Coarse- and Fine-Grained Fusion Hierarchical Network for Hole Filling in View Synthesis

Guangcheng Wang, Kui Jiang, *Member, IEEE*, Ke Gu, *Senior Member, IEEE*, Hongyan Liu, Hantao Liu, *Senior Member, IEEE*, and Wenjun Zhang, *Fellow, IEEE*

***Abstract*— Depth image-based rendering (DIBR) techniques play an essential role in free-viewpoint videos (FVVs), which generate the virtual views from a reference 2D texture video and its associated depth information. However, the background regions occluded by the foreground in the reference view will be exposed in the synthesized view, resulting in obvious irregular holes in the synthesized view. To this end, this paper proposes a novel coarse and fine-grained fusion hierarchical network (CFFHNet) for hole filling, which fills the irregular holes produced by view synthesis using the spatial contextual correlations between the visible and hole regions. CFFHNet adopts recurrent calculation to learn the spatial contextual correlation, while the hierarchical structure and attention mechanism are introduced to guide the fine-grained fusion of cross-scale contextual features. To promote texture generation while maintaining fidelity, we equip CFFHNet with a two-stage framework involving an inference sub-network to generate the coarse synthetic result and a refinement sub-network for refinement. Meanwhile, to make the learned hole-filling model better adaptable and robust to the "foreground penetration" distortion, we trained CFFHNet by generating a batch of training samples by adding irregular holes to the foreground and background connection regions of high-quality images. Extensive experiments show the superiority of our CFFHNet over the current state-of-the-art DIBR methods. The source code will be available at https://github.com/wgc-vsfm/view-synthesis-CFFHNet.**

***Index Terms*— Depth image-based rendering, hole filling, coarse and fine-grained fusion, hierarchical network, two-stage framework.**

Manuscript received 12 July 2021; revised 12 May 2022, 3 September 2022, 6 February 2023, and 20 September 2023; accepted 27 November 2023. Date of publication 15 December 2023; date of current version 20 December 2023. This work was supported in part by the Beijing Natural Science Foundation under Grant JQ21014; in part by the National Science Foundation of China under Grant 62322302, Grant 62273011, Grant 62076013, Grant 42001239, and Grant 62021003; in part by the Project under Grant 2023ZY01015; and in part by the Key Laboratory of Intelligent Control and Optimization for Industrial Equipment of Ministry of Education, Dalian University of Technology, under Grant LICO2022TB03. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Adrian Munteanu. *(Guangcheng Wang and Kui Jiang are co-first authors.) (Corresponding authors: Ke Gu; Hongyan Liu.)*

Guangcheng Wang is with the School of Transportation and Civil Engineering, Nantong University, Nantong 226019, China (e-mail: wangguangcheng0428@163.com).

Kui Jiang is with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin 150001, China (e-mail: jiangkui@hit.edu.cn).

Ke Gu and Hongyan Liu are with the Faculty of Information Technology, the Engineering Research Center of Intelligent Perception and Autonomous Control of Ministry of Education, the Beijing Laboratory of Smart Environmental Protection, the Beijing Key Laboratory of Computational Intelligence and Intelligent System, and the Beijing Artificial Intelligence Institute, Beijing University of Technology, Beijing 100124, China, and also with the Key Laboratory of Intelligent Control and Optimization for Industrial Equipment of Ministry of Education, Dalian University of Technology, Dalian 116024, China (e-mail: guke.doctor@gmail.com; liuhy9221@gmail.com).

Hantao Liu is with the School of Computer Science and Informatics, Cardiff University, CF24 3AA Cardiff, U.K. (e-mail: liuh35@cardiff.ac.uk).

Wenjun Zhang is with the Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: zhangwenjun@sjtu.edu.cn).

Digital Object Identifier 10.1109/TIP.2023.3341303

I. Introduction

As THE pursuit of visual experience is constantly upgraded, traditional 2D display technologies are challenging to meet people's demands for work, life, and entertainment, so 3D-related techniques, including virtual reality, augmented reality, and mixed reality, have been developed rapidly. Free-viewpoint TV (FTV) can provide viewers with any viewing angle and position, due to the excellent human-computer interaction experience and viewing immersion, which is considered to be the main development direction of digital TV in the future. Free-Viewpoint Video (FVV) has attracted increasing attention and interest owing to its wide application scenarios, such as video conference, remote education, immersive entertainment, medical applications, military area, and more [1], [2]. The MPEG-I (I: Immersive) standard, formulated by the MPEG committee for immersive media, focuses on the encoding representation of multi-view + depth videos [3].

Fig. 1 shows the free-viewpoint butterfly deployment of "Dancing Miracle" in the Huawei exhibition at the China International Audio-visual Conference. To realize the function of freely switching viewpoints, it is necessary to set up cameras in different directions to shoot the same scene. When a free-viewpoint video requires N viewpoints for users, this video will generate N times the data in a regular single-viewpoint video, putting tremendous pressure on data collection, storage, and transmission. The virtual viewpoint synthesis technology uses the scene captured at the reference viewpoint to synthesize the scene obtained when the virtual viewpoint faces the same scene, so as to greatly save the cost of streaming media data collection, storage, and transmission bandwidth. In addition, 6 degrees of freedom (6-DoF) navigation needs to render the required viewpoints in real-time according to the user's arbitrary navigation paths, where the view synthesis provides the alternative solution for this task.

The view synthesis technology can be mainly divided into two categories: 3D model-based rendering (MBR) and depth-image-based rendering (DIBR) [4], [5], [6]. Compared

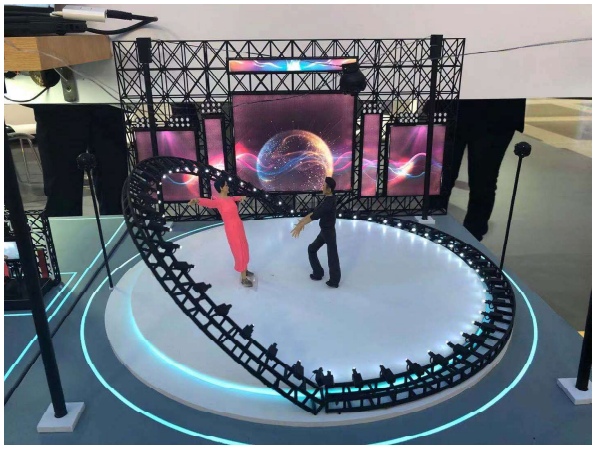


Fig. 1. The free-viewpoint butterfly deployment of “Dancing Miracle”. This figure is from http://ciac.org.cn/detail?id=6516.

Fig. 2. Examples of hole filling. (a) is an unfilled image with holes. (b) is produced by the Criminisi et al.’s method [14]. (c) is produced by the Daribo et al.’s method [15]. (d) is the ground-truth image corresponding to (a). The distortions in the green box are the “foreground penetration” artifacts.

with MBR techniques’ complex 3D reconstruction process, DIBR technologies only need a reference 2D video and its corresponding depth maps to generate the free-view video. Therefore, DIBR techniques are more easily deployed in real-time applications, such as sports live, video conference, remote diagnosis, remote education, and more. Unfortunately, the captured depth map is not accurate enough, or the background area may be occluded by some foreground objects in the reference viewpoint and become visible at the virtual viewpoint. Consequently, the synthesized virtual viewpoint usually contains irregular holes. How to fill holes to guarantee the affinity and fidelity between the hole and known contextual regions of the synthesized view remains a non-trivial problem in DIBR.

In recent decades, massive DIBR algorithms have been proposed, which can be divided into two categories. The first is to reduce hole regions by preprocessing the depth map [7], [8], [9], [10]. The Low-pass filters, such as symmetric Gaussian low-pass filter [7], [8], asymmetric filter [9], or adaptive edge-oriented smoothing filter [10] are employed to smooth the depth map. The low-pass filtering-based DIBR algorithms can effectively reduce the hole regions in the synthesized view. Those methods are dedicated to solving scenarios where the reference and virtual viewpoints are relatively close. When the reference and virtual viewpoints are far away, they are far from producing satisfactory results.

To solve the shortcomings of the aforementioned methods, the second type of algorithm uses the patch matching strategy to select the best-matching patch from hole-free regions to fill hole regions without preprocessing the depth map. The multi-reference viewpoint-based DIBR algorithms [6], [11], [12], [13] can greatly reduce the hole areas in the synthesized view but come at the cost of huge transmission bandwidth. By contrast, the single-reference viewpoint-based DIBR methods are more practical. For example, Criminisi et al. [14] employed an exemplar-based sampling method combining the advantages of inpainting and texture synthesis. By calculating the priority of the hole boundary pixels, it can search for the best-matching patch from non-hole regions and copy the selected patch to the highest priority area. However, the foreground textures are usually used to fill the hole regions. To alleviate this defect, Daribo et al. [15] further introduced the depth map of the virtual viewpoint on the basis of Criminisi et al.’s method [14] to compute the priority of the hole boundary pixels and patch distance. However, the depth map of the virtual viewpoint is not always available in the actual scenarios. To overcome this problem, Ahn and Kim [16] and Buyssens et al. [17] proposed to predict the depth map of the virtual view while filling the hole regions. Ceulemans et al. presented a Markov random field-based inpainting method for multiview video [18]. The method steers the Markov random field optimization towards completion from background to foreground and exploits the available depth information to avoid bleeding artifacts. These aforementioned methods all perform patch matching in a single frame, which are suitable for small hole filling. When the distance between the reference viewpoint and the virtual viewpoint is far away, the hole regions in the synthesized view are too large, and the performance of these algorithms is relatively limited. To deal with this issue, Luo and Zhu [19] and Luo et al. [20], [21] proposed to exploit the background information from the whole video sequence to fill the hole regions of the current frame. Specifically, they first split the foreground and background of the whole synthesized video and then use the patch matching-based hole-filling methods to search for the best-matching block in the known background regions of all frames to fill the hole regions. These works [19], [20], [21] can cover large holes more effectively but come at the cost of great computational and memory consumption, making it infeasible for real-time application scenarios. Fig. 2 shows some hole-filling examples of partial patch-matching-based hole-filling algorithms. Obviously, these algorithms commonly tend to fill the foreground texture into the hole areas, thus creating the “foreground penetration” artifact [21]. Though Daribo et al.’s method [15], Ahn et al.’s method [16], Zhu et al.’s method [19], and Luo et al.’s method [21] further introduce depth information or separate the foreground and background, these algorithms still cause “foreground penetration” because the obtained depth information is not accurate or the precision of segmentation is not enough.

To deal with the "foreground penetration" distortion, we propose a coarse and fine-grained fusion hierarchical network (CFFHNet) for hole filling in view synthesis. To promote the texture generation while keeping the fidelity, we equip CFFHNet with a two-stage framework involving a generative adversarial network (GAN)-based inference sub-network (ISN) to generate the coarse synthetic result with pleasant visual contents and a convolutional neural network (CNN)-based refinement sub-network (RSN) to promote the fidelity and consistency. In particular, we introduce the residual recurrent memory to exploit the spatial contextual correlations from the visible regions to infer the missing details of the hole regions. Meanwhile, we specially construct the backbone of CFFHNet with a hierarchical framework for the multi-scale cooperative representation. Overall, the design principle of CFFHNet is to capture and fuse multi-scale texture features in a coarse-to-fine manner while guaranteeing the affinity and fidelity between the hole and known contextual regions, which is crucial for dealing with the "foreground penetration" distortion. By doing so, our synthesized views are more faithful to the ground-truth and make the visual experience more natural.

We highlight our major contributions as follows:

- We propose to exploit the spatial contextual correlations from the visible regions to infer the missing texture details of the hole regions, and construct a novel CFFH-Net for hole filling in view synthesis. ISN introduces an adversarial training strategy to improve texture generation. RSN is responsible for inferring more realistic texture details.
- CFFHNet extracts the pixel-level spatial contextual correlations to restore the synthesized view, which can effectively reduce the "foreground penetration" distortion introduced by the patch-matching-based image inpainting methods.
- Our CFFHNet exhibits excellent performance in filling the hole regions existed in the synthesized view, which outperforms the popular and state-of-the-art (SOTA) DIBR algorithms.

The rest of this paper is organized as follows. Section II introduces some related works and the research motivations of this work. Section III elaborates on the design of our network. In Section IV, we compare our method with dozens of the popular and SOTA DIBR algorithms. In section V, we conduct various ablation studies to demonstrate the effectiveness of our network. Section VI concludes this work.

II. Related Works and Motivations

A. Deep Learning-Based Image Inpainting

The CNN and GAN have facilitated the rapid development of image inpainting tasks [22], [23], [24], [25], [26], [27], which usually repair damaged images by learning the intrinsic statistics knowledge from massive data. For instance, Li et al. designed a recurrent feature inference network consisting of a recurrent feature inference module and a knowledge-consistent attention module [22]. Mimicking the process of the human brain in dealing with difficult problems by first solving the easier parts and then using the preliminary results to solve the difficult parts, the network iteratively infers the hole boundaries of the convolutional feature maps and then uses them as clues for further inference. In literature [23], Xu et al. adopted the edge structure information to guide the image inpainting task. Lahiri et al. designed a prior-oriented GAN for semantic inpainting [24], which maps the implicit noise prior distribution to the manifold of natural images when training the generative model, thereby using the noise prior to enhance the structural prior to improve the inpainting reconstruction result. To precisely employ the valid information in an image to repair the damaged regions, Wang et al. proposed a dynamic selection network to distinguish the damaged regions from the undamaged valid regions [25]. Yu et al. presented a novel free-form image inpainting system based on an end-to-end generative network with gated convolution, trained with the pixel-wise ℓ_1 loss and a patch-based GAN loss [26]. Zeng et al. proposed a learnable auxiliary contextual reconstruction loss combined with the traditional inpainting loss (i.e., ℓ_1 loss and the the adversarial loss) to encourage the generator network to select appropriate known regions as references to fill the missing regions [27]. The deep learning-based image inpainting algorithms can fill holes, but there still exits obvious limitations in the following two main aspects. First, most existing image inpainting algorithms are designed for some regular holes. Second, existing deep learning-based image inpainting methods are not designed with the characteristics of DIBR techniques, i.e., they do not focus on how to avoid misfilling the foreground textures into the background regions.

B. Motivations

The general deep learning-based image inpainting algorithms tend to fill the foreground textures into the background holes when filling the holes generated by view synthesis. This is because the large-scale holes in synthesized images are generated by foreground occlusion. This also indicates that it is difficult to avoid the "foreground penetration" artifact only through network architecture and loss function. Moreover, some works have been devoted to reducing the probability of filling the foreground texture into the background area by segmenting the foreground and background of synthesized images and then filling the hole area with only the background texture information [19], [21]. However, these methods are still susceptible to the inaccurate foreground and background segmentation, filling foreground textures into background regions and resulting in "foreground penetration" artifacts. Inspired by the shortcomings of the above two kinds of work, we consider adopting the specific sample (preferably directly generated by warping) and model co-driving strategy to design a hole-filling algorithm more suitable for view synthesis. Based on the targeted design model (i.e., network architecture and loss function), the model is further endowed with the ability to suppress the "foreground penetration" artifact through specific training samples. Next, we will introduce the design motivation for the particular training data, network architecture, and loss functions.

1) Training Data: However, in the actual environment, there are no massive multiview video-plus-depth (MVD)

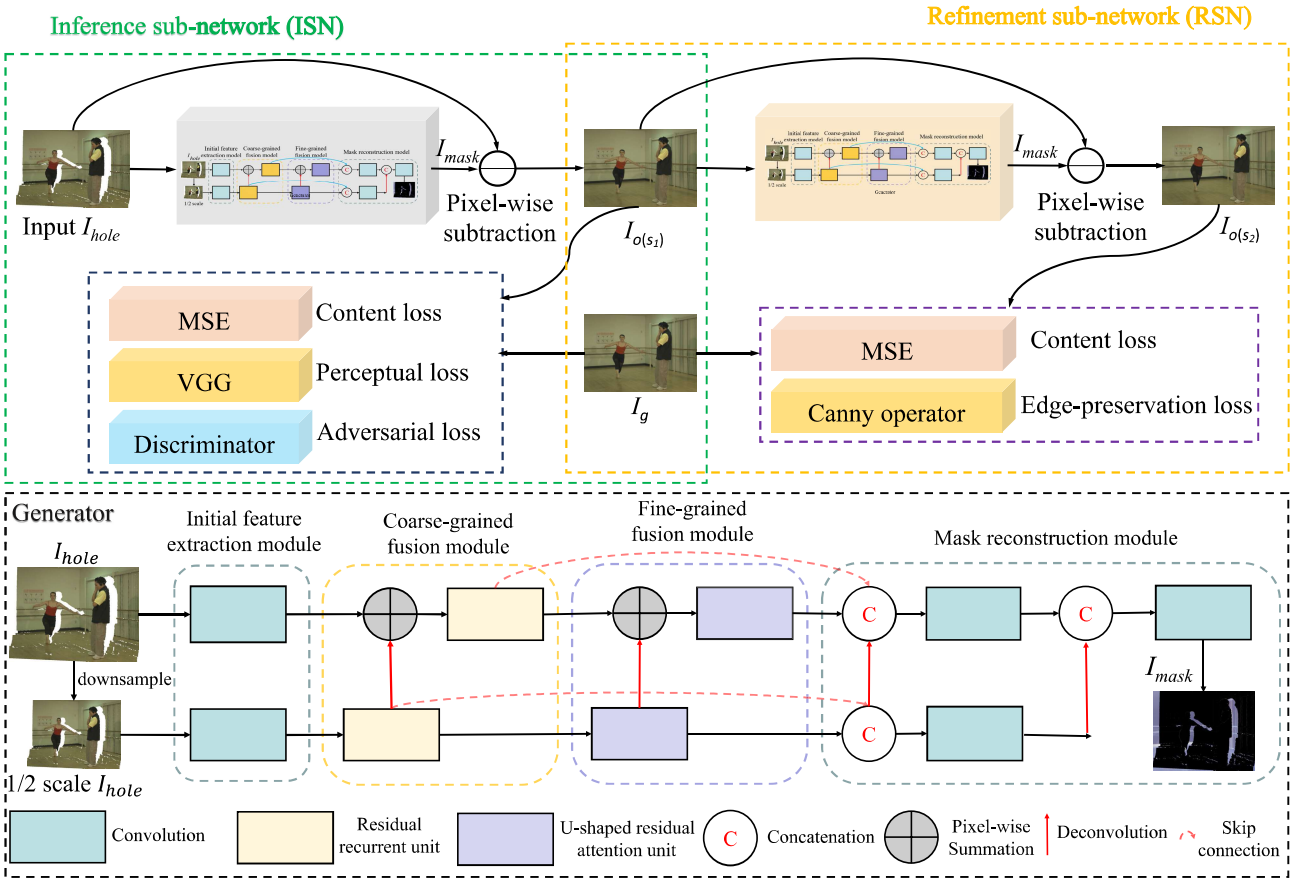


Fig. 3. Framework of the proposed coarse- and fine-grained fusion hierarchical network (CFFHNet) for hole filling in view synthesis.

sequences with different scenes to generate many training samples that can be used for training hole-filling networks. Therefore, we generate a batch of training samples by adding irregular holes in the foreground-and-background junction regions so that the learned CFFHNet can protect semantic information, that is, it can effectively suppress the "foreground penetration" artifact, which is the next best thing. The specific generation method of the training samples is detailed in Section IV-A.

2) Network Design: This paper proposes a coarse and fine-grained fusion hierarchical network to solve the hole-filling problem in view synthesis. We adopt the residual recurrent unit in the coarse-grained fusion module to better extract the global contextual texture relationship to repair hole regions. And the channel attention mechanism is introduced in the fine-grained module to express the contextual texture relationship learned in the coarse-grained fusion module more finely. In addition, we introduce a hierarchical architecture to fuse information of different spatial resolutions, which can also effectively protect the semantic information of images, thereby effectively avoiding filling foreground textures into background areas.

3) Learning Strategy: We define the hole filling problem as "$I_{hole} = I_g + I_{mask}$", i.e., I_{mask} is the residue map of the hole image I_{hole} and its ground-truth image I_g. This paper regards the cause of the hole regions in a synthesized image as adding a degradation factor I_{mask} to its associated ground-truth image. The purpose of the proposed CFFHNet is to learn this residue map. Then, computing "$I_{hole} - I_{mask}$" can obtain the restored synthesized view. Note that we define the hole filling problem as "$I_{hole} = I_g + I_{mask}$", which is different from this type of deep learning-inpainting algorithms [22], [26] that these methods learn the projection directly from the hole image to its ground-truth image. This work proposes a two-stage learning strategy for hole filling, including the GAN-based inference sub-network to promote the texture generation and the CNN-based refinement sub-network to guarantee the fidelity and affinity. The GAN-based methods have been widely used in the image enhancement and image inpainting tasks in recent years, where the powerful texture generation capability of GAN is crucial for filling the large holes. Since some large irregular holes will also be introduced due to disocclusion in view synthesis, this work introduce GAN to solve the hole filling problem in view synthesis. The original intention of GAN is to make the generated images more realistic. However, in the view synthesis applications, the fidelity of the synthesized images is crucial. Thus, we introduce the content loss (i.e., the MSE loss) on the basis of adversarial loss to guarantee the consistency of the inferred contents to the ground truth. Moreover, to eliminate the high-frequency noise brought by adversarial training, a CNN-based refinement sub-network is further introduced to refine

the textures to promote the affinity between the predicted hole regions and the known contents. In particular for the optimization of RSN, the content loss and edge-preservation loss are used for the joint optimization.

III. Methodology

A. Coarse and Fine-Grained Hierarchical Network

This part first details the design of the proposed coarse and fine-grained fusion hierarchical network. As shown in Fig. 3, our proposed two-stage CFFHNet contains two sub-networks: ISN and RSN. For convenience, ISN and RSN share the same generator framework, and it is designed with a hierarchical framework to promote the feature fusion and representation. We take the first stage as an example to detail its components. The generator consists of the initial feature extraction module, the coarse-grained fusion module (CFM), the fine-grained fusion module (FFM), and the mask reconstruction module. As illustrated in Fig. 3, for a given hole image I_{hole}, we first use a Gaussian kernel to down-sample the original image to generate the multi-scale hole images, such as 1/2 scale. Then, the initial feature extraction module takes them as inputs to extract the corresponding initial features using multiple parallel initial layers. After that, we integrate the residual recurrent unit (RRU) into the hierarchical framework to construct CFM to exploit the global contextual correlations. For a better fusion of multi-scale features, unlike CFM, FFM replaces the residual recurrent unit in CFM with the U-shaped residual unit, where the pyramid attention mechanism can guide the scale-specific knowledge aggregation. Finally, the mask reconstruction module obtains the I_{mask} by integrating the spatial contextual correlations generated from CFM and FFM. In particular, the mask reconstruction module first concatenates the output of CFM with the output of FFM, and then uses convolutional layers to learn the interdependence of the two models. Finally, the mask information at different hierarchies is fused (that is, up-sampling and concatenating) to predict the I_{mask}. The design principle of our proposed CFFHNet is to alleviate the "foreground penetration" distortion by fusing low-level visual features (such as edges and colors) and high-level semantic features extracted from different spatial scales. In the following paragraphs, we explain the design principle and architecture of CFM and FFM in detail.

As shown in Fig. 3, CFM uses multiple parallel RRUs to further extract and fuse the outputs from the initial feature extraction module. We present the structure of a RRU in Fig. 4. The RRU combines residual learning and recurrent calculation to extract the global texture of images. Specifically, the convolutional long short term memory (Conv-LSTM) [28] is employed to model the information flow of the contextual textures with the recursive memory, in which the contextual texture correlations are transformed into structured cyclic dependencies to capture the texture information. The feature extraction procedures in the RRU can be formulated as

$$
\begin{aligned}
X_t &= H_{conv}(X),\\
i_t &= \sigma(W_{xi} \odot X_t + W_{hi} \odot H_{t-1} + W_{ci} \circ C_{t-1} + b_i),\\
f_t &= \sigma(W_{xf} \odot X_t + W_{hf} \odot H_{t-1} + W_{cf} \circ C_{t-1} + b_f),
\end{aligned}
$$

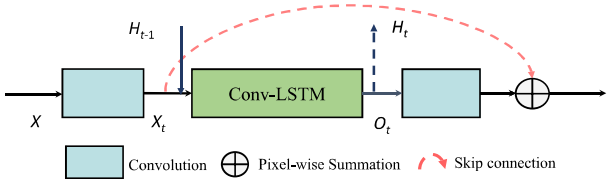


Fig. 4. Structure of the residual recurrent unit (RRU).

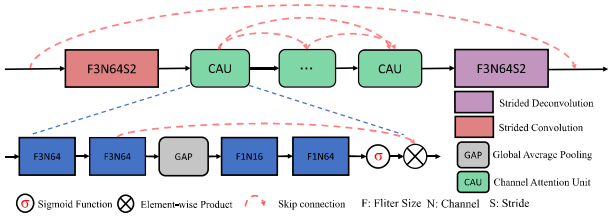


Fig. 5. Structure of the U-shape residual attention unit. F3N64S2 denotes the convolutional layers with filter size of 3×3, number of channels of 64 and stride size of 2.

$$
\begin{aligned}
C_t &= f_t \circ C_{t-1} + i_t \circ tanh(W_{xc} \odot X_t + W_{hc} \odot H_{t-1} + b_c),\\
o_t &= \sigma(W_{xo} \odot X_t + W_{ho} \odot H_{t-1} + W_{co} \circ C_t + b_0),\\
H_t &= o_t \circ tanh(C_t),\\
F_{RRU} &= H_{conv}(o_t) + X_t,
\end{aligned} \tag{1}
$$

where $H_{conv}(\cdot)$ represents the convolution layer, which takes the preceding spatial contextual texture correlation feature as input. Then given the input X_t in Eq (1), we recurrently revise the estimation of the spatial contextual texture in current RRU with the collaboration of the gate units (the input gate i_t, the forget gate f_t, and the output gate o_t), the cell body C_t, and the hidden state in the t_{th} state via Hadamard product ($\circ$) and convolution operation ($\odot$). W and b are the weight parameters and bias. The estimated spatial contextual correlation o_t and the updated cell C_t can be determined by the preceding features X_t and the previous hidden states H_{t-1}. Through the memory (the hidden state), the current stage can explore the useful components for refinement from the previous stage. Meanwhile, the recursive memory calculation decomposes the learning task into multiple stages, greatly releasing the learning burden. Note that similar to the sequence features representation, the Conv-LSTM in this study is used to recurrently extract the image features via multiple memory stages, which is more powerful for exploring the contextual texture compared to the standard convolution.

FFM refines the spatial contextual texture information extracted by CFM. The structure of FFM is similar to CFM, using multiple parallel U-shape residual attention units to refine the outputs of CFM. As shown in Fig. 5, the proposed U-shape residual attention unit consists of stride convolution, deconvolution, skip connection, and multiple channel attention units. The specific structure of the channel attention unit is depicted in Fig. 5. The design basis of the proposed U-shape residual attention unit considers three aspects: (a) To make the collaborative representation more effective, we introduce the channel attention unit to focus on the most informative specific-scale knowledge. (b) We adopt strided convolution to reduce spatial dimensions of features, thereby saving

computational resources. (c) Deconvolution is employed to prevent the loss of resolution information. The design fundamental of combining CFM and FFM is to finely represent the global texture of the hole image I_{hole}, so as to accurately estimate the error map I_{mask} between the hole image I_{hole} and its corresponding ground-truth image I_g.

B. Two-Stage Learning for Virtual View Synthesis

To improve the generative ability of the proposed CFFH-Net, we introduce generative adversarial training for virtual view synthesis. Compared with CNN-based methods, generative adversarial training shows impressive capability in generating visual pleasant contents. Meanwhile, due to the instability of generative adversarial training, some redundant high-frequency noise will be introduced [29]. To promote the texture generation while keeping fidelity, we further refine the output of ISN using RSN. Fig. 3 presents the framework of our proposed two-stage learning strategy.

1) Inference Sub-Network Training Stage: As depicted in Fig. 3, the generator aims to predict the error map I_{mask} between the hole image I_{hole} and its associated ground-truth image I_g. Then, the discriminator tries to distinguish between fake $I_{o(s_1)}$ (i.e., I_{hole}-I_{mask}) and ground-truth I_g. For the discriminator, we use the network architecture of the discriminator in the literature [31] as a reference. The adversarial loss for optimizing the generator and discriminator is defined as

$$\begin{aligned} L_{adv}(\theta_D) = &-\log D(I_g, \theta_D) \\ &- \log[1 D(I_{hole} - G(I_{hole}), \theta_D)], \end{aligned} \tag{2}$$

where $G(\cdot)$ and $D(\cdot)$ denote the functions of the generator and discriminator, respectively. θ_D is the parameters of the discriminator network.

The perceptual loss proposed by Johnson et al. can guide the generator to generate visually pleasing results [32], and this perceptual loss has been widely used in GAN-based image enhancement tasks [33], [34], [35]. Thus, this paper introduces the perceptual loss to improve the visual quality of virtual views. The specific definition of the perceptual loss is as follows:

$$L_{per} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(\Phi(I_{o(s_1),i}) - \Phi(I_{g,i}))^2}, \tag{3}$$

where $\Phi(\cdot)$ represents the pretrained 4th resblock before the 5th max-pooling layer of VGG19-net [36]. N is the size of a mini-batch. Moreover, the content loss, i.e., Mean Squared Error loss, has been commonly used as the constraint in visual tasks [37], [38], [39], which can ensure that the synthesized virtual view and the ground-truth are as close as possible at the pixel-level. Therefore, the content loss is further employed to constrain the generator, which is calculated as

$$L_{con} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(I_{o(s_1),i} - I_{g,i})^2}. \tag{4}$$

The whole loss function used in the generative adversarial training is given by

$$L_{gan} = L_{per} + \alpha \cdot L_{adv} + \beta \cdot L_{con}, \tag{5}$$

where the parameters α and β adjust the contribution of different loss components. The settings of parameters α and β are detailed in Section IV-C 3).

2) Refinement Sub-Network Training Stage: To guarantee the fidelity of the generated texture, we further use RSN to refine the output of ISN. Specifically, we still use the proposed generator as the CNN module and use edge-preservation loss and content loss to optimize the parameters of the CNN module. The definition of the proposed edge-preservation loss is as follows:

$$L_{edg} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(\phi(I_{o(s_2),i}) - \phi(I_{g,i}))^2}, \tag{6}$$

where ϕ is a differentiable Canny operator [40], [42] to extract the edge map of a given image. Then, we combine the proposed edge-preservation loss with the content loss (i.e., equation (4)) as the overall loss in the refinement stage, as shown below.

$$L_{cnn} = \frac{1}{N} \sum_{i=1}^{N} \sqrt{(I_{o(s_2),i} - I_{g,i})^2} + \gamma \cdot L_{edg}, \tag{7}$$

where the weight parameter γ is set to 0.05 to balance the loss terms. The ablation study of parameter γ is detailed in Section IV-C 3).

IV. Validation and Discussion

We conduct massive experiments on ten MVD sequences to evaluate the performance of the proposed method, seven state-of-the-art (SOTA) DIBR algorithms and two deep learning-based inpainting algorithms.

A. Experimental Setup

Training and validation datasets: Due to the limited number of MVD sequences, it is difficult to train our proposed DIBR algorithm based on deep learning. To effectively train our proposed method, we use 10k images from the COCO2017 dataset as well as randomly generated hole masks to synthesize massive hole/ground-truth image pairs. 80% and 20% of the collected image pairs serve as the training and validation datasets, respectively. Here, we briefly describe the used hole generation method. To effectively alleviate the dilemma faced by the general deep learning-based image inpainting algorithms in view synthesis, the generated holes need to have the following two properties: (a) the holes should destroy the texture information at the boundaries of the image foreground and background. Thus, the trained hole-filling model can protect the edge semantic information, effectively suppressing the 'foreground penetration' artifact. (b) the generated irregular holes should be diverse to avoid overfitting. To this end, this paper proposes a simple irregular hole generation method that can destroy images' foreground and background areas. First, the proposed hole generation method employs the instance segmentation algorithm [41] to roughly locate the image's foreground and background intersection area, which can help the Canny operator [42] to detect important edges of the foreground and background intersection area with a high

Fig. 6. Examples of images with irregular holes generated by the proposed hole generation method.

probability. Then, some important edges are selected by a randomized algorithm, and white lines similar to the edge direction are drawn in these edge areas. At the same time, the white lines are rotated repeatedly at small angles (the angle range can be set to 0°-15°) to generate hole regions. We have drawn many white lines with different angles and lengths in advance. Fig. 6 shows some images with holes generated by the proposed hole generation method.

Testing dataset: Six MVD sequences, including *Ballet* (BA) [43], *Breakdancers* (BR) [43], *Dancer* (DA) [44], *Painter* (PA) [45], *Classroom* (CL) [46], [47], and *ChocofountainBxl* (CH) [48], [49] are employed to test the performance of our method and the representative SOTA DIBR algorithms. Compared with the sequences BA, BR and DA, the sequences PA, CL and CH have richer texture information and more accurate depth information. The sequence warped from reference view 4 to virtual view 1 of BA is named as 'BA4→1'. Note that the hole image I_{hole} at the virtual viewpoint is obtained by the 3D image warping process based on the texture information, depth information, and camera settings information at the reference viewpoint. This 3D image warping procedure is not new and coincides with the approach applied in most peering methods [21], [50], [51].

Objective visual quality evaluation criterion: In our work, we employ three types of image/video quality assessment methods to evaluate the performance of the hole filling algorithms. Similar to other related studies [16], [21], the first category of performance evaluation indicators includes two general-purpose image quality assessment algorithms, namely Peak Signal-to-Noise Ratio (PSNR) and Structural SIMilarity (SSIM) [52]. PSNR evaluates the synthesized image quality by calculating the mean squared error of the pixel values between the synthesized image and its ground-truth image. SSIM measures the synthesized image quality by comparing the luminance, contrast, and structural similarities between the synthesized image and its ground-truth image. The second type of performance evaluation index consists of the morphological wavelet PSNR (MW-PSNR) [53], the morphological pyramid PSNR (MP-PSNR) [54] and the LOcal Geometric distortions and global Sharpness (LOGS) [55], which are designed for the local geometric distortion (i.e., the "foreground penetration" artifact) existing in the DIBR-synthesized images. The third type of evaluation index is the flickering distortion intensity (FDI) [56], which measures the visual quality of DIBR-synthesized videos by quantifying the temporal domain flickering distortion. The larger values of PSNR, SSIM, MP-PSNR, MW-PSNR, and LOGS, while the smaller value of FDI represent the better performance of the corresponding hole filling algorithm. The quality assessment criterion MP-PSNR, MW-PSNR, LOGS and FDI designed for DIBR-synthesized images can more accurately measure the performance of the hole filling algorithms compared with the general-purpose image quality evaluation methods PSNR and SSIM.

B. Implementation Details

In our CFFHNet, we set the hierarchical to 2, that is, the original scale and 1/2 down-sampling scale. Corresponding to the increasing resolution, we set the number of filters for each recurrent Conv-LSTM in the CFM to 32 and 64, respectively. The number of the channel attention unit in the FFM is set to 3. In the ISN training stage, we input the hole images into the network with a batch size of 16. The Adam optimizer [57] is used to optimize our ISN model with the initial learning rate at 10^{-3}, and the learning rate decreases to 1×10^{-6} with a decay rate of 0.7 after each epoch. We conduct experiments on an AMD Ryzen Threadripper 2950X CPU and one NVIDIA GTX 2080Ti GPU. We trained our network on TensorFlow platform. The implementation details of the RSN-based refinement are the same as the ISN training stage. According to the aforementioned settings, we trained our ISN and RSN for 20 epochs in two stages, respectively.

C. Ablation Studies

In this part, the network architecture, learning strategy and loss functions of the proposed work are studied for ablation, and the experimental validations are carried out on the sequences BA and BR.

1) Ablation Studies of the Hierarchy Number: This section investigates the effect of the number of network hierarchies on the model performance. Specifically, we design another two models with the hierarchy to 1, 2 and 3, and evaluate their performance on the MVD sequences BA and BR, respectively. The hierarchy numbers 1, 2 and 3 represent that the proposed method fills the hole regions based on the contextual texture relationships with the original spatial scale, the original and 1/2 spatial scales, and the original, 1/2 and 1/4 spatial scales, respectively. Table I tabulates the comparison results in terms of the visual quality assessment indicators PSNR, SSIM, MP-PSNR, MW-PSNR and LOGS.

When the number of network hierarchies is set to 1, the mean values of MP-PSNR, MW-PSNR, PSNR, SSIM and LOGS obtained by the proposed method on MVD sequences BA and BR are 32.48, 29.67, 28.30, 0.874 and 9.548, respectively. Except for SSIM, the network with a hierarchy number of 1 performs worse on both the MVD sequences BA and BR than the networks with a hierarchy number of 2 and 3. This suggests that fusing contextual texture relations on multiple

TABLE I

THE PSNR, SSIM, MP-PSNR, MW-PSNR AND LOGS VALUES OF THE PROPOSED NETWORK WITH DIFFERENT HIERARCHY NUMBERS ON MVD SEQUENCES BA AND BR. THE BEST PERFORMANCE IS HIGHLIGHTED

Seq.	BA4→1BA4→3BA5→2BA5→4BR4→1BR4→3BR5→2BR5→4								Avg.
Hierarchy Number: 1									
MP-PSNR↑	27.10	32.20	28.69	33.24	33.45	35.58	33.58	36.02	32.48
MW-PSNR↑	24.97	29.67	26.21	30.35	30.04	32.57	30.27	33.29	29.67
PSNR↑	24.36	30.43	24.75	30.30	26.41	30.72	27.43	31.99	28.30
SSIM↑	0.838	0.924	0.838	0.926	0.841	0.890	0.843	0.892	0.874
LOGS↑	9.671	9.362	9.761	9.562	9.778	9.784	9.376	9.087	9.548
Hierarchy Number: 2									
MP-PSNR↑	27.33	32.16	28.76	33.30	34.00	36.17	34.36	36.64	**32.84**
MW-PSNR↑	25.07	29.26	26.30	30.48	30.96	33.41	30.98	33.75	30.03
PSNR↑	24.97	30.64	25.28	30.71	27.98	30.90	27.76	31.90	**28.77**
SSIM↑	0.839	0.924	0.838	0.926	0.841	0.890	0.843	0.892	0.874
LOGS↑	9.645	9.332	9.790	9.867	9.751	9.664	9.469	9.521	**9.630**
Hierarchy Number: 3									
MP-PSNR↑	27.22	31.85	28.58	33.11	33.92	35.98	34.42	36.52	32.70
MW-PSNR↑	25.05	29.77	26.24	30.64	30.81	33.53	30.73	33.88	**30.08**
PSNR↑	24.36	29.80	24.73	30.18	27.56	30.69	28.86	33.46	28.71
SSIM↑	0.842	0.924	0.840	0.930	0.879	0.917	0.885	0.922	**0.892**
LOGS↑	9.657	9.468	9.718	9.510	9.829	9.794	9.386	9.104	9.558

TABLE II

THE PSNR, SSIM, MP-PSNR, MW-PSNR AND LOGS VALUES OF THE PROPOSED NETWORKS WITH TWO HOLE FILLING STRATEGIES ON MVD SEQUENCES BA AND BR. THE BEST PERFORMANCE IS HIGHLIGHTED

Seq.	BA4→1BA4→3BA5→2BA5→4BR4→1BR4→3BR5→2BR5→4								Avg.
Filling Strategy I: $I_{hole} \rightarrow$ Network $\rightarrow I_g$									
MP-PSNR↑	27.45	32.06	28.64	33.30	34.00	36.00	34.19	35.97	32.70
MW-PSNR↑	25.34	29.65	26.30	30.38	30.63	33.02	30.60	33.25	29.89
PSNR↑	24.61	30.14	25.33	30.83	26.89	30.66	28.02	32.18	28.58
SSIM↑	0.855	0.931	0.857	0.937	0.890	0.920	0.893	0.922	**0.900**
LOGS↑	9.733	9.450	9.693	9.441	9.822	9.741	9.401	9.445	9.59
Filling Strategy II: $I_{hole} \rightarrow$ Network $\rightarrow I_{mask} \rightarrow I_g$									
MP-PSNR↑	27.33	32.16	28.76	33.30	34.00	36.17	34.36	36.64	**32.84**
MW-PSNR↑	25.07	29.26	26.30	30.48	30.96	33.41	30.98	33.75	**30.03**
PSNR↑	24.97	30.64	25.28	30.71	27.98	30.90	27.76	31.90	**28.77**
SSIM↑	0.839	0.924	0.838	0.926	0.841	0.890	0.843	0.892	0.874
LOGS↑	9.645	9.332	9.790	9.867	9.751	9.664	9.469	9.521	**9.630**

TABLE III

THE IMPACT OF THE PARAMETER β ON THE PERFORMANCE OF THE PROPOSED CFFHNET. THE BEST PERFORMANCE IS HIGHLIGHTED

Seq.	BA4→1BA4→3BA5→2BA5→4BR4→1BR4→3BR5→2BR5→4								Avg.
β = 1									
MP-PSNR↑	27.45	31.18	28.81	32.54	32.43	33.89	32.08	34.10	31.56
MW-PSNR↑	24.97	28.46	25.99	28.94	29.42	31.34	29.01	31.75	28.73
PSNR↑	20.32	22.87	21.39	24.13	25.16	28.71	25.57	29.62	24.72
SSIM↑	0.818	0.890	0.824	0.896	0.837	0.882	0.832	0.885	0.858
LOGS↑	9.702	9.447	9.706	9.533	9.680	9.653	9.551	9.482	9.594
β = 5									
MP-PSNR↑	27.33	32.16	28.76	33.30	34.00	36.17	34.36	36.64	**32.84**
MW-PSNR↑	25.07	29.26	26.30	30.48	30.96	33.41	30.98	33.75	**30.03**
PSNR↑	24.97	30.64	25.28	30.71	27.98	30.90	27.76	31.90	**28.77**
SSIM↑	0.839	0.924	0.838	0.926	0.841	0.890	0.843	0.892	0.874
LOGS↑	9.645	9.332	9.790	9.867	9.751	9.664	9.469	9.521	**9.630**
β = 10									
MP-PSNR↑	27.14	32.10	28.75	33.06	33.49	35.55	33.79	36.01	32.49
MW-PSNR↑	25.02	29.47	26.14	29.97	30.18	32.55	30.33	33.10	29.60
PSNR↑	25.01	30.54	25.29	30.43	26.82	30.92	27.64	31.58	28.53
SSIM↑	0.858	0.929	0.859	0.935	0.888	0.927	0.889	0.926	**0.901**
LOGS↑	9.585	9.351	9.782	9.769	9.762	9.643	9.493	9.575	9.620

TABLE IV

THE IMPACT OF THE PARAMETER γ ON THE PERFORMANCE OF THE PROPOSED CFFHNET. THE BEST PERFORMANCE IS HIGHLIGHTED

Seq.	BA4→1BA4→3BA5→2BA5→4BR4→1BR4→3BR5→2BR5→4								Avg.
γ = 0.01									
MP-PSNR↑	27.18	31.83	28.38	33.55	34.07	35.93	34.15	36.18	32.66
MW-PSNR↑	25.07	28.98	26.06	30.41	30.79	33.00	30.14	33.25	29.71
PSNR↑	24.57	29.60	25.48	30.86	27.19	30.24	28.38	32.50	28.60
SSIM↑	0.850	0.927	0.855	0.934	0.899	0.930	0.901	0.932	**0.900**
LOGS↑	9.637	9.366	9.599	9.299	9.872	9.863	9.476	9.109	9.530
γ = 0.05									
MP-PSNR↑	27.33	32.16	28.76	33.30	34.00	36.17	34.36	36.64	**32.84**
MW-PSNR↑	25.07	29.26	26.30	30.48	30.96	33.41	30.98	33.75	**30.03**
PSNR↑	24.97	30.64	25.28	30.71	27.98	30.90	27.76	31.90	**28.77**
SSIM↑	0.839	0.924	0.838	0.926	0.841	0.890	0.843	0.892	0.874
LOGS↑	9.645	9.332	9.790	9.867	9.751	9.664	9.469	9.521	**9.630**
γ = 0.5									
MP-PSNR↑	27.14	31.58	28.58	33.32	34.03	36.01	34.10	35.93	32.59
MW-PSNR↑	25.09	28.78	26.13	30.52	30.86	33.22	30.56	32.85	29.75
PSNR↑	24.83	30.16	25.39	31.32	27.48	31.00	27.07	32.32	28.70
SSIM↑	0.858	0.930	0.859	0.939	0.898	0.929	0.896	0.931	**0.900**
LOGS↑	9.702	9.447	9.706	9.533	9.680	9.653	9.551	9.482	9.590

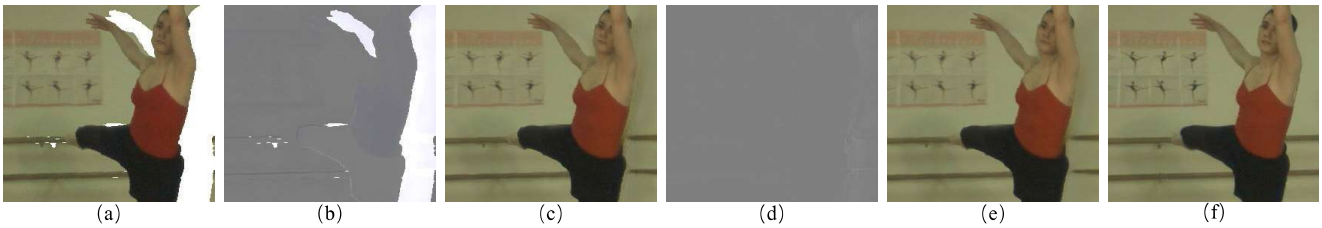
spatial scales is more helpful in filling the hole regions. The network with a hierarchy number of 2 performs the best on the performance evaluation metrics MP-PSNR, PSNR and LOGS. The network with a hierarchy number of 3 performs the best on the performance evaluation metrics MW-PSNR and SSIM. Compared to the model with hierarchy number of 3, the model with hierarchy number of 2 gains similar or better scores on the DIBR-synthesized image quality assessment indexes while with less model parameters and complexity. Combining the above experimental results, we finally set the hierarchy number of the proposed network as 2 to optimize the performance of the proposed hole filling network.

2) Ablation Study of the "$I_{hole} = I_g + I_{mask}$": We investigate the ablation of two types of hole filling strategies, including the direct prediction to the ground truth, and the residual learning between the hole image and the ground truth to generate the final result via subtraction. Table II shows the performance of the models obtained by the proposed networks with two hole filling strategies on the MVD sequence BA and BR, respectively. The filling strategy I-based network, that is, directly learns the filling process from the hole image to its ground-truth image, achieves the MP-PSNR, MW-PSNR, PSNR, SSIM and LOGS mean values of 32.70, 29.89, 28.58, 0.90 and 9.59, respectively. The proposed network based on the filling strategy II, that is, repairing the hole regions by learning the error map between the hole image and its ground-truth image, outperforms the filling strategy I-based network in the performance indicators MP-PSNR, MW-PSNR, PSNR and LOGS. The reason may lie in that when learning the projection from the hole image to its ground-truth image directly, the hole-free regions may be degraded due to the local connectivity and translation equivariance of convolution operation. By contrast, defining the problem to be "$I_{hole} = I_g + I_{mask}$" greatly simplifies the learning task, where only the residue (hole regions) are required to be focused on during the inference.

TABLE V

THE PERFORMANCE COMPARISON OF ISN, RSN, AND THE COMPLETE CFFHNET ON THE EVALUATION INDEXES PSNR, SSIM, MP-PSNR, MW-PSNR, AND LOGS. THE BEST PERFORMANCE IS HIGHLIGHTED

Seq.	PSNR↑			SSIM↑			MP-PSNR↑			MW-PSNR↑			LOGS↑		
	ISN	RSN	CFFHNet	ISN	RSN	CFFHNet	ISN	RSN	CFFHNet	ISN	RSN	CFFHNet	ISN	RSN	CFFHNet
BA4→1	24.55	24.53	**24.97**	0.820	0.816	**0.839**	27.22	27.27	**27.33**	25.03	25.03	**25.07**	9.637	9.633	**9.645**
BA4→3	30.28	29.91	**30.64**	0.911	0.907	**0.924**	32.09	32.03	**32.16**	29.18	29.11	**29.26**	9.346	9.460	**9.532**
BA5→2	24.66	24.83	**25.28**	0.811	0.803	**0.838**	28.74	28.69	**28.76**	26.26	26.25	**26.30**	9.766	9.764	**9.790**
BA5→4	29.63	29.74	**30.71**	0.915	0.902	**0.926**	33.12	33.17	**33.30**	30.36	30.36	**30.48**	9.853	9.850	**9.867**
BR4→1	26.47	26.53	**27.98**	0.834	0.830	**0.841**	33.33	32.47	**34.00**	29.91	29.23	**30.96**	9.737	9.697	**9.751**
BR4→3	30.62	30.36	**30.90**	0.876	0.869	**0.890**	35.55	34.51	**36.17**	32.68	31.91	**33.41**	9.625	9.619	**9.664**
BR5→2	27.49	27.10	**27.76**	0.838	0.827	**0.843**	33.15	32.24	**34.36**	29.96	29.17	**30.98**	9.429	9.428	**9.469**
BR5→4	31.20	31.17	**31.90**	0.878	0.870	**0.892**	35.93	34.55	**36.64**	33.05	32.15	**33.75**	9.444	9.449	**9.521**



(a) (b) (c) (d) (e) (f)

Fig. 7. An example of a phased visualization of our network learning. (a) is the input image (i.e., the hole image). (b) is the predicted residue I_{mask} in ISN. (c) is the base recovery result generated by ISN (i.e., Fig. 7 (a) - Fig. 7 (b)). (d) is the predicted residue I_{mask} in RSN. (e) is the final recovery result generated by RSN (i.e., Fig. 7 (c) - Fig. 7 (d)). (f) is the ground-truth image.

3) Parameter Sensitivity Analysis: For the inference sub-network, the balanced weights of perceptual loss and adversarial loss are referred to the pioneering study of GAN on image reconstruction by setting to 1 and 10^{-3} (the parameter α) [31]. In our work, the weight parameter β of the content loss is set to 5 to guarantee the consistency and facticity of the inferenced contents. Additionally, we set the parameter β to another two values 1 and 10, and train two models. The comparison results on the test datasets are tabulated in Table III, showing that the model with β to 5 gains the highest scores on average on the evaluated indexes MP-PSNR, MW-PSNR, PSNR and LOGS. A possible reason for the best performance achieved by the model with β to 5 is that a larger weight of β may weaken the inference capability of generator while a smaller value causes the false content generation with worse consistency.

For the refinement sub-network, we adjust the weight value of the edge-preservation loss by additionally setting the parameter γ to 0.5 and 0.01 to train another two models. The quantitative results are shown in Table IV, showing that the model with the parameter γ to 0.05 gains the highest scores on average on the performance evaluation indicators. A possible reason for the best performance achieved by the model with γ to 0.05 that a larger or a smaller value of the parameter γ will produce the over-sharp or over-smooth result.

4) Ablation Studies of Inference Sub-Network and Refinement Sub-Network: We introduce ISN with the adversarial learning to enhance the texture generation ability of our method. After ISN, we further employ RSN to restrain noise and refine contents with better fidelity and affinity. To further analyze the reasonableness of the proposed two-stage learning strategy, we respectively verify the performance of only ISN or RSN architecture on MVD sequences BA and BR. The training details of the ablation experiments are similar to training the whole model. For the fair comparison, we trained the ISN and RSN models with the same hyper-parameter setting and the similar model parameters to the complete CFFHNet. Table V shows the PSNR, SSIM, MP-PSNR, MW-PSNR and LOGS values of the ISN, RSN, and complete CFFHNet on 10 synthesized sequences. The best results are highlighted in boldface. Experimental results show that the complete CFFHNet performs better than the ISN and RSN, indicating that the ISN and RSN can complementarily improve the visual quality of the synthesized views.

For more convincing and better understanding to the two-stage framework, we visualize the intermediate results, including the predicted residue and the corresponding base recovery result by ISN, as well as the refined components and final result by RSN. As shown in Fig. 7, with the guidance of the GAN loss and perceptual loss, ISN can infer photo-realistic contents, but GAN-based model tends to generate results with obvious noises and visual inconsistency to the ground truth. Through the detail refinement of RSN, the final result enjoys better fidelity and clarity with the supplementary of the refined components.

5) Ablation Studies of the Proposed Training Data Generation Method: Here, we conduct ablation research on the effectiveness of our training data generation method for view synthesis. Compared with the warped data (the holes occurring in disoclussion areas), we believe that using the data polluted in the foreground and background intersection areas (as shown in Fig. 6) is more beneficial to the ability of the hole-filling network to suppress "foreground penetration" artifacts. To validate this hypothesis, we compared the performance of the hole-filling model trained by the proposed generated training data with the hole-filling model retrained by the warped data using MVD sequences. Table VI presents the performance comparison of the above two models on sequences CH and PA. The data for training model I is obtained from the proposed

Fig. 8. Subjective visual quality comparisons of model I and model II. (a) is the hole image. (b) is the recovered result by model I. (c) is the recovered result by model II. (d) is the ground-truth image.

training data generation method. Model II is retrained from 2,000 hole images generated by MVD sequences (BA, BR, DA, and CL) through 3D warping on the basis of model I. From the experimental results, we can easily see that model II, retrained with 3D warping data, has a decline in six performance indicators, which directly demonstrates the effectiveness of the proposed training data generation scheme in the hole-filling of view synthesis. In addition, Fig. 8 shows some hole-filling examples of the two models. The yellow circle region has obvious "foreground penetration" artifacts, which indicates that model II has weakened the suppression ability of "foreground penetration" artifacts compared with model I. The hole-filling model learned from our generated training data performs better in subjective visual quality. Thus, the proposed training data generation scheme is beneficial to solve the "foreground penetration" artifact introduced by hole filling in view synthesis. Moreover, note that as described in Section II-B, we improve the ability of the obtained hole-filling model to suppress the "foreground penetration" artifact (that is, effectively protect image semantic information) in view synthesis by combining the training data generation method and the design of the hole-filling network. The first stage of our hole-filling network repairs the synthesized images based on the GAN. The second stage of our hole-filling network uses CNN to fine-tune the output of the first stage network, which mainly focuses on protecting the edges of synthesized images constrained by edge loss. The two-stage sub-network's coarse- and fine-grained fusion modules integrate the global and local contextual textures to protect the image semantic information by using visual signals with different resolutions.

TABLE VI

ABLATION EXPERIMENT RESULTS OF THE PROPOSED TRAINING DATA GENERATION SCHEME

Sequence: CH02, 11, 13, 22→12						
Metric	PSNR↑	SSIM↑	MP-PSNR↑	MW-PSNR↑	LOGS↑	FDI↓
Model I	29.24	0.939	34.94	31.09	9.646	10.746
Model II	24.16	0.930	32.93	30.75	9.310	10.762
Sequence: CH01, 02, 03, 11, 13, 21, 22, 23→12						
Metric	PSNR↑	SSIM↑	MP-PSNR↑	MW-PSNR↑	LOGS↑	FDI↓
Model I	29.02	0.937	34.69	31.84	9.608	10.727
Model II	28.99	0.934	32.48	30.35	9.281	10.741
Sequence: PA4, 5, 7, 8→6						
Metric	PSNR↑	SSIM↑	MP-PSNR↑	MW-PSNR↑	LOGS↑	FDI↓
Model I	34.29	0.964	38.92	35.28	9.401	10.262
Model II	33.75	0.957	38.71	34.83	9.152	10.291
Sequence: PA2, 3, 4, 5, 7, 8, 9, 10→6						
Metric	PSNR↑	SSIM↑	MP-PSNR↑	MW-PSNR↑	LOGS↑	FDI↓
Model I	36.77	0.977	41.62	37.84	8.988	10.278
Model II	35.55	0.963	40.85	37.29	8.414	10.285

D. Objective Visual Quality Comparison

To verify the effectiveness of our CFFHNet, we collect a large amount of traditional patch matching-based DIBR methods and deep learning-based image inpainting methods for comparison. The traditional patch matching-based DIBR methods include Criminsi et al.'s method [14], Daribo et al.'s method [15], Ahn et al.'s method [16], Zhu et al.'s method [19]], Luo et al.'s method [21], VSRS [58] and RVS [59]. Moreover, we select the GAN-based image inpainting algorithms that are good at filling large-area irregular holes, including Li et al.'s method [22] and Yu et al.'s method [26] as the competing algorithms. These methods may effectively fill the large-area irregular holes caused by the DIBR technique. Table VII shows the average PSNR, SSIM, MP-PSNR, MW-PSNR, LOGS and FDI values of our CFFHNet, the patch matching-based DIBR algorithms, and the GAN-based image inpainting algorithms on MVD sequences BA, BR and DA.

TABLE VII

OBJECTIVE VISUAL QUALITY EVALUATIONS OF OUR CFFHNET AND OTHER SOTA DIBR ALGORITHMS USING PSNR, SSIM, MP-PSNR, MW-PSNR, LOGS AND FDI. THE BEST AND SECOND-BEST RESULTS ARE SHOWN IN BOLD AND UNDERLINED, RESPECTIVELY

Seq.	PSNR↑									SSIM↑								
	Criminsi	Daribo	Ahn	Zhu	Luo	VSRS	Li	Yu	Pro.	Criminsi	Daribo	Ahn	Zhu	Luo	VSRS	Li	Yu	Pro.
BA4→1	22.76	22.56	23.27	23.54	24.33	22.23	17.64	22.14	**24.97**	0.739	0.713	0.748	0.755	0.779	0.761	0.723	0.769	**0.839**
BA4→3	25.08	27.63	28.15	28.72	29.31	25.93	29.26	25.84	**30.64**	0.839	0.835	0.847	0.850	0.859	0.851	0.885	0.861	**0.924**
BA5→2	24.38	23.97	24.29	25.10	**25.33**	23.89	18.90	23.20	25.28	0.742	0.720	0.739	0.755	0.776	0.765	0.746	0.769	**0.838**
BA5→4	26.56	29.60	30.54	31.93	**32.06**	27.60	29.02	26.45	30.71	0.847	0.845	0.855	0.863	0.866	0.858	0.888	0.866	**0.926**
BR4→1	25.87	26.92	26.91	27.07	27.59	27.03	27.76	21.76	**27.98**	0.764	0.764	0.774	0.775	0.788	0.781	0.822	0.752	**0.841**
BR4→3	29.74	30.20	30.40	30.41	30.75	29.61	30.68	22.82	**30.90**	0.820	0.815	0.822	0.823	0.825	0.813	0.868	0.797	**0.890**
BR5→2	26.23	27.55	27.32	27.66	**28.55**	26.40	28.16	21.89	27.76	0.766	0.768	0.774	0.776	0.789	0.761	0.821	0.752	**0.843**
BR5→4	30.24	30.86	30.27	31.14	31.62	30.25	31.08	23.46	**31.90**	0.822	0.818	0.823	0.823	0.821	0.813	0.870	0.803	**0.892**
DA1→5	26.74	27.64	27.80	28.07	30.46	26.42	27.52	27.60	**31.48**	0.941	0.943	0.945	0.944	0.949	0.943	0.959	0.962	**0.967**
DA5→9	26.69	27.56	27.32	27.96	30.25	26.22	27.73	28.25	**32.16**	0.940	0.942	0.943	0.944	0.949	0.941	0.958	0.961	**0.967**
Average	26.43	27.45	27.63	28.16	29.03	26.56	26.78	24.34	**29.38**	0.822	0.816	0.827	0.831	0.840	0.829	0.854	0.829	**0.893**

Seq.	MP-PSNR↑									MW-PSNR↑								
	Criminsi	Daribo	Ahn	Zhu	Luo	VSRS	Li	Yu	Pro.	Criminsi	Daribo	Ahn	Zhu	Luo	VSRS	Li	Yu	Pro.
BA4→1	27.31	26.48	27.50	27.67	**28.13**	27.26	25.96	26.05	27.33	24.17	23.62	24.72	24.53	**25.24**	24.80	24.29	23.28	25.07
BA4→3	30.57	31.04	31.44	31.85	**32.24**	32.10	32.01	30.26	32.16	27.15	28.07	28.62	28.52	**29.58**	29.18	29.20	27.16	29.26
BA5→2	28.30	27.70	28.05	28.45	**29.06**	28.52	26.92	26.63	28.76	24.98	24.41	24.62	25.14	25.68	25.43	24.32	23.58	**26.30**
BA5→4	30.40	32.17	32.34	33.01	**33.31**	33.25	32.64	31.02	33.30	26.35	28.63	28.93	29.31	29.61	28.89	28.88	27.70	**30.48**
BR4→1	31.74	32.86	33.19	31.75	33.74	33.50	33.47	31.17	**34.00**	28.34	29.41	29.85	27.86	30.24	30.03	30.05	27.65	**30.96**
BR4→3	35.65	35.92	35.73	34.59	35.73	35.54	35.40	32.04	**36.17**	32.60	32.67	32.62	30.88	32.69	32.55	32.46	28.68	**33.41**
BR5→2	32.42	33.21	33.03	32.12	32.95	33.03	33.55	30.97	**34.36**	28.70	29.46	29.24	28.16	29.53	29.13	29.08	27.49	**30.98**
BR5→4	35.25	35.04	35.40	33.81	35.77	34.58	35.66	32.14	**36.64**	32.24	31.95	32.19	29.91	32.58	31.27	32.27	28.93	**33.75**
DA1→5	29.94	30.55	30.71	26.53	32.19	29.74	30.10	31.71	**33.01**	27.07	27.46	27.68	23.42	28.65	27.05	27.99	27.71	**32.69**
DA5→9	30.23	30.78	30.82	30.73	32.60	29.91	29.84	31.90	**33.18**	27.27	27.50	27.26	27.71	29.30	27.12	28.28	28.51	**31.67**
Average	31.18	31.58	31.82	31.05	32.57	31.74	31.56	30.39	**32.89**	27.89	28.32	28.57	27.54	29.31	28.55	28.68	27.07	**30.46**

Seq.	LOGS↑									FDI↓								
	Criminsi	Daribo	Ahn	Zhu	Luo	VSRS	Li	Yu	Pro.	Criminsi	Daribo	Ahn	Zhu	Luo	VSRS	Li	Yu	Pro.
BA4→1	9.592	9.605	9.600	9.590	**9.793**	9.598	9.521	9.546	9.645	10.93	12.25	10.98	10.92	10.87	11.18	10.91	10.94	**10.84**
BA4→3	9.293	9.293	**9.599**	9.297	9.596	9.286	9.268	9.294	9.532	10.84	11.32	10.87	10.88	**10.77**	10.85	10.87	10.83	10.78
BA5→2	9.669	9.660	9.645	9.655	**9.795**	9.684	9.611	9.637	9.790	10.83	11.97	10.89	10.86	**10.74**	11.15	10.86	10.85	10.75
BA5→4	9.070	9.025	9.017	8.991	9.810	9.090	9.775	9.758	**9.867**	10.81	11.27	10.83	10.83	**10.75**	10.86	10.78	10.82	10.77
BR4→1	9.605	9.611	9.603	9.605	9.717	9.633	9.638	9.663	**9.751**	10.84	10.99	10.69	**10.64**	10.84	10.70	10.92	10.90	10.69
BR4→3	9.522	9.535	9.521	9.521	9.632	9.547	9.520	9.622	**9.664**	10.48	10.54	10.48	10.49	**10.47**	10.58	10.52	10.59	10.49
BR5→2	9.441	9.369	9.453	9.461	9.460	9.417	9.220	9.339	**9.469**	10.86	11.02	10.67	10.66	**10.01**	10.78	10.82	10.78	10.40
BR5→4	9.487	9.482	9.458	9.482	**9.581**	9.414	9.230	9.294	9.521	10.59	10.69	10.58	**10.52**	10.57	10.66	10.64	10.60	10.59
DA1→5	8.105	8.145	8.110	8.225	8.338	8.098	8.011	8.106	**8.389**	11.68	11.67	11.69	11.73	**10.68**	11.60	11.65	11.77	10.79
DA5→9	8.108	8.122	8.091	8.119	8.317	8.080	8.007	8.070	**8.395**	11.67	11.67	11.69	11.74	**10.50**	11.60	11.65	11.79	11.18
Average	9.189	9.185	9.210	9.194	**9.404**	9.185	9.180	9.233	9.402	10.95	11.34	10.94	10.93	**10.62**	11.00	10.96	10.99	10.73

The best and second-best results are respectively highlighted on red and blue. On sequences BA5→2, BA5→4, and BR5→2, our proposed method has lower PSNR values than the Luo's method [21]. When the interval between the virtual and reference viewpoints is smaller, the warped image contains fewer holes. The temporal information-based DIBR method [21] can utilize more useful information to fill the holes, which makes the visual quality of the rendered image higher. When the virtual viewpoint is far from the reference viewpoint, the warped image contains numerous holes. Hence, there are little visible areas to be utilized for temporal information-based DIBR methods [19], [21], which limits the superiority of the DIBR algorithms based on video sequence over the single frame-DIBR algorithm. The PSNR results in Table VII show the performance of the proposed algorithm outperforms other competing DIBR algorithms on the MVD sequences with a large distance between the virtual and reference viewpoints, i.e., BA4→1, BR4→1, DA1→5, and DA5→9. Notably that the average SSIM values of our proposed method are the best on all test MVD sequences. This is also because SSIM focuses on perceiving the structural information of images, and we also adopt edge loss to constrain the structure of DIBR-synthesized views.

On sequences BR and DA, the proposed CFFHNet outperforms all competing methods on the evaluation metrics MP-PSNR and MW-PSNR. The MP-PSNR and MW-PSNR values obtained by the proposed method on sequence BA are also second only to the Luo et al.'s hole filling method based on temporal information. In addition, the overall performance of the proposed method is better than the competing methods in terms of the mean values of MP-PSNR and MW-PSNR on sequences BA, BR, and DA. On the evaluation index LOGS, the proposed method performs best on sequences BA5→4, BR4→1, BR4→3, BR5→2, DA1→5 and DA1→5. The mean LOGS values obtained by the proposed method on sequences BA, BR and DA are also second only to Luo et al.'s method. On the evaluation index FDI, Luo et al.'s method shows the best comprehensive performance on sequences BA, BR and DA, which indicates that filling hole regions based on temporal information can effectively protect temporal consistency of

TABLE VIII

PERFORMANCE COMPARISONS OF OUR CFFHNET AND THE RVS ALGORITHMS ON MVD SEQUENCES PA, CL AND CH. THE BEST AND SECOND-BEST RESULTS ARE SHOWN IN BOLD AND UNDERLINED, RESPECTIVELY

Seq.	PSNR↑				SSIM↑				MP-PSNR↑			
	Li	Yu	RVS	Pro.	Li	Yu	RVS	Pro.	Li	Yu	RVS	Pro.
PA3, 4, 6, 7→5	32.17	32.58	35.12	**36.82**	0.939	0.936	0.945	**0.978**	39.17	38.60	40.04	**41.73**
PA4, 5, 7, 8→6	30.31	31.46	32.81	**34.29**	0.918	0.906	0.926	**0.964**	36.34	35.75	37.77	**38.92**
PA5, 6, 8, 9→7	27.19	27.27	29.49	**30.36**	0.867	0.864	0.887	**0.932**	32.44	32.46	33.37	**34.45**
CL3, 4, 6, 7→5	33.04	33.54	34.95	**36.64**	0.934	0.932	0.941	**0.973**	37.14	37.20	38.30	**40.00**
CL4, 5, 7, 8→6	30.12	30.98	31.90	**33.38**	0.918	0.916	0.929	**0.958**	34.28	34.17	35.25	**36.37**
CL5, 6, 8, 9→7	28.05	28.74	29.62	**30.49**	0.887	0.889	0.894	**0.938**	31.98	31.76	32.99	**34.07**
CH01, 10, 12, 21→11	25.63	25.69	26.02	**27.01**	0.867	0.868	0.872	**0.886**	31.30	31.60	34.57	**35.89**
CH02, 11, 13, 22→12	27.66	28.06	28.25	**29.24**	0.907	0.905	0.915	**0.939**	31.01	31.68	33.62	**34.94**
CH03, 12, 14, 23→13	23.25	23.78	24.84	**26.48**	0.879	0.876	**0.892**	0.881	31.16	31.47	32.24	**34.62**
Average	28.86	29.12	30.33	**31.64**	0.902	0.899	0.911	**0.939**	33.87	33.85	35.35	**36.78**
PA1, 2, 3, 4, 6, 7, 8, 9→5	34.83	35.26	35.68	**37.43**	0.947	0.941	0.951	**0.981**	39.38	39.52	40.55	**42.56**
PA2, 3, 4, 5, 7, 8, 9, 10→6	34.37	34.48	34.95	**36.77**	0.933	0.939	0.945	**0.977**	37.14	37.80	38.91	**41.62**
PA3, 4, 5, 6, 8, 9, 10, 11→7	31.34	31.92	32.22	**33.60**	0.902	0.903	0.913	**0.957**	31.19	32.13	34.13	**35.51**
CL1, 2, 3, 4, 6, 7, 8, 9→5	33.92	34.16	35.00	**36.72**	0.931	0.935	0.944	**0.970**	37.12	37.26	38.19	**40.19**
CL2, 3, 4, 5, 7, 8, 9, 10→6	31.86	32.64	33.87	**35.56**	0.928	0.930	0.941	**0.969**	34.98	35.04	36.18	**38.77**
CL3, 4, 5, 6, 8, 9, 10, 11→7	30.78	31.84	32.30	**33.58**	0.892	0.896	0.907	**0.951**	32.08	32.16	33.80	**35.05**
CH00, 01, 02, 10, 12, 20, 21, 22→11	25.58	26.38	27.95	**28.93**	0.896	0.894	0.916	**0.920**	31.38	31.99	35.23	**35.56**
CH01, 02, 03, 11, 13, 21, 22, 23→12	26.29	26.60	28.04	**29.02**	0.896	0.897	0.917	**0.937**	31.63	31.96	34.35	**34.69**
CH02, 03, 04, 12, 14, 22, 23, 24→13	23.90	24.54	24.93	**26.51**	0.879	0.877	**0.895**	0.883	31.46	31.89	32.23	**34.60**
Average	30.32	30.87	31.66	**33.13**	0.911	0.912	0.925	**0.949**	34.04	34.42	35.95	**37.62**
	MW-PSNR↑				LOGS↑				FDI↓			
	Li	Yu	RVS	Pro.	Li	Yu	RVS	Pro.	Li	Yu	RVS	Pro.
PA3, 4, 6, 7→5	34.23	33.26	36.60	**38.92**	9.215	9.205	9.369	**9.398**	10.431	10.422	**10.325**	**10.325**
PA4, 5, 7, 8→6	33.91	32.88	34.37	**35.28**	9.303	9.315	**9.437**	9.401	10.297	10.291	10.276	**10.262**
PA5, 6, 8, 9→7	30.38	30.44	31.21	**32.25**	9.217	9.206	**9.397**	9.237	10.351	10.359	**10.305**	**10.305**
CL3, 4, 6, 7→5	33.26	34.39	35.18	**37.49**	9.336	9.340	9.498	**9.520**	10.453	10.498	**10.305**	**10.305**
CL4, 5, 7, 8→6	32.06	32.27	33.14	**34.05**	9.301	9.293	**9.491**	9.413	10.401	10.398	10.299	**10.298**
CL5, 6, 8, 9→7	28.36	28.48	29.33	**30.37**	9.012	8.996	**9.353**	9.128	10.509	10.513	**10.400**	10.401
CH01, 10, 12, 21→11	27.03	27.22	28.32	**28.71**	9.419	9.427	**9.511**	9.482	10.778	10.749	**10.542**	**10.542**
CH02, 11, 13, 22→12	30.28	30.93	31.08	**31.09**	9.482	9.508	**9.674**	9.646	10.753	10.756	**10.742**	10.746
CH03, 12, 14, 23→13	28.06	28.01	28.68	**29.08**	9.165	9.181	**9.529**	9.372	10.792	10.798	10.700	**10.693**
Average	30.84	30.88	31.99	**33.03**	9.272	9.275	**9.473**	9.400	10.529	10.532	10.433	**10.430**
PA1, 2, 3, 4, 6, 7, 8, 9→5	36.48	36.24	36.99	**39.51**	9.115	9.131	9.335	**9.385**	10.515	10.521	10.318	**10.315**
PA2, 3, 4, 5, 7, 8, 9, 10→6	35.02	35.47	36.05	**37.84**	8.881	8.875	9.169	**8.988**	10.386	10.391	10.284	**10.278**
PA3, 4, 5, 6, 8, 9, 10, 11→7	31.33	31.26	32.26	**33.60**	8.848	8.853	**9.254**	9.029	10.395	10.391	**10.280**	10.281
CL1, 2, 3, 4, 6, 7, 8, 9→5	34.03	34.14	35.26	**37.55**	9.501	9.543	**9.796**	9.761	10.397	10.389	**10.280**	10.281
CL2, 3, 4, 5, 7, 8, 9, 10→6	33.98	34.16	35.04	**36.70**	8.970	8.981	**9.493**	9.190	10.396	10.398	10.297	**10.296**
CL3, 4, 5, 6, 8, 9, 10, 11→7	28.17	28.43	29.77	**31.02**	8.866	8.847	**9.273**	9.060	10.391	10.385	10.266	**10.265**
CH00, 01, 02, 10, 12, 20, 21, 22→11	28.08	28.92	30.70	**31.07**	9.429	9.423	**9.649**	9.603	10.736	10.850	**10.500**	10.501
CH01, 02, 03, 11, 13, 21, 22, 23→12	30.95	29.19	31.80	**31.84**	9.473	9.434	**9.638**	9.608	10.744	10.775	**10.725**	10.727
CH02, 03, 04, 12, 14, 22, 23, 24→13	27.03	27.58	28.75	**29.12**	9.282	9.253	**9.544**	9.385	10.786	10.783	10.679	**10.676**
Average	31.67	31.71	32.96	**34.25**	9.152	9.149	**9.461**	9.334	10.527	10.543	10.403	**10.402**

the synthesized sequences. Pleasingly, the proposed method, filling the holes based only on the contextual texture relationship of the synthesized image itself, performs overall second only to Luo et al.'s method in terms of temporal consistency. In summary, the performance of the proposed single-frame-based hole-filling method is very close to that of the temporal information modeling-based hole filling method designed by Luo et al. in terms of DIBR-synthesized image/video quality evaluation metrics.

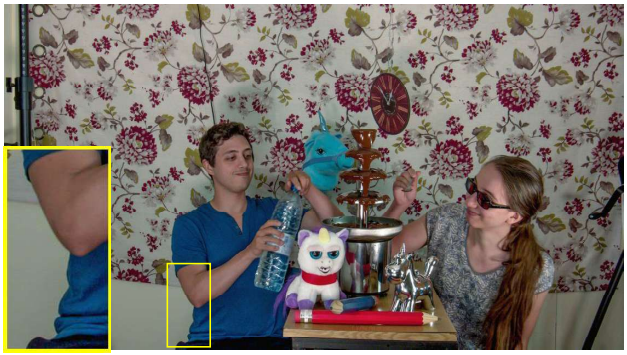
Compared with MVD sequences BA, BR and DA, sequences PA, CL and CH have more complex texture and accurate depth information. To this end, Table VIII further compares the performance of the proposed CFFHNet, RVS, Li et al.'s method and Yu et al.'s method on sequences PA, CL and CH. The RVS is a well-performing view synthesis algorithm at present, which replaces the VSRS into the latest MPEG-I standard. As described in the literature [59], the RVS method performs better on four inputs and eight inputs than the current mainstream view synthesis algorithms based on machine learning and non-machine learning. Therefore, we test the experimental results of view synthesis with four and eight reference views as input, which can effectively verify the effectiveness of the proposed CFFHNet. From Table VIII, we can easily see that the proposed CFFHNet performs better than the RVS method on the first type of general-purpose evaluation metrics (i.e., PSNR and SSIM). Especially on SSIM, this is because our sub-network RSN of CFFHNet focuses more on the inference of the overall structure of the synthesized image. On the second category of evaluation metrics for DIBR-synthesized images, the average MP-PSNR and MW-PSNR values obtained by our method are about 1.5 dB higher than that of the RVS method, but our method

Fig. 9. Subjective visual quality comparisons for BA4→1 and BA4→3. The first and second rows are the images contained in sequence BA4→1. The third and fourth rows are the images contained in sequence BA4→3. The fifth and sixth rows are the images contained in sequence BR4→3. The seventh and eighth rows are the images contained in sequences DA5→9 and DA1→5, respectively.

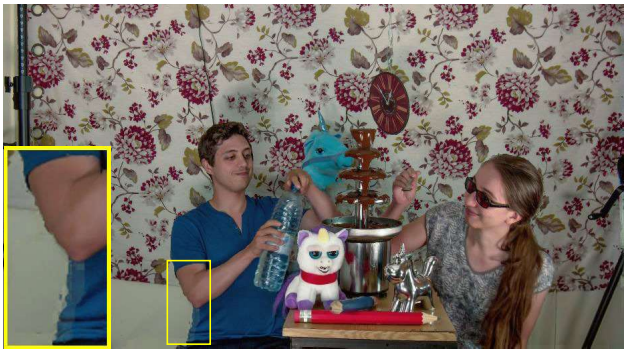
is slightly weaker than the RVS method on LOGS. In the third category of DIBR-synthesized video quality evaluation metric FDI, our proposed method is very close to the FDI values obtained by RVS, i.e., the performance of our proposed method is very close to the RVS algorithm in terms of temporal consistency. Li et al.'s method and Yu et al.'s method are inferior to RVS and the proposed CFFHNet in the six objective visual quality assessment criterion. In summary, this paper still achieves competitive performance on MVD sequences with complex textures. In addition, combined with Table VII and Table VIII, we find that the proposed method performs better in temporal consistency for sequences PA, CL and CH than for sequences BA, BR and DA. This result indicates that more accurate depth information is very beneficial to enhance the visual effect of viewpoint synthesis.

E. Subjective Visual Quality Comparison

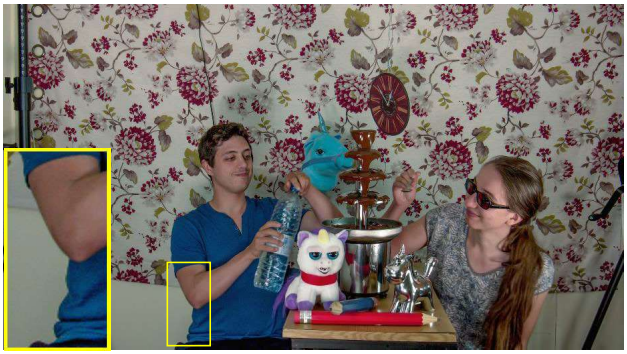
Fig. 9 shows the subjective visual quality comparisons for the MVD sequences with different distances between the virtual and reference viewpoints. The Criminsi et al.'s method [14], Daribo et al.'s method [15], Ahn et al.'s method [16], VSRS [58], Li et al.'s method [22], and Yu et al.'s method [26] all mistakenly fill the foreground texture into the hole regions, resulting in the "foreground penetration" artifacts. Subject comparison results show that our proposed method can restrain noise and preserve sharp edges more successfully than other competing DIBR methods. However, our method is not natural enough to deal with the boundaries between the hole and visible areas. In fact, besides Ahn et al.'s approach, other DIBR methods leave traces at the boundaries of the hole and visible regions. The Ahn et al.'s method introduces depth information to weaken the filling traces at the boundaries of the hole and visible areas. Since the performance of the depth estimation approach is not robust, the Ahn et al.'s method still produces the "foreground penetration" artifacts, such as the sequences BA4→1 and DA1→5 obtained by Ahn et al.'s algorithm in Fig. 9. Compared with BA4→1, the filling traces of the Zhu et al.'s method [19] and Luo et al.'s method [21] on BA4→3 are much weaker. Thus, when the distance between the virtual and reference viewpoints is small, utilizing multi-frame information can effectively reduce the filling traces. However, the Zhu et al.'s method and Luo et al.'s method need to adopt the background texture of the whole video sequence to fill the hole areas, and so these DIBR methods are difficult to apply in real-time application scenarios. In addition, due to the inaccurate segmentation of foreground and background, the hole filling algorithms based on temporal background information modeling still have a chance to fill the foreground texture into the background area, as shown in the sequence BR4→3 generated by Zhu et al.'s method. Fig. 10 shows the visual quality comparison of the proposed method against the ground-truth and the RVS software. The synthesized images and its ground-truth are derived from the synthesized sequence CH 00, 01, 02, 10, 12, 20, 21, 22→11 and CH 11, respectively. From Fig. 10, it can be observed that the proposed method and the RVS method perform very closely in the subjective visual quality, and both



(a) ground-truth



(b) result from RVS



(c) proposed method

Fig. 10. Visual comparison of our method against ground truth and the RVS Software. We use a yellow rectangular box to mark the area with significant differences and make a local enlargement.

are very close to the ground-truth. Fig. 11 shows the visual comparisons of the proposed and some SOTA methods on the translucent object filling. The competing methods employed are the deep learning-based image inpainting algorithms of Li et al. [22] and Yu et al. [26] and the Luo et al.'s algorithm [21], which performs similarly to the proposed algorithm. The red rectangular box areas are the regions where the filling quality of the competing algorithms is obviously inferior to that of the proposed method. Experimental results show that the proposed hole-filling method can also deal with complex samples such as transparent objects well.

F. Running Speed Comparison

For fairness, we compare the running speed of our CFFHNet with the deep learning-based image inpainting algorithms proposed by Li et al. [22] and Yu et al. [26]. The test environments

Fig. 11. Visual comparison of the proposed method, Li et al. 's method [22], Yu et al.' s method [26], and Luo et al. 's method [21] for filling transparent objects.

are AMD Ryzen Threadripper 2950X CPU and one NVIDIA GTX 2080Ti GPU. Li et al.'s method, Yu et al.'s method and the proposed CFFHNet require 0.90 seconds, 1.03 seconds and 0.24 seconds, respectively, to test an RGB image with the resolution of 1024×768. Besides the performance superiority, our CFFHNet also enjoys faster inference speed over the competing methods.

V. Conclusion

In this paper, we have proposed a coarse and fine-grained hierarchical network (CFFHNet) for hole filling in view synthesis. The proposed CFFHNet fills the irregular holes produced by view synthesis with the spatial contextual correlation in a visually plausible manner. The spatial contextual correlation is extracted by using recurrent calculation. The hierarchical structure and attention mechanism are introduced to lead the fine-grained fusion of the spatial contextual correlations at different spatial scales, which fuses low-level visual features and high-level semantic features to fill the irregular holes. We introduce a two-stage learning framework to ensure the affinity and fidelity between the hole and visible areas of the synthesized view. Extensive experiments verify the superiority of the proposed CFFHNet over the state-of-the-art hole filling algorithms for view synthesis. Our future work further utilizes depth information and adjacent frames to improve the visual quality of the synthesized view while removing filling traces. Moreover, due to the limitation of GPU's memory, existing works usually only train hole filling models with low-resolution images, which can easily lead to deep learning-based hole filling models that may introduce blur effects in local areas of synthesized images. It is also of great significance to carry out research to address this issue in the future work.

References

[1] O. Stankiewicz, M. Domanski, A. Dziembowski, A. Grzelka, D. Mieloch, and J. Samelak, "A free-viewpoint television system for horizontal virtual navigation," *IEEE Trans. Multimedia*, vol. 20, no. 8, pp. 2182–2195, Aug. 2018.

[2] S.-P. Lu, B. Ceulemans, A. Munteanu, and P. Schelkens, "Spatio-temporally consistent color and structure optimization for multiview video color correction," *IEEE Trans. Multimedia*, vol. 17, no. 5, pp. 577–590, May 2015.

[3] *Coded Representation of Immersive Media*, Standard ISO/IEC 23090, MPEG-I, 2018.

[4] J. Lei, C. Zhang, Y. Fang, Z. Gu, N. Ling, and C. Hou, "Depth sensation enhancement for multiple virtual view rendering," *IEEE Trans. Multimedia*, vol. 17, no. 4, pp. 457–469, Apr. 2015.

[5] H. R. Kaviani and S. Shirani, "An adaptive patch-based reconstruction scheme for view synthesis by disparity estimation using optical flow," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 7, pp. 1540–1552, Jul. 2018.

[6] Y. Qiao, L. Jiao, S. Yang, B. Hou, and J. Feng, "Color correction and depth-based hierarchical hole filling in free viewpoint generation," *IEEE Trans. Broadcast.*, vol. 65, no. 2, pp. 294–307, Jun. 2019.

[7] C. Fehn, "Depth-image-based rendering (DIBR), compression, and transmission for a new approach on 3D-TV," *Proc. SPIE*, vol. 5291, pp. 93–104, May 2004.

[8] W. J. Tam, G. Alain, L. Zhang, T. Martin, and R. Renaud, "Smoothing depth maps for improved steroscopic image quality," *Proc. SPIE*, vol. 5599, pp. 162–172, Oct. 2004.

[9] Y.-R. Horng, Y.-C. Tseng, and T.-S. Chang, "Stereoscopic images generation with directional Gaussian filter," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2010, pp. 2650–2653.

[10] P.-J. Lee and Effendi, "Nongeometric distortion smoothing approach for depth map preprocessing," *IEEE Trans. Multimedia*, vol. 13, no. 2, pp. 246–254, Apr. 2011.

[11] C. Zhu and S. Li, "Depth image based view synthesis: New insights and perspectives on hole generation and filling," *IEEE Trans. Broadcast.*, vol. 62, no. 1, pp. 82–93, Mar. 2016.

[12] S. Li, C. Zhu, and M.-T. Sun, "Hole filling with multiple reference views in DIBR view synthesis," *IEEE Trans. Multimedia*, vol. 20, no. 8, pp. 1948–1959, Aug. 2018.

[13] Y. Mao, G. Cheung, and Y. Ji, "On constructing Z-dimensional DIBR-synthesized images," *IEEE Trans. Multimedia*, vol. 18, no. 8, pp. 1453–1468, Aug. 2016.

[14] A. Criminisi, P. Perez, and K. Toyama, "Region filling and object removal by exemplar-based image inpainting," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1200–1212, Sep. 2004.

[15] I. Daribo and H. Saito, "A novel inpainting-based layered depth video for 3DTV," *IEEE Trans. Broadcast.*, vol. 57, no. 2, pp. 533–541, Jun. 2011.

[16] I. Ahn and C. Kim, "A novel depth-based virtual view synthesis method for free viewpoint video," *IEEE Trans. Broadcast.*, vol. 59, no. 4, pp. 614–626, Dec. 2013.

[17] P. Buyssens, O. L. Meur, M. Daisy, D. Tschumperlé, and O. Lézoray, "Depth-guided disocclusion inpainting of synthesized RGB-D images," *IEEE Trans. Image Process.*, vol. 26, no. 2, pp. 525–538, Feb. 2017.

[18] B. Ceulemans, S.-P. Lu, G. Lafruit, P. Schelkens, and A. Munteanu, "Efficient MRF-based disocclusion inpainting in multiview video," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2016, pp. 1–6.

[19] G. Luo and Y. Zhu, "Foreground removal approach for hole filling in 3D video and FVV synthesis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 27, no. 10, pp. 2118–2131, Oct. 2017.

[20] G. Luo, Y. Zhu, Z. Li, and L. Zhang, "A hole filling approach based on background reconstruction for view synthesis in 3D video," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1781–1789, doi: 10.1109/CVPR.2016.197.

[21] G. Luo, Y. Zhu, Z. Weng, and Z. Li, "A disocclusion inpainting framework for depth-based view synthesis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 6, pp. 1289–1302, Jun. 2020.

[22] J. Li, N. Wang, L. Zhang, B. Du, and D. Tao, "Recurrent feature reasoning for image inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 7757–7765.

[23] S. Xu, D. Liu, and Z. Xiong, "E2I: Generative inpainting from edge to image," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 4, pp. 1308–1322, Apr. 2021.

[24] A. Lahiri, A. K. Jain, S. Agrawal, P. Mitra, and P. K. Biswas, "Prior guided GAN based semantic inpainting," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 13693–13702.

[25] N. Wang, Y. Zhang, and L. Zhang, "Dynamic selection network for image inpainting," *IEEE Trans. Image Process.*, vol. 30, pp. 1784–1798, 2021.

[26] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. Huang, "Free-form image inpainting with gated convolution," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 4470–4479.

[27] Y. Zeng, Z. Lin, H. Lu, and V. M. Patel, "CR-fill: Generative image inpainting with auxiliary contextual reconstruction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 14144–14153.

[28] X. Shi, Z. Chen, H. Wang, D. Yeung, W. Wong, and W. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. 28th Int. Conf. Neural Inf. Process. Syst.*, 2015, pp. 802–810.

[29] K. Jiang, Z. Wang, P. Yi, G. Wang, T. Lu, and J. Jiang, "Edge-enhanced GAN for remote sensing image superresolution," *IEEE Trans. Geosci. Remote Sens.*, vol. 57, no. 8, pp. 5799–5812, Aug. 2019.

[30] P. Yi, Z. Wang, K. Jiang, J. Jiang, T. Lu, and J. Ma, "A progressive fusion generative adversarial network for realistic and consistent video super-resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 5, pp. 2264–2280, May 2022, doi: 10.1109/TPAMI.2020.3042298.

[31] C. Ledig et al., "Photo-realistic single image super-resolution using a generative adversarial network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 105–114.

[32] J. Johnson, A. Alahi, and L. Feifei, "Perceptual losses for real-time style transfer and super-resolution," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 694–711.

[33] M. Chu, Y. Xie, J. Mayer, L. Leal-Taixé, and N. Thuerey, "Learning temporal coherence via self-supervision for GAN-based video generation," *ACM Trans. Graph.*, vol. 39, no. 4, pp. 1–12, Aug. 2020.

[34] J. Cao, Y. Hu, B. Yu, R. He, and Z. Sun, "3D aided duet GANs for multi-view face image synthesis," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 8, pp. 2028–2042, Aug. 2019.

[35] J. Liu, D. Xu, W. Yang, M. Fan, and H. Huang, "Benchmarking low-light image enhancement and beyond," *Int. J. Comput. Vis.*, vol. 129, no. 4, pp. 1153–1184, Jan. 2021, doi: 10.1007/s11263-020-01418-8.

[36] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[37] A. Lucas, S. López-Tapia, R. Molina, and A. K. Katsaggelos, "Generative adversarial networks and perceptual losses for video super-resolution," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3312–3327, Jul. 2019.

[38] J. Hu, Y. Tang, and S. Fan, "Hyperspectral image super resolution based on multiscale feature fusion and aggregation network with 3D convolution," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 13, pp. 5180–5193, 2020.

[39] K. Gu, Y. Zhang, and J. Qiao, "Random forest ensemble for river turbidity measurement from space remote sensing data," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 11, pp. 9028–9036, Nov. 2020.

[40] M. Sun et al., "Can shape structure features improve model robustness under diverse adversarial settings?" in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7506–7515.

[41] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 2980–2988.

[42] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-8, no. 6, pp. 679–698, Nov. 1986.

[43] C. L. Zitnick, S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski, "High-quality video view interpolation using a layered representation," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 600–608, Aug. 2004.

[44] *Common Test Conditions of 3DV Core Experiments*, document JCT3V-F1100, ITU-R, Geneva, Switzerland, Oct. 2013.

[45] D. R. Dore, "MPEG-MIV-database—A [data set]," Zenodo, Tech. Rep., 2021, doi: 10.5281/zenodo.4651305.

[46] K. Bart, *3DoF+ Test Sequence Classroomvideo*, Standard ISO/IEC JTC1/SC29/WG11, MPEG2018/M42415, San Diego, CA, USA, Apr. 2018.

[47] K. Bart, *Full Depth Maps for ClassroomVideo*, Standard ISO/IEC JTC1/SC29/WG11 MPEG2018/M42944, Ljubljana, Slovenia, Jul. 2018.

[48] D. Bonatto, S. Fachada, M. Teratani, and G. Lafruit, "ULB ChocoFountainBxl [data set]," Zenodo, 2022, doi: 10.5281/zenodo.5960227.

[49] A. Schenkel, D. Bonatto, S. Fachada, H.-L. Guillaume, and G. Lafruit, "Natural scenes datasets for exploration in 6DOF navigation," in *Proc. Int. Conf. 3D Immersion (IC3D)*, Dec. 2018, pp. 1–8.

[50] A. Oliveira, G. Fickel, M. Walter, and C. Jung, "Selective hole-filling for depth-image based rendering," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Apr. 2015, pp. 1186–1190.

[51] A. Q. de Oliveira, T. L. T. de Silveira, M. Walter, and C. R. Jung, "A hierarchical superpixel-based approach for DIBR view synthesis," *IEEE Trans. Image Process.*, vol. 30, pp. 6408–6419, 2021.

[52] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[53] D. Sandic-Stankovic, D. Kukolj, and P. Le Callet, "DIBR synthesized image quality assessment based on morphological wavelets," in *Proc. 7th Int. Workshop Quality Multimedia Exper. (QoMEX)*, May 2015, pp. 1–6.

[54] D. Sandić-Stanković, D. Kukolj, and P. Le Callet, "DIBR-synthesized image quality assessment based on morphological multi-scale approach," *EURASIP J. Image Video Process.*, vol. 2017, no. 1, pp. 1–4, Dec. 2016.
[55] L. Li, Y. Zhou, K. Gu, W. Lin, and S. Wang, "Quality assessment of DIBR-synthesized images by measuring local geometric distortions and global sharpness," *IEEE Trans. Multimedia*, vol. 20, no. 4, pp. 914–926, Apr. 2018.
[56] Y. Zhou, L. Li, S. Wang, J. Wu, and Y. Zhang, "No-reference quality assessment of DIBR-synthesized videos by measuring temporal flickering," *J. Vis. Commun. Image Represent.*, vol. 55, pp. 30–39, Aug. 2018.
[57] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Represent.*, May 2015.
[58] M. Tanimoto, T. Fujii, K. Suzuki, N. Fukushima, and Y. Mori, *Reference Softwares for Depth Estimation and View Synthesis*, Standard M15377, Archamps, France, Apr. 2008.
[59] D. Bonatto, S. Fachada, S. Rogge, A. Munteanu, and G. Lafruit, "Real-time depth video-based rendering for 6-DoF HMD navigation and light field displays," *IEEE Access*, vol. 9, pp. 146868–146887, 2021.
[60] M. Solh and G. AlRegib, "Hierarchical hole-filling for depth-based view synthesis in FTV and 3D video," *IEEE J. Sel. Topics Signal Process.*, vol. 6, no. 5, pp. 495–504, Sep. 2012.

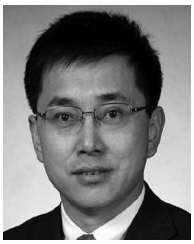
Guangcheng Wang received the Ph.D. degree from the School of Computer Science, Wuhan University, Wuhan, China, in 2022. He is currently an Associate Professor with the School of Transportation and Civil Engineering, Nantong University, Nantong, China. His research interests include image processing and machine learning.

Kui Jiang (Member, IEEE) received the Ph.D. degree from the School of Computer Science, Wuhan University, Wuhan, China, in 2022. He is currently an Associate Professor with the School of Computer Science and Technology, Harbin Institute of Technology, Harbin, China. His research interests include image/video processing and computer vision. He received the 2022 ACM Wuhan Doctoral Dissertation Award.

Ke Gu (Senior Member, IEEE) received the B.S. and Ph.D. degrees from Shanghai Jiao Tong University, Shanghai, China, in 2009 and 2015, respectively. He is currently a Professor with the Beijing University of Technology, Beijing, China. His research interests include industrial vision, environmental perception, image processing, and machine learning.

Hongyan Liu received the B.S. degree from the Beijing University of Technology, Beijing, China, in 2021, where she is currently pursuing the Ph.D. degree. Her research interests include industrial vision, environmental perception, image processing, and machine learning.

Hantao Liu (Senior Member, IEEE) received the Ph.D. degree from the Delft University of Technology, Delft, The Netherlands, in 2011. He is currently an Assistant Professor with the School of Computer Science and Informatics, Cardiff University, Cardiff, U.K. He is serving for the IEEE MMTC, as the Chair of the Interest Group on Quality of Experience for Multimedia Communications. He is an Associate Editor of IEEE TRANSACTIONS ON HUMAN MACHINE SYSTEMS and IEEE TRANSACTIONS ON MULTIMEDIA.



Wenjun Zhang (Fellow, IEEE) received the B.S., M.S., and Ph.D. degrees in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 1984, 1987, and 1989, respectively. From 1990 to 1993, he was a Postdoctoral Fellow with Philips Kommunikation Industrie AG, Nuremberg, Germany, where he was actively involved in developing HDMAC system. He joined the Faculty of Shanghai Jiao Tong University in 1993 and became a Full Professor with the Department of Electronic Engineering in 1995. As the National HDTV TEEG Project Leader, he successfully developed the first Chinese HDTV prototype system in 1998. He was one of the main contributors to the Chinese Digital Television Terrestrial Broadcasting Standard issued in 2006 and is leading team in designing the next generation of broadcast television system in China since 2011. He holds more than 40 patents and published more than 90 papers in international journals and conferences. His main research interests include digital video coding and transmission, multimedia semantic processing, and intelligent video surveillance. He is a Chief Scientist of the Chinese National Engineering Research Centre of Digital Television (NERC-DTV), an industry/government consortium in DTV technology research and standardization and the Chair of the Future of Broadcast Television Initiative (FOBTV) Technical Committee.