Mobile Web Augmented Reality in 5G and Beyond: Challenges, Opportunities, and Future Directions

Xiuquan Qiao[1,*], Pei Ren[1], Guoshun Nan[1], Ling Liu[2], Schahram Dustdar[3], Junliang Chen[1]

[1] State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
[2] College of Computing, Georgia Institute of Technology, Atlanta, GA 30332, USA
[3] Distributed Systems Group, TU Wien, 1040 Vienna, Austria
* The corresponding author, email: qiaoxq@bupt.edu.cn

Abstract: The popularity of wearable devices and smartphones has fueled the development of Mobile Augmented Reality (MAR), which provides immersive experiences over the real world using techniques, such as computer vision and deep learning. However, the hardware-specific MAR is costly and heavy, and the App-based MAR requires an additional download and installation and it also lacks cross-platform ability. These limitations hamper the pervasive promotion of MAR. This paper argues that mobile Web AR (MWAR) holds the potential to become a practical and pervasive solution that can effectively scale to millions of end-users because MWAR can be developed as a lightweight, cross-platform, and low-cost solution for end-to-end delivery of MAR. The main challenges for making MWAR a reality lie in the low efficiency for dense computing in Web browsers, a large delay for real-time interactions over mobile networks, and the lack of standardization. The good news is that the newly emerging 5G and Beyond 5G (B5G) cellular networks can mitigate these issues to some extent via techniques such as network slicing, device-to-device communication, and mobile edge computing. In this paper, we first give an overview of the challenges and opportunities of MWAR in the 5G era. Then we describe our design and development of a generic service-oriented framework (called MWAR5) to provide a scalable, flexible, and easy to deploy MWAR solution. We evaluate the performance of our MWAR5 system in an actually deployed 5G trial network under the collaborative configurations, which shows encouraging results. Moreover, we also share the experiences and insights from our development and deployment, including some exciting future directions of MWAR over 5G and B5G networks.

Keywords: 5G; edge computing; augmented reality; mobile augmented reality; Web augmented reality

I. INTRODUCTION

Mobile Augmented Reality (MAR) [1]-[3] recently gained popularity thanks to the proliferation of smartphones, tablet PCs, and wearable devices, such as Google Glasses, MAD Gaze, and Microsoft HoloLens. Cisco reported that the traffic of MAR will increase seven-fold between 2016 and 2021, with 3 Petabytes per month in 2016 [4]. Recently we have witnessed many use-cases of MAR in a broad

Received: May 11, 2019
Revised: Jul. 02, 2019
Editor: Honggang Zhang

We describe our design and development of a generic service-oriented framework (called MWAR5) to provide a scalable, flexible, and easy to deploy MWAR solution.

area of games (*e.g.*, Pokémon GO), education (*e.g.*, IWow), and commerce (*e.g.*, IKEA Place), which can be largely classified into wearable device-based and application-based (App-based). These MAR applications provide attractive enriched experiences over the real world through cameras, GPS, and compass sensors. They are powered by new technologies, such as deep learning, computer vision, Internet of Things (IoT), and edge computing.

However, to date, wearable device-based and App-based MAR implementations have suffered from some inherent limitations, which hinder their potential deployment to millions of users, designer, engineers, and entrepreneurs. First, a hardware-specific MAR is costly and only a small number of users can afford it. Second, hardware-specific MAR approaches are heavy due to the requirement for additional hardware (*e.g.*, Meta Glass) and their platform dependency: each is designed for a specific physical device or an operating system (*e.g.*, Embedded Linux, iOS, or Android). Third, App-based MAR solutions have varying capability and interfaces among different applications. They also require the download and installation of an application-specific package onto the user's mobile phone in advance, which raises the barriers to entry for users to some extent. For example, when users travel to a city, if they wish to access a business promotion, such as advertising, shopping, or tourism, they first have to download and install one or more App-based MAR software packages to enjoy the AR experience on mobile device. This requirement is inconvenient and unfriendly to potential consumers or visitors, especially compared to alternative solutions that enable users to have on-demand use of the MAR service whenever they need it; for example, through a pay-as-you-go MAR service provisioning at edge cloud. Another limitation is that App-based MAR solutions are heavily application-dependent and device platform-dependent (iPhone *vs.* Android), and they lack cross-platform support among different applications. For example, the AR functionalities that are offered on Alipay cannot be directly used in other applications.

Nowadays, the Web has become one of the most pervasively used and fundamental infrastructures over the Internet [5]. One can do almost everything on a Web browser, such as surfing the Internet, reading news, and so on. This is possible because the HTTP communication protocol and the HTML/XML description languages with JavaScript are highly simplified and standardized, they are also flexible for both software development and Web-scale deployment. Hence, mobile Web AR (MWAR) holds considerable potential to become a scalable solution to promote AR to millions of online users via a pervasive and cross-platform user interface [6].

Previous work [7]-[9] took the first step and implemented a mobile Web AR browser. However, this requires the browser's kernel support for object detection and tracking and, hence, cannot work on other platforms without modifying the browser's kernel. The follow-up solutions, JS-ArUco [10], JSARToolkit [11], and the latest achievement (in 2017), *i.e.*, AR.js [12], are all pure front-end solutions that use JavaScript in a mobile Web browser. Unfortunately, both implementations are burdened with two problems over today's wireless networks.

- First, these solutions are limited to low Frame Per Second (FPS) due to the computing efficiency of JavaScript in a mobile Web browser for dense computing tasks. Although computing on the cloud will help to improve FPS, it introduces large latency for interactions and real-time tracking caused by wireless networks. To achieve a good immersive experience, one confronts a technical dilemma: running AR tasks on a Web browser might yield low FPS. However, placing these computation-intensive tasks on the cloud might incur high interactive latency (up to hundreds of milliseconds) [3].
- The second problem is that both implementations suffer from a lack of standardization in mobile Web browsers, Web AR enabling technologies, content producing tools, and

5G and B5G enabling technologies (*e.g.*, Mobile Edge Computing (MEC) [13], Device-to-Device (D2D) communication [14]). For example, Chrome, Firefox, and built-in browsers in social platforms (*e.g.*, Facebook, WeChat) yield different FPS results for the same mobile Web AR application on a same mobile phone [15]. In addition, diverse browsers also have different levels of support for mobile Web AR enabling technologies, such as WebRTC [16] (only Safari 11 and later versions start to support this technology), WebGL [17], *etc.* Meanwhile, Web 3D objects generated by different tools, such as 3D Max, Maya, and Blender, are incompatible with each other. And diverse communication protocols for device-to-device and deployment environment for mobile edge computing also make the pervasive promotion of MWAR challenging.

In this paper, we conjecture that these difficulties can be partially mitigated by the emerging 5G and B5G networks via new techniques such as network slicing, MEC, D2D communication, and much higher channel capacities (*i.e.*, 10x larger than 4G LTE) [18]. Hence, 5G and B5G mobile network communication technology holds the potential to boost MWAR over the World Wide Web (WWW) ecosystem to attract millions of end-users [19]. The main contributions of this paper can be summarized as follows.

- *Architecture*. Based on the previous discussion of key features of 5G and B5G networks, we present the design of a novel and generic mobile Web AR service architecture, termed Mobile Web AR over 5G (MWAR5), which is proposed to provide developers with a scalable, flexible, and easy-to-deploy Web AR solution.
- *Experiment*. To get the performance comparisons, we further detail the MWAR5 pipeline over the 5G trial network and then conduct experiments on a testbed using Chrome with three different working modes.
- Experience. The deployment and evaluations provide several valuable insights for future research of MWAR in the context of 5G networks, we summarize the lessons learned and future directions as well to provide guidelines for both researchers and developers.

The remainder of this paper is organized as follows. Section II details the state of the art in Mobile Web AR (MWAR). Section III presents the challenges of MWAR over the 3G/4G networks from three aspects. Section IV summarizes the opportunities of MWAR in the 5G era and gives the overview of the generic mobile Web AR architecture. Section V reveals a service framework of MWAR over 5G, and the experimental analyzation is given in Section VI. Section VII presents the lessons learned. Section VIII outlines avenues for future research. Finally, Section IX concludes the paper.

Table I. *MAR taxonomy and development tool kits.*

Classes	Hardware interaction	Hardware control based [20]	Google Glass, Epson BT Series, ODG R Series
		Semantic gestures based [21]	Microsoft HoloLens, HiAR Glasses
		Semantic gestures + direct contact [22]	Meta 2
	App	Navigation [23]: NaviCam, Localscope, Find Your Car with AR, Heads Up Navigator	
		Commerce [24]: iGreet, IKEA Place, Sephora Virtual Artist, Amikasa, Quiver, Tap Painter	
		Visual Art [25]: Junaio, MoMAR Gallery, ILEVA, StraxAR, Art++	
		Education [26]: iWow, Construct3D, Chemistry AR, SkyView, AR Circuits, Anatomy 4D	
		Game [27]: Pokémon GO, Star Wars Rebels, AR air hockey, Titans of Space	
		Music [28]: ARmony, AR Mixer, ControllAR, AR/DJ, LCoM AR	
	Web AR	Argon [9], JS-ArUco [10], JSARToolkit [11], AR.js [12], Web XR [29]	
Tool Kits	CloudRidAR, Vuforia, ARToolKit, Catchoom CraftAR, ARLab, Mobinett AR, Wikitude, Blippar, Layar		

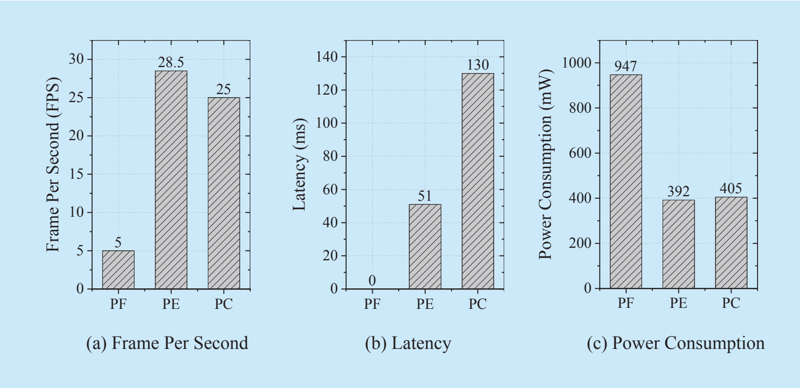
II. STATE OF THE ART IN MWAR

We present a whole picture of MAR's taxonomy in Table I, which classifies the applications into wearable device-based, App-based and Web-based. This section summarizes mobile Web AR implementations over current wireless networks.

Web browser-based: This implementation approach aims to achieve near-native performance for MWAR applications by fully utilizing the computing capability of Unit Equipment (UE). Argon [9] was one of early efforts in this direction. It was designed as a pervasive AR application provisioning platform that uses a pre-defined URL for AR service accessing. The state of the art from Mozilla and Google are Web XR [29], WebARonARKit (based on ARKit) and WebARonARCore (based on ARCore) [30], respectively. All of these efforts aim to achieve performance improvement and pervasive MWAR service provisioning to reduce the development threshold. However, the cross-platform requirement is still challenging for MWAR applications due to the lack of standardization between different MWAR-supported browsers.

JavaScript-based: This implementation approach meets the cross-platform requirement of MWAR applications. The newly proposed AR.js [12] is a MWAR library, which can run on any mobile browser with WebGL and WebRTC techniques. Note that WebGL provides a rendering acceleration solution for MWAR applications and the WebRTC achieves real-time streaming using the camera on a UE. The experiment shows that AR.js can reach 60 FPS on a 2-year-old phone for a simple fiducial marker. Although many efforts are now available to implement MWAR by using JavaScript, such as JS-ArUco [10], awe.js [31], the performance of the MWAR application is still limited due to the low computing efficiency of JavaScript, which will further degrade the user experience.

Our proposed MWAR5 can schedule distributed computing between the cloud, network edges, and UEs, which therefore has the advantages of both cross-platform support and performance guarantee compared with existing approaches. Moreover, with the advance of 5G and B5G enabling technologies, MWAR5 applications can achieve more powerful functionalities. For example, the D2D communication together with WebRTC technology make collaborative MWAR5 applications and short-distance data sharing possible.



(a) Frame Per Second (b) Latency (c) Power Consumption

Fig. 1. *The performance of mobile Web AR over 3G/4G networks in terms of FPS, latency, and power consumption (PF: Pure Front-end; PE: Pure MEC; PC: Pure Public Cloud).*

III. CHALLENGES OF MWAR OVER A 3G/4G NETWORK

As we discussed, MWAR is a promising solution to attract millions of online users via standardized Web technologies. Unfortunately, it does not come for free and this section will outline some of the challenges that confront the state of the art Web-based MAR over today's 3G/4G wireless networks.

Computing efficiency for CPU hungry tasks on a mobile Web browser: AR computations include video or image processing, object recognition and tracking, *etc.* All of the tasks require dense computing and they may lead to very low FPS when using JavaScript on a mobile Web browser. The widely used JavaScript performs poorly for highly complex computation tasks, such object detection and feature extraction over a mobile Web browser. Figure 1 shows that the FPS of a pure front-end solution (*e.g.*, AR.js) for an image marker is as low as 5. Meanwhile, the corresponding battery consumption is very high; *i.e.*, an average 947 mW per 5 minutes. In addition, embedding AR's API into a mobile Web browser may sacrifice the generalization ability. Although mainstream Web browsers have already supported WebAssembly [32] technology, which is designed to encode procedures (*e.g.*, C, C++, Rust) into a size- and load-time-efficient binary format, the performance is still limited when applying MAWR applications on the Web. Hence, there is still room for improvement to achieve efficient computing on a mobile Web browser.

Latency for real-time interactions: Due to the low computing efficiency of a mobile Web browser and the limited battery capacity of a UE, offloading the tasks to a public cloud can improve FPS, although it incurs additional delay during the interactions. Previous work shows that the user experience of MAR is highly sensitive to communication latency between a user and the back-end servers. Large latency cannot meet high interactive demands and this will lead to poor user experience. As shown in figure 1, although the FPS of pure MEC and public cloud can reach 28.5 and 25, respectively, the average latency can be as large as 51 ms and 130 ms, and will be much worse over a deteriorated wireless channel that suffers from interference, congestion, and signal fading, *etc.*

Standardization: The goal of MWAR is to build a hardware-independent, browser-independent, and producer-independent service framework using standardized Web technologies. However, achieving this goal is very challenging due to the following difficulties: (1) There is no standardization of MEC and D2D techniques to define the common protocols and APIs to simplify the deployment of MWAR; (2) Mobile hardware such as smartphones, tablet PCs, and laptops have different levels of support for AR; *e.g.*, capturing video with the camera over a Web browser is restricted on some mobile phones; (3) Modern Web browsers, such as Chrome, Firefox, and built-in browsers have very different capabilities to support AR applications [15] in terms of FPS and processing time, let alone the distinctions of the built-in Web browsers over social platforms such as WeChat; (4) We still lack a standard for Web 3D objects, which

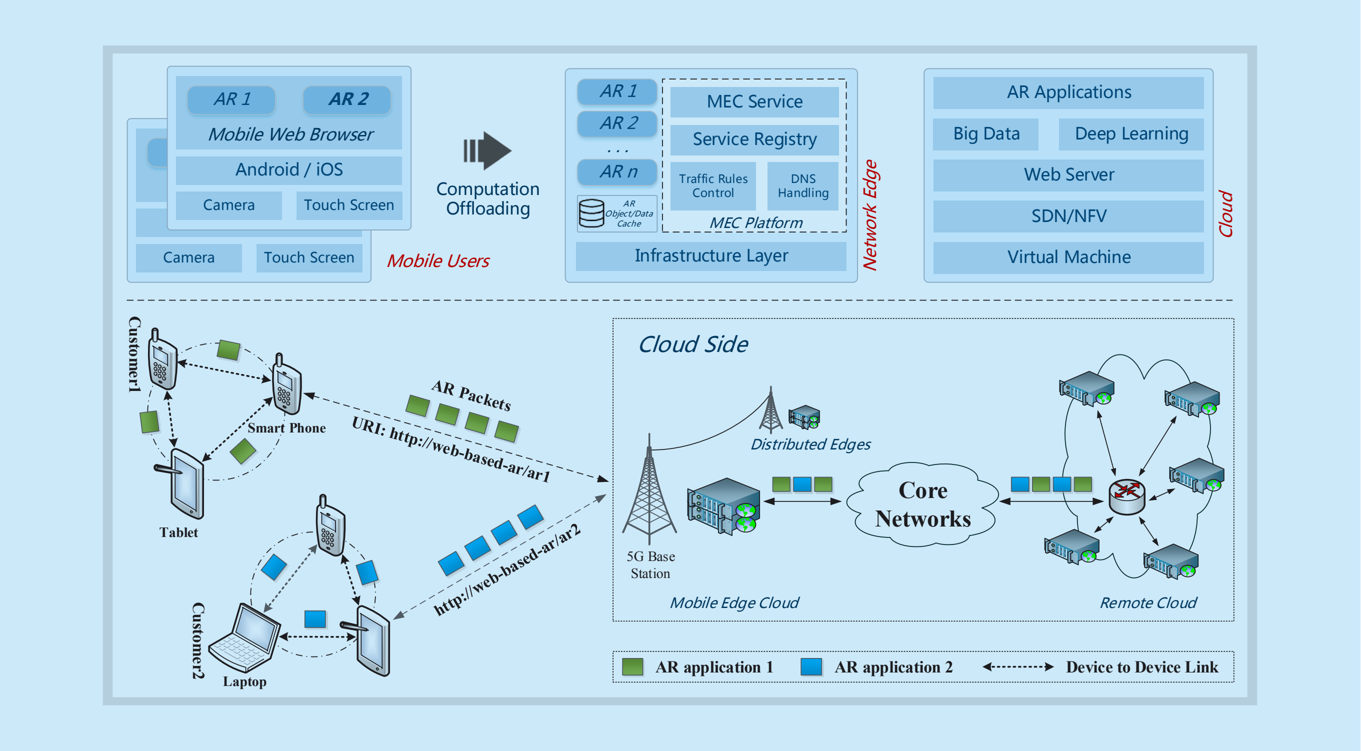


Fig. 2. *Generic Mobile Web AR Architecture over 5G networks. Two MAR applications (i.e., AR1 and AR2) are demonstrated in this case.*

leads to compatibility issues among different 3D tools, such as 3D Max, Maya, and Blender. The MWAR standardizations are therefore expected to be released.

IV. OPPORTUNITIES OF MWAR IN THE 5G ERA

To meet the ever-growing traffic demands over wireless networks, the 5G and B5G cellular networks [33] have been proposed as the next-generation wireless network architectures. In particular, 5G aims to provide about 1000x system capacity, 25x average cell throughput, green energy and very high data rates (typically of Gbps order), as well as extremely low latency (1-10 ms) compared to current 4G LTE networks. Fundamentally, 5G and B5G will shift the cellular network architecture from a base station-centric model to a device or user-centric model. It also innovates several new techniques, such as D2D, MEC, Multiple-Input Multiple-Output (MIMO), spatial modulation, *etc.* Thus, 5G and B5G hold the potential to foster new business models among network operators, MAR enterprise providers, and even individual designers. Figure 2 demonstrates how MWAR works in the context of 5G networks and the ecosystem over WWW.

As illustrated in figure 2, at the client side, there are six mobile users belonging to two different groups, which are classified by two MAR applications, *AR1* and *AR2*. Mobile users in the same group can exchange data directly thanks to the D2D communication technique of the 5G networks at the Radio Access Network (RAN), which will result in a significant improvement of latency and FPS. As shown, two clients, *customer1* and *customer2*, are experiencing MAR applications *AR1* and *AR2* on mobile Web browsers via the URLs *http://Web-based-ar/ar1* and *http://Web-based-ar/ar2*, respectively, without any pre-installation required. The MEC cloud is deployed at a 5G based station. We harness the generic system framework of this cloud, which is composed of three main layers; *i.e.*, the infrastructure layer, the network function layer, and the service layer.

Moreover, Web-based mobile augmented reality as a computation-intensive and delay-sensitive application, which has higher requirements for the network performance, especially for bandwidth and end-to-end latency. Fortunately, the upcoming 5G proposed three typical application scenarios, *i.e.*, enhanced Mobile Broadband (eMBB), Ultra Reliable Low Latency Communications (URLLC), and massive Machine Type Communications (mMTC). By using the network slicing technology [34], which enables flexible resource allocation to meet the different application requirements, it can therefore easily accommodate the communication demands of AR services, and will further improve the utilization of network resource.

Meanwhile, both of these clients can be served by cached objects and data or by AR services running over the service layer of MEC cloud [35][36]. Intuitively, the latency and FPS can be improved without accessing the public cloud, which may be congested at peak hours. On a public cloud, high computing demands tasks, such as object recognition and training via a deep learning program, can be performed more efficiently over data center infrastructures.

V. A SERVICE FRAMEWORK OF MWAR OVER 5G AND B5G NETWORKS

We have discussed the challenges of MWAR over current 3G/4G networks and have also articulated the opportunities of MWAR in the context of the 5G and B5G networks. This section will present our design of a service framework for a MWAR, which is called MWAR5, by exploring the technical potentials in the 5G era.

The goal of MWAR5 is to deliver scalable, flexible, and easy to deploy MAR for millions of users, designers, engineers, and companies over a pervasive mobile Web browser that uses standard Web technologies, such as CSS, JavaScript, *etc.* The framework of the MWAR5

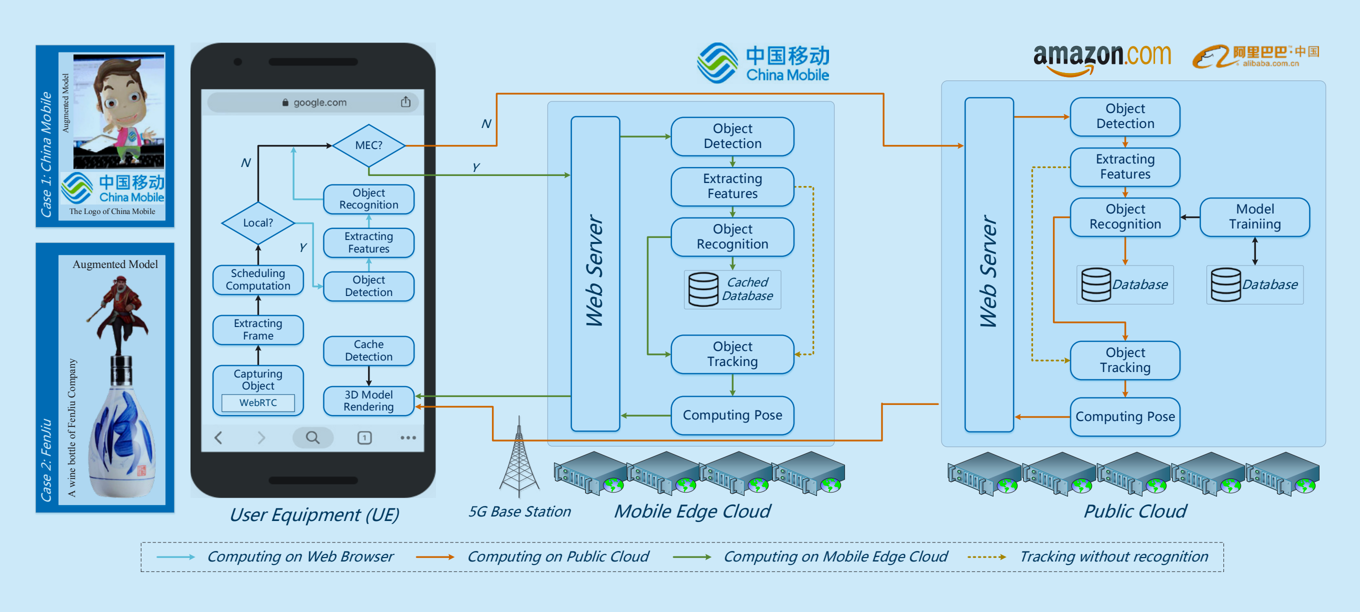


Fig. 3. *The MWAR5 pipeline over 5G and B5G networks.*

is presented in figure 3 and the workflow is described as follows.

The first step *Capturing Object* is to stream a real object (*i.e.*, an image marker or a real object) by a smartphone's or tablet's camera over a Web browser. The standardized WebRTC technology provides Web browsers with the ability to access the camera on UEs, which provides a basis for the MWAR5 implementation to achieve real-world and object capturing in a video stream format. The video is then converted into separate frames for object detection and feature extraction. Note that WebRTC as a real-time peer-to-peer communication technology also provides a basis for collaborative MWAR5 applications (*e.g.*, multiplayer interactive AR games) with the help of D2D communication technology in 5G networks.

Object Detection in the second step checks for the existence of the object and locates the Regions of Interest (RoI) corresponding to the target. Meanwhile, extracting features means finding the key point for each RoI from a frame, such as a rotated image. Due to the high computation demands (*e.g.*, RON algorithm [37]) of such tasks, the scheduler will make a decision to offload the tasks to mobile edges, the public cloud, or execute them over the browser. To achieve minimal delay, the tasks can be further broken down into sub-tasks, if possible, and they will then be adaptively allocated via reinforcement learning approaches among a mobile phone, network edges, and public cloud, respectively, depending on the service requirements and environmental conditions. Furthermore, object detections on a mobile Web browser can be tailored using JavaScript with WebAssembly technique to improve the computing efficiency and save battery energy.

The third step lies in *Object Recognition* using a Content-Based Image Retrieval (CBIR) approach [38] and then computing a new pose for augmented models over the real target. Moreover, recent algorithms that are based on deep learning, such as BIND [39], can improve the accuracy of the recognition. These tasks require dense computing and, hence, are deployed on the mobile edge cloud and public cloud.

The fourth step refers to real-time *Object Tracking*. Given the initialized state of a target object in a frame of a video stream, *Object Tracking* estimates the states of the target in the subsequent frames. The challenges lie in variations due to geometric changes and photometric factors, occlusions, and non-linear motion, *etc*. Although many accuracy and robust trackers have been proposed [40] based on Convolutional Neural Network (CNN) [41]-[44] and correlation filters (*e.g.*, kernelized correlation filters [45]), it is infeasible to deploy them directly on a mobile Web browser to yield high FPS due to limited computing capabilities. Thus, an adaptive policy using reinforcement learning may be able to help to dynamically allocate tracking tasks among mobile clients, edge nodes, and on the public cloud.

VI. PERFORMANCE EVALUATION

The experiments are conducted with two different applications, and figure 4 demonstrates the operating process.

To access the mobile Web AR application, users first need to click the pre-defined URL links in the Web browser on a mobile phone, whose camera will be activated by MWAR5. The user then scans the logo of China Mobile (or the wine bottle of FenJiu), an augmented 3D model will be downloaded and rendered in the users' Web browser over the logo or bottle. Finally, the user interacts with the augmented model to acquire more product information. These interactions are much more attractive when compared with current popular sales promotions using QR Codes.

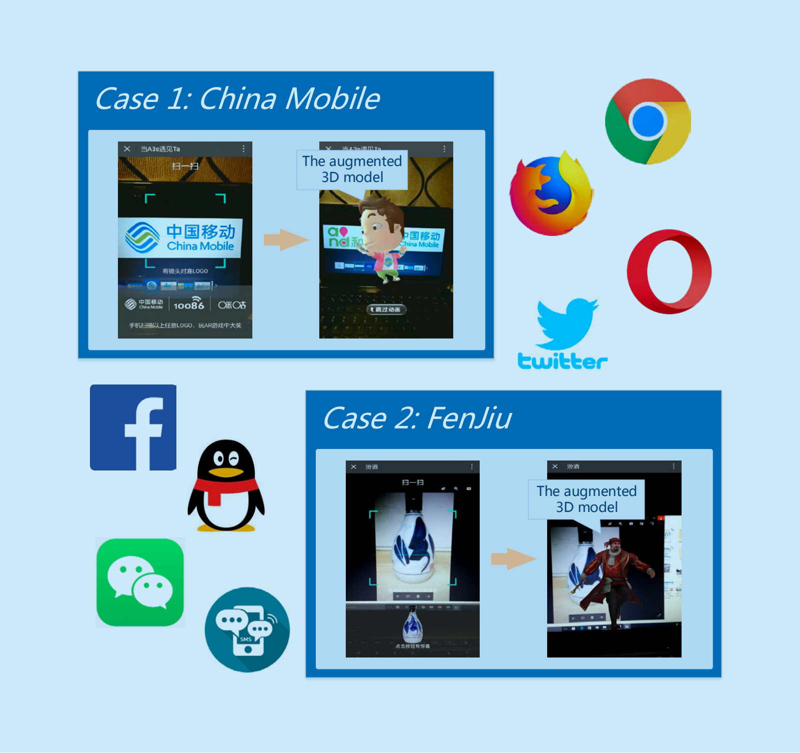


Fig. 4. *Two experimental cases of the use of MWAR5 for sales promotions by China Mobile and FenJiu, the largest mobile operator in the world and a very famous wine company in China, respectively. By scanning the logo of China Mobile or the wine bottle of FenJiu, the augmented model therefore can be activated then display to users. Multiple entrances are available for users to experience mobile Web AR applications, here we use Chrome as the demonstration.*

We conducted the experiments in an actually deployed 5G trial network which has been supported by China Mobile Communications Group Beijing Co., Ltd. and Huawei Technologies Co., Ltd. 20 times with different user devices (*e.g.*, Mi Mix 2, Samsung Galaxy Note 8, Huawei P10, Meizu MX5, Samsung Galaxy S8). All of the experiment results show a similar trend in terms of FPS, latency and power consumption. Here, we choose one result for analysis, which uses Chrome and a Huawei P10 mobile phone, equipped with a 2.4 GHz and 8 core Hisilicon CPU, 4 GB RAM. The mobile phone connects to the edge server which is deployed on the 5G base station via Customer Premise Equipment (CPE) then it connects to the MWAR5 server on the Alibaba Cloud via the core network as shown in figure 5. The network latency between the user device and the network edge is about 8.76 ms. Between the network edge and the MWAR5 cloud server, it is about about 27.04 ms, on average.

We present the design of MWAR5 by leveraging techniques that hold potential for us in the 5G and B5G networks. We also deliver a proof-of-concept development and deployment over an actually deployed 5G trial network. Three different working modes are supported in MWAR5: Front-end plus mobile Edge cloud (FE), Front-end plus public Cloud (FC), and Front-end plus mobile Edge cloud plus public Cloud (FEC). The percentages of tasks computed on a UE, edge cloud, and public cloud are about 20%, 80% for both FE and FC, and are about 20%, 60%, 20% for FEC, respectively (the computation offloading assumption scenario in our experiment is obtained based on the proportion of time required for each sub-task execution on a specific computing platform[1]). Specifically, the image preprocessing, object tracking, and augmented model rendering tasks are assigned to perform on the mobile Web browser (*i.e.*, UE). While the others, such as object detection, feature extraction, object recognition, and template matching, are performed on the edge or public cloud. Moreover, for the FEC computing mode, the template matching is assigned to perform on the public cloud to alleviate the computing pressure on the edge servers. Note that different computation partitioning policy will obviously result in different MWAR application performance in term of FPS, latency, and power consumption. For simplification, we adopted a straightforward partitioning approach in this paper only for demonstration purpose, there is still a lot of room for further study about computation offloading especially for mobile Web augmented reality service.

Our design has demonstrated that the user experience of the MWAR5 prototype application can achieve skip-type improvement, especially for latency (7 ms (5G) *vs.* 51 ms (4G), on average). In this part, we compare the performance of the MWAR5 prototype system under the FE, FC, and FEC modes, using a number of performance metrics, such as FPS, latency, and the UE's power consumption.

As shown in figure 6, the FE mode performs best in terms of FPS (*i.e.*, 31.4 in average), compared with FC (*i.e.*, 27.3) and FEC (*i.e.*, 29.0) mode, respectively. The user device shoulders about 20% of the computation tasks in three hybrid modes. It is clear that network performance has a significant impact on the performance of MWAR applications (25 FPS (4G) *vs.* 34 FPS (5G) for the PF mode in our experiment). Note that a different partitioning method will lead to a different result. Meanwhile, FE yields less average latency (*i.e.*, 5.6 ms) than FC (*i.e.*, 44.0 ms) and FEC (*i.e.*, 8.8 ms). The reduction stems from AR dense computations because the core network introduces delay into the FC and FEC, which may be worse at peak times due to network congestion. The power consumption on the mobile phone only has slight difference among three approaches, which is about 560 mW every 5 minutes. This is much less than the mode of a pure front-end (*i.e.*, 947 mW). Finally, we would like to remark that although MWAR5 is designed for a 5G network, our prototype development experience and our experiments also show that several design concepts can help to improve the application performance in current mobile networks. By deploying mobile Web AR applications to data centers or anywhere close to the user, we can often achieve performance improvement in term of FPS and latency. However, the AR service deployment on network edges in current mobile networks is more challenging and expensive because of the lack of a standard management mechanism and platform. The upcoming 5G and B5G networks will be able to provide a flexible and pervasive MWAR service deployment infrastructure on network edges, which

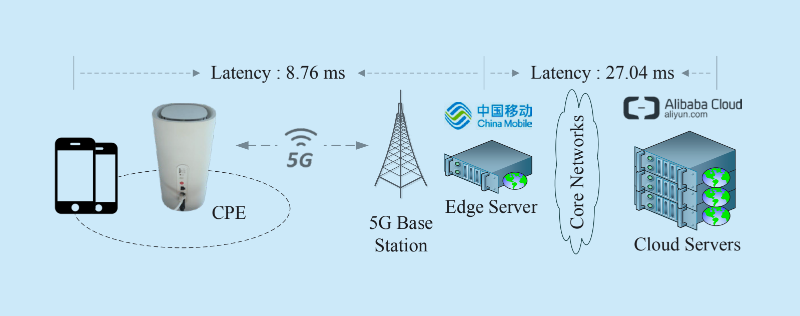
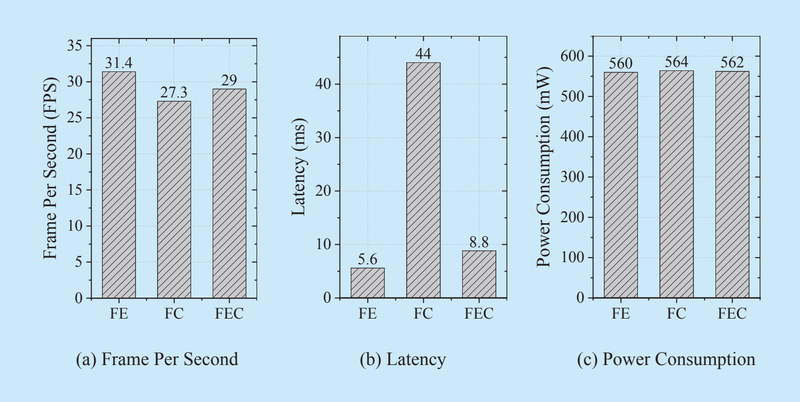


Fig. 5. *Experimental 5G network environment.*



(a) Frame Per Second (b) Latency (c) Power Consumption

Fig. 6. *Performance comparisons of different hybrid modes for MWAR5 (FE: Front-end plus mobile Edge cloud; FC: Front-end plus public Cloud; and FEC: Front-end plus mobile Edge cloud plus public Cloud).*

[1] Specifcation of the computing platform: IBM System x3650 M4, Ubuntu 16.04, Intel Xeon E5-2600 v2 @ 2.0 GHz, 16 GB RAM.

will greatly reduce the deployment difficulty for developers [3].

VII. LESSONS LEARNED

Through our research, we have learned several lessons. The first is that MWAR implementations open the door for Web users to more easily experience MAR services and, therefore, is a promising direction for MAR. Hardware specific MAR solutions are costly and inconvenient to carry, while App-based ones require a download and installation in advance, they also lack cross-platform compatibility. In contrast to the existing MAR implementations, the Web-based implementations can provide lightweight and cross-platform MAR services and, therefore, achieve pervasive promotion for MAR over WWW.

The second lesson is related to the upcoming 5G and B5G networks. Although AR applications require real-time interaction, current 3G/4G mobile networks cannot meet this requirement. The emerging 5G networks provide a higher data rate and lower communication delay, which will greatly improve the data transmission on the Internet. In this work, we have designed and implemented a MWAR5 service provisioning framework through a prototype development and deployment effort. Our experience shows that many of the limitations can be partially mitigated by the newly emerging 5G networks, powered by some advances in computer vision and deep neural networks. The MEC paradigm reduces the communication delay. Meanwhile, the D2D communication will enable efficient collaborative mobile Web AR applications, and also achieve data sharing. The development of networking will provide many opportunities for the pervasive promotion of MWAR5 applications.

The third lesson is related to collaborative computing. By gathering distributed computing and storage capabilities on the Internet, this computing paradigm can achieve flexible resource scheduling and energy saving. We have also shared our experiences with the development and deployment of collaborative MWAR5 by leveraging several techniques that hold considerable potential on 5G networks. In particular, remote or network edge server has stronger computing capability and sufficient storage space. Furthermore, reasonable resource scheduling can effectively improve the performance of MARW5 applications. Considering the diverse capability of the end devices, an adaptive collaborative computing paradigm should be considered in the future development process to balance the cost and user experience.

VIII. FUTURE DIRECTIONS

The practical deployment and performance results sheds some light for future studies of MWAR application over the 5G and B5G networks. In this section, we will detail these insights and will further discuss these issues.

Computing at the front-end and mobile edge cloud: Figure 6 indicates that computing at the front-end and mobile edge cloud stands out as a low latency and high FPS solution. Ideally, high demand tasks are expected to be processed at the front-end or network edge to reduce interactive latency. Recently, industries such as Tencent have developed an AR API in its beta version of a Web browser, which promises to improve computing efficiency. Meanwhile, offloading some tasks to the mobile edge cloud is amenable to the computing capabilities of a Web browser [46]-[49].

Real-time object recognition and tracking: These two tasks are CPU hungry, and they require high FPS and low latency. In the 5G networks, the problem can be formulated as follows: given the battery constraint of mobile phones, caching or storage size and computing capability of mobile phone, edge cloud and public cloud, and network delay among mobile phone, edge nodes and public cloud, how can we achieve maximized performance in terms of FPS and interactive latency? The challenge lies in optimal decision making under a dynamic environment; *e.g.*, CPU, memory usage, network latency and computing requirement.

Low latency mobile Web AR: Latency is one of most important performance metrics for mobile Web AR over 5G networks because of the processing time at the front-end and back-end and the network delivery delay. Therefore, reducing the application's latency is a difficult task that includes the algorithms for video processing, dynamic resource allocation between front-end and mobile edge nodes, although 5G can mitigate the bottlenecks of latency at the access network.

Collaborating with neighbors: Device-to-Device (D2D) communication technique in the 5G network provides us with more opportunities for both development and deployment. *From development aspect:* D2D-based mobile Web AR implementation has a wide range of applications in many fields, such as entertainment, education. *From deployment aspect:* both computing and storage capabilities can be shared between different devices by using this approach, it therefore can achieve a more intelligent and flexible computing and communication paradigm for mobile Web AR applications.

Standardization: The MEC and D2D techniques have been proposed for many years and they require a standardization to align the development mobile Web AR over the 5G and B5G networks. There are also compatible issues between the implementations of WebGL (*e.g.*, Three.js) and the Web 3D models generated by tools such as 3D Max, Maya and Blender. This results in a reduction of animation effects and computing efficiency. Hence, a standard Web 3D object and development API on Web browsers is expected to be released to address these problems.

IX. CONCLUSION

Mobile Augmented Reality (MAR) supplements the reality with virtual objects and provides immersive experiences to end-users. Web-based MAR is a pervasive solution over WWW compared with wearable device-based and App-based ones. However, there are several technical challenges related to ease of deployment, cross-platform support, energy efficiency, standardization, and some of them due to the limitations of 3G/4G wireless networks. We design and implement a MWAR5 service framework, our experience shows that many of the limitations can be partially mitigated by new emerging 5G and B5G networks. We also shared our experiences with the development and deployment of MWAR5 by leveraging several techniques that hold high potentials in 5G networks. We believe that our research and development efforts in exploring the potential of mobile Web AR on 5G network are valuable experiences to share and that our findings of the challenges, opportunities, and future directions will be able to enlighten more research interests and efforts on delivering life-enriching, Web-based MAR experiences to the rapidly growing mobile and wireless business and consumer industry of the twenty-first century.

ACKNOWLEDGEMENTS

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, and in part by the BUPT Excellent Ph.D. Students Foundation under Grant CX2019213.

References

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

[2] M. Billinghurst, A. Clark, and G. Lee, "A Survey of Augmented Reality," *Foundations and Trends in Human-Computer Interaction*, vol. 8, no. 2-3, 2015, pp. 73-272.

[3] T. X. Tran, A. Hajisami, P. Pandey, and D. Pompili, "Collaborative Mobile Edge Computing in 5G Networks: New Paradigms, Scenarios, and

Challenges," *IEEE Commun. Mag.*, vol. 55, no. 4, 2017, pp. 54-61.

[4] Cisco Visual Networking Index (VNI): Global Mobile Data Traffic Forecast Update, 2016-2021 White Paper. Accessed: Sept. 25, 2018. [Online]. Available: https://www.cisco.com/c/en/us/solutions/service-provider/visual-networking-index-vni/index.html

[5] O. Tim et al., "What is Web 2.0: Design Patterns and Business Models for the Next Generation of Software," *Communications & Strategies*, no. 1, 2007, pp. 17-37.

[6] 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, 2019, pp. 651-666.

[7] R. Kooper and B. MacIntyre, "Browsing the Real-World Wide Web: Maintaining Awareness of Virtual Information in an AR Information Space," *International Journal of Human-Computer Interaction*, vol. 16, no. 3, 2003, pp. 425-446.

[8] S. Perry, "Wikitude: Android App with Augmented Reality: Mind Blowing," *digital-lifestyles. info*, vol. 23, no. 10, 2008.

[9] B. MacIntyre, A. Hill, H. Rouzati, M. Gandy, and B. Davidson, "The Argon AR Web Browser and standards-based AR application environment," *Proc. of the IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, pp. 65-74, 2011.

[10] J. Mellado. Js-ArUco - A JavaScript library for Augmented Reality applications - Google Project Hosting. Accessed: March 7, 2018. [Online]. Available: http://code.google.com/p/js-aruco

[11] I. Heikkinen. JSARToolKit - JavaScript port of FLARTOOLKit. Accessed: Feb. 12, 2018. [Online]. Available: https://github.com/kig/JSARToolKit

[12] J. Etienne. AR.js project homepage. Accessed: Feb. 13, 2018. [Online]. Available: https://github.com/jeromeetienne/AR.js

[13] 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, 2016, pp. 637-646.

[14] Y. Xu, "Smart caching for QoS-guaranteed device-to-device content delivery," *China Communications*, vol. 15, no. 1, 2018, pp. 128-139.

[15] A. Karhu, A. Heikkinen, and T. Koskela, "Towards Augmented Reality Applications in a Mobile Web Context," *Proc. of the 2014 Eighth International Conference on Next Generation Mobile Apps*, Services and Technologies, 2014, pp. 1-6.

[16] A. Bergkvist, D. C. Burnett, C. Jennings, A. Narayanan, and B. Aboba, "WebRTC 1.0: Real-time Communication Between Browsers," *Working draft*, W3C, vol. 91, 2012.

[17] C. Marrin, "WebGL specification," *Khronos WebGL Working Group*, 2011.

[18] J. G. Andrews, S. Buzzi, W. Choi, S. V. Hanly, A. Lozano, A. C. Soong, and J. C. Zhang, "What Will 5G Be?," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, 2014, pp. 1065-1082.

[19] 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, 2018, pp. 46-55.

[20] G. Schall, E. Mendez, E. Kruijff, E. Veas, S. Junghanns, B. Reitinger, and D. Schmalstieg, "Handheld Augmented Reality for Underground Infrastructure Visualization," *Personal Ubiquitous Comput.*, vol. 13, no. 4, 2009, pp. 281-291.

[21] S. Antoshchuk, M. Kovalenko, and J. Sieck, "Gesture Recognition-Based Human-Computer Interaction Interface for Multimedia Applications," *Digitisation of Culture: Namibian and International Perspectives*, 2018, pp. 269-286.

[22] K. Pulli, "Immersive Optical-See-Through Augmented Reality," *Proc. of the IEEE International Conference on Image Processing (ICIP)*, 2017.

[23] W. Narzt, G. Pomberger, A. Ferscha, D. Kolb, R. Müller, J. Wieghardt, H. Hörtner, and C. Lindinger, "Augmented reality navigation systems," *Universal Access in the Information Society*, vol. 4, 2006, pp. 177-187.

[24] M. Y. C. Yim, S. C. Chu, and P. L. Sauer, "Is Augmented Reality Technology an Effective Tool for E-commerce? An Interactivity and Vividness Perspective." Journal of Interactive Marketing, vol. 39, 2017, pp. 89-103.

[25] Z. He, L. Wu, and X. R. Li, "When art meets tech: The role of augmented reality in enhancing museum experiences and purchase intentions," *Tourism Management*, vol. 68, 2018, pp. 127-139.

[26] P. Chen, X. Liu, W. Cheng, and R. Huang, "A review of using Augmented Reality in Education from 2011 to 2016," *Innovations in Smart Learning*, 2017, pp. 13-18.

[27] M. Serino, K. Cordrey, L. McLaughlin, and R. L. Milanaik, "Pokémon Go and augmented virtual reality games: a cautionary commentary for parents and pediatricians," *Current opinion in pediatrics*, vol. 28, no. 5, 2016, pp. 673-677.

[28] S. Serafin, A. Adjorlu, N. Nilsson, L. Thomsen, and R. Nordahl, "Considerations on the use of Virtual and Augmented Reality Technologies in Music Education," *Proc. of the 2017 IEEE Virtual Reality Workshop on K-12 Embodied Learning through Virtual & Augmented Reality (KELVAR)*, 2017.

[29] Mozilla WebXR project homepage. Accessed: Aug. 14, 2018. [Online]. Available: https://deeplearnjs.org

[30] Google AR project homepage. Accessed: July 16, 2017. [Online]. Available: https://github.com/google-ar

[31] awe.js project homepage. Accessed: Sept. 19, 2018. [Online]. Available: https://awe.media

[32] WebAssembly Community Group. WebAssem-

bly Specification. Accessed: March 13, 2018. [Online]. Available: https://webassembly.github.io/spec

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

[34] X. Foukas, G. Patounas, A. Elmokashfi, and M. K. Marina, "Network Slicing in 5G: Survey and Challenges," *IEEE Communications Magazine*, vol. 55, no. 5, 2017, pp. 94-100.

[35] X. Liu, X. Wang, and Y. Liu, "Power allocation and performance analysis of the collaborative NOMA assisted relaying systems in 5G," *China Communications*, vol. 14, no. 1, 2017, pp. 50-60.

[36] P. Ren, X. Qiao, J. Chen, and S. Dustdar, "Mobile Edge Computing - a Booster for the Practical Provisioning Approach of Web-Based Augmented Reality," *Proc. of the 2018 Third ACM/IEEE Symposium on Edge Computing*, 2018, pp. 349-350.

[37] T. Kong, F. Sun, A. Yao, H. Liu, M. Lu, and Y. Chen, "RON: Reverse Connection with Objectness Prior Networks for Object Detection," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 5936-5944.

[38] J. Wan, D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. Zhang, and J. Li, "Deep Learning for Content-Based Image Retrieval: A Comprehensive Study," *Proc. of the 22nd ACM International Conference on Multimedia*, 2014, pp. 157-166.

[39] J. Chan, J. A. Lee, and Q. Kemao, "BIND: Binary Integrated Net Descriptors for Texture-Less Object Recognition," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 2068-2076.

[40] Y. Wu, J. Lim, and M.-H. Yang, "Online Object Tracking: A Benchmark," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 2411-2418.

[41] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *nature*, vol. 521, no. 7553, 2015, pp. 436-444.

[42] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN Features Off-the-Shelf: An Astounding Baseline for Recognition," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2014, pp. 806-813.

[43] Y. Kalantidis, C. Mellina, and S. Osindero, "Cross-Dimensional Weighting for Aggregated Deep Convolutional Features," *Proc. of the European Conference on Computer Vision (ECCV)*, 2016, pp. 685-701.

[44] M. Danelljan, G. Bhat, F. S. Khan, and M. Felsberg, "ECO: Efficient Convolution Operators for Tracking," *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 6638-6646.

[45] J. F. Henriques, R. Caseiro, P. Martins, and J. Batista, "High-Speed Tracking with Kernelized Correlation Filters," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 37, no. 3, 2015, pp. 583-596.

[46] Q. Zhu, B. Si, F. Yang, and Y. Ma, "Task offloading decision in fog computing system," *China Communications*, vol. 14, no. 11, 2017, pp. 59-68.

[47] F. Wei, S. Chen, W. Zou, "A greedy algorithm for task offloading in mobile edge computing system," *China Communications*, vol. 15, no. 11, 2018, pp. 149-157.

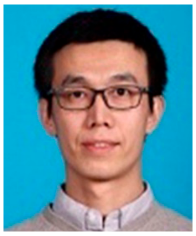
[48] G. Premsankar, M. Di Francesco, and T. Taleb, "Edge Computing for the Internet of Things: A Case Study," *IEEE Internet of Things Journal*, vol. 5, no. 2, 2018, pp. 1275-1284.

[49] Y. Huang, X. Qiao, P. Ren, L. Liu, C. Pu, and J. Chen, "A Lightweight Collaborative Recognition System with Binary Convolutional Neural Network for Web Augmented Reality," *Proc. of the IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, 2019, pp. 1497-1506.

Biographies

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 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. His current research interests include the future Internet, services computing, computer vision, augmented reality, virtual reality, and 5G networks.

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. His current research interests include the future Internet architecture, services computing, computer vision, machine learning, augmented reality, edge computing, and 5G networks.



Guoshun Nan, received his Ph.D. in Computer Science and Technology from the Beijing University of Posts and Telecommunications in 2017. His research interests lie in internet architecture, wireless multimedia communication, computer vision, machine learning, augmented reality, virtual reality and 5G networks. He has published papers in Computer Networks, ACM SIGCOMM Computer Communication Review, SIGCOMM, etc.

Ling Liu, (Fellow, IEEE) is currently a Professor at the School of Computer Science, Georgia Institute of Technology, Atlanta, GA, USA. 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 is the Editor-in-Chief of the ACM Transactions on Internet Technology.

Schahram Dustdar, (Fellow, IEEE) is currently a Professor of Computer Science with the Distributed Systems Group, TU Wien, 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, the IBM Faculty Award, and the IEEE TCSVC Outstanding Leadership Award. He is the Co-Editor-in-Chief of the ACM Transactions on Internet of Things and the Editor-in-Chief of Computing (Springer).

Junliang Chen, 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 Research Institute 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.