Multi-view 3D Reconstruction Based on Deformable Convolution and Laplace Pyramid Residuals

Zhaoming Hao, Ziyang Zhang, Hongyan Li, Baoqing Xu, Xiaoqiong Zhang, Meng Xu, Weifeng Wang

***Abstract*—The current deep learning-based multi-view stereo point cloud reconstruction method has been found to have low reconstruction accuracy for target boundary contours. This paper proposes a high-precision and high-completeness multi-view stereo reconstruction network (DL-PatchMatchNet) based on an improved PatchMatchNet. Firstly, to increase the robustness of the model's feature extraction, a deformable convolution-based feature extraction network is proposed. Secondly, to improve model reconstruction of target contours and boundaries, the Laplace pyramid residuals are introduced to guide the decoding process of the model. Lastly, a fused loss function (GSS) is proposed to enhance the accuracy of point cloud reconstruction by simultaneously considering geometric consistency loss, structural similarity metric and smoothing loss. The results of the experimental analysis on the DTU dataset demonstrate that the DL-PatchMatchNet model exhibits a lower mean absolute error (MAE) and error rate (ER) than other competing networks. This performance is reflected in the high accuracy and completeness of reconstruction achieved by the DL-PatchMatchNet model.**

***Index Terms*—Point cloud, Deformable convolution, Laplace pyramid residuals, PatchMatchNet**

I. Introduction

In Computer Vision and Computer Graphics, Multi-view Stereo Reconstruction (MVS Reconstruction) has been a pivotal and challenging task. Its objective is to recover the three-dimensional information of a scene and reconstruct its three-dimensional model using images captured by a camera from different viewpoints. This is achieved through the application of techniques such as feature point extraction and matching, camera pose estimation, triangulation and beam method leveling. Over the past few years, the application of 3D point cloud reconstruction has become widespread in several areas, including the estimation of human poses, the detection of unmanned aerial vehicles (UAVs), the navigation of robots, and the development of autonomous vehicles [1].

The traditional MVS method [2] employs a number of techniques to compute the similarity of multiple views. These include photometric consistency and geometric consistency, which are employed to derive the three-dimensional structure of the scene based on the geometric relationship. This is accomplished by extracting feature points from images with multiple viewpoints and calculating the camera pose [3]. However, this approach is overly reliant on the initial reconstruction results, which can lead to limitations when dealing with complex scenes, missing textures and occlusions [4].

The advent of deep learning techniques has prompted numerous researchers to shift their research focus from traditional 3D reconstruction methods to 3D reconstruction of target scenes through the use of neural networks. In comparison to traditional methods, deep learning-based MVS methods employ convolutional neural networks to generate more comprehensive and accurate point clouds, which possess the advantages of self-learning, adaptability and scalability. Deep learning-based MVS methods can be classified into three categories: voxel-based MVS [5], point cloud-based MVS [6][7] and mesh-based MVS [8][9].

Voxels are an extension of pixels in three-dimensional space that discretise the scene into a three-dimensional mesh, which makes it easy to represent the position and shape of objects in three-dimensional space, making the reconstruction process more intuitive and easier to understand. The VoxNet network, first proposed by Maturana et al. [10], proposes using three-dimensional convolutional neural networks to process the meshed voxels of the target. Wu et al. [11] proposed the use of 3D ShapeNets models directly on 3D voxels for 3D convolutional operations. While this approach offers superior reconstruction results compared to traditional 3D reconstruction methods, the number of required voxels increases exponentially with scene complexity, leading to significant memory consumption challenges in large-scale reconstruction. In order to get around the inherent trade-off

Manuscript received February 27, 2024; revised June 11, 2024. This work is supported by the National Nature Science Foundation of China (NSFC, Grant no. 52074213), and by the National key research and development program in China (Grant no.2021YFE0105000).

Zhaoming Hao is an Associate Professor of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 2278285824@qq.com).

Ziyang Zhang is a postgraduate student in Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 1119308629@qq.com).

Hongyan Li is a Senior Engineer of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: lihongyan@xust.edu.cn).

Baoqing Xu is a postgraduate student in Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 2586678038@qq.com).

Xiaoqiong Zhang is a postgraduate student in Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 2321255257@qq.com).

Meng Xu is a postgraduate student in Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: xumeng9808@foxmail.com).

Weifeng Wang is a Professor of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: wangwf03@126.com).

between the amount of memory used and the accuracy of the reconstruction, Wang et al. [12] from Microsoft Research Asia devised O-CNN, which employs an adaptive voxel convolution technique with an octree data structure. This limits the computation of planar surfaces to the neighbourhood of planar surfaces, significantly reducing the overhead of voxel computation. However, it remains inadequate for complex shapes or objects with regular boundaries [13].

To address the shortcomings of 3D voxels in dealing with scene details, the point cloud-based MVS method [14] initially generates point clouds from disparate viewpoints and subsequently employs a distance metric function to derive a three-dimensional point cloud representation of the scene. However, this method often requires a significant amount of time to complete. In order to enhance the efficacy of point cloud generation, MVSNet [15] proposes the utilisation of depth maps in the context of 3D reconstruction. This represents a pioneering approach, whereby the depth of each view is initially estimated, and subsequently, the depth maps are subjected to regression and fusion in order to form the final point cloud model. Depth map-based multi-view stereo (MVS) methods [16-19] encode and extract both global and local information from the scene, thereby enhancing the robustness of MVS matching, facilitates the reconstruction of low-texture regions and Fellenberg surface regions, and significantly enhances the completeness and overall quality of the reconstruction. Although the MVS method based on depth map has been shown to greatly improve the reconstruction effect of the model, it still faces two major difficulties: a large computational volume and a long training time. To address this problem, Wang et al. [20] proposed to combine deep learning with the PatchMatch method to estimate the depth of the view by PatchMatch [21], which not only reduces model training time, but also effectively reduces the number of model parameters. Nevertheless, the method still encounters difficulties in processing thin structures or untextured surfaces and in achieving an optimal reconstruction of edges.

The main contributions of this paper in response to these questions are as follows:

1) We propose a novel feature extraction network that fuses deformable convolution (DeConv) into the feature extraction process. The property of deformable convolution is employed to replace some redundant convolutional blocks in the model, with the use of small convolutional kernels. This improves the model's ability to extract features while reducing the number of parameters.

2) The introduction of Laplacian Pyramid Residual (LPR) calculation serves to guide the model decoding, thereby enhancing the network's ability to learn features on the target contour and boundary, and improving the reconstruction quality of the model on the target contour.

3) A new fusion-type loss function is proposed that simultaneously considers geometric consistency loss, structural similarity metric, and smoothing loss, with the objective of improving the reconstruction quality of the model while ensuring model accuracy.

II. Related Work

Over the past few years, researchers have devoted a great deal of attention to image-based 3D reconstruction in computer vision applications. This paper focuses on the enhancement of the existing network, PatchMatchNet, and the achievement of high accuracy 3D reconstruction of targets.

A. PatchMatch Algorithm

Since its inception in 2009, the PatchMatch algorithm has been extensively employed due to its high efficiency and quality, which enhances the performance of image restoration in comparison to all preceding algorithms within the image restoration class. The PatchMatch algorithm is comprised of three primary components: initialisation, propagation, and random search. The initialisation phase involves the assignment of an initial value to the nearest neighbour field. This value may be completely random or may include the incorporation of a priori information. Once the initialisation phase is complete, the iterative process commences. Each iteration constitutes a full-field scanning process. Iterations are divided into two categories: odd and even. In order to scan the odd iterations, the scanning process begins at the top and progresses in a bottom-to-top, left-to-right direction. Conversely, the even iterations are scanned in a bottom-to-top, right-to-left direction. Each scan encompasses two processes: propagation and random search. The propagation process is employed to identify the optimal value within each iteration. Propagation can be expressed by the following formula for odd iterations:

$$f(x,y) = \arg\min_f [D(f(x,y)), D(f(x-1,y)), D(f(x,y-1))] \tag{1}$$

where $f(x,y)$ denotes the value corresponding to scanning into row x and column y; $f(x-1,y)$ denotes the value corresponding to the left side of (x,y); $f(x,y-1)$ denotes the value corresponding to the upper side of (x,y); and $D(v)$ denotes the matching error. In contrast, random search involves introducing a random perturbation in order to escape the local optimum and identify the global optimum matching block.

B. PatchMatchNet Stereo Reconstruction Model

The powerful feature extraction capability of deep learning has driven the rapid development of MVS. The utilisation of deep learning in MVS methods is gradually superseding the traditional approach. In light of the outstanding performance of the MVSNet model by Yao et al. on the outdoor dataset Tanks and Temples, researchers have identified the reduction of the model's memory consumption as a key objective. In comparison to the MVSNet family of models, PatchMatchNet exhibits superior performance in terms of speed, memory usage, image resolution, and suitability for resource-constrained devices.

The network structure of PatchMatchNet comprises two components: feature extraction and learnable PatchMatch. The feature extraction component employs a feature pyramid structure analogous to that of FPN, which effectively fuses the semantic information present in the deep feature map with the positional information present in the shallow feature map. This enables the transmission of target feature information of varying sizes, and enhances the network's multi-scale

prediction capabilities. The Learnable PatchMatch component extends the core algorithm of PatchMatch to include a learning-based propagation and evaluation module that is based on deep features. This module provides a novel and teachable scheme for propagating and evaluating each iteration. It preserves the advantages of PatchMatch's small memory footprint, rendering the model disparity range independent and eliminating the need for three-dimensional cost-volume regularisation. This significantly enhances the model's computational speed.

C. Deformable Convolution

Convolution plays a pivotal role in deep learning, enabling the model to effectively process data with structural information such as images and texts. This improves the performance and efficiency of the model. However, ordinary convolution has certain limitations in dealing with target deformation, such as target rotation or scale change. This results in an inability to accurately capture the characteristics of the target. To address this challenge, Dai et al. [22] put forth a novel deformable convolution approach, depicted in Figure 1. Understanding the deformations of the data allows the model to adapt its learning field to the input data, improving the model's ability to learn. The introduction of an offset allows the input feature map to be translated and deformed, increasing the expressiveness of the model.

The conventional two-dimensional convolution process comprises two principal components: the sampling of the feature graph x utilising a regular grid R; and the subsequent weighted and summed calculation of the sampling values by W. The grid size and expansion rate are defined by R. For illustrative purposes, if the convolution kernel size is 3×3 and the expansion rate is 1, R can be expressed as follows:

$$R=\{(-1,-1),(-1,0),\ldots,(0,1),(1,1)\} \tag{2}$$

The P_0 value of each pixel in the output feature map y can be calculated as follows:

$$y(P_0)=\sum_{P_n\in R} W(P_n)\cdot X(P_0+P_n) \tag{3}$$

Deformable convolution introduces an offset of $\{\Delta P_n \mid n=1,2,\ldots,N\};(N\in|R|)$ to each point in R, as calculated by Eq.3. This offset is derived from the input feature map and additional convolution, which is typically expressed in decimal form. The deformable convolution formula is expressed as follows:

$$y(P_0)=\sum_{P_n\in R} W(P_n)\cdot X(P_0+P_n+\Delta P_n) \tag{4}$$

The position calculated by the aforementioned equation is typically a decimal number and does not correspond to the actual pixel point on the feature map. Consequently, it must also be interpolated and calculated, typically using the bilinear interpolation method. The formula is expressed as follows:

$$\begin{aligned} x(p) &= \sum_q G(q,p)\cdot x(q) \\ &= \sum_q g(q_x,p_x)\cdot g(q_y,p_y)\cdot x(q) \\ &= \sum_q \max(0,1-|q_x-p_x|)\cdot\max(0,1-|q_y-p_y|)\cdot x(q) \end{aligned} \tag{5}$$

where p denotes an arbitrary position; q enumerates all spatial positions in the feature map x; and $G(\cdot,\cdot)$ is a bilinear interpolation kernel.

III. Methodology

This paper presents an improvement to the feature extraction backbone network of the PatchMatchNet stereo reconstruction framework. The network is enhanced by the use of deformable convolution instead of ordinary convolution, allowing it to adaptively extract features. Furthermore, Laplace pyramid residuals are introduced to enhance the edge information of the extracted feature maps, thereby improving the reconstruction effect of the edges of the objects. Its general layout can be seen in Figure 2.

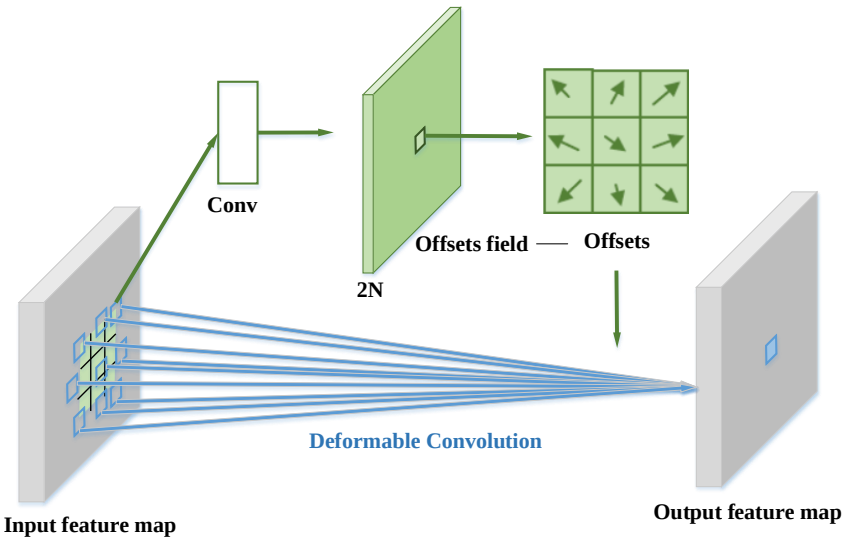

Fig. 1. Deformable convolute on schematic diagram

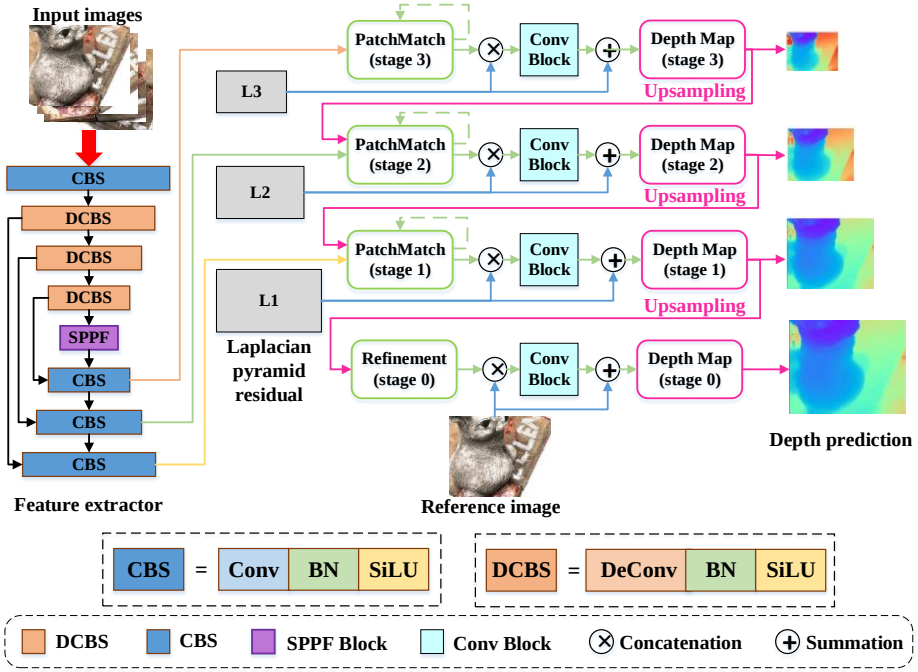

Fig. 2. DL-PatchMatchNet stereo reconstruction network structure diagram

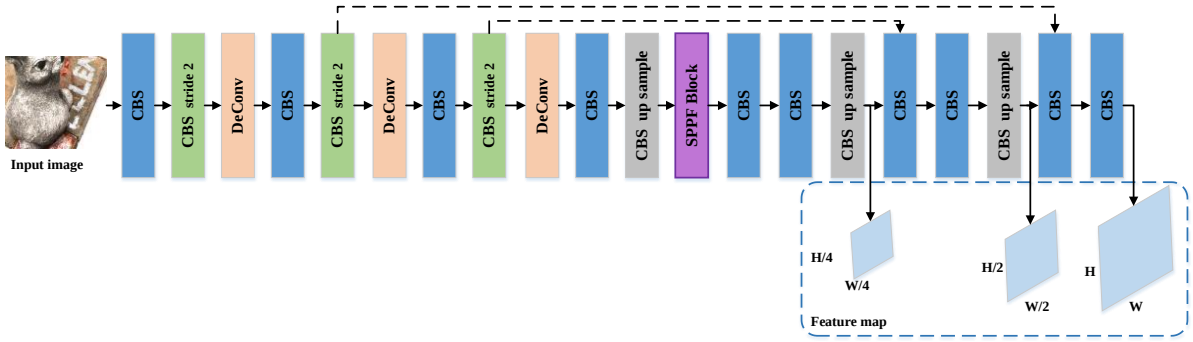

Fig. 3. Adaptive feature extraction network based on deformable convolution

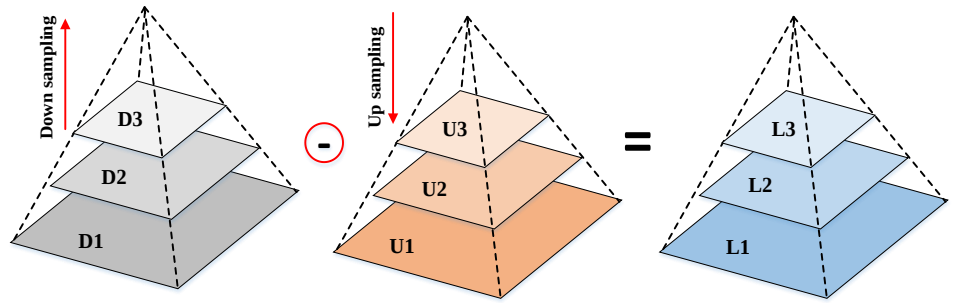

Fig. 4. Laplacian pyramid residual structure diagram

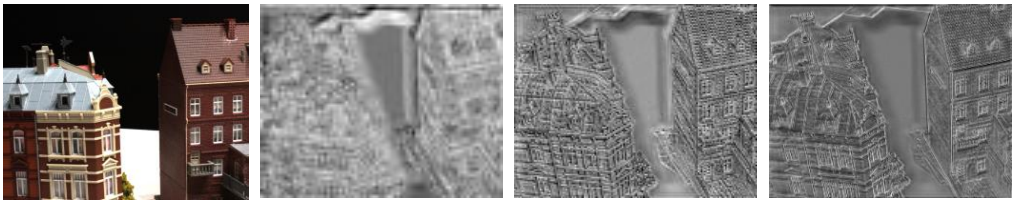

Fig. 5. Plots of Laplace pyramid residuals at different scales, from left to right, original, residuals with 8-fold down-sampling, residuals with 4-fold down-sampling, residuals with 2-fold down-sampling

A. Adaptive Feature Extraction

In the context of 3D reconstruction, it is of paramount importance to enhance the model in order to extract discernible and dependable features, thereby improving the accuracy of the reconstruction. However, conventional 2D convolution is unable to extract the most efficacious features when dealing with reflective surfaces, untextured or less textured regions due to its fixed receptive field [23]. To address this issue, we propose that the 2D convolution be enhanced with a larger perceptual field. However, in the context of regions with a high degree of texture, a smaller perceptual field may be more effective in extracting more robust features.

In order to improve the ability of the model to extract features from the data, deformable convolution is employed in this paper to extract features adaptively. The introduction of deformable convolution into the feature extraction network replaces some of the conventional convolution operations, as illustrated in Figure 3. This enables the network to adaptively aggregate contextual information over different texture richness and multi-scale regions, thereby obtaining more robust features. This methodology enhances the model's capacity to quantify the depth of an object.

In the feature extraction network, this paper employs four distinct scales of feature maps, with dimensions of $H \times W$, $\frac{H}{2} \times \frac{W}{2}$, $\frac{H}{4} \times \frac{W}{4}$, and $\frac{H}{8} \times \frac{W}{8}$. The feature map of $H \times W$ serves as a shallow feature in the subsequent feature decoding process, while the feature maps of the latter three scales are processed using three single-step deformable convolutions with varying parameters. The smaller features are subsequently upsampled to the feature maps of $H \times W$ using bilinear interpolation and skip-step concatenation.

Notwithstanding, while the incorporation of deformable convolutions will enhance the model's feature extraction capacity, it will also necessitate an increase in the computational burden and parameters associated with the model. In order to reduce the cost of computation and to maintain the accuracy of the reconstruction of the model, in this work, the size of the receptive field adaptively adapts to the input by means of deformable convolution. The downsampling component of the feature extraction process has been optimised, with the convolution kernel size of the downsampling convolution being adjusted from the original value of 5×5 to 3×3. This not only results in a reduction in network parameters, but also ensures that the model has a strong feature extraction capability.

B. Residual Bootstrap Decoding

The majority of existing depth estimation models employ a feature extraction approach that utilises the features extracted from the encoder. Subsequently, the up-sampled features are converted into a depth map. This conversion is achieved through the use of a symmetric decoding structure, which up-samples the features back to their original dimensional size. However, this conversion does not take into account the depth boundary information of the target at different scale levels, which may result in an inaccurate estimation of the target's boundary depth [24].

To solve this problem, this paper makes use of the Laplace operator's capacity to retain the local information present in the input data. Furthermore, it employs the Laplace pyramid structure, which places emphasis on the differences in the space of different scales, which are highly correlated with the boundaries of the objects. This is illustrated in Figure 4. The encoded features are initially processed through stacked convolutional blocks, where residuals are computed at each pyramid layer. These residuals are then combined to generate a depth map, progressing from coarse to fine. This process enhances the model's capacity to predict the depth of object edges. The utilisation of Laplace pyramid residuals enables a more efficient utilisation of coded features to estimate the depth information of the target.

The residual L_k for each level of the Laplace pyramid is calculated as follows:

$$L_k = D_k - U_k; \left(k = 1, 2, 3\right) \tag{6}$$

Where k denotes the level index of the Laplace pyramid; D_k is obtained by down-sampling the original input image; U_k a is obtained by up-sampling D_k, and bilinear interpolation is used in the process of resizing the image.

In the bootstrap model decoding process, assuming that R_{k} is the residual obtained from the k-th pyramid, the residual is obtained by concatenating the shallow features x_k with L_k and the up-sampling result of the deep residual, which is obtained from the Laplace pyramid of the $(k+1)$-th layer. This result is then added to L_k after stacking the convolution block B_k. This process can be expressed by the following equation:

$$R_k = B_k([x_k, L_k, up(R_{k+1})]) + L_k; \left(k = 1, 2, 3\right) \tag{7}$$

where B_k generates the result as a single-channel feature map of the same size as L_k; $up(\cdot)$ denotes the up-sampling function. The decoding process guided by L_k is able to better recover local details on different scale spaces and improve the boundary prediction of the depth map.

Figure 5 illustrates the Laplace pyramid residuals at varying scales. As evidenced by the figure, the Laplace pyramid residuals effectively retain the boundary information of the object at different scales, thereby enhancing the network's capacity to accurately reconstruct the target boundary.

C. Loss Function

The loss function employed in the network proposed in this paper comprises three principal components: Geometric Consistency Loss(L_{GC}) [25], Structure Similarity Index Measure (SSIM) [26], and smoothing loss(L_{smooth}) [27]. The proposed fused loss function is designated as GSS.

In the event that two adjacent images, designated I_a and I_b, are provided, the depth of I_a is initially predicted by the network model. Subsequently, the predicted depth map, D_a,

and the camera position, P_{ab} , are employed to project D_a to $D_b^{'}$ via microscopic bilinear interpolation. The discrepancy between the predicted depth, $D_b^{'}$, and its true depth, D_b^T , is then quantified using the following equation:

$$L_{GC} = \frac{1}{|V|}\sum_{p\in V}\frac{\left|D_b^T(p) - D_b^{'}(p)\right|}{D_b^T(p) + D_b^{'}(p)} \tag{8}$$

where V denotes the number of valid points of the depth map $D_b^{'}$ pixel points for the reconstruction target.

However, in a real-world setting, light intensity fluctuates in real time. Therefore, in the aforementioned equation, an additional similarity metric is incorporated to normalise the brightness of pixels, thereby enabling the model to more effectively address light changes in images. The structural similarity loss function is as follows:

$$L_s = \frac{1 - SSIM_{bb^{'}}(p)}{2} \tag{9}$$

where $SSIM_{bb^{'}}$ is the similarity between $D_b^{'}$ and D_b^T computed by the $SSIM$ function.

The generated depth maps are often characterised by the presence of noise, which is less effective in the bottom texture regions and in repetitive regions. In order to achieve a more effective output depth map, the depth map is adjusted using the smoothing loss function, as shown below:

$$L_{smooth} = \sum_p (e^{-\nabla I_a(p)} \cdot \nabla D_a(p))^2 \tag{10}$$

where ∇ denotes the first order derivative in the spatial direction, which ensures the smoothness of the image edges.

The weighted combination of the three losses is employed as a novel loss function, with the following formula:

$$L = \alpha L_{GC} + \beta L_s + \gamma L_{smooth} \tag{11}$$

where α , β , and γ are the weight values, which are taken in this paper as $\alpha = 0.3$, $\beta = 0.3$, $\gamma = 0.5$ respectively, The details of the experiment are shown in Table 2.

IV. Experiments

A. Experimental Settings

In this study, we implemented DL-PatchMatchNet on PyTorch 1.12.1 and trained it on the DTU training dataset. Subsequently, the trained model was subjected to an evaluation on the DTU test set. In this paper, the resolution of the input image is adjusted to 640×512 , the number of input images $N = 5$, and the optimiser $Adam(\beta1 = 0.9, \beta2 = 0.999)$ is used with an initial learning rate of 0.001, which is multiplied by 0.5 at 10, 12 and 14 iterations. The training batch is set to 1, and 16 rounds of training are performed on the model.

The computer configuration employed in this experiment was an AMD Ryzen 7 5800H CPU with a 3.20 GHz clock speed, 16 GB of RAM, a GeForce RTX 3060 graphics card, and the Windows 11 operating system with CUDA version 11.6.

B. Evaluation Indicators

The experiment employs the Mean Absolute Error (MAE) between the predicted depth and the true depth to quantify the overall magnitude of the error in the prediction results, as illustrated in Eq. 12.

$$V = \left|\frac{\sum_{i=1}^{n} X_i^m - Y_i^m}{n}\right| \tag{12}$$

The depth value of each point in the predicted depth map for the reconstructed target (denoted by X_i^m) is calculated. Similarly, the depth value of each point in the true depth map for the reconstructed target (denoted by Y_i^m) is calculated. Finally, the number of pixel points in the current depth map for the reconstructed target (denoted by n) is calculated. The distance from each point in the predicted depth to the nearest true depth surface is calculated, averaged, and then its absolute value is taken as the result (denoted by V). This metric is employed to assess the efficacy of the network reconstruction model in approximating the target model. The smaller the value of V , the more closely the prediction result aligns with the actual value.

In addition, the predictive efficacy of the model is gauged through an examination of the error rate (ER) for absolute error between the predicted depth and the authentic depth in distinct threshold contexts. The calculation process is illustrated in Eq. 13:

$$W = \frac{\sum_{i=1}^{n}\left(\left|X_i^m - Y_i^m\right| > T\right)}{n} \tag{13}$$

Where T is a set threshold, in this paper T is 1mm, 4mm, 8mm. the smaller W is, the higher the prediction accuracy of the model.

C. Experimental Results And Analysis

A series of ablation experiments was conducted to validate the DL-PatchMatchNet network. Firstly, the aforementioned improvement methods were analysed in detail. Secondly, ablation experiments were designed for each improvement method, all of which were carried out in the same experimental environment. The process involved a total of 16 rounds of iterations, model training on the DTU dataset, and a subsequent evaluation of the resulting model on the DTU test set.

This paper employs the technique of migration learning to enhance the efficacy of the training process. This technique enables the model, at the outset of training, to leverage pre-trained network parameters. The outcomes of the model are presented in Table 1.

The original network is employed as the base network, with DeConv, LPR and GSS being incorporated into the basic model as ablation experiments. The average absolute error of the original network is 4.93, with error rates of 40.16%, 11.98% and 7.27% being observed for equal

distances of 1mm, 4mm and 8mm, respectively. This benchmark is used to assess the efficacy of the combined improved methods. The average absolute error of the model is 4.80, and the error rates are 36.67%, 11.22%, and 6.92%. The percentage decrease in error rates for threshold values of 1mm, 4mm, and 8mm, respectively, was 3.49%, 0.76%, and 0.35%, respectively, in comparison to the original network.

The experimental results demonstrate that the reconstruction effect of the model is enhanced to varying degrees following the incorporation of the enhanced methodology proposed in this study. The reconstruction effect of the model with solely LPR incorporated is more pronounced, as illustrated in Figure 6, which depicts the comparison of the reconstruction effect of the model before and after the incorporation of LPR. This figure reveals that the reconstruction effect of the model on the edges is significantly enhanced.

In order to reduce the influence of hyperparameter adjustment on the reconstruction effect of the model, we adjusted the parameters of the GSS function and conducted a comparison test, as shown in Table 2. In this test, the parameters of the GSS function, α, β and γ, respectively, were set to 0.3, 0.3 and 0.5. The reconstruction effect of the model is optimal, resulting in a reduction of the model's mean absolute error to 4.88. Furthermore, the error rate in the case of the thresholds T, respectively equal to 1mm, 4mm, and 8mm, is 38.86%, 11.61%, and 7.26%.

In conclusion, the enhancement of the PatchMatchNet model through the integration of DeConv, LPR, and GSS has led to a notable improvement in the reconstruction efficacy of the model. To further substantiate the advancement of the algorithm presented in this paper, we conducted performance comparison experiments with other 3D reconstruction network models operating within the same experimental environment. As illustrated in Table 3, the DL-PatchMatchNet network exhibits a markedly diminished average absolute error and error rate relative to other network models, thereby enhancing the model's capacity to predict the target depth. This, in turn, facilitates a more comprehensive and precise reconstruction of the point cloud (Figure 7).

V. Conclusion

This paper presents a network model, DL-PatchMatchNet, which offers enhanced reconstruction accuracy compared to the PatchMatchNet multiview 3D reconstruction network. The paper outlines three key contributions. Firstly, in the feature extraction network, deformable convolution is employed in place of ordinary convolution, with adaptive feature extraction of deformable convolution implemented to enhance the model's feature extraction capabilities. Secondly, while the model is utilised for depth estimation from coarse to fine, Laplace pyramid residuals are fused at different scales to guide the decoding process of the model, thereby improving the model's ability to predict the details of the target boundaries. Finally, a fused type loss function is employed, which fuses geometric consistency loss, structural similarity metric and smoothing loss by weighting, thus improving the model's overall performance. The fused data is then utilised at different scales to guide the decoding process of the model, thereby improving the model's prediction capability of the details of the target boundaries. Finally, a fused type loss function is employed, which fuses geometric consistency loss, structural similarity metric and smoothing loss by weighting, thus improving the comprehensive performance of the model. A comparison of the DL-PatchMatchNet algorithm with other advanced 3D point cloud reconstruction methods on the DTU dataset indicates that the proposed algorithm achieves superior performance.

TABLE I
THE RESULTS OF THE ABLATION EXPERIMENTS

PatchMatchNet	DeConv	LPR	GSS	MAE/mm	ER(T=1)/%	ER(T=4)/%	ER(T=8)/%
√				4.93	40.16	11.98	7.27
√	√			4.88	39.66	11.88	7.20
√		√		4.85	39.26	11.68	7.04
√			√	4.90	38.96	11.61	7.26
√	√	√		4.88	38.27	11.75	7.02
√	√	√	√	**4.80**	**36.67**	**11.22**	**6.92**

where DeConv, LPR, and GSS represent the improved methods in this paper, were added to the experiments for comparison. The data in bold in the table have the best results.

TABLE II
Comparison of model experiments with different parameters of GSS

PatchMatchNet	α	β	γ	MAE/mm	ER(T=1)/%	ER(T=4)/%	ER(T=8)/%
√	0.1	0	0	4.93	40.06	11.97	7.35
√	0.3	0	0	4.92	39.02	11.91	7.26
√	0.5	0	0	4.95	39.83	12.00	7.30
√	0.7	0	0	4.95	39.87	12.01	7.27
√	0.9	0	0	4.95	40.23	12.04	7.26
√	0.3	0.1	0	4.98	39.87	11.97	7.28
√	0.3	0.3	0	4.93	39.72	11.82	7.27
√	0.3	0.5	0	4.95	39.79	11.93	7.27
√	0.3	0.7	0	4.95	39.81	11.93	7.26
√	0.3	0.9	0	4.97	39.82	11.92	7.27
√	0.3	0.3	0.1	4.93	39.79	11.85	7.28
√	0.3	0.3	0.3	4.90	39.72	11.83	7.27
√	**0.3**	**0.3**	**0.5**	**4.88**	**38.86**	**11.61**	**7.26**
√	0.3	0.3	0.7	4.93	39.52	11.79	7.26
√	0.3	0.3	0.9	4.93	41.02	11.75	7.26

where α , β , and γ denote the weighting coefficients of L_{GC} , L_s , and L_{smooth} , respectively. The data in bold in the table have the best results.

TABLE III
Comparison of experimental results with other algorithms

model	MAE/mm	ER(T=1)/%	ER(T=4)/%	ER(T=8)/%
MVSNet[28]	9.40	43.20	14.26	9.87
Fast-MVSNet	8.96	42.63	13.62	8.53
AA-RMVSNet[29]	6.38	41.49	12.16	7.83
R-MVSNet[30]	6.70	41.70	12.68	8.26
PatchMatchNet	4.93	40.16	11.98	7.27
DL-PatchMatchNet(ours)	**4.80**	**36.67**	**11.22**	**6.92**

Compare with other 3D reconstruction network models and record their corresponding MAE, ER metrics The data in bold in the table have the best results.

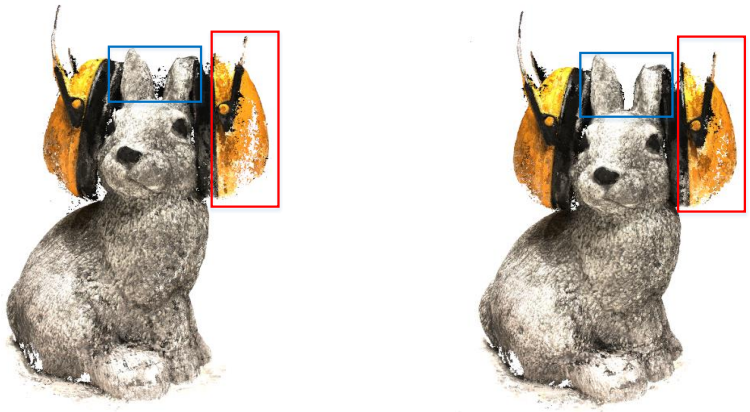

Fig. 6. Comparison of the reconstruction of Scan33 for the DTU dataset.

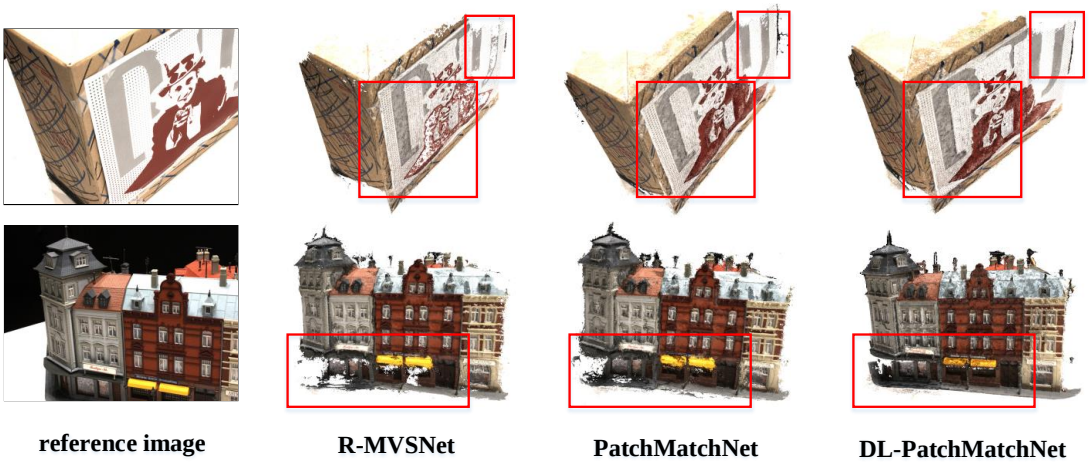

Fig. 7. Comparison results of different models on Scan9 and Scan13 in the DTU dataset.

REFERENCES

[1] J. F. Ren, X. D. Jiang. "A three-step classification framework to handle complex data distribution for radar UAV detection," *Pattern Recognit, 111 (2021)*, Article 107709.

[2] B. F. Fang, G. F. Mei, X. H. Yuan, et al. "Visual SLAM for robot navigation in healthcare facility," *Pattern Recognit*, 113 (2021), Article 107822.

[3] Y. D. Wang, T. Ran, Y. Liang, G. Q. Zheng. "An attention-based and deep sparse priori cascade multi-view stereo network for 3D reconstruction," *Computers & Graphics*, 116: 383-392, 2023.

[4] J. N. Gao, D. H. Kong, S. F. Wang, J. H. Li, B. C. Yin. "CIGNet: Category-and-Intrinsic-Geometry Guided Network for 3D coarse-to-fine reconstruction," *Neurocomputing*, 554: 126607, 2023.

[5] C. Liu, D. Kong, S. Wang, J. Li and B. Yin, "DLGAN: Depth-Preserving Latent Generative Adversarial Network for 3D Reconstruction," *IEEE Transactions on Multimedia*, vol. 23, pp. 2843-2856, 2021.

[6] Y. Tong, H. Chen, N. Yang, M. I. Menhas, B. Ahmad. "3D-CDRNet: Retrieval-based dense point cloud reconstruction from a single image under complex background," *Displays*, 78: 102438, 2023.

[7] P. Wang, L. Liu, H. X. Zhang, T. S. Wang. "CGNet: A Cascaded Generative Network for dense point cloud reconstruction from a single image," *Knowledge-Based Systems*, 223: 107057, 2021.

[8] N. Wang et al. "Pixel2Mesh: 3D Mesh Model Generation via Image Guided Deformation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, pp. 3600-3613, 1 Oct. 2021.

[9] C. Wen, Y. Zhang, C. Cao, Z. Li, X. Xue, Y. Fu. "Pixel2Mesh++: 3D Mesh Generation and Refinement From Multi-View Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 2, pp. 2166-2180, 1 Feb. 2023.

[10] Y. Li, Z. J. Zhao, J. H. Fan, et al. "ADR-MVSNet: A cascade network for 3D point cloud reconstruction with pixel occlusion," *Pattern Recognition*, 125 (2022), Article 108516, pp. 0031-3203.

[11] Z. R. Wu, S. R. Song, A. Khosla, et al. "3D ShapeNets: A Deep Representation for Volumetric Shapes," *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1912-1920, 2015.

[12] P. S. Wang, Y. Liu, Y. X. Guo, et al. "O-CNN: octree-based convolutional neural networks for 3D shape analysis," *ACM Transactions on Graphics*, 2017, 36(4): 72.

[13] M. Y. Li, W. Chen, S. S. Wang, et al. "Survey on 3D Reconstruction Methods Based on Visual Deep Learning," *Journal of Frontiers of Computer Science and Technology*, 2023, 17(2): 279-302.

[14] Y. Wei, S. H. Liu, W. Zhao, et al. "Conditional single-view shape generation for multi-view stereo reconstruction," *Proceedings of the IEEE/CVF Conference on Artificial Intelligence (2019)*, pp. 9651-9660, 2019.

[15] Y. Yao, Z. Luo, S. Li, et al. "MVSNet: depth inference for unstructured multi-view stereo," *Proceedings of the European Conference on Computer Vision (2018)*, pp. 785-801, 2018.

[16] Z. Yu and S. Gao, "Fast-MVSNet: Sparse-to-Dense Multi-View Stereo With Learned Propagation and Gauss-Newton Refinement," *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, 2020, pp. 1946-1955.

[17] X. C. Ye, S. D. Chen, R. Xu, "DPNet: Detail-preserving network for high quality monocular depth estimation," *Pattern Recognition*, 09 (2021), Article 107578.
[18] P. H. Chen, H. C. Yang, K. W. Chen, et al. "MVSNet++: learning depth-based attention pyramid features for multi-view stereo," *IEEE Transactions on Image Processing*, vol. 29, pp. 7261-7273, 2020.
[19] R. Chen, S. F. Han, J. Xu, et al. "Visibility-Aware Point-Based Multi-View Stereo Network," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 10, pp. 3695-3708, 1 Oct. 2021.
[20] F. J. H. Wang, S. Galliani, C. Vogel, et al. "PatchmatchNet: Learned Multi-View Patchmatch Stereo," *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Nashville, TN, USA, 2021, pp. 14189-14198.
[21] S. Duggal, S. Wang, W. -C. Ma, R. Hu and R. Urtasun, "DeepPruner: Learning Efficient Stereo Matching via Differentiable PatchMatch," *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, Seoul, Korea (South), 2019, pp. 4383-4392.
[22] J. F. Dai, H. Z. Qi, Y. W. Xiong, et al. "Deformable Convolutional Networks," *2017 IEEE International Conference on Computer Vision (ICCV)*, Venice, Italy, 2017, pp. 764-773.
[23] H. J. Tao. "A label-relevance multi-direction interaction network with enhanced deformable convolution for forest smoke recognition," *Expert Systems with Applications*, 236: 121383, 2024.
[24] A. Zhang, Y. Ma, J. Liu and J. Sun. "Promoting Monocular Depth Estimation by Multi-Scale Residual Laplacian Pyramid Fusion," *IEEE Signal Processing Letters*, vol. 30, pp. 205-209, 2023.
[25] L. Zou, Z. J. Huang, N. J. Gu, G. P. Wang. "Learning geometric consistency and discrepancy for category-level 6D object pose estimation from point clouds," *Pattern Recognition*, 145: 109896, 2024.
[26] S. B. Liang, Z. Y. Liu, H. K. Sun, et al. "Self-supervised Monocular Image Depth Estimation Primed by Transformer and Multi-scale Attention Scheme," *Journal of Chinese Computer Systems*. 2023, 44(4): 825-831.
[27] C. X. Wang, J. Zhou. "An adaptive index smoothing loss for face anti-spoofing," *Pattern Recognition Letters*, 153: 168-175, 2022.
[28] S. Q. Wang, J. Q. Zhang, L. Y. Li, et al. "Application of MVSNet in 3D Reconstruction of Space Objects," *Chinese Journal of Lasers*, 2022, 49(23): 2310003.
[29] Z. Z. Wei, Q. T. Zhu, C. Min, et al. "AA-RMVSNet: Adaptive Aggregation Recurrent Multi-view Stereo Network," *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, Montreal, QC, Canada, 2021 pp. 6167-6176.
[30] R. Zhao, F. F. Cai, X. Z. Wang, et al. "A Two-step Spatiotemporal Based Image Enhancement Algorithm and Its Application in 3D Reconstruction," *Industrial Control Computer*, 2022, 35(8): 78-80.