**Face Mask Detection System Using AI and ML**

**Submitted for**

**Statistical Machine Learning CSET211**

Submitted by:

**(E23CSEU2197) SHOBHIT JAIN**

**(E23CSEU2210) SHUBHANG SINGH**

**(E23CSEU2214) YUVRAJ PASSAI**

Submitted to

**SUSMITA DAS**

**July-Dec 2024**

**SCHOOL OF COMPUTER SCIENCE AND ENGINEERING**

A close-up of a logo

Description automatically generated

**Abstract**

In response to the global COVID-19 pandemic, face mask detection has become an essential safety measure in public spaces to mitigate the spread of the virus. This project focuses on developing an AI-based face mask detection system using machine learning techniques, implemented in Python using the PyCharm IDE. The system can accurately detect whether a person is wearing a mask or not in real-time through webcam integration. Leveraging convolutional neural networks (CNN) for image classification, this project aims to provide a solution for ensuring compliance with mask-wearing protocols, potentially aiding in public health safety measures.

**1. Introduction**

The COVID-19 pandemic has fundamentally changed the way we interact with public spaces, making face masks a critical component of public health. Enforcing the usage of masks, especially in crowded areas, is essential to prevent virus transmission. Manual monitoring, however, is time-consuming and inefficient. This project addresses the challenge by introducing an automated, AI-powered face mask detection system that can monitor compliance in real-time.

The face mask detection system utilizes deep learning, specifically Convolutional Neural Networks (CNN), to classify individuals as either "With Mask" or "Without Mask" using live video feed from a webcam. This technology has significant applications in various sectors such as healthcare, retail, transportation, and public administration.

**2. Related Work**

Several studies and projects have explored the use of computer vision for face detection and classification. Most notably, advancements in deep learning, particularly using CNNs, have made it possible to achieve high accuracy in image classification tasks. Existing models such as MobileNet, VGG16, and ResNet have been applied to similar problems, but this project focuses on a custom-trained CNN model optimized for face mask detection.

Unlike previous solutions, this project integrates real-time video stream analysis, making it more suitable for practical, real-world applications like surveillance in public places.

**3. Methodology**

**3.1. Data Collection and Preprocessing**

* **Dataset:** The dataset used for training the model consists of images categorized into two classes: "With Mask" and "Without Mask".
* **Data Augmentation:** To enhance model performance and prevent overfitting, data augmentation techniques such as rotation, flipping, and scaling were applied.
* **Image Preprocessing:** Images were resized to 128x128 pixels, converted to RGB, and normalized for optimal model training.

**3.2. Model Architecture**

* **Convolutional Neural Network (CNN):** The model is a custom-built CNN designed to detect face masks in real-time. The architecture includes:
  + **Input Layer:** Accepts images of size 128x128x3.
  + **Convolutional Layers:** Multiple layers with ReLU activation to extract features.
  + **Pooling Layers:** MaxPooling layers to reduce spatial dimensions.
  + **Dense Layers:** Fully connected layers leading to a binary classification output.
  + **Output Layer:** Sigmoid activation function for binary classification ("With Mask" or "Without Mask").

**3.3. Model Training and Evaluation**

* **Training:** The model was trained using the Adam optimizer with a learning rate of 0.001, and binary cross-entropy as the loss function.
* **Evaluation Metrics:** Accuracy, precision, recall, and F1-score were used to evaluate the model's performance on the test set.
* **Real-Time Detection:** OpenCV was utilized for capturing live video feed and detecting faces using the Haar Cascade Classifier. The trained model was then applied to classify detected faces.

**3.4. Live Video Stream Integration**

* **Webcam Integration:** The system uses OpenCV to access the webcam for real-time mask detection.
* **User Interface:** The detected faces are highlighted with bounding boxes, and the classification result ("With Mask" or "Without Mask") is displayed on the video feed.

**4. Hardware/Software Required**

**Hardware**

* **PC/Laptop**: For development and testing.
* **Webcam**: For real-time video stream analysis (USB webcam for desktops).

**Software**

* **Operating System**: Windows 10
* **Programming Language**: Python 3.x
* **Integrated Development Environment (IDE)**: PyCharm
* **Libraries**:
  + **TensorFlow/Keras**: For building and training the CNN model.
  + **OpenCV**: For image and video processing.
  + **NumPy and Pandas**: For data manipulation.
  + **Matplotlib**: For visualizing results.

**5. Experimental Results**

* **Training Accuracy**: Achieved 98% accuracy on the training dataset.
* **Validation Accuracy**: Maintained an accuracy of 96% on the validation dataset.
* **Real-Time Detection Accuracy**: Successfully classified mask usage with over 95% accuracy in real-time video feeds.
* The system was tested under various lighting conditions and angles, demonstrating robust performance in detecting masks.

**Sample Output**

* The system draws a bounding box around the detected face and displays a label indicating "With Mask" or "Without Mask".
* The prediction confidence score is also displayed for each detection.

**6. Conclusion**

The Face Mask Detection System developed in this project provides a reliable and efficient solution for monitoring mask compliance in real-time. The system's high accuracy and responsiveness make it suitable for deployment in public spaces to ensure health safety. By leveraging deep learning techniques and real-time video analysis, this project offers a scalable approach to automated surveillance.

**7. Future Scope**

The current system performs exceptionally well under controlled conditions. However, several enhancements can be made for broader application:

* **Multi-Face Detection**: Expanding the model to detect multiple faces simultaneously.
* **Mask Type Classification**: Identifying different types of masks (e.g., surgical, cloth, N95).
* **Integration with IoT Devices**: Connecting with edge devices like Raspberry Pi for smart surveillance systems.
* **Deployment in Public Areas**: Using the system in airports, shopping malls, and hospitals for automated compliance checks.
* **Mobile App Development**: Extending the application to mobile platforms for broader accessibility.

**8. GitHub Link of Your Complete Project**

You can access the complete code and related files of this project on GitHub: <https://github.com/AiMG0D0P/FaceMaskFinal>