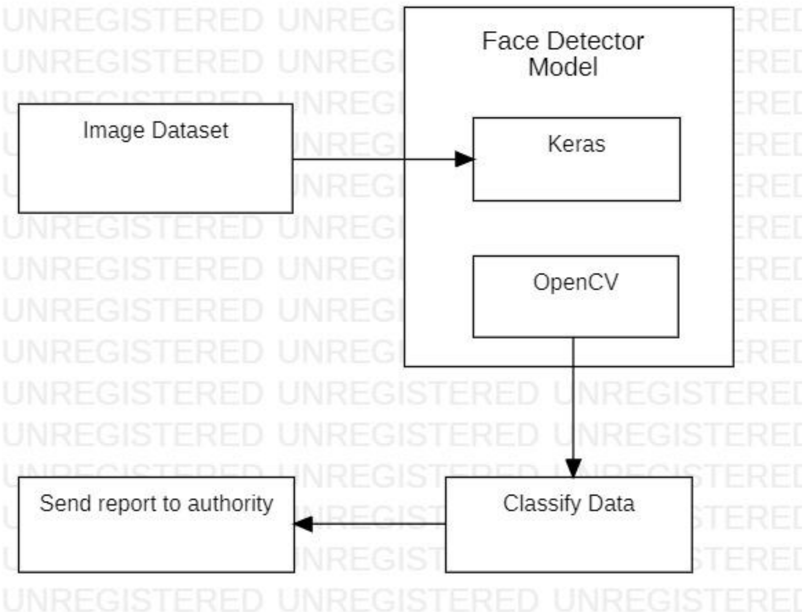
The Rectified Linear Unit (ReLU) and MaxPooling levels come after the First Convolution layer. The Convolution layer has 200 filters to learn from. The height and width of the 2D convolution window are specified by the kernel size, which is set to 3 x 3. Because the model must be aware of the shape of the expected input, the initial layer of the model must be provided with input shape information. Following layers can conduct shape reckoning instinctively. In this scenario, input shape is set to data.shape, which yields the data array's dimensions starting at index 1. When the spatial dimensions are allowed to truncate and the padding is set to "valid," the default padding is "valid." The Conv2D class's activation parameter is set to "relu." It depicts a nearly linear function with all of the advantages of linear models and the ability to be easily optimised using gradient-descent methods. It outperforms other activation functions in terms of performance and generalisation in deep learning. Max Pooling is used to lower the output volume's spatial dimensions. The final output has the form (number of rows or columns) of: shape of output = (input shape – pool size + 1) / strides, where strides is set to the default value (1, 1).

The second Convolution layer comprises 100 filters with a 3 x 3 Kernel size. The ReLu and MaxPooling layers come after it. The long vector of input is routed via a Flatten layer, which converts the matrix of features into a vector that can be fed into a fully connected neural network classifier, to insert the data into CNN. A Dropout layer with a 50% chance of setting inputs to zero is added to the model to reduce overfitting. Then a 64-neuron dense layer with a ReLu activation function is introduced. The Softmax activation function is used in the final layer (Dense), which has two outputs for two categories.

The compile method must be used to set up the learning process first. The "adam" optimizer is utilised in this case. As a loss function, categorical crossentropy, commonly known as multiclass log loss, is used (the 14 objective that the model tries to minimize). The measurements are set to "accuracy" because the problem is a classification challenge.

Splitting the data and training the CNN model: The model must be trained using a certain dataset and then evaluated against a new dataset after the blueprint for data analysis has been established. When constructing a forecast, a good model and an optimised train test split can help you get accurate results. The test size is set to 0.1, which means that 90% of the data in the dataset is trained and 10% is used for testing. ModelCheckpoint is used to track validation loss. The images from both the training and test sets are then fitted to the Sequential model. Validation data is used to replace 20% of the training data. The model is trained for 20 epochs (iterations) to maintain a balance of accuracy and overfitting risk. The graphic below shows a visual representation of the proposed model.

System Architecture



Literature study

Following the rapid outbreak of a novel Coronavirus illness (COVID-19) case in Wuhan, China in December 2019, the World Health Organization (WHO) determined that this is a hazardous virus that may be transmitted from humans to people via droplets and airborne transmission.

The advent of the COVID-19 pandemic is currently regarded as a severe worldwide health problem that poses a serious risk to people's health and safety. This new illness examines not just an organization's resilience, but also its ability to adapt to these trigger events by implementing a new strategy.

H. Adusumalli et al.,[6] The Covid-19 issue creates an unpredictable climate for their workers, posing immediate concerns to company performance and long-term viability. As a result, firms' executives are revamping their human resource strategies in order to effectively address these crucial situations. Resilient businesses can better assist their employees in adjusting to the changing work climate. In order to attain great performance in a dynamic environment, decision makers must adopt the organisational resilience approach.

S. Susanto et al.,[9] The authors describe how the danger of transmission is greatest in public locations in their publication. Wearing a face mask in open areas is one of the best methods to avoid becoming infected, according to the World Health Organization (WHO). While it comes to preventive, wearing a face mask is a must when going outside or meeting new people. Some reckless people, on the other hand, refuse to wear a face mask for a variety of reasons. Furthermore, in this scenario, establishing a face mask detector is critical.

H. Adusumalli et al.,[5] provides a method for detecting face masks on people using TensorFlow and OpenCV. A bounding box created across the person's face defines whether the person is wearing a mask or not in this system as a measure to detect individuals not wearing masks. If a person's face is saved in the database, it detects the name of the person who is not wearing a face mask and sends an email to that person alerting them to the fact that they are not wearing a mask and advising them to take precautions.

Recommendations

This project's scope can include everything from installing monitoring and surveillance devices in public locations to monitoring people who do not follow Covid-19 guidelines.It can be used in corporations, hospitals, public transit, and shopping malls to detect and penalize those who do not wear facemasks.

It can also be extended to recognize persons and deliver notices, instructions, and information about covid-19 safety protocols, assisting the government in adopting covid-19 safety protocols and educating the public about the necessity of safety precautions.

CONCLUSIONS

In this project, the system exhibited the model's learning and performance task. Using basic machine learning tools and simplified methodologies, the system has achieved a fair level of accuracy. It can be used for a variety of things. Given the circumstances surrounding Covid-19, wearing a mask may become required in the not-too-distant future. In order to receive government services, many agencies will require people to wear masks correctly. The paradigm that has been implemented will have a substantial impact on the public health care system. It could be improved in the future to identify whether a person is correctly wearing the mask. The model should be improved further to determine whether or not the mask is virus-prone, i.e., whether it is surgical, N95, or not.

REFERENCES

1. “Coronavirus disease 2019 (covid-19): situation report 205", 2020.
2. "Coronavirus Disease 2019 (COVID-19) – Symptoms", Centers for Disease Control and Prevention, 2020. [online] Available: https://www.cdc.gov/coronavirus/2019-ncov/symptomstesting/symptoms.html.
3. P. S. Reddy, M. Nandini, E. Mamatha, K. V. Reddy and A. Vishant, "Face Mask Detection using Machine Learning Techniques," 2021. 5th International Conference on Trends in Electronics and Informatics.
4. (ICOEI), 2021; 1468-1472. doi: 10.1109/ICOEI51242.2021.9452826.
5. "Advice on the use of masks in the context of COVID-19: interim guidance", 2020.
6. H. Adusumalli, D. Kalyani, R. K. Sri, M. Pratapteja and P. V. R. D. P. Rao, "Face Mask Detection Using OpenCV," 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), 2021; 1304-1309. doi: 10.1109/ICICV50876.2021.9388375.
7. F. Hohman, M. Kahng, R. Pienta and D. H. Chau, "Visual Analytics in Deep Learning: An Interrogative Survey for the Next Frontiers", IEEE Transactions on Visualization and Computer Graphics, 2019; 25(8): 2674-2693.
8. C. Kanan and G. Cottrell, "Color-to-Grayscale: Does the Method Matter in Image Recognition?", PLoS ONE, 2012; 7(1): e29740.
9. S. Susanto, F. A. Putra, R. Analia and I. K. L. N. Suciningtyas, "The Face Mask Detection For Preventing the Spread of COVID-19 at Politeknik Negeri Batam," 3rd International Conference on Applied Engineering (ICAE), 2020.

10. M. Hashemi, "Enlarging smaller images before inputting into convolutional neural network: zero-padding vs. interpolation", Journal of Big Data, 2019; 6(1).
11. S. Ghosh, N. Das and M. Nasipuri, "Reshaping inputs for convolutional neural network: Some common and uncommon methods", Pattern Recognition, 2019; 93: 79-94.
12. R. Yamashita, M. Nishio, R. Do and K. Togashi, "Convolutional neural networks: an overview and application in radiology", Insights into Imaging, 2018; 9(4): 611-629.
13. "Guide to the Sequential model - Keras Documentation", [online] Available: Faroit.com, 2020.
14. C. Nwankpa, W. Ijomah, A. Gachagan and S. Marshall, "Activation Functions: Comparison Of Trends In Practice And Research For Deep Learning", 2020. [online] Available: https://arxiv.org/abs/1811.03378.2020.
15. "Keras documentation: MaxPooling2D layer", 2020, [online] Available: Keras.io.
16. "Prajnasb/observations", 2020. [online] Available: https://github.com/prajnasb /observations/tree/master/experiements/data.
17. "Face Mask Detection", 2020. [online] Available: Kaggle.com.
18. "TensorFlow White Papers", TensorFlow, 2020. [online] Available: https://www.tensorflow.org/about/bib.
19. "Keras documentation: About Keras", 2020. [online] Available: Keras.io.
20. "OpenCV", 2020. [online] Available: Opencv.org.
21. D. Meena and R. Sharan, "An approach to face detection and recognition", International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 2016; 1-6.
22. S. Ge, J. Li, Q. Ye and Z. Luo, "Detecting Masked Faces in the Wild with LLE-CNNs", 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2017; 426-434.