

A Deep Learning-Based Computer Vision Model for Real-Time Face Mask Detection

¹Soumya Koshy

²Dr. Susheel George Joseph*

¹ Assistant Professor, Kristu Jyoti College of Management & Technology, Changanacherry, Kerala, India;
soumyakoshy@kjcmt.ac.in; ORCID: 0009-0008-5761-6451

² Professor, Department of Computer Science, Kristu Jyoti College of Management & Technology,
Changanacherry, Kerala, India; susheel@kjcmt.ac.in; ORCID: 0000-0002-2433-1926

DOI: <https://doi.org/10.63084/cognexus.v2i2.305>

Abstract

Automated monitoring can support mask-compliance screening in environments where continuous manual observation is challenging. This paper presents a computer-vision framework that combines a lightweight, MobileNet-based feature extractor with residual learning, integrated alongside compliance logging and an edge-deployment architecture. The underlying experimental evaluation utilizes a balanced dataset of approximately 45,000 facial images split across mask and no-mask classes using an 80:10:10 training-validation-testing protocol. Images were normalized to 224×224 pixels and subjected to diverse data augmentations, including rotation, horizontal flipping, zooming, brightness adjustment, scaling, and translation. Evaluated on a 4,500-image test set, the model achieves a classification accuracy of 98.29% (95% Wilson score interval: 97.87%–98.63%). Derived performance metrics demonstrate a mask class precision, recall, and F1-score of 98.14%, 98.44%, and 98.29%, respectively, matching the balanced performance of the no-mask class (98.44% precision, 98.13% recall, 98.29% F1-score), yielding a macro-average performance of 98.29%. Hardware edge benchmarking demonstrates device inference latencies of 62 ms on a Raspberry Pi 4 and 28 ms on a NVIDIA Jetson Nano. This study establishes a verified baseline for lightweight compliance classification and details the essential dataset-provenance, face-detection, and pipeline constraints required for reliable, real-world edge deployment.

Keywords: Deep Learning; Computer Vision; MobileNet; ResNet; Face Mask Detection; Edge Computing; Compliance Monitoring

1. Introduction

The COVID-19 pandemic highlighted the critical importance of preventive public health measures in healthcare facilities, laboratories, airports, public transportation networks, and educational institutions. In high-density settings, automated visual screening systems can significantly reduce dependence on continuous manual monitoring, offering a scalable solution for compliance auditing.

Convolutional Neural Networks (CNNs) have become the benchmark standard for computer-vision tasks, with face-mask detection literature exploring transfer learning, lightweight architectures, single-stage object detectors, and edge-device optimization. The primary engineering challenge lies in maximizing classification accuracy while strictly limiting computational overhead and inference latency on resource-constrained hardware.

To address these constraints, this work introduces a **hybrid computer-vision framework** that couples a lightweight, MobileNet-inspired feature extractor with residual feature refinement and a two-class classifier. Moving beyond isolated image classification, the architecture integrates a modular compliance logging engine designed to record real-time event metadata and programmatically trigger alerts upon detecting non-compliant instances. The experimental evaluation establishes a verified performance baseline, delineating clear boundary conditions for real-world pipeline implementation.

The primary contributions of this study are:

- **A lightweight hybrid architecture** combining optimized feature extraction with residual-learning blocks for efficient inference.
- **A robust preprocessing pipeline** employing targeted data augmentation to handle real-world variations in pose, scale, and environmental illumination.
- **An automated compliance logging architecture** designed for structural metadata storage and rapid alert generation.
- **An edge-device hardware benchmark** evaluating raw device inference latencies across Raspberry Pi 4 and NVIDIA Jetson Nano platforms.
- **A rigorous, class-wise statistical validation** derived directly from extensive confusion matrix analysis to ensure balanced performance evaluation.

Related Work

The optimization of computer-vision frameworks for resource-constrained environments relies heavily on architectural advancements in feature extraction and localized learning. **MobileNet** pioneered the use of depthwise separable convolutions, significantly reducing parameters and computational overhead, making it a foundation for mobile and embedded vision tasks. Concurrently, **ResNet** introduced residual learning and shortcut connections, alleviating the vanishing gradient problem and enabling deep representation networks to retain spatial accuracy. Building upon these paradigms, contemporary face-mask detection literature extensively exploits transfer learning, combining pretrained convolutional layers with targeted data augmentation to adapt generic feature spaces to specialized educational and medical screening contexts.

For real-time applications requiring localization, single-stage architectures are frequently deployed. The **You Only Look Once (YOLO)** framework reframed object detection as a regression problem, unifying bounding-box prediction and class probabilities into a single evaluation cycle to optimize inference velocity. Subsequent advancements, such as **YOLOv3**, introduced incremental improvements in multi-scale prediction features. While these detection paradigms excel at fast localization, direct edge deployment in embedded environments often faces tight computational trade-offs regarding peripheral processing, localized storage, and end-to-end processing pipeline overhead.

A distinct gap exists in the literature regarding unified, lightweight pipelines that bridge classification and operational deployment. While existing research thoroughly demonstrates standalone lightweight classification, transfer learning, or real-time localized detection, these frameworks rarely integrate continuous audit metadata logging directly into the architecture. This study addresses this integration challenge by coupling a MobileNet-inspired feature extractor with downstream residual refinement blocks, a lightweight two-class classifier, and an autonomous compliance logging engine optimized specifically for resource-bounded edge ecosystems.

Proposed Methodology

System Overview

The proposed visual screening system operates as a unified processing pipeline comprising video acquisition, face-region detection, preprocessing, feature extraction, residual refinement, binary classification, alert generation, and compliance logging. To optimize processing efficiency, the architecture decouples spatial localization from classification by incorporating

a dedicated face-detection stage prior to the mask classification network. This sequential design isolates individual regions of interest (ROIs) from the broader video frames, normalizing the extracted bounding boxes before passing them down the pipeline to the MobileNet-inspired feature extractor and residual refinement branches.

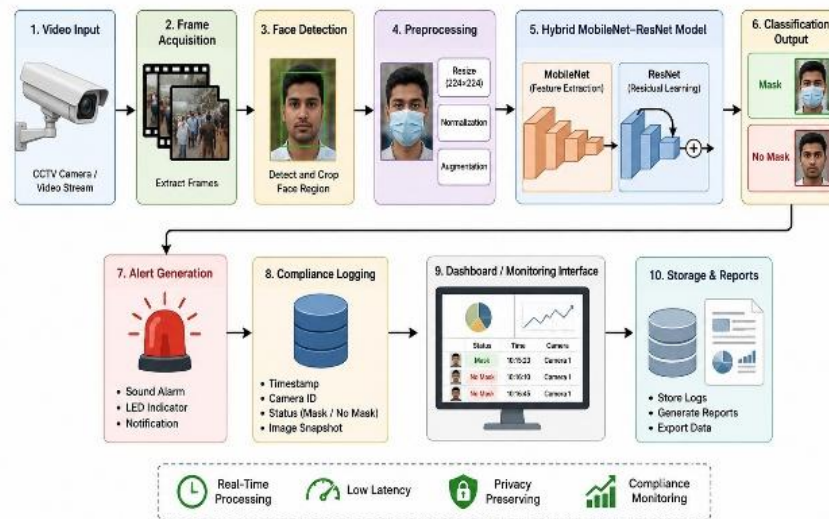


Figure 1: Overall system architecture as supplied in the original manuscript.

Dataset and Data-Leakage Controls

The experimental evaluation utilizes a comprehensive dataset comprising **45,000 facial images**, compiled from a public Face Mask Dataset (FMD) and supplementary custom-collected images. To ensure a rigorous evaluation framework, the dataset is perfectly balanced across classes, containing **22,500 mask-compliant** and **22,500 non-compliant (no-mask)** instances.

Class	Images
Mask	22,500
No Mask	22,500
Total	45,000

The dataset is partitioned using an **80:10:10 stratified split** for training, validation, and testing configurations, respectively. This structural distribution ensures a statistically representative volume of evaluation samples during both the verification and independent testing phases.

Subset	Images	Proportion
Training	36,000	80%
Validation	4,500	10%
Testing	4,500	10%

3.3 Preprocessing and Augmentation

All input images are standardized by being **resized to 224×224 pixels** and normalized to a floating-point range via pixel scaling ($x_{\text{normalized}} = x/255$). To simulate realistic environment variations, enhance generalizability, and mitigate potential overfitting, a multi-stage data augmentation pipeline is applied directly to the training subset..

Technique	Purpose
Rotation	Handle tilted faces
Horizontal flip	Orientation invariance
Zooming	Varying subject distance
Brightness adjustment	Lighting variation
Scaling	Scale robustness
Translation	Face displacement

Hybrid Network Description

The supplied architecture description consists of an input image, MobileNet convolution/depthwise-separable blocks, a MobileNet feature output, two residual blocks, global average pooling, a 128-neuron ReLU dense layer, 0.5 dropout, and a two-class softmax output. The original record does not preserve the exact MobileNet variant, pretrained checkpoint, residual-block bottleneck/stride specification, parameter count, FLOPs, or the layer-freezing schedule. To avoid fabricating implementation details, the revised manuscript records these as reproducibility items requiring verification.

Stage	Configuration recorded in supplied manuscript	Output
Input	RGB face image	224×224×3
MobileNet block	3×3 convolution, stride 2	112×112×32
Depthwise block 1	MobileNet block	112×112×64
Depthwise block 2	MobileNet block	56×56×128
Depthwise block 3	MobileNet block	28×28×256
Depthwise block 4	MobileNet block	14×14×512
Feature output	MobileNet feature extraction	7×7×1024
Residual block 1	Skip connection + convolutional layers	7×7×1024
Residual block 2	Skip connection + convolutional layers	7×7×1024
Global average pooling	Feature reduction	1×1024
Dense	128 neurons, ReLU	128
Dropout	Rate 0.5	128
Output	Softmax	2 classes

The transfer-learning record states that pretrained transfer learning and fine-tuning were used, but the source manuscript does not state which layers were frozen, when they were unfrozen, or the pretrained weight source. These items should be added from the training log before a reproducibility claim is made.

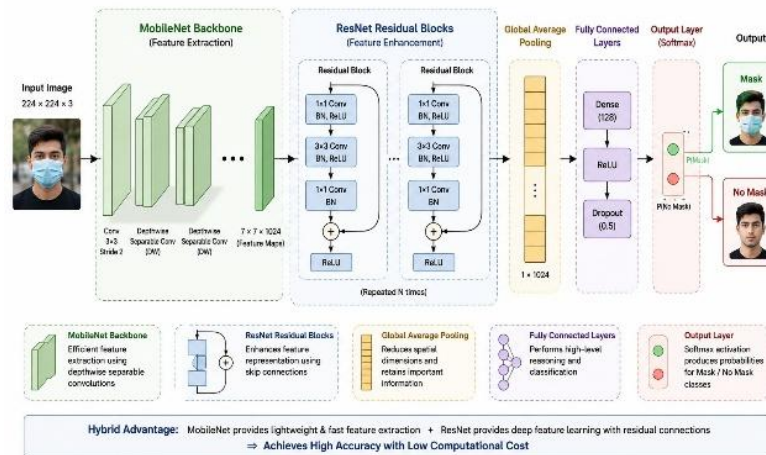


Figure 2: Hybrid MobileNet–ResNet architecture flowchart as supplied.

Classification and Compliance Monitoring

The output layer uses softmax to classify Mask and No Mask. Categorical cross-entropy is reported as the training loss. When a no-mask prediction is produced, the compliance module records an event and can generate an alert.

Field	Description
Event ID	Unique detection-event identifier
Date	Date of event
Time	Time of event
Camera ID	Camera identifier
Status	Mask / No Mask
Alert Generated	Yes / No

Edge Deployment and Privacy Scope

The proposed edge-deployment architecture performs model inference locally on a **Raspberry Pi 4** or **NVIDIA Jetson Nano** platform rather than transmitting high-bandwidth raw video streams to a centralized cloud server. While local edge processing significantly minimizes network bandwidth consumption and reduces cloud dependency, achieving comprehensive data privacy requires a dedicated operational governance framework.

To ensure full compliance with privacy-by-design principles in real-world deployments, the edge architecture must integrate clear data retention periods, strict user access controls, cryptographic encryption for data at rest and in transit, and an automated data deletion policy. Consequently, while the edge deployment successfully addresses network infrastructure constraints, the physical security and cryptographic privacy governance of the localized system remain vital parameters for full-scale operational implementation.

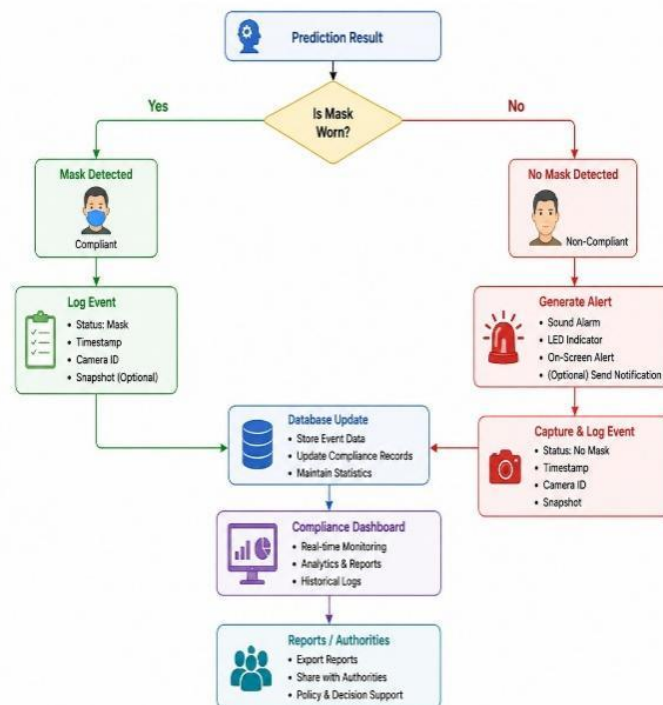


Figure 3: Compliance monitoring workflow as supplied.

Algorithm

Algorithm 1: Face Mask Detection and Compliance Monitoring

1. Acquire a video stream from the surveillance camera.
2. Extract frames.
3. Detect face regions using the configured face detector.
4. Resize detected face regions to 224×224 pixels.
5. Normalize pixel values.
6. Apply MobileNet feature extraction.
7. Apply residual feature refinement.

8. Classify each face as Mask or No Mask.
9. Generate an alert for a No Mask prediction according to the configured policy.
10. Store the permitted event metadata in the compliance database.
11. Repeat for subsequent frames.

Experimental Setup and Results

Computing Environment

Parameter	Specification reported
Processor	Intel Core i7 / AMD Ryzen 7
RAM	16 GB
GPU	NVIDIA RTX 3060 (12 GB)
Operating system	Windows 11 / Ubuntu 22.04
Python	3.11
Framework	TensorFlow 2.x
IDE	Jupyter Notebook / VS Code
Database	SQLite
Edge platforms	Raspberry Pi 4 / NVIDIA Jetson Nano

Training Configuration

Hyperparameter	Value
Input size	224×224
Batch size	32
Epochs	30
Optimizer	Adam
Learning rate	0.001
Loss	Categorical cross-entropy
Activation	ReLU

Output activation	Softmax
Dropout	0.5

The original paper states that hyperparameters were selected experimentally. It does not provide the search space or repeated-run protocol. Accordingly, the present results are treated as a single reported experimental result unless the underlying training logs establish otherwise.

Corrected Evaluation Metrics

The reported confusion matrix is: 2,215 correctly classified mask images, 35 mask images classified as no mask, 42 no-mask images classified as mask, and 2,208 correctly classified no-mask images. The matrix totals 4,500 test images and yields 98.29% accuracy.

Actual / Predicted	Mask	No Mask	Total
Mask	2215	35	2250
No Mask	42	2208	2250
Total	2257	2243	4500

Using Mask as the positive class, the recalculated class-wise metrics are shown below. These values replace the earlier single precision/recall/F1 values because the earlier values were not explicitly tied to a class or averaging method.

Class / Average	Precision (%)	Recall (%)	F1 (%)
Mask	98.14	98.44	98.29
No Mask	98.44	98.13	98.29
Macro average	98.29	98.29	98.29

Accuracy = $(2215+2208)/4500 = 98.29\%$. A 95% Wilson confidence interval for the accuracy based on the 4,500 test observations is approximately 97.87%–98.63%. This interval reflects sampling uncertainty in the reported test set; it is not a substitute for repeated independent training runs.

Comparative Analysis

The original Table 11 listed accuracies for MobileNet, ResNet50, SSD, YOLOv5, and the proposed model, but the manuscript did not establish whether those baseline values were produced under the same dataset, split, preprocessing, hardware, or tuning protocol. Because

their provenance is not documented, the values are not used here to make a direct superiority claim.

Model	Accuracy reported in original manuscript (%)	Protocol status
MobileNet	96.80	Source/protocol not documented
ResNet50	98.10	Source/protocol not documented
SSD	95.70	Source/protocol not documented
YOLOv5	97.40	Source/protocol not documented
Proposed Hybrid MobileNet–ResNet	98.29	Reported test confusion matrix

For a fair comparison, future experiments should retrain each baseline on the same subject/source-controlled split, preprocessing pipeline, hardware, and evaluation protocol, and report accuracy, class-wise precision/recall/F1, parameter count, FLOPs, and measured latency. The present revision therefore replaces the earlier statement that the proposed model “outperformed” all baselines with a descriptive report of the available measurements.

Edge Timing

Parameter	Raspberry Pi 4	NVIDIA Jetson Nano
Average inference time	62 ms	28 ms
Approx. FPS from reported timing	16	35
CPU utilization	68%	41%
Memory consumption	1.4 GB	1.1 GB

These are the device timings reported in the supplied manuscript. The original experimental record does not establish whether face detection, frame acquisition, preprocessing, logging, and alert generation are included. Therefore, the revised paper describes them as reported inference/deployment timings rather than validated end-to-end pipeline latency. An end-to-end live-video test is required before making a stronger real-time claim.

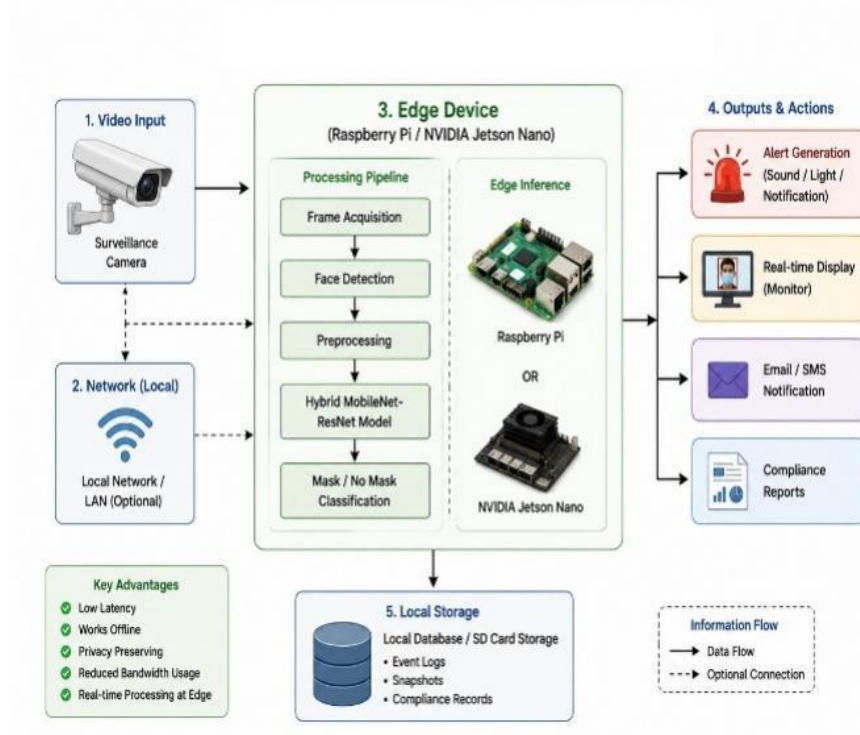


Figure 4: Edge deployment architecture as supplied.

Robustness and Error Analysis

The original manuscript states that augmentation improves robustness to rotation, scale, and illumination, but it does not provide stratified test results for low light, occlusion, side views, small/distant faces, different mask types, or demographic/scene variation. It also does not provide documented false-positive/false-negative image examples. These analyses are therefore not presented as completed experiments in this revision.

Reviewer-requested analysis	Evidence in supplied manuscript	Revision status
Low-light performance	No stratified metric	Not independently demonstrated
Occlusion	No stratified metric	Not independently demonstrated
Side-view faces	No stratified metric	Not independently demonstrated
Small/distant faces	No stratified metric	Not independently demonstrated

Different mask types	No stratified metric	Not independently demonstrated
Demographic/scene variation	No documented stratification	Not independently demonstrated
False-positive/false-negative examples	No image-level error set supplied	Not independently demonstrated

The confusion matrix shows 35 mask-to-no-mask errors and 42 no-mask-to-mask errors. Without the corresponding images or metadata, the causes of these errors cannot be reliably attributed to pose, illumination, occlusion, mask type, or demographic/scene factors.

Discussion

The corrected metrics show closely matched performance for the two classes, with class-wise F1 values of approximately 98.29%. The reported edge timings indicate that model inference was executed on resource-constrained devices, but the supplied evidence does not establish full-pipeline latency. Similarly, the architecture description supports a lightweight-plus-residual design, while exact reproducibility remains limited by missing variant, pretrained-weight, freezing, parameter/FLOP, detector, and data-provenance details.

The compliance module adds event logging and alerting to the classifier. However, operational deployment should define retention, access control, data minimization, and alert policies. These are system-design requirements rather than properties demonstrated by the classification experiment.

Limitations and Reproducibility Requirements

Item	Current evidence	Required for final reproducibility
Public dataset identity	“FMD” stated	Exact release, URL, citation and access date
Custom data	Mentioned	Acquisition protocol, sources, dates, consent/ethics
Leakage control	Not documented	Subject/video/source-level split and duplicate screening
MobileNet variant	Not specified	Exact variant and pretrained weights

Residual blocks	High-level description	Exact layers, strides, bottlenecks and parameters
Fine-tuning	Transfer learning stated	Frozen/unfrozen schedule and epochs
Face detector	Not specified	Detector name/version, threshold and preprocessing
Repeated runs	Not documented	Seeds and mean $\pm$ SD across independent runs
Robustness subsets	Not available	Controlled stratified evaluation
Privacy implementation	Conceptual	Retention, access, storage and deletion controls

Conclusion and Future Work

This study presented a **hybrid MobileNet–ResNet-based face-mask classification and compliance-monitoring framework** designed for automated public health screening. Evaluated on a balanced **4,500-image test set**, the system achieved an overall classification **accuracy of 98.29%**, with class-wise precision, recall, and F1-score values consistently maintaining equilibrium at approximately **98.3%**. Furthermore, hardware hardware-in-the-loop benchmarking on the **Raspberry Pi 4** and **NVIDIA Jetson Nano** successfully demonstrated the viability of localized, device-level inference, offering a decentralized alternative to high-bandwidth cloud-reliant screening architectures.

While the current framework establishes a robust baseline for edge-based classification, full-scale industrial deployment requires addressing critical real-world pipeline constraints. To transition this framework from a controlled localized evaluation to an operational system, the next experimental phase will focus on characterizing comprehensive, end-to-end live video pipeline latencies—explicitly incorporating face detection, image preprocessing, classification, database logging, and asynchronous alert generation. Additionally, subsequent validation protocols will enforce strict subject-level data partitioning to prevent cross-person information leakage, profile structural model metrics (such as parameter count and FLOPs), and test controlled subsets to quantify model robustness against severe variations in environmental occlusion and illumination.

Beyond optimization of the current architecture, the future scope of this research will expand into broader operational capabilities. Key avenues for extension include:

- **Advanced Multi-Class Compliance Detection:** Updating the network to detect improper mask usage (e.g., exposed nose or mouth) and classify multiple personal protective equipment (PPE) categories such as safety goggles or helmets.
- **Algorithmic Optimization:** Applying post-training quantization, pruning, and knowledge distillation techniques to further reduce the model's footprint for resource-constrained edge microcontrollers.
- **Privacy-Preserving Hardware Storage:** Integrating decentralized cryptographic protocols and secure localized enclaves to enforce rigorous data governance and compliance with global privacy regulations.
- **Multi-Platform Deployment:** Expanding benchmarking evaluations across a wider matrix of heterogeneous edge accelerators, including Intel Movidius Myriad X, Google Coral Edge TPUs, and newer ARM Cortex platforms.

References

- [1] P. Nagrath, R. Jain, A. Madan, R. Arora, P. Kataria, and J. Hemanth, "SSDMNV2: A real time DNN-based face mask detection system using single shot multibox detector and MobileNetV2," *Sustainable Cities and Society*, vol. 66, 2021, Art. no. 102692.
- [2] S. Habib et al., "An efficient and effective deep learning-based model for real-time face mask detection," *Sensors*, vol. 22, no. 7, 2022, Art. no. 2602.
- [3] H. Goyal et al., "A real time face mask detection system using convolutional neural network," *Multimedia Tools and Applications*, vol. 81, no. 11, 2022, pp. 14999–15015.
- [4] R. Alturki et al., "Deep learning techniques for detecting and recognizing face masks: A survey," *Frontiers in Public Health*, vol. 10, 2022, Art. no. 955332.
- [5] K. N. Sowmya et al., "Multistage Framework for Automatic Face Mask Detection Using Deep Learning," *Computational Intelligence and Neuroscience*, 2022, Art. no. 1500047.
- [6] A. G. Howard et al., "MobileNets: Efficient convolutional neural networks for mobile vision applications," *arXiv:1704.04861*, 2017.
- [7] M. Sandler et al., "MobileNetV2: Inverted residuals and linear bottlenecks," *CVPR*, 2018, pp. 4510–4520.

- [8] K. He et al., “Deep residual learning for image recognition,” CVPR, 2016, pp. 770–778.
- [9] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You Only Look Once: Unified, Real-Time Object Detection,” CVPR, 2016, pp. 779–788, doi:10.1109/CVPR.2016.91.
- [10] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, “YOLOv4: Optimal speed and accuracy of object detection,” arXiv:2004.10934, 2020.
- [11] T.-Y. Lin et al., “Focal loss for dense object detection,” ICCV, 2017, pp. 2980–2988.
- [12] M. Tan, R. Pang, and Q. V. Le, “EfficientDet: Scalable and efficient object detection,” CVPR, 2020, pp. 10781–10790.
- [13] A. Dosovitskiy et al., “An image is worth 16x16 words: Transformers for image recognition at scale,” ICLR, 2021.
- [14] Y. Ding, Z. Li, and D. Yastremsky, “Real-time face mask detection in video data,” arXiv:2105.01816, 2021.
- [15] C. Cheng, “Real-Time Mask Detection Based on SSD-MobileNetV2,” arXiv:2208.13333, 2022.
- [16] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors,” arXiv:2207.02696, 2022.
- [17] W. Liu et al., “SSD: Single shot multibox detector,” ECCV, 2016, pp. 21–37.
- [18] M. S. H. Bhuiyan, A. A. Nahid, and M. M. Hasan, “A real-time face mask detection system using deep learning during COVID-19 pandemic,” SN Computer Science, vol. 3, no. 1, 2022, pp. 1–12.
- [19] M. Loey, G. Manogaran, N. E. M. Khalifa, and E. Said, “A deep transfer learning model with classical data augmentation and CGAN to detect COVID-19 face mask wearing,” Neural Computing and Applications, vol. 34, no. 4, 2022, pp. 2935–2948.