

Masked Enforcerer

Jose Moreno

Tier 1 – Core Project

ITAI 1378: Computer Vision and AI

The Problem

Problem: Difficulty monitoring public health compliance in crowded areas.

Who Cares: Private businesses, schools, hospitals, public spaces.

Why Important: Manually monitoring compliance is bound to be subjected to human error, creating the necessity of automated systems; automatization saves time and improves accuracy.

Your Solution (Overview)

Automated system that detects if individuals wear masks properly if Any via using computer vision.

Input: Camera feed or image → Model detects face and mask status.

Output: Labeled image: 'Mask', 'No Mask', or 'Incorrect Mask'.

Technical Approach

Technique: Object detection + classification.

Model: YOLOv8 (Ultralytics, pre-trained, fine-tuned).

Framework: PyTorch (Google Colab environment).

Justification: Fast, accurate, and lightweight for real-time detection.

Optimization Plan:

- Use pre-trained weights to minimize training time.
- Train on a reduced dataset subset first, then fine-tune.
- Apply image resizing (e.g., 416×416) to reduce compute load.
- Use mixed precision (FP16) to speed up training.

Data Plan

Source: Public Kaggle 'Face Mask Detection Dataset' (~7,000 images).

Labels: Three classes – Mask, No Mask, Incorrect Mask.

Preparation: Dataset cleaning and normalization; split into train/val/test.

Augmentation: Apply rotation, flip, and brightness adjustments.

System Diagram

[Input Image] → [YOLOv8 Detection] → [Mask Classification] →
[Labeled Output]

Success Metrics

Primary Metric: Detection Accuracy $\geq 90\%$.

Secondary Metric: Inference Speed < 1 second per image.

Qualitative: Correctly identifies all mask conditions in test images.

Week-by-Week Plan

- Week 10 (Oct 30): Collect dataset, set up environment → Dataset ready.
- Week 11 (Nov 6): Train/fine-tune YOLOv8 model → Model working.
- Week 12 (Nov 13): Evaluate and improve performance → Good accuracy.
- Week 13 (Nov 20): Create demo video → Demo ready.
- Week 14 (Nov 27): Final testing/documentation → Complete.
- Week 15 (Dec 4): Present project → 🎉 Presentation day.

Challenges & Backup Plans

Challenge: Low dataset quality → Plan B: Use Roboflow dataset.

Challenge: Model underperforms → Plan B: Apply augmentation or larger model.

Challenge: Limited compute → Plan B: Train smaller YOLOv8n model.

Resources Needed

Compute: Google Colab (Free Tier).

Frameworks: PyTorch, Ultralytics YOLOv8.

Storage: Google Drive for dataset and results.

Estimated Cost: \$0.