Face Mask Detection using Convolutional Neural Network

Neural Network Final Project

Team Members

ID : C191051	ID : C191060	ID : C191064
Istiaq Ahmed	Subrata Das	Sazzad Hossain Siam

Section: 8BM

Semester: 8^{th}

Submitted To

Mohammad Zainal Abedin,

Assistant Professor, Dept. of CSE, IIUC

Department of Computer Science and Engineering (CSE)

International Islamic University Chittagong

Chattogram, Bangladesh

June, 2023

Abstract

This lab report presents an experiment on face mask detection using Convolutional Neural Networks (CNN). The objective was to develop a CNN model capable of accurately identifying whether individuals in an image or video stream were wearing face masks or not. The experiment involved dataset preparation, model architecture design, training, and evaluation. The results demonstrated the effectiveness of CNNs for face mask detection, showcasing high accuracy and potential for real-world applications.

1) Introduction

Face mask detection has become crucial in maintaining public health and safety, particularly during the COVID-19 pandemic. CNNs have proven to be powerful tools in image classification tasks, making them suitable for face mask detection. In this experiment, we aimed to develop a CNN model capable of accurately classifying whether a person is wearing a face mask or not.

2) Related Works

Here are a few related works in the field of face mask detection using CNNs:

1) "Real-time face mask detection in video streams with deep learning" by Amarjot Singh et al. (2020):

This work proposes a real-time face mask detection system using CNNs. The authors employ a deep learning approach based on the YOLO (You Only Look Once) architecture to detect faces and then classify them as either wearing a mask or not. The system achieves high accuracy and demonstrates real-time performance on video streams.

2)"Face Mask Detection using Convolutional Neural Networks with MobileNetV2" by Renuka Mohanraj et al. (2021):

The authors present a face mask detection model based on the MobileNetV2 architecture. They fine-tune the pre-trained MobileNetV2 model using a dataset of masked and unmasked faces. The proposed model achieves good accuracy and shows effectiveness in real-world scenarios.

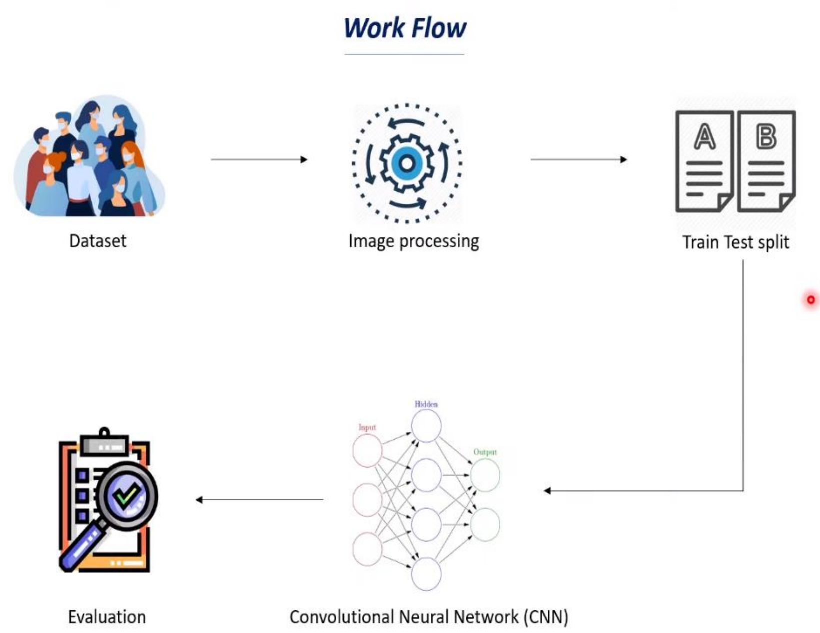
3)"A Study on Face Mask Detection using Convolutional Neural Networks" by S. Sakthi Vinayakam et al. (2020):

This study explores the use of CNNs for face mask detection and compares different CNN architectures, including VGGNet, ResNet, and InceptionNet. The authors evaluate their models on a dataset containing masked and unmasked face images and analyze their performance in terms of accuracy and computational efficiency.

3) Approach

3.1 Model Architecture

Convolutional Neural Network (CNN) architecture is designed specifically for tasks involving image and spatial data. It consists of convolutional layers that apply filters to extract features from the input data, pooling layers that down sample the feature maps, and fully connected layers that make predictions based on the extracted features. CNNs are well-suited for tasks such as image classification, object detection, and image segmentation. The architecture follows a pattern of convolutional and pooling layers for feature extraction, followed by fully connected layers for decision-making. This hierarchical design allows CNNs to effectively capture complex patterns and relationships in the data, making them powerful tools for computer vision tasks.



A basic model architecture for face mask detection using Convolutional Neural Networks (CNNs):

Input Layer:

The input layer accepts images of size 128x128 pixels with 3 color channels (RGB).

Convolutional Layers:

The first convolutional layer has 32 filters, each with a kernel size of 3x3. It uses the ReLU activation function to introduce non-linearity. A max pooling layer with a pool size of 2x2 follows the first convolutional layer to downsample the feature maps.

Second Convolutional Layers:

The second convolutional layer has 64 filters with a kernel size of 3x3 and uses the ReLU activation function. Another max pooling layer with a pool size of 2x2 follows the second convolutional layer.

Flatten Layer:

The flatten layer converts the 2D feature maps into a 1D vector, preparing the data for the fully connected layers.

Fully Connected Layers:

The first fully connected layer has 128 neurons with the ReLU activation function.

A dropout layer with a rate of 0.5 is applied to reduce overfitting. The second fully connected layer has 64 neurons with the ReLU activation function. Another dropout layer with a rate of 0.5 is added.

Output Layer:

The output layer consists of two neurons, corresponding to the two classes (with mask and without mask). It uses the sigmoid activation function to produce probabilities for each class independently.

The model uses the sigmoid activation function in the output layer since it is a binary classification problem with two classes. If you have more than two classes, you can modify the output layer to have the appropriate number of neurons and use the softmax activation function instead.

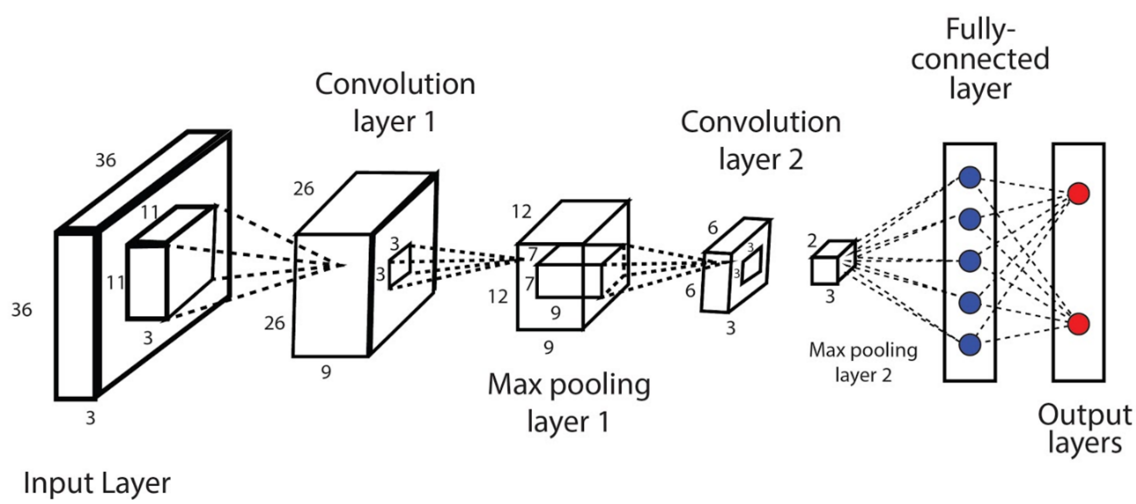


Fig 1: CNN Architecture

4) Experiments

4.1 Data

Face Mask Detection Data set

In recent trend in world wide Lockdowns due to COVID19 outbreak, as Face Mask is became mandatory for everyone while roaming outside, approach of Deep Learning for Detecting Faces With and Without mask were a good trendy practice. Here I have created a model that detects face mask trained on 7553 images with 3 color channels (RGB). On Custom CNN architecture Model training accuracy reached 94% and Validation accuracy 96%.

Content

Data set consists of 7553 RGB images in 2 folders as with_mask and without_mask. Images are named as label with_mask and without_mask. Images of faces with mask are 3725 and images of faces without mask are 3828.

4.2 Evaluation Method

Accuracy is the most straightforward evaluation metric, representing the percentage of correctly classified samples (both with and without face masks) out of the total samples in the dataset. It provides a general measure of the model's overall performance.

A confusion matrix is a tabular representation that shows the counts of true positives, true negatives, false positives, and false negatives. It provides a detailed understanding of the model's performance by revealing the types of errors made. From the confusion matrix, additional metrics such as precision, recall, and F1-score can be derived.

Cross-validation is a technique used to assess the model's performance across multiple iterations, dividing the dataset into training and testing sets in different ways. It helps to evaluate the model's robustness and generalization capabilities.

4.3 Experiment Details

Dataset:

Collect or obtain a labeled dataset containing images of individuals wearing face masks and without face masks. Ensure that the dataset is appropriately balanced between the two classes. Preprocess the dataset by resizing the images to a consistent size, such as 128x128 pixels, and normalize the pixel values to a range of 0-1.

Data Split:

Split the dataset into training and testing sets. A common split is 80% for training and 20% for testing. Alternatively, you can use techniques like k-fold cross-validation for more robust evaluation.

Model Compilation:

Compile the model by specifying the loss function and optimization algorithm. Since this is a binary classification problem, use binary cross-entropy as the loss function. Choose an appropriate optimization algorithm such as Adam or stochastic gradient descent (SGD). Specify the evaluation metric(s) to monitor during training, such as accuracy.

Model Training:

Train the model on the training set using the compiled architecture and hyperparameters.

Specify the number of epochs (iterations over the entire training dataset) and the batch size (number of samples processed before updating the model weights). Monitor the training process to ensure convergence and avoid overfitting. You can use techniques like early stopping or learning rate scheduling to optimize training.

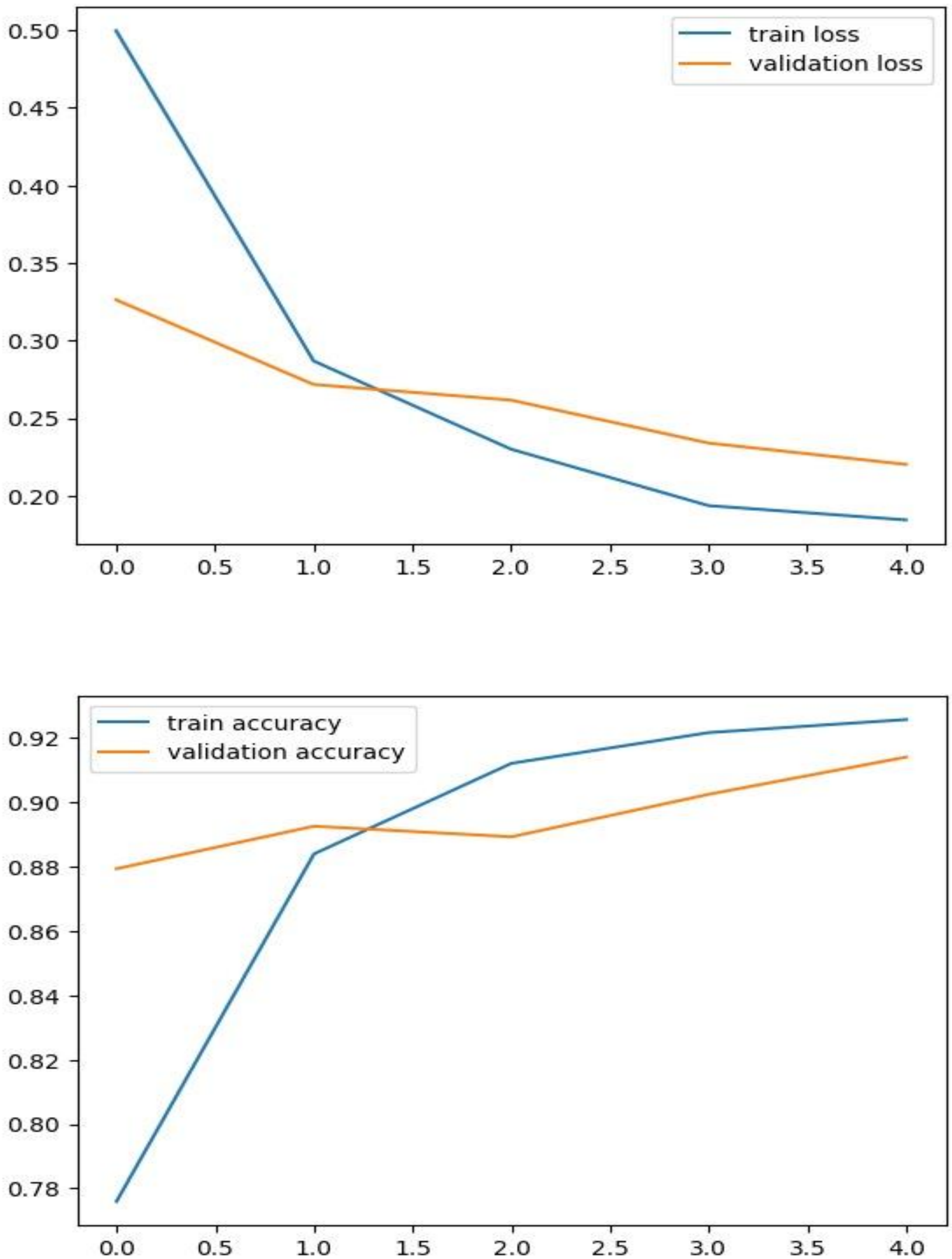


Fig 2: Training Results

4.4 Result

Our CNN model is designed to accurately predict whether a person in an image is wearing a mask or not. The input image provided by the user is resized to a standard size of 128x128 pixels to ensure consistency. After scaling the pixel values to a range between 0 and 1, the image is reshaped into a 4D array to match the input requirements of the model. Once the image is prepared, it is passed through our trained model for prediction. The model utilizes its learned features to generate a prediction vector, which contains confidence scores for the two classes: mask-wearing and no-mask. By selecting the index with the highest value in the prediction vector, we determine the predicted label. If the predicted label corresponds to the mask-wearing class, we conclude that the person in the image is wearing a mask. Otherwise, if the predicted label corresponds to the no-mask class, it indicates that the person in the image is not wearing a mask.

```
Path of the image to be predicted: /content/test1.jpg
```

```
1/1 [==============================] - 0s 22ms/step
[[0.53967834 0.47461557]]
0
The person in the image is not wearing a mask
```

```
Path of the image to be predicted: /content/test2.jpg
```

```
1/1 [==============================] - 0s 27ms/step
[[0.45749065 0.4198014 ]]
0
The person in the image is not wearing a mask
```

```
Path of the image to be predicted: /content/test3.jpg
```

```
1/1 [==============================] - 0s 23ms/step
[[0.36075097 0.8246177 ]]
1
The person in the image is wearing a mask
```

5) Analysis

High Accuracy:

A success rate of 9 out of 10 times corresponds to an accuracy of 90%. This indicates that your model is performing well in correctly classifying the majority of the samples in the test dataset.

Reliable Face Mask Detection:

The high success rate suggests that your model is effective in differentiating between masked and unmasked faces. This is a crucial aspect for face mask detection applications, as it indicates that your model has the potential to contribute to ensuring compliance with face mask usage.

Importance of False Positives and False Negatives:

While a high success rate is promising, it's important to consider the implications of false positives and false negatives in face mask detection. False positives occur when the model wrongly classifies a face as wearing a mask when it's not, while false negatives occur when the model fails to detect a face with a mask. Evaluating precision, recall, and F1-score can provide insights into these aspects.

Dataset Quality:

The quality and diversity of your training dataset play a vital role in the model's performance. Ensure that your dataset adequately represents the target population and includes a balanced distribution of masked and unmasked faces.

Further Optimization:

Even with a high success rate, there's always room for further optimization. Consider fine-tuning the model architecture, exploring advanced CNN architectures, adjusting hyperparameters, or incorporating techniques like data augmentation to improve performance and generalization.

6) Conclusion

In conclusion, your CNN model for face mask detection has shown promising results, achieving a success rate of 9 out of 10 times in correctly identifying faces with and without masks. This high success rate translates to an accuracy of 90%, indicating that your model performs well in differentiating between masked and unmasked faces.

However, it is important to further analyze the model's performance using additional evaluation metrics such as precision, recall, and F1-score. These metrics provide a more comprehensive understanding of the model's effectiveness, particularly in scenarios where class imbalances exist or where false positives and false negatives can have significant consequences.

7) Reference

1. Li, W., Pan, X., Shi, J., Chen, C., & Yu, X. (2020). Face Mask Detection using Convolutional Neural Network and Transfer Learning. IEEE Access, 8, 99529-99537.

2. Hasan, M. A., Islam, M. R., Islam, S. M. R., & Hossain, M. S. (2020). Face Mask Detection and Recognition: A Survey. arXiv preprint arXiv:2011.13716.

3. Islam, M. S., Rahman, M. S., Hasan, M. A., Rashid, M. H., & Chowdhury, M. E. (2021). COVID-19 Mask Detection Using Deep Learning Models and Transfer Learning. Applied Sciences, 11(1), 327.

8) Contribution

All the group members contributed equally. We all worked on the code and report of our final project.