Title: Face Mask Detection using Convolutional Neural Networks

Author: Ramvarinder singh

Date: September 9, 2025

Introduction

In response to global health situations, the use of face masks has become a critical measure for public safety. Automating the process of monitoring mask compliance can significantly aid in ensuring safety in public and private spaces. This project focuses on developing a deep learning model to automatically detect whether a person in an image is wearing a face mask. The core technology used is a Convolutional Neural Network (CNN), a specialized type of neural network highly effective for image recognition tasks.

Abstract

This project details the development of a binary image classification model to distinguish between individuals with and without face masks. A Convolutional Neural Network was constructed using the TensorFlow and Keras libraries. The model was trained on a public dataset from Kaggle, which contained thousands of labeled images. Key steps included data preprocessing with

ImageDataGenerator, building a sequential CNN architecture, training the model for 10 epochs, and evaluating its performance. The resulting model achieved a validation accuracy of over 93%, demonstrating its effectiveness in accurately identifying the presence of a face mask. The final trained model was saved for potential integration into a real-time detection system.

Tools Used

- **Programming Language:** Python
- **Deep Learning Framework:** TensorFlow & Keras
- **Data Handling:** NumPy
- **Data Visualization:** Matplotlib
- **Development Environment:** Jupyter Notebook
- **Dataset Source:** Kaggle

Steps Involved in Building the Project

The project was executed following a structured machine learning workflow:

1. **Data Collection and Preparation:** A labeled dataset was acquired from Kaggle, containing thousands of images. The data was organized into a directory structure with two subfolders:

with_mask and without_mask, which is a requirement for supervised learning with image generators.

2. **Data Preprocessing:** The Keras ImageDataGenerator was used to automate the loading and preprocessing of images. This step included resizing all images to a uniform 128x128 pixels, normalizing pixel values to a range of [0, 1], and automatically splitting the dataset into an 80% training set and a 20% validation set.
3. **Model Architecture:** A Sequential CNN was designed. The architecture consisted of:
   - Two Conv2D layers (with 32 and 64 filters respectively) to extract features from the images.
   - Two MaxPooling2D layers to reduce the spatial dimensions and highlight the most important features.
   - A Flatten layer to convert the 2D feature maps into a 1D vector.
   - A Dense layer with 128 neurons and a Dropout layer to prevent overfitting.
   - A final Dense output layer with a sigmoid activation function to produce a single probability score indicating the presence of a mask.
4. **Model Compilation:** Before training, the model was compiled with the adam optimizer, binary_crossentropy loss function (suitable for two-class classification), and accuracy as the performance metric.
5. **Training:** The model was trained for 10 epochs using the .fit() method. During this phase, the model learned to differentiate between the two classes by iterating over the training data, and its performance was checked against the validation data at the end of each epoch.
6. **Saving the Model:** After the training was complete, the final, trained model was saved to a single file (my_custom_mask_detector.h5), making it reusable for future applications like real-time video detection.

Conclusion

This project successfully demonstrates the creation of an effective face mask detection model using Convolutional Neural Networks. The model achieved a high validation accuracy, proving that deep learning is a viable solution for this real-world problem. The final saved model is a portable and efficient asset that can be deployed in various applications to help enforce public health guidelines.