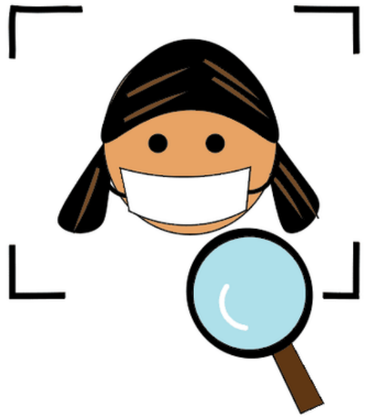
MASKGUARD

Real-Time Face Mask Detection using CNN's

LIST OF CONTENTS

ABSTRACT



“In recent times, the widespread use of face masks has become a critical measure in combating the transmission of contagious diseases. Automated face mask detection systems have gained significance to ensure compliance with mask- wearing protocols. This project proposes a solution for real-time face mask detection using Convolutional Neural Networks (CNNs). The system utilizes a dataset of annotated images containing individuals with and without masks. A pre-trained CNN architecture is fine-tuned and trained on this dataset to accurately classify individuals into two categories: 'with mask' and 'without mask. This project takes not only a single person image but also with multiple persons with in a picture .The proposed model is capable of robustly detecting mask presence in various lighting conditions, angles, and facial orientations. Experimental results demonstrate high accuracy and efficiency in real- time detection, making it suitable for deployment in public spaces, healthcare facilities, and other settings requiring adherence to mask-wearing guidelines.”

introducing

ABOUT MASKGUARD

An effective framework for real-time identification of people wearing or not wearing masks is provided by CNNs. This technology enables the quick and accurate recognition of mask adherence in a variety of circumstances, which has the potential to improve safety, ease enforcement activities, and support public health initiatives.

PROBLEM STATEMENT

01 Mask Non-Compliance

02 Manual Detection Has Its Limitations

03 The Risk of Disease Spread

OBJECTIVE

- Instant Detection
- Accuracy
- Cost-Effective

METHODOLOGY

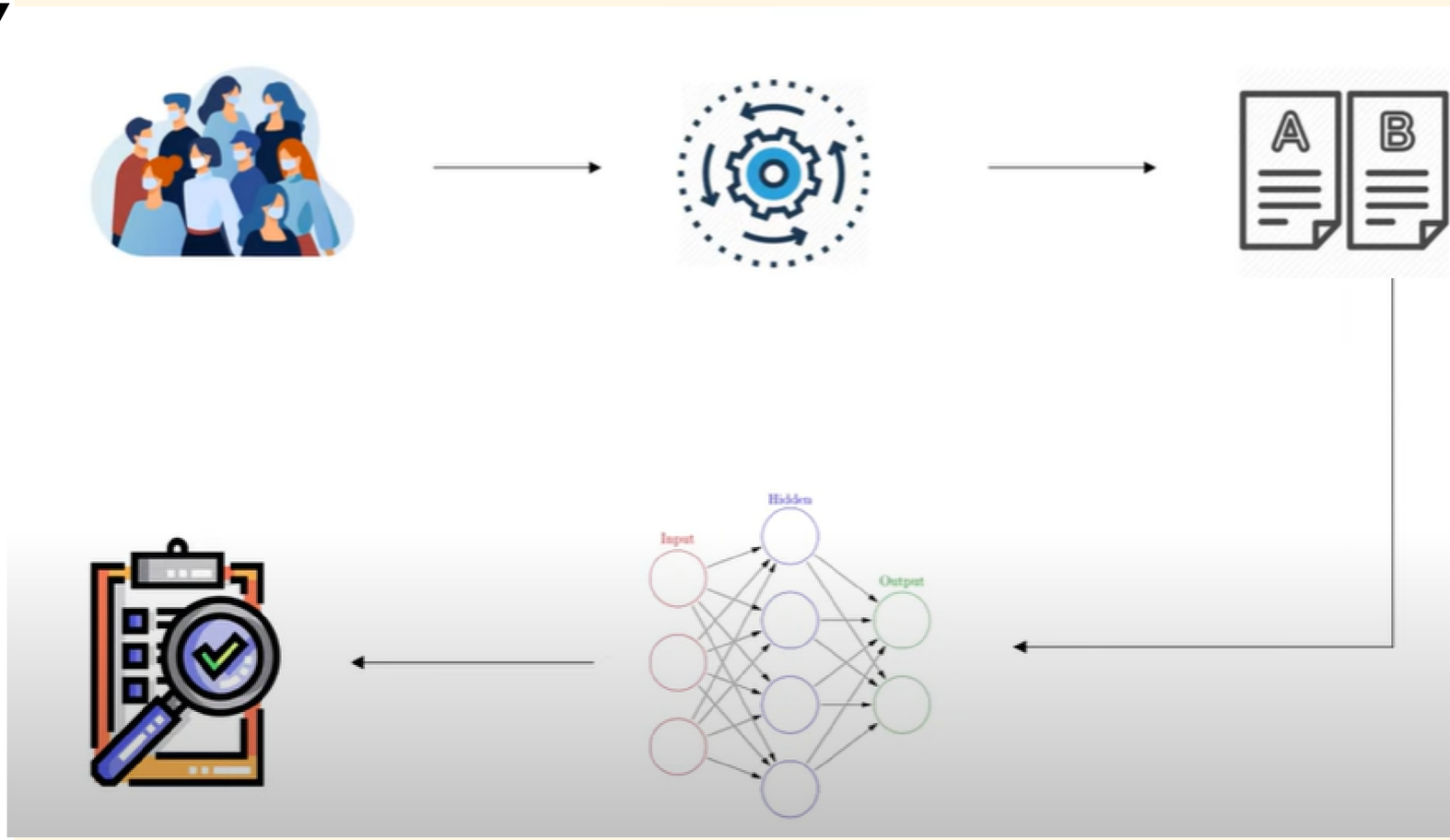
01 Data Set

02 Image PreProcessing

03 Train Test Split

04 CNN

05 Evaluation



DATA SET

https://www.kaggle.com/datasets/omkargurav/face-mask-dataset

7553 Images

3725 Images of Face with out Mask

3828 Images of Face with Mask

IMAGE PREPROCESSING

- Image size 128X128
- Converting images into numpy arrays

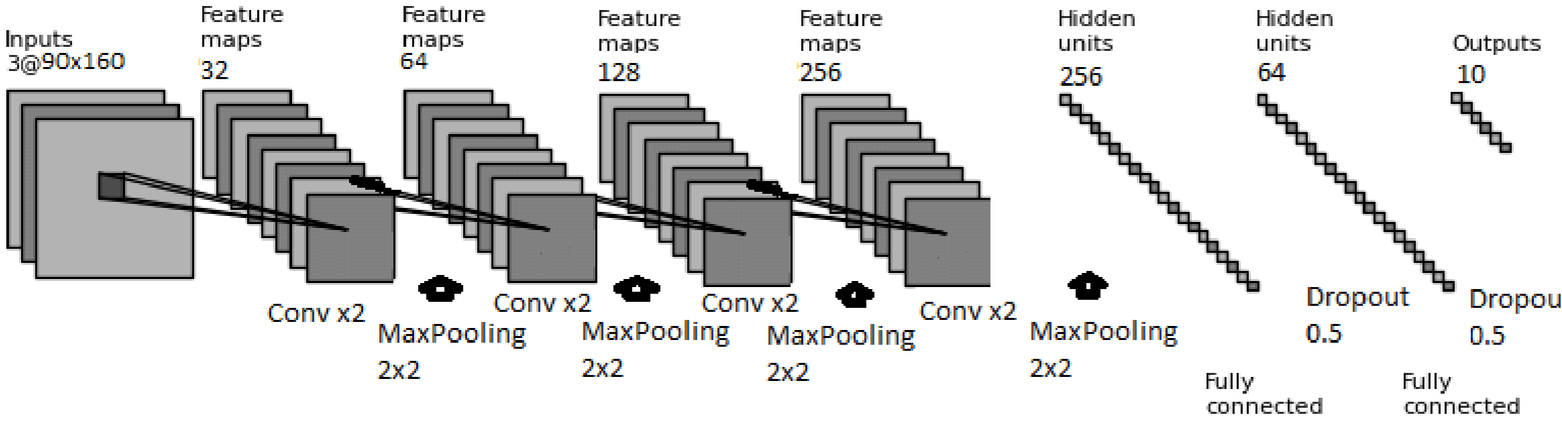
TRAIN TEST

- Train - 80%
- Test - 20%

WHY NOT ANN?

- Flattening multi-dimensional data, such as images, can lead to the loss of spatial information.
- Networks with a high number of trainable parameters tend to train more slowly due to the increased number of computations involved in each training iteration.
- Having a large number of trainable parameters increases the risk of overfitting.
- Attackers may try to infer sensitive information about the training data by analyzing the trainable parameters of a trained model.

WHY CNN?



- Convolutional neural network (CNN) is a special type of neural network that is designed to process data through multiple layers of arrays. A CNN is well-suited for applications like image recognition and is often used in face recognition software.

Step 1: Preprocessing

Step 2: Convolutional Layers

Step 3: Activation Functions and Non-linearity

Step 4: Pooling Layers (Downsampling)

Step 5: Flattening and Fully Connected Layers

Step 6: Output Layer

Step 7: Post-processing for Detection

Step 8: Visualization

Step 9: Model Evaluation and Tuning

CONCLUSION

In conclusion, the mini-project successfully demonstrated the implementation of Convolutional Neural Networks (CNNs) for face mask detection. By training on a dataset containing masked and unmasked faces, the CNN model showcased its ability to accurately classify whether individuals were wearing masks or not.

THANK YOU

Nandini Venishetty

Rohan Gadewar

Akshaya Garrepalli