Learning-Based Stereoscopic View Synthesis with Cascaded Deep Neural Networks

Wei Liu*,†, Liyan Ma**, and Mingyue Cui*

*College of Electromechanic Engineering, Nanyang Normal University
No.1638 Wolong Road, Wolong District, Nanyang, Henan 473061, China
E-mail: lw3171796@163.com, cuiminyue@sina.com

**College of Computer Engineering and Science, Shanghai University
No.99 Shangda Road, Baoshan District, Shanghai 200444, China
E-mail: liyanna@shu.edu.cn

†Corresponding author

[Received October 25, 2021; accepted March 8, 2022]

Depth image-based rendering (DIBR) is an important technique in the 2D to 3D conversion process, which renders virtual views with a texture image and the associated depth map. However, certain problems, such as disocclusion, still exist in current DIBR systems. In this study, a new learning-based framework that models conventional DIBR synthesis pipelines is proposed to solve these problems. The proposed model adopts a coarse-to-fine approach to realize virtual view prediction and disocclusion region refinement sequentially in a unified deep learning framework that includes two cascaded joint filter block-based convolutional neural networks (CNNs) and one residual learning-based generative adversarial network (GAN). An edge-guided global looping optimization strategy is adopted to progressively reconstruct the scene structures on the novel view, and a novel directional discounted reconstruction loss is proposed for better training. In this way, our framework performs well in terms of virtual view quality and is more suitable for 2D to 3D conversion applications. The experimental results demonstrate that the proposed method can generate visually satisfactory results.

Keywords: DIBR, deep neural networks, hole filling, view synthesis

1. Introduction

Nowadays, three-dimensional (3D) videos are becoming more and more popular than the conventional two-dimensional (2D) ones due to their richer representation of real-world scenes. However, creating content directly in a suitable 3D format is still difficult, and the cost of making 3D videos remains high. For these reasons, 2D to 3D conversion technology, which can add 3D effects to existing media data in 2D format, is an effective way to enable exciting applications.

Depth image based rendering (DIBR) technology is one of the most important rendering techniques in 2D to 3D conversion, which can synthesize virtual views from different viewpoints using a 3D warping process [1]. The image plus depth data format, which is used by DIBR, has certain significant advantages over the conventional broadcasting system. When a depth map is available, DIBR systems can generate several views instead of multi-camera systems; thus, the cost of 3D media creation can be reduced. Additionally, the DIBR system allows users to customize the parallax of the generated virtual view images to achieve different depth effects and experience different 3D perceptions. Based on these advantages, it was regarded as another promising solution by the European Information Society Technologies (IST) project, advanced three-dimensional television system technologies (ATTEST) [2]. However, owing to the sharp horizontal changes among the different depth layers, the warping process may reveal areas that are occluded in the original view and become visible in the new virtual views. To address this problem and achieve high-quality 3D effects, these holes should be filled.

In general, a classical DIBR system that can cope with disocclusions always follows pipelines, as shown in Fig. 1. It always contains multiple parts, including the preprocessing of the depth map, image warping, and hole filling. In the preprocessing step, the depth map is usually smoothed to reduce hole occurrences. Then, in the hole-filling step, the newly exposed areas known as holes (or disocclusions) are filled after 3D image warping. According to the different steps these approaches emphasize when realizing hole filling in the DIBR process of 2D to 3D conversion, we categorize these methods into two types. The first one adopts a strategy to deal with the problem in the pre-processing procedure, which reduces depth map edge discontinuities by filtering the depth maps. In this manner, holes are diminished in the first case rather than being filled later. Early methods used a Gaussian filter to smoothen depth images [3]. The main drawback of this method is that additional geometric distortions can be introduced. To solve this problem, several methods have been proposed using asymmetric smoothing [4, 5], scene
structure and content-related adaptive filters [6, 7]. However, geometric distortions in some severe sharp-depth discontinuities in the filtered areas are still inevitable. The second type adopts a strategy to deal with the problem in the post-processing procedure, which focuses on virtual image inpainting after DIBR. These methods can achieve hole filling with auxiliary information around disocclusion regions by either texture replication or structure continuation after DIBR [8–10]. Because an in-depth discussion of inpainting is beyond the scope of this paper, interested readers are referred to a summary of these methods [11]. However, inpainting is a challenging problem, and it is more difficult with stereoscopic content, as image features need to have consistent disparity across the two generated views. In addition, in our previous work [12], methods that combine both procedures in one framework were studied to realize better hole-filling effects.

Inspired by the success of deep learning in a variety of applications, such as image denoising [13], super-resolution [14], and deblurring [15], several methods have been developed to predict depth from a single image using deep learning [16–19], which are strongly related to the problems of 2D-to-3D conversion. In this study, we focus on using deep learning to model stereoscopic synthesis, where the goal is to create a stereoscopic view from a reference view with an image plus depth data format.

However, research in this field has been limited. Some studies have adopted strategies that combine single image depth estimation with the DIBR process in one convolutional neural network (CNN) framework. For example, Deep3D [20] and previous work [21] proposed a probabilistic selection layer to model the rendering process in a differentiable manner so that it could be trained together with a depth map prediction network. Another study [22] used a spatial transformer module [23] to bridge the relationship between different views. In these methods, depth estimation and virtual view warping were trained together in a united end-to-end mapping, and the original image view was provided as the only input. Therefore, stereoscopic synthesis can be realized in a relatively simplified manner. Unlike these approaches, our method handles the image plus depth data formats, where the depth maps given to the system may be captured by active approaches with range devices or generated by a 2D to 3D converter from different sources [24]. Therefore, the accuracy of the given depth maps can significantly affect the quality of the synthesized view. In a classical DIBR system, as shown in Fig. 1, a depth preprocessing step is specially designed to solve this problem. Inspired by this, in this study, we attempted to model stereoscopic synthesis by following the pipelines of a typical DIBR process and explicitly adopting a cascaded sub-network to optimize the initial depth maps.

In current learning-based studies, disocclusion hole filling has always been regarded as a generative image inpainting challenge, and learning-based inpainting techniques [25, 26] are usually used to restore the occlusion regions in warped views. In our option, for these methods, not the entire process but only post-processing in the classical pipelines is modeled. In addition, these methods may also ignore certain important implicit priors in the DIBR process, such as those that predicted textures in the missing occlusion regions, which are more similar to those of the scene background. Therefore, they cannot properly handle complicated scenes in stereoscopic synthesis applications.

In this study, to make the training more tractable, a new learning-based framework that models the integral classical view synthesis pipelines shown in Fig. 1 is proposed. This framework is composed of two parts: virtual view prediction and disocclusion region refinement. It is realized by two cascaded joint filter block-based convolution neural networks (CNNs) and one residual learning-based generative adversarial network (GAN). In this way, our proposed methods are more suitable for 2D to 3D conversion application than existing similar deep learning-based techniques. In summary, we make the following contributions.

- A novel learning-based framework that models conventional DIBR pipelines is proposed. Both depth map pre-processing and hole-filling refinement are combined in a unified framework using different cascaded network blocks. Through this coarse-to-fine approach, the proposed framework is not only more efficient but also more optimal than other similar networks [25, 26], where only one process in the DIBR pipelines is modeled.
- For depth map pre-processing, our scheme first performs scene reconstruction of the occlusion regions with an edge-guided global looping (EGGL) optimization form, and then the restored structure information is used as priors to guide novel view generation by the joint filter block-based CNNs. For occlusion filling refinement, a novel directional discounted reconstruction loss (DDRL) is proposed to provide additional constraints for the GAN. In this
way, our framework performs well in terms of virtual view quality and is more suitable for 2D to 3D conversion applications.

The remainder of this paper is organized as follows. In Section 2, the technical scheme of the approach and architecture of the cascaded framework are introduced. Then, each part of the architecture, such as joint filter block-based CNNs and residual learning-based GAN, is discussed separately and elaborately. Section 3 presents the experimental results and discussion. Some concluding remarks are presented in Section 4. Experiments and comparisons demonstrated the effectiveness of the proposed methods over similar approaches, in both qualitatively and quantitatively, on the Middleburry and Make3D datasets. It should be noted that because of the regularity of stereo camera positions in a structured parallel configuration of 3D warping, depth and disparity are closely related, and we use them interchangeably in this paper. In the remainder of this paper, the depth map is represented as an 8-bit grayscale image. The continuous depth range was quantized into 255 discrete depth values. The nearest object to the camera image sensor is assigned a value of 255, and the farthest object is assigned a value of 0.

2. Proposed Methods

Figure 2 shows the framework of the proposed learning-based stereoscopic synthesis approach. We can observe that our network architecture mainly incorporates two component parts, which correspond to the pre-processing and post-processing pipelines in a classical DIBR system.

The scheme of the first part bears some resemblance to that of the previous work [27], in which convolutional neural networks were used to predict novel views with sparse input views and the positions of the novel views in the light field. By contrast, our rendering system cannot obtain this information in advance. Instead, two cascaded joint filter blocks [28] are adopted to extract the novel view structures using scene guidance priors \( \{ G_l, G_r \} \) from both the original view and the reconstructed 3D warped view. The core of the second part is an improved GAN based on a previous study [29], where \( \{ G, D \} \) are the generator and discriminator in the GAN block of this part, and a novel loss function DDRL is proposed to enhance occlusion-filling refinements in training. Next, we describe each part of the network and training procedure in detail.

2.1. Joint Filter Block-Based Cnn’s for Novel View Prediction

As shown in the first stage of Fig. 2, two sequential CNN blocks were used to estimate the novel view. General 2D to 3D conversion cannot rely on multiple camera depth estimations or a priori depth maps, and depth must be estimated from a single view. Therefore, the edge information of the initial depth map may not match that of the original view image. Thus, incorrect depth values, especially around the edges, may greatly affect the quality of the synthesized view. To ensure the accuracy of the depth edges as much as possible, our first CNN block works as a pre-processing step to align depth discontinuities in the depth map to color discontinuities in the textured image. Then, our second CNN block adopts a multi-scale backward warping module to further realize novel view synthesis with the predicted depth map in the novel view instead of direct texture image mapping. In contrast to the texture information of the same scene, the depth image mainly reflects sharp object edges and smooth regions inside objects instead of complicated texture details. Thus, the disocclusion problem was easier to solve.

These two CNN blocks share a common network architecture. As shown in Fig. 3, it contains two basic parts: feature extraction and feature fusion, and a specially designed multiscale backward warping module for the second CNN block. The two branches CNNF and CNNR first act as feature extractors to determine informative features from both the target and guidance images, respectively. These features are then concatenated as inputs of CNNF in the feature fusion part to transfer common structures and reconstruct the filtered output. ResBlock represents basic residual unit. For each residual layer, two sequential convolutions with batch normalization were contracted with
one identity residual connection, followed by a ReLu activation function.

This architecture is an improved deep-joint image filter. Some improvements were made compared with the original method [28]. First, pooling and deconvolution were introduced in the feature extraction and feature fusion parts in pairs. Each of the pipelines (\(CNN_F - CNN_F'\) and \(CNN_G - CNN_F\)) has a fully convolutional encoder-decoder architecture, which is beneficial for feature fusion of different views with large displacements. Second, skip connections are added between \(CNN_G\) and \(CNN_F\). The features shuttled by skip connections carry many details of the guided branch, which helps to maintain the estimation accuracy and reduce the side effects of pooling. Third, features are concatenated not only in the feature fusion part but also between \(CNN_G\) and \(CNN_F\) after each layer in the feature extraction part. The reason behind this is that the inputs of \(CNN_G\) in this network can be regarded as pre-processed initialized features that contain edge information. Thus, fusion manipulation can be performed in advance for high-performance-boosting training.

In Fig. 3, if we denote \(I\) as the input for each branch in the feature extraction, then after each layer that contains a pooling and several stacked residual units, the resolution of the feature maps is halved, and the number of feature maps in each residual unit is doubled. In this study, for \(LayerN\), they are \(1/2^{N-1}\) and \(16 \times 2^N\), respectively. In \(CNN_F\), deconvolution operations are performed for each fusion layer, so the final output has the same resolution as the input. For the first CNN block, the number of layers is set to three. Considering that the second CNN block performs joint filtering between views of large disparity, it is set to four for a deeper network architecture.

We trained our network with datasets \(\{I_l, D_l, I_r, D_r\}\), where \(I_l, D_l\) denotes the texture image and denotes the corresponding depth map in the original view. For simplicity, we set this as the left view. Similarly, \(I_r, D_r\) are shown in the right view. In this work, the task is to estimate the best right view \(I'_r\) by relating a left view \(I_l\) with its given left view \(D_l\). Therefore, \(I_l\) is the ground truth, and is denoted as \(I_{gt}\). \(D_r\) was used as auxiliary information for network training.

The first CNN block works with the following process to realize depth map correction.

\[
\begin{align*}
D'_l &= F_{CNN1}(\{D_l, G_1\}, \theta_{CNN1}) \\
G_1 &= \xi(I_l)
\end{align*}
\]  

where \(F_{CNN1}\) represents the end-to-end mapping function of the first CNN block with estimated parameters \(\theta_{CNN1}\). \(\xi(\cdot)\) is a gradient operator, which is used to extract the edge map of the original view. Previous study [28] has shown that edge information is more effective in providing direct and important guidance without other redundant mixtures that require additional network computation. After this step, misalignments between the texture image and depth map are weakened, and the apparent flaws caused by this reason can also be reduced.

The second CNN block had a similar network architecture, but with different input configurations. It takes \(D'_l\) as the targeted input and uses the warped intermediate results in a novel view to guide the filter as:

\[
\begin{align*}
D'_r &= F_{CNN2}(\{D'_l, G_r\}, \theta_{CNN2}) \\
G_r &= \xi(I_{gt})
\end{align*}
\]  

where \(F_{CNN2}\) represents the mapping function of the proposed second CNN block with the estimated parameters \(\theta_{CNN2}\). \(G_r\) denotes the warped prior in the novel view, which is set to \(G_r = \xi(I_{gt})\) for training. In testing, the ground truth \(I_{gt}\) cannot be provided; therefore, we follow another strategy to extract the guidance priors. This is discussed in detail in Section 2.3.

We performed backward warping by sampling the input image based on the predicted depth map in the novel view, as shown in Fig. 4. This module fuses the information of multiple shortcut paths, which can be considered as performing a summation fusion of multiple residual connections with the necessary dimensions or resolution adaptation. The multiscale architecture is beneficial for the network to enhance details. Thus, the synthesized
right-view can be expressed as:

\[ I'_r = F_{BW} \left( \{ p_{scale}(I), p_{scale}(D'_r) \}, \theta_{BW} \right) \]  \quad (3)

where \( F_{BW} \) represents the mapping function of the proposed multiscale backward warping module with the estimated parameters \( \theta_{BW} \). \( p_{scale}(\cdot) \) represents multiscale schematic process for each input. Finally, we obtain the predicted right view \( I'_r \).

2.2. Residual Learning Based GAN for Occlusion Filling Refinement

The classical pipeline of DIBR in Fig. 1 deals with disocclusion using both depth map filtering and occlusion filling as pre-processing and post-processing steps, respectively. According to this process, the second stage of our proposed framework is constructed to further refine the predicted novel view \( I'_r \). As shown in Fig. 5, part of this network is based on recent research on learning-based inpainting [29] and is built on an adversarial model, which consists of a generator/discriminator pair. The generator has an encoder that downsamples the image twice, followed by eight residual layers, and a decoder that upsamples the image back to the original resolution. Dilated convolutions were used for the residual layers. For the discriminator, we used the PatchGAN architecture, which determines whether overlapping image patches are real. All convolutional layers employ a stride of 2 × 2 pixels to decrease the image resolution while increasing the number of output filters. The two losses proposed in [26], commonly known as perceptual loss \( L_{perc} \) and style loss \( L_{style} \) are included in our loss functions.

Certain improvements are made in this section to make this adversarial model more suitable for stereoscopic synthesis. First, the generator of the GAN has a residual learning architecture that includes short and long-range residual connections in the model. Short-range residual connections refer to local shortcut connections in each residual layer, whereas long-range residual connections refer to the connections between the module input and output, as shown in Fig. 5. This architecture has two main advantages. First, the model of this part is just one sub-network in our proposed framework; therefore, with the long-range residual connection, gradients can be directly propagated to earlier layers to speed up training for all network components. Second, for the newly exposed hole regions, the initial priors from the predicted novel...
view \( I'_r \) are provided by our cascaded CNN blocks. This is different from the original works on generative image inpainting, where no additional information is provided for the missing regions. Therefore, our network can be trained more effectively by the long-range residual connection and avoid the hallucination of pixels at the same time.

Second, a directional discounted reconstruction loss (DDRL) is proposed for the GAN training. As shown in Fig. 6, the region of the newly exposed hole is denoted by \( \Omega \), the region of the foreground is \( \phi_F \), the region of the background is \( \phi_B \), the contour between \( \Omega \) and \( \phi_F \) is \( \delta \Omega_F \), and the contour between \( \Omega \) and \( \phi_B \) is \( \delta \Omega_B \). Intuitively, the missing pixels near the hole boundaries have much less ambiguity than those closer to the center of the hole. In particular, in this study, the directional heuristic, where a newly exposed hole is usually located in the back-hole. In this way, more hallucinated cues from the backgrounds instead of the foreground would be learned during training.

For each horizontal line across the hole shown in Fig. 6, the weight \( W_{p_i} \) in \( \delta \Omega_B \) is larger than \( W_{p_0} \) in \( \delta \Omega_F \). Then, the weight at pixel \( p \) of \( M_d \) around the hole regions can be expressed as:

\[
W_p = \begin{cases} 
W_{p_0} + \frac{(W_{p_1} - W_{p_0}) \| p - p_0 \|}{\| p_1 - p_0 \|}, & p \in (\delta \Omega_F, \delta \Omega_B) \\
W_{p_0}, & p \in \delta \Omega_F \\
W_{p_1}, & p \in \delta \Omega_B 
\end{cases}
\] ........................ (4)

Let \( G \) and \( D \) be the generator and discriminator of GAN, respectively. Then, it uses the predicted novel-view \( I'_r \) as input, conditioned with a mask \( M_h \), which is constructed as a binary image with one for predicted hole region and zero elsewhere. It is trained over a joint loss consisting of \( L_I \) loss, adversarial loss, perceptual loss, and style loss. Our DDRL is integrated into the \( L_I \) loss as:

\[
L_{ddr} (I'_r, M_h) = \| M_h \odot M_d \odot (G(I'_r, M_h) - I_{gt}) \| \] ........................ (5)

where \( \odot \) is the pixel-wise multiplication and \( \| \| \) denotes the Euclidean norm. \( I_{gt} \) denotes the corresponding ground truth of \( I'_r \), and \( M_{ij} \) is the directional discounted reconstruction mask, where the weight of each pixel is determined by Eq. (4).

The adversarial loss \( L_{adv} \) is defined as:

\[
L_{adv} (I'_r, I_{gt}, M_h) = \max_D \mathbb{E} [\log D(I_{gt}, M_h)] \\
+ \log (1 - D(G(I'_r, M_h), M_h)) \] ........................ (6)

The perceptual loss \( L_{perc} \) and style loss \( L_{style} \) are formulated as follows:

\[
L_{perc} (I'_r, I_{gt}) = \mathbb{E} \left[ \sum_i \mathbb{E} \left[ \left\| \phi(I'_r) - \phi(I_{gt}) \right\| \right] \right] \] ........................ (7)

\[
L_{style} (I'_r, I_{gt}) = \mathbb{E} \left[ \left\| \mathbb{E}_j \left[ \left\| \mathbb{E}_i \left[ \left\| G_i^p (I'_r) - G_i^p (I_{gt}) \right\| \right] \right] \right] \right] \] ........................ (8)

where \( N_i \) is the number of elements in the \( i \)-th activation layer and \( \phi \) is the activation map of \( i \)-th-layer of a pre-trained network. In our study, \( \phi \) corresponds to the activation maps from the layers of the VGG-19 network pre-trained on the ImageNet dataset, which are also used to compute style loss. \( Gr^p_{ij} \) is the Gram matrix constructed from the activation map \( \phi \). Therefore, the overall loss function is:

\[
L_G = \alpha_{ddr}L_{ddr} + \alpha_{adv}L_{adv} + \alpha_{perc}L_{perc} + \alpha_{style}L_{style} \] ........................ (9)

where \( \alpha_{ddr} \), \( \alpha_{adv} \), \( \alpha_{perc} \), \( \alpha_{style} \) are the loss term weights. For our experiments, we selected \( \alpha_{ddr} = 1 \), \( \alpha_{adv} = \alpha_{perc} = 0.1 \), \( \alpha_{style} = 250 \).

2.3. Training Strategy and Global Looping Optimization

All layers of our proposed framework are differentiable, and thus end-to-end training with a single loss at the end, comparing the synthesized novel view and ground truth view, is feasible. However, for more efficient training, a step-by-step strategy was employed in this study. It has been proven to be an effective method for network training in various applications, usually with coarse-to-fine or complicated GAN-based network architectures [29–31], similar to ours.

The overview framework of our network, shown in Fig. 2, consists of two stages. Correspondingly, the training procedure was divided into two phases. First, the joint filter block-based CNNs were explicitly trained with the reconstruction losses. Subsequently, the refinement network is trained with reconstruction and GAN losses.

In the first stage, two reconstruction losses were harnessed. The first is formulated by directly minimizing the mean squared error (MSE) between the synthesized \( I'_r \) and ground-truth images \( I_{gt} \). The second is enhanced by an additional auxiliary loss, which is based on the MSE between the synthesized \( D'_r \) and ground-truth depth maps \( D_r \). These are formulated as follows:

\[
L_r = E(|| I'_r - I_{gt} ||) \] ........................ (10)

\[
L_{rec} = \lambda_rE(|| I'_r - I_{gt} ||) + \lambda_dE(|| D'_r - D_r ||) \] ........................ (11)
where $\lambda_c$ and $\lambda_d$ are the weights of the different terms in the loss function. This part of our framework aims to optimize CNNs to estimate good virtual views, rather than disparity maps. Therefore, the master branch loss in Eq. (10) takes the most responsibility, whereas the auxiliary term in Eq. (11) only helps optimize the learning process. During training, an auxiliary intermediate loss can be inserted to guarantee that the learned parameters carry their corresponding physical meanings as well. Based on these considerations, we further divided the training procedures of this stage into two sub-phases: the MSE loss from Eq. (11) was adopted for the first half of the total iterations $T_{S_1}$ in this stage, and the loss from Eq. (10) was used for the rest of the training.

In the second stage, the joint filter block-based CNNs were fixed, and the refinement net was trained with the loss defined in Eq. (9) for $T_{S_2}$ iterations. Finally, both parts were trained jointly for $T_{S_3}$ iterations until the end of training. The pretraining of cascaded CNNs and discriminator networks has proven critical for successful training. In contrast to the training, the guidance prior $G_r$ in Eq. (2) cannot be provided directly during testing. Instead, we performed an edge-guided global looping (EGGL) optimization process with the features warped from the original view in the testing step. Then, the definition of $G_r$ can be updated as:

\[
G_r = \begin{cases} 
\xi(I_i) & \text{In training} \\
EG\left(\psi_{\text{warp}}(I_i, D_i')\right) & \text{In testing, } k = 1 \\
\xi\left(I_i^{(l-1)'}\right) & \text{In testing, } k \geq 2 
\end{cases}
\]

where $\xi(.)$ still denotes a gradient operator, $\psi_{\text{warp}}(i, j)$ denotes the forward 3D warping operation that generates the novel view of image $i$ with its corresponding depth map $j$. We adopted Eq. (13) to obtain two types of warped edges, which are regarded as joint edge features, where $w_i$ and $w_d$ are the two weight coefficients. We selected for our experiments $w_i = w_d = 0.5$. $EG$ denotes the edge-reconstruction process realized by an additional edge generator [30].

New holes may appear after the warping operation, and it is easier to realize edge map hole filling than image completion in this way. First, it should be noted that unlike 3D warping for an entire image, the influence of warping on the gradient scene structures is limited. For gradient scene structures, the main available information is the discretely distributed scene layer boundary lines rather than large homogenous regions; therefore, the disc-occlusion and occlusion phenomenon caused by adjacent region layers of different depth values can be reduced naturally. Second, in Eq. (12), an edge generator $EG$ is deployed to realize the initial edge recovery. In the experiments, we found that the pre-trained parameters in the previous study [30] for this module can achieve satisfactory results for our edge map reconstruction process; hence, it remains fixed during training. Third, the looping process of this part can help to update and improve the joint edge feature maps $\psi_{\text{warp}}$ progressively. The number of looping operations $n$ is determined by the virtual view quality. With an increase in $n$, the quality of the synthesized view $I_i^n$ gradually improved. When $n$ reaches a certain value, the quality of the composite viewpoint tends to stabilize. In the subsequent experiments, we set $n = 3$.

### 3. Experiments

#### 3.1. Experimental Setup

In our experiments, we collected 20 datasets from the Middlebury stereo 2014 database as training data for the network model. For each set, depth maps and texture images of two different views $S_T = \{I_i, D_i, I_j, D_j\}$ are provided, where $S_T$ denotes the newly resized set with scaling $T$ from the original one. A total of 27280 sub-images (256 x 256) were generated from the 20 sets of three different scaling sizes $S_{0.8}$, $S_{1}$, $S_{1.2}$. In this manner, our training sets include scenes of multiple disparity sets, which can be used to better evaluate the guidance performance of the joint filter-based CNN blocks in our framework. Data augmentation was performed on the fly by applying a random transformation to the training data. This includes translation, flipping, and brightness shifts. In addition, degradation operations such as Gaussian filtering, erosion, and dilating are randomly performed on $D_i$, which are beneficial for the network to deal with image sets with imperfect initial depth maps. We implemented our proposed scheme using the TensorFlow platform, combining the implementation with an edge generator module running in PyTorch on a standard desktop with a 32 GB NVIDIA Quadro GPU with a batch size of 16. The model was optimized using the Adam optimizer with $\beta_1 = 0.0005$ and $\beta_2 = 0.9$.

Different datasets were adopted to test the proposed framework. One part is the remaining datasets in the Middlebury stereo 2014 database, which are not contained in the training data. The other part is from the Make3D database, where the original depth maps are from laser scanners of low resolution, and we obtained the aligned maps using other depth pre-processing methods. In our experiments, to better record and lay out, they are denoted as two groups and listed in Table 1, where the datasets in Test set A are from Middlebury, and those in Test set B are from Make3D. In the following, we first qualitatively and quantitatively compare our approach with some other

| Table 1. Test sets used in the experiments. |
| Test set A | Test set B |
| Adirondack | Street |
| Sticks | Building |
| Umbrella | |

Vol. 26 No. 3, 2022 Journal of Advanced Computational Intelligence and Intelligent Informatics

Stereoscopic View Synthesis with Deep Neural Networks
similar works and then further analyze the impact of each component of our model individually.

3.2. Qualitative Comparisons

In this section, the synthesized virtual view images on the test sets from both Middleberry and Make3D by the proposed methods and certain other view synthesis methods are compared, which include the following:

- Conventional methods. Here, an asymmetric filter (AF) [32] and patchmatch (PM) [33] are used as the depth preprocessing and image post-inpainting modules, respectively, in a classical DIBR system to deal with the disocclusion holes.

- Recent deep-learning-based image-inpainting methods. In our experiments, the edge connect (EC) [30] and gated convolutions (GC) [34], which treat the hole-filling problem as generative image inpainting with the processing scheme described in [26], were compared with our method.

The experimental results are shown in Figs. 7 and 8. Based on these results, several observations were made. First, depth-smoothing methods are generally more inclined to cause geometric distortion as shown in areas marked with dotted rectangles in Figs. 7(a) and 8(a). Although an asymmetric Gaussian filter may reduce this phenomenon, additional side effects may be introduced, such as the area in the dotted rectangle in Fig. 7(a), where one arm of the chair is obviously over-smoothed in the vertical direction. Second, image inpainting can preserve scene structures better than depth-smoothing methods, but in regions with less texture information, it may cause obvious irregular artifacts, as shown in the rectangular areas in Fig. 7(b) or ghost shadows in Fig. 8(b).

Deep-learning-based inpainting methods have some advantages over traditional inpainting methods. The results from GC in Figs. 7(d) and 8(d) were better than those from EC in Figs. 7(c) and 8(c), which may be due to the relatively longer processing scheme of the coarse-to-fine network architecture in GC. Even so, for these methods, not the entire scheme, only one part of the pipeline is followed, which limits further improvements in the experiments. Third, the proposed method, which has the most complete processing pipelines and special scene constraints for stereoscopic synthesis in the training phase with DDRL, naturally synthesizes the virtual views. As shown in Figs. 7(e) and 8(e), the proposed method provides a reasonable quality of virtual views. Especially for the distant background details marked with dotted rectangles in Fig. 8(e), the visual quality with our method is much better than that of any other algorithm. In summary, the proposed scheme is more suitable for high-quality stereoscopic synthesis in the application of 2D to 3D conversions.

3.3. Quantitative Comparisons

Test set A has the ground truth of newly generated novel views; therefore, the synthesized virtual view images can be further evaluated by PSNR and SSIM comparisons [35] with the ground truth. Table 2 gives the PSNR and SSIM comparison results, and the best results are highlighted in boldface. From this table, we can observe that the depth smoothing methods always obtain the lowest values owing to geometric distortions by global filtering in the whole depth map. Inpainting-related methods only focus on hole regions; therefore, they usually obtain higher scores. However, as in the aforementioned experiments, small obvious texture artifacts or ghost shadows that appeared on the inpainted image lowered their scores. The method proposed in this study has the best PSNR and SSIM performances.

Subjective viewing tests were performed on 15 individuals with normal or correct-to-normal visual acuity. The participants watched the synthesized images in random order and were asked to provide a satisfaction score. The scores range from 0 to 5, with higher scores indicating higher image quality. The results of the subjective quality evaluations are presented in Table 3. Generally, these scores were consistent with our previous analysis. Synthesized images with serious distortions due to depth over-smoothing or obvious irregular artifacts due to less-constrained inpainting usually scored poorly. We noticed that the test sets in Test set B, where the disocclusion holes always appear in the background, had relatively higher scores. These regions also had lower visual saliency. This indicates that visual saliency is an important cue for stereoscopic synthesis, and in our work, it will be studied further for subsequent improvements in our proposed scheme.

3.4. Analysis of Experimental Details

In this section, ablation studies are conducted to see how each of our proposed components contributed to the final performance, and the processing details of our proposed network are displayed in different experiments.

3.4.1. Intermediate Outputs of the Cascaded Joint Filter Block-Based CNNs

To deal with the disocclusion problem in traditional DIBR pipelines, our scheme presents a stepwise refinement strategy with a cascaded joint filter block-based CNN architecture, where the first CNN block preprocesses the initial depth maps and the second one realizes 3D warping in an edge-first and global looping optimization method. Two experiments were conducted to show the results in each step separately.

As discussed in Section 2.1, our first CNN block can refine the initial depth maps to ensure the accuracy of the depth edge as much as possible. The datasets in Test set B did not have accurate initial depth maps. First, one experiment on Test set B was mainly analyzed to show the good preprocessing performance of our proposed network.

The intermediate results of our first CNN block are compared with depth maps preprocessed by other methods, as shown in Fig. 9. The original depth map resolution (305 × 55) in Test set B does not match the resolution
of the corresponding texture images (1704 × 2272). In Fig. 9(b), bilinear interpolation was adopted to resize the original depth maps to the same size as the texture images, and we can see that the details of the scene structure cannot be presented clearly in the zoomed depth maps. A deep learning-based depth prediction method [16] was used for comparison. As shown in Fig. 9(c), the predicted depth maps have greater advantages than the interpolated results shown in Fig. 9(b). It should be noted that this data-driven solution uses the original texture image as the only input. In contrast, our first CNN block combines both the texture image and initial depth cues in Fig. 9(b) into the network. In this way, hallucinated cues learned from databases and local cues learned from initial depth maps are considered together, so it can perform more optimization on depth maps and display the richest depth layers, as shown in Fig. 9(d). Meanwhile, it can be seen that our method introduces some errors from the original initial depth map at the same time, such as the areas shown in the dotted boxed rectangles in Figs. 9(d) and (b). For-
Fig. 8. Virtual view images using Test set B including (from left to right) Street and Building: virtual images generated by (a) AF, (b) PM, (c) EC, (d) GC, (e) our proposed method.

Table 2. PSNR and SSIM comparisons.

<table>
<thead>
<tr>
<th>Test set</th>
<th>PSNR [dB]</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>AF</td>
<td>PM</td>
</tr>
<tr>
<td>Adirondack</td>
<td>27.21</td>
<td>29.29</td>
</tr>
<tr>
<td>Sticks</td>
<td>28.64</td>
<td>29.86</td>
</tr>
<tr>
<td>Umbrella</td>
<td>28.26</td>
<td>29.49</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>AF</th>
<th>PM</th>
<th>EC</th>
<th>GC</th>
<th>Proposed approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adirondack</td>
<td>0.8864</td>
<td>0.9212</td>
<td>0.9245</td>
<td>0.9362</td>
<td><strong>0.9452</strong></td>
</tr>
<tr>
<td>Sticks</td>
<td>0.9149</td>
<td>0.9315</td>
<td>0.9321</td>
<td>0.9387</td>
<td><strong>0.9517</strong></td>
</tr>
<tr>
<td>Umbrella</td>
<td>0.8945</td>
<td>0.9348</td>
<td>0.9218</td>
<td>0.9376</td>
<td><strong>0.9412</strong></td>
</tr>
</tbody>
</table>
The other experimental results for the intermediate outputs of the second cascaded CNN block are presented in Fig. 10. In this experiment, to avoid the influences of other factors, the original accurate depth map $D_1$ on the dataset Umbrella from Test set A was directly set as one input of the second CNN block instead of the intermediate depth maps $D'_1$ from the previous first CNN block. The original image is regarded as a left view; then, for the right view, the areas of the newly exposed holes are located along the right side of the foreground objects, as shown in Fig. 10(a). The solid rectangular area in Fig. 10(a) is set as the region of interest (ROI), and in the following we mainly focus on the intermediate experimental results in this ROI. The warped edge map in Fig. 10(b) was used as a primarily guided constraint by our second CNN block, where the disocclusion regions are marked with solid rectangle. From the corresponding recovered edge map in Fig. 10(c), it can be observed that the primary restored parts marked in the dotted rectangle for an edge...
map are line structures. Therefore, it is much easier to deal with the disocclusion for a warped edge map than for a warped image, for which a more complicated texture recovery process would be carried out. **Fig. 10(d)** shows the intermediate result without edge recovery in the ROI using our second CNN block. Different depth layers are not clearly distinguished in the areas marked with dotted rectangle. In our scheme, an EGGL optimization process was performed. **Figs. 10(e) and (f)** show the optimized results with edge recovery of different loop times, and we can see that after three loops, the predicted depth map in the ROI is more satisfactory. Through this experiment, we can see the effectiveness of our edge-first and global-looping optimization strategy adopted in the second cascaded joint filter CNN block for progressive scene structure reconstruction.

### 3.4.2. The Effect of DDRL

For occlusion-filling refinement, a novel loss function DDRL was proposed to provide additional constraints for network training in our work. To show its efficiency, we conducted an ablation experiment on the dataset *Umbrella*. Similar to the experiment in **Fig. 10**, we also focused on the ROI of the warped image in **Fig. 11(a)** in this experiment. Although deep-learning-based inpainting [30] has achieved great success in general image restoration, it still cannot fill the holes well alone, as shown in **Fig. 11(b)**. The main reason for this is that the disocclusion areas are always the transitional regions among different object layers, and hence, it is more difficult to restore them using correct background cues without special constraints. We retrained this model by adding the overall loss function to our DDRL. It can be seen that the newly generated result in **Fig. 11(c)** is significantly improved. The best result was achieved, as shown in **Fig. 11(d)**, using our entire processing scheme. This is because the previous cascaded CNN blocks in our network generated a more reasonable coarse prediction for the final refinement stage.

### 3.4.3. Quantitative Analysis of Ablation Study

We further conducted quantitative ablation studies on *Test set A*. The quantitative comparison in **Table 4** indicates that the proposed strategies in our scheme, including the EGGL optimization process in the cascaded CNN blocks and training with DDRL in the refinement stage, considerably enhance the performance of our network. Furthermore, this experiment proves that scene structure restoration for coarse disocclusion hole region prediction is more crucial than the subsequent refinement stage in the proposed scheme.

### 4. Conclusions

This study presents a new learning-based framework that models conventional DIBR synthesis pipelines. The proposed scheme combines both depth map preprocessing and hole-filling refinement using cascaded joint filter block-based CNNs and a residual learning-based GAN in a unified framework. Thus, the synthesized virtual images can be progressively optimized. In addition to the newly introduced framework, other contributions are proposed. First, in cascaded CNN blocks, our scheme performs scene structure reconstruction in the occlusion regions using an EGGL optimization strategy. Then, for better hole-filling refinement, a novel loss function, DDRL, is adopted as an additional constraint to improve image restoration among different object layers during training. The experimental results demonstrate that the proposed...
Fig. 11. Intermediate experimental results using our residual learning based GAN on Umbrella. (a) Warped texture image, deep learning based inpainting [30], (b) without our DDRL in the ROI, (c) with our DDRL in the ROI, (d) generated novel view through our whole processing scheme in the ROI.

Table 4. Ablation study results.

<table>
<thead>
<tr>
<th>Methods</th>
<th>PSNR [dB]</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full</td>
<td>30.87</td>
<td>0.9460</td>
</tr>
<tr>
<td>Full w/o EGGL optimization</td>
<td>29.16</td>
<td>0.9378</td>
</tr>
<tr>
<td>Full w/o DDRL</td>
<td>29.74</td>
<td>0.9413</td>
</tr>
</tbody>
</table>

method outperforms the current related stereoscopic synthesis methods in terms of both quantitative metrics and subjective visual quality.

Acknowledgments

This work was supported in part by the Foundation of Excellent-Young-Backbone Teacher of Colleges and Universities in Henan Province under Grant 2019GGJS182, and in part by the Key Scientific Research Project of Henan Colleges and Universities under Grant 21B120001.

References:

Liu, W., Ma, L., and Cui, M.


