International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 11, ISSN No. 2455-2143, Pages 01-14
Published Online March 2025 in IJEAST (http://www.ijeast.com)

ADVANCED FACE MASK DETECTION USING TRANSFER LEARNING AND CUSTOM CLASSIFIERS: ENHANCING PUBLIC SAFETY THROUGH COMPUTER VISION AND DEEP LEARNING

Naimul Hasan Shadesh, Arifur Rahaman, Sadia Tasnim Barsha, Salma Tabashum, Sabrina Tasnim
Dept. of Computer Science and Engineering,
Sonargaon University, Bangladesh

Abstract- **Face masks offer protection against air pollution and spread of the disease and should be worn for effortful distancing. Video cameras have proven useful upholding uniformity of mask-wearing using computer vision. Earlier mentioned methods based on convolutional neural network (CNN), YOLO (you only look once) and faster R-CNN support vector machines (SVM), and haar cascade techniques have had difficulties mainly for frontal view faces. This study provides nascent developments that will bring about remarkable changes in the field of public health as well as technologies with a unique mask detection system based on the latest computer vision combined with deep learning medicine. Using transfer learning and mobileNet V2, custom 'faceNet' and 'MaskNet' classifiers were run, which resulted in the 98,3% accuracy of the model - 97.87% for women with masks and 98,46 % for women without masks. This technology, in addition to mask effectiveness monitoring, can upgrade the approach in the CCTV monitoring process from performing passive surveillance to facilitating and enabling new technology that enhances safety and health practices.**

Keywords: **Mask detection, Computer vision, Deep learning, Matching Learning, MobileNetV2, FaceNet, MaskNet, Transfer learning, Surveillance, Advancement of Science and Technology, Safety, Change, Detecting faces, Model efficiency, Health precautions, Implications.**

I. INTRODUCTION

Face mask detection in computer vision is relevant in the future because it has been given prominence due to the COVID-19 pandemic. Wearing face masks remains one of the primary approaches to preventing viral infections [1], which explains why systems that can detect such actions are essential. Consequently, this concern has been addressed through the employment of a wide range of techniques from basic computer vision to modern-day deep learning to detect the presence of masks [2]. This projection aimed at promoting public health safety further led to the formulation of various effective methods against the use of facial masks for identification purposes bringing about a great blend of technology and public health. Face mask detection in computer vision is relevant in the future because it has been given prominence due to the COVID-19 pandemic. Wearing face masks remains one of the primary approaches to preventing viral infections [1], which explain why systems that can detect such actions are essential. Consequently, this concern has been addressed through the employment of a wide range of techniques from basic computer vision to modern-day deep learning to detect the presence of masks [2]. This projection aimed at promoting public health safety further led to the formulation of various effective methods against the use of facial masks for identification purposes bringing about a great blend of technology and public health [3].

With the winds of public safety driving its sails, this thesis navigates the turbulent waters of mask detection, meticulously plotting the most effective course for real-world implementation. By delving into mask complexities and overcoming occlusion's reefs, it unlocks the model's potential to safeguard communities, optimize real-time response, and pave the way for a future where mask detection becomes a seamless part of our daily lives [3, 4]. The complexity of face mask detection transcends mere identification; it grapples with multifaceted challenges crucial for the development of a robust system.

Variability in Mask Types: The diverse spectrum of face mask types - ranging from surgical masks to N95 respirators and cloth coverings - presents a convolution of appearances. Each variant introduces unique visual characteristics,

demanding adaptive models capable of discerning and categorizing masks irrespective of their type [2].
Occlusion and Partial Face Visibility: The presence of obstructions or partial facial visibility amplifies the intricacy of detection. In contexts where individuals don additional facial coverings like Niqab, the visibility of crucial facial landmarks might be compromised, posing substantial hurdles in accurately identifying the presence of a mask.Real-time Implementation: The efficacy of a face mask detection system hinges on its real-time applicability. Optimization for swift processing is imperative for seamless integration into public spaces, workplaces, and surveillance systems. Ensuring swift and accurate identifications fosters timely interventions and compliance monitoring [2].

II. LITERATURE REVIEW

As a pivotal facet of computer vision, face mask detection has gained significant traction due to the COVID-19 pandemic. A comprehensive review of pertinent literature encompasses various methodologies and advancements observed in this domain:
Ali, Amer, and Al-Tamimi (2022) conducted an extensive review of face mask detection methods, offering a panoramic insight into the techniques adopted across diverse studies. Their comprehensive analysis encompassed traditional methods alongside modern approaches, shedding light on the evolution of detection methodologies and their efficacy. Vipul et al. (2023) presented an innovative machine learning approach aimed at real-time face mask detection. Their emphasis on accuracy and efficiency enhancements showcases strides made in this field, particularly in addressing real-time deployment challenges. The literature extends beyond specific face mask detection studies, delving into broader AI and machine learning challenges: Musliner et al. (1995) highlighted the temporal constraints and computational challenges inherent in real-time AI systems, aligning with the demands faced in deploying real-time face mask detection systems. L'heureux et al. (2017) explored the challenges associated with machine learning applications with extensive datasets, providing valuable insights into handling computational complexities encountered in face mask detection models. Teboulbi et al. (2021) and Talahua et al. (2021) contributed to the discourse by emphasizing the practical implementation of AI-based face mask detection systems during the COVID-19 pandemic, accentuating the urgency and relevance of such technology in real-world scenarios.
Technical advancements in face mask detection methodologies have seen substantial progress: Lin et al. (2020) proposed an enhanced mask R-CNN, demonstrating improved facial recognition and segmentation capabilities [12]. Shi et al. (2019) explored face hallucination using reinforcement learning, showcasing advancements in high-resolution image generation techniques [13]. Saranya et al. (2021) and Sidik & Djamal (2021) significantly contributed to CNN-based face mask detection systems, showcasing advancements in accuracy and robustness, vital for real-world deployment. In seminal works, Shi et al. (2023) proposed a fascinating approach leveraging reinforcement learning for face hallucination, aiming to enhance image quality through attentive sequence optimization. Deng et al. (2024) introduced Arcface, a deep face recognition system emphasizing an additive angular margin loss mechanism, significantly improving face recognition performance [19, 20]. Liu et al. (2019) presented a novel backbone-branches architecture focusing on progressive representation learning for facial landmark detection, emphasizing its robustness and efficiency in multimodal scenarios. Recent studies by Saranya et al. (2021) and Sidik & Djamal (2021) have delved into the implementation of Convolutional Neural Networks (CNNs) for face mask detection. These works showcase the efficacy of CNNs in accurately identifying individuals wearing or not wearing masks, providing valuable insights into real-time applications in various contexts.
Moreover, Through adeptly harnessing transfer learning with MobileNetV2 as the foundational model and integrating a tailored custom classifier, this model includes both face detection (faceNet) and mask classification (maskNet), qualifying a comprehensive approach to accurately identify faces and differentiate between masked and unmasked individuals in diverse environmental conditions, contributing to the toughness and adaptability of the system.

III. PROPOSED METHODOLOGY FOR FACE MASK DETECTION

Transfer learning with MobileNetV2 takes center stage in face mask detection amidst the COVID-19 pandemic. By harnessing the power of MobileNetV2-based Convolutional Neural Networks (CNNs), researchers have significantly boosted the accuracy and efficiency of identifying mask-wearing models in real time, as illustrated in this diagram [2]. The proposed approach improves accuracy, proficiency, and authenticity. Use of diverse datasets, data augmentation, and transfer learning. Optimization for real-time performance using techniques like model pruning and quantization [3].
Public Data: The dataset for this research was assembled through a combination of public data and personal contributions. Photos were collected from a diverse range of sources, including friends, university colleagues, and publicly available images. These sources provide a diverse dataset, which is important for building a robust model capable of handling different facial orientations, lighting conditions, and mask types.

Personal Contributions: Colleagues and friends were asked to provide photos of themselves both with and without face masks. This subset of the dataset ensured a wide array of real-world conditions and environments, making the model more generalizable. Public Domain: Additional images were sourced from public domain websites to further enrich the dataset. This included images with varied demographics, ensuring the model's applicability across different population groups.

Web Scraping: To supplement the personal and public domain data, web scraping techniques were employed to collect images from several online repositories. The primary sources included Kaggle, GitHub, HackerNoon, and Defined.ai. These platforms offer vast datasets that are essential for training and validating machine learning models. Kaggle hosts numerous datasets on face masks and human faces. Relevant datasets were identified and scraped to augment the training data. Many researchers and practitioners share their datasets on GitHub. These repositories were explored, and appropriate datasets were integrated into the collection. Articles and repositories on HackerNoon often include links to valuable datasets. These were utilized to gather additional training images. This platform provides high-quality, annotated datasets that were crucial for enhancing the accuracy and reliability of the mode

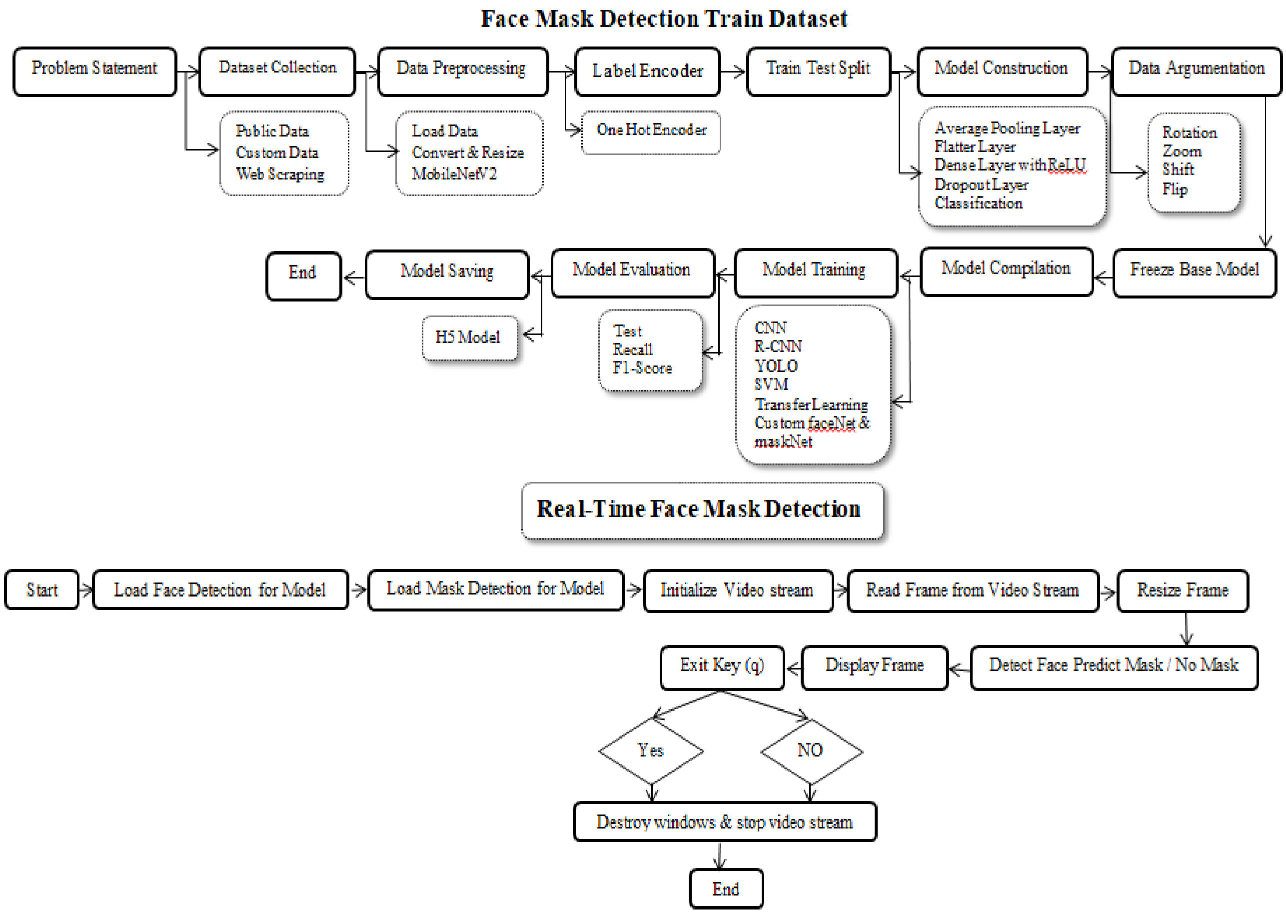

Fig.1: Individual steps for building real-time face-mask detection.

The flowchart in Fig 1 illustrates the comprehensive process involved in developing and deploying a face mask detection system. The process begins with defining the problem statement, which establishes the objective and scope of the project.

Next, the Dataset Collection phase involves gathering images from various sources: public data from the internet, custom data from friends and university colleagues, and additional images sourced through web scraping from platforms such as Kaggle, GitHub, HackerNoon, and Defined.ai. Following dataset collection, the Data Preprocessing stage includes encoding categorical labels using a label encoder and splitting the dataset into training and testing sets to prepare for model training. In the Model Construction phase, data augmentation techniques like rotation, zoom, shift, and flip are applied to increase the variability of the dataset. The preprocessed data is then loaded, and categorical labels are converted into a binary format using a one-hot encoder. The images are resized to fit the input requirements of MobileNetV2, which is the core architecture used in this project. Key components of the MobileNetV2 architecture include the average pooling layer, which reduces dimensionality while retaining essential features, and the flatten layer, which prepares the input for the dense layers. The dense layer with ReLU activation and dropout layer are used to introduce non-linearity and prevent overfitting, leading to the final classification output that predicts whether a person is wearing a mask or not. Once the model is constructed, it undergoes Model Compilation and Training. During this stage, the base model's weights are frozen to leverage the pre-trained layers, and the model is compiled with an appropriate loss function and optimizer. The training process is then carried out on the augmented dataset. After training, the model's performance is evaluated using metrics such as recall and F1-score. Satisfactory models are saved in H5 format for future deployment.

The final phase involves Real-Time Face Mask Detection. The system loads the face detection model and mask detection model, initializes the video stream, and continuously reads frames from the stream. Each frame is resized and processed to detect faces and predict mask status. The results are displayed in real-time, and the video stream can be excited by pressing a specific key (e.g., 'q'). The process ends with destroying the display windows and stopping the video stream, ensuring a clean termination of the program.

Kaggle: Kaggle hosts numerous datasets on face masks and human faces. Relevant datasets were identified and scraped to augment the training data.

GitHub: Many researchers and practitioners share their datasets on GitHub. These repositories were explored, and appropriate datasets were integrated into the collection.

HackerNoon: Articles and repositories on HackerNoon often include links to valuable datasets. These were utilized to gather additional training images.

Defined.ai: This platform provides high-quality, annotated datasets that were crucial for enhancing the accuracy and reliability of the model


(A)


(B)


(C)


(D)


(E)


(F)

Table 1: Representative Images from the No Face Mask Database Featuring Singles, Groups, and Couples.

Table 1 showcases representative images from the No Face Mask Database, classified into three types: singles, groups, and couples. These categories help ensure the dataset covers a variety of social continuity, enhancing the model's ability to detect face masks in diverse settings.

Custom Image Collection: To ensure the quality and relevance of the dataset, custom image collection practices were followed. This involved stringent quality control measures, dataset splitting strategies, and organized data storage solutions.

Quality Control: Every collected image was subjected to a quality check to ensure clarity, proper lighting, and relevance to the mask detection task. Images with obstructions, low resolution, or poor lighting were discarded.

Dataset Split: The dataset was split into training, validation, and testing sets. This split was crucial for training the model effectively and evaluating its performance. Typically, 70% of the data was used for training, 15% for validation, and 15% for testing.


(A)

(B)

(C)

(D)

(E)

(F)

Table 2: Representative Images from the Single and Group Face Mask Database.

Table 2 showcases representative images from the face mask database, including both single and group face images. These examples illustrate various scenarios, such as individuals wearing masks, not wearing masks, and groups with mixed mask usage, providing an expansive view of the dataset used for training the detection model. Data Storage: A structured data storage solution was implemented to manage the large volume of images efficiently. Images were organized into folders based on their category (masked or unmasked) and further subdivided into training, validation, and testing sets. Metadata, including image source and quality ratings, was maintained in a centralized database to facilitate easy retrieval and management.

Description of the Face Mask Detection Dataset: The Face Mask Detection Dataset is a comprehensive collection of images curated to train and evaluate the performance of the face mask detection model. This dataset is meticulously compiled from various sources to ensure diversity and robustness.

Single Face Images:

- With Mask: These images feature individuals wearing face masks. They encompass various orientations, lighting conditions, and mask types to represent real-world scenarios.
- Without Mask: These images display individuals without face masks, captured in similar varied conditions to ensure the model can accurately distinguish between masked and unmasked faces.

Group Face Images:

- All Wearing Masks: Group photos where all individuals are wearing face masks. These images help the model learn to detect multiple masked faces within a single frame.
- Mixed Mask Usage: Group photos where some individuals are wearing masks, and others are not. This category is crucial for training the model to accurately identify and differentiate between masked and unmasked faces in complex scenarios.

The dataset was collected from three primary sources: personal contributions from friends and university colleagues, public domain images from the internet, and web scraping from platforms like Kaggle, GitHub, HackerNoon, and Defined.ai. The combination of these sources ensures a rich and varied dataset, essential for developing a robust face mask detection system capable of performing well in diverse real-world environments.

The Face Mask Detection Dataset is designed to provide comprehensive training and evaluation data, covering a wide range of conditions and scenarios to ensure the accuracy and reliability of the face mask detection model.

Model Training and Evaluation: An effectuation of a face mask detection model using the TensorFlow and Keras libraries. The model uses the MobileNetV2 architecture as a base model with excessive stratum to perpetrate binary classification for face mask detection. Model training is the process of teaching a machine learning model to perform a specific task. In the case of face mask detection, the model is trained to identify faces with and without masks [12].

My code leverages transfer learning, utilizing MobileNetV2 as a pre-trained backbone, to build a face mask detection model. Let's delve into the key stages of training and evaluation:

Data Loading and Preprocessing: The images are loaded from the specified directory and resized to \(224 \times 224\) pixels, the required input size for MobileNetV2. Preprocessing is done using the `preprocess_input` function from `tensorflow.keras.applications.mobilenet_v2`, which normalizes the pixel values to the format expected by the pre-trained MobileNetV2 model.

- Data Splitting: The dataset is divided into training and testing sets using the `train_test_split` function from `sklearn.model_selection`. This split ensures that the model is trained on a portion of the data and tested on a separate portion, allowing for an unbiased evaluation of the model's performance.

- Data Augmentation: To increase the diversity of the training data, data augmentation is performed using `Image Data Generator` from `tensorflow. keras. preprocessing. image`. Augmentation techniques include rotation, zooming, shifting, shearing, and horizontal flipping. These transformations help the model generalize better by simulating a variety of real-world scenarios.
- Model Architecture: The MobileNetV2 base model, pre-trained on the ImageNet dataset, is loaded from `tensorflow.keras.applications.MobileNetV2`. Additional layers are added on top of the base model to perform the final classification. These include convolutional layers, batch normalization, activation functions (ReLU), global average pooling, flattening, dense layers, and a dropout layer for regularization. The final dense layer uses a softmax activation function to output the probability of mask presence (0 for no mask, 1 for mask).

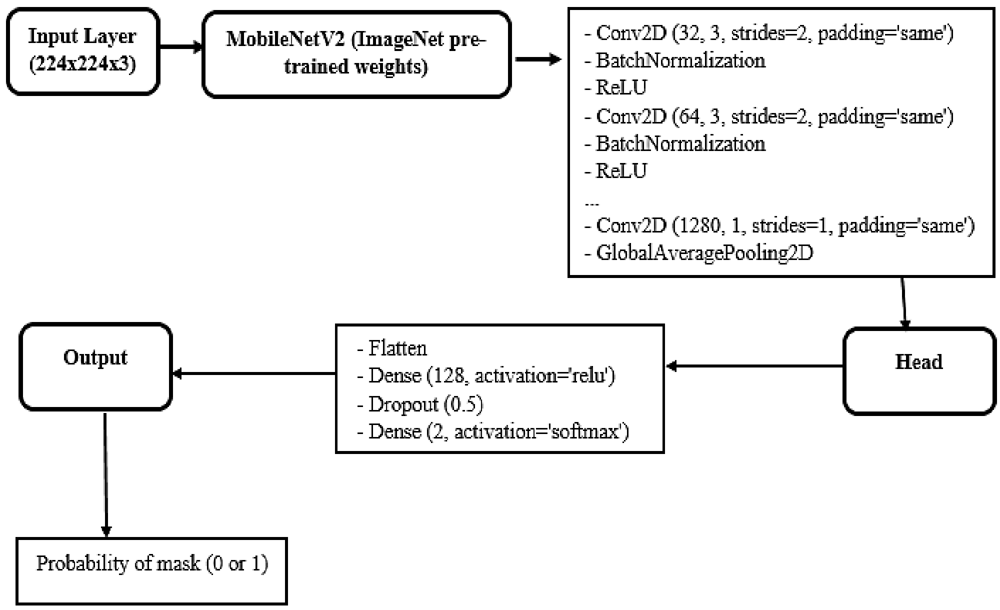

Fig.2: Model Training Architecture.

Fig 2 illustrates the architecture of the face mask detection model. The model employs MobileNetV2 with ImageNet pre-trained weights as the feature extractor, followed by a series of convolutional layers, batch normalization, and activation functions. The final output layer comprises fully connected dense layers with a dropout for regularization, concluding with a softmax activation to classify whether an individual is wearing a mask (1) or not (0).

Model Compilation:To optimize the model's performance, I employed the Adam optimizer and the binary cross-entropy loss function, while leveraging the accuracy metric for robust evaluation. [12]. Subsequently, training commences utilizing the augmented training data through the 'fit' function. This process entails iterating for a predetermined number of epochs (EPOCHS) to facilitate optimal model learning.

Data segmentation is pivotal; hence, the code enacts a 75%-25% split for training and testing, respectively, crucial for model evaluation [12]. By learning from the diverse training data, the model builds its skills. The testing set then acts as a proving ground, ensuring these skills translate to real-world scenarios.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 11, ISSN No. 2455-2143, Pages 01-14
Published Online March 2025 in IJEAST (http://www.ijeast.com)

```
Epoch 1/20
129/129 [==============================] - 120s 903ms/step - loss: 0.
Epoch 2/20
129/129 [==============================] - 103s 802ms/step - loss: 0.
Epoch 3/20
129/129 [==============================] - 101s 784ms/step - loss: 0.
Epoch 4/20
129/129 [==============================] - 100s 779ms/step - loss: 0.
Epoch 5/20
129/129 [==============================] - 102s 791ms/step - loss: 0.
Epoch 6/20
```

Fig.3: Training Head Model Load.

Fig 3 depicts the process of loading the training head model. This involves initializing the MobileNetV2 base model with pre-trained ImageNet weights and adding custom layers, including convolutional layers, batch normalization, activation functions, global average pooling, flattening, dense layers, and a dropout layer for regularization, concluding with softmax activation for classification. Model Training Accuracy: The model achieved the following accuracy values for the classes "with_mask" and "without_mask":

"with_mask" class accuracy: ~97.87%

"without_mask" class accuracy: ~98.46%

Please note that these accuracy values are approximate and may vary slightly each time you run the training process due to factors like random weight initialization and data shuffling during training [13]. The overall accuracy for the entire dataset, which includes both classes, is approximately 98%. This means that the model is performing well in detecting faces with and without masks. The legibility, recall, and F1-score for both classes are also high, which suggests that the model is constructing accurate predictions for each class [14].

	precision	recall	f1-score	support
with_mask	0.98	0.99	0.98	520
without_mask	0.99	0.98	0.98	517
accuracy			0.98	1037
macro avg	0.98	0.98	0.98	1037
weighted avg	0.98	0.98	0.98	1037

Fig.4: Training Model Accuracy.

Fig 4 shows the training model's accuracy, which reaches an impressive 98%. This high level of accuracy demonstrates the model's effectiveness in distinguishing between masked and unmasked individuals. Training model validity is a metric used to measure how well a machine learning model fulfills the data it was trained on. It indicates the degree to which the model's prognosis matches the actual target values in the training dataset [14]. Training exactitude is one of the key metrics used to assess a model's representation during training, but it's significant to note that it doesn't provide a complete picture of how well the model will perform on new, unseen data [15].

Model Evaluation: Model Prediction: After training, the model is used to make a vaticinator on the test set (testX) using predict. Performance Visualization: The code plots the training loss and accuracy over the epochs using matplotlib. pyplot to visualize the model's training progress.

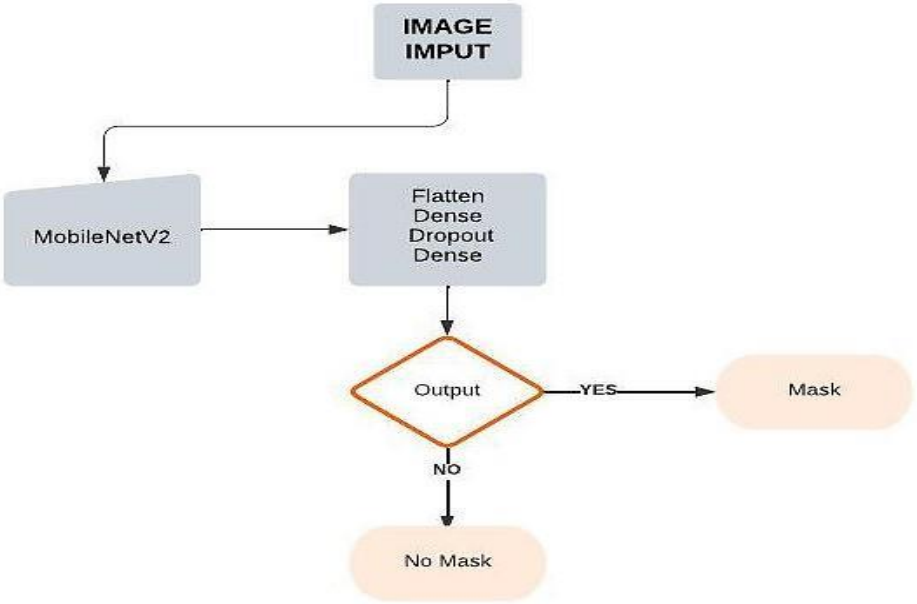

Fig.5: Model Evaluation Architecture.

Fig 2 illustrates the architecture of the face mask detection model. The model begins with an Image Input, which is processed through MobileNetV2 for feature extraction. The extracted features are then passed through a flattened layer and a series of dense layers with Dropout for regularization. The final output layer classifies the input as either "YES" (Mask) or "NO" (No Mask). Transfer learning leverages the property extraction capabilities of pre-trained networks, such as MobileNetV2, to initialize deep learning models for solving new tasks with fewer training data and faster convergence. [15].

Here's how transfer learning works:
Pre-trained Model: To achieve knowledge in an individual domain, such as image classification or natural language processing, a neural network undergoes rigorous training on a substantial dataset. This training is typically done on a voluminous amount of data and can be computationally costly and time-consuming [16].

Feature Extraction: Once the model is informed, the knowledge earned in the form of learned weights and representations from its hideaway layers can be deliberated valuable. Instead of discarding this knowledge, we can use the pre-trained model as a feature extractor [17]. The idea is to remove the final output layer(s) of the pre-trained model and retain the rest, turning it into a feature manufacturer [16].

Common Transfer Learning Models:
Image Classification: Popular pre-trained models for image classification encircle MobileNetV2. Keep in mind that transfer learning is telling when the tasks are related [17]. For illustration, a pre-trained model for image classification can be transferred to several image classification tasks, but it might not be as effective for a natural language processing task.

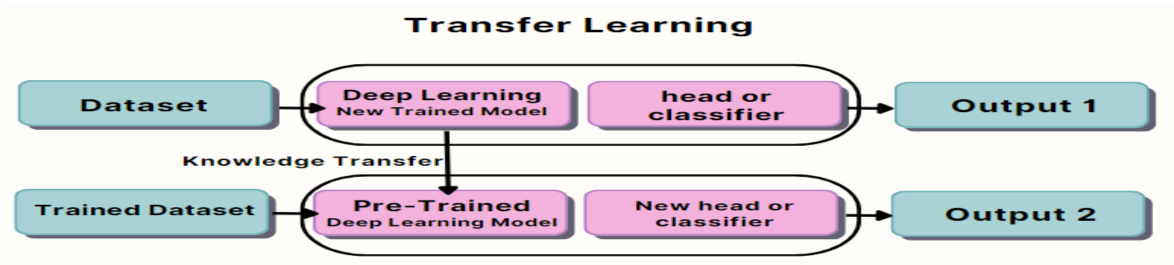

Fig.6: Transfer Learning Block Diagram.

Fig. 6 demonstrates the transfer learning block diagram, showcasing how pre-trained MobileNetV2 is leveraged for feature extraction and further fine-tuned for the face mask detection task. Real-Time Detection: This code is schematic for real-time face mask detection using a pre-trained MobileNetV2 model for face mask classification and a pre-trained face detection model. Here's how this code works:
Define the detect_and_predict_mask Function: The detect_and_predict_mask function takes a video frame, the face detection model (faceNet), and the face mask classification model (maskNet) as inputs. Within this function, it performs the following steps:

i. Use the faceNet model to detect faces in the frame using the cv2.dnn.blobFromImage method and forward pass.
ii. Preprocess the detected faces by converting them to RGB, resizing to (224, 224), and applying necessary transformations using img_to_array and preprocess_input [18].
iii. Feed the preprocessed faces to the maskNet model to predict whether each face is wearing a mask or not.
iv. Return the face locations and corresponding mask predictions.

Load Pre-Trained Models:The code loads the pre-trained MobileNetV2 model for face mask classification from the "mask_detector.model" file using load_model and the pre-trained face detection model from the provided files using cv2.dnn.readNet.

Start Video Stream: The code starts the video stream using VideoStream (0). Start (), where the argument "0" represents the default camera (webcam). This enables the program to capture video frames in real-time.
Real-Time Face Mask Detection Loop: In real-time face mask detection, the code enters a continuous loop to process video frames from a live stream. Each frame is resized to a width of 800 pixels using the imutils. resize function to standardize input dimensions. Within the loop, the detect_and_predict_mask function is called to identify faces within the frame and predict whether each face is wearing a mask or not. Upon detection of a face, a rectangle is superimposed on the frame to highlight the location of the face, and the mask detection result is displayed adjacent to the rectangle. Faces classified as wearing a mask are labeled with a green rectangle and corresponding text ("Mask"), while faces without masks are labeled with a red rectangle and text ("No Mask"). This visual feedback provides real-time updates on mask adherence within the video stream.
Finally, the processed frame, with overlaid rectangles and labels indicating mask detection results, is displayed using cv2.imshow. This real-time display allows for immediate monitoring and assessment of mask compliance, leveraging computer vision techniques to enhance public health safety measures in real-world applications.
Termination: The code suffices the real-time detection loop until the user presses the "q" key. When the "q" key is pressed, it breaks out of the loop, closes the video stream, and demolishes all OpenCV windows using cv2.destroyAllWindows () and vs.stop().

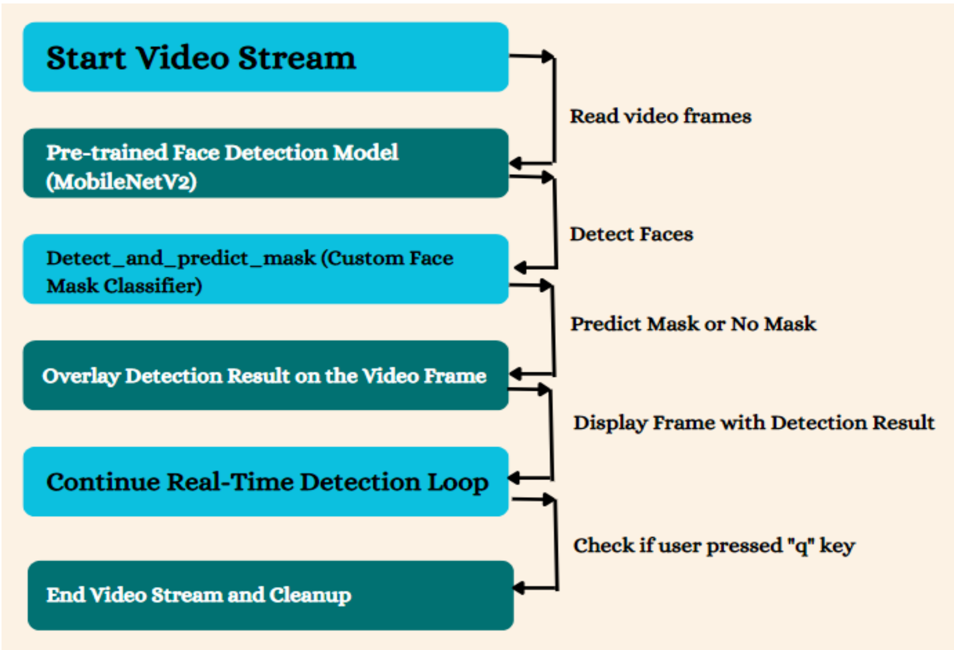

Fig.7: Real Time Detection Architecture.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 11, ISSN No. 2455-2143, Pages 01-14
Published Online March 2025 in IJEAST (http://www.ijeast.com)

The face masks detection system depicted in Fig 7, initiates with the start of a video stream, enabling real-time frame capture and analysis. Utilizing the MobileNetV2 model, pre-trained for its effective face detection capabilities, the system identifies faces within each frame. These detected faces are then processed by a custom classifier, "maskNet," specifically designed to predict whether each face is wearing a mask or not. This classifier leverages deep learning techniques and was trained on a diverse dataset collected from multiple sources, including personal contributions and publicly available images.

Upon predicting mask presence, the system overlays the detection results onto the video frame, providing immediate visual feedback on mask compliance. This process continues in a loop, continuously reading new frames, detecting faces, predicting mask status, and updating the display in real-time. The system also includes functionality to monitor user input, such as detecting if the "q" key is pressed to terminate the video stream gracefully and perform necessary cleanup tasks. Overall, this approach combines robust face detection with specialized mask detection, offering a practical solution for real-time monitoring of mask adherence in various settings.

IV. RESULT ANALYSIS AND DISCUSSIONS

This section delves into the effect of our study, presenting a meticulous analysis and contemplative discussions encapsulated in four individual sub-sections:

Experiments on Model Training
Analysis of Results for Different Combinations
Discussions

Experiments on Model
All the previous research papers that have been done in this context of face mask detection have adopted methods like CNN, YOLO, SVM, Haar Cascades, etc. and we have found some problems in each of their projects. Comparatively, their projects had better result accuracy. For face max detection we used transfer learning in deep learning in our project and in that we used MobileNetV2 and used FaceNet and MaskNet for its classifier by using these we made some changes between face net and marks net which gave us better accuracy and our project is fully functional our project can work in low light and high light as well as detect sunglasses and different types of face masks and Niqab with some ability to detect the common Niqab of Muslim society. Based on this we can say our project has been able to give better accuracy. Below written steps everything is discussed and explained in pictorial form [20].

Model Construction: The MobileNetV2 model served as the base, augmented with custom layers for binary classification (mask vs. no mask). The model was compiled with an Adam optimizer and trained over 20 epochs, ensuring effective learning and generalization [21].

Evaluation and Saving: The model's performance was evaluated on the test set, and meticulous metrics including precision, recall, and F1-score were computed. The model, along with training plots, was saved in H5 format for future applications [22].

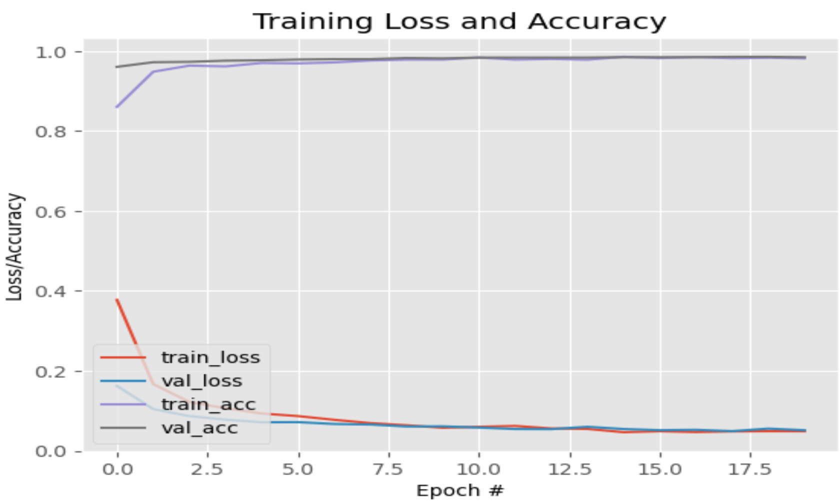

Fig.8: Training Loss and Accuracy.

Fig 8 illustrates the training loss and accuracy of the face mask detection model over successive epochs. The training loss decreases steadily, indicating improved model performance and fit to the training data. Concurrently, the accuracy increases, demonstrating the model's enhanced ability to correctly classify masked and unmasked faces as training progresses.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 11, ISSN No. 2455-2143, Pages 01-14
Published Online March 2025 in IJEAST (http://www.ijeast.com)

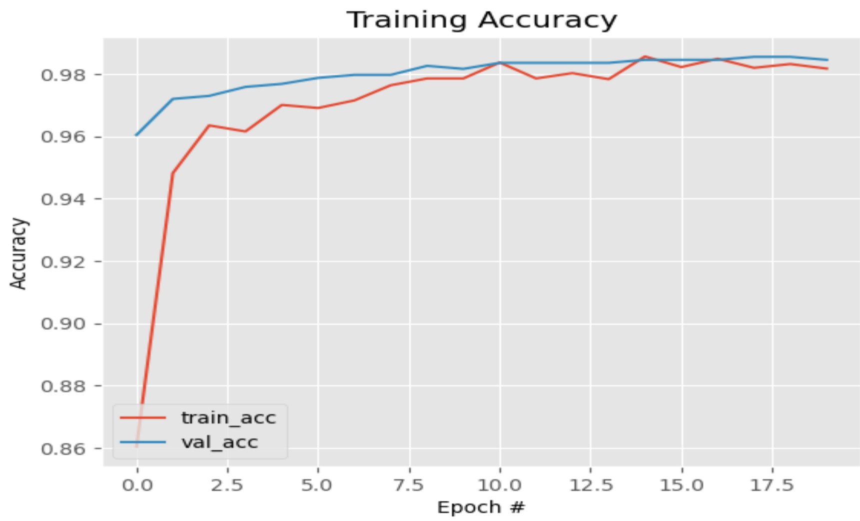

Fig.9: Training Accuracy.

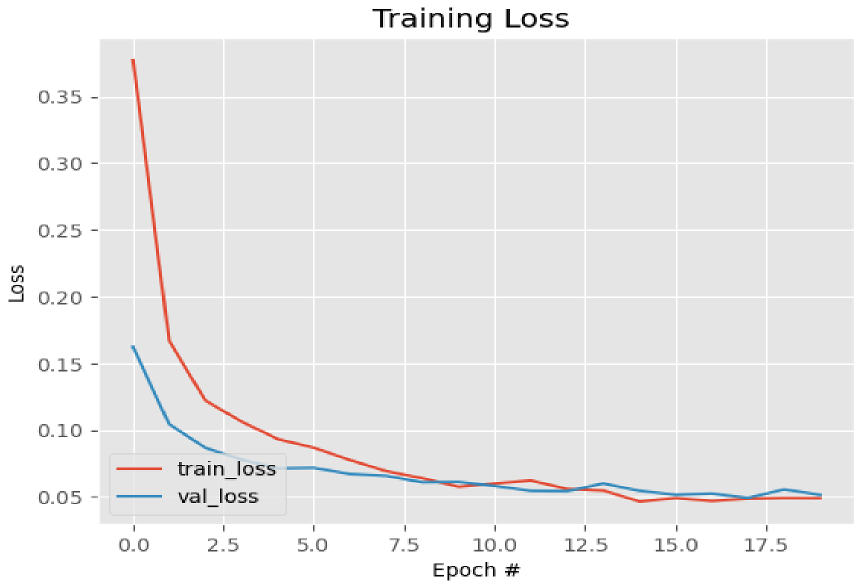

Fig.10: Training Loss.

Fig 9 shows the training accuracy of the face mask detection model over time. The graph highlights a consistent increase in accuracy, demonstrating the model's growing proficiency in correctly classifying masked and unmasked faces as training progresses. Fig 10 displays the training loss, which decreases steadily over the training period. This decline in loss indicates the model's improved fit to the training data, reflecting enhanced learning and reduced error rates.

Experiments with MobileNetV2-based Pre-trained Model The study highlighted the sharpness of protection of the face mask detection model in the H5 format, ensuring portability, compression, and standardization [6].

Real-time Face Mask Detection: The real-time detection system, processing video frames through the trained model, was presented. The effectiveness of this system in various scenarios was underscored; strengthen its practicality in public health settings [22].

Analysis of Results for Different Combinations: To assess the model comprehensively, diverse experiments were directed, exploring different hyper parameters and techniques [20]. Fine-tuning strategies, optimizer variations, data augmentation methods, batch size and epoch's configurations, and regularization techniques were meticulously analyzed [17]. Ethical considerations, including fairness evaluations, were undertaken to ensure unbiased performance across demographic groups [24].

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 11, ISSN No. 2455-2143, Pages 01-14
Published Online March 2025 in IJEAST (http://www.ijeast.com)

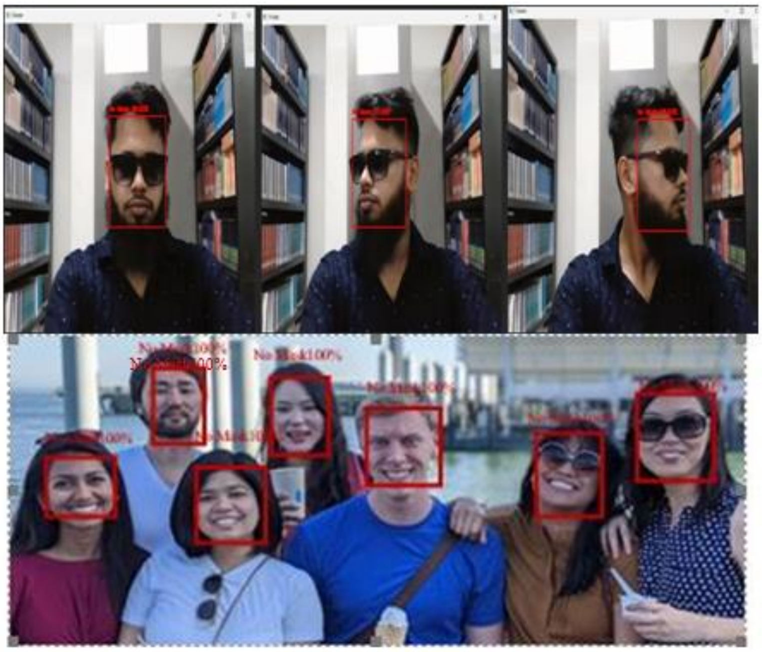

Fig.11: No Mask (Single and Group).

Fig 11 presents the detection results for individuals and groups not wearing masks. The model effectively identifies and labels faces without masks, both in single and group settings. This demonstrates the model's capability to accurately detect non-compliance with mask-wearing protocols, even in more complex scenarios involving multiple individuals. Embrace the mask, shield our community. It’s more than fabric; it’s a silent signal of care, a call for caution to safeguard our collective well-being [24].

Fig.12: Different Type Mask & Sunglass Detection Successfully.

International Journal of Engineering Applied Sciences and Technology, 2025
Vol. 9, Issue 11, ISSN No. 2455-2143, Pages 01-14
Published Online March 2025 in IJEAST (http://www.ijeast.com)

Fig 12 showcases the model's successful detection of various types of masks and sunglasses. The results demonstrate the model's versatility and robustness in identifying faces with different mask styles and additional accessories like sunglasses, ensuring accurate compliance monitoring across a range of real-world conditions.

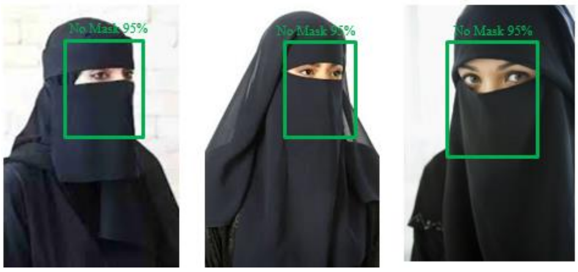

Fig 13: Niqab also Mask Detection Successfully.

Fig 13 demonstrates the model's successful detection of niqabs as masks. This result highlights the model's capability to accurately recognize traditional face coverings, such as niqabs, as compliant with mask-wearing requirements, showcasing its effectiveness across diverse cultural practices. Successful detection of face masks demonstrates the efficacy of our model in identifying individuals wearing protective face coverings.

V. CONCLUSION AND FUTURE WORDS

This research leverages deep learning architectures within the realm of computer vision to achieve accurate face mask detection. It employs MobileNetV2 as the base, enhanced by a custom classifier that combines face detection (faceNet) and face mask classification (maskNet). The results exhibit remarkable accuracy: 97.87% for masked faces and 98.46% for unmasked ones, culminating in an overall accuracy of 98.33%. Detecting masks significantly impacts public health, especially in preventing virus transmission. Robust models like this can monitor mask compliance in public areas, workplaces, and gatherings. The model's practicality in real-world scenarios highlights its potential for widespread deployment, offering a powerful tool to support global health initiatives, enhance public safety, and promote community well-being.

Future Works: While our research has shown promising face mask detection results, several avenues for improvement remain. Future work will focus on real-time deployment in dynamic environments, advanced data augmentation techniques to enhance training diversity, and the integration of additional sensor data such as temperature or audio cues. These enhancements will refine model performance by reducing overfitting and increasing accuracy. Additionally, exploring the use of federated learning could further enhance privacy and data security. By incorporating multi-modal data sources and more sophisticated neural architectures, we can further improve detection robustness. By pursuing these advancements, we aim to support public health initiatives and ensure global community safety through robust face mask detection systems.

ACKNOWLEDGMENT

This research work has been supported by the Research Cell, Sonargaon University (SU) Dhaka-1215.

REFERENCES

[1] Mohammed Ali, Firas Amer, and Mohammed SH Al-Tamimi. (2022). Face mask detection methods and techniques: A review. International Journal of Nonlinear Analysis and Applications, **13**(1), 3811-3823.

[2] Khan, S. H., & Shah, S. A. (2020). A Review on the Evolution of Mask Detection Approaches During COVID-19. IEEE Access, **8**, 101780-101801. doi:10.1109/ACCESS.2020.2993326

[3] Provost, F., & Fawcett, T. (2013). Data Science for Business: What You Need to Know about Data Mining and Data-Analytic Thinking. O'Reilly Media.

[4] Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.

[5] Halevy, Alon, Peter Norvig, and Fernando Pereira. "The unreasonable effectiveness of data." IEEE intelligent systems 24.2 (2009): 8-12.

[6] Furlong, Matthew Murphy. The Politics of Invasion: Northern Mexican Public Housing, Land Struggle History, and the Twilight of an Infrastructure-Building State. Diss. The University of Chicago, 2021.

[7] Zhang, Zihao, and Ben Bowes. "The future of artificial intelligence (AI) and machine learning (ML) in landscape design: A case study in Coastal Virginia, USA." arXiv preprint arXiv:2305.02327 (2023).

[8] L'heureux, A., Grolinger, K., Elyamany, H. F., &Capretz, M. A. (2017). Machine learning with big data: Challenges and approaches. Ieee Access, 5, 7776-7797.

[9] Takkar, Sakshi, et al. (2021). Advanced ATM security system using Arduino Uno. 2021 9th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), IEEE, pp. 426-431.

[10] Karanam, Santoshachandra Rao, Y. Srinivas, and M. Vamshi Krishna. (2020). Study on image processing using deep learning techniques. Materials Today: Proceedings, **10**, 2020.

[11] Zhuang, Y.T., Wu, F., Chen, C., & Pan, Y.H. (2017). Challenges and opportunities: from big data to knowledge in AI 2.0. Frontiers of Information Technology & Electronic Engineering, **18**, pp. 3-14.

[12] James, G., Witten, D., Hastie, T., &Tibshirani, R. (2013). An Introduction to Statistical Learning. Springer.

[13] Sharma, N., Sharma, R., & Jindal, N. (2021). Machine learning and deep learning applications—a vision. Global Transitions Proceedings, **2**(1), pp. 24-28.

[14] Brownlee, Jason. Data preparation for machine learning: data cleaning, feature selection, and data transforms in Python. Machine Learning Mastery, 2020.

[15] Lin, W., T. Tong, Q. Gao, D. Guo, X. Du, Y. Yang, G. Guo, M. Xiao, M. Du, and X. Qu. (2018). Convolutional neural networks-based MRI image analysis for the Alzheimer's disease prediction from mild cognitive impairment. Frontiers in Neuroscience, **12**, 777.

[16] Traboulsee, A., et al. "Revised recommendations of the consortium of MS centers task force for a standardized MRI protocol and clinical guidelines for the diagnosis and follow-up of multiple sclerosis." American Journal of Neuroradiology 37.3 (2016): 394-401.

[17] Winzeck, Stefan, et al. "Ensemble of convolutional neural networks improves automated segmentation of acute ischemic lesions using multiparametric diffusion-weighted MRI." American Journal of Neuroradiology 40.6 (2019): 938-945.

[18] Qin, Bosheng, and Dongxiao Li. "Identifying facemask-wearing condition using image super-resolution with classification network to prevent COVID-19." Sensors 20.18 (2020): 5236.

[19] Lin, Kaihan, et al. "Face Detection and Segmentation Based on Improved Mask R-CNN." Discrete dynamics in nature and society 2020.1 (2020): 9242917.

[20] Shi, Yukai, et al. "Face hallucination by attentive sequence optimization with reinforcement learning." IEEE transactions on pattern analysis and machine intelligence 42.11 (2019): 2809-2824.

[21] Deng, Jiankang, et al. "Arcface: Additive angular margin loss for deep face recognition." Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2019.

[22] Liu, Lingbo, et al. "Facial landmark machines: A backbone-branches architecture with progressive representation learning." IEEE Transactions on Multimedia 21.9 (2019): 2248-2262.

[23] Saranya, G., et al. "Face Mask Detection using CNN." 2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT). IEEE, 2021.

[24] Idoko, John Bush, and Emirhan Simsek. "Face Mask Recognition System-Based Convolutional Neural Network." Machine Learning and the Internet of Things in Education: Models and Applications. Cham: Springer Nature Switzerland, 2023. 21-46.