Computer Modeling in Engineering & Sciences

Tech Science Press

DOI: 10.32604/cmes.2022.020491

ARTICLE

Video Conference System in Mixed Reality Using a Hololens

Baolin Sun[1,#], Xuesong Gao[2,#], Weiqiang Chen[2], Qihao Sun[2], Xiaoxiao Cui[1], Hao Guo[1], Cishahayo Remesha Kevin[1], Shuaishuai Liu[2] and Zhi Liu[1,*]

[1]School of Information Science and Engineering, Shandong University, Qingdao, 266000, China

[2]State Key Laboratory of Digital Multi-Media Technology, Hisense Co., Ltd., Qingdao, 266000, China

*Corresponding Author: Zhi Liu. Email: liuzhi@sdu.edu.cn

#Baolin Sun and Xuesong Gao contributed equally to this paper

Received: 26 November 2021 Accepted: 16 March 2022

ABSTRACT

The mixed reality conference system proposed in this paper is a robust, real-time video conference application software that makes up for the simple interaction and lack of immersion and realism of traditional video conference, which realizes the entire process of holographic video conference from client to cloud to the client. This paper mainly focuses on designing and implementing a video conference system based on AI segmentation technology and mixed reality. Several mixed reality conference system components are discussed, including data collection, data transmission, processing, and mixed reality presentation. The data layer is mainly used for data collection, integration, and video and audio codecs. The network layer uses Web-RTC to realize peer-to-peer data communication. The data processing layer is the core part of the system, mainly for human video matting and human-computer interaction, which is the key to realizing mixed reality conferences and improving the interactive experience. The presentation layer explicitly includes the login interface of the mixed reality conference system, the presentation of real-time matting of human subjects, and the presentation objects. With the mixed reality conference system, conference participants in different places can see each other in real-time in their mixed reality scene and share presentation content and 3D models based on mixed reality technology to have a more interactive and immersive experience.

KEYWORDS

Mixed reality; AI segmentation; hologram; video conference; Web-RTC

1 Introduction

Image matting and compositing are image processing technologies that extract the attractive targets from the background in a static image or a video sequence and, at the same time, synthesize the separated attractive targets with other background images [1]. And image matting is a popular research direction of computer vision. With improved hardware computing power, image matting algorithms based on deep learning are widely used in film and television special effects, intelligent advertising design, and other fields to extract target objects [2].

Network video conferences have been widely used with the development of network technology, 5G communication, and the further improvement of infrastructure. In particular, triggered by the necessity of social distancing due to the current pandemic, people increasingly need video conference technology for various activities such as study and work [3]. People can easily communicate and conduct instant meetings and online learning remotely through the network video conference system. Admittedly, the technology of network video conference systems has been perfectly and widely used in various fields of social life. However, the traditional network video conference system has apparent shortcomings in many applications that require high three-dimensional sensory effects and can flexibly operate the 3D research model during the conference. At the same time, how to improve the authenticity of video conferencing and protect privacy has become a problem that needs to be paid attention to in the research of video conference system.

Mixed Reality MR [4] is a technology that combines physical and digital worlds to coexist with one another and deals with maximum user interaction in the real world compared to other similar technologies [5]. Mixed reality technology, including augmented reality and virtual reality, has developed rapidly and achieved many application results. They are widely used in scientific research, teaching, entertainment, and other social life, improving scientific research conditions and enriching people's daily cultural lives [6].

Generally speaking, traditional commercial video conference system mainly refers to 2D video conference, conference participants communicate with each other and share presentation documents through videos, such as Zoom, Cisco WebEx Teams, and Google Meet [7], which provides videotelephony and online chat services through a cloud-based peer-to-peer software platform and is used for teleconferencing, telecommunicating, distance education, and social relations [8]. Furthermore, a more immersive video conference system based on telepresence technology increases the natural perception of the environment through communication media. For telepresence technology [9], some mature commercial products have emerged, such as Polycom RealPresence Immersive Studio [10], Cisco Immersive TelePresence [11]. They generally need to build a conference room with relatively large space, adopt an integrated conference table, hidden microphone, and speaker, so that participants do not need to deliberately close to the microphone, maintain a natural way of communication, and provide excellent listening and position recognition function to realize a sense of space.

The commercial teleconference system already has a high immersion framework and essential interactive functions. However, the presentation dimension is still two-dimensional, far from actual participation in the conference. The two-dimensional scene taken by the camera is not displayed on the screen with a three-dimensional sense. With the development of 3D reconstruction technology, audio, and video transmission, and 3D presentation technology, some video conference systems are upgraded from 2D to 3D based on virtual reality and mixed reality technology. MeetinVr [12], a business meeting and VR collaboration software, offers human interaction more intuitive and effective than real scene by creating a new reality optimized for exceptional collaboration. Holoportation [13] presented an end-to-end system for augmented and virtual reality telepresence, demonstrates high-quality, real-time 3D reconstructions of an entire space, including people, furniture, and objects, using a set of new depth cameras, and allows users to wear virtual or augmented reality displays to see, listen and interact with remote participants in 3D. Mixed Reality Video Calling [14] for Microsoft Teams Together mode uses AI segmentation technology to digitally place participants in a shared background and make it feel like they are sitting in the same room with everyone else in the meeting. Still, Microsoft Teams Together mode is also a two-dimensional display through a PC or smart mobile phone. A low-cost mixed reality telepresence system presented by Joachimczak et al. [15] performs real-time 3D reconstruction of a person or an object and wirelessly transmits the reconstructions to Microsoft's Hololens.

At present, there are no real network video conference systems that combine AI segmentation technology based on computer vision with MR or AR head-mounted display devices. Furthermore, aiming at the immersion, interactivity, and authenticity of traditional network video conference systems, we propose implementing a novel network video conference system based on MR display devices and AI segmentation technology. We use the instance segmentation and image matting technology to extract the remote meeting participants from the background and interact with them in real-time in a mixed reality or argument reality head-mounted display device. Our mixed reality conference system is implemented by mixed reality technology and AI segmentation, thus the name of our system, **MRCS**.

2 System Requirements Analysis

To make the video conference system based on mixed reality and portrait matting described in this paper can be used in practice, and at the same time have a certain degree of advancement than the existing video conference system, we need to clarify the system requirements before development [16]. Requirement analysis is a detailed analysis of the functions that the system needs to realize, interface presentation, and interactive performance. After research and analysis, the requirements to be met by the system are confirmed, and then the design and development work can be gradually carried out. System requirements are divided into functional requirements and performance requirements [17]. Functional requirements ensure the integrity of system functions and have a guiding significance for the division of system modules; performance requirements ensure that the system can be implemented and used in practical applications.

2.1 System Function Requirements Analysis

The mixed reality conference system is mainly for users in different places. Through this conference system, they can appear in the same mixed reality conference room simultaneously, that is, you can observe the human portraits of other users. In addition to supporting traditional two-dimensional video conferences, it also supports manipulating 3D models in virtual scenes. Based on the above problems to be solved, we divide the functions of the mixed reality conference system into five parts:

(1) Mixed reality display function: When users use this conference system, they can see the person who breaks away from the remote environment and enters the local holographic space. Therefore, this system should provide a remote mixed reality display function.

(2) Real-time portrait matting function: As a conference system based on mixed reality, this system should provide a real-time portrait matting function, which can perform real-time portrait matting of users and register them in the mixed reality conference holographic space and human image compositing allows users to achieve an immersive experience.

(3) Real-time audio and video transmission function: Real-time audio and video transmission is the most basic functional requirement for a video conference system. At the same time, it supports the local LAN network and WAN network.

(4) Multi-person online function: To simulate the real meeting room scene, the system must support at least three people online simultaneously.

(5) Conference presentation function: As an improvement of the real conference, this system needs to provide essential conference functions, such as PowerPoint presentation and video playing.

2.2 System Performance Requirements Analysis

To be genuinely used in practical applications, the system must achieve the above functions and meet specific performance requirements.

(1) Real-time performance: For a video conference system, real-time performance is the essential requirement to ensure the regular progress of the meeting. Low delay can guarantee the quality of the meeting and improve the participant experience. Therefore, the modules of the system need to be closely connected, each module cannot have too much delay, and the total delay cannot exceed 1000 ms.

(2) Image matting frame rate: The mixed reality conference system allows users to merge their body images with the remote meeting scene after removing the background. The performance of the portrait matting algorithm will directly affect the system's fluency, and a smooth portrait display can significantly improve the user experience. Delays, freezes, or loss of portraits will substantially affect the user's immersion. Therefore, the frame rate of the image matting algorithm of this system is required to be no less than 60 FPS to ensure that the portrait model in the remote mixed reality scene can recover user actions and expressions smoothly.

(3) Scalability: The system will be expanded and upgraded as needed in practical application. Therefore, the system needs to have a modular structure, which is convenient for adding new functional modules and upgrading existing modules to have good scalability.

(4) Security: The meeting room requires a set of keys to establishing a session, and different sessions are independent of each other and will not cause interference.

3 Design of System Framework

The system framework and software interfaces determine the skeleton and skin of the entire system software. The data and control flow can be clarified through the system framework design. The whole system can be divided into modules to facilitate the design and implementation.

3.1 System Overview

According to the workflow of the mixed reality conference system, the system is organized into four parts: data layer, network layer, processing layer, and presentation layer; the data information flows from bottom to top. The data layer contains the underlying physical equipment and is mainly used for the input and integration functions of the original video and audio information. The network layer includes client nodes and cloud server programs. It is primarily used to receive audio and video information and interactive information from the data layer and send the processed audio and video data to the presentation layer. The processing layer splits the received data, the video flows to the image matting module, and the interactive instruction flows to the conference presentation object function module. The presentation layer is used for mixed reality display of the entire system, including user interface, mixed reality scenes, human images, and conference presentation objects. Fig. 1 shows the workflow of our MRCS system, and the system architecture diagram is shown in Fig. 2.

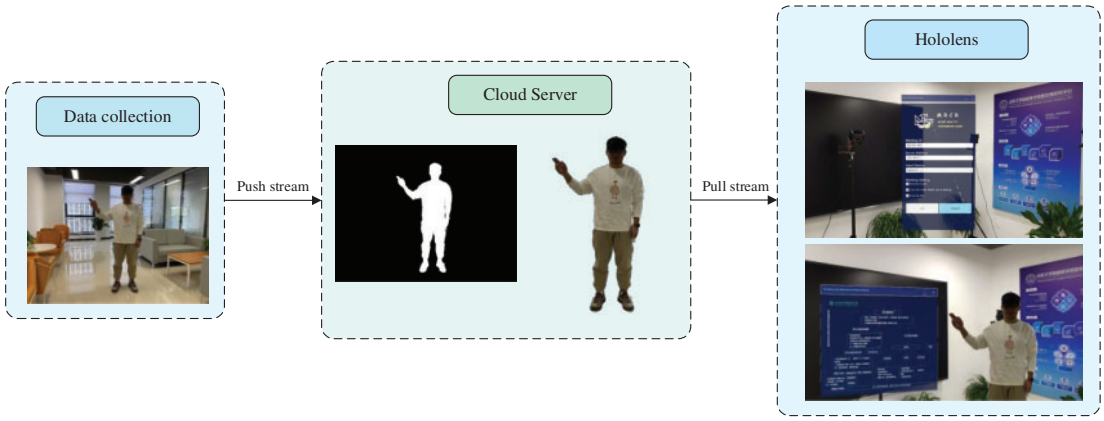


Figure 1: Workflow of the mixed reality conference system

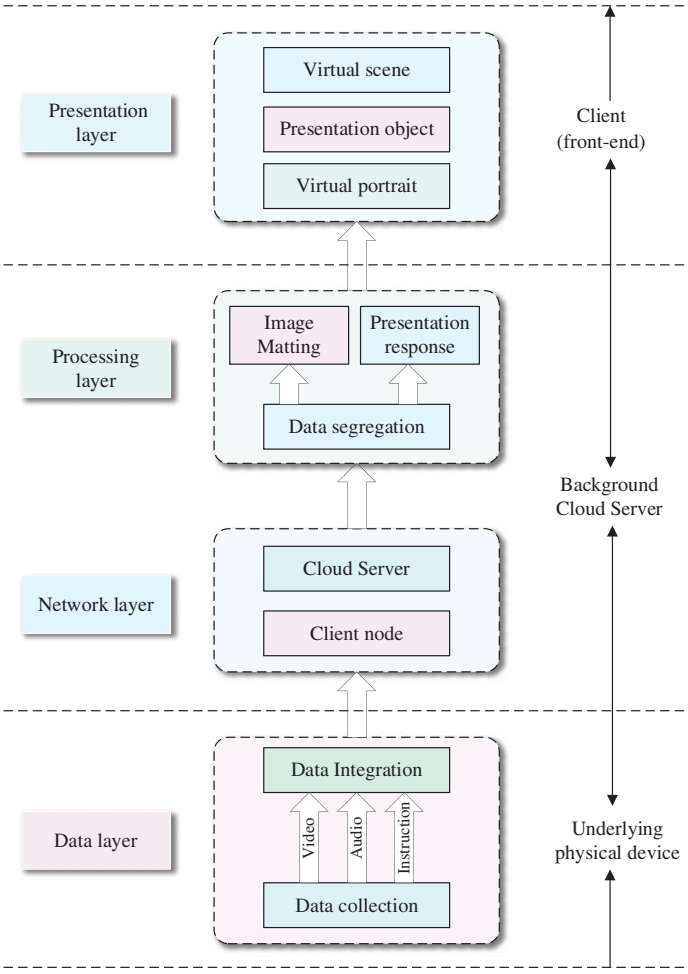


Figure 2: MRCS system architecture diagram

3.2 Design of Data Layer

The data layer is mainly for data collection and data integration. The video is collected through the camera, the video data format is RGB, the audio is collected through the microphone, and the audio data format is PCM. Encode the video stream from RGB format to H.264 [18], and encode audio data from PCM format to AAC [19]. Optionally, screen recording information can be collected and encoded in the same way. Encode video and audio files into a multimedia container format with streaming media characteristics. Furthermore, this system is based on the WebRTC framework and pushes the audio and video streams to the cloud server for subsequent processing.

3.3 Design of Network Layer

This system selects WebRTC [20] as the network framework for application development. WebRTC (Web Real-Time Communication) is a real-time communication technology without plug-ins introduced by Google. It does not require any other plug-ins or applications for audio, video streaming, and data sharing, allowing browsers to directly exchange media information with other browsers in a peer-to-peer manner while providing higher security than other various streaming media systems. WebRTC provides the core technology of video conferencing. It currently supports VP8 and H.264 video codec standards. At the same time, WebRTC has realized audio and video codec transmission, NAT (Network Address Translation) penetration, and other functions, so we can focus on developing transmission content under the framework of WebRTC.

Because WebRTC is based on P2P connection, the conference network topology of the system is a Mixer structure, that is, a fully connected structure. The design takes the cloud server as the center and establishes a peer-to-peer (P2P) data transmission with each client, and clients do not interfere with each other, which increases the system stability. The WebRTC mixer structure is shown in Fig. 3; through the central cloud server, clients do not need to establish a P2P connection with other clients directly but only establish a P2P connection with the central cloud server. After receiving the audio and video streams from the client, the server is responsible for processing and forwarding them to other clients. Before transmission, it can also mix or transcode multiple users audio and video streams. Through the WebRTC Mixer structure, the system can easily make different customization requirements on the server-side. But the Mixer structure may introduce additional delay and loss of video and audio quality because of decoding and recoding on the server.

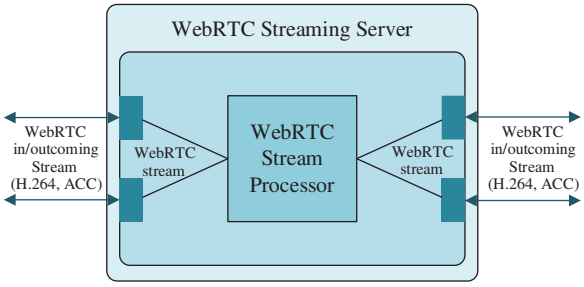


Figure 3: The mixer structure base on WebRTC

The system is developed on Unity3D [21], we select the plugin WebRTC Video Chat on the Unity Asset Store as the framework and expand on this basis. The plug-in allows users to stream audio and video between the two programs, send text messages, and have a complete signaling server program.

Furthermore, in the process of audio and video transmission, we protect the privacy of MRCS participants by E2EE (End to End Encryption) [22], which is one of the most reliable methods to ensure the security of data exchange in the field of information security. We implement the application encryption class and encryption function based on WebRTC API. We generate two encryption keys for each user: the public key and private key; both keys are generated using the PGP (Pretty Good Privacy) [23] encryption algorithm, which has not been cracked since it was released in 1991. The public key encrypts information and exchange between user applications with audio and video transmission. The private key decrypts the information data encrypted by the public key. The private key is stored locally and not sent from the user's device. Since the server does not participate in the key generation process, what the server receives and sends is the user encrypted message. Therefore, even if the user information is leaked on the server side, the user information cannot be parsed out.

3.4 Design of Processing Layer

The processing layer is the main part of the system, mainly divided into two parts: the portrait matting module and the human-computer interaction module. The portrait matting module first disassembles the mixed frame from the client and then extracts the human body image from the background based on the image matting algorithm. The human-computer interaction module is mainly used to operate presentation objects and 3D models in holographic scenes. The system pulls the multimedia stream pushed by the data layer in the cloud server, decomposes the stream in the multimedia container format into video data (H.264) and audio files (AAC), and performs GPU hardware decoding on the H.264 video data to the video stream with the video format as RGB [24]. Through the parsed video stream, the portrait matting and hand posture estimation [25] are performed subsequently.

We use AI segmentation technology (image segmentation algorithm and human matting method) for human target extraction and background replacement in the processing layer. With image segmentation, each annotated pixel in an image belongs to a single class, often used to label images for applications requiring high accuracy. The algorithm is manually intensive because it requires pixel-level accuracy. The output of image segmentation is a mask that outlines the object shape in the image. Image segmentation annotations come in many different types, such as semantic segmentation, instance segmentation, panoptic segmentation; the practice of image segmentation generally describes the need to annotate every pixel of the image with a class [26]. Instance segmentation comes from object detection and semantic segmentation. Object detection or localization provides the classes and location of the image objects. Semantic segmentation gives fine inference by predicting labels for every pixel in the input image. Each pixel is labeled according to the object class enclosed within it. Furthering this evolution, instance segmentation gives different labels for separate objects belonging to the same class [27]. Hence, we tried to use an instance segmentation algorithm to simultaneously solve the problem of human object detection and semantic segmentation. Our Instance segmentation algorithm is based on SOLO [28] and has a lightweight implementation. The model architecture diagram is shown in Fig. 4.

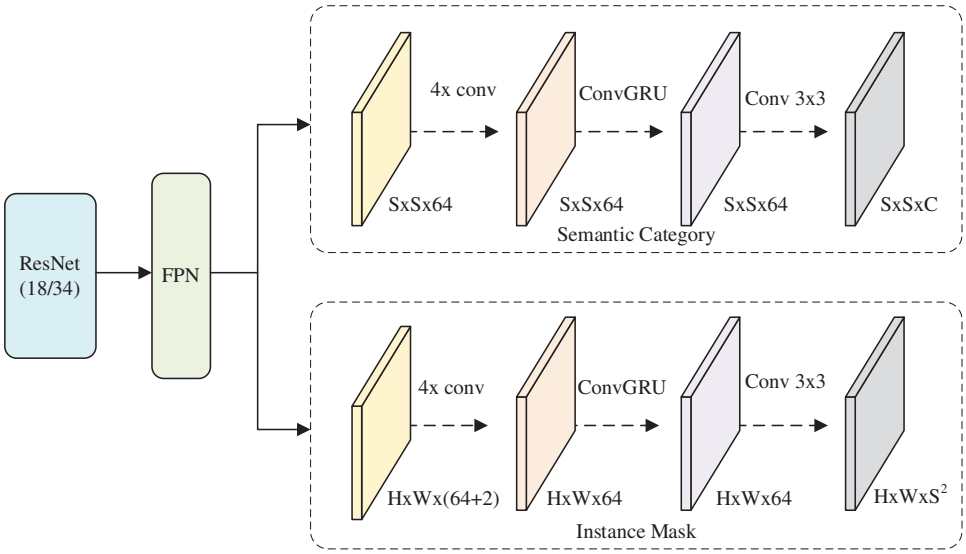


Figure 4: Our instance segmentation model architecture diagram

Our instance segmentation model draws on the model design concept of the MMDetection [29] framework and divides the model into three parts: backbone, neck, and head. The model takes resnet50 as the backbone network for feature extraction, and the neck network uses FPN [30] to generate feature maps of different sizes with fixed 64 channels at each level. Each FPN network is followed by a head component that predicts the object category and mask. The head component is divided into two separate branches: semantic category and instance mask branch. The semantic category branch meshes the FPN feature map to $S \times S \times 64$, where a feature map is divided into a grid of $S \times S$ cells and then extracts features through four 3×3 convolution layers. Next, fixed multiple frames are input to ConvGRU [31] layers to take temporal information into account to improve video instance segmentation quality. Finally, the features are output as $S \times S \times C$ through the 3×3 convolution layer, where C is the number of object classes. The instance mask branch first performs CoordConv on the highest level features of the FPN. It then extracts features through four 3×3 convolution layers and ConvGRU layers to take inter-frame correlation into account. Finally, the features are output as $H \times W \times S^2$ through convolution layers. In our lightweight instance segmentation model based on SOLO, we adopt ConvGRU to aggregate temporal. GRU network is a recurrent neural network, which is a variant of LSTM (Long Short-Term Memory) [32], and ConvLSTM [33] uses a convolution kernel to replace the full connection layer in LSTM. In addition, the ConvGRU network is more parameter efficient than ConvLSTM. Formally, ConvGRU is defined as:

$$\begin{cases} z_t = \sigma\left(w_{zx} * x_t + w_{zh} * h_{t-1} + b_z\right) \\ r_t = \sigma\left(w_{rx} * x_t + w_{rh} * h_{t-1} + b_r\right) \\ o_t = \tanh\left(w_{ox} * x_t + w_{oh} * \left(r_t \cdot h_{t-1}\right) + b_o\right) \\ h_t = z_t \cdot h_{t-1} + (1 - z_t) \cdot o_t \end{cases} \tag{1}$$

where w and b are the convolution kernel and bias term. h_t is the hidden state which is used as both the output and the recurrent state to the next time step as h_{t-1}.

The training loss function is defined as follows:

$$L = L_{cate} + \lambda L_{mask} \tag{2}$$

where L_{cate} is the conventional focal loss for semantic category classification, L_{mask} is the dice loss for mask prediction:

$$L_{mask} = \frac{1}{N_{pos}} \sum_{k} I_{\{p_k^* > 0\}} d_{mask} \left(m_k, m_k^*\right) \tag{3}$$

N_{pos} is the number of positive samples, k is the index of the grid cells, p_k^* and m_k^* represent category and mask target respectively. And $d_{mask}(\cdot, \cdot)$ is the dice loss which is defined as:

$$d_{mask}(p,\ q) = 1 - \frac{2 \sum_{x,y} \left(p_{x,y} \cdot q_{x,y}\right)}{\sum_{x,y} p_{x,y}^2 + \sum_{x,y} q_{x,y}^2} \tag{4}$$

where $p_{x,y}$ and $q_{x,y}$ refer to the value of pixel located at (x, y) in predicted soft mask p and ground truth mask q.

Our instance segmentation model is firstly trained on COCO and VOC datasets, and then the pre-trained model is next trained on the portrait segmentation dataset of Hisense. COCO dataset provides 80 categories, and we only use the samples of humans to train our model. Similarly, we also only use the samples of humans in the VOC dataset to train our model. The portrait segmentation dataset of Hisense has 50000 pictures in the form of video frames. The human object is collected at a distance of 1–5 meters from the camera. The annotated contour of human objects is large and clear, and the pictures are 4 K (3840 × 2060) resolution. And the scenes are living rooms, bedrooms, study rooms, and offices. The scene types are similar and applicable to the video conference portrait segmentation model designed in this paper. We divide the dataset into 40000/5000/5000 clips for train/val/test splits. By using our instance segmentation model based on SOLO, we achieve HD (1920 × 1080) 67 FPS and 4 K (3840 × 2060) 59 FPS on Nvidia Tesla V100, which the accuracy of the model decreased to a certain extent within the acceptable range.

We also use the human video matting method to extract human objects from the image background. Still, this method cannot give different labels for separate instances of objects belonging to the same class. Image matting is the process of predicting the alpha matte and foreground color from an input frame [34]. Formally, a frame I can be viewed as the linear combination of a foreground F and a background B through an α coefficient:

$$I = \alpha F + (1 - \alpha) B \tag{5}$$

By extracting α and F, we can composite the foreground object to a new background, achieving the background replacement effect.

Portrait matting aims to predict a precise alpha matte that can be used to extract people from a given image or video [35]. In the MRCS system, we use RVM [36] (Robust Video Matting) as the image matting algorithm. RVM is a robust, real-time, high-resolution human video matting method to achieve new state-of-art performance. We do not modify the network structure of the model because the simplification of the model structure does not lead to the improvement of the model speed but leads to the decline of model accuracy. Based on the RVM author's pre-trained model, we train and fine-tune the model on our dataset (the portrait segmentation dataset of Hisense) since our dataset has a large target of human objects, so it is more suitable for human subjects in a meeting room scene. By using the RVM algorithm and training it on the portrait segmentation dataset of Hisense, we achieve HD (1920 × 1080) 154 FPS and 4 K (3840 × 2060) 117 FPS on Nvidia Tesla V100, which is considered real-time for video conference and meet the requirements of the matting frame rate. To simulate the

indoor scene of the video conference, our test results for indoor multi-targets are shown in Fig. 5b, and Fig. 5a is the original input image.

(a) (b)

Figure 5: The matting result of indoor multi-target. (a) the original input image from the camera, (b) the matting result of the original input image

Table 1 compares our modified lightweight method based on SOLO and RVM trained on the portrait segmentation dataset of Hisense at low-resolution (1920×1080) and high-resolution (3840×2060).

Table 1: The comparison between our modified model and the original model at low-resolution and high-resolution

Method	Resolution	mIOU	FPS
SOLO	HD	83%	28
	4 K	91%	31

(Continued)

Table 1 (continued)

Method	Resolution	mIOU	FPS
RVM	HD	86%	154
	4 K	84%	117
SOLO (Ours)	HD	89%	67
	4 K	90%	59
RVM (Ours)	HD	92%	151
	4 K	90%	112

Human-computer interaction uses voice and gestures as interactive commands to control PowerPoint presentations and video playback. Since we use the commercial mixed reality device Microsoft Hololens2 [37] as the system presentation layer implementation tool, the speech recognition in human-computer interaction is implemented by Microsoft Azure Natural Language Understanding Model (LUIS) [38]. By adding custom command keywords to the LUIS system applications, we build application modules that understand natural language to identify user intentions and extract key information from conversation phrases and implement corresponding methods to perform the corresponding operation. Since we use Unity3D as the development and design tool, the gesture recognition function is implemented by Unity high-level composite gesture API (Gesture Recognizer) to recognize the user spatial input gestures [39]. We only need a few steps required to capture gestures using a Unity3D gesture recognizer: create a new Unity3D gesture recognizer object, specify which gestures to watch for, subscribe to events for those gestures, start capturing gestures. In addition to using the grab, click, and stare gestures that come with the Hololens2 system, we have designed operation gestures including left swipe, right swipe, pull up, pull-down, and so on.

3.5 Design of Presentation Layer

The presentation layer is the critical part of the MRCS system, which directly affects interactivity and immersion, and determines the user experience. We use mixed reality technology to provide user with an immersive experience during the meeting. The mixed reality device used in the MRCS system is Microsoft Holelens2, a pair of mixed reality smartglasses developed and manufactured by Microsoft. It is widely used in manufacturing, engineering, healthcare, and education [40]. The Hololens2 is a combination of the waveguide and laser-based stereoscopic full-color mixed reality smartglasses [41] features an inertial measurement unit (IMU) (which includes an accelerometer, gyroscope, and a magnetometer), four "environment understanding" sensors for spatial tracking, four visible light cameras for head tracking, two infrared radiation (IR) for eye tracking, 1-MP time-of-flight (ToF) depth sensor, an 8-megapixel photographic video camera, a four-microphone array, and an ambient light sensor [42]. What's more, the Hololens2 has a diagonal field of view (FOV) of 52 degrees, improving over the 34 degrees field of view of the first edition of the Hololens. In addition to an Intel Cherry Trail SoC containing the CPU and GPU, the Hololens2 features a custom-made Microsoft Holographic Processing Unit (HPU) [43]. Although there are some mature commercial AR or MR smart glasses, like Google Glass [44], HTC VIVE [45], Meta2 [46], and so on. Google Glass is a lightweight smart glasses (an optical head-mounted display designed in the shape of a pair of glasses)

that can only display digital information in a small range. Meta2 and HTC Vive are similar to Hololens, but Hololens far exceed Meta2 and HTC in the combination of reality and virtual information, and Hololens is more convenient for engineers to develop. Taking all these into consideration, we choose Microsoft Hololens2 as the MRCS system mixed reality hardware device, mainly for the following two reasons: on the one hand, the Hololens is one of the best mixed reality smartglasses at present, one the other hand, it runs the Windows Mixed Reality platform under the windows10 computer operating system and can be developed by Unity3D and Visual Studio, so it is more suitable for developing our MRCS system. The top and side views of the Microsoft Hololens2 are shown in Fig. 6.

Figure 6: Top and side views of the microsoft Hololens2

According to MRCS system functional requirements, before establishing a meeting room, we need to enter the unique number of the meeting room, so that only users who enter the same number can enter the same mixed reality scene. When designing the software interface, the interface is divided into a two-dimensional interface which is used to realize user login function and system input device selection, and a three-dimensional interface which is used for mixed reality presentation after entering the meeting room. The interface is designed following the principles of conciseness, complete functions, easy operation, and learning, which is convenient for users to get started.

The primary function of the initial two-dimensional interface is to guide users to create and join a mixed reality conference room, as shown in Fig. 7. This interface is displayed on the user's computer screen. The interface is divided into two states, the initial state, and the conference state. In the initial state, the user has not yet entered the meeting room, so you need to enter the room number and server address on this interface, and then click the "Join" button to enter the designated meeting room. At the same time, the device drop-down button can select the device to collect video and audio data. After entering the meeting room, the interface becomes the meeting state.

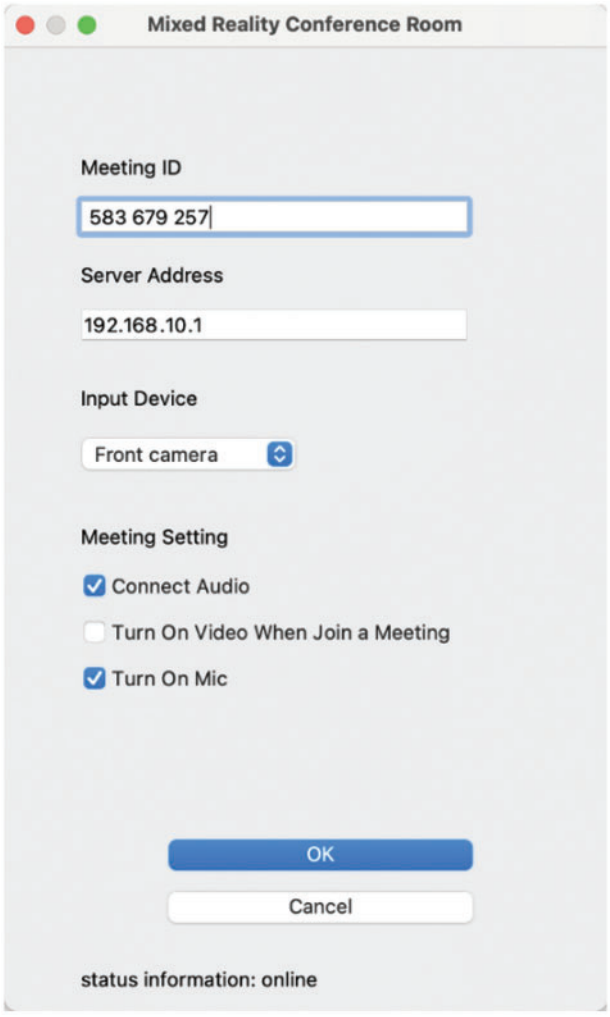


Figure 7: The two-dimensional interface of the conference system

At the same time, as shown in Fig. 8, we have also implemented the mixed reality login window interface, which allows users to join the MRCS mixed reality meeting completely away from their personal computers. The mixed reality interface has all the same functions as the two-dimensional interface in Fig. 7.

The mixed reality device used in the MRCS system is Microsoft Hololens2, which is developed based on Unity3D and MRTK [47] (Microsoft's Mixed Reality Toolkit). Use Mixed Reality WebRTC to realize peer-to-peer real-time audio and video communication, and the application of MRCS system on PC is developed through Qt5.

After the user successfully creates or joins the room, he enters the working interface. The user can wear the mixed reality device at this time. Users can experience the conference room function of the MRCS system in this interface.

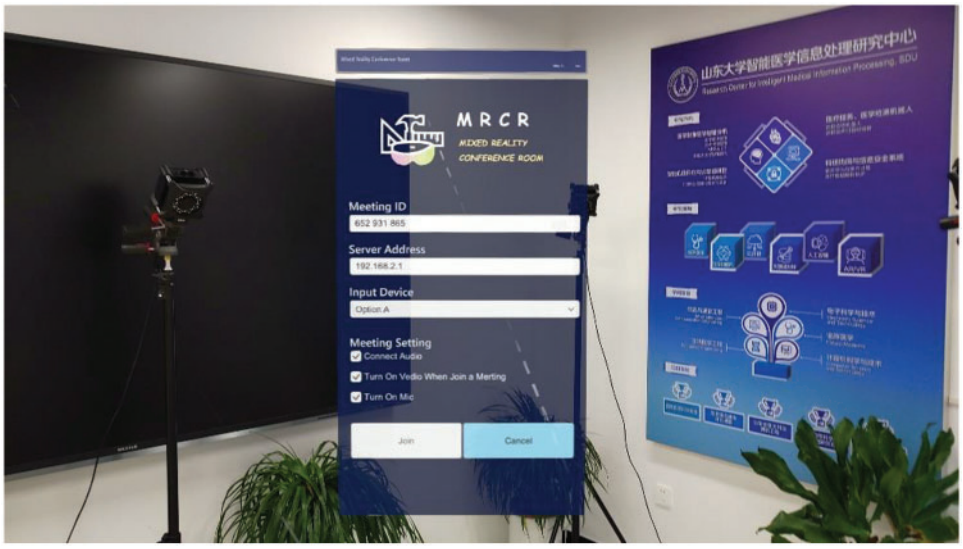
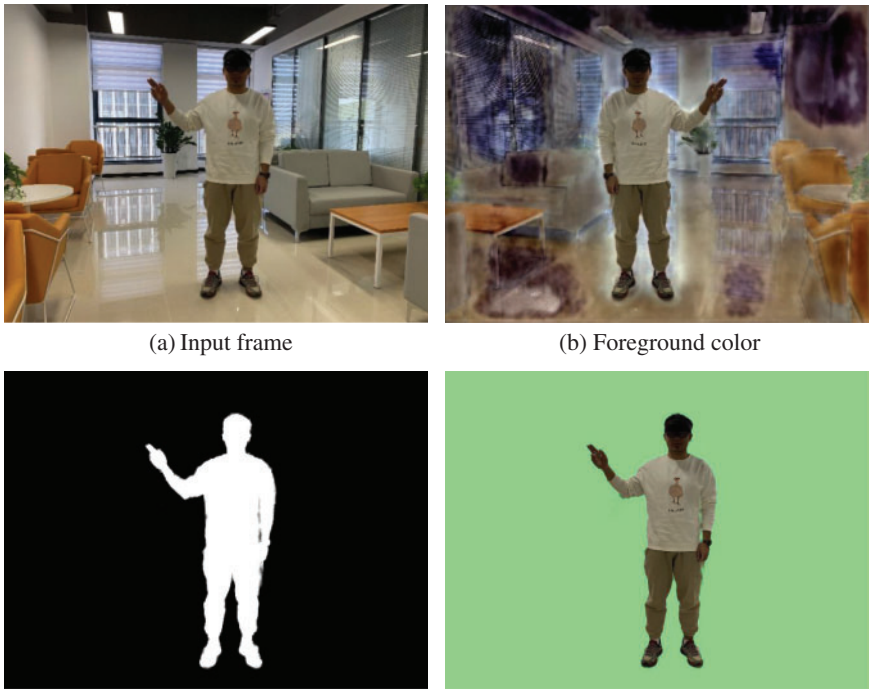


Figure 8: Mixed reality interface of the conference system

In the remote meeting scene, as shown in Fig. 2, the camera collects the real-time video stream of the remote meeting human subjects, encodes the video stream from RGB to H.264, and then pushes it to the server of the network layer. The cloud server pulls the video stream and decodes it. The video is processed through RVM or the instance segmentation model. The MRCS system performs portrait matting to get the alpha matte and the foreground image, and further obtains the accurate matting video stream of the portrait of the remote meeting object through the alpha value and the foreground image, and encodes the video stream from RGB to H.264 on the cloud server again.

The mixed reality device pulls the matting video stream from the network layer server in real-time. It synchronously displays the human image constructed by the local mixed reality device in the conference scene. The final display of the MRCS system is shown in Fig. 10. There are three main parts of mixed reality design: the display of human subjects in the mixed reality device, the live sharing of 3D models in the mixed reality device, the human-computer interaction in the mixed reality device.

As shown in Fig. 3, each mixed reality device (Microsoft Hololens2) can pull the video and audio stream from the server in real-time through the WebRTC Mixer structure. Firstly, the Hololens2 obtains video and audio streams through local media devices; Secondly, establish a WebRTC P2P connection with the remote cloud server; Then start or close a session with signaling communication; Finally, the Hololens2 and cloud server exchange video and audio streams. Note that the video stream that the Hololens2 pulls from the cloud server is only the final prediction foreground, that is, the human subjects, as shown in Fig. 9d. We add a green background to the final prediction result for ease of display in the paper. And then, take the ground and the Hololens2 as the coordinate origin to establish the world coordinate system. Display the human subject video stream at fixed coordinates, the system default is 2 meters in front of the Hololens. What's more, the location and size of the human subjects can be changed easily by the user who wears the Hololens through the voice and gestures command.



(a) Input frame (b) Foreground color

(c) Alpha matta (d) Final prediction foreground

Figure 9: (a) is the input frame collected from the camera. Through performing portrait matting, the foreground color, alpha matte is shown in (b) and (c). And (d) is the final prediction result with green background

Figure 10: The final display result of the MRCS conference system, the person in the image is the remote conference object, and the blue background window is the conference presentation object

For the live sharing of 3D models, we build a multi-user experience using Photon Unity Networking [48] (PUN), one of several networking options available to mixed reality developers to create shared experiences [49]. Firstly, we set up Photon Unity Networking to get the PUN App ID; Secondly, Through setting the same PUN App ID, we connect the unity project (the Hololens project) to the PUN application to connect multiple devices; Thirdly, we config PUN to instantiate the 3D models which need to be live shared. Finally, we integrate Azure spatial anchors into a shared experience for multi-device alignment. All participants of the MRCS conference system can collaborate and view each other's interactions and see the 3D models move when other users move them.

Human-computer interaction uses voice and gestures as interactive commands to control PowerPoint presentations, 3D models, and video playback. We use Microsoft Azure Natural Language Understanding Model (LUIS) to train and recognize speech commands. Adding custom command keywords to the LUIS system applications builds application modules that understand natural language to identify user intentions, extract key information from conversation phrases, and implement corresponding methods to perform the corresponding operation. Meanwhile, based on Hololens system gestures, we customize MRCS system gestures through unity high-level composite gesture API.

4 System Evaluation

This section mainly tests and evaluates the MRCS conference system based on mixed reality technology. The system performance evaluation primarily focuses on the system real-time performance, image quality, render frame rate when the MRCS system is running. In this paper, the hardware devices used in the MRCS system consist of a monocular camera, audio input device, client host, server host, mixed reality device, network devices. The specific devices and parameters are shown in Table 2.

Table 2: Hardware configuration of the MRCS system

Item	Parameters
Camera	Resolution: 1920 × 1080; FPS: 30 frames per sec
Audio input	Sample frequency: 32 KHz; Full duplex mode
Client host	GPU: Nvidia GTX1060 6 GB; CPU: Core i5-8500
Server host	GPU: Nvidia GTX1080Ti 11 GB; CPU: Intel Xeon CPU E5-2620
Mixed reality device	Microsoft Hololens2; Refresh rate: 90 Hz; Resolution: 1440 × 936
Network devices	Download speed: 300 Mbps; Upload speed: 60 Mbps

Through the analysis of the system, the time delay of MRCS mainly consists of five parts: processing delay, algorithm delay, network delay, transmission delay, buffer delay. Processing delay refers to the time required for video and audio codec algorithms which is equal to the complexity of the codec algorithm divided by the execution speed of the hardware encoder and multiplied by video length. Algorithmic delay refers to the time consumed by segmentation of human image instances in the video which has been discussed in Section 3.4. Network delay is the time taken for the signal to reach the destination through the physical transmission medium, which depends on the quality of network connection and communication distance. Transmission delay refers to the time required to transmit the smallest bit of a decodable video signal. Buffer delay is caused by the storage of real-time data and the unpredictability of audio and video arrival time. Due to the WebRTC framework used in the system, it is not convenient to calculate the delay of five subparts separately. We calculate the overall time-consuming of the MRCS system. First, the Hololens A sends data to the Hololens B records the sending time t_1. Then Hololens B returns the data after receiving the data, and Hololens A records the receiving time t_2 after receiving the complete returned data. Through the following formula, the overall transmission delay RTT can be obtained:

$$RTT = (t_2 - t_1)/2 \tag{6}$$

Repeat the experiment 1000 times and get an average system time delay $\overline{RTT}$ of 537 milliseconds.

Image quality includes two parts: the image quality that the portrait instance segmentation is superimposed on the background of the physical environment, the quality of the PowerPoint or 3D models that need to be displayed. As shown in Figs. 9 and 10, the portrait result of instance segmentation is almost the same as the original except for the edge of the portrait. Because of the instance segmentation algorithms, the edge of the portrait is a little blurred or unclear. Because PowerPoint or 3D models are built by Photon Unity Networking (PUN), they are no different from the original.

For the render frame rate, we design a program to monitor it in real-time. When the program is running, there is no delay when the user wears the HoloLens, as shown in Fig. 11, the render frame rate of the system is stable at 60 fps while MRCS is running, but there is a significant drop in the operation of 3D models, but the drop range is within the acceptable range. Fig. 12 shows the field of view of Hololens while MRCS is running, and the left and right views are consistent with the head movement.

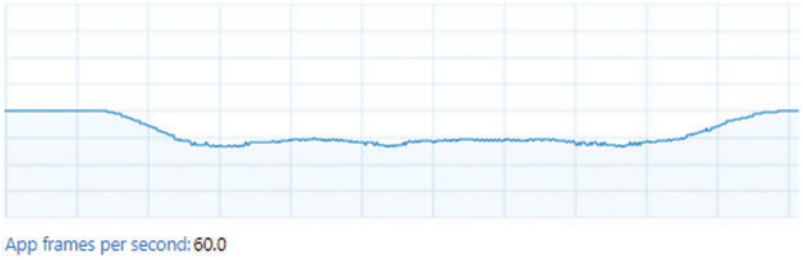


Figure 11: The MRCS system frames per second

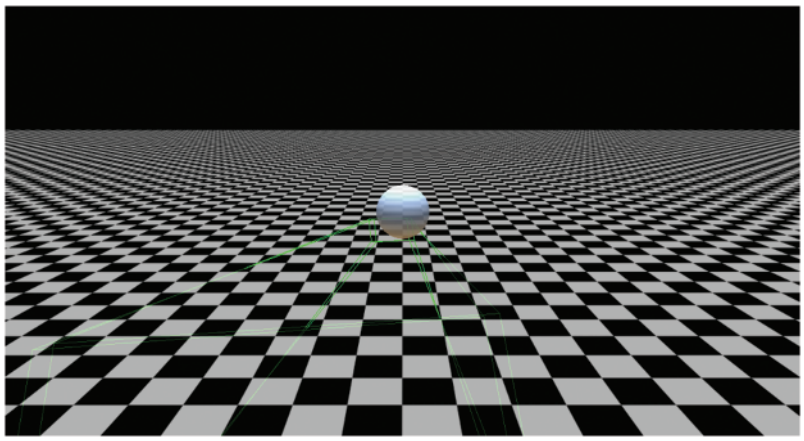


Figure 12: Field of view test for Hololens while MRCS is running

5 Conclusions

In this paper, a network video conference system based on mixed reality and portrait matting is designed and implemented to improve the authenticity and immersion of traditional video conferences, which can make up for the shortcoming of traditional video conferences. The system realizes the functions of traditional video conference from a higher dimension, allowing users to participate in a conference in a new way, and is a breakthrough in the implementation of mixed reality technology. With the development of 5G communication technology, the information carried on the network will inevitably be upgraded to a higher dimension, not only simple audio and video but also the integration of three-dimensional information. Finally, it will be perfectly presented to people through mixed reality technology. Starting from the technical implementation of mixed reality application, this paper discusses the design of mixed reality video conference based on portrait matting, audio and video transmission based on traditional video coding and decoding technology, and human-computer interaction in mixed reality, and finally realized the mixed reality remote video conference system MRCS.

The MRCS system implements the remote video conference system in the mixed reality scene based on Microsoft Hololens2, realizes the integration of virtual and real scenes, improves the immersion and interactivity significantly during the conference. The system combines the AI instance segmentation algorithm with mixed reality technology, "moves" the remote conference participants to the current physical environment, which significantly improves the authenticity and immersion of the conference. And the MRCS system builds a multi-user live shared experience using Photon Unity Networking (PUN) based on the Hololens2, allowing users to share the movements of 3D models so that all MRCS conference participants can collaborate and view each other interactions. However, the MRCS system has four drawbacks due to the limited hardware and software. First, there is a system time delay in the MRCS system due to the network transmission, video and audio codecs, and portrait instance segmentation. Second, the camera used in the system is monocular, so the system cannot carry out an effective 3D reconstruction of the human subjects. The MRCS can be improved and upgraded later by using a binocular camera and depth sensor. Thirdly, when the human body moves fast, the edge of the human subjects obtained by the image matting algorithm is blurred, because of the portrait instance segmentation algorithm. Finally, the mixed-reality device used in the system is Microsoft Hololens2, which is relatively bulky and heavy, and needs to be worn for a long time. And

the field of view (FOV) of Hololens2 is only 52 degrees, which may affect user experience. But the Hololens2 is the most advanced mixed reality smartglasses, so there is no better choice in the mixed reality hardware device. For future applications, we will develop instance segmentation algorithms to improve the quality of human image matting, use the binocular camera and depth sensor to get the image depth information for three-dimensional reconstruction, and explore mixed reality hardware devices to improve the user experience.

Data Availability: The data and codes to support the findings of this study are available from the corresponding author upon request.

Funding Statement: The authors would like to acknowledge the support by State Key Laboratory of Digital Multi-Media Technology of Hisense and School of Information Science and Engineering of Shandong University. This work was supported in part by the Major Fundamental Research of Natural Science Foundation of Shandong Province under Grant ZR2019ZD05; Joint fund for smart computing of Shandong Natural Science Foundation under Grant ZR2020LZH013; Open project of State Key Laboratory of Computer Architecture CARCHA202002; Human Video Matting Project of Hisense Co., Ltd. under Grant QD1170020023.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Zhu, Q., Shao, L., Li, X., Wang, L. (2015). Targeting accurate object extraction from an image: A comprehensive study of natural image matting. *IEEE Transactions on Neural Networks and Learning Systems, 26(2),* 185–207. DOI 10.1109/TNNLS.2014.2369426.
2. Chen, Q., Ge, T., Xu, Y., Zhang, Z., Yang, X. (2018). Semantic human matting. *Proceedings of the 26th ACM International Conference on Multimedia*, pp. 618–626. Seoul, Korea.
3. Assaqty, M. I. S., Gao, Y., Musyafa, A., Wen, W., Wen, Q. (2020). Independent public video conference network. *2020 International Seminar on Application for Technology of Information and Communication (iSemantic)*, pp. 142–148. Semarang, Indonesia.
4. Speicher, M., Hall, B. D., Nebeling, M. (2019). What is mixed reality? *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pp. 1–15. Glasgow, UK.
5. Rokhsaritalemi, S., Sadeghi-Niaraki, A., Choi, S. M. (2020). A review on mixed reality: Current trends, challenges and prospects. *Applied Sciences, 10(2),* 636. DOI 10.3390/app10020636.
6. Maas, M. J., Hughes, J. M. (2020). Virtual, augmented and mixed reality in K–12 education: A review of the literature. *Technology, Pedagogy and Education, 29(2),* 231–249. DOI 10.1080/1475939X.2020.1737210.
7. Archibald, M. M., Ambagtsheer, R. C., Casey, M. G., Lawless, M. (2019). Using zoom videoconferencing for qualitative data collection: Perceptions and experiences of researchers and participants. *International Journal of Qualitative Methods, 18,* 1609406919874596. DOI 10.1177/1609406919874596.
8. Singh, R., Awasthi, S. (2020). *Updated comparative analysis on video conferencing platforms-Zoom, Google Meet, Microsoft Teams, WebEx Teams and GoToMeetings*, pp. 1–9. EasyChair: The World for Scientists.
9. Abdullah, A., Kolkmeier, J., Lo, V., Neff, M. (2021). Videoconference and embodied VR: Communication patterns across task and medium. *Proceedings of the ACM on Human-Computer Interaction, 5,* 1–29. DOI 10.1145/3479597.
10. Polycom, Inc. (2021). Polycom RealPresence Immersive: Video conferencing & telepresence solutions. https://www.poly.com/us/en/products/video-conferencing/studio.

11. Cisco, Inc. (2021). Immersive TelePresence Systems. http://www.cisco.com/c/en/us/products/collaboration-endpoints/immersive-telePresence/index.html.
12. MeetinVR, Inc. (2021). Business meetings & collaboration in VR. https://www.meetinvr.com.
13. Orts-Escolano, S., Rhemann, C., Fanello, S., Chang, W., Kowdle, A. et al. (2016). Holoportation: Virtual 3d teleportation in real-time. *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, pp. 741–754. Tokyo, Japan.
14. Hubbard, M., Bailey, M. J., Hellebro, M. (2021). Meetings in teams. *Mastering Microsoft Teams*, pp. 73–104. Berkeley, CA, Apress.
15. Joachimczak, M., Liu, J., Ando, H. (2017). Real-time mixed-reality telepresence via 3D reconstruction with HoloLens and commodity depth sensors. *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pp. 514–515. Glasgow, UK.
16. O'Driscoll, K. (2016). The agile data modelling & design thinking approach to information system requirements analysis. *Journal of Decision Systems, 25(1),* 632–638. DOI 10.1080/12460125.2016.1189643.
17. Zhang, Y., Liu, X., Wang, Z., Chen, L. (2012). A service-oriented method for system-of-systems requirements analysis and architecture design. *Journal of Software, 7(2),* 358–365. DOI 10.3724/SP.J.1001.2008.00358.
18. Tew, Y., Wong, K. (2013). An overview of information hiding in H.264/AVC compressed video. *IEEE Transactions on Circuits and Systems for Video Technology, 24(2),* 305–319. DOI 10.1109/TCSVT.2013.2276710.
19. Watson, M. A., Buettner, P. (2000). Design and implementation of AAC decoders. *IEEE Transactions on Consumer Electronics, 46(3),* 819–824. DOI 10.1109/30.883454.
20. Jansen, B., Goodwin, T., Gupta, V., Kuipers, F., Zussman, G. (2018). Performance evaluation of WebRTC-based video conferencing. *ACM SIGMETRICS Performance Evaluation Review, 45(3),* 56–68. DOI 10.1145/3199524.3199534.
21. Unity Technologies, Inc. (2021). Unity Real-Time Development Platform|3D, 2D VR; AR Visualizations. https://unity.com/.
22. Endeley, R. E. (2018). End-to-end encryption in messaging services and national security—Case of WhatsApp messenger. *Journal of Information Security, 9(1),* 95–99. DOI 10.4236/jis.2018.91008.
23. Shafinah, K., Ikram, M. M. (2011). File security based on pretty good privacy (PGP) concept. *Computer and Information Science, 4(4)*. DOI 10.5539/cis.v4n4p10.
24. Shen, G., Gao, G. P., Li, S., Shum, H. Y., Zhang, Y. Q. (2005). Accelerate video decoding with generic GPU. *IEEE Transactions on Circuits and Systems for Video Technology, 15(5),* 685–693. DOI 10.1109/TCSVT.2005.846440.
25. Zhou, Y., Jiang, G., Lin, Y. (2016). A novel finger and hand pose estimation technique for real-time hand gesture recognition. *Pattern Recognition, 49,* 102–114. DOI 10.1016/j.patcog.2015.07.014.
26. Haralick, R. M., Shapiro, L. G. (1985). Image segmentation techniques. *Computer Vision, Graphics, and Image Processing, 29(1),* 100–132. DOI 10.1016/S0734-189X(85)90153-7.
27. Hafiz, A. M., Bhat, G. M. (2020). A survey on instance segmentation: State of the art. *International Journal of Multimedia Information Retrieval, 9(3),* 171–189. DOI 10.1007/s13735-020-00195-x.
28. Wang, X., Kong, T., Shen, C., Jiang, Y., Li, L. (2020). Solo: Segmenting objects by locations. *European Conference on Computer Vision*, pp. 649–665. Glasgow, UK.
29. Chen, K., Wang, J., Pang, J., Cao, Y., Xiong, Y. et al. (2019). MMDetection: Open mmlab detection toolbox and benchmark. arXiv preprint arXiv:1906.07155.
30. Lin, T. Y., Dollár, P., Girshick, R., He, K., Hariharan, B. et al. (2017). Feature pyramid networks for object detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125. Hawaii, USA.
31. Ballas, N., Yao, L., Pal, C., Courville, A. (2015). Delving deeper into convolutional networks for learning video representations. arXiv preprint arXiv:1511.06432.

32. Hochreiter, S., Schmidhuber, J. (1997). Long short-term memory. *Neural Computation, 9(8),* 1735–1780. DOI 10.1162/neco.1997.9.8.1735.
33. Xingjian, S. H. I., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K. et al. (2015). Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: *Advances in neural information processing systems,* pp. 28. La Jolla, California, NIPS.
34. Boda, J., Pandya, D. (2018). A survey on image matting techniques. *2018 International Conference on Communication and Signal Processing*, pp. 0765–0770. Chennai, India.
35. Ke, Z., Li, K., Zhou, Y., Wu, Q., Mao, X. et al. (2020). Is a green screen really necessary for real-time portrait matting? arXiv preprint arXiv:2011.11961.
36. Lin, S., Yang, L., Saleemi, I., Sengupta, S. (2022). Robust high-resolution video matting with temporal guidance. *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 238–247. Hawaii, USA.
37. Microsoft, Inc. (2021). Microsoft HoloLens|Mixed Reality Technology for Business: For precise, efficient hands-free work. https://www.microsoft.com/en-us/hololens.
38. Salvaris, M., Dean, D., Tok, W. H. (2018). *Deep learning with azure*. Berkeley, CA: Apress.
39. Le, H. Q., Kim, J. I. (2017). An augmented reality application with hand gestures for learning 3D geometry. *2017 IEEE International Conference on Big Data and Smart Computing*, pp. 34–41. Jeju, Korea.
40. Moro, C., Phelps, C., Redmond, P., Stromberga, Z. (2021). HoloLens and mobile augmented reality in medical and health science education: A randomised controlled trial. *British Journal of Educational Technology, 52(2),* 680–694. DOI 10.1111/bjet.13049.
41. Noor, A. K. (2016). The hololens revolution. *Mechanical Engineering, 138(10),* 30–35. DOI 10.1115/1.2016-Oct-1.
42. Liu, Y., Dong, H., Zhang, L., El Saddik, A. (2018). Technical evaluation of HoloLens for multimedia: A first look. *IEEE MultiMedia, 25(4),* 8–18. DOI 10.1109/MMUL.2018.2873473.
43. Evans, G., Miller, J., Pena, M. I., MacAllister, A., Winer, E. (2017). Evaluating the Microsoft HoloLens through an augmented reality assembly application. In: *Degraded environments: Sensing, processing, and display 2017*, 101970V. International Society for Optics and Photonics. Anaheim, California, USA.
44. Google, Inc. (2021). Google Glass. https://www.google.com/glass/start/.
45. HTC, Inc. (2021). VIVE-VR Headsets, Game, and Metaverse Life. https://www.vive.com/us/.
46. Pulli, K. (2017). 11-2: Invited paper: Meta 2: Immersive optical-see-through augmented reality. *SID Symposium Digest of Technical Papers*, vol. 48, no. 1, pp. 132–133. Los Angeles, CA, USA.
47. Ong, S., Siddaraju, V. K. (2021). Introduction to the mixed reality toolkit. *Beginning Windows Mixed Reality Programming*, pp. 85–110. Berkeley, CA, Apress.
48. Du, J., Shi, Y., Mei, C., Quarles, J., Yan, W. (2016). Communication by interaction: A multiplayer VR environment for building walkthroughs. *Construction Research Congress 2016*, pp. 2281–2290. San Juan, Puerto Rico.
49. Chen, H., Lee, A. S., Swift, M., Tang, J. C. (2015). 3D collaboration method over HoloLen and skype end points. *Proceedings of the 3rd International Workshop on Immersive Media Experiences*, pp. 27–30. Brisbane, Australia.