Edge AR X5: An Edge-Assisted Multi-User Collaborative Framework for Mobile Web Augmented Reality in 5G and Beyond

Pei Ren, Xiuquan Qiao, Yakun Huang, Ling Liu, *Fellow, IEEE,* Calton Pu, *Fellow, IEEE,* Schahram Dustdar, *Fellow, IEEE,* and Junliang Chen

Abstract—Multi-user mobile Augmented Reality (AR) has been successfully used in various fields as a novel visual interaction technology. But current mainstream wearable device-based and app-based solutions are still facing cross-platform, real-time communication, and intensive computing requirements. Mobile Web technology is envisioned to be a promising supporting technology for cross-platform application of mobile AR especially in 5G networks, which provide pervasive communication and computing resources thereby forming a formidable framework for the practical application of multi-user mobile Web AR. However, the problem of how to use these new techniques properly to achieve efficient communication and computing collaboration is obviously paramount in order for multi-user mobile Web AR to be realized in 5G networks. In this article, we propose the first edge-assisted multi-user collaborative framework for mobile Web AR in the 5G era. Firstly, we propose a heuristic mechanism BA-CPP for efficient communication planning, which allows multi-user interaction synchronization to be achieved. Secondly, we introduce a motion-aware key frame selection mechanism called Mo-KFP to optimize the computational efficiency of the edge system, and simultaneously alleviate the initialization problem by collaborating with nearby mobile devices using the Device-to-Device (D2D) communication technique. Experiments are conducted in a real-world 5G network, and the results demonstrate the superiority of our proposed collaborative framework.

Index Terms—Edge Computing, Collaborative Computing, Mobile Computing, Web-based Augmented Reality, 5G Networks.

1 Introduction

As an innovative visual interaction paradigm, multi-user collaborative mobile Augmented Reality (AR) [1], [2], [3] demonstrates tremendous potential for application in various fields, for example, education, entertainment, design, and maintenance [4], [5], [6], [7]. It is clear that when all users are gathered into one "augmented world", more interesting and efficient applications become possible. This has now become a topic of great research interest, and has received considerable attention from Google, Apple, and other companies [8], [9], [10], [11].

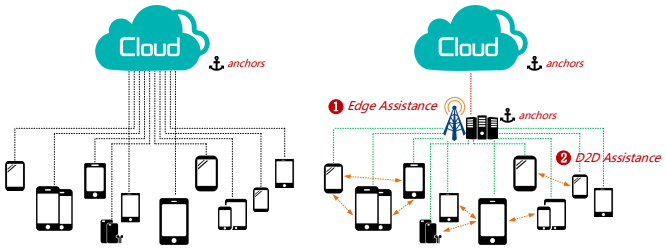
Achieving the full promise of multi-user mobile AR involves the following key aspects: *(1) Cross-platform requirement*. All users should be able to interact with other AR subscribers via any access mode such as AR glasses and smartphones. *(2) Efficient communication requirement*. AR is a latency-sensitive application, and all interactions in the AR world need to be efficiently synchronized with all subscribers in real time. *(3) Economical computing requirement*. Mobile platforms with limited computing capability cannot afford to carry out complex AR computations. When additional computing resources are used for assistance, attention also needs to be paid to the offloading efficiency.

However, current mainstream multi-user mobile AR solutions still facing serious problems: *(1) Restricted cross-platform experience*. Both wearable device-based (e.g., HoloLens 2, Spatial AR, and Magic Leap) and app-based (e.g., ARCore and ARKit) schemes suffer from cross-platform issues. In particular, AR subscribers can only interact with others using the same type of equipment, the same operating system, the same AR SDK, or even the same dedicated app, factors which significantly hamper its large-scale applications. *(2) Inefficient multi-user communication*. All AR service subscribers rely on the cloud-based services to communicate with each other by broadcasting messages via established links with the cloud. Besides increased communication latency, intensive data transmission also consumes a large amount of bandwidth resources (especially wireless access networks), resulting in increased communication costs, as illustrated in Figure 1(a). Meanwhile, the use of unstable links means that there is no guarantee of persistent communication over a deteriorated wireless channel. *(3) Unsatisfactory offloading efficiency*. Complex AR computations are typically offloaded to the cloud for acceleration [12], [13]. However, existing motion-agnostic offloading approaches will cause high cloud computing costs while satisfying the User eXperience (UX). In detail, the lack of perception of user behavior causes inefficient computing, and thus unsatisfactory offloading efficiency. Moreover, this also in-

- *P. Ren, X. Qiao, Y. Huang, and J. Chen are with the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, P.R.China. E-mail: {renpei, qiaoxq, ykhuang, chjl}@bupt.edu.cn.*
- *L. Liu and C. Pu are with the College of Computing, Georgia Institute of Technology, Atlanta, GA 30332, USA. E-mail: {ling.liu, calton.pu}@cc.gatech.edu.*
- *S. Dustdar is with the Distributed Systems Group, Technische Universität Wien, 1040 Vienna, Austria. E-mail: dustdar@dsg.tuwien.ac.at.*

Manuscript received XX XX, 2020; revised XX XX, 2020. (Corresponding author: Xiuquan Qiao.) Digital Object Identifier no. XX.XXXX/TCC.2020.XXXXXXX

troduces additional communication, which in turn leads to high initialization time[1] [14], [15].



(a) Cloud-based multi-user mobile Web AR in current network. (b) Edge-assisted multi-user mobile Web AR in upcoming 5G Era.

Fig. 1. Overview of multi-user mobile Web AR architecture.

Mobile Web technology provides a lightweight, cross-platform service provisioning approach, greatly attracted the attention of World Wide Web Consortium (W3C), especially Immersive Web Working Group, to actively promote Web-based mobile AR [16], [17], and is thus envisioned as one of the promising supporting technologies [18]. Meanwhile, the emerging 5G networks appear to be able to provide the low latency communication and ubiquitous computing support for the development of multi-user mobile Web AR, thus show great potential in terms of addressing the above concerns [19].

Specifically, all users can easily interact with others via the mobile Web platform. And both the computations and interactions can be assisted by the edge server and nearby mobile devices for computation acceleration and message forwarding by using the mobile edge computing [20] and Device-to-Device (D2D) [21] communication techniques in 5G networks [22], as illustrated in Figure 1(b). Compared to the cloud-based AR service provisioning, each user will have multiple choices, whether for multi-user communication or computational assistance, in 5G networks. Employing these new techniques in multi-user mobile Web AR over the cloud, the edge server, and end devices is therefore still challenging, as follows:

- The limitation of existing multi-user communication approach is rooted in the constraint that message synchronization between all users relies solely on the cloud for forwarding. No edge-assisted[2] multi-user communication framework is currently practical for mobile Web AR in 5G networks. The first challenge therefore lies in designing an efficient decentralized communication scheme for message forwarding. One straightforward approach is to adopt a mesh topology among all edge nodes to broadcast messages; however, this obviously involves a waste of network resources.
- Furthermore, communication planning is also difficult. Since more paths (i.e., edge server-based and D2D-based forwarding) are available, we need to consider not only end-to-end latency but also mobile energy consumption and public spectrum resource occupation, which are two important concerns for users and Internet service providers. The system also needs to cope with dynamic changes in the network.
- Feature extraction and object recognition are two key components in mobile AR applications. Considering the limited computing capability of the mobile Web browsers, these complex computations are typically offloaded to the back-ends, while the end-users only perform lightweight computations locally, such as image pre-processing and object tracking [**?**]. Due to the loss of feature points caused by object tracking algorithm, intermittent key frame[3] selection is therefore adopted for error correction [15]. Tracking performance in AR is closely related to the user movement, but existing offloading schemes (more accurately, periodic and threshold-based key frame selection schemes) are all motion-agnostic, which suffer from the inefficient computation. An adaptive motion-aware key frame selection is thus necessary but also challenging since recognizing the user's movement behavior based on the isolated video frame is difficult.
- Moreover, the complex feature extraction and object recognition computations and additional communication also result in increased initialization time thus degrades the UX when adopting the offloading mechanism. Adopting lightweight algorithms on the edge server is a simple but practical solution, however, it will obviously face the problem of low tracking performance, which is a trade-off of speed and accuracy. Therefore, designing a D2D-assisted strategy to address the initialization problem is thus recommended by collaborating nearby mobile devices, although this is challenging.

To address these concerns, we introduce Edge AR X5, a framework that allows the collaboration of computing and communication resources to enable high-quality service provisioning for multi-user mobile Web AR systems. Edge AR X5 has an improved centralized communication mechanism that uses heterogeneous networking technologies for multi-user message synchronization, and allows for dynamic collaboration between the computing resources of an edge server and nearby devices. More specifically, we first propose a hybrid communication scheme for message synchronization in multi-user mobile Web AR applications. A heuristic communication planning algorithm is also proposed to generate communication solutions dynamically, based on the network context and the requirements of users and Internet service providers. For adaptive key frame selection, we employ a prediction-based motion-aware runtime scheduler, which selects the key frame based on the user's mobility to ensure accurate object tracking, and which also achieves computational cost savings as redundant computations are avoided. Furthermore, we introduce a D2D-based supplementary feature extraction mechanism to alleviate

1. The initialization time is defined as the duration the user waits from sending the first video frame to receiving the response.

2. The "edge" here refers to all the computing and communication resources between the data source (i.e., AR service subscriber) and the cloud, including edge servers and nearby mobile devices [20] in our Web-based multi-user mobile AR scenario. For simplicity, both the edge server and nearby mobile device are referred to as "edge nodes" when there is no confusion.

3. A key frame refers to a video frame that needs to be uploaded to the back-end server for feature extraction and object recognition.

the service initialization problem. In this way, Edge AR X5 provides an efficient multi-user mobile Web AR solution for both users and service providers.

Our experiments were conducted at Beijing University of Posts and Telecommunications (one of the places that have achieved full 5G coverage in Beijing, P.R. China) using three Huawei Mate 20 X (5G) smartphones. For performance evaluation purposes, we developed a multi-user AR application on the mobile Web platform, in which users can control their "augmented" 3D characters to interact with others in the activated "augmented world".

The results of our experiments indicate that Edge AR X5 enables not only dynamic communication planning but also efficient key frame selection. Our proposed multi-user communication mechanism BA-CPP gives improvements of 35.51%, 85.61%, and 26.19% in communication efficiency compared with the edge-based, D2D-based, and random communication solution, respectively. The proposed motion-aware key frame selection mechanism Mo-KFP also achieves efficiency improvements of 54.69% and 14.09% in feature extraction-tracking compared with the periodic and threshold-based selection mechanisms, respectively.

The contributions of this study summarized as follows:

- To the best of our knowledge, Edge AR X5 represents the first collaborative framework for mobile Web AR that enables multi-user interaction in 5G networks.
- We propose a heuristic multi-user communication planning mechanism called BA-CPP to ensure the interactions synchronization quality of AR subscribers.
- We provide a prediction-based approach called Mo-KFP for key frame selection based on information on the AR subscriber's movement. Moreover, a D2D-based composite solution is introduced to address the initialization problem.
- For demonstration purposes, we develop a multi-user mobile Web AR application called Panda Betrayal and examine the performance in real-world 5G networks operated by China Unicom.

The reminder of this paper is organized as follows. Section 2 reviews background and motivations on AR service provisioning. Section 3 describes the proposed Edge AR X5 system architecture. Section 4 gives details on multi-user communication solution design. Section 5 presents the edge-assisted collaborative computing design. Section 6 evaluates the performance of Edge AR X5. Section 7 outlines avenues for future research. Section 8 concludes the paper.

2 Background and Motivation

Although AR has been around since 1997 [23], following improvements in the computing capability of user devices and the development of mobile networking, this interactive computer vision technology has now undergone revolutionary changes in portability and mobility.

2.1 On-Device Mobile AR

One of the most important reasons for the phenomenal growth of on-device mobile AR is the continuous improvements in the computing capabilities of mobile devices [24]. For example, the launch of Pokémon GO in 2016 provides many people their first-hand experience with interactive multi-user AR on a mobile phone without additional equipment, which has attracted extensive interest in on-device mobile AR. According to reports [25], its downloads reached 500 millions in just two months after its first public release, involving more than 100 countries around the world. But from the technical perspective, Pokémon GO adopted GPS for user location tracking, which suffered from the low-precision positioning problem inherent from civilian grade GPS, in addition to other technical issues such as server overload, overtop mobile energy consumption, and so forth. To improve the user experience in the AR applications, vision-based solutions (including marker-based and markerless methods) are being adopted by more and more AR applications [18]. Since then, many other efforts have been made in this field, such as ARCore and ARKit, released by Google and Apple, respectively, which have significantly simplified the development of AR applications [26]. However, users need to download and install a specific application in advance to experience the AR service.

In contrast, the emergence of Web-based mobile AR[4] provides a flexible method of service provisioning. Users can enjoy the AR experience anytime, anywhere, by accessing a pre-defined URL. This Web-based implementation is envisioned as a promising solution for the large-scale application of AR technology. In addition to industry, standards organizations are also showing great enthusiasm for this Web-based mobile AR [17].

In this work, we aim to solve the problem of multi-user interaction in mobile Web AR applications. Based on the new features and technologies of 5G networks, we will optimize multi-user communication and collaborative computing modes to promote the on-device mobile AR.

2.2 Cloud-Assisted Mobile Web AR

Unlike self-contained implementations [24], a cloud-assisted solution provides not only computation acceleration but also opportunities for multi-user interaction by using the global anchor mechanism, and is thus seen as a promising approach for mobile AR, and particular mobile Web AR.

2.2.1 Cloud-Assisted Computation Offloading

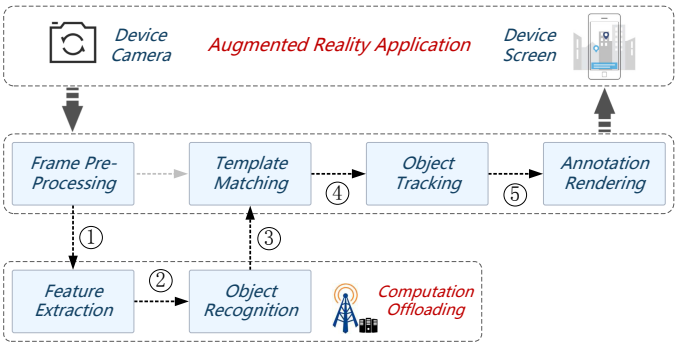


Fig. 2. Typical cloud-assisted mobile web AR pipeline.

4. Here we mainly focus on the visual-based AR implementation. Although several factors can be leveraged for object tracking, such as motion sensors, GPS, IMU, and compass, the visual-based solutions can also provide relevant semantic information at the same time.

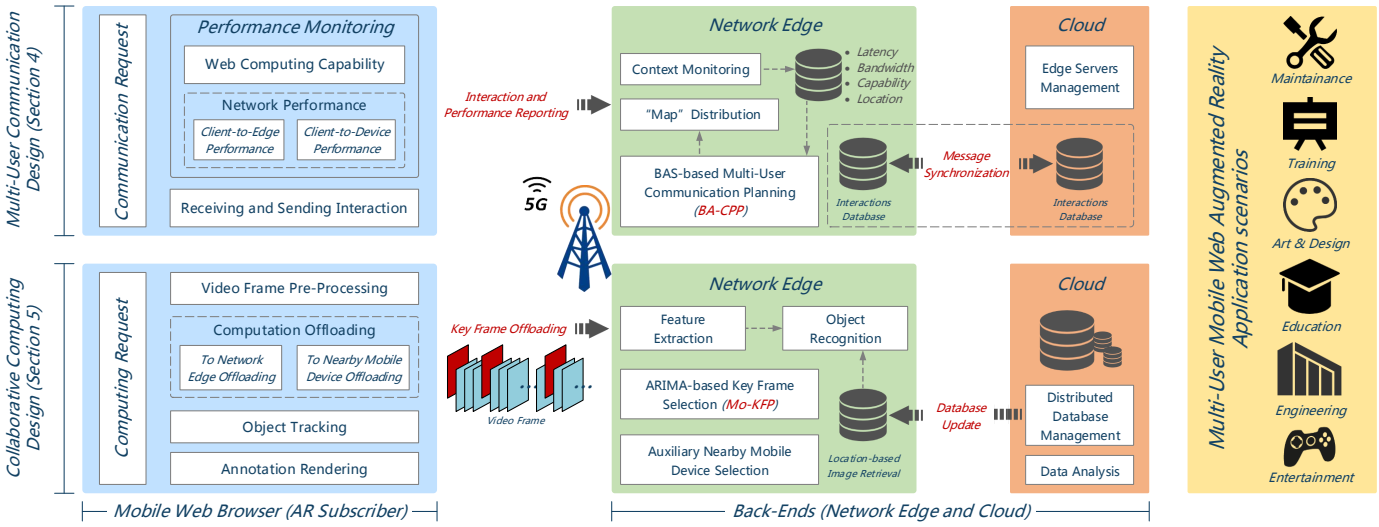


Fig. 3. Overview of the Edge AR X5 framework for multi-user mobile Web AR.

The outsourcing of computationally intensive tasks to the cloud (e.g., a remote cloud in current 4G LET networks) can usually provide a better AR experience [12] compared with the aforementioned on-device solutions. A typical cloud-assisted mobile Web AR pipeline is illustrated in Figure 2. Captured video frames are first processed (e.g., by downscaling and graying) on the mobile user's Web browser, and are then transmitted to the cloud for feature extraction and object recognition. Following this, the recognized result will be returned to the client to perform template matching for position and orientation estimation, which then will be used as the initial input for object tracking. Finally, the "augmented" contents are rendered and displayed to AR subscribers via the mobile Web browser.

To avoid frequent feature extraction and object recognition, which are time-consuming processes, object tracking is usually adopted for frame-by-frame pose estimation. Thus, the client only needs to intermittently select video frames to send to the cloud for further processing (i.e., tracking error correction). Key frame selection is therefore one of the most basic components of cloud-assisted mobile Web AR; however, unlike traditional mechanisms [27], [28] that need to analyze all the video frames in advance, AR applications require online key frame selection.

Current mainstream key frame selection approaches in mobile AR applications generally select video frames either periodically or based on a pre-defined threshold (e.g., the number of trackable key points) [15]. These straightforward approaches are obviously inefficient, and thus cause UX degradation, and we therefore propose a prediction-based movement-aware key frame selection approach to improve computational efficiency.

2.2.2 Cloud-Assisted Multi-User Interaction

In addition to the offloading of AR computations, the cloud can also assist in multi-user interaction. Here, we take WebARonARCore as an example, in which AR subscribers can use the anchor to interact with others in the same "augmented world", and all anchor information is stored in the cloud [29]. When other users are at a specific location, the augmented contents will be activated; the users can then interact with these contents easily, and all of these interactions can also be synchronized to the cloud. However, the problem of unstable communication between mobile users and cloud cannot be ignored.

2.3 Web-Based Mobile AR in the 5G Era

The emerging 5G networks are promising communication solutions for providing a better UX for mobile AR applications, and especially mobile Web AR [30]. Specifically, their low latency and high bandwidth communication features can achieve more efficient data transmission [31]. Moreover, the introduction of the edge servers enables AR service provisioning (e.g., location-based image retrieval and computing service) that is closer to users [32], and the D2D communication technology provides more choices for message synchronization [33] and collaborative computing.

In this work, we make full use of the features of 5G networks, where mobile devices not only enjoy the advantages of high-bandwidth and low-latency mobile network performance, but also leverage the mobile edge computing mechanisms and D2D communication technology to further reduce the consumption of public spectrum resource, and to enable mobile Web AR technology, one of the main contributions of our paper.

3 Overview of Edge AR X5

The architecture of the Edge AR X5 framework is illustrated in Figure 3. In this scheme, mobile Web AR subscribers collaborate with the back-ends to facilitate multi-user interaction. We will discuss this aspect separately from the communication and computing schemes.

3.1 Multi-User Communication Phase

Unique to this Web-based multi-user mobile AR application is the fact that subscribers can easily enjoy an AR experience with others simply by accessing a pre-defined URL. When they enter the agreed room, the communication request is invoked, and all the mobile devices are then connected via WebRTC communication links, based on the D2D technique.

Their interactions will be passed to others via a WebRTC data channel API [34]. At this point, all AR subscribers have entered the same "augmented world".

Potentially, each client can connect to all other mobile devices and the edge server, which form a mesh network topology. To optimize message synchronization, a communication planning thread is maintained on the edge server to periodically specify the message forwarding paths, based on the observed contextual information, and the new communication "map" then will be distributed to all subscribers. For this purpose, each Web client maintains a performance monitoring thread to periodically report network connectivity (including latency and bandwidth to all edge nodes) and computing capability. The user's location information is also collected for location-based service provisioning, as discussed below. Moreover, the cloud is responsible for the management of edge servers, such as resource allocation, service deployment, data analysis, etc. [35]

Once the communication links between edge nodes are established, all users can start sharing their interactions. Meanwhile, these interactions (i.e., operations in the virtual environment) will be recorded in the edge server in a chronological order and synchronized to the cloud server to ensure data persistence and system reliability as illustrated in Figure 3. In detail, when a new user joins, they can directly obtain the latest status of the virtual model from the interactive database. If the user leaves or the D2D link is disconnected, all the involved mobile devices will be temporarily taken over by the edge server and the communication "map" will be regenerated. Benefiting from the synchronization of the interactions between the edge server and the cloud server, even if the service deployed on the edge temporarily fails, its continuity can be guaranteed.

3.2 Collaborative Computing Phase

When the user activates a webcam on a smartphone and captures the first video frame, a computing request is then generated. Typically, image pre-processing, object tracking, and annotation rendering, which have a low computational burden, are carried out in the mobile Web browser, while feature extraction is performed using the computing resources of the edge server.

3.2.1 Edge-Assisted Service Provisioning

Edge servers are typically deployed as a supplement to the remote cloud in terms of computing and storage capability.

- The pre-processed frames are selected based on our proposed motion-aware runtime scheduler Mo-KFP, and they are then transmitted to the edge server for feature extraction. Object recognition (i.e., image retrieval) is then performed based on the obtained feature points. Finally, the recognition result is sent as feedback to the client for further processing.
- Besides edge-assisted computing service, the edge servers can also provide location-based service. In the edge cloud service, the VLAD[5] extracted features of all reference images are organized into location lists based on their location information, which is represented by the ID of the edge server to which these images will be served. The edge server therefore only needs to maintain a part of the retrieval database to help reduce the search space during the retrieval process, improving the real-time recognition efficiency and accuracy. Management of this distributed database is carried out in the cloud server.

3.2.2 Have You Asked Your Neighbors

Due to the problem of initialization performance degradation caused by computation offloading, the client needs to leverage the computing resources of nearby mobile devices for lightweight feature extraction, in order to optimize the initialization process.

The first video frame is transmitted to the edge server for feature extraction using not only complex SIFT algorithm [39] but also lightweight ORB algorithm [40]. Before receiving the SIFT-based response to the first frame from the edge server, subsequent video frames are also selected and intermittently transmitted to a specific nearby mobile device, for lightweight feature extraction after receiving the ORB-based response from the edge server.

Note that this D2D-assisted scheme will significantly reduce the computational cost of the back-ends, and the selection of the nearby device is performed on the edge server based on the computing capability of the mobile devices, which is collected during the performance monitoring.

3.3 Edge AR X5 Pipeline

For clarity, we describe the processing pipeline of Edge AR X5 here. The request is processed using the following steps:

- AR subscribers request service by accessing a predefined URL using their mobile Web browser.
  /* **Stage 1: Establishment of communication** */
- All participants periodically (for example, every 15 seconds) activate the performance monitoring module, and the results will be reported to the edge server only when the detected change in Web computing capability or network communication performance exceeds a certain threshold[6].
- Once the edge server receives the updated context information, the communication planning module will be activated and generates a new "map" then distributes the results to all AR service subscribers.

5. To achieve efficient retrieval, all the local features of the reference images obtained through SIFT or ORB algorithms will be encoded to the k-dimensional Vector of Locally Aggregated Descriptors (VLAD) [36] features in advance. In addition, Bag of Features (BoF) [37] and Fisher Vector (FV) [38] algorithms can also be used for local feature coding. We adopt VLAD considering its low computational burden compared to BoF and the smaller codebook and higher accuracy compared to FV. Note that the codebook is first learned using the k-means algorithm and the retrieval accuracy is related to the value of k.

6. In a dynamic environment where users move around, the network performance between edge nodes will obviously change dynamically. But each user still decides whether to upload the detected results to the edge server according to the changes in network performance. The edge server invokes the planning algorithm to generate a new communication solution after receives the updated context information.

Since all mobile devices have a Socket.IO-based link with the edge server, it is thus easy to broadcast the communication configuration messages to all users.

/* **Stage 2.1: Edge-assisted service provisioning** */

- The client then activates the webcam, selects key frames using the proposed Mo-KFP approach, and sends them to the edge server for feature extraction and object recognition.

/* **Stage 2.2: D2D-based initialization** */

- Meanwhile, the previous video frames are processed on nearby mobile devices (via D2D communication channels) for lightweight feature extraction.
- Finally, the client continues the subsequent AR processes based on the feature points obtained.

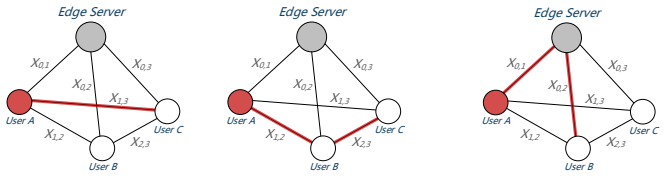
An edge server is overloaded when it experiences heavy traffics. When the edge server is overloaded, the feature extraction algorithm SIFT will be temporarily replaced by ORB to ensure that basic services will not be interrupted, for the following two reasons. First, compared with SIFT, the ORB algorithm is more efficient and requires fewer computing resources, which can alleviate the computational burden of feature extraction on the edge server. Second, the ORB-based features take up less storage space (the SIFT descriptor is 128 dimensions, while it is only 32 dimensions for ORB), which will also effectively alleviate the pressure of data transmission on the edge network. However, since fewer feature points are extracted, the potential problem caused by replacing to ORB algorithm is the degradation of tracking performance, which is a trade-off between accuracy and efficiency.

4 Multi-User Communication Design

The establishment of communication forms a basis for AR subscribers to interact with others. An efficient communication "map" is therefore extremely important. In this section, we analyze the necessity of communication planning and formulate the planning problem in a heterogeneous communication environment. Then the proposed adaptive communication planning algorithm is presented.

4.1 Why Communication Planning is Needed

One of the most important reasons for adopting dynamic communication planning is to cope with frequent changes in the mobile network, in order to provide optimal message forwarding options. In particular, subscribers are sensitive to interactive latency and mobile energy consumption, and different communication options can have a large impact on application performance, and thus UX.



(a) D2D-based comm. (b) Centralized comm.

Fig. 4. Two methods of multi-user communication in 5G era.

Mobile users now have more ways to forward their interactive information in 5G networks, which is a completely different approach from the cloud-based forwarding mechanism in current 4G LTE networks. More specifically, the user's interactions regarding virtual content in the "augmented world" can be synchronized to all participants in the following two ways, as illustrated in Figure 4.

- D2D-based forwarding: the user's interactions can be directly or indirectly forwarded to other participants via the established D2D communication links, in one or more forwarding hops.
- Centralized forwarding: all users have also established connections with the edge server, the interactions can be forwarded by the network edge server.

The communication "map" is finally obtained by selectively combining these two communication options.

Note that the edge server provides opportunities for message forwarding, but is likely to consume more public spectrum resources than the D2D-based communication in 5G networks. Since each different communication solution will result in different latency and mobile energy consumption (from the mobile user's perspective) as well as a different occupation of public spectrum resources (from Internet service provider's perspective), our aim is to find the lowest cost communication "map", in order to jointly optimize the above metrics and improve the communication efficiency.

4.2 Formulation of the Planning Problem

As mentioned above, our main areas of focus in this study are: (1) communication latency; (2) mobile energy consumption; and (3) public network spectrum occupancy. Given a hybrid mesh topology with one edge server and n mobile users, the problem can be defined as follows:

$$\min_{\mathbb{X}} \quad F(\mathbf{X}) = (T_{\text{global}}(\mathbf{X}), E_{\text{mobi}}(\mathbf{X}), S_{\text{edge}}(\mathbf{X})). \tag{1}$$

Here, we denote by $\mathbf{X}$ the obtained global communication "map", $X_{i,j} \in \{0,1\}$, and $i \in [0, n-1]$, $j \in [1, n]$. More specifically, $X_{i,j} = 1$ indicates that nodes i and j are directly connected, and vice versa. We use subscript 0 to represent the edge server, and thus $X_{0,j}$ indicates the connection status between the given edge server and mobile device j. Remarkably, $X_{i,j}$ will be 0 if $i = j$. And $\mathbb{X}$ represents all the potential communication solutions.

$$\mathbf{X} = \{\underbrace{X_{0,1}, X_{0,2}, ..., X_{0,n}}_{\text{Centralized Connections}}, \underbrace{X_{1,2}, X_{1,3}, ..., X_{i,j}, ..., X_{n-1,n}}_{\text{D2D Connections}}\}$$

The motivation for communication planning is to jointly optimize both the communication experience and efficiency, based on the observed (one-way) network latency $\mathbf{L}$ and bandwidth $\mathbf{B}$ during the subscribers' interactions. In more detail, $\mathbf{L}$ records the communication latency of each link, which has size $n(n+1)/2$. Considering the uplink and downlink bandwidth between the mobile device and the edge server, the size of $\mathbf{B}$ is therefore $n(n-1)/2 + 2n$. More details on these optimization objectives are discussed below.

4.2.1 *Objective 1: Communication Latency*

Although message forwarding in the network is carried out in parallel, we cannot directly regard the longest path of the undirected graph constructed by the obtained communication "map" as the time required to synchronize the user's

interactions with all other participants. Here, we present a simplified example for illustration purposes (see Figure 4) with two communication modes (i.e., $X_{0,1}+X_{0,2}+X_{0,3}$ and $X_{0,1}+X_{0,3}+X_{1,2}$) for which the message synchronization time is l_1+l_3 ($l_2<l_3$ and $l_1+l_3>l_4$).

The network latencies of the links $X_{0,1}$, $X_{0,2}$, $X_{0,3}$, and $X_{1,2}$ are l_1, l_2, l_3, and l_4, respectively. Assuming $l_2<l_4$, in the second communication scenario, user B needs to wait longer, i.e., l_4-l_2, to receive the message sent by user A (here, we consider only the communication latency).

The addition of a higher-latency link will greatly degrade the user's communication experience. From a global perspective, the communication latency in this problem is therefore defined as

$$T_{\text{global}}(\mathbf{X})=\sum_{i=0}^{n-1}\sum_{j=1}^{n}L_{i,j}\cdot X_{i,j}\quad(L_{i,j}\in\mathbf{L}). \tag{2}$$

4.2.2 Objective 2: Mobile Energy Consumption

Different communication modes will give rise to different energy consumption for mobile devices. For a given communication decision $\mathbf{X}$, the mobile energy consumption is therefore given by

$$E_{\text{mobi}}(\mathbf{X})=\sum_{j=1}^{n}E_{0,j}^{\text{5G}}\cdot X_{0,j}+\sum_{i=1}^{n-1}\sum_{j=1}^{n}E_{i,j}^{\text{D2D}}\cdot X_{i,j}. \tag{3}$$

Specifically, each centralized communication link is not only responsible for sending (upload) an AR subscriber's interactions but also for receiving (download) messages from the other $n-1$ participants, the per link communication energy consumption is therefore defined as $E_{i,j}^{\text{5G}}=P_{\text{5G}}^{\text{U}}+(n-1)\cdot P_{\text{5G}}^{\text{D}}$, here P_{5G}^{U} and P_{5G}^{D} are the energy consumption of upload and download respectively; while a D2D-based communication link only needs to transmit (including send and receive) messages between two mobile devices, the energy consumption for communication is thus $E_{i,j}^{\text{D2D}}=2\cdot P_{\text{D2D}}$. In detail, we denote by $P=\varphi\cdot r+\theta$ the transmission power, where r (i.e., $B_{i,j}\in\mathbf{B}$) is the available data rate. The parameters φ and θ for P in 5G mobile networks are 65/6.5 (uplink/downlink) mW/Mbps and 11475.97 mW, and 283.17 mW/Mbps and 132.86 mW in D2D networks, respectively [41].

4.2.3 Objective 3: Public Network Spectrum Occupancy

D2D-based communication can effectively reduce the occupation of public network spectrum resources, which is important for Internet service providers. In this problem, we associate the network spectrum occupancy with the number of communication links established between mobile devices and the edge server. It is therefore defined as follows:

$$S_{\text{edge}}(\mathbf{X})=\sum_{j=1}^{n}X_{0,j}. \tag{4}$$

In addition to synchronizing messages between AR participants, all interactions also need to be saved to the edge and cloud servers, thereby allowing newly added users to be updated on the latest state of the "augmented world". Therefore, we also need to guarantee $S_{\text{edge}}(\mathbf{X})\geq 1$, that is, at least one mobile user connects to the edge server.

4.3 Multi-User Communication Planning

In this part, we first analyze this planning problem, and then present our proposed multi-user communication planning algorithm BA-CPP in more detail.

4.3.1 A Heuristic Planning Approach is Needed

Theoretically, for a mesh topology network consisting of m nodes, there will be $m(m-1)/2$ communication links in total. Each link has two statuses, selected and ignored, meaning that the size of solution space of the communication "map" therefore will be $2^{m(m-1)/2}$. To obtain the optimal solution, the time complexity is thus $\mathcal{O}(a^m)$. Given the huge search space, a heuristic algorithm is therefore recommended for addressing these challenges.

Meanwhile,a lightweight approach is also necessary to quickly update the communication "map" to cope with dynamic changes in the mobile network. Although the improvement in the speed of decision making will result in a loss of decision accuracy, this is a trade-off that must be made. In this problem, the communication "map" is periodically updated, and although the optimal result sometimes cannot be produced, this approach can still provide subscribers with a good communication experience overall.

Based on the above discussion, we adopt a fast communication planning method based on the Beetle Antennae Search (BAS) algorithm [42] to provide AR subscribers with a high-quality communication service, as described below.

4.3.2 TOPSIS-Based Evaluation Approach

During the communication "map" searching phase, we first need to evaluate the quality of the obtained solution. Here we adopt the Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) [43] for evaluation purposes. More specifically, by selecting the positive ideal point $S^+=\{S_{\max}^1,S_{\max}^2,S_{\max}^3\}$ and the negative ideal point $S^-=\{S_{\min}^1,S_{\min}^2,S_{\min}^3\}$, the quality of the obtained "map" is evaluated according to its distance from the positive ideal point. The problem can therefore be formulated as

$$\begin{aligned}\min_{\mathbb{X}}\quad F'(\mathbf{X})=&\alpha\cdot\frac{T_{\text{global}}(\mathbf{X})-S_{\max}^1}{S_{\min}^1-S_{\max}^1}+\beta\cdot\frac{E_{\text{mobi}}(\mathbf{X})-S_{\max}^2}{S_{\min}^2-S_{\max}^2}\\&+\gamma\cdot\frac{S_{\text{edge}}(\mathbf{X})-S_{\max}^3}{S_{\min}^3-S_{\max}^3}.\end{aligned} \tag{5}$$

Here we denote by $S_{\max}^1$ and $S_{\min}^1$ the shortest and longest times required to synchronize the user's interaction, respectively, and by $S_{\max}^2$ and $S_{\min}^2$ the minimum and maximum mobile energies consumed for communication, respectively. These four ideal values can be obtained by applying the Minimum Spanning Tree (MST) algorithm. When only one mobile device is connected to the edge server, the spectrum occupation is $S_{\max}^3=1$, while in the case where all mobile devices are connected to the edge server, $S_{\max}^3=n$, that is, the edge server provides the forwarding service for all AR subscribers. The weighting factors α, β, and γ are used to balance the importance of communication latency, mobile energy consumption, and network spectrum occupancy in this problem, respectively.

4.3.3 BA-CPP: BAS-Based Communication Planning

Due to the particularities of our problem, the newly developed BAS algorithm is appropriate. This was first proposed in 2018 and can achieve faster searching than other heuristic algorithms (e.g., particle swarm optimization [44] and genetic algorithm [45]) for optimization problems.

The BAS algorithm was inspired by the searching and detecting behavior of longhorn beetles. A beetle explores an unknown environment by randomly searching, and its two antennae are defined as $\boldsymbol{x}_r^t = \boldsymbol{x}^t + d^t \cdot \vec{\boldsymbol{b}}$ (right antenna) and $\boldsymbol{x}_l^t = \boldsymbol{x}^t - d^t \cdot \vec{\boldsymbol{b}}$ (left antenna). In detail, $\boldsymbol{x}^t$ denotes the current position of the beetle, d^t the sensing length, and $\vec{\boldsymbol{b}}$ the randomly generated searching direction. And in the detecting phase, the beetle updates its position based on the sensed odor concentration (i.e., fitness value) using the antennae. The detailed odor detection behavior is given by $\boldsymbol{x}^t = \boldsymbol{x}^{t-1} + \delta^t \cdot \vec{\boldsymbol{b}} \cdot \text{sign}(f(\boldsymbol{x}_r) - f(\boldsymbol{x}_l))$, where δ^t is the searching step size and $f(\cdot)$ is the fitness function. Based on the discussion above, we propose the multi-user communication planning approach described by Algorithm 1.

Algorithm 1 BAS-Based Communication Planning

Input:
Network features: **B** (bandwidth) and **L** (latency);
Ideal points: S^+ (positive) and S^- (negative);
BAS parameters: P' (penalty factor), δ, and N.
Output:
Optimal multi-user communication solution: $\boldsymbol{X}$.

Initialize the position $\boldsymbol{x}^t$ and direction $\vec{\boldsymbol{b}}$ of the beetle
for $epoch$ = 1 to N **do**
  Environment search with antennae: $\boldsymbol{x}_r^t$ and $\boldsymbol{x}_l^t$
  /* Restrictions on the detecting phase */
  if $\sum X_{0,j} = 0$ **then** $\text{Cond}_1 \leftarrow 1$ ▷ Restriction (1)
  if $\sum X_{i,j} \neq n-1$ **then** $\text{Cond}_2 \leftarrow 1$ ▷ Restriction (2)
  for e = 1 to n **do**
    Set $X_{0,e}$ and $X_{e,j}$ to 0 then obtain X'
    if $T_{\text{global}}(\mathbf{X}) = \sum L_{i,j} \cdot X'_{i,j}$ **then**
      Set $\text{Cond}_3 \leftarrow 1$ and Break ▷ Restriction (3)
    end if
  end for
  /* Communication "map" punishment */
  if $\text{Cond}_1 + \text{Cond}_2 + \text{Cond}_3 > 0$ **then**
    $F''(X) = F'(X) + P'$
  end if
  Update beetle's position $\boldsymbol{x}^{t+1}$ with random direction
end for
return $\arg\min F''(x)$

Our proposed approach BA-CPP further improved the BAS-based heuristic approach for communication planning by combining multi-user interaction prior knowledge, that is, the Restriction 1, 2, and 3 in Algorithm 1, through mobile Web AR-based adaption for interaction synchronization. This enhancement improves the algorithm efficiency by guaranteeing that all subscribers are connected by the shortest path. Based on our experiments, the straightforward use of BAS will lead to an unsatisfactory problem solving process because it is more easily being trapped into local optimum and produce unsatisfactory results because all users tend to forward their interactions using edge server, thus forming a star topology, which ignores the advantages of D2D communication.

We summarize the restrictions introduced as follows:

- Restriction 1. At least one mobile device needs to be connected to the edge server to upload all AR subscribers' interactions for synchronization.
- Restriction 2. In the case of n communication nodes (including the edge server), the obtained communication "map" needs $n-1$ communication links.
- Restriction 3. The generated communication solution must avoid loop to ensure that all edge nodes can communicate with each other based on Restriction 2.

Finally, after a certain number of iterations, the beetle will return the search result (i.e., its current position), and the communication "map" is then distributed to all users.

4.3.4 Grouping-Based Path Planning

A unique feature of this Web-based mobile AR solution is that subscribers no longer need to download the specific apps in advance to experience the augmented virtual services, that is, the Web technology eliminates the performance differences in AR interaction mechanisms due to heterogeneous mobile devices, users can enjoy the AR experience anytime and anywhere by simply using a predefined URL, which holds the potential to promote the application of mobile AR on a large scale.

Although the proposed communication planning algorithm BA-CPP promises efficient problem-solving capability, when the number of subscribers increases excessively, the path planning remains a challenging problem. For example, when there are 10 edge nodes requesting communication, a total of 45 communication paths are available; then when the edge nodes are increased to 30, the number of communication paths will increase to 465. The increase in the dimension of the solution space reflects the complexity of the problem, making efficient optimization more challenging. Not only does the time required for the problem-solving increase, but also it is more easily being trapped into local optimum.

Algorithm 2 Grouping-Based Path Planning

Input:
Network features: **B** (bandwidth) and **L** (latency);
Ideal points: S^+ (positive) and S^- (negative);
BAS parameters: P' (penalty factor), δ, and N;
Mobile user collection **M** and location information **P**.
Output:
Optimal multi-user communication solution: $\boldsymbol{X}$.

Mobile user grouping using cluster analysis
for i = 1 to K **do** ▷ Parallel processing
  $\boldsymbol{X}_i = \text{BA-CPP}(G_i, \mathbf{B}_i, \mathbf{L}_i, S^+, S^-, P', \delta, N)$
end for
return $\boldsymbol{X} = \{\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_K\}$

To address the challenge of efficient large-scale communication planning, we adopt a grouping-based path planning mechanism, as described by Algorithm 2. n mobile service subscribers are divided into K communication groups, $G = \{G_1, G_2, \ldots, G_K\}$, based on their location information using cluster analysis. All groups use the BA-CPP algorithm

in parallel to obtain their in-group communication "map", and all interactions are forwarded by the edge server between groups. Remarkably, the granularity of the grouping depends on the cost of network spectrum resources and the service provider's pre-defined requirements for communication path update frequency. The more groups, the more spectrum resources are occupied (for K groups, the system network spectrum occupancy $S^3_{\max}$ is K), but the faster the communication planning process for each group.

An alternative is to orchestrate the computing resources adaptively in order to support highly concurrent user requests, which has been widely studied in both cloud computing scenarios [46] and mobile edge computing scenarios [32]. However, such cloud based solutions are complementary and will be one item in our further research agenda.

5 Collaborative Computing Design

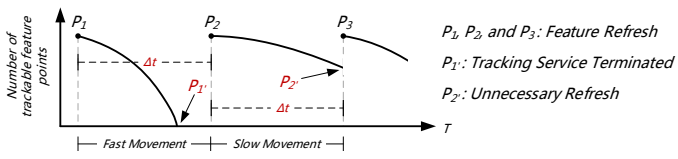
Both the edge server and the nearby mobile devices provide opportunities for collaborative computing. In this section, we propose a motion-aware runtime scheduler for adaptive key frame selection. Moreover, by coordinating local and nearby computing resources, a supplementary feature extraction mechanism is also designed.

5.1 Collaboration with Edge Server: Key Frame Selection

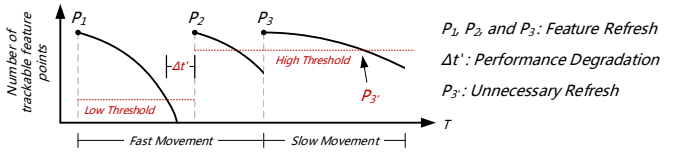
Object tracking through continuous feature extraction will face enormous challenges for its practical application. By combining feature extraction and lightweight object tracking algorithms together, it will therefore be a compromise but practical solution, which currently has been adopted by many mobile Web AR implementations.

5.1.1 Key Frame is Not Key

The intermittent feature extraction not only provides a means of tracking error correction (since the number of traceable feature points will decrease due to the user's movement during the object tracking process) but also achieves savings in terms of computational cost (since it avoids massive feature extraction requests, which are typical computation-intensive tasks). The question then arises as to which frames should be selected as key frames.



(a) Periodic key frame selection scheme.



(b) Threshold-based key frame selection scheme.

Fig. 5. Mainstream approaches for key frame selection.

Note that in AR applications, the homography matrix, that is, the transformation relation between two frames, can be obtained by analyzing the corresponding matched feature point pairs, which is used to update the virtual model. The more matching point pairs, the more accurate the homography matrix is, and thus the more exact the virtual model can be updated, thereby improving the UX. Therefore, it would be better if enough feature points can be obtained. For simplicity but without losing accuracy, we use the number of detected feature points to assess the service quality. And the threshold here refers to the minimum number of feature points that can be tracked by the K–L optical flow algorithm on the mobile device which is pre-defined by the application service provider according to the characteristics of reference images and specific service requirements. When the number of traceable feature points is less than the pre-defined threshold, the feature extraction process will be activated again.

Current mobile Web AR implementations select the key frames periodically (i.e., at regular intervals, for example, the first video frame per second) or based on the number of trackable key points (i.e., by setting a specific threshold), and these are the two most common approaches for key frame selection. However, the following reasons prompted us to develop a new key frame selection mechanism.

Both of the aforementioned approaches suffer from inherent weaknesses, as illustrated in Figure 5. Specifically, the first method is a periodic key frame selection mechanism, and obviously it cannot meet the requirements for timely correction of tracking errors caused by dynamic changes in the user's Field of View (FoV). In cases where the user moves fast, the number of feature points that can be tracked will decrease sharply; this means that accurate object tracking cannot be guaranteed, thereby reducing the experience of mobile Web AR subscribers (or the service continuity perceived by users). In contrast, when the user moves slowly or stays still, this mechanism will cause redundant computation. Since the current number of trackable feature points is sufficient to provide accurate tracking, the feature extraction request is unnecessary, although it is still activated (this key frame is therefore not a real "key" frame). In the second type, the threshold-based key frame selection mechanism, unpredictable user movements pose a dilemma in terms of threshold setting. If the user moves fast but the threshold (i.e., the minimum number of feature points needed to perform accurate object tracking) is low, then when the number of trackable feature points reaches the threshold, the video frame at that moment will be selected as a key frame and transmitted to the back-end for feature extraction. From this point on, however, it will be impossible to provide stable object tracking, and the service will even be interrupted, which will greatly decrease the UX. On the other hand, when the threshold is high but the user moves slowly, performing feature extraction will not be necessary, resulting in a waste of computing resources. This is because the current number of trackable feature points are sufficient to guarantee the quality of the tracking service. One naive approach is to set the threshold higher when the user moves faster, and vice versa; however, identifying the user's state of motion is not easy, since each video frame is isolated.

5.1.2 Motion-Aware Runtime Scheduler Mo-KFP

Based on the discussion above, a context-aware key frame selection mechanism is therefore recommended that not only guarantees the quality of the object tracking service (from the user's perspective) but also avoids the waste of computing resources (from the application service provider's perspective).

More specifically, the selection of key frames is based on an analysis of the user's continuous movement. By monitoring the number of feature points that can be tracked in real time, we can therefore indirectly obtain information on the user's movement. We then use the pre-defined threshold as the baseline, combined with the time required for back-end processing (including the time required for round-trip data transmission), to infer which video frame should be selected as the real "key" frame.

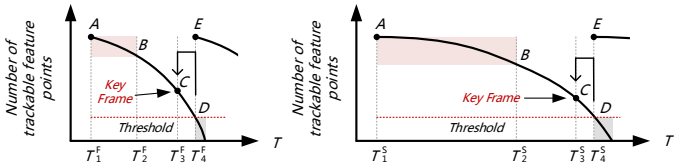


Fig. 6. Motion-aware key frame selection approach Mo-KFP.

A simplified example is illustrated in Figure 6. Here we present the details of the key frame selection mechanism in two scenarios where the user's movement is different.

1. At time T_1^{F} and T_1^{S} (we use superscripts F and S to indicate the two states of the user's movement, that is, fast and slow, respectively), the user receives the extracted feature points, and then starts to perform object tracking locally.
2. Within a given period of time (e.g., from T_1 to T_2), we first record information on the user's movement (including the frame timestamp and the corresponding number of trackable feature points, using the Lucas-kanade (L–K) optical flow algorithm [47]).
3. The edge server analyzes the movement records, and then the key frame prediction model (i.e., the motion-aware runtime scheduler) is obtained.
4. Following this, we can therefore predict when the number of trackable feature points will decrease to the pre-defined threshold given the current state of movement (i.e., T_4 in Figure 6).
5. Finally, based on the time required for cloud processing (i.e., $T_4 - T_3$), the key frame can be selected.

Ideally, when the number of feature points that can be tracked decreases to the threshold, the client will receive the extracted features of the newly selected key frame.

Regardless of whether the user moves fast or slowly, there are still sufficient feature points to support accurate object tracking using our proposed key frame selection mechanism during the period in which the user has requested feature extraction but has not yet received the result (i.e., the period from T_3 to T_4). In the case of fast user movement, our proposed scheduler can transmit video frames to the back-end server for feature extraction in a timely manner, thus satisfying the user's requirements for accurate object tracking. When the user moves slowly, it can also reduce the frequency of feature extraction accordingly, which can improve the utilization of system resources.

However, predicting user movement is difficult. More specifically, model-based approaches need to specify the movement pattern in advance for curve fitting, such as linear and polynomial regressions. While for irregular movement by the user, it is impossible to define a specific model that is suitable for all cases.

For addressing this time series problem, there are currently two mainstream solutions worth considering, that is, statistical methods and machine learning-based methods. The latter solution, e.g., Recurrent Neural Network (RNN), has achieved remarkable prediction accuracy improvements in recent years, but it cannot be ignored that these methods require large amounts of data for neural network training; meanwhile, due to their high computational complexity, running RNNs on a device without GPU will take longer and thus cannot meet the fast prediction of key frames in AR applications. In contrast, Autoregressive Integrated Moving Average (ARIMA) [48] as a linear regression-based method, is more suitable for single-step or short-term prediction and is a statistical method with satisfactory performance [49]. In the meantime, the automatic optimal model search will also simplifies the process of key frame selection, and thus can adaptively deal with various types of user movement. And its fast predictions (tens of milliseconds) enable online selection to be used. Considering the above reasons, here we adopt ARIMA for the key frame selection.

Moreover, based on our observations, here we adopt a compound prediction mechanism for key frame selection:

- When the number of trackable feature points predicted by Mo-KFP is constant, this indicates that the user tends to be stable, then we will predict again by increasing the number of user's movement records.
- When the predicted key frame is farther away from the current video frame, we use the logistic regression result as the final predicted result, in order to avoid prediction error. Our experiments indicate that logistic regression performs better than linear or polynomial regression (see Section 6).
- When the predicted key frame is closer to the current frame (i.e., the last frame in the records uploaded by the user), considering the time required for data analysis and transmission, we adopt the video frame at the moment the user receives the response as the key frame for feature extraction.

5.2 Collaboration With Nearby Mobile Devices: Service Initialization Optimization

In addition to edge server assisted collaborative computing, mobile devices near to the user can also provide computational assistance via the D2D communication technique in 5G networks, thus making collaboration between various distributed computing resources more flexible. In this section, we describe the collaboration between the computing resources of the user device and nearby mobile devices to address the AR service initialization problem.

5.2.1 Occurrence of the Initialization Problem

More complex feature extraction algorithms can always achieve more accurate object recognition and tracking; how-

ever, they will incur longer processing times (for clarity, we briefly compare the performance of several feature extraction algorithms in Table 1, all images are pre-processed to have equal length and width), thus causing service initialization problems. In this case, the user needs to wait for a significant amount of time to receive the processing result after capturing the first frame, which degrades the UX.

TABLE 1
Comparison between different feature extraction algorithms

Feature Extraction Algorithms [50]	Features Detected			Processing Time (ms)		
	300×	500×	800×	300×	500×	800×
SIFT (2004)	893	1415	1994	24.63	56.07	142.13
SURF (2008)	1354	2156	3526	53.86	145.96	337.69
ORB (2011)	461	469	471	5.17	8.49	14.91
BRISK (2011)	1566	2683	3989	11.67	17.93	24.78
AKAZE (2013)	436	829	1451	15.76	38.51	97.27

One straightforward approach is to reduce the accuracy requirements of the object recognition and tracking service in exchange for a faster service response, thereby shortening the initialization time for the AR service. Although lightweight feature extraction algorithms have tremendous advantages in terms of processing time (i.e., computational complexity), their drawbacks are also obvious. Fewer feature points makes it difficult to provide a high-quality object tracking service, thereby leading to frequent feature extraction, as illustrated in Figure 7. Due to the serial execution of feature extraction and object tracking, regardless of whether the user performs feature extraction locally or on the back-end servers, the problem of object tracking performance degradation during the time period from T_2 to T_3 and from T_4 to T_5 cannot be avoided.

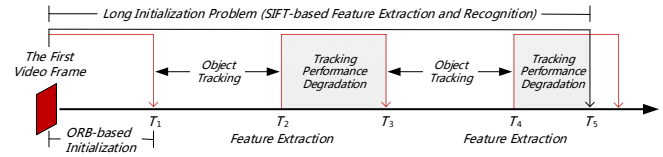


Fig. 7. Feature extraction and tracking during initialization.

5.2.2 D2D-Enhanced Feature Extraction Mechanism

To avoid service initialization performance degradation as mentioned above, we propose a composite feature extraction and object tracking mechanism based on orchestrating the computing resources of the user device and nearby mobile devices using the D2D communication technique in 5G networks. In this scheme, the user device performs object tracking based on the L–K optical flow algorithm, and the nearby devices assist the client in feature extraction, thereby parallelizing the AR service provisioning and thus optimizing the initialization performance.

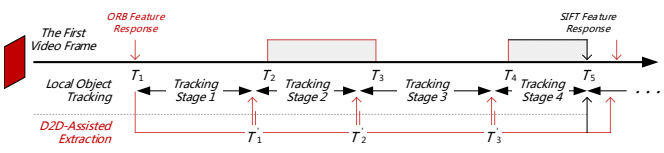


Fig. 8. D2D-enhanced feature extraction mechanism.

A simplified example is illustrated in Figure 8. At time T_1, the client receives the ORB-based recognition result from the back-end server, and then activates the tracking service. At the same time, the client transmits the current video frame to a specific nearby mobile device to perform feature extraction. Assuming the client receives the extracted features at time $T_1^{'}$, this result is then used to update the reference feature points for the next object tracking stage. Similarly, the video frame at $T_1^{'}$ will be transmitted to a nearby mobile device via the D2D channel for feature extraction. In this way, mobile devices can be effectively combined to provide collaborative computing for performance optimization. Note that the selection of candidate mobile devices is primarily based on response latency; the mobile device with the shortest communication and computing latency will be selected as an auxiliary device.

Algorithm 3 Auxiliary Nearby Mobile Device Selection

Input:
Client capability C_s and feature extraction time t_s;
Latency of D2D communication links $\mathbf{L}'$;
Nearby Mobile devices $\mathbf{H}$ ($H_i \in \mathbf{H}$) with computing capability $\mathbf{C}$ ($C_i \in \mathbf{C}$).

Output:
The auxiliary nearby mobile device: H_x.

1: **for** each nearby device H_i connected to the client **do**
2: $P_i \leftarrow t_s \times C_s / C_i$ ▷ Feature extraction latency
3: $T_i' \leftarrow P_i + L_i'$ ▷ Response latency
4: **end for**
5: **return** $\arg\min T_i'$ ▷ Greedy approach for selection

This composite scheme was adopted for the following reasons: (1) Assigning the feature extraction described above to the edge server introduces an additional computational burden, and data transmission will also cause the occupation of public network spectrum resources; (2) The computing capability of current mobile devices (even mobile Web browsers) is sufficient to complete lightweight feature extraction, and D2D-based communication provides another effective solution for data transmission.

6 Evaluation of the Edge AR X5

For performance evaluation, we developed a multi-user mobile Web AR application called Panda Betrayal, and conducted experiments using real-world 5G networks. In this section, we describe the experimental setup, and present an evaluation of the proposed collaborative communication and computing mechanisms.

6.1 Experimental Setup

The experiments were conducted in a real-world 5G network at Beijing University of Posts and Telecommunications, where full 5G coverage has been achieved in Beijing, P.R. China. The experimental network environment is illustrated in Figure 9. And three Huawei Mate 20 X (5G) smartphones were interconnected via the D2D (Wi-Fi Direct) communication technique. The distance between users was about 1∼3 meters, the bandwidth of the D2D communication links was 85∼150 Mbps, and the latency was

5∼25 ms. Moreover, the communication latency between the mobile device and edge platform was 10∼20 ms, and the uplink and downlink bandwidths were 100∼150 Mbps and 600∼700 Mbps, respectively.

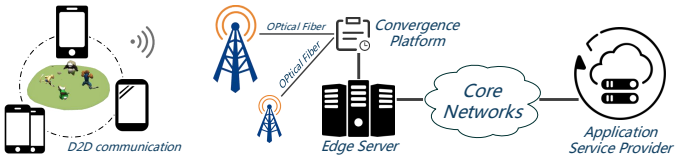


Fig. 9. Experimental 5G network environment.

All users accessed the AR application through the mobile Chrome browser. The 5G base station first transmitted all received signals via an optical fiber to the convergence platform, then to the edge platform and to the cloud. The convergence platform provided edge platform accessibility for all 5G base stations within 5 kilometers.

Note that to improve computation efficiency, our approach maximizes the offloading of the computations to either mobile device or its edge server. When the nearby mobile device can take part of computations while satisfying the user experience, our approach chooses to offloading to the device. But if the neighborhood mobile devices are not available, we naturally choose the edge server for offloading. Given this offloading design is straightforward in our system, we omit the discussion in the paper.

6.2 Performance Evaluation

In this evaluation of the Edge AR X5 system, we will analyze the proposed BA-CPP and Mo-KFP mechanisms, which are used in Edge AR X5 for collaborative communication and computing, respectively, since these are the two most important issues, as mentioned earlier.

6.2.1 Multi-User Communication Analysis

In addition to a communication scenario involving 3 mobile users, we also compare the communication efficiency in scenarios with 5 and 10 AR subscribers for generalization.

One of the most important advantages of the BAS algorithm is its efficient search capability. Here, we analyze the communication planning performance of our proposed BA-CPP mechanism. As discussed earlier, the weighting factors α, β, and γ are used to balance the importance of the communication latency, mobile energy consumption, and spectrum occupancy in cost function, respectively.

A good communication "map" can be found after about 100 iterations in scenario with 3 mobile users, as illustrated in Figure 10. For scenarios in which 5 or 10 mobile users communicate typically requires a high number of search iterations to obtain a good solution due to the exponential increase in the search space. But the BAS algorithm only compares the detected odor concentration (i.e., fitness value) of its antennae at each iteration, time complexity of the algorithm is thus $\mathcal{O}(2m)$, which does not impose a huge computational burden on the system.

For the purposes of performance evaluation, we compare our proposed solution with the other three mechanisms, as follows: (1) All-Edge, in which the edge platform broadcasts all AR subscribers' interactions; (2) All-D2D, in which all mobile users communicate via the D2D technique, and we randomly select one user to synchronize interactions with the edge server; and (3) Random, in which the communication "map" is generated randomly.

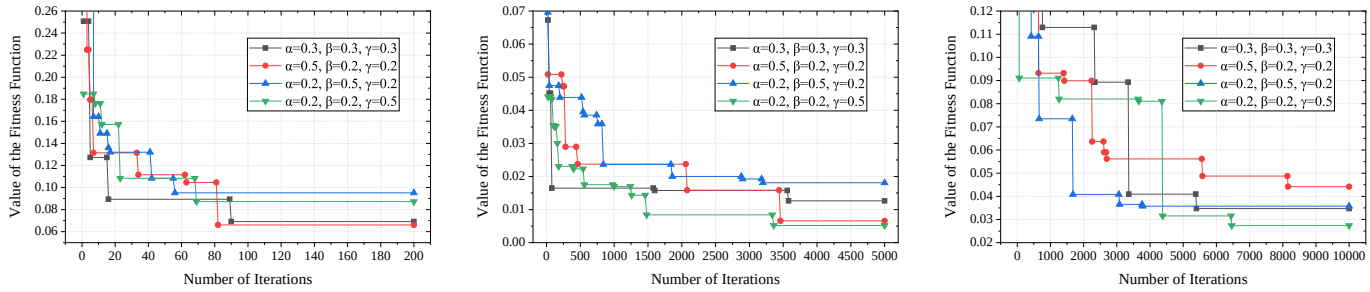
TABLE 2
A brief summary of the number of trackable feature points in different movement scenarios

Movement	Fast Movement (Move-1)									
Frame No.	1	10	20	30	40	50	60	70	80	90
SIFT	354	329	273	232	209	178	161	157	155	155
SURF	468	407	317	264	225	185	161	153	149	146
ORB	406	360	280	242	191	147	118	114	112	109
BRISK	673	589	479	402	354	298	268	262	259	255
AKAZE	324	268	223	187	164	141	136	134	134	133
Movement	Camera Rotation (Move-2)									
Frame No.	1	10	20	30	40	50	60	70	80	90
SIFT	544	521	467	457	430	387	375	324	303	281
SURF	797	655	563	548	498	456	425	369	349	320
ORB	500	492	445	439	401	369	331	245	213	170
BRISK	943	910	815	803	749	667	636	566	546	511
AKAZE	426	406	362	352	332	301	286	247	234	211
Movement	Forward and Backward Scaling (Move-3)									
Frame No.	1	10	20	30	40	50	60	70	80	90
SIFT	594	587	518	516	515	500	475	461	453	452
SURF	915	782	600	583	581	560	532	509	500	498
ORB	500	500	468	468	467	453	436	394	393	389
BRISK	941	941	858	845	843	816	786	756	739	732
AKAZE	471	464	412	407	405	388	363	342	336	336
Movement	Camera Tile (Move-4)									
Frame No.	1	10	20	30	40	50	60	70	80	90
SIFT	951	927	916	915	911	894	894	894	894	816
SURF	1282	1196	1149	1117	1109	1044	1036	1020	1020	912
ORB	500	500	494	494	494	478	478	478	478	435
BRISK	1550	1547	1534	1529	1524	1474	1474	1474	1474	1362
AKAZE	838	832	821	812	809	776	774	772	771	705

The experimental results demonstrated that our approach performs better than the other solutions (here we adopt the same weighting factors as an example, i.e., $\alpha = \beta = \gamma = 0.3$), as illustrated in Figure 11. In particular, the performance of our approach is similar to that of the All-D2D mechanism, but the better choice of communication links between mobile devices and the edge server provides lower communication latency and mobile energy consumption, that is, an improvement of about 35.51%, 85.61%, and 26.19% for the three communication scenarios, respectively. Since the BA-CPP and All-D2D mechanisms have only one mobile device connected to the edge server, the spectrum occupancy is therefore zero (i.e., at the positive ideal point). The most significant difference is that our approach is able to choose the most efficient communication link from the edge-based and D2D-based alternatives. Note that the final value of the fitness function is related to network delay and bandwidth. Because these network attributes in the experiments are randomly generated within a certain range, the final value therefore will not show a certain trend in the communication scenario with different numbers of users.

6.2.2 Motion-Aware Key Frame Selection Analysis

A unique feature of our key frame selection approach is that it can predict the video frames that need to be offloaded to the back-end for feature extraction based on the user's movement. For a comprehensive comparison, we analyzed four different user behaviors (i.e., changes in FoV): (1) fast movement; (2) camera rotation; (3) forward and backward scaling; and (4) camera tilt, and we recorded the number of feature points that could be tracked by the K–L optical flow



(a) Communication between three users. (b) Communication between five users. (c) Communication between ten users.

Fig. 10. Communication "map" searching and performance comparison for different multi-user scenarios.

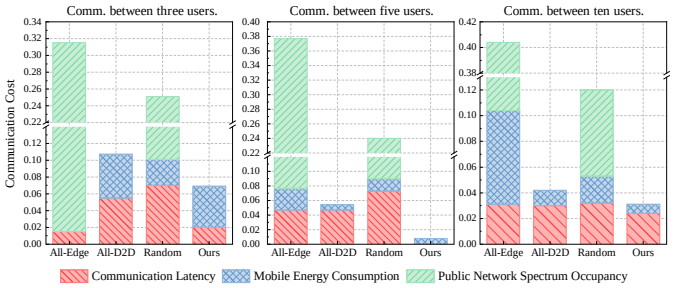


Fig. 11. Performance comparison for multi-user communication.

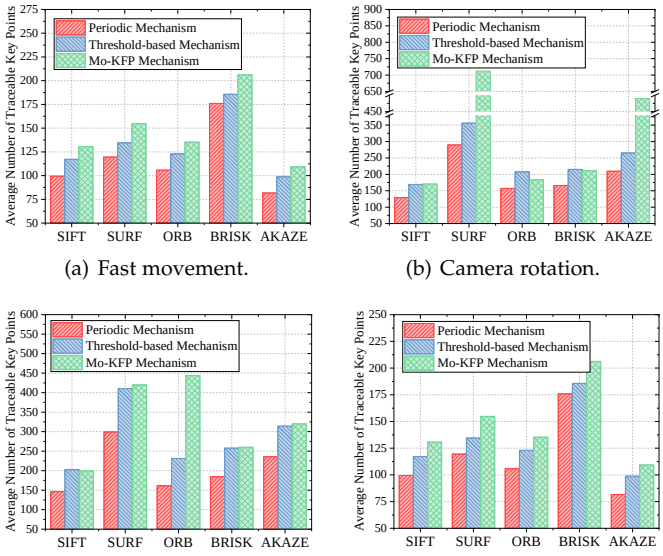
algorithm over three seconds (90 video frames in total). A brief summary is given in Table 2.

TABLE 3
Detailed key frame prediction results comparison

		Target	Mo-KFP		Logistic		Polynomial		Linear	
		KPN	FID	KPN	FID	KPN	FID	KPN	FID	KPN
Move-1	SIFT	157	48	178	50	178	51	178	42	208
	SURF	155	45	222	46	221	53	184	41	224
	ORB	115	52	146	52	146	63	116	46	188
	BRISK	157	56	270	43	353	51	297	42	353
	AKAZE	134	41	164	48	142	48	142	40	164
Move-2	SIFT	338	70	324	51	387	66	364	58	375
	SURF	381	79	351	71	364	48	463	40	498
	ORB	270	82	183	N/A	–	N/A	–	82	183
	BRISK	588	74	560	41	725	68	570	59	635
	AKAZE	260	66	276	58	286	63	279	55	294
Move-3	SIFT	461	64	463	N/A	–	38	516	37	516
	SURF	510	63	512	32	582	31	582	N/A	–
	ORB	394	89	389	N/A	–	65	395	60	436
	BRISK	758	58	438	68	758	44	817	42	832
	AKAZE	342	71	341	49	388	48	388	45	388
Move-4	SIFT	894	61	894	66	894	35	915	35	915
	SURF	1024	46	1049	72	1020	42	1101	40	1109
	ORB	478	57	478	59	478	61	478	60	478
	BRISK	1474	65	1474	65	1474	83	1473	82	1474
	AKAZE	772	74	773	N/A	–	71	772	70	772
Normalized Accuracy			**100%**		**88.39%**		**72.74%**		**57.31%**	

To verify the effectiveness of our proposed key frame selection mechanism Mo-KFP, we analyzed the data obtained and manually labeled the target key frame (the 65th video frame) as the ground truth. We then predicted the key frame (i.e., FID) based on the recorded number of trackable feature points (i.e., KPN) at the 65th frame, for different types of user movement. The prediction results and normalized prediction accuracy are presented in Table 3. It can be seen that our proposed ARIMA-based key frame selection mechanism performs more accurately than the other linear, polynomial, and logistic regression-based methods. Specifically, Mo-KFP achieves average improvements in prediction accuracy of 41.32% (linear), 25.52% (polynomial), and 15.48% (logistic).

More accurate predictions will obviously lead to higher computational efficiency. Although the prediction error exists, the advantage of our proposed Mo-KFP is that it can balance the user experience and computational costs simultaneously. Specifically, our Mo-KFP mechanism can provide better tracking performance with a low offloading frequency compared with the previous periodic and threshold-based key frame selection mechanisms.



(a) Fast movement. (b) Camera rotation.

(c) Forward & backward scaling. (d) Camera tile.

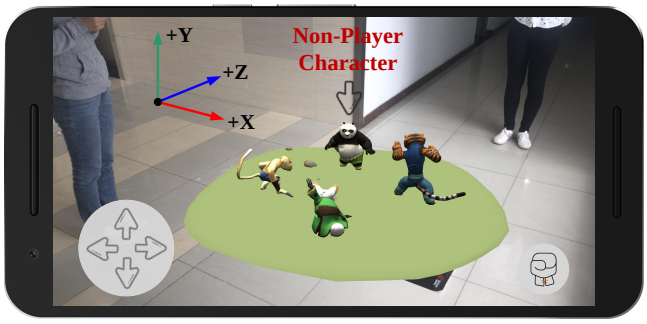
Fig. 12. Comparison of feature extraction-and-tracking efficiency.

In detail, we summarize the feature extraction-and-tracking efficiency, which is defined as the ratio of the average number of trackable feature points to the number of selected key frames, over a period of time, as illustrated in Figure 12. Our approach achieves the best feature extraction-and-tracking efficiency, with average improvements of 39.67%/1.21%, 60.39%/20.88%, 70.98%/24.92%, 63.39%/22.73%, and 39.05%/0.69% compared with the above two key frame selection mechanisms, for five feature extraction algorithms (SIFT, SURF, ORB, BRISK, and AKAZE, respectively).

6.3 Application Implementation

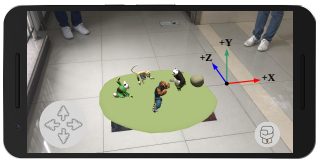
We implemented a multi-user online game called Panda Betrayal (see Figure 13) for mobile Web AR as an example,

in which all users could access the AR application through a pre-defined URL. When the user targeted a specific template image (e.g., poster), the "augmented world" will be activated and presented to the user. In addition to mobile Web browsers, this link can also be embedded into many other applications such as Facebook, Twitter, WeChat, etc.



(a) Master Shifu view.

(b) Master Monkey view.



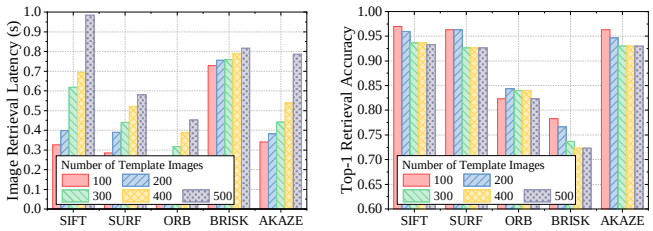
(c) Master Tigress view.

Fig. 13. Multi-user mobile Web AR application Panda Betrayal.

Specifically, all mobile devices connect to the edge server using Socket.IO technology. The edge server generates and maintains the communication "map", which is a JavaScript object that holds all link information. When the monitored context (see Figure 3) is updated, the new "map" will be packaged into JSON format[7] and delivered to all the mobile devices. Once the connection is established, the mobile device starts to monitor user operations in real-time, and update the virtual model accordingly, these operations will also be forwarded to other participants, simultaneously; when received operations from others, this JSON document will first be parsed into a JavaScript object and then used to update the virtual model. In Panda Betrayal, each player triggers the attack event by clicking the button on the right of the screen and controls the movement of their respective characters (i.e., Shifu, Monkey, and Tigress) using the directional pad to avoid the non-player character's attack. The three players will collaborate to defeat the betrayer (Panda). Remarkably, the scale of the virtual scene can change with the distance between the user and the augmented target (the poster in Panda Betrayal) or can be manually set by the user.

Here, we take the SIFT and ORB algorithms as an example of the AR service initialization problem described above. The edge server performs the feature extraction and image retrieval (100 templates) processes, taking on average 450 ms and 180 ms (including transmission latency) for the SIFT and ORB algorithms, respectively. Our proposed solution can therefore effectively improve the initialization time by about 60% (although the number of feature points that can

7. JSON as a lightweight data exchange format, can effectively improve network transmission efficiency and thus reduce the burden of data transmission on the network. In the case of 10 communication nodes, the generated JSON document is only 1.053 kilobytes.



(a) Retrieval latency. (b) Top-1 retrieval accuracy.

Fig. 14. Edge-assisted image (500×500 pixels) retrieval.

be tracked is reduced by 22.31% compared to the SIFT algorithm). D2D-based ORB feature extraction takes approximately 200 ms. Also user's different movement states will affect the initialization performance. For the aforementioned four movement scenarios, the number of trackable feature points can be increased by about 5.2‰ at the same time, when compared to the scenario without D2D assistance. Note that in different user movement and feature extraction algorithms, our approach will achieve different application performance improvement.

In addition to the communication and computing advantages that the 5G network offers, another inherent feature of the network edge server is the ability to provide location-based service as discussed in Section 3. Owing to the close correlation of the augmented targets and spatial geological location, for example, when the subscriber is located in Beijing, P.R. China, the augmented target may be Tian'anmen Rostrum, and it will not be the Statue of Liberty in the United States. By referring to the location information of the user, the search space of the reference images (i.e., retrieval database) can be effectively reduced using locality information. Here we compared the retrieval latency and the Top-1 retrieval accuracy under the different number of reference images as illustrated in Figure 14 for demonstration purpose. Obviously, the location-based reference image filtering mechanism can simultaneously and effectively improve the retrieval speed and accuracy.

7 Discussion

In this paper, we have presented the first multi-user collaborative framework for mobile Web AR in 5G networks. By coordinating pervasive communication and computing resources, we have been able to significantly improve the UX of mobile Web AR applications.

However, there are still many challenges to address in the field of Web-based mobile AR.

On-Web AI Service. The improvement in the computing capability of mobile Web browsers have made it increasingly possible to carry out part of complex computations locally [51]. Meanwhile, deep neural network-based feature extraction and object tracking techniques, such as LIFT [52] and FlowNet [53], offer better performance than traditional computer vision techniques. But considering the computational complexity and the size of the DNN models, it is currently impossible to apply them directly to a mobile Web browser, especially built-in browsers [54].

Multi-Edge Collaboration AI. For demonstration purposes, only one edge server was used in this work to assist AR

subscribers, but considering the mobility of users, a flexible edge server collaboration (e.g., service migration and mobility management) is also worth considering [32].

5G-Enabled Networking AI. In addition to the computing and communication techniques mentioned here, network slicing [55] can also be used as a support networking technology under 5G networks, thus enabling perform intelligent network resource scheduling based on different network services, providing further opportunities for improvements in the UX of mobile Web AR.

8 Conclusions

Web-based multi-user mobile AR can be used to achieve a lightweight and cross-platform solution that is a promising research direction for various applications. The emergence of 5G networks has introduced both opportunities and challenges, especially for multi-user communication and computation outsourcing. In this paper, we propose a collaborative framework called Edge AR X5 that provides a better UX by coordinating pervasive communication and computing resources. Specifically, we proposed the BA-CPP mechanism for multi-user communication planning, which balances the requirements of users and Internet service providers. We also proposed a motion-aware key frame selection mechanism called Mo-KFP, which improves the computational efficiency of the system and allows collaboration between the computing resources of nearby mobile devices via the D2D technique to addressing the issue of a long initialization times. Experiments were conducted on real-world 5G networks and demonstrated the effectiveness of our proposed collaborative multi-user mobile Web AR framework. Our current efforts represent a preliminary attempt towards the multi-user mobile Web AR in the 5G era, and there is a pressing need for joint efforts between academia and industry to promote this technology.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2018YFE0205503, in part by the National Natural Science Foundation of China (NSFC) under Grant 61671081, in part by the Funds for International Cooperation and Exchange of NSFC under Grant 61720106007, in part by the 111 Project under Grant B18008, in part by the Beijing Natural Science Foundation under Grant 4172042, in part by the Fundamental Research Funds for the Central Universities under Grant 2018XKJC01, in part by the BUPT Excellent Ph.D. Students Foundation under Grant CX2019213, and in part by the China Scholarship Council under Grant 201906470033.

References

[1] W. Zhang, B. Han, P. Hui, V. Gopalakrishnan, E. Zavesky, and F. Qian, "CARS: Collaborative Augmented Reality for Socialization," in *Proceedings of the 19th International Workshop on Mobile Computing Systems & Applications*. ACM, 2018, pp. 25–30.

[2] A. H. Hoppe, K. Westerkamp, S. Maier, F. van de Camp, and R. Stiefelhagen, "Multi-user Collaboration on Complex Data in Virtual and Augmented Reality," in *International Conference on Human-Computer Interaction*. Springer, 2018, pp. 258–265.

[3] B. Ridel, L. Mignard-Debise, X. Granier, and P. Reuter, "EgoSAR: Towards a Personalized Spatial Augmented Reality Experience in Multi-user Environments," in *2016 IEEE International Symposium on Mixed and Augmented Reality*. IEEE, 2016, pp. 64–69.

[4] P. Roberto, F. Emanuele, Z. Primo, M. Adriano, L. Jelena, and P. Marina, "Design, Large-Scale Usage Testing, and Important Metrics for Augmented Reality Gaming Applications," *ACM Transactions on Multimedia Computing, Communications, and Applications*, vol. 15, no. 2, pp. 1–18, 2019.

[5] M. Abramovici, M. Wolf, S. Adwernat, and M. Neges, "Context-aware Maintenance Support for Augmented Reality Assistance and Synchronous Multi-user Collaboration," *Procedia CIRP*, vol. 59, pp. 18–22, 2017.

[6] J. G. Shin, G. Ng, and D. Saakes, "Couples Designing their Living Room Together: a Study with Collaborative Handheld Augmented Reality," in *Proceedings of the 9th Augmented Human International Conference*. ACM, 2018.

[7] C. J. Dede, J. Jacobson, and J. Richards, "Introduction: Virtual, Augmented, and Mixed Realities in Education," in *Virtual, Augmented, and Mixed Realities in Education*. Springer, 2017, pp. 1–16.

[8] "ARCore," https://developers.google.com/ar/.

[9] "Google Just a Line," https://justaline.withgoogle.com/.

[10] "ARKit," https://developer.apple.com/augmented-reality/.

[11] "Facebook AR Studio," https://sparkar.facebook.com/ar-studio/.

[12] Z. Huang, W. Li, P. Hui, and C. Peylo, "CloudRidAR: A Cloud-based Architecture for Mobile Augmented Reality," in *Proceedings of the 2014 Workshop on Mobile Augmented Reality and Robotic Technology-Based Systems*. ACM, 2014, pp. 29–34.

[13] X. Qiao, P. Ren, S. Dustdar, and J. Chen, "A New Era for Web AR with Mobile Edge Computing," *IEEE Internet Computing*, vol. 22, no. 4, pp. 46–55, 2018.

[14] T. Y.-H. Chen, L. Ravindranath, S. Deng, P. Bahl, and H. Balakrishnan, "Glimpse: Continuous, Real-Time Object Recognition on Mobile Devices," in *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*. ACM, 2015, pp. 155–168.

[15] W. Zhang, S. Lin, F. H. Bijarbooneh, H. F. Cheng, and P. Hui, "CloudAR: A Cloud-based Framework for Mobile Augmented Reality," in *Proceedings of the on Thematic Workshops of ACM Multimedia*. ACM, 2017, pp. 194–200.

[16] "WebXR Device API," https://www.w3.org/TR/webxr/.

[17] "World Wide Web Consortium (W3C) Immersive Web Working Group," https://www.w3.org/community/immersive-web/.

[18] X. Qiao, P. Ren, S. Dustdar, L. Liu, H. Ma, and J. Chen, "Web AR: A Promising Future for Mobile Augmented Reality–State of the Art, Challenges, and Insights," *Proceedings of the IEEE*, vol. 107, no. 4, pp. 651–666, 2019.

[19] X. Qiao, P. Ren, G. Nan, L. Liu, S. Dustdar, and J. Chen, "Mobile Web Augmented Reality in 5G and Beyond: Challenges, Opportunities, and Future Directions," *China Communications*, vol. 16, no. 9, pp. 141–154, 2019.

[20] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge Computing: Vision and Challenges," *IEEE Internet of Things Journal*, vol. 3, no. 5, pp. 637–646, 2016.

[21] M. N. Tehrani, M. Uysal, and H. Yanikomeroglu, "Device-to-Device Communication in 5G Cellular Networks: Challenges, Solutions, and Future Directions," *IEEE Communications Magazine*, vol. 52, no. 5, pp. 86–92, 2014.

[22] M. Agiwal, A. Roy, and N. Saxena, "Next Generation 5G Wireless Networks: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials*, vol. 18, no. 3, pp. 1617–1655, 2016.

[23] R. T. Azuma, "A Survey of Augmented Reality," *Presence: Teleoperators & Virtual Environments*, vol. 6, no. 4, pp. 355–385, 1997.

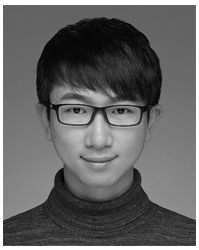
[24] D. Chatzopoulos, C. Bermejo, Z. Huang, and P. Hui, "Mobile Augmented Reality Survey: From Where We Are to Where We Go," *IEEE Access*, vol. 5, pp. 6917–6950, 2017.

[25] J. Hamari, A. Malik, J. Koski, and A. Johri, "Uses and Gratifications of Pokémon GO: Why do People Play Mobile Location-Based Augmented Reality Games?" *International Journal of Human–Computer Interaction*, vol. 35, no. 9, pp. 804–819, 2019.

[26] J. Linowes and K. Babilinski, *Augmented Reality for Developers: Build practical augmented reality applications with Unity, ARCore, ARKit, and Vuforia*. Packt Publishing Ltd, 2017.

[27] X. Chen, Y. Zhang, Q. Ai, H. Xu, J. Yan, and Z. Qin, "Personalized Key Frame Recommendation," in *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*. ACM, 2017, pp. 315–324.

[28] Z. Gao, G. Lu, and P. Yan, "Key-Frame Selection for Video Summarization: an Approach of Multidimensional Time Series Analysis," *Multidimensional Systems and Signal Processing*, vol. 29, no. 4, pp. 1485–1505, 2018.
[29] M. Lanham, *Learn ARCore–Fundamentals of Google ARCore: Learn to build augmented reality apps for Android, Unity, and the web with Google ARCore 1.0*. Packt Publishing Ltd, 2018.
[30] J. Orlosky, K. Kiyokawa, and H. Takemura, "Virtual and Augmented Reality on the 5G Highway," *Journal of Information Processing*, vol. 25, pp. 133–141, 2017.
[31] P. Ren, X. Qiao, J. Chen, and S. Dustdar, "Mobile Edge Computing–a Booster for the Practical Provisioning Approach of Web-Based Augmented Reality," in *Proceedings of the IEEE/ACM Symposium on Edge Computing*. IEEE, 2018, pp. 349–350.
[32] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative Mobile Edge Computing in 5G Networks: New Paradigms, Scenarios, and Challenges," *IEEE Communications Magazine*, vol. 55, no. 4, pp. 54–61, 2017.
[33] R. I. Ansari, C. Chrysostomou, S. A. Hassan, M. Guizani, S. Mumtaz, J. Rodriguez, and J. J. Rodrigues, "5G D2D Networks: Techniques, Challenges, and Future Prospects," *IEEE Systems Journal*, vol. 12, no. 4, pp. 3970–3984, 2017.
[34] A. B. Johnston and D. C. Burnett, *WebRTC: APIs and RTCWEB Protocols of the HTML5 Real-Time Web*. Digital Codex LLC, 2012.
[35] X. Chen, L. Jiao, W. Li, and X. Fu, "Efficient Multi-User Computation Offloading for Mobile-Edge Cloud Computing," *IEEE/ACM Transactions on Networking*, vol. 24, no. 5, pp. 2795–2808, 2015.
[36] H. Jégou, M. Douze, C. Schmid, and P. Pérez, "Aggregating Local Descriptors into a Compact Image Representation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2010, pp. 3304–3311.
[37] J. Yu, Z. Qin, T. Wan, and X. Zhang, "Feature Integration Analysis of Bag-of-Features Model for Image Retrieval," *Neurocomputing*, vol. 120, pp. 355–364, 2013.
[38] J. Sánchez, F. Perronnin, T. Mensink, and J. Verbeek, "Image Classification with the Fisher Vector: Theory and Practice," *International Journal of Computer Vision*, vol. 105, no. 3, pp. 222–245, 2013.
[39] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
[40] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An efficient alternative to SIFT or SURF," in *Proceedings of the International Conference on Computer Vision*. IEEE, 2011, pp. 2564–2571.
[41] S. Kitanov, B. Popovski, and T. Janevski, "Quality Evaluation of Cloud and Fog Computing Services in 5G Networks," in *Enabling Technologies and Architectures for Next-Generation Networking Capabilities*. IGI Global, 2019, pp. 1–36.
[42] X. Jiang and S. Li, "BAS: Beetle Antennae Search Algorithm for Optimization Problems," *International Journal of Robotics and Control*, vol. 1, no. 1, pp. 1–5, 2018.
[43] M. Behzadian, S. K. Otaghsara, M. Yazdani, and J. Ignatius, "A state-of the-art Survey of TOPSIS Applications," *Expert Systems with Applications*, vol. 39, no. 17, pp. 13 051–13 069, 2012.
[44] R. Poli, J. Kennedy, and T. Blackwell, "Particle Swarm Optimization," *Swarm Intelligence*, vol. 1, no. 1, pp. 33–57, 2007.
[45] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.
[46] D. Weerasiri, M. C. Barukh, B. Benatallah, Q. Z. Sheng, and R. Ranjan, "A Taxonomy and Survey of Cloud Resource Orchestration Techniques," *ACM Computing Surveys*, vol. 50, no. 2, pp. 1–41, 2017.
[47] J. L. Barron and N. A. Thacker, "Internal Tutorial : Computing 2D and 3D Optical Flow," *Imaging Science and Biomedical Engineering Division, Medical School, University of Manchester*, vol. 1, 2005.
[48] L.-L. Lee, Yi-Shian ans Tong, "Forecasting Time Series Using a Methodology Based On Autoregressive Integrated Moving Average and Genetic Programming," *Knowledge-Based Systems*, vol. 24, no. 1, pp. 66–72, 2011.
[49] S. Makridakis, E. Spiliotis, and V. Assimakopoulos, "Statistical and Machine Learning Forecasting Methods: Concerns and Ways Forward," *PLoS ONE*, vol. 13, no. 3, pp. 1–26, 2018.
[50] Y. Li, S. Wang, Q. Tian, and X. Ding, "A Survey of Recent Advances in Visual Feature Detection," *Neurocomputing*, vol. 149, pp. 736–751, 2015.
[51] P. Ren, X. Qiao, Y. Huang, L. Liu, S. Dustdar, and J. Chen, "Edge-Assisted Distributed DNN Collaborative Computing Approach for Mobile Web Augmented Reality in 5G Networks," *IEEE Network*, vol. 34, no. 2, pp. 254–261, 2020.
[52] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "LIFT: Learned Invariant Feature Transform," in *European Conference on Computer Vision*. Springer, 2016, pp. 467–483.
[53] A. Dosovitskiy, P. Fischer, E. Ilg, P. Hausser, C. Hazirbas, V. Golkov, P. van der Smagt, D. Cremers, and T. Brox, "FlowNet: Learning Optical Flow With Convolutional Networks," in *The IEEE International Conference on Computer Vision*, 2015.
[54] Y. Huang, X. Qiao, P. Ren, L. Liu, C. Pu, and J. Chen, "A Lightweight Collaborative Recognition System with Binary Convolutional Neural Network for Mobile Web Augmented Reality," in *Proceedings of the International Conference on Distributed Computing Systems*. IEEE, 2019, pp. 1497–1506.
[55] H. Zhang, N. Liu, X. Chu, K. Long, A.-H. Aghvami, and V. C. Leung, "Network Slicing Based 5G and Future Mobile Networks: Mobility, Resource Management, and Challenges," *IEEE Communications Magazine*, vol. 55, no. 8, pp. 138–145, 2017.



Pei Ren is currently working toward the Ph.D. degree at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China. He is currently a Visiting Scholar with the School of Computer Science, Georgia Institute of Technology, Atlanta, GA, USA, funded by the China Scholarship Council. His current research interests include the future Internet architecture, machine learning, augmented reality, edge computing, and 5G networks.

Xiuquan Qiao is currently a Full Professor with the Beijing University of Posts and Telecommunications, Beijing, China, where he is also the Deputy Director of the State Key Laboratory of Networking and Switching Technology, Network Service Foundation Research Center of State. He has authored or co-authored over 60 technical papers in international journals and at conferences, including the IEEE Communications Magazine, Computer Networks, IEEE Internet Computing, the IEEE Transactions on Automation Science and Engineering, and the ACM SIGCOMM Computer Communication Review. His current research interests include the future Internet, services computing, computer vision, augmented reality, virtual reality, and 5G networks. Dr. Qiao was a recipient of the Beijing Nova Program in 2008 and the First Prize of the 13th Beijing Youth Outstanding Science and Technology Paper Award in 2016. He serves as the associate editor for the Computing (Springer) and the editor board of China Communications Magazine.

Yakun Huang is currently working toward the Ph.D. degree at the State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing, China. His current research interests include computer vision, distributed deep learning, machine learning, augmented reality, edge computing, and 5G networks.

Ling Liu (Fellow, IEEE) is currently a Professor at the School of Computer Science, Georgia Institute of Technology, Atlanta, GA, USA. She directs the research programs at the Distributed Data Intensive Systems Lab, examining various aspects of large-scale big data systems and analytics, including performance, availability, security, privacy, and trust. Her current research is sponsored primarily by the National Science Foundation and IBM. She has published over 300 international journal and conference articles. Dr. Liu was a recipient of the IEEE Computer Society Technical Achievement Award in 2012 and the Best Paper Award from numerous top venues, including ICDCS, WWW, IEEE Cloud, IEEE ICWS, and ACM/IEEE CCGrid. She served as the general chair and the PC chair for numerous IEEE and ACM conferences in big data, distributed computing, cloud computing, data engineering, very large databases, and the World Wide Web fields. She served as the Editor-in-Chief for the IEEE Transactions on Service Computing from 2013 to 2016. She is the Editor-in-Chief of the ACM Transactions on Internet Technology.

Calton Pu (Fellow, IEEE) received the Ph.D. degree from the University of Washington, in 1986 and served on the faculty of Columbia University and Oregon Graduate Institute. Currently, he is holding the position of professor and John P. Imlay, Jr. Chair in Software in the College of Computing, Georgia Institute of Technology, Atlanta, GA, USA. He has worked on several projects in systems and database research. He has published more than 70 journal papers and book chapters, 200 conference and refereed workshop papers. He served on more than 120 program committees. His recent research has focused on big data in Internet of things, automated N-tier application deployment and denial of information.

Schahram Dustdar (Fellow, IEEE) was an Honorary Professor of Information Systems at the Department of Computing Science, University of Groningen, Groningen, The Netherlands, from 2004 to 2010. From 2016 to 2017, he was a Visiting Professor at the University of Sevilla, Sevilla, Spain. In 2017, he was a Visiting Professor at the University of California at Berkeley, Berkeley, CA, USA. He is currently a Professor of Computer Science with the Distributed Systems Group, Technische Universitt Wien, Vienna, Austria. Dr. Dustdar was an elected member of the Academy of Europe, where he is the Chairman of the Informatics Section. He was a recipient of the ACM Distinguished Scientist Award in 2009, the IBM Faculty Award in 2012, and the IEEE TCSVC Outstanding Leadership Award for outstanding leadership in services computing in 2018. He is the Co-Editor-in-Chief of the ACM Transactions on Internet of Things and the Editor-in-Chief of Computing (Springer). He is also an Associate Editor of the IEEE Transactions on Services Computing, the IEEE Transactions on Cloud Computing, the ACM Transactions on the Web, and the ACM Transactions on Internet Technology. He serves on the Editorial Board of IEEE Internet Computing and the IEEE Computer Magazine.

Junliang Chen received the B.S. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 1955, and the Ph.D. degree in electrical engineering from the Moscow Institute of Radio Engineering, Moscow, Russia, in 1961. He has been with the Beijing University of Posts and Telecommunications, Beijing, China, since 1955, where he is currently the Chairman and a Professor with the State Key Laboratory of Networking and Switching Technology. His current research interests include communication networks and next-generation service creation technology. Dr. Chen was elected as a member of the Chinese Academy of Sciences in 1991 and a member of the Chinese Academy of Engineering in 1994 for his contributions to fault diagnosis in stored program control exchange. He received the First, Second, and Third prizes of the National Scientific and Technological Progress Award in 1988, 2004, and 1999, respectively.