This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051844, IEEE Internet of Things Journal

Real-time Mask Identification for COVID-19: An Edge Computing-based Deep Learning Framework

Xiangjie Kong, *Senior Member, IEEE*, Kailai Wang, Shupeng Wang, Xiaojie Wang, Xin Jiang, Yi Guo, Guojiang Shen, Xin Chen, and Qichao Ni

***Abstract*—During the outbreak of COVID-19, while bringing various serious threats to the world, it reminds us that we need take precautions to control the transmission of the virus. The rise of Internet of Medical Things (IoMT) has made related data collection and processing, including healthcare monitoring systems, more convenient on the one hand, and requirements of public health prevention are also changing and more challengeable on the other hand. One of the most effective non-pharmaceutical medical intervention measures is mask-wearing. Therefore, there is an urgent need for an automatic real-time mask detection method to help prevent the public epidemic. In this paper, we put forward an edge computing-based mask identification framework (ECMask) to help public health precautions, which can ensure real-time performance on the low-power camera devices of buses. Our ECMask consists of three main stages: video restoration, face detection, and mask identification. The related models are trained and evaluated on our Bus Drive Monitoring Dataset and public dataset. We construct extensive experiments to validate the good performance based on real video data, in consideration of detection accuracy and execution time efficiency of the whole video analysis, which have valuable application in COVID-19 prevention.**

***Index Terms*—COVID-19, Public Health Prevention, Internet of Things, Edge Computing, Deep Learning, Mask Identification.**

I. Introduction

THE Coronavirus Disease 2019 (COVID-19) caused by the Severe Acute Respiratory Syndrome coronavirus 2 (SARS-CoV-2), has given rise to a global epidemic [1]. According to the report of World Health Organization (WHO) on the 3rd July, 2020, there were over more than 10 million confirmed cased and 500 thousand deaths, which are alarming numbers [2]. Besides, the growing number still reminds us that we need to take preventive measures. Based on previous studies and measures taken in many regions [3], mask-wearing is proved to be an effective non-pharmaceutical intervention measure, which is non-invasive, convenient, and cheap to lower the infection and spread of COVID-19 [4]. Especially in China, a populous country with Mega cities, the interaction and contact between people are frequent in the process of daily travel and work. If there is no timely prevention, the possibility of infection is higher. Therefore, it is essential to develop a method to automatically detect mask-wearing, which can prevent public epidemic.

During COVID-19, as one of the public transportation, bus is not only a common way to travel, but also a public place where people gather. Due to the crowd, some medical prevention measures have been taken, including sterilization on time and social distance of passengers. Besides, bus drivers are more likely to have certain interactions with passengers. Therefore, it is necessary for bus drivers to detect mask-wearing for the prevention and control of COVID-19.

For the past few years, as the technology of the Internet of Things (IoT) developing rapidly [5], the data sensing, collection and analysis become efficient [6]. IoT will bring revolutionary changes to work and life [7], [8]. Online interactive classrooms have difficulties with large bandwidth, long link transmission, and wide coverage. Alibaba Cloud uses IoT and mobile edge computing (MEC) to provide services on the edge of the network closer to the terminal, which significantly improves the overall low-latency and strong interactive experience in interactive classroom business scenarios. For some campus monitoring scenarios, Huawei uses IoT to solve the problem that WiFi/fiber cannot be used in the factory environment, which prevents data from leaving the campus. In the medical field, the application of IoT is typically named IoMT, which makes a significant contribution to healthcare systems from medical monitoring to smart sensors [9], [10]. Meanwhile, deep learning and artificial intelligence applications with high computational overhead could be implemented and applied in the real industry environment [11], [12]. In particular, under the low-cost and low-power processing capabilities of camera sensing devices, video analysis is not processed directly on the device. Instead, the video data can be transmitted to the cloud service platform for analysis through cloud computing [13], which shortens a degree of analysis time.

However, the actual application has higher requirements for real-time performance, especially in healthcare systems. For example, time-effective alerting and notification to patients can ensure preventive care and medical management [14]. Facing

This work was partially supported by the National Natural Science Foundation of China (62072409, 62073295, 61931019, 62001073), Zhejiang Provincial Natural Science Foundation (LR21F020003), and Fundamental Research Funds for the Provincial Universities of Zhejiang (RF-B2020001). *(Corresponding authors: Shupeng Wang; Xiaojie Wang.)*

X. Kong, and G. Shen are with the College of Computer Science and Technology, Zhejiang University of Technology, Hangzhou 310023, China (e-mail: xjkong@ieee.org; gjshen1975@zjut.edu.cn).

K. Wang, X. Chen, and Q. Ni are with the School of Software, Dalian University of Technology, Dalian 116620, China (e-mail: kailai.w@outlook.com; chenxin20@mail.dlut.edu.cn; niqichao666@gmail.com).

S. Wang is with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China (email: wangshupeng@iie.ac.cn).

X. Wang is with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China (email: xiaojie.kara.wang@ieee.org).

X. Jiang, and Y. Guo are with the Second Clinical Medical College (Shenzhen People's Hospital), Jinan University, Guangzhou 510632, China (e-mail: jiangxinsz@163.com; xuanyi_guo@163.com).

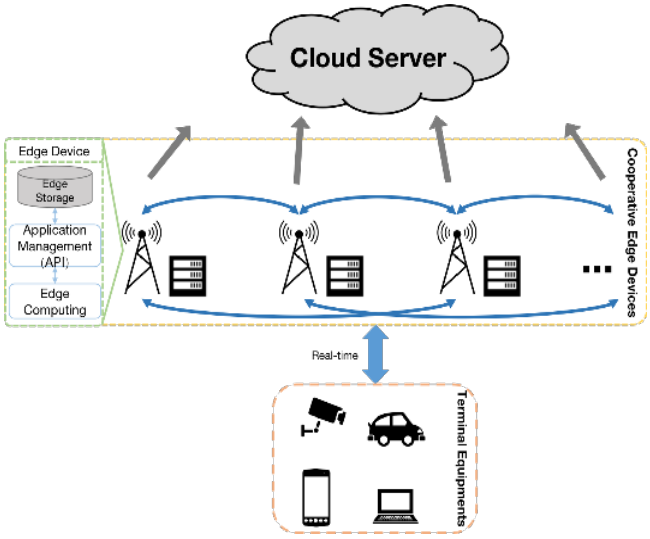


Fig. 1: The basic paradigm of cooperative edge computing under IoT.

a huge amounts of data generated by edge devices, video analysis through cloud computing requires a large amount of bandwidth, which results in the latency for detection. Therefore, edge computing is proposed to execute deep learning with high computational overhead and reduce the network latency by data transmission [15]. With the rise of edge computing, it has become possible to improve efficiency of the various tasks [16], [17]. Besides, due to the limitation of single edge device, cooperation in edge computing is considered to take full advantage of storage computing power [18]. As shown in Fig. 1, the whole process of cooperative edge computing mainly consists of terminal equipment and cooperative edge devices, which perform the calculation of required algorithms. Near the data source, that is, the edge of the network, the shared resources are deployed to perform computation and inference on the edge device [19].

The latest advances in computer vision provide opportunities for practical healthcare applications. Among them, deep neural networks (DNN), particularly convolutional neural networks (CNN), has wide application in various fields, such as object detection, image segmentation, and image classification. Benefiting from them, face detection, as one of the object detection, have been made much progress from two aspects of face detection accuracy and speed [20]–[24]. However, in order to expand coverage and lower costs, camera sensing devices cannot guarantee the high-quality images leading to some problems like noise, blur, and shake for industrial applications. Therefore, it is necessary to improve the video super-resolution, which is video restoration [25]. Different from image restoration, there exists a temporal correlation among neighboring frames of video. After video restoration, the low-quality video is improved, and subsequent detection and recognition are more reliable and accurate.

In this paper, we describe our efforts to propose a framework of the mask identification on the facial image (ECMask) to identify mask-wearing in real-time during COVID-19 based on edge computing. First, based on real video data of bus driver monitoring, we utilize blur detection method and video restoration to improve detection accuracy for the video data with blur problems from low-cost cameras, which could be regarded as part of the video preprocessing. Then, the face detector is trained and verified by the public datasets and our Bus Drive Monitoring Dataset. After obtaining and cropping face areas of Bus Drive Monitoring Dataset, mask identification model can be trained, which is also the image classification task of faces (a binary classification), that is, those who wear a mask (wearing correctly, Masked) and those who are not (including wearing incorrectly, Non-Masked). Finally, in order to achieve edge computing of the above deep learning models on a low-cost device, the Intel Neural Compute Stick 2 (NCS) is added in edge devices and used to accelerate the models through its high performance operation.

The main contributions can be summarized as follows:

- We present a framework, ECMask, to detect mask-wearing with deep learning, including video restoration, face detection, and mask identification, which is able to prevent infection of COVID-19 and provide public health precaution reminders in real-time.
- We apply edge computing in the video analysis process to enhance the effectiveness of detection and identification at the edge devices through Intel Neural Compute Stick 2 (NCS) .
- We have constructed comprehensive experiments to verify the excellent efficiency of our ECMask. Based on the real Bus Drive Monitoring Dataset, the outcomes indicate that video restoration can heighten detection accuracy and edge computing-based method has excellent performance in inference time efficiency of the whole video analysis.

The rest of this article is structured as follows. In Section 2, we briefly introduce related work. The third section mainly explains the details of our proposed model and framework (ECMask). Data description and analysis of experiment result will be given in Section 4. Finally, we will conclude and provide further discussion in Section 5.

II. Related Work

In this section, we illustrate the existing works closely related to the content of this paper.

A. IoMT with Edge Computing

As an emerging field of research, edge computing plays a significant role in IoMT, due to its adavantages including faster processing data, reducing the budget, offloading network traffic, improving application efficiency, and security and privacy protection. Pace et.al. [26] proposed a IoMT system architecture, BodyEdge, which was designed to support different healthcare scenarios, including workers in a factory, athletes, and patients in a hospital. Focus on sustainability and energy utilization, Han et.al. [27] proposed a clustering model for medical applications (CMMA) for cluster head selection, which considered additional factors specially for IoMT network, such as capacity and queue of the medical devices. Dong et.al. constructs an edge computing based

healthcare system in IoMT, which applied Nath bargaining solution in intra-WBANs and developed a non-cooperative game based decentralized method to minimize the costs in beyond-WBANs. Besides, deep learning model can also be combined with edge computing in IoMT. In [28], authors proposed a effective training scheme for the deep learning neural network (ETS-DNN) in edge computing enabled IoMT systems, which incorporated a Hybrid Modified Water Wave Optimization technique to improve the healthcare system efficiency.

B. Video Restoration

As deep learning developing gradually, the realization of the super-resolution of images and videos becomes better. On the one hand, as the pioneering work, SRCNN [29] designed a simple architecture with CNN, which achieved the optimal efficiency in terms of image super-resolution. From here, more and more related work about image super-resolution was inspired [30]. On the other hand, video super-resolution contains more temporal information that is different from image super-resolution and leads to the challenge of temporal alignment and fusion. Tian et al. [31] proposed TDAN, which introduce deformable convolutions, instead of computing optical flow, to address the problem about temporal alignment of the frames of video. Xue et al. [32] proposed TOFlow, which estimated the optical flow field and used a flow image as a motion representation. For video deblurring, Su et al. [33] introduces a deep learning solution to fuse neighboring frames, which lessen the requirements for accurate temporal alignment. In [25], EDVR was proposed to solve the problem of video super-resolution and deblurring, which utilized pyramid structure with deformable convolution and attention mechanism to improve efficiency. In our ECMask, we focus on the effect of video restoration when needed. Therefore, considering the computational overhead of the whole process, we try to employ the EDVR to improve the subsequent detection and identification, if the raw video data has problems, such as noise and blur.

C. Face Detection

Face detectors based on CNN have been extensively studied in recent years. Zhang et al. [24] developed a multitask cascaded architecture using CNN to extract the locations of the face and landmark from coarse to fine. Following that Faster R-CNN [34] proposed the concept of the anchor, it was widely used in object detectors to ensure accuracy and speed up at the same time, including face detectors. Besides, the pyramid network structure can improve the small object detection. In [35], authors put forward a novel face detector, named Single Shot Scale-Invariant (S^3FD), that uses different scale anchors at different convolutional layers and lowers threshold to enhance the recall rate of tiny faces outline detection. Moreover, the contextual information is important in face detection. For example, Najibi et al. [36] introduced Single Stage Headless (SSH) face detector, which integrates context layers into the detection modules to improve the mean average precision. Meanwhile, to apply detectors to actual system, the real-time is an important consideration. Redmon et al. [37] proposed YOLO, which is an anchor-based detector and can select feature maps directly to achieve real-time performance. In [38], anchor-based FaceBoxes designed the Rapidly Digested Convolutional Layers (RDCL) for acceleration on the CPU devices to ensure real-time. Inspired by FaceBoxes, our ECMask adopts C.ReLu [39] and inception module [40] to achieve real-time of face detection.

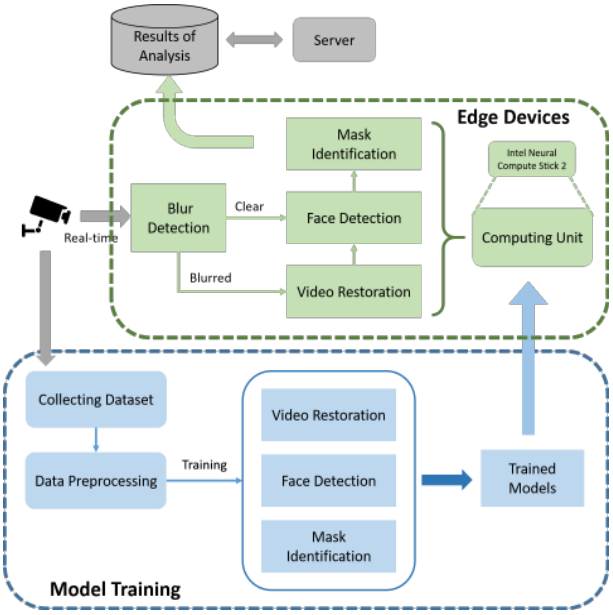


Fig. 2: The architecture of our ECMask.

III. Design of Framework

This details of the procedures and modules will be illustrated in our proposed ECMask shown in Fig. 2.

A. Overview

ECMask is shown in Fig. 2 and mainly includes model training and real-time video analysis at the edge nodes. According to the goal of video analysis, ECMask needs to train three models including video restoration, face detection, and mask identification after collecting and preprocessing our Bus Drive Monitoring Dataset and the public dateset. These trained models will be deployed on the edge devices, which are also optimized to maximize performance. Therefore, in the framework, monitoring data will be transmitted to high-performance equipment for auxiliary model training, and then transmitted to edge equipment for real-time inspection. To be specific, in the part of real-time video analysis, we will execute the video blur detection with Laplacian operator to determine whether video restoration is needed, which can reduce the huge computational overhead brought by video restoration. Then, the real-time video data is inputted on the subsequent models to obtain the results of detection and identification, which are transmitted back to show to management.

B. Model Inference with Cooperative Edge Computing

The high computing power is required in the most effective deep learning algorithms. In traditional research, there may

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/JIOT.2021.3051844, IEEE Internet of Things Journal

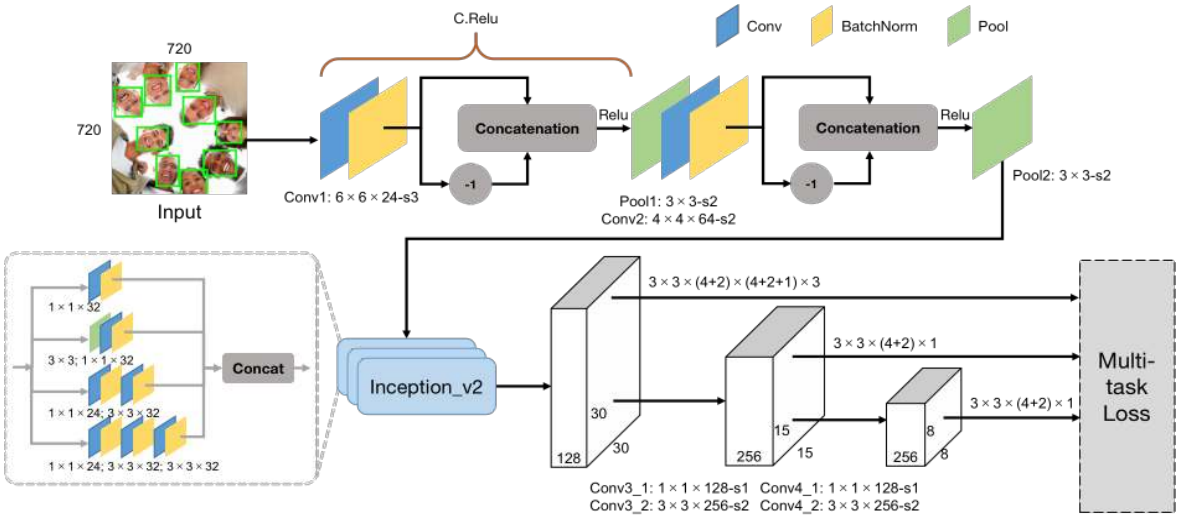


Fig. 3: The architecture and details of our face detector.

be no concern about power consumption and computational complexity. However, in actual application and deployment, it is a challenging task, which has led to the demand for low-cost but powerful inference processors for deep learning. Therefore, edge computing has been promoted and become popular due to its ability of low network latency, and privacy protection with low-cost. In our framework, we apply Intel's Neural Compute Stick 2 (NCS) in the edge computing device, which is a small USB 3.0 Type-A deep learning device and can be used in computer vision methods at the edge and IoT. Compared with CPU, its Intel Movidius Myriad Vision Processing Unit (VPU) is a chip, which is dedicated to computer vision. It has low power consumption to process images and videos, thus can be regarded as a micro GPU to accelerate calculation in neural networks. Besides, the OpenVINO toolkit provides model optimizer, inference engine, and distributed computing which will be used with multiple NCSs to implement cooperative edge computing. The trained deep learning model is optimized and stored in the NCSs with the inference engine. In the end, the input video data is obtained from the connected camera devices.

C. Video Restoration

The first stage of ECMask is to restore and deblur the raw video data to enhance the accuracy of subsequent detection. Therefore, we employ Video Restoration framework with Enhanced Deformable convolutions (EDVR) as the first stage of ECMask due to its good performance in video restoration. The architecture of EDVR mainly includes four modules, such as PreDeblur module, PCD alignment module, TSA fusion module, and Reconstruction module.

- PreDeblur module is similar to encoder-decoder network, which is a pyramid structure and consists of down-sampling layers using convolution layers of 2 stride and upsampling layers. Furthermore, the feature map is extracted through the residual block at each layer of the pyramid. In this way, the frames of video can be deblurred, which is the preprocessing part before the alignment module and fusion module.
- In PCD alignment module, to avoid the disadvantages of optical-flow based methods, like higher computation overhead, the deformable convolution is employed to align features of each neighboring frame to its reference frame. For the three-level pyramid structure, the cascade deformable alignment method is used to refine the coarsely aligned feature, that is, the method from coarse to fine is designed to enhance the pixel alignment accuracy.
- In TSA fusion module, fusing feature information of the aligned neighbouring frames is its main goal. For video restoration task, some unavoidable reasons like object moving and camera shaking could produce different degrees of blur of frames, which lead to the different contributions of neighboring frames to the restored reference frames. Therefore, the attention mechanism is used to assign different pixel-level aggregation weights in the temporal and spatial dimensions of feature maps. To be specific, in temporal attention mechanism, the features of each frame, F_t, is embedded in lower dimension space, then their similarity is computed to indicate the attention, which is a temporal attention map. The similarity of neighboring frames is calculated as follows:

$$Sim(F_{t+i}, F_t) = \sigma(\Phi(F_{t+i})^T \cdot \Phi(F_t)) \tag{1}$$

where σ is an activation function (like sigmoid), and $\Phi(\cdot)$ is the embedding of features obtained by a simple convolutional operation. Therefore, the attention-modulated feature is written as:

$$\widetilde{F}_{t+i} = F_{t+i} \odot Sim(F_{t+i},\ F_t) \tag{2}$$

where $\odot$ denotes the element-wise multiplication. These attention-modulated features are fused in the fusion convolutional layer. Similarly, the pyramid structure is employed to increase attention receptive field for spatial attention map.

- Reconstruction module is composed of several residual blocks. Besides, Charbonnier penalty function is used as the loss function:

$$L = \sqrt{\left\| \hat{O}_t - O_t \right\|^2 + \varepsilon^2} \tag{3}$$

where ε is set to 1×10^{-3}.

D. Face Detection

Considering the demand for real-time, the inference process of our face detection is carried out at the low-power edge device. Inspired by end-to-end FaceBoxes [38], the architecture of the face detector is shown in Fig. 3, which is an anchor-based face detector. The network structure of the face detector can be divided into two stages that make the detector accurate and efficient on the edge devices with shrinking the spatial size of input, enriching the receptive fields.

In practical application, the frames of video as input images have high resolution. Therefore, in order to reduce the computational cost, we need to accelerate downsampling as soon as possible, that is, most of the reduction in the width and height of the feature map could be completed due to suitable stride size and kernel size at the first stage. The stride value of convolutional layer and pooling layer are shown in Fig. 3. If the pixel size of the input image is 720×720, the total stride size of the first stage is 24 ($3 \times 2 \times 2 \times 2$), which means that it achieves 24 times downsampling. Besides, the C.ReLU activation function [39] is utilized to help to further reduce computing overhead. C.ReLU is designed from the statistical observation that there is a negative correlation in the lower layers. Based on this, the number of output channels is concatenated with its negation before the ReLU activation function.

In the second stage, three Inception modules [40] are used to enrich the receptive fields, which are able to detect various scales of faces. The Inception module can factorize convolution into smaller convolutions, which makes the whole network wider, but with the smaller number of parameters and computation. In this way, it provides multiple branches with different kernels, that are different receptive fields. Then, through the several convolutional layers, the last two downsampling is completed. Similar to SSD, multi-scale feature maps are generated for detection. Besides, considering that most of the face box is square, this part uses default boxes with 1:1 aspect ratio for prediction, that is anchor-based methods. We also employ the anchor densification strategy from [38]. As the definition of the tiling density of anchor, the equation can be written as follows:

$$A_{density} = \frac{A_{scale}}{A_{interval}} \tag{4}$$

where, A_{scale} is the scale of default anchor boxes, and $A_{interval}$ is the interval of the anchor. In our face detector, $A_{interval}$ also denotes the multiple of downsampling (i.e. 24, 24, 24, 48, and 96 for default anchors). We set the scale of default anchor to 24, 48, and 96 pixels for the third Inception modules, 192 pixels for Conv3_2, and 384 pixels for Conv4_2, respectively. Therefore, we obtain $A_{density}$ of each anchor (i.e. 1, 2, 4, 4, and 4). To have a better ability of the detection of small size faces, the anchor densification strategy is used that keeps the value of $A_{density}$ constant (i.e. 4), that is, increase the number of anchors at a center point by translating anchors.

In the end, we adopt loss function which is the same as Faster R-CNN [34], that is, binary cross-entropy for classification and the smooth L1 loss for regression are employed as loss function:

$$\begin{cases} \mathcal{L}\left(P_{face}, Q\right) = \frac{1}{N}\left(\sum\limits_{p_i \in P} \mathcal{L}_{cls}(p_i) + \alpha \sum\limits_{q_i \in Q} \mathcal{L}_{reg}(q_i)\right), \\ \mathcal{L}_{cls}(p_i) = -[p_i^* \log(p_i) + (1 - p_i^*) \log(1 - p_i)], \\ \mathcal{L}_{reg}(q_i) = smooth_{L1}(q_i - q_i^*), \end{cases} \tag{5}$$

where N is the number of positive samples in one batch, and α is the balancing parameter. P_{face} is the set of all predicted probability of anchor as the face object (p_i), and p_i^* is an indicator function of p_i, that is, the positive anchor is $p_i^* = 1$; otherwise $p_i^* = 0$. Q is the set of vectors representing of the predicted bounding box ($q_i = [x_i, y_i, w_i, h_i]$), and q_i^* denotes the ground-truth box related with a positive anchor. Besides, the probability in $\mathcal{L}_{cls}$ is computed by softmax loss function.

E. Mask Identification

After face detection, our goal is to identify the condition of mask-wearing based on the cropped faces (i.e. Masked and Non-Masked). In essence, it is also an image binary classification task, and it can be regarded as a function, $f : I \mapsto P_{mask}$, where I is the face image as input , and P_{mask} is the output probability of mask-wearing that is used to obtain the classification result. In our Bus Drive Monitoring Dataset, we label incorrectly mask-wearing as Non-Masked, which may only be slightly different from Masked. Therefore, the network puts forward some requirements for the ability of extracting feature.

As the image classification performance of CNN, Mobilenet-V2 is adopted to identify the condition of mask-wearing, which not only has high accuracy, but also is a lightweight image classification network. Therefore, Mobilenet-V2 is well suited for edge devices. Compared with traditional CNN, Mobilenet-V2 can decrease the amount of computation and the number of model parameters by replacing the standard convolutional layer with the depthwise separable convolution, which can ensure the high image classification accuracy. The depthwith separable convolution includes two parts that are depthwise convolution and pointwise convolution, which are used to filter and combine, respectively. Besides, the inverted residuals are used to enhance the ability of propagating the gradient to multiplier layers and memory efficiency, which increase and then decrease the number of channels. In the bottleneck inverted residual block (BIR), the linear bottleneck is used instead of ReLU after the second pointwise convolution to retain feature diversity. The network structure of Mobilenet-V2 is shown in TABLE I, where n is the number of repetitions and k is expansion ratio.

Similarly, for the classification task, the softmax function is utilized to compute the confidence of classes (i.e. Masked

Fig. 4: Visualized results of video restoration (VR) compared with ground truth (GT) on Bus Drive Monitoring Dataset including 1 Non-Mask frame and 3 Masked frames. (Zoom in for suitable view, and faces portions are pixelated for keep anonymity.)

TABLE I: The network structure of Mobilenet-V2.

Input	Operator	Output	n	k
$224^2 \times 1$	Conv2d	$112^2 \times 32$	1	-
$112^2 \times 32$	BIR	$112^2 \times 16$	1	1
$112^2 \times 16$	BIR	$56^2 \times 24$	2	6
$56^2 \times 24$	BIR	$28^2 \times 32$	3	6
$28^2 \times 32$	BIR	$14^2 \times 64$	4	6
$14^2 \times 64$	BIR	$14^2 \times 96$	3	6
$14^2 \times 96$	BIR	$7^2 \times 160$	3	6
$7^2 \times 160$	BIR	$7^2 \times 320$	1	6
$7^2 \times 320$	Conv2d 1×1	$7^2 \times 1280$	1	-
$7^2 \times 1280$	AvgPool 7×7	$1^2 \times 1280$	1	-
$1^2 \times 1280$	Conv2d 1×1	1	-	-

and Non-Masked), and the cross-entropy is till utilized as the object function. Considering the problem of overfitting, we add an L2 regularization term to the object function, which is written as follows:

$$\begin{cases} \mathcal{L}(M) = -\dfrac{1}{|M|}(\mathcal{L}_{cls}(M) + \mathcal{L}_{reg}) \\ \mathcal{L}_{cls}(M) = \displaystyle\sum_{m_i \in M} [m_i^* \log(m_i) + (1 - m_i^*) \log(1 - m_i)] \\ \mathcal{L}_{reg} = \lambda W \cdot W^T / 2 \end{cases} \tag{6}$$

where M is the set of predicted probabilities of samples (cropped faces), and m_i^* is the indicator function of m_i, that is, $m_i^* = 1$ which denotes face image i that belongs to label Masked, otherwise $m_i^* = 0$. For the rest term of $\mathcal{L}$, L2 regularization term ($\mathcal{L}_{reg}$), W is the learned parameters of network, and $\lambda > 0$ is the regularization coefficient.

IV. Experiments

In this section, we first describe the collected Bus Driver Monitoring Dataset and training details. Then, we show the performance and evaluation of three analysis tasks in our ECMask, including video restoration, face detection, and mask identification. Finally, we present the inference time efficiency of ECMask at the edge nodes.

A. Dataset Description and Training Details

1) Dataset Description: For the pubic health prevention and the mask identification during COVID-19, we collect the real bus driver monitoring video clips (each with 104 consecutive frames) as Bus Drive Monitoring Dataset, which are standard quality (720×576), provided by Panda Bus Company. The dataset contains 642 video clips, which are labeled as Masked and Non-Masked. After labeling, the dataset is divided into 488 clips with label Masked and 154 clips with label Non-Masked. Bus Drive Monitoring Dataset consists of 80% training clips, 10% validation clips, and 10% testing clips (i.e. 516, 63, and 63, respectively). For the training of video restoration, we execute specific processing, which is downsampling to 144×115 and Gaussian Blur with 7×7 kernel size and 5 standard deviation, to complete the training dataset.

2) Training Details:

Video Restoration. The training of video restoration is the same as EDVR [25], except that our Bus Drive Monitoring Dataset is added for training.

Face Detection. The training of face detection utilizes 12880 images of the WIDER FACE training set [1]. The training data is augmented with random $90°$ rotations and horizontal flips. The model is trained with SGD, which sets 10^{-3} initial learning rate, 0.9 momentum, and 5×10^{-4} weight decay. Besides, the learning rate is adjusted with a drop factor of 0.9 every 30 epochs. During training, the threshold of Jaccard overlap of matching anchors to faces is set to 0.35.

Mask Identification. After face detection on Bus Drive Monitoring Dataset, we obtain the position of the faces, which are cropped as the inputs of the model of mask identification. Similarly, we use SGD to train the model with 0.045 initial learning rate and 0.9 momentum, and the learning rate is

[1]The dataset is available at http://shuoyang1213.me/WIDERFACE/

Fig. 5: Visualized results of our face detector with video restoration (VR) and without VR on Bus Drive Monitoring Dataset including 1 Non-Mask frame and 3 Masked frames. (Faces portions are pixelated for keep anonymity.)

Fig. 6: Visualized results of mask identification with video restoration (VR) and without VR on Bus Drive Monitoring Dataset including 1 Non-Mask frame and 3 Masked frames. (Faces portions are pixelated for keep anonymity.)

adjusted with cosine decay. Besides, λ, the factor of L2 regularization, is 4×10^5.

Our models are implemented based on the PyTorch framework, and trained using NVIDIA GPU with CUDA and cuDNN enabled in the edge server.

B. Experiment Results

1) Video Restoration: For the blur detection, we make sure that the number of clips that need to be restored is few. Therefore, we set the threshold to 115 with the variance of the Laplacian operator based on the statistic analysis on Bus Drive Monitoring Dataset. There are a small number of clips that require video restoration, accounting for 3% (i.e. 13 Masked clips, and 6 Non-Masked clips). We evaluate the performance of video restoration on two quality metrics.

- Peak Signal-to-Noise Ratio (PSNR). PSNR is extensively applied to evaluate the quality of an image after processing compared with its original image. The higher value of PSNR means they have smaller difference. The formula is as follows:

$$PSNR(X,Y) = 10\log_{10}\left(\frac{(2^n-1)^2}{MSE(X,Y)}\right) \quad (7)$$

where MSE means mean square error operation, and n denotes the number of bits per pixel (generally $n = 8$).

- Structural SIMilarity index (SSIM). As a common image standard evaluation index, SSIM can evaluate the similarity between two images from different perspectives including brightness, contrast, and structure. The value range of SSIM is $[0, 1]$, and the closer to 1, the higher similarity. Its equation can be defined as:

$$SSIM(X,Y) = \frac{(2\mu_X\mu_Y + c_1)(2\sigma_{XY} + c_2)}{(\mu_X^2 + \mu_Y^2 + c_1)(\sigma_X^2 + \sigma_Y^2 + c_2)} \quad (8)$$

where μ and σ are the mean and covariance operations. c_1 and c_2 are positive constants to avoid the denominator being 0.

TABLE II demonstrates the calculation results of the quantitative metrics compared with ground-truth clips. Experiments present that the effect of video restoration is verified, and the

TABLE II: Quantitative results of video restoration.

Approach	PSNR	SSIM
Downsample	28.79	0.8730
Gaussian Blur	30.55	0.9037
Video Restoration	**35.56**	**0.9495**

TABLE III: The accuracy results of our face detector compared with different methods.

Approach	Dataset		
	FDDB	WIDER FACE (easy)	WIDER FACE (medium)
Faceness	90.3%	71.6%	60.4%
CMS-RCNN	90.6%	89.2%	87.0%
LDCF+	93.3%	79.7%	77.2%
MTCNN	95.0%	85.1%	82.0%
ScaleFace	**96.0%**	86.7%	86.6%
Ours	**95.9%**	**89.8%**	**87.1%**

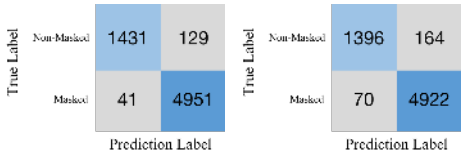
values of PSNR and SSIM after restoration are higher than those after downsampling and Gaussian Blur. Moreover, the visualized results are shown in Fig. 4, including 3 Masked clips and 1 Non-Mased clip whose scores of the variance of the Laplacian operator are less than 115, which present that our video restoration can enhance face details.

2) Face Detection: For face detection, our fade detector is evaluated against other methods [20]–[24] on the popular face detection benchmarks, including the FDDB dataset and WIDER FACE dataset (easy and medium subsets) in TABLE III. Our face detector almost outperforms others on two subsets of WIDER FACE. ScaleFace has similar performance to ours on the FDDB on the FDDB. Furthermore, we design the simple ablation experiment to illustrate the effect of video restoration based on our Bus Driver Monitoring Dataset, that is, each frame of clips is detected after video restoration or not. As shown in TABLE IV, the performance evaluation results with and without video restoration are compared. These results prove that video restoration can improve the accuracy of face detection, that is, video restoration effectively increases accuracy by 1.23%. Our face detector reaches a high accuracy of 97.98%, which can meet the needs of face detection accuracy for most industrial environments. The examples of visualized results are shown in Fig. 5.

3) Mask Identification: After face detection, we perform mask identification based on the previous results of faces obtained in each frame of clips. Similarly, we compute the confusion matrixes to demonstrate the performance of mask identification including those with and without video restoration as shown in Fig. 7. Our mask identification method can correctly classify 6382 frames of 6552 frames (the accuracy is

TABLE IV: The accuracy results of our face detector with video restoration (VR) and without VR.

Approach	Bus Drive Monitoring Dataset	
VR	×	✓
Accuracy	96.75%	97.98%



(a) With VR (b) Without VR

Fig. 7: Confusion Matrix results of mask identification (a) with video restoration (VR), and (b) without VR on Bus Drive Monitoring Dataset.

97.41%), which is better than those without video restoration. In addition, we observed and analyzed the failed cases of mask identification, and found several reasons for identification errors, that is, low clip quality, the arm covering a large area of the face, and incomplete face (looking back). The examples of mask identification results are given in Fig. 6.

C. Inference Time Efficiency

In order to evaluate runtime efficiency, we calculate the average inference time of face detection and mask identification, and use their sum as the whole inference time. In our experiments, we use the NCS and Raspberry Pi 4 as the low-cost edge device, which is compared with the higher prices hardware, CPU (Intel Xeon E5-2690 v4@2.60). In addition, the distributed computing method is adopted with multiple NCSs to further reduce the inference time and achieve real-time, which can be regarded as cooperation between the edge devices. Fig. 8 shows the results of average inference time. Obviously, the performance on one single NCS is reluctant for real-time, that is, our ECMask can run at the average 6.09 FPS by using one NCS to accelerate. However, through the distributed computing method, the more NCSs are used to accelerate, the more efficiency ECMask has. Moreover, the efficiency of 3 NCSs with distributed computing can be increased by 2.1 times, which is enough to meet the real-time industrial requirement of video analysis. As the number of NCS reaches 4, the increase in efficiency is no longer obvious. Therefore, as the number of NCSs increases, the growth curve of the FPS would tend to smooth. Furthermore, on the basis of saving costs and improving performance as much as possible, low-cost edge devices with edge computing through NCSs can reach mostly the performance of the higher prices CPU hardware.

V. Conclusion

In this paper, we developed an edge computing-based mask identification framework (ECMask), which can identify the condition of mask-wearing in the videos from bus driver monitoring in real-time. Firstly, the blur detection with the Laplacian operator is used to determine whether video restoration is needed. Then, after possible video restoration, the accuracy of face detection can be improved. Finally, the cropped face images are further used for mask identification. Cooperative edge computing is implemented by the distributed computing with multiple NCSs as low-cost devices. The trained deep

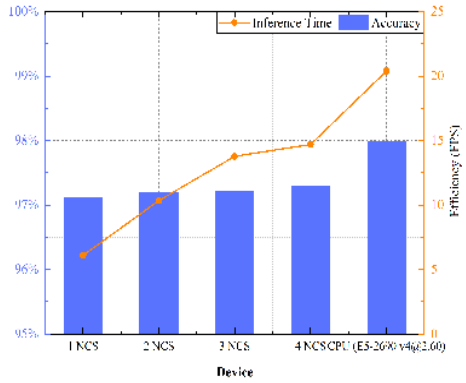


Fig. 8: Inference time and accuracy results for mask identification on the multiple NCSs and CPU (E5-2690 v4@2.60), respectively.

learning model is optimized and stored in the NCSs with the inference engine. Our results present that ECMask has not only high accuracy in face detection and mask identification, but also has the real-time ability of video analysis, which is significant for the healthcare systems of COVID-19 in public places, like buses.

REFERENCES

[1] A. Sharma, S. Tiwari, M. K. Deb, and J. L. Marty, "Severe acute respiratory syndrome coronavirus-2 (sars-cov-2): a global pandemic and treatment strategies," *International Journal of Antimicrobial Agents*, p. 106054, 2020.

[2] W. H. Organization *et al.*, "Coronavirus disease (covid 19): situation report, 165," 2020.

[3] M. U. Kraemer, C.-H. Yang, B. Gutierrez, C.-H. Wu, B. Klein, D. M. Pigott, L. Du Plessis, N. R. Faria, R. Li, W. P. Hanage *et al.*, "The effect of human mobility and control measures on the covid-19 epidemic in china," *Science*, vol. 368, no. 6490, pp. 493–497, 2020.

[4] S. Feng, C. Shen, N. Xia, W. Song, M. Fan, and B. J. Cowling, "Rational use of face masks in the covid-19 pandemic," *The Lancet Respiratory Medicine*, vol. 8, no. 5, pp. 434–436, 2020.

[5] Z. Ning, P. Dong, X. Wang, X. Hu, J. Liu, L. Guo, B. Hu, R. Kwok, and V. C. M. Leung, "Partial computation offloading and adaptive task scheduling for 5g-enabled vehicular networks," *IEEE Transactions on Mobile Computing*, 2020.

[6] X. Kong, X. Liu, B. Jedari, M. Li, L. Wan, and F. Xia, "Mobile crowdsourcing in smart cities: Technologies, applications, and future challenges," *IEEE Internet of Things Journal*, vol. 6, no. 5, pp. 8095–8113, 2019.

[7] D. Chen, P. Bovornkeeratiroj, D. Irwin, and P. Shenoy, "Private memoirs of iot devices: Safeguarding user privacy in the iot era," in *2018 IEEE 38th International Conference on Distributed Computing Systems (ICDCS)*, 2018, pp. 1327–1336.

[8] Z. Ning, S. Sun, X. Wang, L. Guo, G. Wang, X. Gao, and R. Y. Kwok, "Intelligent resource allocation in mobile blockchain for privacy and security transactions: A deep reinforcement learning based approach," *SCIENCE CHINA Information Sciences*, 2020.

[9] A. Gatouillat, Y. Badr, B. Massot, and E. Sejdić, "Internet of medical things: A review of recent contributions dealing with cyber-physical systems in medicine," *IEEE Internet of Things Journal*, vol. 5, no. 5, pp. 3810–3822, 2018.

[10] X. Zhou, W. Liang, K. I. Wang, H. Wang, L. T. Yang, and Q. Jin, "Deep-learning-enhanced human activity recognition for internet of healthcare things," *IEEE Internet of Things Journal*, vol. 7, no. 7, pp. 6429–6438, 2020.

[11] F. Al-Turjman, M. H. Nawaz, and U. D. Ulusar, "Intelligence in the internet of medical things era: A systematic review of current and future trends," *Computer Communications*, vol. 150, pp. 644–660, 2020.

[12] X. Zhou, Y. Hu, W. Liang, J. Ma, and Q. Jin, "Variational lstm enhanced anomaly detection for industrial big data," *IEEE Transactions on Industrial Informatics*, 2020, doi: 10.1109/TII.2020.3022432.

[13] Z. Deng, Y. Zhou, D. Wu, G. Ye, M. Chen, and L. Xiao, "Utility maximization of cloud-based in-car video recording over vehicular access networks," *IEEE Internet of Things Journal*, vol. 5, no. 6, pp. 5213–5226, 2018.

[14] R. Basatneh, B. Najafi, and D. G. Armstrong, "Health sensors, smart home devices, and the internet of medical things: An opportunity for dramatic improvement in care for the lower extremity complications of diabetes," *Journal of Diabetes Science and Technology*, vol. 12, no. 3, pp. 577–586, 2018.

[15] Z. Ning, P. Dong, X. Wang, X. Hu, S. Guo, T. Qiu, and R. Y. Kwok, "Distributed and dynamic service placement in pervasive edge computing networks," *IEEE Transactions on Parallel and Distributed Systems*, 2020.

[16] X. Wang, Z. Ning, and S. Guo, "Multi-agent imitation learning for pervasive edge computing: a decentralized computation offloading algorithm," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 2, pp. 411–425, 2021.

[17] Z. Ning, P. Dong, X. Wang, X. Hu, L. Guo, B. Hu, Y. Guo, T. Qiu, and R. Kwok, "Mobile edge computing enabled 5g health monitoring for internet of medical things: A decentralized game theoretic approach," *IEEE J. Sel. Areas Commun.*, pp. 1–16, 2020.

[18] X. Kong, S. Tong, H. Gao, G. Shen, K. Wang, M. Collotta, I. You, and S. Das, "Mobile edge cooperation optimization for wearable internet of things: A network representation-based framework," *IEEE Transactions on Industrial Informatics*, 2020.

[19] Z. Ning, K. Zhang, X. Wang, L. Guo, X. Hu, J. Huang, B. Hu, and R. Y. Kwok, "Intelligent edge computing in internet of vehicles: A joint computation offloading and caching solution," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–14, 2020.

[20] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "From facial parts responses to face detection: A deep learning approach," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, December 2015.

[21] E. Ohn-Bar and M. M. Trivedi, "To boost or not to boost? on the limits of boosted trees for object detection," in *2016 23rd International Conference on Pattern Recognition (ICPR)*, 2016, pp. 3350–3355.

[22] C. Zhu, Y. Zheng, K. Luu, and M. Savvides, "Cms-rcnn: contextual multi-scale region-based cnn for unconstrained face detection," in *Deep Learning for Biometrics*. Springer, 2017, pp. 57–79.

[23] S. Yang, Y. Xiong, C. C. Loy, and X. Tang, "Face detection through scale-friendly deep convolutional networks," 2017.

[24] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, 2016.

[25] X. Wang, K. C. Chan, K. Yu, C. Dong, and C. Change Loy, "Edvr: Video restoration with enhanced deformable convolutional networks," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019.

[26] P. Pace, G. Aloi, R. Gravina, G. Caliciuri, G. Fortino, and A. Liotta, "An edge-based architecture to support efficient applications for healthcare industry 4.0," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 1, pp. 481–489, 2019.

[27] T. Han, L. Zhang, S. Pirbhulal, W. Wu, and V. H. C. de Albuquerque, "A novel cluster head selection technique for edge-computing based iomt systems," *Computer Networks*, vol. 158, pp. 114–122, 2019.

[28] I. V. Pustokhina, D. A. Pustokhin, D. Gupta, A. Khanna, K. Shankar, and G. N. Nguyen, "An effective training scheme for deep neural network in edge computing enabled internet of medical things (iomt) systems," *IEEE Access*, vol. 8, pp. 107 112–107 123, 2020.

[29] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *European Conference on Computer Vision*. Springer, 2014, pp. 184–199.

[30] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[31] Y. Tian, Y. Zhang, Y. Fu, and C. Xu, "Tdan: Temporally-deformable alignment network for video super-resolution," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[32] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video enhancement with task-oriented flow," *International Journal of Computer Vision*, vol. 127, no. 8, pp. 1106–1125, 2019.

[33] S. Su, M. Delbracio, J. Wang, G. Sapiro, W. Heidrich, and O. Wang, "Deep video deblurring for hand-held cameras," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1279–1288.

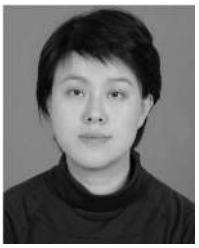
[34] S. Ren, K. He, R. Girshick, and J. Sun, "Faster r-cnn: Towards real-time object detection with region proposal networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 91–99.
[35] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, "S3fd: Single shot scale-invariant face detector," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.
[36] M. Najibi, P. Samangouei, R. Chellappa, and L. S. Davis, "Ssh: Single stage headless face detector," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017.
[37] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.
[38] S. Zhang, X. Zhu, Z. Lei, H. Shi, X. Wang, and S. Z. Li, "Faceboxes: A cpu real-time face detector with high accuracy," in *2017 IEEE International Joint Conference on Biometrics (IJCB)*, 2017, pp. 1–9.
[39] W. Shang, K. Sohn, D. Almeida, and H. Lee, "Understanding and improving convolutional neural networks via concatenated rectified linear units," in *International Conference on Machine Learning*, 2016, pp. 2217–2225.
[40] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2818–2826.

Xiangjie Kong (M'13-SM'17) received the B.Sc. and Ph.D. degrees from Zhejiang University, Hangzhou, China. He is currently a Full Professor with College of Computer Science and Technology, Zhejiang University of Technology. Previously. Previously, he was an Associate Professor with the School of Software, Dalian University of Technology, China. He has published over 130 scientific papers in international journals and conferences (with over 100 indexed by ISI SCIE). His research interests include network science, mobile computing, and computational social science.

Kailai Wang received the B.Sc. degree in software engineering from the Dalian University of Technology, China, in 2019, where he is currently pursuing the master's degree with the School of Software. His research interests include analysis of complex networks, network science, and urban computing.

Shupeng Wang received the B.Sc. degree in software engineering from the Dalian University of Technology, China, in 2019, where he is currently pursuing the master's degree with the School of Software. His research interests include analysis of complex networks, network science, and urban computing.

Xiaojie Wang received the M.S. degree from Northeastern University, China, in 2011. From 2011 to 2015, she was a software engineer in NeuSoft Corporation, China. She received the PhD degree from Dalian University of Technology, Dalian, China, in 2019. Currently, she is a Distinguished Professor with Chongqing University of Posts and Telecommunications, China. Her research interests are wireless networks, mobile edge computing and machine learning.



Xin Jiang received her Ph.D. degree from Tongji Medical College, Huazhong University of Science and Technology in 2008. From 2009 to 2010, she was a visiting scholar at the Prince of Wales Hospital, Chinese University of Hong Kong. She also studied at the Global Clinical Scholars Research Training Program from 2015 to 2016. She is the chief physician, an associate professor, and the deputy director of Geriatrics at Jinan University.

Yi Guo received the Ph.D. degree from the University of Greifswald, Greifswald, Germany, in 1997. He is currently the Chief of neurology with the Second Clinical Medical College, Jinan University, a member of the Cerebrovascular Disease Group, Chinese Medical Association neurology branch, and the Chairman of the Shenzhen Medical Association of Neurology and the Shenzhen Medical Association of Psychosomatic Medicine. His major research areas are cerebrovascular diseases, dementia, movement disorder diseases, sleep disorder, and depression and anxiety.

Guojiang Shen received the B.Sc. degree in control theory and control engineering and the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 1999 and 2004, respectively. He is currently a Professor with College of Computer Science and Technology, Zhejiang University of Technology. His current research interests include artificial intelligence theory, big data analytics, and intelligent transportation systems.

Xin Chen received the B.Sc. degree in information security from Harbin Engineering University, Harbin, China, in 2020. He is currently a graduate student at the School of Software at Dalian University of Technology. His research interests include graph neural network, mobile computing, and big data analytics.

Qichao Ni received the BSc degree in automation (program for excellent engineers) from Yanshan University, China, in 2019. He is currently working toward the master's degree in the School of Software, Dalian University of Technology, China. His research interests include urban data mining, network embedding, and multiple heterogeneous data fusion.