Final Project

KIRUTHIGA R
AU211521243090
PANIMALAR INSTITUTE OF TECHNOLOGY

PROJECT TITLE

"MaskNet: Real-Time Mask Detection using Convolutional Neural Networks with TensorFlow"

AGENDA

This project discusses the need to develop an accurate and efficient system for detecting the presence or absence of face masks in public spaces using convolutional neural networks (CNNs) and TensorFlow, aiming to enhance safety measures and reduce the transmission of infectious diseases.

PROBLEM STATEMENT

"Developing a robust face mask detection system using CNN architecture and TensorFlow to enhance public health safety by enforcing mask-wearing protocols in various settings."

PROJECT OVERVIEW

1.Goal: Utilizing CNN, TensorFlow, and Keras, create a face mask detection model.

2.Data: For training, used Kaggle datasets with and without mask pictures.

3.Method:Used TensorFlow and Keras to implement CNN architecture for effective model training.

4.Impact: Promote mask wearer compliance and stop the transmission of illness by enabling real-time detection.

WHO ARE THE END USERS?

The end users include public health authorities, business owners, and the general public who benefit from enforcing mask-wearing protocols for safety, as well as developers and researchers interested in advancing computer vision and public health technology.

YOUR SOLUTION AND ITS VALUE PROPOSITION

Convolutional neural networks (CNNs) constructed using TensorFlow and Keras are the key to solving the problem; these networks will be used to create a face mask identification system. Its value offer is to accurately and instantly identify people who are wearing masks, support public health safety initiatives, and assist in the enforcement of mask laws.

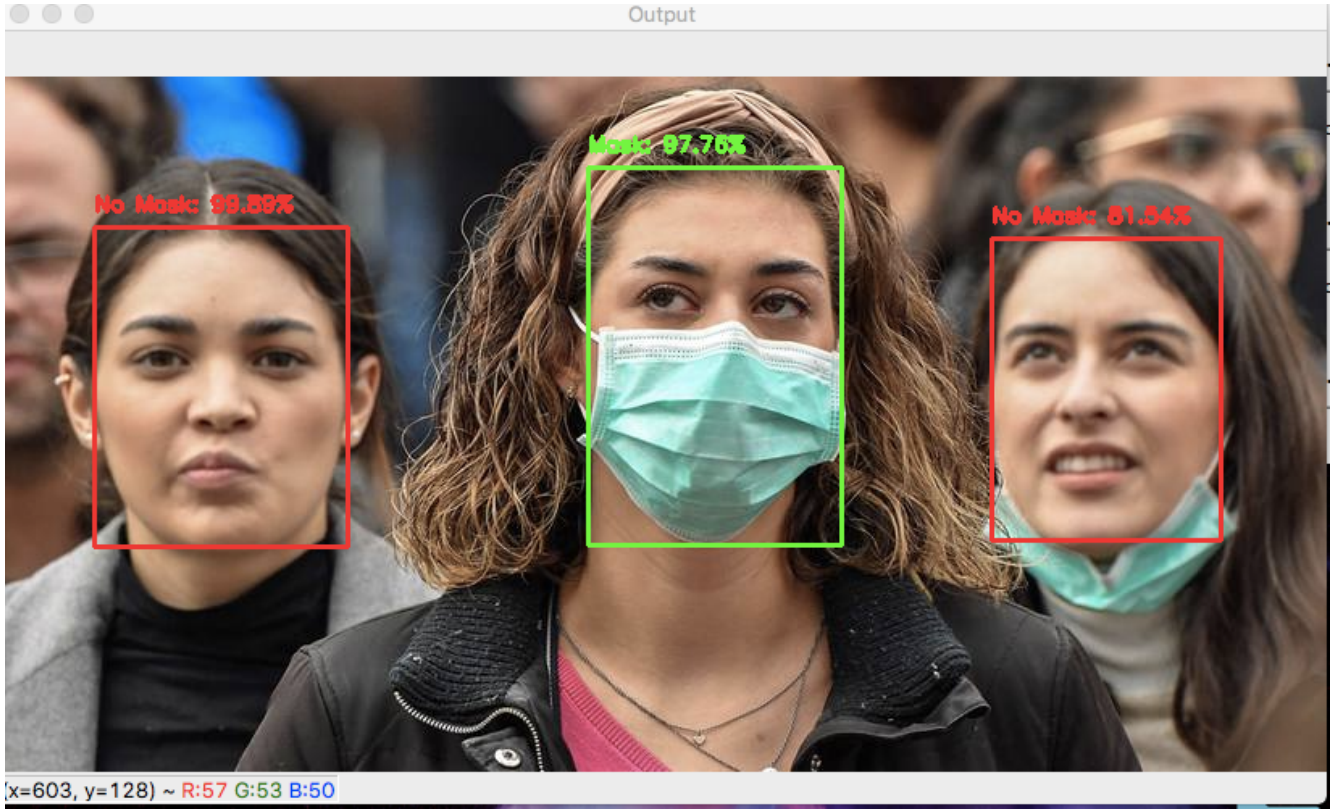
THE WOW IN YOUR SOLUTION

CNNs are highly effective at learning complex patterns and features from data.Tensorflow and Keras provide a flexible and customizable environment.They can achieve real-time or near-real-time performance.By integrating OpenCV for image preprocessing,the solution offers an end-to-end pipeline for face mask detection

MODELLING

For data preprocessing, OpenCV libraries such as cv2 and os were utilized. Two empty lists, namely "data" and "target," were created. In "data," grayscale images were appended, while corresponding labels (e.g., "with mask" and "without mask") were appended to "target." Using numpy, the arrays were loaded into a CNN model constructed with Keras models and layers, which included operations like flattening and pooling. The data was then split into training and testing sets, with callbacks functions employed to save the model at the end of each epoch. Visualization techniques, such as matplotlib, provided an overview of the data. Finally, in the detection model, CascadeClassifier and videocapture to detect faces, with masks highlighted in green boxes and without masks in red.

RESULTS



Demo Link