Multimedia Tools and Applications (2024) 83:787–806
https://doi.org/10.1007/s11042-023-15194-3

Fixing algorithm of Kinect depth image based on non-local means

Lin Wang[1] · Chengfeng Liao[1] · Runzhao Yao[2] · Rui Zhang[1] · Wanxu Zhang[1] · Xiaoxuan Chen[1] · Na Meng[1] · Zenghui Yan[1] · Bo Jiang[1] · Cheng Liu[1]

Received: 14 July 2021 / Revised: 2 May 2022 / Accepted: 30 March 2023 /
Published online: 24 May 2023

Abstract

The three-dimensional (3D) geometrical information that depth maps contain is useful in many applications such as 3D reconstruction or simultaneous localization and mapping (SLAM). Kinect is widely used in depth image acquisition due to its low cost and good real-time performance. However, the quality of depth images obtained by Kinect is influenced by holes which make depth image inadequate for further applications. To suppress the influence of holes on a subsequent application, a fixing algorithm of Kinect depth image based on non-local means (NLM) is proposed in this paper. The holes in depth image are filled using the weights which are calculated on the corresponding gray image by distance factor and value consistent factor. And the experiment results demonstrate that the proposed method achieves good performance in both evaluation in metrics and subjectively visual effect. This research provides a solution idea for depth image fixing algorithm with low complexity.

Keywords Kinect · Image fixing · Depth image · Non-local means

1 Introduction

Depth Images is a kind of image used to describe the spatial distance information of a scene, and is widely used in the computer vision studies such as three-dimensional (3D) reconstruction [14, 28], object segmentation [12] , automatic driving [4, 7] and gait analysis [1] etc. In order to obtain highly accurate depth images, numerous low-cost depth acquisition devices have been developed which are mainly based on three kinds of technology, structured-light reflection [24] which emits coded infrared light and estimates the depth by

✉ Bo Jiang
jiangbo@nwu.edu.cn

✉ Cheng Liu
lc@nwu.edu.cn

[1] School of Information Science and Technology, Northwest University, Shaanxi Xi'an 710127, China

[2] College of Artificial Intelligence, Xi'an Jiaotong University, Shaanxi Xi'an 710049, China

measuring speckle pattern, time-of-flight (ToF) [26] which estimates the depth by measuring the phase difference between the emitted light and the captured light after reflection, and laser scanning [2] which estimates the depth by using lidar sensor to emit the laser and calculates the time between the laser emitted and captured.

Microsoft's Kinect V2 [25] is a kind of ToF sensor that gives great convenience for real-time and active acquisition to scene depth information. However, due to occlusion or measurement range limitation, the depth images obtained by Kinect inevitably emerge holes in the texture edge area or the flat area, which greatly degrades the quality of depth images' subsequent applications such as 3D reconstruction, for the reconstructed model will be broken in the hole area and makes it difficult to use.

As shown in Fig. 1, we take a gray image (a) and a corresponding depth image (b) by Microsoft Kinect V2, the two images are strictly aligned in edges and contours. The holes randomly emerge in the homogeneous area like (c) on the wall or in the object boundaries like (d), the edge of the robot. We find that the depth values of the object boundaries are consistent with the boundaries in the gray image like (e) or (f). Moreover, blocks with similar gray values in a gray image also have similar depth values in the depth image.

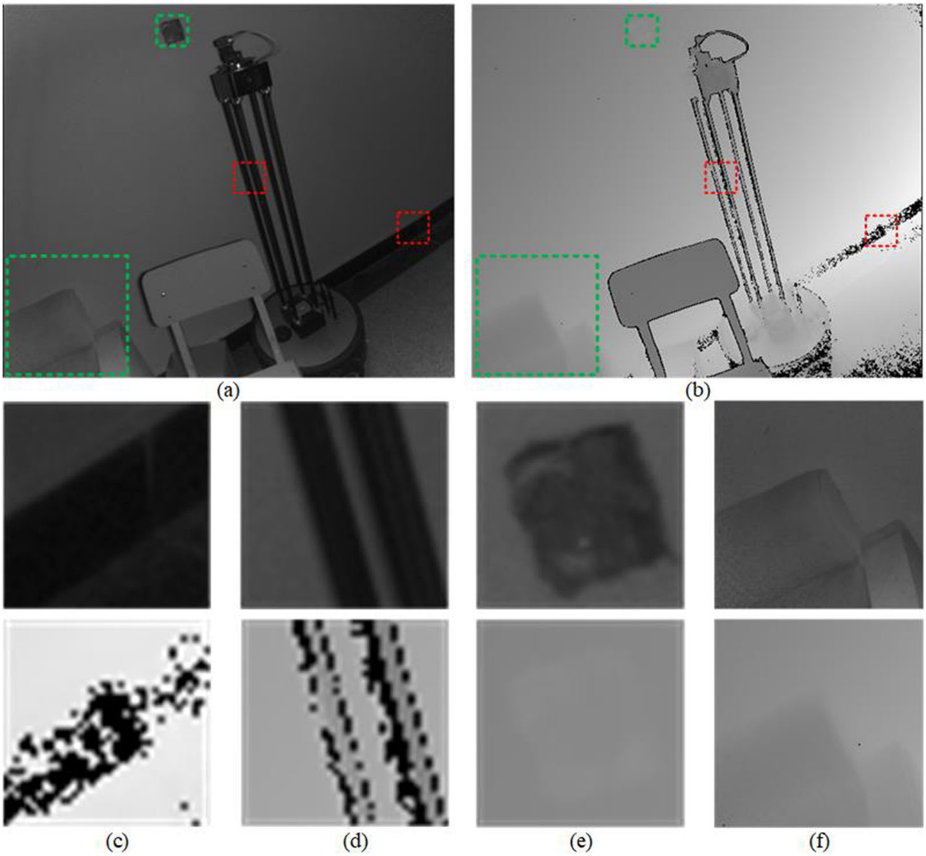

Fig. 1 The depth image and gray image captured by Kinect V2 and the zoom in area. (a) and (b) the gray image and the aligned depth image, (c) the holes in homogeneous area, (d) the holes in object boundaries, (e) and (f) boundaries consistent area

Inspired by the discovery, we propose a depth image fixing algorithm based on non-local means (NLM) [3]. The proposed method uses the gray image as the reference, and computes the grayscale weights of the neighbor points of the hole by NLM in order to measure the similarity between the hole and non-hole neighbor points, also we add distance weight to measure distance similarity which increases the accuracy of the hole filling. Overall, the proposed method shows high robustness in restoring the depth images and fills the holes with high accuracy.

The rest of the paper is organized as follows. The related works are reviewed in Section 2, and the non-local means algorithm is briefly introduced in Section 3. The proposed NLM-based fixing algorithm for Kinect depth image is given in Section 4, including the overview of the proposed method, the computation of weights, and the steps to fill the hole. In Section 5, the experimental results are analyzed to demonstrate the effectiveness and accuracy of the proposed method. Finally, the conclusion is given in Section 6.

2 Related work

Based on the feature of holes' emergence, numerous scholars have put forward their research. The hole filling algorithms can be mainly categorized into three kinds, the methods based on imaging principles, the exemplar-based methods and the texture filter-based methods.

The first kind of method takes advantage of the imaging principle to fill the hole area. According to the principle of depth image generation, Rossi et al. proposed a method using the location information from the RGB image to compute the inverse depth value of holes [20], and then comprehensively estimates the depth value by the normal map of the texture image and the non-hole area of the depth image. The method estimates the depth value of holes with the minimum disparity in the compared dataset. While Lee and Han [15] studied the theory of how do holes emerge in the depth image, and proposed a hole concealment algorithm to fill the holes. The algorithm first extracts the patches with shape and location information, then the patches are divided into different classes based on the gradient of color pixel values, and holes are filled with the neighbor data which are in the same class with the hole. This kind of method is good in filling effect, but has high algorithm complexity.

The exemplar-based methods use the exemplar patch to fill the holes in the images. The method proposed by Criminisi [9] is a typical exemplar-based hole filling algorithm that is widely adopted as the fundamental hole filling method. The method computes the filling priorities of the blocks in the contour missing area and uses the best-match patch to fill the maximum priority block in the source area.

Based on this exemplar filling method, Xiang et al. proposed an arbitrary-shape patch matching algorithm to search hole area with irregular boundaries and a cross-modal matching algorithm to search the patch that has minimum difference with the hole area, which has a good effect on processing irregular hole area [22]. Nguyen et al. raised a new priority function to find the best-matched reference patch based on spatio-temporal background information, showing a better visual effect after restoration than Criminisi's algorithm [18]. Zhang et al. proposed an object-oriented segmentation method that obtains the best sample block by object segmentation and uses it to fill the broken area [31], the fixing effect mainly depends on the accuracy of the segmentation, inaccurate segmentation can result in blurry or an unnatural filling in edge contour. From a similar angle, Bi et al. proposed a method to divide the complex scene of a depth image into several simple layers [27], the filling direction of hole area is determined by whether the hole area is more similar to the foreground layer or the background layer, the method has good filling effect in both metrics and visual effect. However, the patch finding algorithm in exemplar-based methods is a time-consuming processing, which decreases the fixing efficiency of this kind of methods.

Different from the exemplar-based method, the texture filter-based methods are basically by proposing different filters to extract the texture information from the gray images, and use the texture information to fill the holes. Jin et al. proposed a method to extract the edge from texture images and used a joint spatio-temporal dithering filter to fill the holes [30]. Similarly, the method Cho et al. proposed uses a color edge map to instruct the filling of holes [8], the edge of depth images after restoration is strictly aligned with the texture images which improves the visual performance of the depth images in color synthesized views. While Pan et al. proposed a joint optimization framework to restore the broken depth image [19]. The method computes the minimum value of the energy function for optimizing the hole filling. Chang et al. put forward a method to compute the texture and depth similarity from eight neighbor directional vectors, and then the method chooses the most similar values to fill the holes [5]. The filter-based methods use the filter to give overall enhancement of the depth image and give a good fixing result in the small area of holes, but the shortage is that they lack the robustness for dealing with the large area of holes.

In this paper, we propose a fixing algorithm based on Kinect as a solution that guarantees high repair accuracy and efficiency while having low algorithm complexity. The proposed method uses the neighbor block of the holes instead of single pixels in weight computation bringing more robustness to the hole filling. The estimation of holes is based on both texture information from the aligned gray image as well as the distance information from the depth image which increases the accuracy of the depth estimation. The weight computing process and the hole estimating process in the algorithm are not complex to keep the proposed method from computation expensive and to make it capable of fast execution.

3 NLM algorithm

Traditional denoising filtering algorithms, such as median filtering and Gaussian filtering, only take a single pixel and its neighborhood to filter the noise. And the NLM algorithm [29] estimates the value of noise images by the weight average of similar neighborhood pixel structure. The algorithm fully utilizes the redundant information of the image, which can denoise the image while keeping the image details maximally.

Given a noisy gray image G, for pixel x needed to be fixed, NLM algorithm computes the weighted average gray value of all the pixels in the image as the estimated gray value of pixel x by using the following formula:

$$NL[G](x) = \sum_{y \in A(x)} \omega(x, y) g(y) \tag{1}$$

where $A(x)$ is the searching block to fix the pixel x, $g(y)$ is the gray value of pixel y, $\omega(x, y)$ is the similar weight between the center pixel x and its neighbor pixel y, which satisfies the conditions $0 \leq \omega(x, y) \leq 1$ and $\sum_{y \in A(x)} \omega(x, y) = 1$. The weight $\omega(x, y)$ is computed as follows:

$$\omega(x, y) = \frac{1}{Z(x)} e^{-\frac{d(x,y)}{h^2}} \tag{2}$$

where $Z(x)$ is the normalizing constant and computed as:

$$Z(x) = \sum_{y \in A(x)} e^{-\frac{d(x,y)}{h^2}} \tag{3}$$

And the parameter h controls the decay rate of the exponential function, $d(x, y)$ is regarded as the similarity between pixel value vectors in the square neighborhood. And $d(x, y)$ is defined as:

$$d(x, y) = \|v(N_x) - v(N_y)\|_{2,a}^2 = e^{-\frac{\|v(N_x)-v(N_y)\|^2}{a^2}} \tag{4}$$

where $v(N_x)$ and $v(N_y)$ are the pixel-value vectors, which are composed of the pixel values in the $(2q + 1) \times (2q + 1)$ square fixing block respectively centered on pixel x and pixel y, named as $B(x)$ and $B(y)$. And the values are listed from left to right and top to bottom. $\| \cdot \|_{2,a}^2$ represents the Gaussian weighted Euclidean distance, where a is the Gaussian kernel standard deviation.

The NLM algorithm takes pixels in the whole image to fix the noise pixel. However, for the efficiency of the algorithm, $A(x)$ is normally set as the square neighborhood of pixel x, such as $(2p + 1) \times (2p + 1)$ neighborhood centered on x, rather than the whole image.

4 Methodology

4.1 Overview of the NLM-based fixing algorithm

The proposed method is based on the following two reasonable assumptions:

1) The depth values of holes have a strong relationship with non-hole points which are distributed in the neighbor area of the hole. In order to bring the texture factor into hole restoration, using a fixing block centered on the hole and composed of the neighbor pixel values rather than using a single point in weight computation brings more robustness to the algorithm.
2) The spatial distance factor, as well as the value consistent factor, should be comprehensively considered in the weights computation of holes.

Our algorithm is designed for Kinect V2 because it can simultaneously obtain depth images and gray image from the same viewpoint, so the two images have an aligned contour. And the value of a hole in depth image obtained by Kinect is zero. Based on this feature, the position of holes in the depth image can be automatically extracted by judging whether the value of the point is zero. The details of the proposed method are depicted below.

First, we take both depth image and gray image as the input images, and the proposed method extract the hole points' positions and iteratively estimates the depth value of each hole point in depth image. For each hole point x in depth image, the algorithm reads the position of the hole point x in depth image and the $p \times p$ searching block $A(x)$ centered on the corresponding position in gray image. Then, the distance weight and grayscale weight of each non-hole neighbor point $y(y \in A(x))$ are computed, and the fixing weight of each point y is the normalized product of the grayscale weight and distance weight of each point y. Finally, the depth value of the hole point x can be estimated as the weighted average of the depth values of its non-hole neighbor points in the searching block. The overview of the proposed method is shown in Fig. 1.

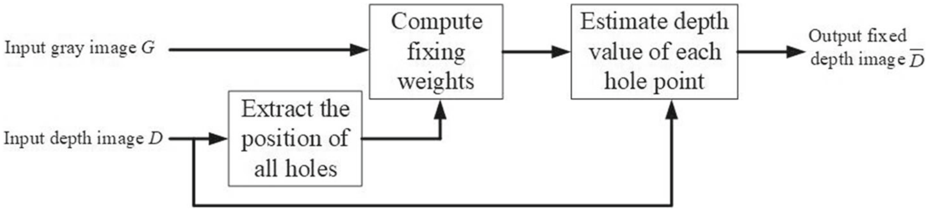
Input gray image G
Input depth image D
Extract the position of all holes
Compute fixing weights
Estimate depth value of each hole point
Output fixed depth image $\overline{D}$

Fig. 2 The overview of the proposed method

4.2 Computation of fixing weights

The fixing weight of each non-hole point around the hole point is computed after estimating the grayscale weight and distance weight. Let x denotes the location of a hole point in the depth image, and y denotes the location of a non-hole point in the neighbor searching block $A(x)$ of x. The diagram of fixing weight computation algorithm is shown in Fig. 2, and the computation algorithm is composed of three parts, that is, the grayscale weight computation, the distance weight computation, and the fixing weight computation (Fig. 3).

(1) Grayscale weight computation

For each non-hole point, y in the neighbor searching block $A(x)$, the grayscale weight of y is computed to measure the grayscale similarity between hole point x and its neighbor non-hole point y. The grayscale weight is computed as:

$$m(x, y) = e^{-\frac{d(x,y)}{h^2}} \tag{5}$$

where $d(x, y)$ is computed as (4).

(2) Distance weight computation

Since the similarity of depth value between hole point x and its neighbor non-hole point y is also related to the distance between the two points, the distance weight of non-hole point y is computed as:

$$\varphi(x, y) = e^{-\frac{\|x-y\|^2}{\sigma^2}} \tag{6}$$

where σ controls the decay rate of the exponential function and $\|\cdot\|^2$ denotes the Euclidean distance.

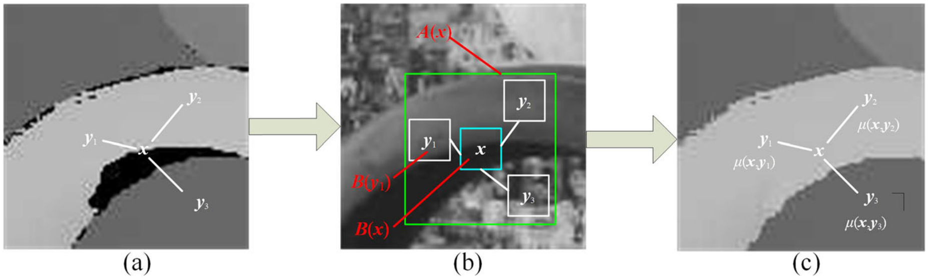

Fig. 3 Diagram of fixing weight computation algorithm.(a) Read each non-hole point y_i in the searching block $A(x)$ on the depth image. (b) Compute the grayscale weight and distance weight of each y_i on gray image. (c) Compute the fixing weight of each y_i and fill the holes on depth image

(3) Fixing weight computation

After the grayscale weight $m(x, y)$ and distance weight $\varphi(x, y)$ of each non-hole point y in $A(x)$ is computed, the fixing weight of each non-hole point y is computed to measure the similarity of depth value between hole point x and its neighbor non-hole point y. And the fixing weight is defined as:

$$\mu(x, y) = \frac{m(x, y)\varphi(x, y)}{Z_1(x)} \tag{7}$$

where $Z_1(x)$ is the normalizing constant to ensure $\sum_{y \in A(x)} \mu(x, y) = 1$, and $Z_1(x)$ is computed as:

$$Z_1(x) = \sum_{y \in A(x)} m(x, y)\varphi(x, y) \tag{8}$$

The fixing weight computation algorithm is given in Algorithm 1.

```
Input: Depth image $D$, gray image $G$, the location $(i_0, j_0)$ of hole point $x$, size of searching
    square block $2p + 1$, size of fixing square block $2q + 1$, decay rate of the grayscale
    weight $h$, decay rate of the distance weight $\sigma$, Gaussian kernel standard deviation $a$.
Output: Fixing weight $\mu(x, y)$ of each non-hole point y in $A(x)$, where $A(x)$ is the
    neighbor searching block of hole point $x$ defined as $A(x) = \{(i, j)|i_0 - p \leqslant i \leqslant
    i_0 + p, j_0 - p \leqslant j \leqslant j_0 + p\}$.
for $i = i_0 - p \rightarrow i_0 + p$ do
    for $j = j_0 - p \rightarrow j_0 + p$ do
        if $D(i, j) \neq 0$ then
            $y = (i, j)$
            Compute the grayscale weight $m(x, y)$ according to Eq.(5)
            Compute the distance weight $\varphi(x, y)$ according to Eq. (6).
        end if
    end for
end for
Compute the normalizing constant $Z_1(x)$ according to Eq. (8).
Compute the fixing weight $\mu(x, y)$ of each non-hole point $y$ according to Eq. (7).
return $\mu(x, y)$
```

Algorithm 1 Fixing weight computation.

4.3 Depth fixing of hole point

For each hole point x, after computing the fixing weight $\mu(x, y)$ of each non-hole point y in the neighbor searching block $A(x)$ of x, the depth value of hole point x can be estimated as follows:

$$\overline{D}(x) = \sum_{y \in A(x)} \mu(x, y)D(y) \tag{9}$$

where $D(y)$ denotes the depth value of non-hole point y in depth image.

The depth image can be fixed by estimating the depth value of each hole point in the depth image. And the fixing algorithm for Kinect depth image is given in Algorithm 2.

```
Input: Depth image $D$, gray image $G$, size of searching square block $2p+1$, size of fixing square block $2q + 1$, decay rate of the grayscale weight $h$, decay rate of the distance weight $\sigma$, Gaussian kernel standard deviation $a$.
Output: Fixed depth image $\overline{D}$.
Read the size of depth image $D$ as$M \times N$.
$D \to \overline{D}$
for $i = 0 \to M$ do
    for $j = 0 \to N$ do
        if $D(i, j) = 0$ then
            Let hole point $x = (i, j)$, define the neighbor searching block $A(x)$ of $x$ as:
            $A(x) = \{(i, j)|i_0 - p \leqslant i \leqslant i_0 + p, j_0 - p \leqslant j \leqslant j_0 + p\}$
            Compute the fixing weight $\mu(x, y)$ of each non-hole point $y$ in $A(x)$ according to Algorithm 1.
            Estimate the depth value of hole point $x$ according to Eq. (9).
        end if
    end for
end for
return Fixed depth image $\overline{D}$
```

Algorithm 2 Fixing algorithm of Kinect depth image.

5 Experiments

5.1 Experimental environment and parameters

In this section, extensive experiments have been conducted to verify the proposed fixing algorithm.

The experiments are carried out on thirty sets of depth and gray images belonging to Middlebury dataset [11, 21], three self-acquired sets of depth and gray images shot by Kinect V2, and ten sets of depth and gray images from NYU v2 dataset. Four complicated scenes 'Art', 'Dolls', 'Moebius' and 'Reindeer' from Middlebury dataset, three self-acquired scenes 'Floor', 'Chairs' and 'Robot' and three scenes 'Bedroom', 'Sofa' and 'Livingroom' from NYU v2 dataset are chosen to measure the visual effect, the three dataset offers the raw depth images in the range of [0,255] and with holes.

Some articles, in their objective analysis, the metrics are calculated between the restored depth images and the raw depth images. However, the raw depth images are incomplete because of the holes, the metrics between the raw and the restored depth images can mislead the analysis and the judgment of the restoration effect. In order to measure the effectiveness correctly, we manually fill the holes of all raw depth images including the thirty Middlebury depth images as well as the three self-acquired depth images as the ground truth depth images. The NYU v2 dataset contains depth images that have been restored, so we use them as ground truth depth images.

In the experiments, the size of searching square block $2p + 1$ is set to 19, the size of fixing square block $2q + 1$ is set to 15, the decay rate of the grayscale weight h is set to 2, the decay rate of the distance weight σ is set to 2, the Gaussian kernel standard deviation a is set to 2. The whole experiments are carried out on an Intel Core i7-8770 CPU (3.2GHz)

PC with 16 GB RAM. All the algorithms are programmed and tested on MATLAB platform (version R2018b).

5.2 Experimental results on Middlebury dataset

The proposed method is compared to four state-of-the-art calibration methods, namely joint bilateral filter (JBF) method [10], fuzzy C-means (FCM) method [16], texture synthesis repair (TSR) method [31] and exemplar-based depth inpainting (EDI) method [22]. The visual effect, as well as the metrics of the four methods, are measured by the following experiment.

The gray images, ground truth depth images, raw depth images and the depth images restored by the four methods of the scene ‘Art’, ‘Dolls’, ‘Moebius’, ‘Reindeer’ are shown in Fig. 4, the main holes are select in green.

In the raw depth image of ‘Art’ scene, large holes emerge mainly in the clay pot’s handle and the edge of the sculpture. JBF method restores the hole area well, the sculpture edge is restored smoothly and the holes in the handle are also properly filled. While after the restoration of FCM method and TSR method, the repaired depth image shows an irregular edge of sculpture, error fillings are also seen in the handle. The result in the select areas of EDI method are different from the surrounding, the holes in the handle part are given the depth values closer to the background than the handle itself. The proposed method fills the holes well in the handle and the sculpture, but on the left of the image, some wrong values are filled in the hole area.

In the raw depth image of ‘Dolls’ scene, holes are distributed in the image, but mainly in the doll’s face and the feet of another doll. The filling effectiveness is well of the JBF, FCM and the proposed method. However, there are some blurs in the contour of the doll’s feet of the TSR’s result which is also seen in the result of EDI method.

In the raw depth image of ‘Moebius’ scene, there is a lack of depth data mainly at the top contour of the dodecahedron box and the octahedron box. In the result of JBF method, reasonable depth values are filled in the select area, but some incorrect fillings are seen in the edge area of the polyhedron where the method uses the background depth value to restore the holes. FCM method gives some unnatural contour of the select area. The select area in the left of the scene has a vague edge in the result of TSR method and has a jagged edge in the result of EDI method, the results are bad in visual effect. As the result of proposed method, the filling of the select holes is well and shows a shape consistent with the gray image.

In the raw depth image of ‘Reindeer’ scene, holes are seen mainly on the feet of reindeer doll and the straw rope behind the doll. From the results of the five methods, JBF method and TSR method use the depth value of reindeer doll to fill the holes on the straw rope, FCM method uses the wrong depth value of the sofa to repair the holes. The results of the proposed method and the EDI method correctly restores the holes on the straw rope, but the holes restored by EDI method on the reindeer doll are wrong.

The effectiveness of the proposed method and the other three methods are also measured by three kind of metrics including structural similarity (SSIM) [13], root mean square error (RMSE) [6, 23] and peak signal to noise ratio (PSNR) [17]. The PSNR and RMSE are used to evaluate the similarity between the raw depth image and the depth image after restoration, the bigger PSNR and the lower RMSE mean the better effectiveness after restoration. SSIM is used to measure the structural similarity between the two depth images, the bigger SSIM means that the contour of depth image are more similar to the ground truth. SSIM, RMSE

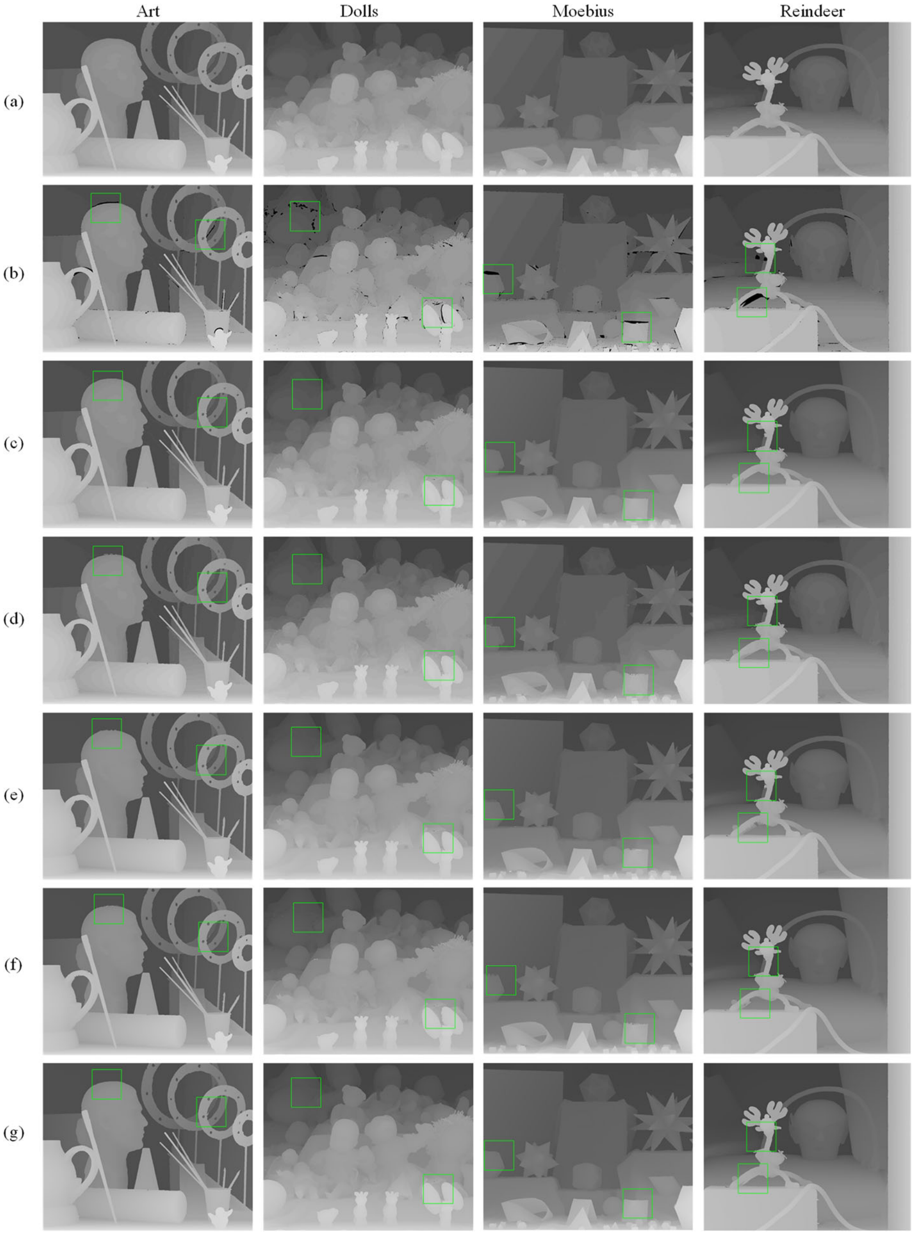

Fig. 4 Comparison of the fixing results on the Middlebury dataset. Row (a) depth images (Ground truth), Row (b) raw depth images , Row (c) Fixing results of the JBF method , Row (d) Fixing results of the FCM method , Row (e) Fixing results of the TSR method, Row (f) Fixing results of the EDI method, Row (g) Fixing results of the proposed method

and PSNR can be computed as follows:

$$SSIM(D_0, D_1) = \frac{(2\eta_{D_0}\eta_{D_1} + c_1)(\gamma_{D_0 D_1} + c_2)}{(\eta^2_{D_0} + \eta^2_{D_1} + c_1)(\gamma^2_{D_0} + \gamma^2_{D_1} + c_2)} \tag{10}$$

$$RMSE(D_0, D_1) = \sqrt{\frac{1}{mn}\sum_{i=1}^{m-1}\sum_{j=1}^{n-1}[D_0(i, j) - D_1(i, j)]^2} \tag{11}$$

$$PSNR(D_0, D_1) = 20\log_{10}(\frac{MAX_{D_0}}{RMSE(D_0, D_1)}) \tag{12}$$

where D_0 is the ground-truth depth image, D_1 is fixed depth image, η_{D_0} and η_{D_1} are the average depth value of D_0 and D_1, respectively, η_{D_0} and η_{D_1} are the variance depth value of D_0 and D_1, respectively, $\gamma_{D_0 D_1}$ is the covariance depth value between D_0 and D_1, $c_1 = (k_1 L)^2$ and $c_2 = (k_1 L)^2$ are two variables to stabilize the division with weak denominator, $k_1 = 0.01$ and $k_2 = 0.03$ by default, L is the gray level of depth images D_0 and D_1, and $MAX_{D_0} = 2^L - 1$ is the maximum possible pixel value of the image.

The metrics of the three methods are shown in Table 1.

From Table 1, the proposed method can achieve better PSNR, SSIM and RMSE parameters than other algorithms in the four presented scenes, also the proposed method has higher average PSNR and RMSE values than other algorithms, while the average SSIM is smaller. The metrics show that the proposed method is worse than other algorithms in keeping the brightness or structure consistent with the ground truth, but its fixing results have a small depth value error with the ground truth.

The average value of PSNR and SSIM is smaller than the value of the displayed image, and the RMSE will be larger. This is because some of the 30 depth images have large hole areas, and the PSNR of these depth images restored by the five algorithms are all smaller than each average PSNR. Such images account for 1/3 of the 30 images, so the average PSNR is lower than PSNR of the displayed images.

From the evaluation of visual effect and metrics, it can be found that the proposed method can obtain the restoration result that is closer to the raw depth image in the displayed scenes, and its restoration of the hole at the edge of the object can basically conform to the outline of the object in the original scene, while other restoration methods will bring about problems like blurry or irregular edge, and the proposed method also has some advantages over other methods in PSNR and RMSE metrics. However, since Middlebury dataset is a stereo dataset, there is a certain gap with the depth image obtained by using Kinect. Therefore the following experiments will be carried out on depth images taken by Kinect device.

5.3 Experimental results on self-acquired images

In this experiment, Kinect v2 is used to obtain depth images and the corresponding gray images of three different scenes. The proposed method, JBF method, FCM method, TSR method and EDI method are adopted to fix depth images, respectively. The three scenes' raw depth images, ground truth depth images with holes manually filled and the results of the four methods are shown in Fig. 5.

As shown in 'Chairs' scene, holes emerge mainly on the handle area of the chairs. JBF method fills the holes with wrong depth values, and this is also seen in the result of FCM method, the wrong depth values of the handle break the depth consistency of the chair, making the restored image less reliable. The result of TSR method is better than JBF and FCM methods, but the fillings are still unnatural on the select area on the left of the image.

Table 1 The metrics result on Middlebury dataset

Image of Scenes	JBF method			FCM method			TSR method			EDI method			Proposed method		
	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR
Art	0.674	0.989	37.814	0.556	0.991	38.644	0.622	0.992	39.474	0.660	0.993	39.461	0.266	0.998	45.803
Dolls	1.045	0.992	43.393	0.959	0.992	43.545	0.879	0.991	41.486	0.938	0.992	43.575	0.663	0.998	46.924
Moebius	0.918	0.991	42.136	1.017	0.990	41.856	1.021	0.989	40.178	0.973	0.992	42.837	0.669	0.997	45.820
Reindeer	0.833	0.992	38.525	1.012	0.992	38.293	0.943	0.992	38.195	0.856	0.993	39.862	0.815	0.998	42.737
Average of 30 images	1.217	0.976	34.203	1.722	0.985	35.075	1.683	0.981	34.268	1.283	0.983	35.679	1.005	0.984	36.547

Similarly, the result of EDI method also restores the holes on the left with the depth value closer to the ground than the depth value of the chair. While the result of the proposed method correctly restores the holes area on the chair.

In the ‘Floor’ scene, holes are around the contour and inside of the heater, some are also seen in the corner of the wall. The result of the JBF method is good, but there are some error

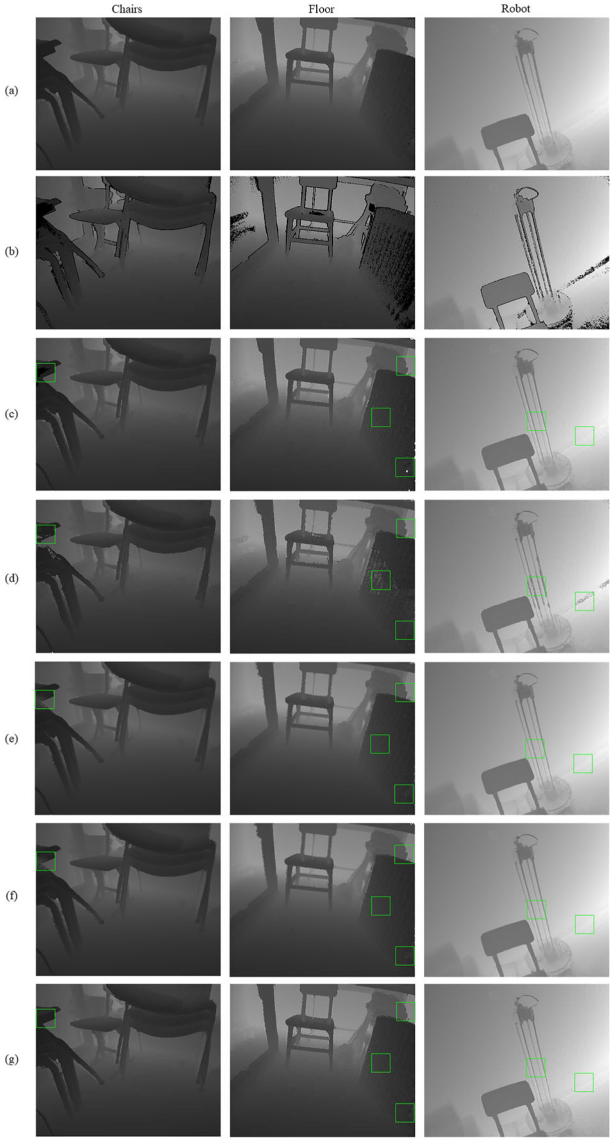

Fig. 5 Comparison of the fixing results on self acquired images. Row (a) depth images (Ground truth), Row (b) raw depth images , Row (c)Fixing results of the JBF method , Row (d) Fixing results of the FCM method , Row (e) Fixing results of the TSR method, Row (f) Fixing results of the EDI method, Row (g) Fixing results of the proposed method

fillings inside the heater as shown in the select block. FCM method fills the holes inside the heater with the depth values that are closer to the depth value of the floor, and there are also some mistakes in the fillings of the chair. The result of the TSR method is good, but the edge of the object in the restored image is not clear making the image blurred. From the result of EDI method, some holes in the chair next to the heater are filled with the depth value closer to that of the heater. While the proposed method gives a restored image with a sharp and clear edge.

Some holes in the ‘Robot’ scene are around the edge of the robot and the chair, and others are in the corner of the wall and on the ground. The visual effect of the JBF method result is good, while FCM method wrongly fills the holes in the edge of the robot and the wall with the depth value of the chair. The holes in the area of robot’s wheel are filled with the depth value of chair in the result of TSR method, which is also seen in the result of the EDI. The proposed method restores the holes with reasonable values and keeps the contour aligned with the ground truth image.

The PSNR,RMSE and SSIM results of each method are shown in Table 2.

From the data in Table 2, although the proposed method can achieve better RMSE than other methods in the three shown scenes, its PSNR in Chairs and Robot scenes are lower than those of TSR method and EDI method, and its SSIM of Chair scene is also lower than that of TSR method which can not prove the proposed method is effective for fixing the depth image acquired by Kinect device.

Due to there being only three images for comparison, and the effectiveness of the proposed method needs to be further verified, we add an extra set of experiments for verifying the proposed method.

5.4 Experimental results on NYU v2 dataset

To further confirm the effectiveness of the proposed method in the depth images captured by the Kinect device, we carry out an extra set of experiments on the NYU v2 dataset. The NYU depth v2 dataset is another RGBD dataset of indoor scenes captured by Microsoft Kinect, the depth images are aligned with the RGB image. 10 sets of depth images in the NYU depth v2 dataset are selected for the experiment and three sets of scenes ‘Bedroom’, ‘Sofa’ and ‘Livingroom’ are shown to compare the subjective visual effect as well as the metrics of each hole repair algorithm in Fig. 6.

In the ‘Bedroom’ scene, the holes mainly appear around the bed and the armrest of the sofa. The JBF method has some wrong fixing at the armrest of the sofa, but it has a good consistency with the ground truth depth image. The FCM method’s result shows illogical restoration at the armrest, and the visual effect is poor. The result of TSR method and the EDI method are close to the ground truth in structure, but there are blurred edges in the select areas. The result of the proposed method has the same error as the JBF method in the select area on the left, but the restoration on the edge of bed has a better consistency with ground truth image than the JBF method.

In the ‘Sofa’ scene, the holes mainly appear at the edge of the armrest sofa and the chair. There are some errors in the results of the JBF method, and a part of holes are repaired by depth values that are closer to the background. The results of FCM method have a more obvious error at the chair than the JBF method. The sofa boundary is vague in the result of the TSR method. While the EDI method restores the holes in the sofa boundary with illogical depth values. And there are no obvious errors in the repair results of the proposed method.

Table 2 The metrics result on self-acquired images

Image of Scenes	JBF method			FCM method			TSR method			EDI method			Proposed method		
	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR
Chairs	1.478	0.971	36.312	1.394	0.957	33.521	1.594	0.974	36.958	1.467	0.970	36.939	1.373	0.971	36.108
Floor	1.117	0.968	31.772	1.847	0.919	31.073	1.791	0.963	34.202	1.551	0.960	34.627	0.853	0.972	35.059
Robot	1.142	0.977	37.164	1.783	0.953	31.480	1.305	0.981	38.486	0.965	0.980	38.100	0.750	0.981	37.578

Table 3 The metrics result on NYU v2 dataset

Image of Scenes	JBF method			FCM method			TSR method			EDI method			Proposed method		
	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR	RMSE	SSIM	PSNR
Bedroom	2.083	0.976	36.313	2.430	0.943	27.761	1.930	0.974	31.508	3.221	0.945	32.249	2.498	0.981	37.108
Sofa	0.376	0.993	45.439	0.933	0.991	46.021	0.813	0.982	46.558	0.656	0.982	38.165	0.504	0.996	51.367
Livingroom	1.747	0.971	35.103	2.120	0.953	34.794	1.851	0.970	33.397	2.490	0.952	33.609	1.563	0.981	37.086
Average of 10 images	1.091	0.986	42.465	1.272	0.978	41.591	1.135	0.976	37.854	1.470	0.974	39.348	0.998	0.989	44.573

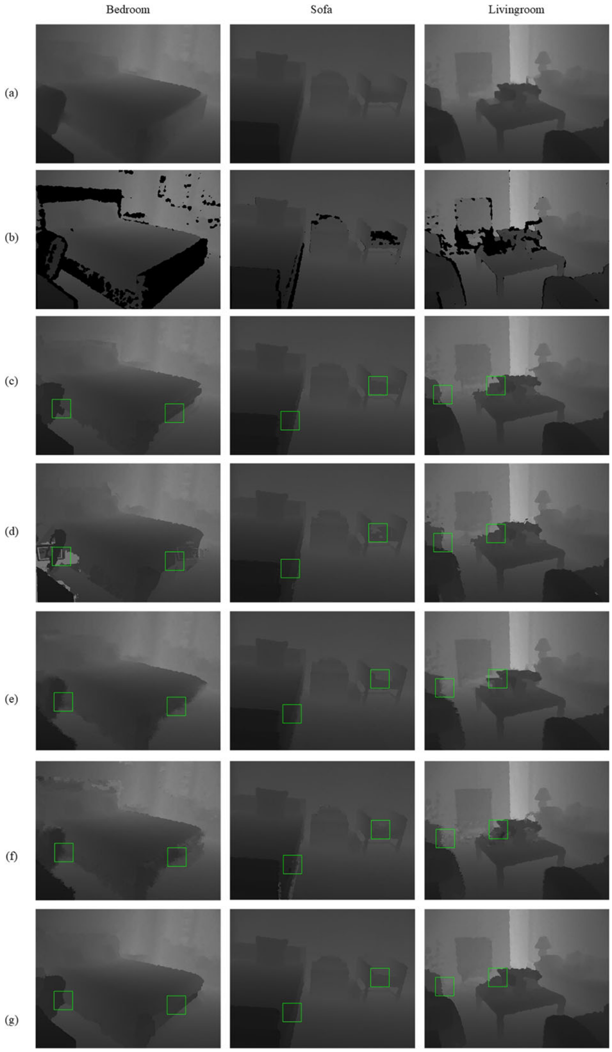

Fig. 6 Comparison of the fixing results on NYU v2 dataset. Row (a) depth images (Ground truth), Row (b) raw depth images , Row (c)Fixing results of the JBF method , Row (d) Fixing results of the FCM method , Row (e) Fixing results of the TSR method, Row (f) Fixing results of the EDI method, Row (g) Fixing results of the proposed method

In the Livingroom scene, the holes are mainly around the edges of the table and the sofa. The result of the JBF method has some obvious errors in the selected area. The result of the FCM method also has irregular edges in the selected area on the left, which is inconsistent with the ground truth image. Both the results of TSR method and the EDI method have blurred edges of sofa and television, and the result of the EDI method is more ambiguous.

The proposed method is partially different from the ground truth on the select table area, but the result has clear object edges, and the visual effect is good.

The PSNR,RMSE and SSIM results of each method are shown in Table 3.

From the table, we can see that the proposed method can achieve better PSNR and SSIM both in average and on the displayed images, although its RMSE parameter performance in the Bedroom scene fails to surpass the TSR method, and in the Sofa scene its RMSE performance fails to outperform the JBF method, but its average RMSE is better than other methods. Overall, the results restored by proposed method have a better performance on the dataset obtained by Kinect device. Based on the experiment results, the effectiveness of the proposed method is fully proved.

6 Conclusion

Kinect is widely used in depth image acquisition due to its low price and real-time performance, while owing to occlusion or measurement range limitation, the depth images obtained by Kinect have holes that greatly influence the subsequent application. In order to improve the quality of depth image, an NLM-based fixing algorithm for Kinect depth image is proposed in this paper. Different from the previous filter-based methods, the proposed method uses the NLM algorithm to take the value vector centered on the hole into the weight calaulation, and the vector contains both the gray value information and the texture information around the hole. In this way, the proposed method keeps robust and estimates the depth value with high accuracy.

We compared the proposed method with the other four efficient methods. The JBF method can quickly restore the holes in the depth image through the bilateral filter, but it is not effective for fixing the large-scale holes, wrong depth values often appear in its result. The FCM method fixes the holes through fuzzy clustering, the clustering effect is unstable which is easy to produce illogical repair results. The TSR method is based on the exemplar-based method, blurred edges are often shown in its results, so the visual effect is vague. The EDI method is also based on the same method as TSR, but the repair effect is worse, and sometimes large-scale errors occur in its results. The proposed method fixes the holes by NLM, the errors are not often occurred in its results, and the edges of objects are clear. Also, the proposed algorithm can achieve better average PSNR and RMSE than other methods in Middlebury dataset and NYU v2 dataset, its average SSIM results are close to the best SSIM.

In general, the proposed method achieves optimal results in terms of both quantitative and qualitative results.

Acknowledgements This work was supported in part by Key Research and Development Program of Shaanxi under Grant 2020KW-010, in part by National Natural Science Foundation of China under Grant 61801384 and Grant 61971343, in part by Natural Science Basic Research Plan in Shaanxi Province of China under Grant 2020JM-415, and in part by Northwest University Paleontological Bioinformatics Innovation Team under Grant 2019TD-012.

Declarations

Conflict of Interests All authors declare that there is no conflict of interest regarding the publication of this article.

References

1. Achanta SDM, Karthikeyan T, Vinothkanna RA (2019) Novel hidden Markov model-based adaptive dynamic time warping (HMDTW) gait analysis for identifying physically challenged persons. Soft Comput 23:8359–8366
2. Behroozpour B, Sandborn PAM, Wu MC, Boser BE (2017) Lidar system architectures and circuits. IEEE Commun Mag 55:135–142
3. Buades A, Coll B, Morel JM (2005) Image denoising by non-local averaging. In: Proceedings. (ICASSP '05). IEEE International conference on acoustics, speech, and signal processing, vol 2, pp 25–28
4. Cai M, Wang Y, Wang S, Wang R, Tan M (2019) ROS-based depth control for hybrid-driven underwater vehicle-manipulator system. In: 2019 Chinese Control Conference (CCC), pp 4576–4580
5. Chang T-A, Liao W-C, Yang J-F (2018) Robust depth enhancement based on texture and depth consistency. IET Signal Proc 12(1):119–128
6. Chen M, Chiang C, Lu Y (2014) Depth estimation for hand-held light field cameras under low light conditions. In: 2014 International conference on 3D imaging (IC3d), pp 1–4
7. Chen Z, Wang H, Wu L, Zhou Y, Wu D (2020) Spatiotemporal guided self-supervised depth completion from liDAR and monocular camera. In: 2020 IEEE International conference on visual communications and image processing (VCIP), pp 54–57
8. Cho J, Park S, Chien S (2020) Hole-filling of RealSense depth images using a color edge map. IEEE Access 8:53901–53914
9. Criminisi A, Perez P, Toyama K (2004) Area filling and object removal by exemplar-based image inpainting. IEEE Trans Image Process 13(9):1200–1212
10. Feng C, Dai SL (2014) Adaptive depth map enhancement based on joint bilateral filter. In: Proceedings of 2014 IEEE Chinese guidance, navigation and control conference, pp 2568–2571
11. Hirschmuller H, Scharstein D (2007) Evaluation of cost functions for stereo matching. In: 2007 IEEE Conference on computer vision and pattern recognition, pp 1–8
12. Hung S, Lo S, Hang H (2019) Incorporating luminance, depth and color information by a fusion-based network for semantic segmentation. In: 2019 IEEE International conference on image processing (ICIP), pp 2374–2378
13. Jayachandran A, Preetha VH (2016) Application of exemplar-based inpainting in depth image based rendering. In: 2016 IEEE International conference on recent trends in electronics, information & communication technology (RTEICT), pp 1117–1121
14. Jing N, Ma X, Guo W (2018) 3D reconstruction of underground tunnel using Kinect camera. In: 2018 International symposium on computer, consumer and control (IS3C), pp 278–281
15. Lee G-W, Han J-K (2021) Hole concealment for depth image using pixel classification in multiview system. In: 2021 IEEE International conference on consumer electronics (ICCE), pp 1–5
16. Liang C, Su S, Chen M (2017) Non-pre-process calibration of depth image based on fuzzy c-mean. In: 2017 International conference on system science and engineering (ICSSE), pp 125–128
17. Liao X, Zhang X (2017) Multi-scale mutual feature convolutional neural network for depth image denoise and enhancement. In: 2017 IEEE Visual communications and image processing (VCIP) 1–4
18. Nguyen T-D, Kim B, Hong M-C (2019) New Hole-Filling method using extrapolated Spatio-Temporal background information for a synthesized Free-View. IEEE Trans Multimed 21(6):1345–1358
19. Pan L, Dai Y, Liu M, Porikli F (2018) Depth map completion by jointly exploiting blurry color images and sparse depth maps. In: 2018 IEEE winter conference on applications of computer vision (WACV), pp 1377–1386
20. Rossi M, Gheche ME, Kuhn A, Frossard P (2020) Joint graph-based depth refinement and normal estimation. In: 2020 IEEE/CVF Conference on computer vision and pattern recognition (CVPR), pp 12151–12160
21. Scharstein D, Hirschmuller H, Kitajima Y, Krathwohl G, Nesic N, Wang X, Westling P (2014) High-resolution stereo datasets with subpixel-accurate ground truth Pattern Recognition. GCPR 2014(8753):31–42
22. Sen X, Huiping D, Lei Z, Jin W, Li Y (2019) Exemplar-based depth inpainting with arbitrary-shape patches and cross-modal matching. Signal Process: Image Commun 71:56–65
23. Sharma N, Achanta SDM, Karthikeyan T, Kumari CU, Jagan BOL (2020) Gait diagnosis using fuzzy logic with wearable tech for prolonged disorders of diabetic cardiomyopathy Gait diagnosis using fuzzy logic with wearable tech for prolonged disorders of diabetic cardiomyopathy: Materials Today. Proceedings
24. Song Z (2018) High-speed 3D shape measurement with structured light methods: A review. Opt Lasers Eng 106:119–131

25. Song W, Le AV, Yun S, Jung S-W, Won CS (2017) Depth completion for kinect v2 sensor. Multimed Tools Appl 76(3):4357–4380
26. Sun M-J, Zhang J-M (2019) Single-pixel imaging and its application in three-dimensional reconstruction: A brief review. Sensors 19(3):732
27. Xiaodong B, Bailin Y, Jia Z, Tianxiang W, Yiming X (2021) A novel holes filling method based on layered depth map and patch sparsity for complex-scene images. Microelectron J 114:105140
28. Zhong Y, Pei Y, Li P, Guo Y, Ma G, Liu M, Bai W, Wu W, Zha H (2021) Depth-based 3D face reconstruction and pose estimation using shape-preserving domain adaptation. IEEE Trans Biom Behav Identity Sci 3:6–15
29. Zhou H, Li Y, Tian X, Li X, Zhang XA (2015) Robust iterative nonlocal means method for electrocardiogram signal denoising. In: 2015 IET International conference on biomedical image and signal processing (ICBISP) 2015, pp 1–5
30. Xin J, Yatong X, Qionghai D (2016) Depth dithering based on texture edge-assisted classification. Signal Process: Image Commun 47:56–71
31. Zhang L, Xia H, Qiao Y (2020) Texture synthesis repair of RealSense D435i depth images with object-oriented RGB image Segmentation. Sensors 20(23):6725

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.