Article

Depth Image Denoising Algorithm Based on Fractional Calculus

Tingsheng Huang [1,2], Chunyang Wang [1,2,*] and Xuelian Liu [2,*]

1 School of Electronics and Information Engineering, Changchun University of Science and Technology, Changchun 130022, China; huangtingsheng99@foxmail.com
2 Xi'an Key Laboratory of Active Photoelectric Imaging Detection Technology, Xi'an Technological University, Xi'an 710021, China
* Correspondence: wangchunyang19@163.com (C.W.); tearlxl@126.com or liuxuelian@xatu.edu.cn (X.L.); Tel.: +86-135-7889-8897 (C.W.); +86-135-7879-6534 (X.L.)

Abstract: Depth images are often accompanied by unavoidable and unpredictable noise. Depth image denoising algorithms mainly attempt to fill hole data and optimise edges. In this paper, we study in detail the problem of effectively filtering the data of depth images under noise interference. The classical filtering algorithm tends to blur edge and texture information, whereas the fractional integral operator can retain more edge and texture information. In this paper, the Grünwald–Letnikov-type fractional integral denoising operator is introduced into the depth image denoising process, and the convolution template of this operator is studied and improved upon to build a fractional integral denoising model and algorithm for depth image denoising. Depth images from the Redwood dataset were used to add noise, and the mask constructed by the fractional integral denoising operator was used to denoise the images by convolution. The experimental results show that the fractional integration order with the best denoising effect was $-0.4 \leq \nu \leq -0.3$ and that the peak signal-to-noise ratio was improved by +3 to +6 dB. Under the same environment, median filter denoising had -15 to -30 dB distortion. The filtered depth image was converted to a point cloud image, from which the denoising effect was subjectively evaluated. Overall, the results prove that the fractional integral denoising operator can effectively handle noise in depth images while preserving their edge and texture information and thus has an excellent denoising effect.

Keywords: depth image; fractional integral; denoising; fractional integral operator; fractional integral denoising operator

1. Introduction

The current, rapidly developing era of artificial intelligence, intelligent driving, and 3D reconstruction research is founded on the availability of high-precision, high-quality depth data. Depth data are represented in depth images, which are acquired through methods such as stereo matching, lidar, and depth camera-based imaging. Although depth cameras can acquire images in real time, most depth camera images are highly sensitive to environmental factors that introduce noise, producing images with low resolution and quality. Removing the noise in these images can greatly improve super-resolution reconstruction. Research has imitated the degradation process of Kinect to generate depth images for testing [1]. For prospective applications in depth image super-resolution reconstruction and 3D reconstruction [2,3], depth image noise processing forms a crucial pre-link [4–11] and has been the focus of research on effective depth image noise processing algorithms.

Currently, depth image denoising algorithms mainly have two modes: non-colour image auxiliary filtering and colour image auxiliary filtering. References [4–7] proposed the use of the ordinary image filter method to denoise depth images, an approach that repairs the depth map image but with the loss of much edge information. References [8–12] reported on a denoising method that uses the image median filter to filter the pixel values in

the depth image and then superimposes the image enhancement. Reference [12] proposed depth image denoising by combining the colour image auxiliary guide filter to smooth the depth image; unfortunately, this restoration approach destroys part of the original depth values. Reference [13] proposed a denoising method that uses deep learning to improve the quality of depth images. However, in this approach, the denoising effect is evaluated solely visually, as the focus is on filling in holes in the depth image to improve target recognition and detection based on depth images. Detection and denoising algorithms can also blur the texture information of depth images. In summary, no optimal depth image processing method has yet been developed, as the numerous available methods are each lacking in some aspect(s). In particular, in the depth image filtering algorithms guided by achromatic images, the edge smoothing process destroys some valid and useful data. The algorithms can remove noise to varying degrees, but they often also remove information such as edge and texture details, which are critical components of depth images. The root cause of this problem is that the denoising operator constructed using the integer-order integral severely reduces the high-frequency information in the image, resulting in the loss of much of this critical information in the denoised image. The application of fractional calculus to image processing has been studied for several years [13–17]. For example, reference [18] successfully applied fractional calculus to the shadow detection of depth images, demonstrating how fractional calculus can be applied to depth images. Although the integer-order filtering algorithm has limitations, it can be optimised by introducing a fractional-order integral model, which is a significant development in the field of depth image denoising algorithms.

Based on the characteristics of fractional calculus, Section 2 deduces the fractional-order integral operator suitable for image processing and constructs a fractional-order integral denoising operator.

Section 3 compares the processing effects of the fractional integral filtering algorithm and median filtering. Simulations show that fractional integral filtering has a stronger noise filtering effect and that it retains more texture information than does median filtering. Comparisons of the proposed algorithm with the classical image denoising algorithms (median filtering and Gaussian filtering) show that the classical algorithms have a relatively high distortion degree. The algorithms are also compared with respect to the distortion of the original noise image, focusing on the distortion of the data point cloud image after Gaussian filtering; the results show that after Gaussian filtering, the degree of distortion is larger by approximately 15–30 dB. Therefore, only the median filter is used as a reference comparison.

2. Fractional Calculus Operator Denoising Theory and Method

Integer-order calculus can describe phenomena very well, whereas fractional-order calculus can observe the world from another perspective and obtain information that integer-order calculus cannot. Integer-order calculus has a clear physical meaning and geometric interpretation, whereas attributing clear physical and geometric meanings to fractional-order calculus has been difficult. Scientists have comprehensively examined this problem from different perspectives and have obtained a diverse set of results. Most of these results are from special cases of fractional calculus, which limits their wider applicability. Podlubny [19] first proposed a physical explanation of fractional calculus in 2001. In physics, personal time τ and cosmic time T are two distinct concepts. Universal time is uniform and equally spaced elapsed time (absolute time), whereas personal time is non-uniform time (relative time). Podlubny asserted that fractional differentiation corresponds to the physical phenomenon of τ observed from the perspective of T.

Fractional calculus is suitable for describing some physical phenomena that cannot be fully described by integer calculus. The description of fractional calculus covers a wide range of physical fields—for example, anomalous diffusion, random walks, viscoelastic dynamics, automatic control PID, signal processing, neural networks, chaotic systems, and image processing.

2.1. Effect of Fractional Integration on Signal and Model Construction

For the energy signal $f(x) \in L^2(R)$ of any square area, the Fourier transform is derived from the basic theory of signal processing as

$$F(\omega) = \int_R f(x)e^{-i\omega x}dx. \tag{1}$$

Assuming the fractional v-th order derivative of the signal $f^v(x)(v \in R)$, according to the Fourier transform, we get

$$D^v f(x) = \overline{D}^v(\omega) \cdot \overline{F}^v(\omega), \tag{2}$$

In

$$\overline{D}^v(\omega) = (i\omega)^v. \tag{3}$$

In Figure 1, the fractional differential operator is an amplitude–frequency curve in the range of $[-2, 2]$. Order $v = 0$ means that the signal does not change. When the order is less than zero ($-2 < v < 0$), the signal represents an integral operation. The figure shows that the signal can be nonlinearly attenuated, and the amplitude of this attenuation is related to the differential order. When the order is greater than zero ($0 < v < 2$), the signal represents a differential operation with enhanced nonlinearity: the greater the differential order, the greater the enhancement amplitude. This feature of fractional-order calculus is applied in the image field. To achieve more prominent image edges and retain the texture information of the smooth area of the image, the fractional derivative can be used to improve the high-frequency components while also nonlinearly retaining the special characteristics of the low-frequency components of the signal. When it is instead necessary to denoise the image while retaining much of the edge and texture information, the opposing characteristics of fractional integration and differentiation can be used to nonlinearly retain the highest frequency components while attenuating the low-frequency component of the signal. Integral image denoising can thus achieve image denoising while preserving as much image edge and texture information as possible, as shown in Figure 1.
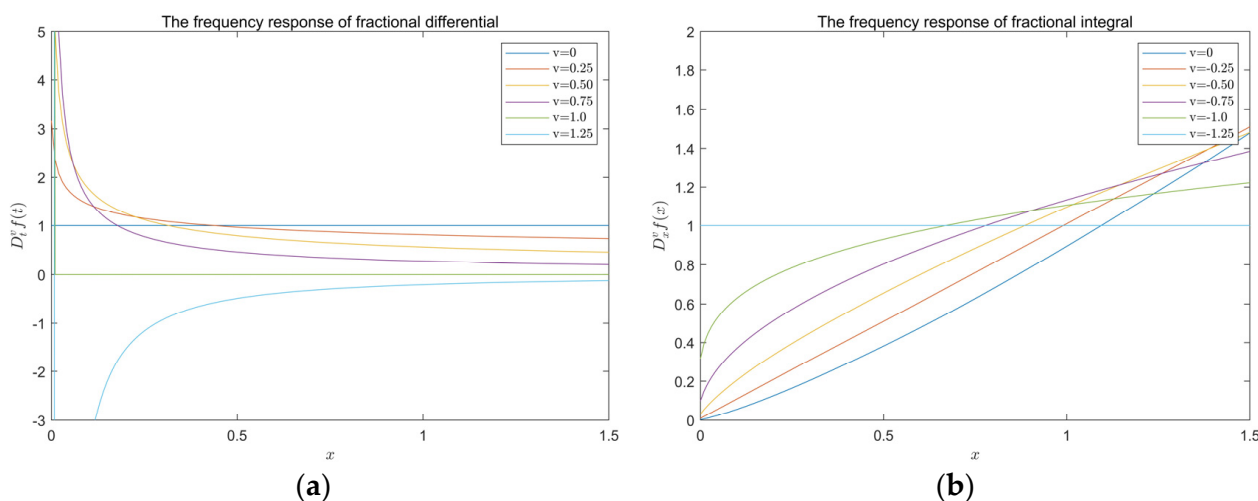
Figure 1. Amplitude−frequency characteristic curve of fractional calculus. (**a**) Fractional differential amplitude−frequency characteristic curve and (**b**) fractional-order integral amplitude-frequency characteristic curve.

2.2. Construction Based on Fractional Integral Operator

The Grünwald–Letnikov (G-L) definition of fractional calculus is

$$
{}_a^G D_x^v = \lim_{h \to 0} \frac{1}{h^v} \sum_{n=0}^{\frac{t-a}{h}} -1^j \frac{\Gamma(v+1)}{j!\Gamma(v-j+1)} f(x-jh), v \in R^+,
\tag{4}
$$

where v is the fractional order, h is the calculus step size, $[t-a/h]$ is the integer part of the variable $(t-a)/h$, a is the lower limit of the fractional calculus, t is the upper limit of the fractional calculus, and $\Gamma(n) = \int_0^\infty e^{-t} t^{n-1} dt = (n-1)!$ is the Gamma function.

The duration interval of the unary signal $f(t)$ is $[a, t]$, divided equally according to the unit $h = 1$, and $m = [(t-a)/h]^{h=1} = [t-a]$. Therefore, the expression of the first-order calculus v of the unary signal is

$$
\frac{d^v t}{dt^v} \approx f(t) + (v)f(t-1) + \frac{(v)(v+1)}{2} f(t-2) + \cdots + \frac{\Gamma(v+m)}{\Gamma(m+1)\Gamma(v)} f(t-m).
\tag{5}
$$

As the fractional calculus becomes larger, it uses the separability of the Fourier transform to define the extension of the fractional calculus from one dimension to two. By dividing the signal into equal parts $I(x, y)$ according to the unit time $h = 1$, the fractional differential formula of the *x*-axis and *y*-axis can be obtained. From Equation (5), the approximation of the partial fractional differential defined by Grünwald–Letnikov can be obtained, so the numerical calculation expression of the fractional integral operator must be solved along both the *x*- and *y*-directions. These are defined as

$$
\frac{d^v I(x, y)}{dx^v} = \lim_{N \to \infty} \left[\sum_{m=0}^{N-1} \frac{\Gamma(v+m)}{\Gamma(m+1)\Gamma(v)} I(x-m, y) \right],
\tag{6}
$$

$$
\frac{d^v I(x, y)}{dy^v} = \lim_{N \to \infty} \left[\sum_{m=0}^{N-1} \frac{\Gamma(v+m)}{\Gamma(m+1)\Gamma(v)} I(x, y-m) \right].
\tag{7}
$$

From Equations (6) and (7), the fractional integral mask coefficients W_m^v can be obtained for v, namely,

$$
W_m^v = \frac{\Gamma(v+m)}{\Gamma(m+1)\Gamma(v)}.
\tag{8}
$$

Setting $N = 3$ means that the size of the convolution 3×3 template is

$$
\frac{d^v I(x, y)}{dx^v} \approx I(x, y) + (v)I(x-1, y) + \frac{(v)(v+1)}{2} I(x-2, y),
\tag{9}
$$

$$
\frac{d^v I(x, y)}{dy^v} \approx I(x, y) + (v)I(x, y-1) + \frac{(v)(v+1)}{2} I(x, y-2).
\tag{10}
$$

To make the integral convolution template of the image have rotation invariance, Equations (8) and (9) are extended to the remaining six directions, and the fractional-order integral operator filter for eight directions can be obtained.

In Figure 2, the mask coefficients are

$$
\begin{aligned}
a_0^v &= 1 \\
a_1^v &= v \\
a_2^v &= \frac{v(v+1)}{2}
\end{aligned}.
\tag{11}
$$

a_2^v	0	a_2^v	0	a_2^v
0	a_1^v	a_1^v	a_1^v	0
a_2^v	a_1^v	$8a_0^v$	a_1^v	a_2^v
0	a_1^v	a_1^v	a_1^v	0
a_2^v	0	a_2^v	0	a_2^v

Figure 2. Fractional integral operator.

Define the positive and negative directions of the x-axis to have a^{90} and a^{180}, respectively, and the positive and negative directions of the y-axis to have a^0 and a^{180}, respectively. In the anti-clockwise direction, there exist a^{45}, a^{135}, a^{225}, and a^{315}. Then, use the 3×3 masks A^v in eight directions for the non-linear filtering of image points $I(x,y)$ of size 5×5 for convolution with computation.

$$I(x,y)^{\theta} = I(x,y) * A^{v}, \tag{12}$$

Linear weighting can be obtained according to the proportion of the convolution sums

$$
\begin{aligned}
(I(x,y)) = {} & \frac{I(x,y)^0}{sum(x,y)} \times I(x,y)^0 + \frac{I(x,y)^{45}}{sum(x,y)} \times I(x,y)^{45} \\
& + \frac{I(x,y)^{90}}{sum(x,y)} \times I(x,y)^{90} + \frac{I(x,y)^{135}}{sum(x,y)} \times I(x,y)^{135} \\
& + \frac{I(x,y)^{180}}{sum(x,y)} \times I(x,y)^{180} + \frac{I(x,y)^{225}}{sum(x,y)} \times I(x,y)^{225} \\
& + \frac{I(x,y)^{270}}{sum(x,y)} \times I(x,y)^{270} + \frac{I(x,y)^{315}}{sum(x,y)} \times I(x,y)^{315},
\end{aligned}
\tag{13}
$$

where

$$sum(x,y) = I(x,y)^0 + I(x,y)^{45} + I(x,y)^{90} + I(x,y)^{135} + I(x,y)^{180} + I(x,y)^{225} + I(x,y)^{270} + I(x,y)^{315}. \tag{14}$$

In the iterative process of the denoising algorithm, the $I(x,y)$ initial condition with noise is used, and the maximum peak signal-to-noise ratio (PSNR) of the image after iterative calculation is used as the termination condition for the iteration.

2.3. Fractional Integral Denoising Operator and Convolution Template for Constructing Depth Images

In the process of collecting depth images, the physical attributes of the equipment and the adaptability of algorithms can make depth images noisy. Depth images typically contain single-point noise, and the data values represent the depth of field data; therefore, the noise data cannot be directly observed in the depth image and need to be converted into a point cloud image for observation.

After noise is added to the acquired depth image, the image signal can be expressed as

$$y(i,j) = \begin{cases} u(i,j) \\ x(i,j) \end{cases}, \tag{15}$$

where $y(i,j)$ is the signal value of the noisy image, $x(i,j)$ is the original image signal value, and $u(i,j)$ is the image signal value with added noise.

The Gaussian algorithm is used to randomly add noise, allowing the image with noise to be compared against point cloud images, as shown in Figure 3.
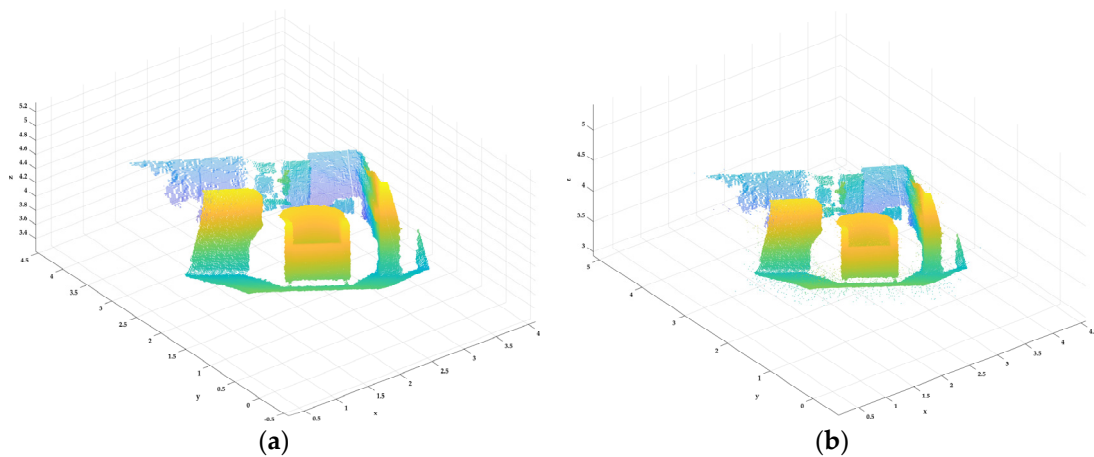
Figure 3. Effect of adding noise to the depth image. The point cloud image corresponding to the depth image (**a**) without and (**b**) with noise added.

From the characteristics of the depth image, the noise image (x, y) is represented by $u(x, y)$, where $u(x, y) = 0$. In the depth images, adjacent pixels share a certain level of similarity. For processing the noise, the feature information of $u(x, y)$ in the local neighbourhood of the target pixel can be used such that the reasonable depth value of the noise can be recovered using (x, y) and the depth value of the surrounding pixels. From Table 1 and Equation (11), the fractional-order mask of the order can be obtained (see ν in Figure 4).

Table 1. Fractional integral filter denoising results.

Dataset Number	V(Best)	PSNR(Best)	PSNR	PSNR
	Fractional Integral Denoising (dB)		Noise (dB)	Median Filter Denoising (dB)
05989	−0.3	52.992	47.307	37.480
03236	−0.3	52.070	47.599	36.107
03528	−0.4	53.847	50.094	37.855
02350	−0.3	57.391	54.611	36.309
09860	−0.4	51.289	43.457	37.326
08343	−0.3	50.870	46.934	32.112

Figure 4. Fractional integral denoising operator.

To reduce unnecessary space and time complexity, the feature information of the local neighbourhood of the target pixel should be fully utilised, as pixels closer to the central target have a higher level of similarity with the target. Therefore, the surrounding depth

image values $y(i, j)$ can be used for filtering, and the fractional integral normalisation factor q can be constructed from Figure 5 as

$$q = \sum_{\theta=0}^{360} I(x, y)^{\theta} \; (\theta = 0, 45, 90, 135, 180, 215, 270, 315). \tag{16}$$
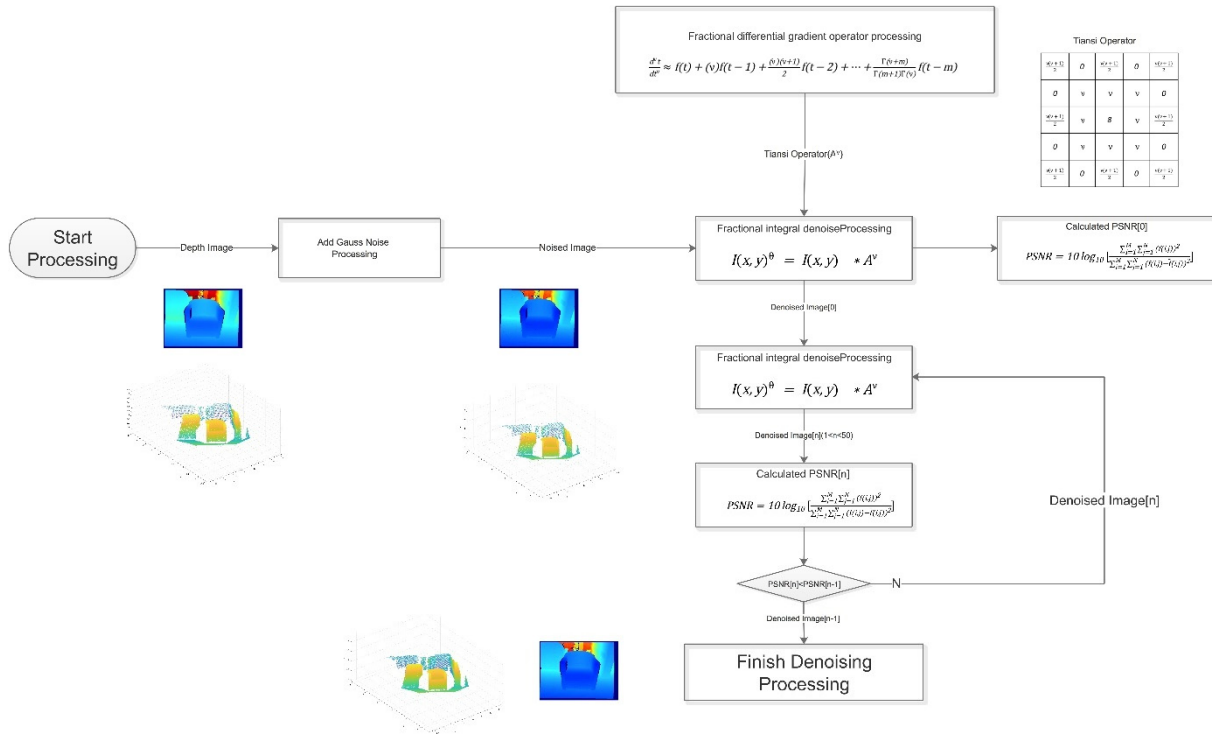

Figure 5. Fractional integral operator denoising flow chart.

By importing q into Equation (9), a new $I(x, y)$ after v-order fractional integral filtering with a 3×3 mask is obtained as

$$\overline{I(x, y)^{v}} = \sum_{\theta=0}^{360} \frac{I(x, y)^{\theta}}{q} * I(x, y)^{\theta} (\theta = 0, \; 45, \; 90, \; 135, \; 180, \; 215, \; 270, \; 315). \tag{17}$$

2.4. Fractional Integral Operator Denoising Algorithm Flow for Depth Images

Depth images are ordered data; hence, parallel computing can be used for computational convolution, thus ensuring efficient computation. The depth image has data orderliness and surrounding correlation, so the noise data can be processed using the surrounding valid data. The fractional integral de-noising algorithm is verified as follows (see Figure 5 for the flow chart):

First, the depth image is obtained from the Redwood dataset [20].

Second, random Gaussian noise is added to the depth image for iterative denoising processing.

Finally, the maximum PSNR of each denoising iteration is determined after the iterative algorithm has been run and terminated.

3. Experimental Results and Analysis

3.1. Evaluation Method of Depth Image Denoising Effect

The proposed depth map filtering algorithm is tested using images from an open-source dataset. Specifically, random noise is added to a depth image, and the results of the denoising algorithm are then evaluated using the PSNR. Considering the characteristics of the depth image and its point distance information, an objective evaluation standard can be used as an intuitive scheme for evaluating the denoising performance of the algorithm;

however, in many cases, this approach cannot reflect the actual denoising effect. Therefore, PSNR values are used as the objective evaluation criteria for evaluating the depth image denoising effect.

Assuming that the original depth image without noise $f(i,j)$ is the ground truth, the depth image to be evaluated is $\overline{f}(i,j)$, where ($0 < x < M$, $0 < y < N$)

$$PSNR = 10log_{10}\left[\frac{\sum_{i=1}^{M}\sum_{j=1}^{N}(f(i,j))^2}{\sum_{i=1}^{M}\sum_{j=1}^{N}\left(f(i,j) - \overline{f}(i,j)\right)^2}\right]. \tag{18}$$

Here, the *M* and *N* sub-tables are both the same width and height as the image.

3.2. Simulated Experiment and Analysis of Depth Image

Depth images obtained from the Redwood dataset [20] were used to verify the denoising effect of the proposed algorithm, following the process in Figure 5. The denoising effect of the proposed algorithm was also compared with that of the median filter algorithm. The simulated experimental results show that although the median filter algorithm performed well in terms of edge problems and the denoising ability, it resulted in the loss of key image information. The PSNR results also show that key information was lost, with the PSNR distorted by approximately -15 to -30 dB. These experimental results verify the effectiveness of the algorithm. In addition, we conclude that the effect of depth image denoising can be evaluated via two approaches: (1) visually aided subjective evaluation through 3D point cloud images, and (2) uploading data for objective PSNR evaluation, as shown in Figure 6.

(a)   **(b)**   **(c)**

(d)   **(e)**   **(f)**
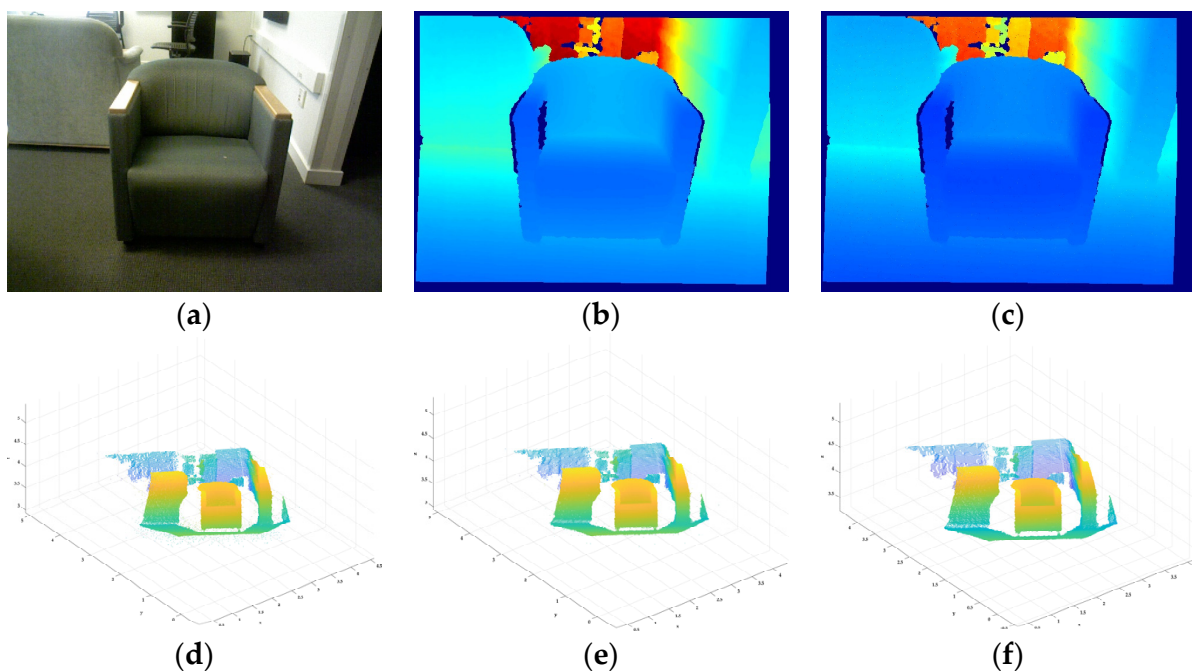
Figure 6. Denoising effect of fractional integral denoising and median filtering. (**a**) RGB image (dataset number 00033); (**b**) deep pseudo-colour map (dataset number 00033); (**c**) noise-free point cloud image; (**d**) point cloud image with added noise, PSNR = 46.58 dB; (**e**) point cloud image filtered by fractional integral denoising ($\nu = -0.3$, PSNR = 52.509 dB); and (**f**) point cloud of the depth image after median filtering and denoising (PSNR = 39.141 dB).

Figure 5 presents the algorithm verification process using data from the Redwood dataset [20] (dataset number 00033) with Gaussian noise added to the depth image, before using either the median filtering or fractional denoising algorithms. The fractional integral operator denoising algorithm also used different orders to verify the optimal order, and it used the PSNR to evaluate the optimal fractional denoising order, as shown in Figures 7 and 8; the simulated experiments yielded the best results when $v = -0.3$.
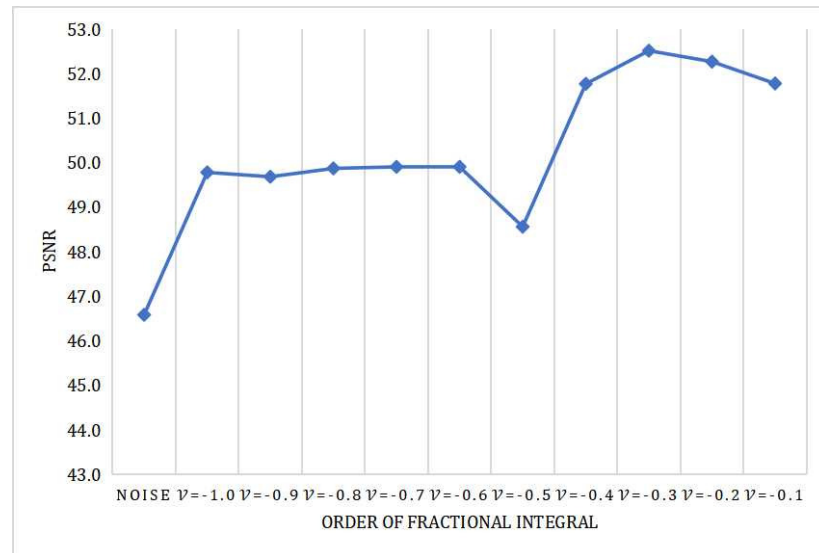
Figure 7. Fractional denoising effect with different orders of the fractional integral (PSNR, dB).
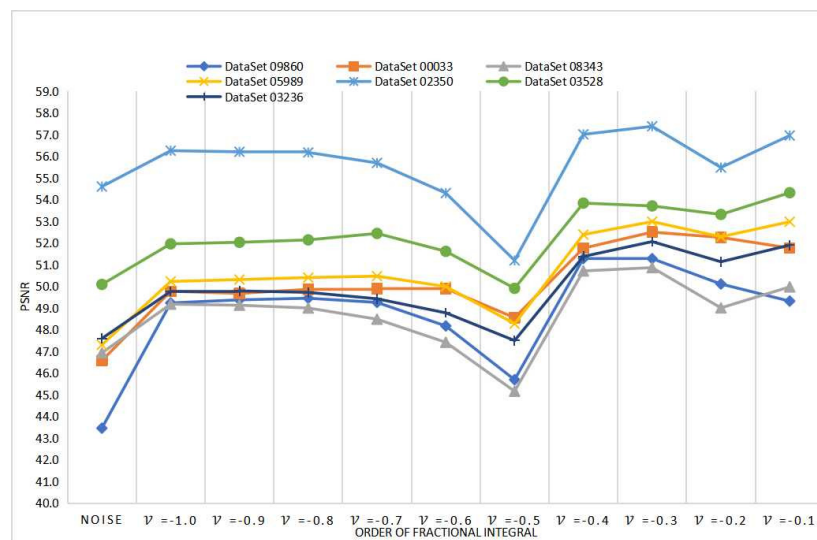

**Figure 8.** Optimal fractional integral filter denoising order.

The algorithm was further verified using more images and the optimal fractional integration order obtained from the experimental data. The six graph datasets of the Redwood dataset [20] were each used for fractional filter denoising. The filter order range was $-1.0 \leq v \leq -0.1$, as shown in Figure 9; the best denoising effect of the fractional order was achieved when $-0.4 \leq v \leq -0.3$.
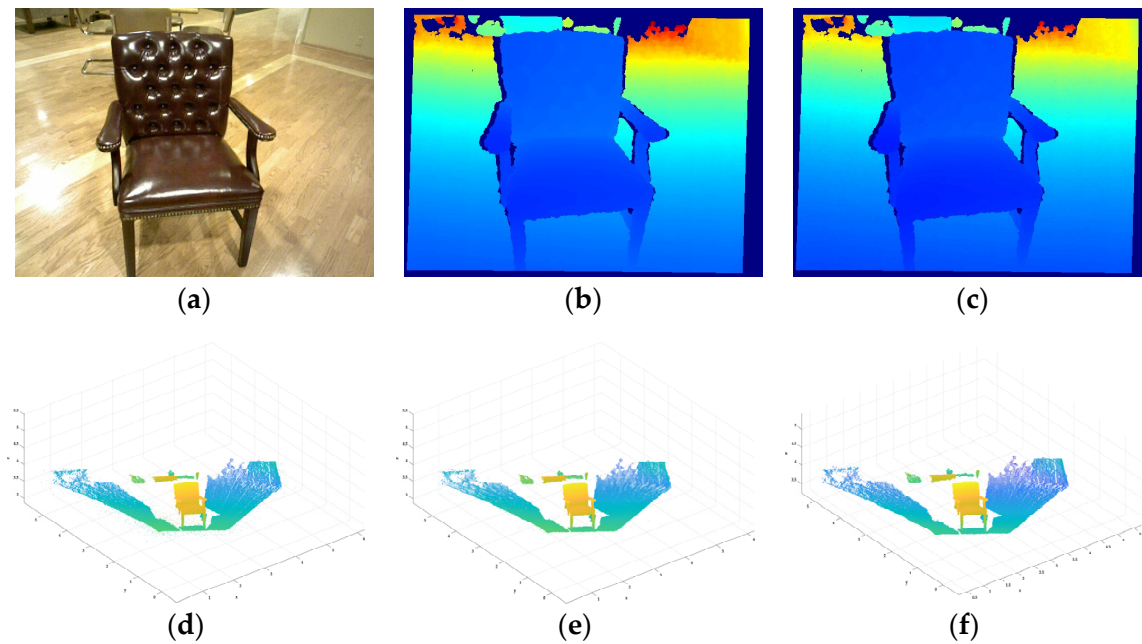
Figure 9. Denoising effects of fractional integral denoising and median filtering. (**a**) RGB image (dataset number 05959); (**b**) deep pseudo-colour map (dataset number 05989); (**c**) noise-free point cloud image; (**d**) point cloud image with added noise, PSNR = 47.307 dB; (**e**) point cloud image filtered by fractional integral denoising ($v = -0.3$, PSNR = 52.992 dB); and (**f**) point cloud of the depth image after median filtering and denoising (PSNR = 37.480 dB).

For these six datasets, either fractional integral operator denoising or median filter denoising were performed. The filtering effects are shown in Table 1.

The filtering effects for the best filtering order for all six datasets are shown in Figures 9–14, on the basis of which the effectiveness of the denoising algorithms can be evaluated.
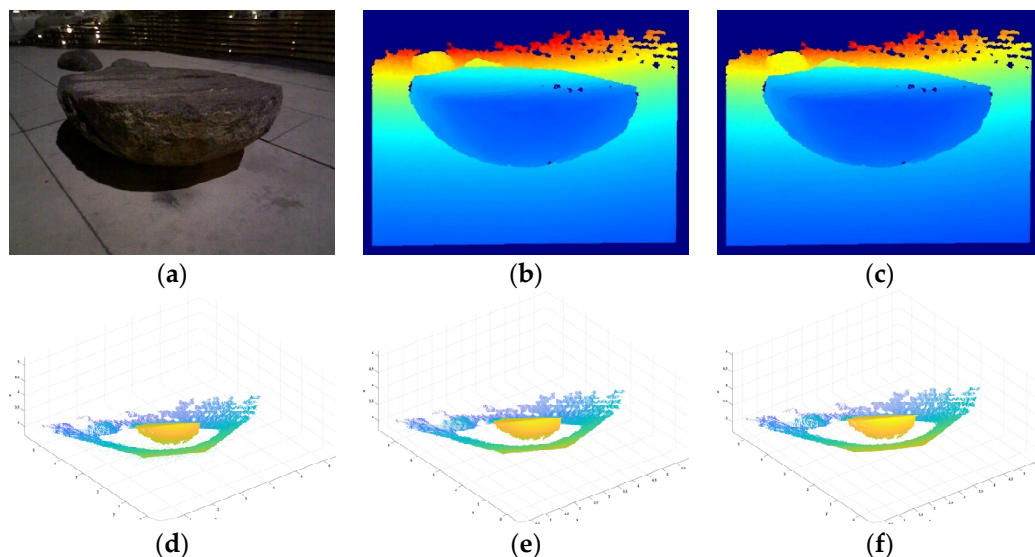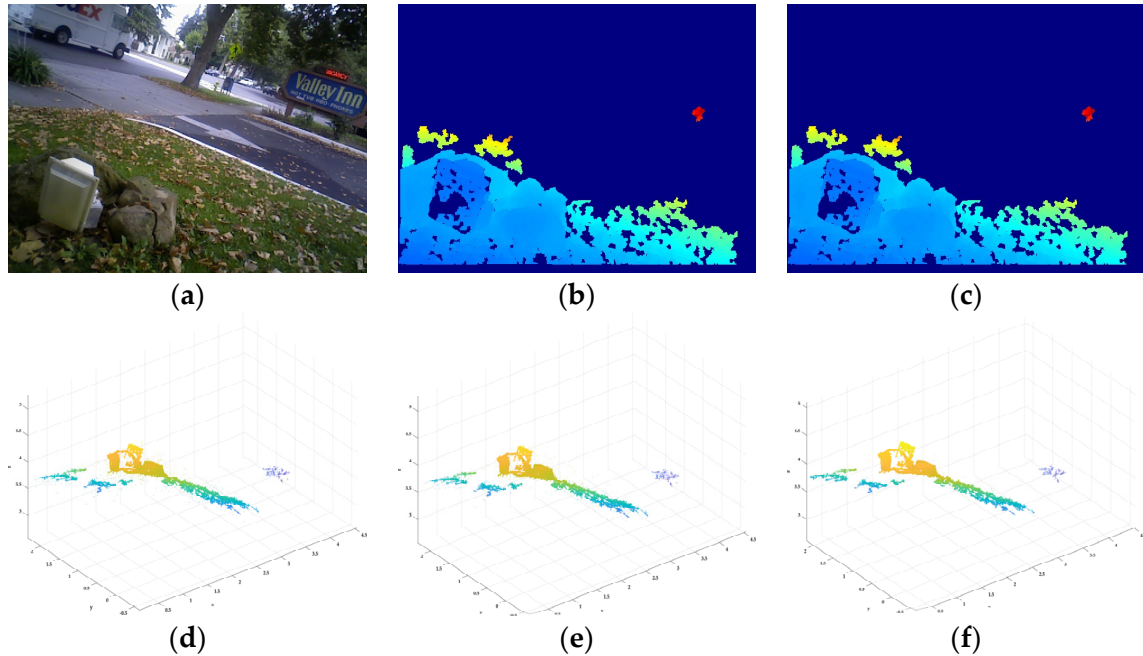

Figure 10. Denoising effects of fractional integral denoising and median filtering. (**a**) RGB image (dataset number 03236); (**b**) deep pseudo-colour map (dataset number 03236); (**c**) no noise point cloud image added; (**d**) point cloud image with added noise, PSNR = 47.599 dB; (**e**) point cloud image filtered by the fractional integral denoising ($v = -0.3$, PSNR = 52.070 dB); (**f**) point cloud of the depth image after median filtering and denoising (PSNR = 36.107 dB).
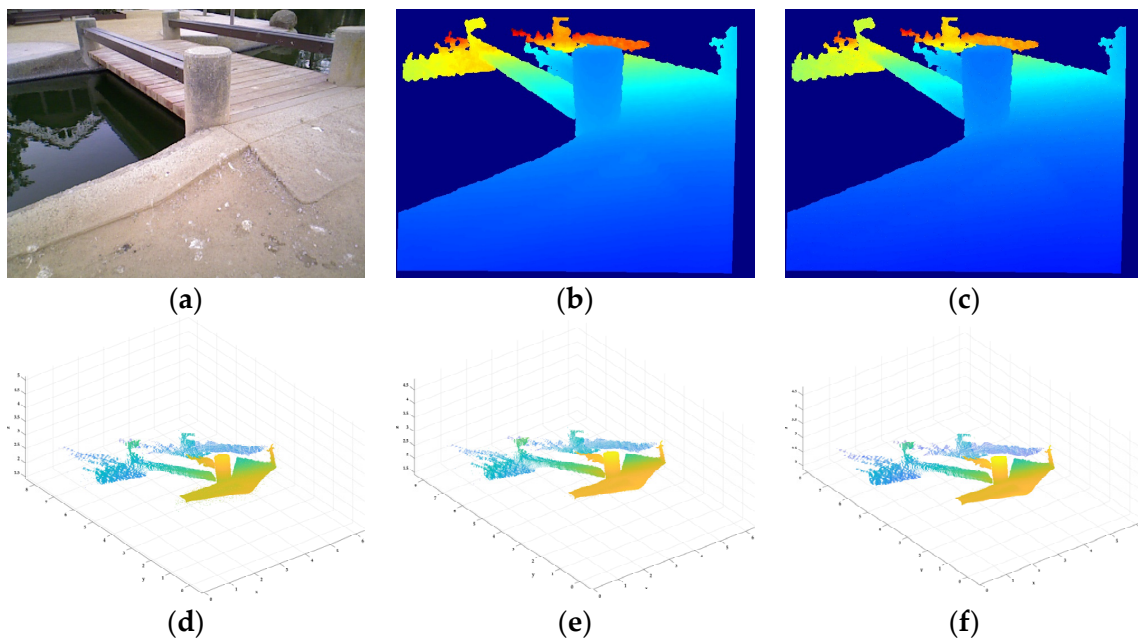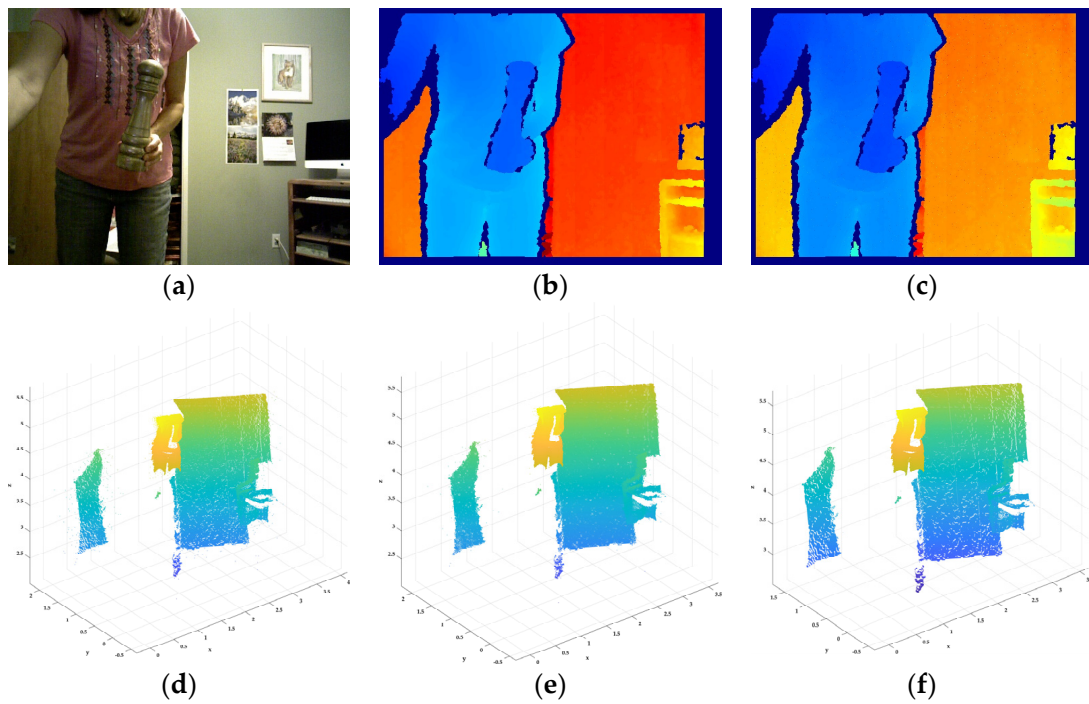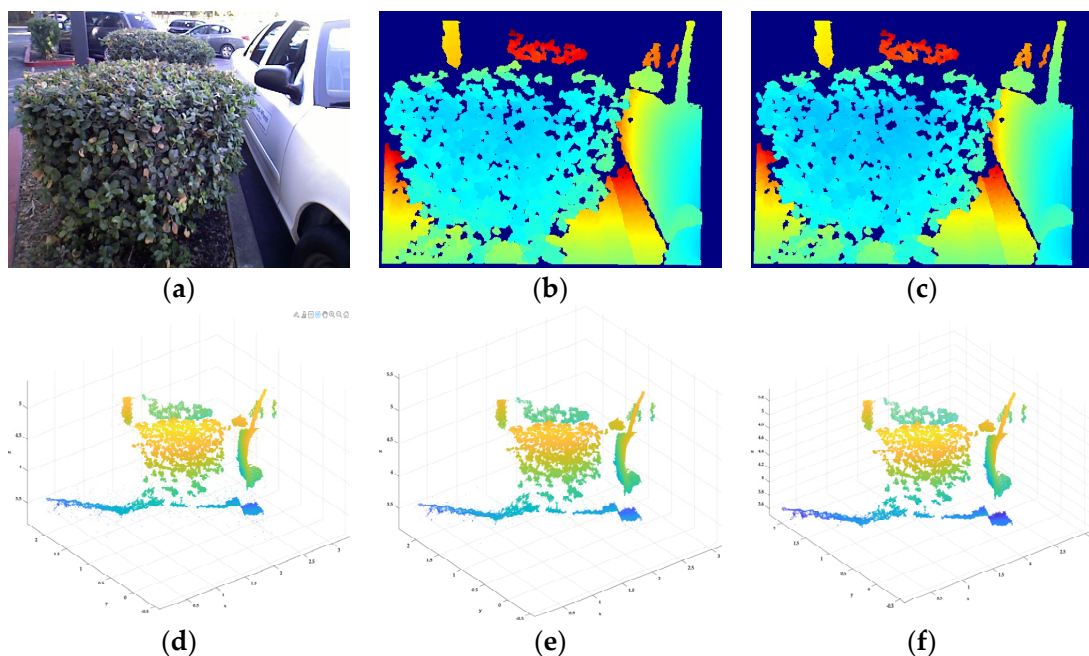
Figure 11. Denoising effects of fractional integral denoising and median filtering. (**a**) RGB image (dataset number 03528); (**b**) deep pseudo-colour map (dataset number 03528); (**c**) noise-free point cloud image; (**d**) point cloud image with added noise, PSNR = 50.094 dB; (**e**) point cloud image filtered by fractional integral denoising ($v = -0.4$, PSNR = 53.847 dB); and (**f**) point cloud of the depth image after median filtering and denoising (PSNR = 37.855 dB).

Figure 12. Denoising effects of fractional integral denoising and median filtering. (**a**) RGB image (dataset number 02350); (**b**) deep pseudo-colour map (dataset number 02350); (**c**) noise-free point cloud image; (**d**) point cloud image with added noise, PSNR = 54.611 dB; (**e**) point cloud image filtered by fractional integral denoising ($v = -0.3$, PSNR = 57.391 dB); and (**f**) point cloud of the depth image after median filtering and denoising (PSNR = 36.309 dB).

Figure 13. Denoising effects of fractional integral denoising and median filtering. (**a**) RGB image (dataset number 09860); (**b**) deep pseudo-colour map (dataset number 09860); (**c**) noise-free point cloud image; (**d**) point cloud image with added noise, PSNR = 43.457 dB; (**e**) point cloud image filtered by fractional integral denoising ($\nu = -0.4$, PSNR = 51.289 dB); and (**f**) point cloud of the depth image after median filtering and denoising (PSNR = 37.326 dB).
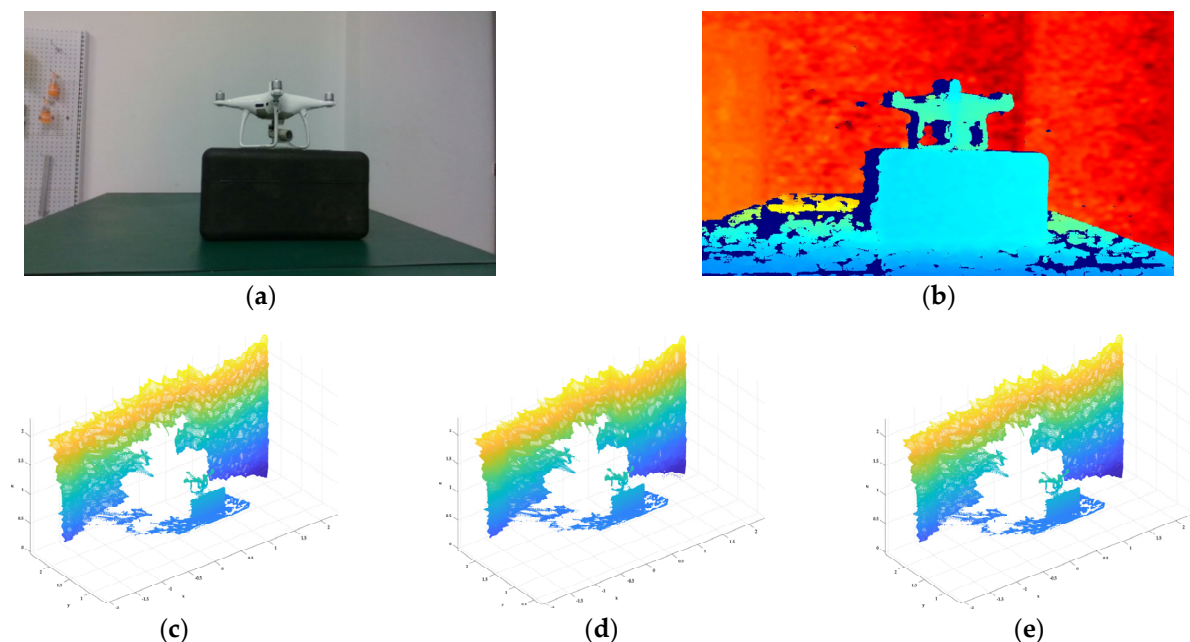

Figure 14. Denoising effects of the fractional integral denoising and median filtering. (**a**) RGB image (dataset number 08343); (**b**) deep pseudo-colour map (dataset number 08343); (**c**) noise-free point cloud image; (**d**) point cloud image with added noise, PSNR = 46.934 dB; (**e**) point cloud image filtered by fractional integral denoising ($\nu = -0.3$, PSNR = 50.870 dB); and (**f**) point cloud of the depth image after median filtering and denoising (PSNR = 32.112 dB).

These six experiments show that fractional integral denoising achieved a 3–6 dB improvement in the PSNR, with each experiment demonstrating a significant denoising effect. The denoised depth images were then converted into point cloud images for comparative analysis. The analysis revealed that a large amount of noise was effectively removed and that the remaining noise could be removed as invalid points. Overall, the results show that the fractional integral denoising operator performs well in filtering noise from depth images and can find significant applications in future research.

4. Laboratory Field Depth Image Denoising Effect

Our experiments were performed in a Python environment, with the Pyrealsense2 and OpenCV2 plugins, using a depth camera (Intel Realsense2 Depth Camera D435) to obtain depth images for verification. The operating system environment was Windows 10, and the CPU was AMD Ryzen 7 4800H. The experimental results prove the effectiveness of the proposed fractional-order denoising algorithm. In the experiment, ν was set to -0.3 for the fractional-order integral denoising operator to filter the acquired depth image, and five iterations were performed, as shown in Figure 15.

(**a**)

(**b**)

(**c**)

(**d**)

(**e**)

Figure 15. Denoising effects of fractional integral denoising and median filtering. (**a**) RGB image; (**b**) deep pseudo-colour map; (**c**) noise-free point cloud image; (**d**) point cloud image of the depth image filtered by the fractional integral denoising ($\nu = -0.3$, PSNR = 53.764 dB); and (**e**) point cloud of the depth image after median filtering and denoising (PSNR = 34.708 dB).

The denoising experiment results are shown in Table 2. The results show that the use of fractional integral denoising resulted in lower effective information loss and that the PSNR (55.734 dB) after fractional integral denoising was higher than that after median filtering. The PSNR (34.708 dB) of the classical filtering algorithm shows that the texture information and edges were well preserved. Table 2 shows that the proposed approach yielded more valid data than median filtering. Thus, the experimental results verify that the denoising effect of the fractional-order integral denoising algorithm is better than that of the integer-order classical filtering algorithm.

Table 2. Fractional integral denoising operator filtering results.

Data Set	PSNR (dB)	Valid Point (Normal)	Valid Point (After Statistical Outlier Removal)
Original depth image	100.00	854,500	831,126
Median filter denoising	34.708	855,799	831,126
Fractional integral denoising ($v = -0.3$)	53.764	854,575	841,309

5. Conclusions

In this paper, fractional calculus is introduced into the field of depth image processing, and a fractional integral denoising operator mask suitable for depth image processing is derived and constructed. The proposed depth image denoising algorithm implemented using fractional integral denoising was validated via simulation experiments on the Red-wood dataset [20]. The experimental results show that the algorithm fully exploited the characteristics of the fractional integral denoising algorithm to preserve texture information and edges and that it performed successively minor adjustments to noisy images through multiple iterations. Regarding the denoising effect, the fractional integral denoising operator clearly outperformed the integral denoising operator. However, a subjective evaluation of the denoising effect of the proposed approach after converting the denoised depth image into a point cloud revealed that some noises, especially noises with small amplitude, were not effectively removed and were preserved as texture information. Much remains to be explored and achieved in depth image denoising by fractional integral denoising. Further improving the proposed fractional order-based denoising algorithm is a future research direction, as is introducing the fractional integral to point cloud denoising. Moreover, to realise self-adaptive order fractional integral denoising, it is necessary to further study the relationship between different orders of the fractional integral and the surface texture.

Depth image processing is also a research hotspot in computer vision and image processing, where such processing not only has theoretical significance but also major practical applications. For example, in computer gaming, to capture a player's gestures and actions and realise human–computer interaction through somatosensory peripheral devices, the acquired depth images must be processed to enhance the quality of the depth data. Similarly, in 3D reconstruction, depth camera-derived depth images can be enhanced by the algorithm proposed in this paper. Furthermore, super-resolution reconstruction of depth images has applications in visual fields such as interactive gaming, biomedicine, and augmented reality, and these applications can benefit from highly accurate depth image data. For instance, high-accuracy super-resolution reconstruction can yield highly realistic 3D object surface models. In addition, both artificial intelligence and autonomous driving require highly accurate depth image data. The foregoing applications are all expected to make the exploration of highly effective processing algorithms a research hotspot.

References

1.   Zou, Y.; Wu, Q.; Zhang, J.; An, P. Explicit Edge Inconsistency Evaluation Model for Color-Guided Depth Map Enhancement. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *28*, 439–453.
2.   Truong, A.M.; Vealer, P.; Philips, W. Depth Map Inpainting and Super-Resolution with Arbitrary Scale Factors. In Proceedings of the 2020 IEEE International Conference on Image Processing (ICIP), Abu Dhabi, United Arab Emirates, 25–28 October 2020; pp. 488–492.
3.   Zhu, F.; Li, S.; Guo, X. A 3D reconstruction method based on RGB-D camera. *J. Phys. Conf. Ser.* **2021**, *1802*, 42–48. [CrossRef]
4.   Kui, W.; Ping, A.; Zhaoyang, Z.; Hao, C.; Hejian, L. Fast Inpainting Algorithm for Kinect Depth Map. *J. Shanghai Univ.* **2012**, *18*, 454–458.
5.   Li, P.; Pei, Y.; Zhong, Y.; Guo, Y.; Zha, H. Robust 3D face reconstruction from single noisy depth image through semantic consistency. *IET Comput. Vis.* **2021**, *15*, 393–404. [CrossRef]
6.   Li, Z.; Chen, Y. Kinect depth image filtering algorithm based on joint bilateral filter. *J. Comput. Appl.* **2014**, *34*, 2231–2234.
7.   Liu, J.; Wu, W.; Chen, C.; Wang, G.; Zeng, C. Depth image inpainting method based on pixel filtering and median filtering. *J. Optoelectron. Laser* **2018**, *29*, 539–544.
8.   Tan, Z.; Ou, J.; Zhang, J.; He, J. A Laminar Denoising Algorithm for Depth Image. *Acta Opt. Sin.* **2017**, *37*, 0510002.
9.   Min, D.; Lu, J.; Do, M.N. Depth Video Enhancement Based on Weighted Mode Filtering. *IEEE Trans. Image Process.* **2012**, *21*, 1176–1190. [PubMed]
10.   Liu, J.; Gong, X. Guided Depth Enhancement via Anisotropic Diffusion. *Pac. Rim Conf. Multimed.* **2013**, *8294*, 408–417.
11.   Liang, J.; Chen, P.; Wu, M. Research on an Image Denoising Algorithm based on Deep Network Learning. *J. Phys. Conf. Ser.* **2021**, *1802*, 032112. [CrossRef]
12.   Xu, L.; Huang, G.; Chen, Q.; Qin, H.; Men, T.; Pu, Y. An improved method for image denoising based on fractional-order integration. *Front. Inf. Technol. Electron. Eng.* **2020**, *21*, 1485–1493. [CrossRef]
13.   Wang, Q.; Ma, J.; Yu, S.; Tan, L. Noise detection and image denoising based on fractional calculus. *Chaos Solitons Fractals* **2020**, *131*, 109463. [CrossRef]
14.   Wei, Y.; Liu, D.; Boutat, D.; Chen, Y. An improved pseudo-state estimator for a class of commensurate fractional order linear systems based on fractional order modulating functions. *Syst. Control Lett.* **2018**, *118*, 29–34. [CrossRef]
15.   Pu, Y.; Zhou, J.; Xiao, Y. Fractional Differential Mask: A Fractional Differential-Based Approach for Multiscale Texture Enhancement. *IEEE Trans. Image Process.* **2010**, *19*, 491–511. [PubMed]
16.   Hu, J.; Pu, Y.; Zhou, J. Fractional Integral Denoising Algorithm. *J. Univ. Electron. Sci. Technol. China* **2012**, *41*, 706–711.
17.   Guo, H.; Li, X.; Li, C.; Wang, M. Image denoising using fractional integral. *IEEE Int. Conf. Comput. Sci. Autom. Eng.* **2012**, *2*, 107–112.
18.   Zhou, C.; Yan, T.; Tao, W.; Liu, S. A Study of Images Denoising Based on Two Improved Fractional Integral Marks. *Int. Conf. Intell. Comput.* **2012**, *7389*, 386–392.
19.   Podlubny, I.; Chechkin, A.; Skovranek, T.; Chen, Y.; Jara, B.M.V. Matrix approach to discrete fractional calculus II: Partial fractional differential equations. *J. Comput. Phys.* **2009**, *228*, 3137–3152. [CrossRef]
20.   Choi, S.; Zhou, Q.; Miller, S.; Koltun, V. A Large Dataset of Object Scans. *Comput. Vis. Pattern Recognit.* **2016**, *1602*, 02481.