Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 6014795, 10 pages
https://doi.org/10.1155/2022/6014795

Research Article

Multiple HD Screen-Based Virtual Studio System with Learned Mask-Free Portrait Harmonization

Jimao Jiang[1], **Jiaxin Lin**[1], **Yuntao Su**[1], **Li Fang**[1,2] and **Long Ye**[1,2]

[1]*School of Data Science and Media Intelligence, Communication University of China, Beijing 100024, China*
[2]*Key Laboratory of Media Audio & Video (Communication University of China), Ministry of Education, Beijing 100024, China*

Correspondence should be addressed to Li Fang; lifang8902@cuc.edu.cn

Received 2 October 2021; Accepted 10 December 2021; Published 2 May 2022

Academic Editor: Ivan Lee

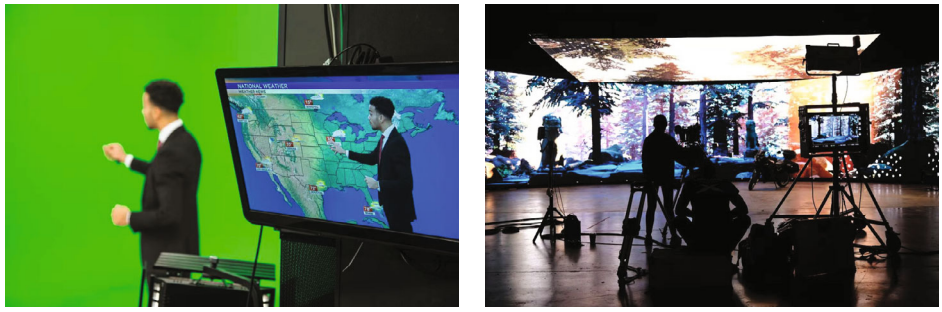
Virtual studio technology allows producers to combine live-action footage and computer-generated imagery and has promoted the development of film and television industry. However, the existing green screen-based virtual studio limits the visual effects just in postproduction. In this paper, we propose a new virtual studio system based on the recent emerging technology using multiple high-definition (HD) screens. Besides using traditional computer graphics like other similar systems, our system enables to capture panoramic video from the real world as the background and project it onto multiple HD screens, so as to recreate the scene in the studio. In addition, we propose a mask-free portrait harmonization network to make sure that the appearances of the foreground region and the background are consistent. Our portrait harmonization network does not need any auxiliary foreground mask and works in an end-to-end manner, which meets the postprocessing requirements for our virtual studio. Experimental result on real composite images shows that our network is superior to the state-of-the-art image harmonization method under the task of portrait harmonization. We also demonstrate our system with a CAVE system.

1. Introduction

Virtual studio technology based on chroma keying has set off a revolution in the film and television production industry in the past few decades. It allows the combination of people or other real objects and computer-generated environments and objects, which makes the whole studio no longer limited to traditional equipment such as site layout, lighting effects, and stage properties. Compared with the traditional studio, it requires less space but is more universal, providing producers with freer and broader creative space.

The main feature of a green screen-based studio is to film or photograph characters and objects against a single-color background (usually blue or green) and then synthesize a virtual background (shown in Figure 1(a)). However, this leads to the fact that characters cannot directly see the virtual elements in the studio, and therefore, the interaction with the virtual elements can only rely on imagination, which will inevitably cause problems such as the lack of the sense of presence of characters, resulting in unnatural program effect.

Recently, with the development of display technology, motion capture, and computer-generated imagery (CGI), multiple high-definition (HD) screens such as quality LED screens are being used as impressive replacements for traditional green screens, enabling producers to capture both live action and real-time rendered environments and objects in camera together, as shown in Figure 1(b). Furthermore, the real-time animation brought by the upgrade of the virtual engine is easy to use and scalable and gradually reduces the dependence on hardware. Nowadays, installation of HD screens in studios is becoming a trend for the filmmaking industry, and implementation of the technology will be more and more common. However, virtual production requires a 3D graphic artist and 3D computer graphics software to create the virtual background and any graphics that appear in front. For natural scenes with complex materials



(a) (b)

Figure 1: Different virtual studios. (a) Green screen-based virtual studio. The host cannot directly see the virtual elements. (b) Multiple HD screen-based virtual studio. The appearances of the foreground and the background are inconsistent.

and lighting, creating desired photorealistic models is costly. Applications that are sensitive to processing delays, such as live news broadcasting and remote face-to-face meeting, cannot use such method to rebuild the scene. Moreover, in order to create an illusion that characters and objects filmed are present in the intended background scene, the lighting in the two scenes must be a reasonable match. As can be seen in Figure 1(b), the appearances of the foreground are inconsistent with those of the background, which may cause unnatural imaging. This poses a challenge to the lighting of the studio.

In this paper, we propose a multiple HD screen-based virtual studio system that enables the use of shooting scenes from the real world as the background, as well as CGI. Specifically, our system takes panoramic videos as the background and utilizes multiple HD screens to recreate the scene in the studio. Characters in the studio seem to be on the spot, so as to get the best sense of presence. In addition, to simplify lighting requirements, we propose a deep portrait harmonization network that can make the appearances of the real part and the virtual part consistent. Unlike other image harmonization method, our network does not need to provide a foreground mask and works in an end-to-end manner, making it very suitable for our virtual studio. Experimental result shows that our network surpasses the state-of-the-art image harmonization method RainNet [1] without given any foreground mask. With the proposed virtual studio, the crew does not need to actually travel to particular locations for shooting. Lengthy and expensive location shoots can now be streamlined into one trip—capture the ideal light once then bring it back to the studio and recreate it as many times as needed in a controlled environment.

In summary, we developed a multiple HD screen-based virtual studio system. The main contributions of this paper are as follows:

(1) We developed a virtual studio system, which is complete and easy to use. Our system takes not only CGI but also panoramic videos as the background and thus provides a low-cost solution for high-fidelity scene recreation

(2) We proposed an end-to-end portrait harmonization method, which can directly process the video taken in the virtual studio without any additional foreground mask, so that the appearances of the characters in the virtual studio can be consistent with the virtual background

The rest of our paper is organized as follows. Section 2 comprehensively introduces the progress of virtual studio and some recent works of image harmonization. Section 3 presents our virtual studio system based on multiple HD screens. Section 4 presents the proposed deep portrait harmonization network for postprocessing. Section 5 is the experiments and detailed results. Section 6 concludes this paper.

2. Background and Related Work

2.1. Virtual Studio. Virtual production uses a suite of software tools to allow studios to combine live-action footage and computer graphics in real time. Digital environment can be created and rendered individually, while characters are physically working on set.

Nowadays, green screen is frequently used in most virtual studios. The BrainStorm company uses green screen combined with InfinitySet, featuring the patented TrackFree™ technology, which includes trackless and tracked camera or a combination of them [2]. CJP Broadcast also uses green screen to complete a 4K-UHD virtual studio system [3]. However, the utilization of green screen may lead to a sophisticated building process, time-consuming postprocessing, and poor sense of presence for characters.

Recently, there is an emerging trend on installation of HD screens instead of green screen. LED walls show characters what the set they are on looks like both through their eyes and in camera, so as to provide a better sense of presence. The Absen company uses advanced LED display combined with a virtual filming system, spatial positioning system, and real-time rendering system to achieve results of professional effect [4]. The Disney movie Mandalorian also chose LED display in filming [5].

2.2. Deep Learning Method for Image Harmonization. To make the synthesized output more realistic and harmonious is a major challenge in the virtual studio system, which has attracted the interest of many researchers. Traditional image

harmonization methods focus on matching the appearances of the foreground and the background using image statistics, such as color statistics [6], gradient-domain statistics [7], and semantic information [8].

With the success of deep learning in computer vision and computer graphics [9, 10], some deep learning-based image harmonization methods have been introduced. In [11], Tsai et al. proposed an end-to-end deep convolutional neural network, which can capture both the context and semantic information of the composite image during image harmonization. In [12], Cun and Pun proposed a novel attention module called Spatial-Separated Attention Module (S2AM) and combined it with the Unet [13] backbone network. In [14], Cong et al. proposed a deep image harmonization method DoveNet using a novel domain verification discriminator.

Among all these methods, RainNet [1], with the ability to make the composite image style-consistent and more realistic by learning style information from the background and adjust the style of the foreground, provides the state-of-the-art performance.

3. Proposed Virtual Studio System Based on Multiple HD Screens with Panoramic Video

To rebuild a scene, instead of 3D computer graphics made by 3D graphic artists, our virtual studio system uses panoramic videos or images shot from real scenes and projects them onto multiple screens. When shooting in the studio, both characters on set and virtual background on screens are captured by a TV camera, and the recorded video is then post processed by the portrait harmonization network resulting to the final output video. Our system is implemented using Unreal Engine 4 (UE4), and its framework is illustrated in Figure 2. In our implementation, we use an Insta360 Pro panoramic camera to capture panoramic videos. Besides, our system also includes necessary tools, such as camera tracking and motion capture, which enable characters to interact with the virtual environment. Please note that even though the major goal of our system is to allow the use of real shots as the background, traditional CGI works as well, and they can also be used together for more complex purposes.

3.1. Panoramic Video Segmentation and Projection of the Panorama onto Multiple HD Screens. The panoramic video is a type of video that breaks through the two-dimensional plane and can record everything happening in a 360-degree sphere with the panoramic camera in the center. As for the shooting of the panoramic video, we use an Insta360 Pro panoramic camera consisting of six fisheye lenses.

Since the panoramic video is spherical and cannot be played directly with planar screens, it needs to be projected first. Assuming that the panoramic video is on a sphere, the projection of the spherical panoramic video to a planar format depends on a one-to-one pixel correspondence relationship between the panorama and each screen. Specifically, take the cubic screen layout as an example, as demonstrated in Figure 3, we first find the center of the sphere of the panorama and the center of the cube, and let them coincide. Next, we, respectively, denote the 6 faces of the cube as $X+$, $X-$, $Y+$, $Y-$, $Z+$, and $Z-$, according to their intersecting coordinate axes. Then, take face $Y+$ as an example, for each point (ρ, θ, φ) on the sphere; we calculate the intersection of the line passing through it and the origin with the cube through equation (1) under the constraint of equation (2).

$$\begin{cases} x = \dfrac{l}{2} \bullet \cot\theta, \\ y = \dfrac{l}{2}, \\ z = \dfrac{l \bullet \cot\varphi}{2\sin\theta}, \end{cases} \tag{1}$$

$$\left\{ \rho, \theta, \varphi | \rho = d, \theta \in \left(\cot^{-1}\left(\frac{w}{l}\right), \tan^{-1}\left(\frac{w}{l}\right) + \frac{\pi}{2} \right), \varphi \in \left(\cot^{-1}\left(\frac{h}{l}\right), \tan^{-1}\left(\frac{h}{l}\right) + \frac{\pi}{2} \right) \right\}, \tag{2}$$

where l, w, and h are the length, width, and height of the cube, respectively, and d is the radius of the sphere. And the resulting (x, y, z) is the coordinate of the corresponding point on the cube of the point (ρ, θ, φ) on the sphere. The projection of the remaining faces is calculated similarly, and finally, we get the projection of the spherical panoramic video.

In our implementation, the above projection process is realized in UE4 through nDisplay. Specifically, we create a sphere model, and the panoramic video is applied to the inner surface of the sphere as materials. Then, the panoramic video is rendered and displayed on multiple screens. The detailed steps are described as follows.

Firstly, create a media player and select creating "media texture" asset which connects to the media player. Then, select media texture and create material. Next, set the material in a detail panel, and establish a blueprint to perform UV coordinate transformation (see Figure 4(a)). The following step is to drag a sphere into the current level and set its size and position. After dragging the material into the sphere material and creating a blueprint (see Figure 4(b)) in the level blueprint, run it and we can get the panoramic video display in UE4.

Note that the size and position of the sphere matter. Smaller sizes lead to fisheye distortion, while larger sizes lead to an unclear background. Therefore, in our system, we empirically set the size of the sphere to 1000 times the original size and set the center of the sphere to the origin.

The panoramic video is then displayed on multiple screens using nDisplay. Figure 5 shows how nDisplay works with multiple HD screens in the network. The host node PC is considered the conductor of nDisplay. It is used for centralized processing of the following: managing data in a synchronous manner and dispatching data to other cluster node PCs in the nDisplay cluster network, such as data inputs, camera tracking data, and custom cluster events, ensuring that the same data and the same inputs are received at the same time. The nDisplay Launcher and nDisplay Listener are used to control instances of UE4 across different computers on a network, each connected to one or many displays [15].

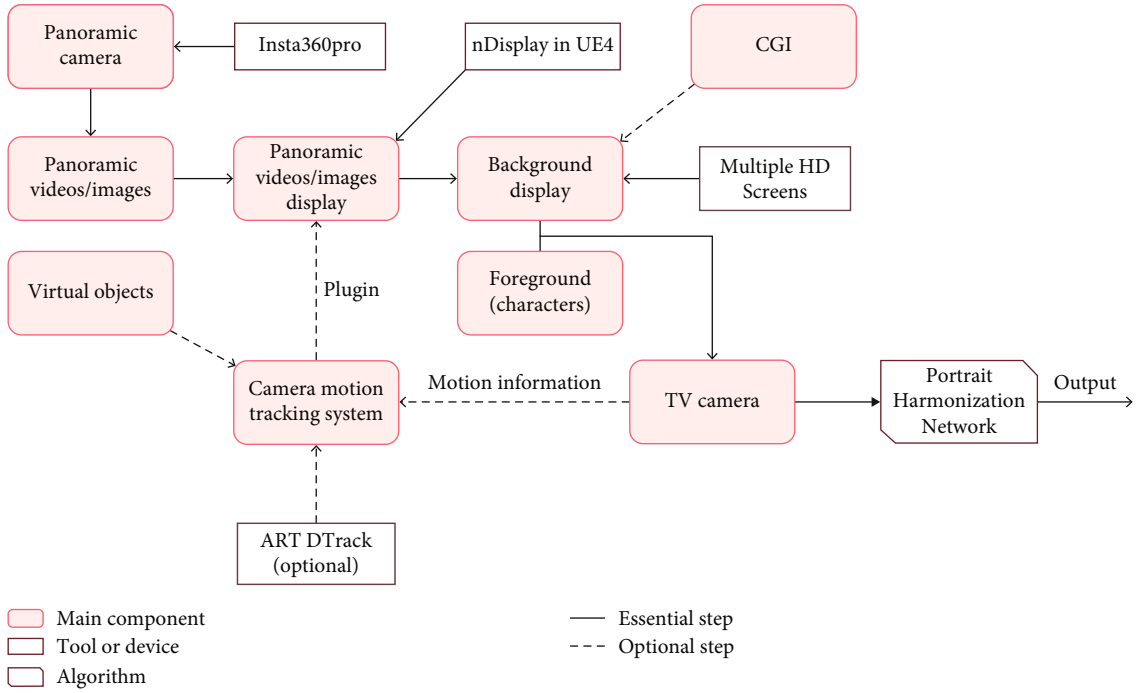


Figure 2: The framework of our virtual studio system.

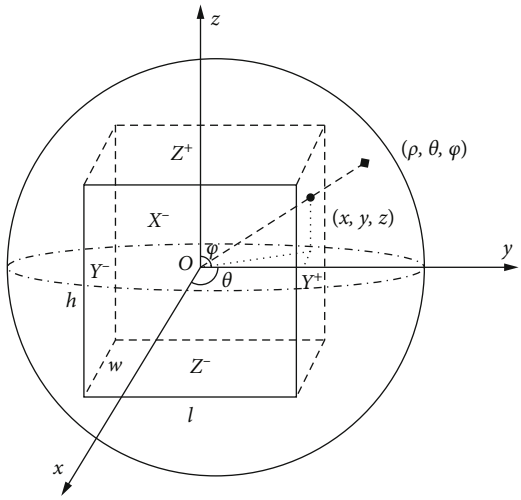


Figure 3: One-to-one pixel correspondence relationship on the panorama and multiple HD screens.

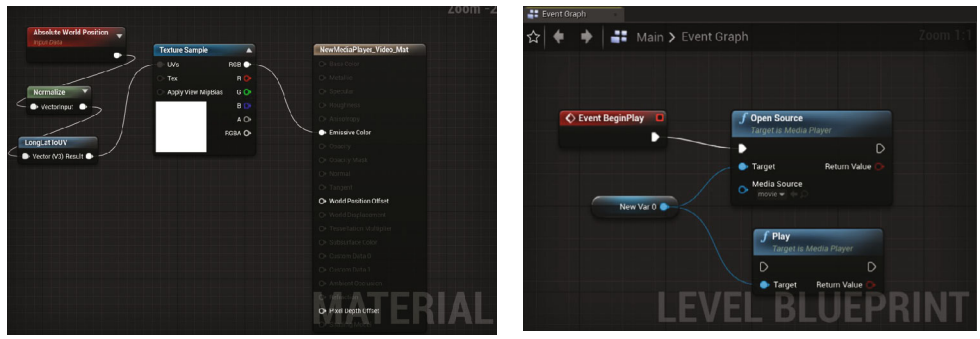
The nDisplay toolset consists of a plugin, a set of configuration files, and applications developed for UE4. The configuration files describe the topology of the display system and overall centralized location for project setting. It is worth noting that the configuration file needs to be modified to be in accordance with the size, resolution, node structure, virtual camera position, and other parameters of the multiple screens and related computers. In general, there is no need to modify the configuration file after confirming the correct operation unless the network topology changes. These are the exceptions: when the physical arrangement of the screens is changed or the computers which we are rendering to need to be changed [15].

With nDisplay Listener on every PC started and the projector turned on, add the appropriate configuration file to "config files" in nDisplay Launcher and run the corresponding EXE file; the content of UE4 can then be played on multiple HD screens. Therefore, when we are in the virtual studio, it is just like we are in the real world.

3.2. Tracking with the ART System. To track objects in the virtual studio system in real time, in our implementation, the ART tracking system, which is an infrared (IR) optical tracking system, is adopted. Tracking means the process of determining the position of a moving object in space. In order to track, these objects need to be equipped with individual marking points or rigid body. The spatial coordinates (X, Y, Z) of the rigid body are called 3DOF, and the measurement process of position and direction parameters is called 6DOF tracking [16].

Individual markers can only meet the 3DOF coordinate requirements, but if 6DOF coordinates are needed, the rigid body is required. The ART system mainly consists of four TRACKPACK/E cameras, four camera cables for power supply and sync, one external sync input cable, the atc-301604008 controllers of ART, several marking points, and DTrack2 software [16].

The principle of infrared optical tracking is shown as follows. The infrared transmitter cameras installed above HD screens send out synchronized IR flashes, and the marking points covered by the retroreflective material could then reflect the infrared light towards the lens. Intelligent tracking cameras create a greyscale image after detecting the reflected IR radiation. Therefore, the cameras can calculate 2D coordinates of the marking points based on pattern recognition. Then, the 2D data are sent to the controller via Ethernet. To get 6DOF data, DTrack2 calculates the position coordinates



(a) (b)

Figure 4: Blueprints in Unreal Engine 4: (a) material blueprint; (b) level blueprint.

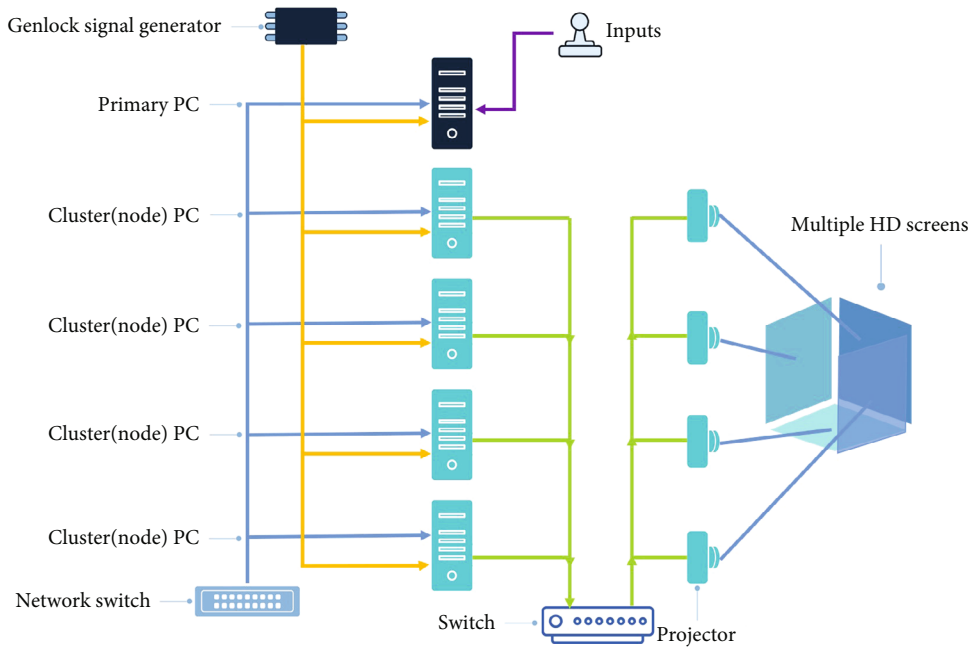


Figure 5: Network setup for multiple HD screens.

of the tracking target and determines the rigid body shape by the position distribution of the marking points. And the principle of optical tracking is shown in Figure 6 [16].

In optical tracking systems, it is necessary that the position of the object to be tracked is limited in the tracking range of the cameras, and the target should not be occluded by other objects because it will lead to problems such as inaccurate tracking.

Before tracking in the virtual studio, DTrack2 should be started. All necessary calculations (3DOF and 6DOF data) are done by the controller. The data and control commands are interchanged via a TCP/IP connection between the controller and the DTrack2 frontend software on the remote PC [16].

In advance, the orientation of each camera should be adjusted and a room calibration should be carried out. After starting the software, select the desired output data in the output settings panel. In UE4, just download the Dtrack-plugin previously and create a connection to Dtrack service source; then, real-time tracking can be achieved by adding components "Live Link Controller" to the target you want to track. As the marker moves, the tracked object on HD screen panels moves with it.

In the virtual studio, characters can better interact with the background through DTrack2. If we add a virtual object onto the panorama in nDisplay and add a live link to this object, the character can control the object using Flystick, which is similar to gamepad used in DTrack2. For example, if the character is performing in the virtual studio (such as football match commentary), he or she can use Flystick to pull up and zoom in/out on the scoreboard.

3.3. Capturing Video with Both the Foreground and Background on Multiple HD Screens. When the panoramic video is played on multiple HD screens and forms the background, the characters can perform in the studio. Now we focus on the shooting part. We should make sure that the

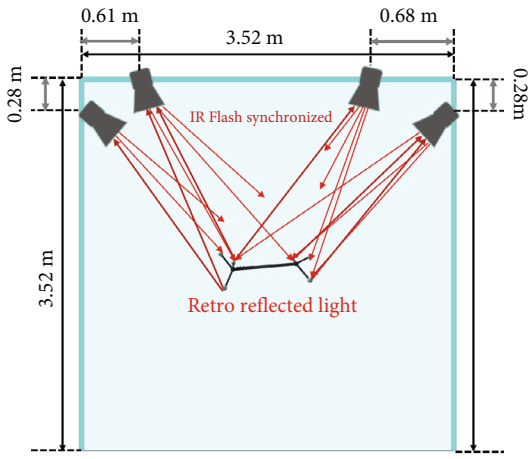


Figure 6: Principle of optical tracking.

camera center is located at the center of the sphere, so that there is no geometric distortion in the output video. To achieve this, we track the camera with the ART system and move it until the two centers coincide.

With the appropriate shooting method, the output of the TV camera combines both the foreground and background. However, due to various factors, the foreground and the background look inharmonious. We will discuss the solution in the next section.

4. Postprocessing Using Portrait Harmonization

4.1. Problem Formulation. In the virtual studio, the TV camera captures both characters on set and background displayed on the HD screens. For the sake of simplicity, we ignore the influence of the imaging process. Let us consider only one frame. Given the foreground image I_f and the background image I_b, this process can be simply regarded as the superposition of two images:

$$I = \alpha \circ I_f + (1 - \alpha) \circ I_b, \tag{3}$$

where α is the alpha values of the respective pixels in the foreground region, "$\circ$" denotes the Hadamard product, and I is the image captured by the TV camera.

However, due to the difference between the shooting conditions of the studio and the scene where the background is collected, such as lighting and exposure, the appearances of the foreground and background are inevitably incompatible. Intuitively, this problem can be solved by applying image harmonization on captured image I. As described in Section 2.2, many image harmonization methods based on deep learning are presented and can achieve impressive results. But to our knowledge, all these methods require a foreground mask as an auxiliary input, making them inappropriate for our virtual studio.

4.2. Architecture of the Proposed Method. Our goal is to make the appearances of the foreground region and the background region consistent, without giving a mask. Intuitively, we can generate a foreground mask and utilize any effective image harmonization network. In our case, the foreground is usually human. Therefore, we propose a novel deep neural network that consists of a portrait matting module and an image harmonization module. Figure 7 demonstrates the architecture of the proposed mask-free portrait harmonization network.

For the portrait matting module, we borrow the network from the state-of-the-art work MODNet [17]. This network consists of 3 interdependent branches, including low-resolution semantic estimation branch S to predict coarse semantic mask s_p, high-resolution detail prediction branch D to calculate the boundary detail matte d_p while considering the dependency between semantics and original images, and semantic-detail fusion branch F to combine semantics and details and to get α_p. Our loss function consists of 3 parts:

$$\mathscr{L}_m = \lambda_s \mathscr{L}_s + \lambda_d \mathscr{L}_d + \lambda_\alpha \mathscr{L}_\alpha, \tag{4}$$

where $\mathscr{L}_s$ is the $L2$ loss of s_p and thumbnail of the ground truth matte $G(\alpha_g)$, i.e., $\mathscr{L}_s = 1/2\|s_p - G(\alpha_g)\|_2$; $\mathscr{L}_d$ is the $L1$ loss of d_p and the ground truth mask, i.e., $\mathscr{L}_d = \|d_p - \alpha_g\|_1$; $\mathscr{L}_\alpha$ is the $L1$ loss of α_p and the ground truth mask plus the compositional loss $\mathscr{L}_c$ from DIM [18], i.e., $\mathscr{L}_\alpha = \|\alpha_p - \alpha_g\|_1 + \mathscr{L}_c$; $\lambda_s, \lambda_d, \lambda_\alpha$ are hyperparameters balancing the three losses. We set $\lambda_s = \lambda_\alpha = 1$ and $\lambda_d = 10$. It is worth noting that we do not use the trimap-dependent binary mask proposed in the original architecture of MODNet, because our training does not require the image trimap. Details of training will be mentioned in Section 4.3.

Finally, we get the mask α of the characters in the input image.

For the image harmonization module, the network from the state-of-the-art work RainNet [1] is used. This network takes a simple U-Net [13, 19] alike network without any feature normalization layers as basic architecture. The baseline also adds three attention blocks in the decoder part. Besides, RainNet has designed a delicate RAIN module that can be applied in any layers of the basic network.

As the foreground mask α is obtained, it is then concatenated with the input image along channel dimensions and sent into the RAIN module. Let $I_f^i \in \mathbb{R}^{H^i \times W^i \times C^i}$ be the activations, where H^i, W^i, C^i denote the height, width, and the number of channels of the input feature map, respectively. Because we want to match the foreground with the background, referring to the method of RainNet [1], the new activation value $I_f^{i\prime}$ at site (h, w, c) in the foreground region is computed by

$$I'_{f(h,w,c)} = \gamma_c^i \frac{I^i_{f(h,w,c)} - \mu_c^i}{\sigma_c^i} + \beta_c^i, \tag{5}$$

where μ_c^i and σ_c^i are the channel-wise mean and variance of the foreground feature and γ_c^i and β_c^i are the mean and standard deviation of the activations of the background in channel c of layer i. γ_c^i and β_c^i represent the statistical style only from

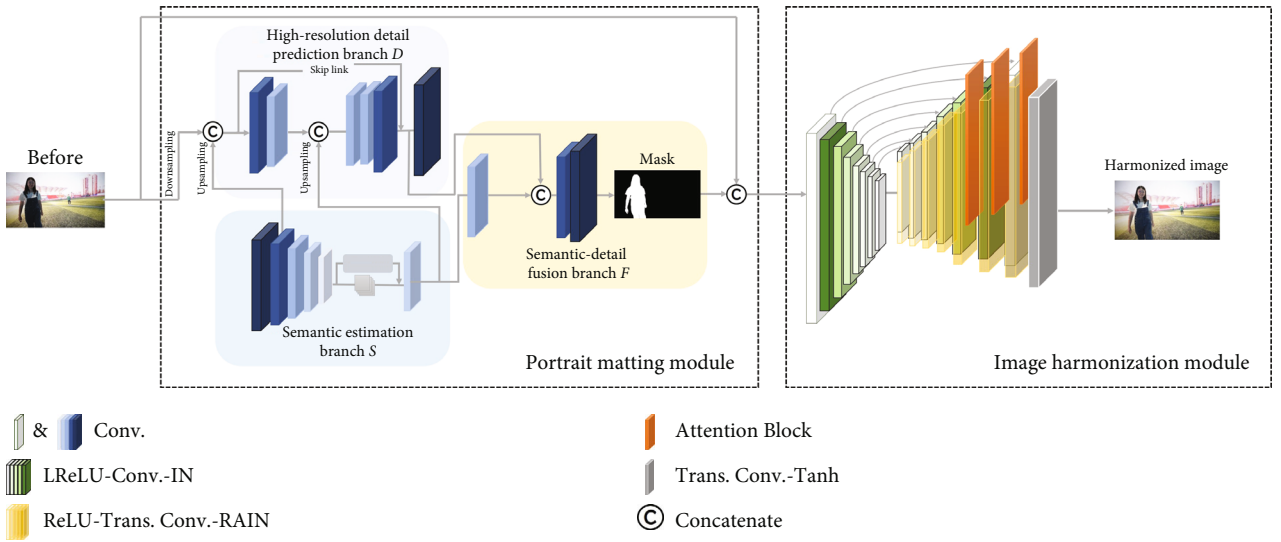


Figure 7: Architecture of the proposed portrait harmonization network. It consists of 2 modules: the portrait matting module and the image harmonization module. When a composite image is input, it can be directly harmonized into a consistent image. The bottom legend: Conv. = Convolution; Trans. = Transposed.

(a) (b)

Figure 8: Implementation of our virtual studio system with CAVE: (a) workstation and multiple screens; (b) shooting environment in CAVE.

(a) (b)

Figure 9: Portrait harmonization example: (a) original image; (b) harmonized image with our proposed method.

background features, and we normalize the foreground features to get the harmonized results accordingly.

As for the loss function of this module, we adopt the adversarial loss used in [1]. Specifically, we define the harmonization model as G and the harmonized image by $I' = G(I, \alpha)$. We optimize the results by calculating the $L1$ loss of I' and the ground truth I_g, i.e., $\mathscr{L}_{\text{rec}}(G, I_g, I, \alpha) = \|G(I, \alpha) - I_g\|_1$. And the additional adversarial loss is calculated following the training strategy in [14]. For the training of our network, we follow the setting of hyperparameters in [1]. As for the full objective,

$$\begin{aligned}\mathscr{L}\left(D, D_v, I_g, I', \alpha\right) &= \lambda_1 \mathscr{L}_{\text{adv}}\left(D, I_g, I'\right) + \lambda_2 \mathscr{L}_v\left(D_v, I_g, I', \alpha\right),\\ \mathscr{L}\left(G, I_g, I, \alpha\right) &= \lambda_1 \mathscr{L}_{\text{adv}}(G, I, \alpha) + \lambda_2 \mathscr{L}_v(G, I, \alpha) + \lambda_3 \mathscr{L}_{\text{rec}}\left(G, I_g, I, \alpha\right),\end{aligned} \tag{6}$$

Table 1: Quantitative comparisons (PSNR/SSIM) of different methods over the PPM-100 dataset. Four images we show in Figure 10 were selected as examples to show individual results. The average values of PSNR/SSIM over the whole PPM-100 dataset are shown in the table as well.

Example	MODNet+RainNet		RainNet with GT alpha matte		Ours	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
(a)	19.32	0.822	27.68	0.865	26.62	0.869
(b)	19.37	0.848	20.95	0.861	27.00	0.894
(c)	20.51	0.792	25.96	0.852	29.58	0.863
(d)	17.74	0.823	20.45	0.863	27.22	0.904
Avg. over the whole PPM-100 dataset	24.59	0.861	27.03	0.880	28.46	0.886

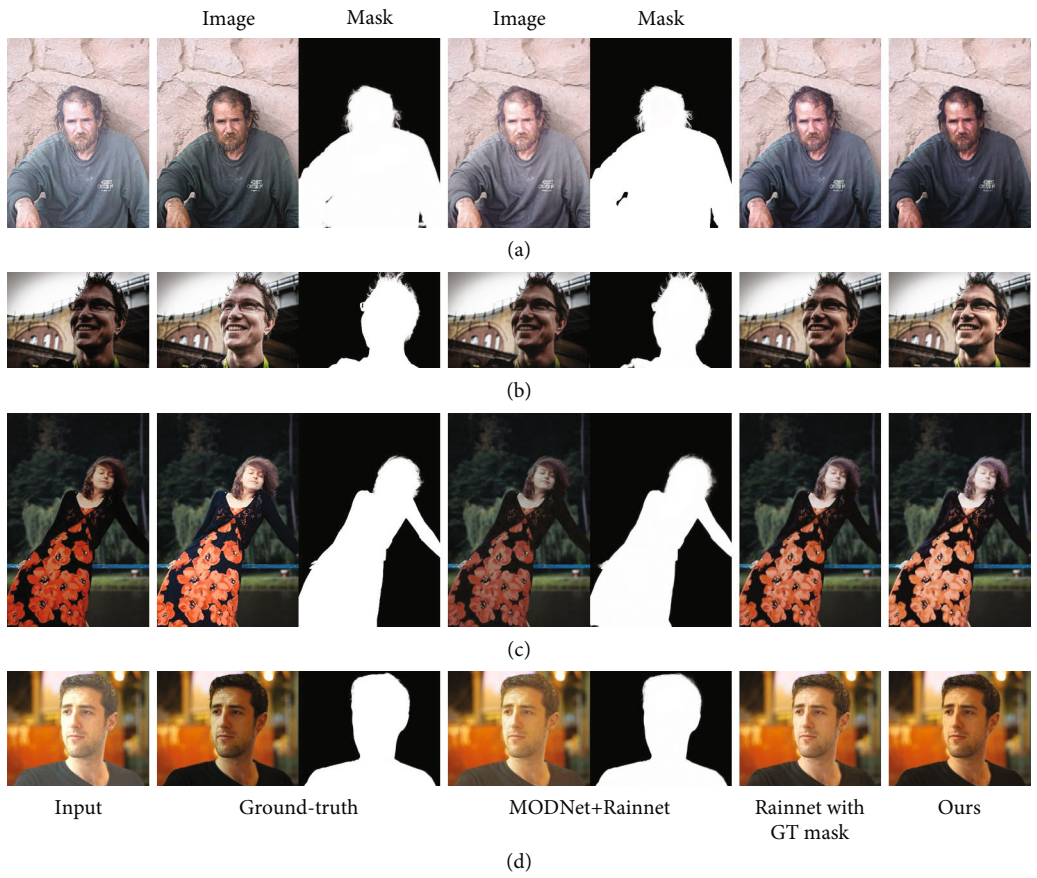


Figure 10: Visualization of comparisons on four randomly picked examples. From left to right: input images, ground truth images and their foreground masks, and harmonized results of MODNet+RainNet, RainNet with ground truth mask, and our methods.

where $\mathscr{L}_{\text{adv}}(D, I_g, I')$ and $\mathscr{L}_{\text{adv}}(G, I, \alpha)$ are adversarial losses and $\mathscr{L}_v(D_v, I_g, I', \alpha)$ and $\mathscr{L}_v(G, I, \alpha)$ are domain verification losses; we all follow the training strategy in [14] and set $\lambda_1 = \lambda_2 = 1$ and $\lambda_3 = 100$.

In our design, the foreground mask is generated and used implicitly. Therefore, better results can be obtained through global optimization, which will be verified in Section 5.

4.3. Implementation and Training Details. Since both the portrait matting module and the image harmonization module are differentiable, the proposed portrait harmonization network can be learned through end-to-end training with backpropagation.

Common datasets used for image harmonization (such as iHarmony4 [14]) include more than portrait images. And most of them do not contain enough good portrait masks. Therefore, we used a new dataset based on the Baidu people segmentation dataset [20] for end-to-end learning. This new dataset is composed of more than 5000 portrait images, corresponding masks, and composite images. This dataset guarantees sample diversity well, for example, (1)

different scenes, such as full-body images and half-body images; (2) diverse background, such as blurred background and dim background; and (3) various postures, such as holding objects and wearing hats. But the Baidu people segmentation dataset only includes natural images and the corresponding mask and lacks images with inharmonious foreground and background. Therefore, we preprocessed this dataset to adapt the portrait harmonization task. To get inharmonious images, we extracted the foreground from the original portrait image using the mask and randomly adjusted the brightness, color temperature, and other parameters of the foreground within a reasonable range and finally superposed the modified foreground over the original background.

For both modules, we first loaded the pretrained models provided by the authors of RainNet and MODNet and then trained it on our dataset with input images resized to 256×256. We used the Adam optimizer for the image harmonization module with a learning rate of 0.00001 and betas set to $(0.5,0.999)$ and the SGD optimizer for portrait matting module with a learning rate of 0.00001 and momentum set to 0.9. Our model was optimized for 75 epochs on an Nvidia GV100GL GPU, with the batch size set to 16.

5. Experimental Results

5.1. Demonstration of the Proposed System with CAVE. Our virtual studio system was implemented and demonstrated using the Cave Automatic Virtual Environment (CAVE) system [21]. Besides, the screens in CAVE can synchronously play video in real time, so characters can see the variation directly, which is beyond the reach of traditional green screen studio. This CAVE consists of four projection hard screens. Although the display effect is not as good as advanced LED walls, it is enough to show our virtual studio system. It is worth noting that our system is applicable to any multiple-screen display device, and we chose CAVE because we only have CAVE.

Our CAVE system consists of four screens—the left, right, front, and floor screens. The four projection surfaces form a cube, in which three adopt the way of the rear projection and the floor takes front projection. The screen of CAVE is 3.52 m long, 3.52 m wide, and 2.20 m high. Meanwhile, the resolution of computer screens is 1920×1200, which can properly suit CAVE.

The workstation contains five computers with 32.0 G RAM and Intel (R) Xeon (R) CPU at 2.50 GHz, one major node that controls the whole system, and four computers connected to the four screens of CAVE, respectively, as shown in Figure 8(a).

In order to display the panoramic video properly in CAVE, we created an nDisplay project in UE4, where we created a sphere and pasted the panoramic video onto the sphere as depicted in Section 3.1. After rendering and adjusting following the steps in Sections 3.2-3.3, the panoramic video was successfully played on CAVE, as shown in Figure 8(a).

When the panoramic video is playing, the characters can step into CAVE and perform. In the meantime, the TV camera captures video containing both the foreground and background. The camera we chose for recording is SONY ILCE-7RM3. Figure 8(b) shows the actual shooting environment in CAVE, and Figure 9(a) gives an example.

It can be seen from Figure 9(a) that the appearances of the character and the background are inharmonious. We can find that characters and the background fuse well, greatly reducing the feelings of separation compared with the green screen studio. But due to the limitation of projection brightness, the illumination of on-site characters does not match the background environment, as well as other features such as color temperature. Although this can be alleviated by appropriate lighting, it is costly. In our system, the inharmonious video frames were then postprocessed by the proposed portrait harmonization network. The foreground is obviously more compatible with the background, and the brightness and color temperature of the foreground and background are more consistent, making the image look more realistic, as shown in Figure 9(b). Our postprocessing with portrait harmonization can achieve a good result without additional hardware. Therefore, our system is very practical.

5.2. Portrait Harmonization Results. In this section, we further carried out experiments to verify our proposed portrait harmonization method and compared it with the state-of-the-art image harmonization method RainNet [1]. Since RainNet was not specifically trained for portrait harmonization, the pretrained model provided by the authors was carefully fine-tuned using our dataset and the suggested training configurations by the authors for fair comparisons.

For testing, we used the PPM-100 dataset, which is provided in MODNet paper [17] with many advantages as a validation set, such as fine annotation, natural background, rich diversity, and high resolution. So we argue that PPM-100 is a more comprehensive benchmark to verify the ability to deal with different scenes and different characters. The PPM-100 dataset was processed in the same way as the training set in advance.

Since RainNet requires a foreground mask, we carried out two different combinations:

MODNet+RainNet. We used MODNet with the pretrained model given by the authors to generate the foreground masks and provided them and corresponding input images to RainNet.

RainNet with GT mask. We directly used the ground truth foreground masks and corresponding input images as inputs of RainNet.

To quantitatively validate our postprocessing method, we followed the evaluation protocols from [1, 14]. We evaluated both methods on the PPM-100 dataset by measuring PSNR and SSIM for the harmonized images. Table 1 gives the results. Our average PSNR (28.46 dB) and SSIM (0.89) are much higher than other two methods, with PSNR increased by 15.74% and 5.29%, respectively, and SSIM increased by 2.90% and 0.68%, respectively.

We also visually compared the harmonized results of different algorithms on four randomly picked images, as shown in Figure 10. It can be seen from Figure 10 that our method performs well without foreground masks, proving the superiority of our method.

6. Conclusion

In this paper, we have presented a new multiple HD screen-based virtual studio system. It can take panoramic videos as the background and employ multiple HD screens to reconstruct an immersive environment. The system also includes methods such as panorama capturing, motion tracking, and shooting, providing a complete set of solutions for virtual production. We have further presented an end-to-end portrait harmonization network that can ensure that the appearances of the foreground region and the background are consistent, making the final video realistic. Experimental result on real composite images shows that the proposed network outperforms the previous state-of-the-art method.

Our current solution exploits panoramic video as the background, which lacks motion parallax and correct-in-all-directions disparity cues. Therefore, when using a panoramic video background, the camera position must be fixed. There is also an emerging trend on recording scenes with the dynamic light field or radiance field. In the future, we will try to introduce these 6DOF formats into our system to better promote the promotion of virtual production technology in the film and television production industry.

Data Availability

The portrait harmonization data supporting these findings of this study are from previously reported studies and datasets, which have been cited. The processed data are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Key R&D Program of China (Grant No. SQ2020YFF0426386), the National Natural Science Foundation of China (Grant Nos. 62001432 and 61971383), the Major Science and Technology Project of Beijing (Grant No. z201100001820024), and the Fundamental Research Funds for the Central Universities (Grant Nos. CUC19ZD006 and CUC21GZ007).

References

[1] J. Ling, H. Xue, L. Song, R. Xie, and X. Gu, "Region-aware adaptive instance normalization for image harmonization," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9357–9366, Nashville, TN, USA, 2021.

[2] "Virtual studios|Brainstorm Index," 2021, https://www.brainstorm3d.com/solutions/virtual-studios.

[3] "CJP Broadcast Index," 2021, https://www.avinteractive.com/markets/media/sunderland-university-completes-led-ready-4k-uhd-virtual-studio-26-08-2021/.

[4] "Absen Virtual Studio LED Solutions Index," 2021, https://www.absen.com/virtual-studio/.

[5] "Insider," 2020, https://www.insider.com/green-screen-virtual-sets-mandalorian-2020-4.

[6] S. Xue, A. Agarwala, J. Dorsey, and H. Rushmeier, "Understanding and improving the realism of image composites," *ACM Transactions on Graphics*, vol. 31, no. 4, pp. 1–10, 2012.

[7] M. W. Tao, M. K. Johnson, and S. Paris, "Error-tolerant image compositing," *International Journal of Computer Vision*, vol. 103, no. 2, pp. 178–189, 2013.

[8] Y.-H. Tsai, X. Shen, Z. Lin, K. Sunkavalli, and M.-H. Yang, "Sky is not the limit," *ACM Transactions on Graphics*, vol. 35, no. 4, pp. 1–11, 2016.

[9] S. Yang, J. Wang, S. Arif, M. Jia, and S. Zhong, "SAL-net: self-supervised attribute learning for object recognition and segmentation," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 2891303, 13 pages, 2021.

[10] M. Yan, Z. Li, X. Yu, and C. Jin, "An end-to-end deep learning network for 3D object detection from RGB-D data based on hough voting," *IEEE Access*, vol. 8, pp. 138810–138822, 2020.

[11] Y.-H. Tsai, X. Shen, Z. Li, K. Sunkavalli, X. Lu, and M.-H. Yang, "Deep image harmonization," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2799–2807, Honolulu, HI, USA, 2017.

[12] X. Cun and C.-M. Pun, "Improving the harmony of the composite image by spatial-separated attention module," *IEEE Transactions on Image Processing*, vol. 29, pp. 4759–4771, 2020.

[13] O. Ronneberger, P. Fischer, and T. Brox, "Unet: convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015*, N. Navab, J. Hornegger, W. Wells, and A. Frangi, Eds., vol. 9351 of Lecture Notes in Computer Science, pp. 234–241, Springer, Cham, 2015.

[14] W. Cong, J. Zhang, L. Niu, Z. Ling, W. Li, and L. Zhang, "DoveNet: deep image harmonization via domain verification," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8391–8400, Seattle, WA, USA, 2020.

[15] "Unreal Engine," 2021, https://www.unrealengine.com/.

[16] "Advanced Realtime Tracking," 2021, https://ar-tracking.com/en/product-program/dtrack.

[17] Z. Ke, K. Li, Y. Zhou et al., "Is a green screen really necessary for real-time portrait matting?," 2020, https://arxiv.org/abs/2011.11961.

[18] N. Xu, B. Price, S. Cohen, and T. Huang, "Deep image matting," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 311–320, Honolulu, HI, USA, 2017.

[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, Honolulu, HI, USA, 2017.

[20] "Baidu people segmentation dataset," 2021, http://www.cbsr.ia.ac.cn/users/ynyu/dataset/.

[21] H. Creagh, "Cave automatic virtual environment," in *Proceedings: Electrical Insulation Conference and Electrical Manufacturing and Coil Winding Technology Conference (Cat. No.03CH37480)*, pp. 499–504, Indianapolis, IN, USA, 2003.