Device-Edge-Cloud Collaborative Acceleration Method Towards Occluded Face Recognition in High-Traffic Areas

Puning Zhang, *Member, IEEE*, Fengyi Huang, Dapeng Wu, *Senior Member, IEEE*, Boran Yang,
Zhigang Yang, and Lei Tan

***Abstract*—Wearing masks can effectively inhibit the spread and damage of COVID-19. A device-edge-cloud collaborative recognition architecture is designed in this paper, and our proposed device-edge-cloud collaborative recognition acceleration method can make full use of the geographically widespread computing resources of devices, edge servers, and cloud clusters. First, we establish a hierarchical collaborative occluded face recognition model, including a lightweight occluded face detection module and a feature-enhanced elastic margin face recognition module, to achieve the accurate localization and precise recognition of occluded faces. Second, considering the responsiveness of occluded face detection services, a context-aware acceleration method is devised for collaborative occluded face recognition to minimize the service delay. Experimental results show that compared with state-of-the-art recognition models, the proposed acceleration method leveraging device-edge-cloud collaborations can effectively reduce the recognition delay by 16% while retaining the equivalent recognition accuracy.**

***Index Terms*—Occluded face recognition, device-edge-cloud collaboration, recognition acceleration, model partitioning.**

I. Introduction

MORE than 530.27 million people were infected by COVID-19 and more than 6.3 million COVID-19 deaths were reported worldwide as of early June 2022. In high-traffic areas such as high-speed rail stations and interstate bus stations, although wearing a mask is effective in eliminating the cross-contamination of the virus, it challenges many effective and convenient face recognition services, e.g., touchless check-in and luggage delivery. As a mask occludes a large area of the face of people inevitably, fewer facial features can be captured and extracted, dramatically reducing the accuracy of commonly adopted face recognition algorithms [1].

To improve the occluded face recognition accuracy, existing studies are focused largely on the optimization models. He et al. [2] simulated face occlusion by local discarding and feature erasing, and designed an attention module to prioritize the impact of non-occluded regions on face recognition. These models mostly rely on powerful neural networks for feature extraction, feature fusion, and feature reconstruction. Their network structures tend to be extremely deep and have countless parameters, which consumed too much computing power to be deployed on the resource-limited smart Internet of Things (IoT) devices.The emergence of cloud computing and edge computing enabling IoT devices to offload their compute-intensive and data-hungry occluded face recognition tasks. Based on edge computing, Wang et al. [3] deployed the recognition model and offloaded inference tasks to edge servers.

Model partitioning[13], [14] is an up-to-date methodology to coordinate and utilize the computing power of multiple devices reasonably and flexibly. Most of the existing model partitioning studies are coarse-grained and only focused on the hierarchical chain and DAG. Therefore, they are incompatible with the challenging occluded face recognition tasks and thereby have inferior performances.In addition, the early exiting mechanism [21] was proposed to augment the deep learning architecture.

Manuscript received 14 July 2022; revised 20 November 2022 and 11 January 2023; accepted 16 January 2023. Date of publication 22 March 2023; date of current version 8 May 2023. This work was supported in part by the National Natural Science Foundation of China under Grants 61901071, 62271096, and U20A20157, in part by the Science and Natural Science Foundation of Chongqing, China under Grant cstc2020jcyj-zdxmX0024 and CSTB2022NSCQ-MSX0600, in part by the University Innovation Research Group of Chongqing under Grant CXQT20017, in part by the Program for Innovation Team Building at Institutions of Higher Education in Chongqing under Grant CXTDX201601020, in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJQN202000626, in part by Youth Innovation Group Support Program of ICE Discipline of CQUPT under Grant SCIE-QN-2022-04, in part by the Chongqing Municipal Technology Innovation and Application Development Special Key Project under Grant cstc2020jscx-dxwtBX0053, and in part by China Postdoctoral Science Foundation under Grant 2022MD723723. The guest editor coordinating the review of this manuscript and approving it for publication was Dr. Shanshan Zhang. *(Corresponding author: Dapeng Wu.)*

Puning Zhang is with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China, also with Advanced Network and Intelligent Connection Technology Key Laboratory of Chongqing Education Commission of China, Chongqing Key Laboratory of Ubiquitous Sensing and Networking, Chongqing 400065, China, and also with Chongqing Innovation Center of Industrial Big-Data Company Ltd., National Engineering Laboratory for Industrial Big-data Application Technology, Chongqing 400707, China (e-mail: zhangpn@cqupt.edu.cn).

Fengyi Huang, Dapeng Wu, and Lei Tan are with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China, and also with the Advanced Network and Intelligent Connection Technology Key Laboratory of Chongqing Education Commission of China, Chongqing Key Laboratory of Ubiquitous Sensing and Networking, Chongqing 400065, China (e-mail: 1293700309@qq.com; wudp@cqupt.edu.cn; s200131185@stu.cqupt.edu.cn).

Boran Yang is with School of Artificial Intelligence, Chongqing University of Technology, Chongqing 401135, China (e-mail: bryangphd@163.com).

Zhigang Yang is with the School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, Chongqing 400065, China (e-mail: yangzhigang@cqupt.edu.cn).

Digital Object Identifier 10.1109/TMM.2023.3240884

To solve the above problems, a device-edge-cloud collaborative occluded face recognition architecture is designed. The contributions of this paper are listed as follows.

1) A device-edge-cloud collaborative recognition architecture is designed. A lightweight occluded face detection model is proposed for devices to achieve local detection, and the overall occluded face recognition model is partitioned and deployed on the edge and cloud partly to realize collaborative and responsive recognition.
2) A hierarchical collaborative occluded face recognition model is established. A lightweight occluded face detection model based on feature similarity estimation is designed, leveraging the correlation between feature mappings to simplify the convolution operation. A feature-enhanced elastic margin face recognition model is devised to improve the feature discriminability by introducing an attention mechanism to emphasize the features from the uncovered areas of occluded faces.
3) A context-aware acceleration method is proposed to construct and optimize the abstract recognition model delay graph by introducing contextual information. A model partitioning algorithm is designed and an early exiting algorithm is developed to further enhance the service responsiveness.

II. Related Works

A. Occluded Face Recognition

The recognition of occluded faces is susceptible to inherent factors such as the lack of facial information, the loss of facial features, and the diversity of occlusions. There have been studies adopting traditional face recognition methods. In [4], Local Binary Patterns (LBP) were introduced to extract the texture and local details of face features.

The recognition methods discarding the occluded region include sparse coding based on subspace regression and feature co-representation [5]. He et al. [2] discarded useless features and enhanced the weight of unoccluded regions for occluded face recognition. Ding et al. [6] designed a two-branch Convolutional Neural Network (CNN) model. Pedro et al. [36] proposed a multi-task architecture based on comparative learning which could improve the focus on areas not covered by masks.

The reconstruction method of occluded faces mainly utilizes the redundancy of image information. Malakar et al. [7] and Song et al. [8] used the unoccluded regions as the principal component to reconstruct the occluded faces. Hao et al. [9] introduced a correction block for the consistency between occluded and unoccluded face features to minimize their distances in the correction space.

B. Collaborative Edge Intelligence

The emerging paradigm of edge computing offloads compute-intensive tasks and services from the network core to the network edge, which opens a new interdisciplinary research direction, edge Artificial Intelligence (AI) a.k.a. Edge Intelligence (EI). It can increase the data processing speed and reduce the response time compared to cloud-based AI model deployments. Most current image recognition applications prefer to utilize edge resources[10]. Wang et al. [3] performed the object area detection at local devices and offloaded the fine-grained small object recognition to the edge. However, the accuracy and latency performances are still limited due to the relatively restricted computation resources of the edge servers.

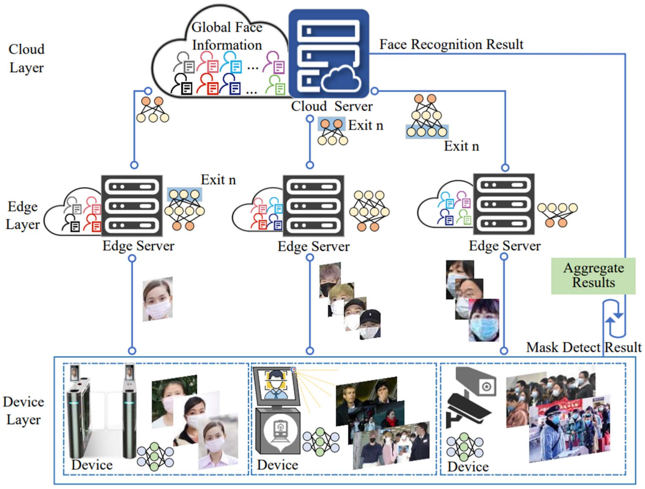

Fig. 1. Device-edge-cloud collaborative recognition architecture.

Current researches in the field of computer vision are combined with device-edge-cloud architecture for task computing, such as Kong et al. [11] presented a device-edge-cloud collaborative face detection framework.

C. Model Partitioning

The deployment of model partitioning algorithms aims at dividing the entire DNN into two or more subnetworks and deploying corresponding subnetworks on suitable equipment with diversified computing power, thereby integrating and fully utilizing the geographically widespread computing resources to accelerate the inference process. Existing model partitioning methods are mainly divided into two categories: static model partitioning [12] and dynamic model partitioning [13]. Hu et al. constructed [14] the residual module and other fixed module structures for model partitioning for the first time, leveraged graph theory to build a time-delay DAG, and solved it using the min-cut algorithm.

III. Device-Edge-Cloud Collaborative Recognition Architecture

A device-edge-cloud collaborative occluded face recognition architecture is designed, which consists of the device layer, edge layer, and cloud layer, as shown in Fig. 1.

The device layer consists of resource-constrained devices responsible for collecting raw images and locating occluded face areas. The proposed lightweight occluded face detection model deploys face detectors on devices with smart cameras to perform mask-wearing detection, transfers cropped occluded face images to edge servers.

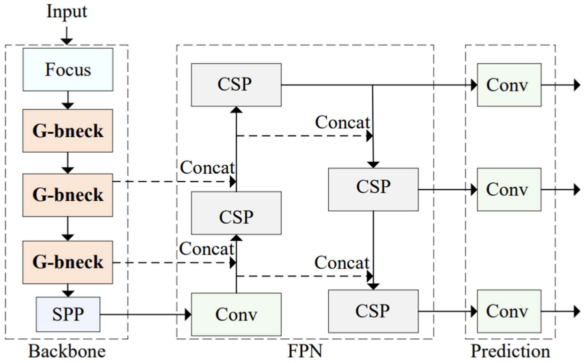

Fig. 2. Feature similarity estimation module.

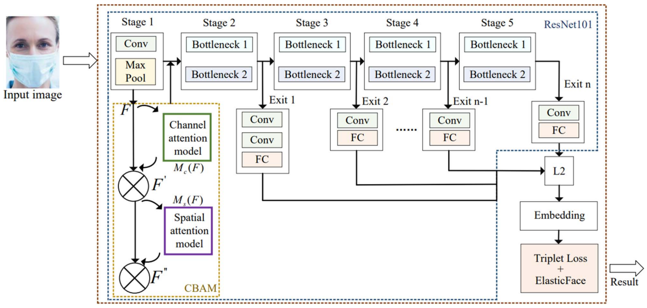

Fig. 3. Feature-enhanced elastic margin face recognition model.

The edge layer consists of multiple edge servers for recognition model partitioning decisions and implementing occluded face recognition tasks partially. A feature-enhanced elastic margin face recognition model is designed. Dynamic recognition model partitioning and early exiting decisions are made based on deep reinforcement learning and contextual information.

The cloud layer contains powerful cloud server clusters with abundant computing resources. The cloud clusters receive the model partitioning policy reported by edge servers and perform the rest of the occluded face recognition tasks on their subnetworks.

IV. Hierarchical Collaborative Occluded Face Recognition Model

A. *Lightweight Occluded Face Detection Model Based on Feature Similarity Estimation*

In this subsection, we propose a lightweight detection model for occluded face images, and then the device layer transmits only cropped occluded face images to edge servers.

The network structure of the proposed occluded face recognition model is shown in Fig. 3, where a lightweight occluded face detection model based on **F**eature **S**imilarity **E**stimation (FSE) is introduced as the detector for three categories of mask-wearing status, i.e., a face, a face_mask, and an incorrect_mask. The lightweight detector can identify the occluded face of a target accurately and is robust against occlusion. YOLOv5s [28] is chosen as the base network for mask detection. The stacked Ghost [15] modules form a residual structure to perform dimension ascending and descending operations. The loss function used in our proposed occluded face recognition model is defined as (1):

$$L_a = L_{obj} + L_{class} + L_{EIOU}, \tag{1}$$

where L_{obj} is the original confidence loss function in YOLOv5s and L_{class} is the loss function of the classification. Our proposed occluded face recognition model introduces an improved prediction frame loss function[16], L_{EIOU}, for recognition accuracy.

B. *Feature-Enhanced Elastic Margin Face Recognition*

We design a **F**eature-enhanced **E**lastic **M**argin face recognition model (FEM) for edge servers, as shown in Fig. 3, which includes a backbone feature extraction network, a hybrid attention module, and an early exiting module. First, ResNet101 is utilized as the feature extraction network similar to FaceNet [17]. we incorporate a lightweight hybrid attention module, Convolution Block Attention Module (CBAM) [18].

Applying CBAM to the first convolutional layer of the feature extraction network enables better attention to the upper and unmasked area of faces to improve the accuracy and robustness of occluded face recognition tasks.

In addition, we propose the joint supervision of the triplet loss and Elastic face-arc loss [19] functions to provide high reliability and validity for feature extractions. The joint loss function is shown in (2):

$$L_b = L_1 + \beta L_2, \tag{2}$$

where L_1 is the triplet loss, L_2 is the Elastic face-arc loss, and β is the equilibrium coefficient. The triplet loss is defined as:

$$L_1 = \sum_{i}^{N} \left[||f(x_i^a) - f(x_i^p)||_2^2 - ||f(x_i^a) - f(x_i^n)||_2^2 \right] + \alpha, \tag{3}$$

where α represents the boundary value for positive and negative samples, and α with higher values indicates higher discrimination degrees. N represents the number of samples contained in the set. x_i^a represents the ith selected sample a to be tested, x_i^p represents the ith positive sample p, and x_i^n represents the ith negative sample n.

$$L_2 = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{e^{s(\cos(\theta_{y_i}+E(m,\sigma)))}}{e^{s(\cos(\theta_{y_i}+E(m,\sigma)))} + \sum_{j=1,j\neq y_i}^{n} e^{s(\cos(\theta_j))}}, \tag{4}$$

θ_j is the angle between full connection layer weight and the deeper feature of the sample, and $E(m,\sigma)$ is a function following the normal distribution and provides more flexibility in classification by returning a random value that matches the Gaussian distribution as the boundary penalty value.

V. Context-Aware Occluded Face Recognition Acceleration Method

To further optimize the latency performance of the occluded face recognition in high-traffic areas, we propose a context-aware acceleration method seeking the optimal partitioning

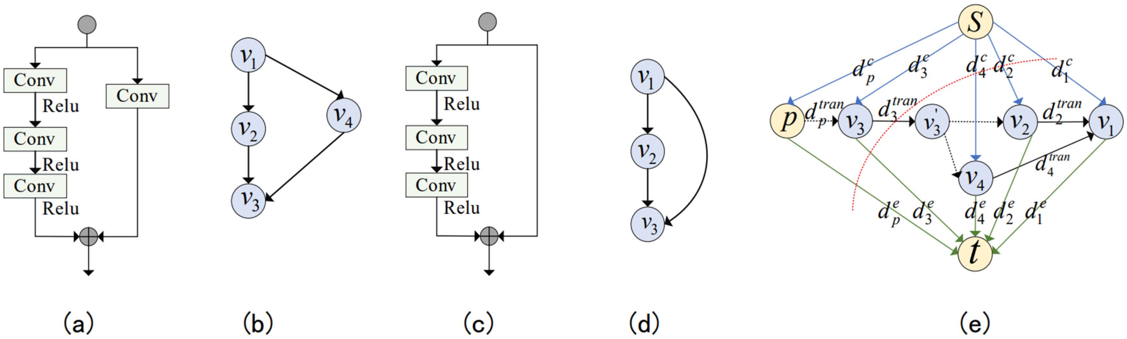

Fig. 4. (a), (c) Two structures of the residual module. (b), (d) Two residual modules abstracted as directed acyclic graphs. (e) Delay modeling G' for (b).

points and exiting points with evolutionary contextual dynamics taken into account in this section.

A. Recognition Delay Graph Optimization

The edge layer employs FEM with a deep network structure, and the backbone ResNet101 is composed of multiple residual modules. In this section, the delay graph of sub-graph residual modules are reasonably optimized for model partitioning.For the selection of model partitioning points, the partitioning point set should contain hierarchical chain partitioning points and subgraph partitioning points. The delay model of the modules containing subgraphs in ResNet101 can be abstracted as a DAG: $G = (V, E)$, in which the optimal partitioning point is found such that the total delay is minimized under unpredictable network dynamics [14]. As shown in Fig. 4, (a) and (c) are the subgraphs of residual modules in ResNet101, (b) and (d) are the latency DAG of ResNet101, and (e) is the latency model of each node.

We create virtual nodes s and t in the subgraph, abstract each DNN layer into a node v_i and associate the delay of each node as $(d_i^e, d_i^{tran}, d_i^c)$. The edge $d_i^c \in D^{clo}$ of s connected to v_i indicates the processing delay of the layer on the cloud.The edge $d_i^e \in D^{edg}$ of t connected to v_i indicates the processing latency of the subnetwork on edge servers. The edge $d_i^{tran} \in D^{tran}$ between layer v_i and layer v_j represents the transmission delay of the data output from that layer as a partitioning layer. The transmission delay $d_i^{tran} = d_i^{in}/B$, and d_i^{in} is the size of the input data at the layer i and B is the dynamically varying bandwidth. Accordingly, the total delay for the ResNet101 partitioning is computed as:

$$D_1 = D^{edg} + D^{tran} + D^{clo}, \tag{5}$$

For DAG-form DNNs, a vertex may have multiple successor nodes, where the communication delay of each layer is calculated multiple times. Consequently, for the out-degree k (k>1) of a vertex, the vertex is replicated k times. The replicated node of v_3 in Fig. 5 is v_3'. Considering that cloud clusters have more computing power than edge servers, d_i^c is always smaller than d_i^e which can easily cause the occluded face recognition DNN to be deployed on the cloud completely. Therefore, we introduce an external node p and set $d_p^c = \infty$, $d_p^e = 0$, d_p^{tran} as the delay for all occluded face images to be transmitted to the cloud, $d_p^{tran} > d_1^{tran}$.

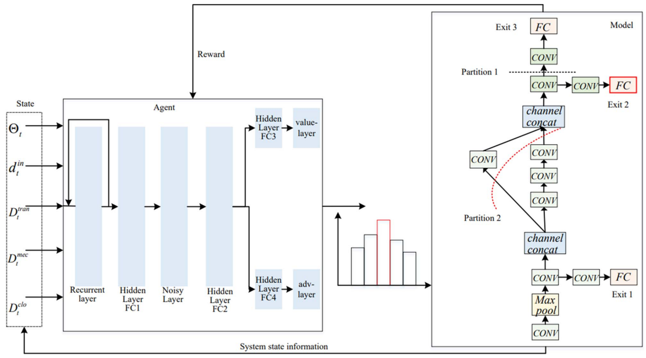

Fig. 5. Workflow of rainbow DQN performing model partitioning and early exiting point selection.

The backbone ResNet101 extracts the delay DAG abstracted from the residual module, employs the Orlin [20] algorithm to calculate the maximum flow minimum cut of current delay graph G', and finally divides the nodes within the residual module into two sets V_c and V_e. The minimum time complexity of the Orlin algorithm is $O(mn)$, which can improve the computational efficiency of model partitioning, where m is the number of edges and n is the number of nodes.

B. Context-Aware Model Partitioning Algorithm

To further accelerate occluded face recognition, we combine the fine-grained model partitioning algorithm with the early exiting mechanism, and a recognition result can be obtained in advance at Exit n. Based on the open-source framework BranchyNet [21], we train the occluded face recognition model to generate subnetworks containing early exiting branches to feedback results when the accuracy respects user requirements. We use Rainbow DQN [22] to generate a selection strategy for partitioning and exiting points to achieve a balance between delay and accuracy.

The state space of model partitioning is composed of the accuracy, data size, and service delay:

$$S = \{s | s_t = (\Theta_t, d_t^{in}, D_t^{tran}, D_t^{edg}, D_t^{clo})\}, \tag{6}$$

where Θ_t represents the model real-time inference accuracy; d_t^{in} represents the size of input data at the device layer, D_t^{tran} represents the latency of the feature mapping from edge servers to cloud clusters; D_t^{edg} represents the processing latency of the subnetwork on the edge; and D_t^{clo} represents the processing latency of the subnetwork on the cloud.

The model partitioning strategy includes choosing the partitioning points in the DNN for the edge-cloud collaboration, and the early exiting points to end the recognition computation and feedback results early. The action space is shown in (7):

$$\begin{aligned} &A = \{a|a_{t,j,k} = (P_{t,j}, E_{t,k})\},\\ &j \in [1,\ldots,J], k \in [1,\ldots,K], \end{aligned} \tag{7}$$

where $P_{t,j}$ represents the set of partitioning points containing points between layers and points within the subgraph. The selection of partitioning points within the subgraph is derived according to the optimization of the delay DAG in Section V-A. $E_{t,k}$ represents an early exiting point from the branch. When choosing partitioning points and exiting points, the closer to the back end of the overall recognition model, the higher the exiting probability of the branch. Exiting point $k \in [1,\ldots,K]$, $k = 3$ represents the output layer of the occluded face recognition DNN.

An agent maximizes the total reward at each time point by deciding the action to take. The action decision can be denoted as a sequence defined by action strategy π, given state s_t^π. Therefore, the objective function for the agent is to find the best strategy π^* within the time interval, expressed as in (8):

$$\begin{aligned} &\max_{\pi\in\Pi} E_S^\pi \left[\sum_{t=1}^{N} r_t(s_t^\pi, a_t)\right],\\ &\text{s.t.}\, C1: 0 \le l_t \le L, t \in \mathrm{T},\\ &C2: \Theta_t \ge P, t \in \mathrm{T},\\ &C3: \sum_{i\in N} p_{i,t}^m \le maxp_t^m,\\ &C4: \sum_{i\in N} p_{i,t}^c \le maxp_t^c, \end{aligned} \tag{8}$$

where $C1$ represents the predetermined latency requirement, $C2$ represents the accuracy requirement, $C3$ represents the computing capacity of edge servers, and $C4$ represents the computing capacity of the cloud.

The computing power of the edge side and the cloud is represented by the computing power of the server. The computing frequencies are set as f_t^m and f_t^c, respectively, and the energy efficiency coefficients are set as ζ_t^m and ζ_t^c, respectively. Therefore, the computing power is $p_t^m = \zeta_t^m \cdot (f_t^m)^3$ on the edge side and $p_t^c = \zeta_t^c \cdot (f_t^c)^3$ on the cloud layer.

Given input state s_t, the Rainbow DQN outputs a series of schemes to obtain model partitioning points and early exiting points, and records the total processing delay and accuracy. Based on the above process, the reward function is obtained as in (9):

$$r_t = \begin{cases} e^{\phi\Theta_t}, l_t \le L,\\ 0, else, \end{cases} \tag{9}$$

where e is the natural base, L is the latency requirement for the occluded face recognition task, Θ_t is the current recognition accuracy, l_t is the current service latency, and ϕ is a hyperparameter set to adjust the magnitude of the reward. The obtained reward will be feedback to update and collect system state information as the next input.The workflow of the Rainbow DQN is shown in Fig. 5.

VI. Experimental Simulation

A. Experimental Setup

The simulation is based on Python for algorithm development, using single Raspberry Pi to simulate IoT device, one desktops with i3 CPUs to simulate edge server, and a server with Nvidia CUDA10.0 to simulate cloud clusters. MaskedFace-Net [23] is an open-source mask-wearing face dataset. And in this paper, images of real faces and masked faces are used from the open source data set [32], [33] of Kaggle platform.

The occluded face recognition model on the edge layer leverages VGG-Face2-train and Masked VGG-Face2 as the training sets, and LFW, Masked LFW, and partial Masked VGG-Face2-test as the test sets. Masked LFW is a dataset for face recognition, and VGG-Face2-test to evaluate the accuracy of occluded face recognition.

B. Model Performance

In this section, the lightweight model designed in this paper is first compared with a state-of-the-art generic target detector as shown in Table I. The proposed lightweight model has only 4.63 M parameters and 8.51 G computing power, which are much smaller than those of other target recognition models, and about 36% and 46% less than the benchmark YOLOv5s, respectively, as shown in Table II.

After occluded face detection, the results are cached on the devices and only the cropped occluded face images are transmitted to edge servers. The occluded face detection test set prensents a reduction up to 92.88% in data sizes. The improvement is even more obvious for high-definition and high-quality images on face verifiers and cameras, which substantially enhances the transfer efficiency of useful data. Three deployment cases are considered on Raspberry Pis simulating resource-constrained IoT devices: (1) MTCNN [29]: a face alignment model, (2) FSE, (3) Device-edge: no image is processed on the device layer and images are directly transmitted to edge layer for processing. (4) Device-cloud:Instead of processing the images on the device and edge, they're sent directly to the cloud layer. The simulation results are shown in Fig. 6 . The time difference between FSE and MTCNN in image processing and transmission on the device layer is not significant. However, the prediction time of MTCNN as a face alignment model for occluded faces is slightly higher than that of the proposed detection model, and the accuracy for occluded face recognition is much lower than the proposed model. Although Remove-to-edge can have a fast image processing speed, it will incur a long transmission delay, and this situation will be aggravated with the improvement of data quality.

TABLE I
COMPARISON OF MODELS WITH ADVANCED OBJECT DETECTORS

Model		Faster R-CNN[24]	SSD[25]	Cascade R-CNN [26]	RetinaNet[27]	Yolov5s[28]	Yolov6s[34]	Yolov7[35]	FSE
Accuracy	No_mask	0.869	0.947	0.954	0.894	0.951	0.970	0.987	0.968
	Face_mask	0.922	0.965	0.976	0.973	0.964	0.982	0.989	0.984
	Incorrect_mask	0.973	0.954	0.965	0.964	0.963	0.975	0.980	0.973
MAP		0.892	0.884	0.889	0.870	0.933	0.948	0.953	0.945

TABLE II
COMPARISON OF PARAMETER QUANTITIES OF MODELS

Model	RetinaNet	SSD	Cascade R-CNN	Cascade R-CNN	Faster R-CNN	Yolov5s	Yolov6s	Yolov7	FSE
Params(M)	37.97	26,29	104.87	137.1	137.1	7.18	17.23	36.86	4.63
Gflops(G)	169.82	62.8	181.45	370.41	370.41	15.8	44.52	105.12	8.51

TABLE III
COMPARISON OF THE ACCURACY OF OCCLUDED FACE RECOGNITION MODELS ON DIFFERENT DATASETS

Model		LBP	Facenet	Arcface	PDSN	FocusFace	FFR-net	FEM
Accuracy	LFW	0.882	0.967	0.978	0.985	0.974	0.992	0.994
	M-LFW	0.656	0.892	0.904	0.915	0.912	0.942	0.937
	M-VGG-Face2-test	0.647	0.833	0.876	0.904	0.896	0.910	0.923

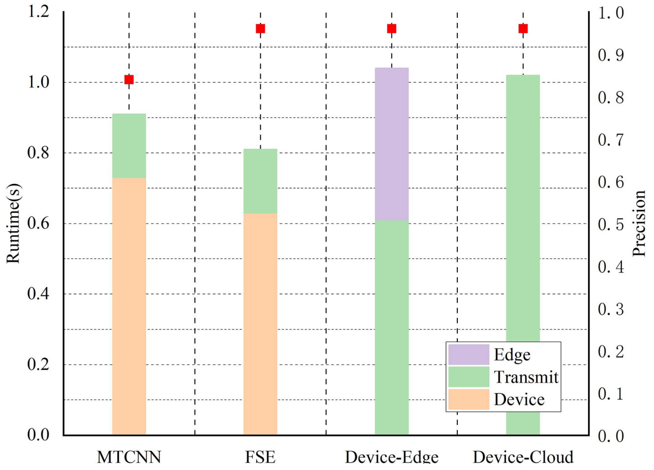

Fig. 6. Device-edge runtime comparison.

TABLE IV
RECOGNITION ACCURACY ON LFW AND MEGAFACE

Model		LCD[2]	FEM
Accuracy	LFW	0.997	0.994
	Megaface	0.836	0.811

TABLE V
ABLATION EXPERIMENT OF OCCLUDED FACE RECOGNITION MODEL FEM

Architecture	Precision
Facenet[27]	0.892
Facenet+CBAM	0.925
Facenet+Elasticloss	0.910
Facenet+CBAM+Elasticloss	0.937

TABLE VI
ACCURACY AND DROP-OUT RATES OF B-FEM

Model	Accuracy	Exit(%)
B-FEM	0.894; 0.917; 0.932	10.34; 26.79; 62.87

As shown in Table II, The traditional face recognition method LBP [4] is not robust and vulnerable enough. Facenet [17] and Arcface [30] have poor performance in occluded face recognition tasks. PDSN [15] exhibits low recognition accuracy on the Masked VGG-Face2 dataset. The state-of-the-art FFR-Net [9] performs best on occluded and unoccluded datasets. FocusFace [36] focuses on the adjustment of the loss function of the contrastive learning architecture, with limited accuracy improvement for occlusion face recognition. Our proposed model achieves the optimal recognition accuracy on Masked VGG-Face2 and outperforms multi-classification occluded face recognition models.

In this paper, the experimental results of LFW and Megaface data sets are as shown in Table IV. Both the method presented in this paper and LCD [3] have better performance on the unshielded common face recognition data set LFW. For Megaface data set, Our algorithm FEM is basically comparable to LCD in terms of recognition accuracy, which reflects the generalization ability of FEM. In addition, LCD only focuses on model improvement, and FEM has more advantages in the aspect of delay optimization. Therefore, FEM is more representative for occluded face recognition in high-traffic area, such as stations, airports.

In order to verify the effectiveness of each part of the occluded face recognition model presented in this paper, a series of ablation experiments are conducted on M-LFW in Table V.

The complete occluded face recognition models are deployed on the edge server and cloud cluster, the running time of each model is measured separately, as shown in Fig. 7.

C. Partitioning Strategy

In this paper,we set 3 early exiting branches[21], and the results are shown in Table VI. The accuracy of these two early exiting points is slightly lower than that of the intact occluded face recognition model, with the sample exiting probability gradually increasing in the three exiting branches.

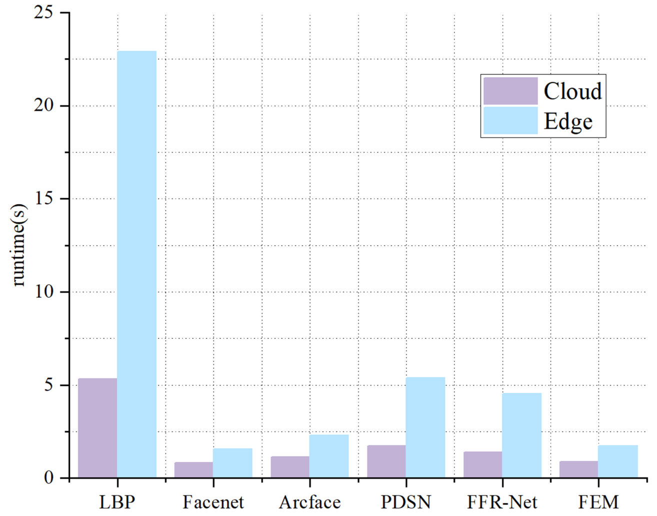

Fig. 7. Comparison of the runtime of different models on the cloud and edge.

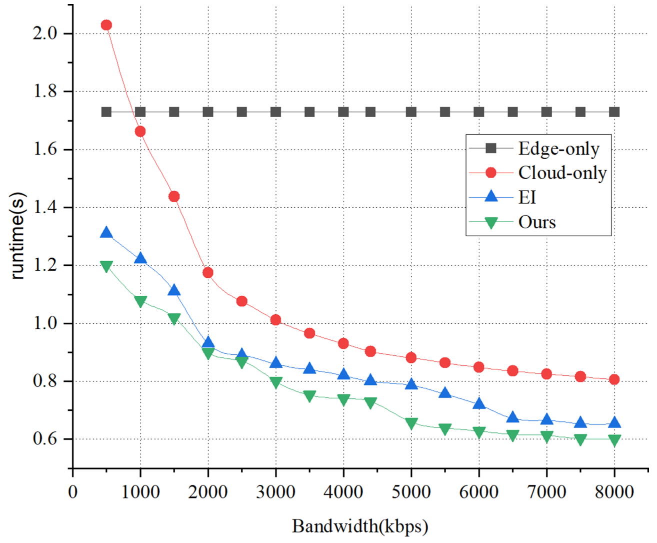

Fig. 8. Comparison of latency of different methods.

The Rainbow DQN is employed to train the model partitioning and early exiting algorithms. As the network bandwidth increases, early exiting points are pushed back, and the model partitioning points are changed.

The task analysis is carried out according to the users' latency tolerance given a fixed network bandwidth, evaluating our proposed acceleration method in selecting model partitioning and early exiting points. As the latency tolerance increases, the early exiting point is pushed back. The change in model partitioning points allows more DNN computations to be processed on the edge layer.

To assess the effectiveness of our proposed acceleration method in occluded face recognition, shown in Fig. 8. Compared with the EI algorithm in [31], which is also based on model partitioning, one can see that the proposed acceleration method has a smaller latency. This is because the proposed acceleration method expands the range of model partitioning points, which covers both hierarchical chain partitioning points and DAG partitioning points for fine-grained model partitioning. Also, we leverage a deep reinforcement learning algorithm to enhance the accuracy of occluded face recognition, which is more sensitive to bandwidth changes. This is because the proposed model partitioning algorithm can adapt to network conditions, thereby effectively reducing the total delay by up to 16%.

VII. Summary

Wearing masks has become a necessary paramedical measure against the big pandemic. A device-edge-cloud collaborative occluded face recognition framework is designed. A hierarchical collaborative occluded face recognition mode is proposed. In addition, an acceleration method is developed for edge-cloud collaborative occluded face recognition to partition and deploy recognition subnetworks on equipment with different computing power. simulation experiments show that the proposed device-edge-cloud collaborative architecture and acceleration algorithm can provide accurate and fast occluded face recognition services.

References

[1] D. Wu, X. Han, Z. Yang, and R. Wang, "Exploiting transfer learning for emotion recognition under cloud-edge-client collaborations," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 2, pp. 479–490, Feb. 2021.

[2] M. He et al., "Locality-aware channel-wisedropout for occluded face recognition," *IEEE Trans. Image Process.*, vol. 31, pp. 788–798, 2022, doi: 10.1109/TIP.2021.3132827.

[3] X. Wang et al., "EdgeDuet: Tiling small object detection for edge assisted autonomous mobile vision," in *Proc. IEEE Conf. Comput. Commun.*, 2021, pp. 1–10, doi: 10.1109/INFOCOM42981.2021.9488843.

[4] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006, doi: 10.1109/TPAMI.2006.244.

[5] C. Jiang, M. Wang, X. Tang, and R. Mao, "Face recognition method based on sparse representation and feature fusion," in *Proc. IEEE Chin. Automat. Congr.*, 2019, pp. 396–400, doi: 10.1109/CAC48633.2019.8997456.

[6] F. Ding et al., "Masked face recognition with latent part detection," in *Proc. 28th ACM Int. Conf. Multimedia*, 2020, pp. 2281–2289.

[7] S. Malakar, W. Chiracharit, K. Chamnongthai, and T. Charoenpong, "Masked face recognition using principal component analysis and deep learning," in *Proc. IEEE 18th Int. Conf. Elect. Eng./Electron., Comput., Telecommun. Inf. Technol.*, 2021, pp. 785–788.

[8] L. Song, D. Gong, Z. Li, C. Liu, and W. Liu, "Occlusion robust face recognition based on mask learning with pairwise differential siamese network," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 773–782.

[9] S. Hao, C. Chen, Z. Chen, and K. -Y. K. Wong, "A unified framework for maskedand mask-free face recognition via feature rectification," in *Proc. IEEE Int. Conf. Image Process.*, 2022, pp. 726–730.

[10] D. P. Wu et al., "Edge-cloud collaboration enabled video service enhancement: A. hybrid human-artificial intelligence scheme," *IEEE Trans. Multimedia*, vol. 23, pp. 2208–2221, 2021.

[11] X. Kong et al., "Real-time mask identification for COVID-19: An edge-computing-based deep learning framework," *IEEE Internet Things J.*, vol. 8, no. 21, pp. 15929–15938, Nov. 2021, doi: 10.1109/JIOT.2021.3051844.

[12] Y. Matsubara and M. Levorato, "Neural compression and filtering for edge-assisted real-time object detection in challenged networks," in *Proc. IEEE 25th Int. Conf. Pattern Recognit.*, 2021, pp. 2272–2279, doi: 10.1109/ICPR48806.2021.9412388.

[13] Y. Kang et al., "Neurosurgeon: Collaborative intelligence between the cloud and mobile edge," *ACM Sigplan Notices*, vol. 52, no. 1, pp. 615–629, 2017.

[14] C. Hu, W. Bao, D. Wang, and F. Liu, "Dynamic adaptive DNN surgery for inference acceleration on the edge," in *Proc. IEEE Conf. Comput. Commun.*, 2019, pp. 1423–1431, doi: 10.1109/INFOCOM.2019.8737614.

[15] K.Y. Han et al., "GhostNet: More features from cheap operations," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2020, 1580–1589.

[16] Y. F. Zhang et al., "Focal and Efficient IOU Loss for Accurate Bounding Box Regression," *Neurocomputing*, vol. 506, pp. 146–157, 2021.

[17] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 815–823.

[18] S. Woo et al., "CBAM: Convolutional block attention module," in *Proc. Eur. Conf. Comput. Vis.*. 2018, pp. 3–19.

[19] F. Boutros, N. Damer, F. Kirchbuchner, and A. Kuijper, "ElasticFace: Elastic margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 1578–1587.

[20] J. B. Orlin, "Max flows in O(nm) time, or better," in *Proc. ACM Symp. Theory Comput.*, 2013, pp. 765–774.

[21] S. Teerapittayanon, B. Mcdaniel, and H. T. Kung, "BranchyNet: Fast inference via early exiting from deep neural networks," in *Proc. IEEE 23rd Int. Conf. Pattern Recognit.*, 2016, pp. 2464–2469, doi: 10.1109/ICPR.2016.7900006.

[22] M. Hessel et al., "Rainbow: Combining improvements in deep reinforcement learning," in *Proc. 32nd AAAI Conf. Artif. Intell.*, 2018, pp. 3215–3222.

[23] A. Cabani et al., "MaskedFace-Net-A dataset of correctly/incorrectly masked face images in the context of COVID-19," *Smart Health*, vol. 19, 2021, Art. no. 100144.

[24] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards realtime object detection with region proposal networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 6, pp. 1137–1149, Jun. 2017.

[25] W. Liu et al., "SSD: Single shot multibox detector," in *Proc. 14th Eur. Conf. Comput. Vis.*, 2016, vol. 9905, pp. 21–37.

[26] Z. Cai and N. Vasconcelos, "Cascade R-CNN: Delving into high quality object detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit*, 2018, pp. 6154–6162.

[27] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 2, pp. 2999–3007, Feb. 2020.

[28] X. Zhu, S. Lyu, X. Wang, and Q. Zhao, "TPH-YOLOv5: Improved YOLOv5 based on transformer prediction head for object detection on drone-captured scenarios," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 2778–2788.

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

[30] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "Arcface: Additive angular margin loss for deep face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 4690–4699.

[31] L. Zeng, E. Li, Z. Zhou, and X. Chen, "Boomerang: On-demand cooperative deep neural network inference for edge intelligence on the industrial Internet of Things," *IEEE Netw.*, vol. 33, no. 5, pp. 96–103, Sep./Oct. 2019.

[32] "Face Mask Dataset," 2022. Online. [Available]: https://www.kaggle.com/datasets/shiekhburhan/face-mask-dataset

[33] "Face mask detector," 2021. Online. [Available]: https://www.kaggle.com/datasets/spandanpatnaik09/face-mask-detectormask-not-mask-incorrect-mask

[34] C. Li et al., "YOLOv6: A. single-stage object detection framework for industrial applications," 2022, *arXiv:2209.02976*.

[35] C. Y. Wang et al., " YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022, *arXiv: 2207.02696*.

[36] P. C. Neto et al., " FocusFace: Multi-task contrastive learning for masked face recognition," in *Proc. IEEE 16th Int. Conf. Autom. Face Gesture Recognit.*, 2021, pp. 01–08.

Puning Zhang (Member, IEEE) received the Ph.D. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2017. He is currently an Associate Professor with the School of Communication and Information Engineering, Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include Internet of Things search, deep learning, and sentiment analysis.

Fengyi Huang is currently working toward the master's degree with the Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include multimedia data processing, object detection, and occluded face recognition.

Dapeng Wu (Senior Member, IEEE) received the Ph.D. degree from the Beijing University of Posts and Telecommunications, Beijing, China, in 2009. He is currently a Professor with the Chongqing University of Posts and Telecommunications, Chongqing, China. He authored more than 100 publications and two books. He is the inventor and co-inventor of 28 patents and patent applications. His research interests include social computing, wireless networks, and big data. Dr. Wu is a TPC Chair of 10th Mobimedia and program committee Member for numerous international conferences and workshops. He is/was the Editor or/and Guest Editor of several technical journals, such as IEEE INTERNET OF THINGS JOURNAL, *Elsevier Digital Communications and Networks*, and *ACM/Springer Mobile Network and Applications*.

Boran Yang received the B.S. and M.S. degrees in 2013 and 2016 from the Chongqing University of Posts and Telecommunications, Chongqing, China, where he is currently working toward the Ph.D. degree. His research interests include edge computing, edge resource sharing, and network security.

Zhigang Yang received the M.S. degree from the Chongqing University of Posts and Telecommunications, Chongqing, China, in 2006. He is currently an Associate Professor with the School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications. He is also an Associate Professor with the Chongqing University of Arts and Sciences, Chongqing. His research interests include edge computing, network security, and privacy.

Lei Tan is currently working toward the master's degree with the Chongqing University of Posts and Telecommunications, Chongqing, China. His research interests include multimedia data processing, object detection, and occluded face recognition.