Face Mask Detection

PROBLEM STATEMENT, VARIABLE DESCRIPTION & DELIVERABLES

Material Quality Prediction – Objective & Deliverables

Objective

AirGuard AI, a company dedicated to improving urban air quality and public health, aims to develop an AI-powered face mask detection system to address the growing concerns of **air pollution exposure**.

Problem Statement

With rising pollution levels in major cities, many people are encouraged to wear protective face masks to reduce health risks associated with fine particulate matter (PM2.5), dust, and toxic air pollutants. However, **ensuring widespread adoption and compliance remains a challenge**. Governments, healthcare agencies, and environmental organizations lack an efficient system to monitor and analyze mask usage trends in highly polluted areas.

To address this issue, you'll develop an **AI-powered face mask detection system** using deep learning and computer vision. By training a model on a dataset of **7,553 images** containing both masked and unmasked faces, the model will be capable of detecting face masks in real-time from surveillance footage or camera feeds. This solution will help **analyze mask usage trends**, **assist policymakers in enforcing public safety measures**, and **integrate with smart city initiatives** to enhance pollution control efforts. The project will not only provide a technological solution to AirGuard AI's challenge but also contribute to broader environmental and public health initiatives.

Recommended Project Steps & Guidelines

To successfully develop and deploy the AI-powered face mask detection system for AirGuard AI, you can follow below steps:

- **Dataset Preparation & Preprocessing**
  - Clean and organize the dataset of 7,553 images into two categories: with mask and without mask.
  - Perform data augmentation (e.g., rotation, brightness adjustments) to improve model generalization.

- Convert images to a uniform size and apply normalization to enhance model performance.

Model Selection & Training

- Choose a deep learning model architecture, such as CNN (Convolutional Neural Network), or fine-tune a pre-trained model like MobileNetV2 or ResNet50.
- Split the dataset into training, validation, and testing sets to ensure model accuracy and prevent overfitting.
- Train the model using TensorFlow/Keras or PyTorch, optimizing it for high accuracy and fast inference.

Model Evaluation & Optimization

- Evaluate model performance using metrics like accuracy, precision, recall, and F1-score.
- Optimize the model by tuning hyperparameters, adding dropout layers, and applying transfer learning if necessary.
- Test the model on unseen images and real-world datasets to assess robustness.

Deployment & Integration

- Convert the trained model into a lightweight format (e.g., TensorFlow Lite, ONNX) for real-time detection.
- Develop a web-based application that can process images or camera feeds and detect face masks in images.

Reporting & Insights Generation

- Develop a dashboard to visualize mask compliance trends and generate reports for policymakers and health organizations.
- Provide data-driven insights to help authorities enforce mask mandates and improve air pollution control strategies.
- Future Enhancements & Scalability

