

Article

Super-Resolution Reconstruction of Depth Image Based on Kriging Interpolation

Tingsheng Huang ^{1,2} , Xinjian Wang ^{1,2}, Chunyang Wang ^{1,2,*} , Xuelian Liu ^{2,*}  and Yanqing Yu ³

¹ School of Electronic and Information Engineering, Changchun University of Science and Technology, Changchun 130022, China

² Xi'an Key Laboratory of Active Photoelectric Imaging Detection Technology, Xi'an Technological University, Xi'an 710021, China

³ College of Photonic and Electronic Engineering, Fujian Normal University, Fuzhou 350117, China

* Correspondence: wangchunyang19@163.com (C.W.); tearlxl@126.com (X.L.)

Abstract: The super-resolution of depth images is a research hotspot. In this study, the classical Kriging algorithm is applied to the spatial interpolation of depth images, together with the fractional-order differential method for edge recognition, to realise the super-resolution reconstruction of depth images. The resulting interpolation model improves the edge performance of Kriging interpolation by harnessing the superior characteristics of fractional-order differential edge recognition and effectively solving the edge blurring problem in super-resolution interpolation of depth images. Experimental results show that, compared with the classical algorithms, the super-resolution reconstruction based on Kriging interpolation is greatly improved in terms of visual effects and the peak signal-to-noise ratio of the depth image. In particular, edge recognition based on fractional-order differentiation solves the image blurring problem at the edges of the depth images. Inspection of the point clouds of the depth images shows that the output of the proposed interpolation model has obvious fractal characteristics.

Keywords: edge recognition; fractional-order calculus; kriging interpolation; super-resolution reconstruction



Citation: Huang, T.; Wang, X.; Wang, C.; Liu, X.; Yu, Y. Super-Resolution Reconstruction of Depth Image Based on Kriging Interpolation. *Appl. Sci.* **2023**, *13*, 3769. <https://doi.org/10.3390/app13063769>

Academic Editors: Zheng Wang, Xian Zhong, Liang Liao and Kui Jiang

Received: 16 February 2023

Revised: 8 March 2023

Accepted: 11 March 2023

Published: 15 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

With the development of computer vision and image processing, deep image processing has become a hot research topic in recent years. Deep image processing has both theoretical significance and important practical applications. For example, in the field of virtual reality video games, where the player's pose movements are acquired through somatosensory devices and subjected to human-computer interaction and modelling, depth images acquired in real time must be processed to improve the quality of the models constructed from depth data. Another topic of importance is the super-resolution reconstruction of depth images, which is critical for applications in vision fields such as virtual reality video games, biomedicine, and augmented reality, as these applications require highly accurate depth image data. High-precision super-resolution reconstruction helps generate highly realistic three-dimensional (3D) object surface models. In addition, artificial intelligence and autonomous driving both require highly accurate depth image data. Developments in computer vision and image processing promise to make the exploration of depth-image-based super-resolution algorithms a research hotspot.

Recently, image super-resolution using deep learning methods has achieved remarkable success in producing high-quality and visually pleasing results [1–3]. However, these methods usually require a large amount of training data and computational resources, making them impractical for real-time applications. To overcome this limitation, some researchers have explored the use of non-deep learning-based techniques, such as interpolation-based methods [4,5]. While these methods are computationally efficient, they may not be suitable for processing complex depth images.

Recently, a few studies have proposed the use of fractional-order differentiation in interpolation-based methods for depth image super-resolution [6,7]. These methods have been shown to be effective in preserving texture information and edge features as well as improving noise reduction performance. In comparison with traditional interpolation-based methods, these methods offer better image reconstruction and greater robustness.

Depth image super-resolution is a signal processing technique for obtaining high-resolution images from single or multiple frames of low-resolution images [8] and has promising practical applications [9]. Image super-resolution techniques can be broadly classified into four categories: interpolation-based [10], reconstruction-based [11], enhancement-based [12], and learning-based [1,2] methods. Fractional-order calculus has been widely applied in cybernetics, probability theory, and signal processing [13]. One study [14] proposed a numerical implementation of fractional-order calculus in the field of signal analysis and its initial application to two-dimensional (2D) image processing; the same study also proposed the Tiansi mask operator based on fractional-order calculus, which achieved good results in the field of image enhancement and edge detection. Other studies [15–17] have proposed image denoising algorithms based on a small probability model and gradient-adaptive fractional-order calculus, achieving good results in image denoising and enhancement.

Spatial interpolation is a technique used to obtain attribute information at the unsampled points (interpolation points) by studying the attribute information of the sampled points. Using a mathematical model, the mapping relationship between the attribute domain and the spatial domain is analysed on the basis of known information from a certain number of sampled points [18]. Spatial interpolation techniques are widely used in tasks such as precipitation prediction, soil level classification, and groundwater simulation [19]. Numerous interpolation methods are in common use, such as inverse distance weighted interpolation [20], Kriging interpolation [21,22], minimum curvature interpolation, the Sheppard method, natural neighbour interpolation [20], nearest-neighbour interpolation, multiple regression interpolation, the radial basis function method, the linear interpolation triangular network method, and the moving average method.

The classical super-resolution algorithm for images introduces the Kriging interpolation algorithm by exploiting the fact that a depth image is also a distance image. In this study, the fractional-order calculus operator is introduced into the super-resolution reconstruction process of depth images, and the fractional-order differential mask is constructed to obtain high-frequency information such as image edges for originally low-resolution images. In addition, fractional-order edge matching is used to improve the accuracy of the Kriging interpolation process, and the fractional-order differential kernel is used to construct and improve the Kriging interpolation diffusion function. The experimental results show that the algorithm effectively improves the super-resolution reconstruction and enhances the subjective and objective outputs of the reconstruction process compared with the traditional algorithm.

2. Depth Image super-resolution Algorithm

Super-resolution of depth images requires image enlargement by interpolation of images. The main theoretical basis of such methods is that there is a purely sampling relationship between low-resolution images and high-resolution images. In the up-sampling process, only the pixel values of the low-resolution image need to be fitted to obtain a continuous image, and then sampling can be performed to obtain the target-resolution image. This method benefits from conceptual simplicity and low computational difficulty and is widely used for image super-resolution in industry. However, due to theoretical limitations, the super-resolution effect is not satisfactory when the images contain noise or blur, i.e., when the relationship between the high- and low-resolution images is not purely a sampling relationship.

2.1. Classical Image Interpolation Methods

OpenCV2's `resize` function includes a number of commonly used interpolation algorithms: nearest-neighbour interpolation (`INTER_NEAREST`) [23], bilinear interpolation (`INTER_LINEAR`) [24,25], resampling interpolation (`INTER_AREA`) [26], cubic interpolation (`INTER_CUBIC`) [27], and LANCZOS4 cubic interpolation (`INTER_LANCZOS4`) [28]. Resampling interpolation is based on resampling using the pixel region relationship. Therefore, resampling is preferred in image extraction (decimation). Meanwhile, when the image is zoomed, `INTER_NEAREST` is preferred. These five classical interpolation algorithms are widely used in industry. However, recent years have seen the emergence of several new interpolation algorithms, including the fractal theory-based algorithm [29], the wavelet algorithm [30], the contour template algorithm [31], the partial differential equation-based algorithm [32], the topography-based inverse distance algorithm [33], and the Kriging algorithm [34]. These algorithms have been shown to have superior performance in various applications, such as image super-resolution, terrain modelling, and geological analysis, and therefore merit consideration for future research.

This study evaluates the effectiveness of the five classical interpolation algorithms mentioned at the start of the preceding paragraph with respect to the super-resolution of depth images. Using experimental data obtained from the depth image dataset in [35], the depth images and point clouds are first 50% down-sampled and then 200% up-sampled to examine the effectiveness of the interpolation algorithms. The super-resolution results are shown in Figures 1–6 and Table 1, the images show good clarity and completeness after scaling.

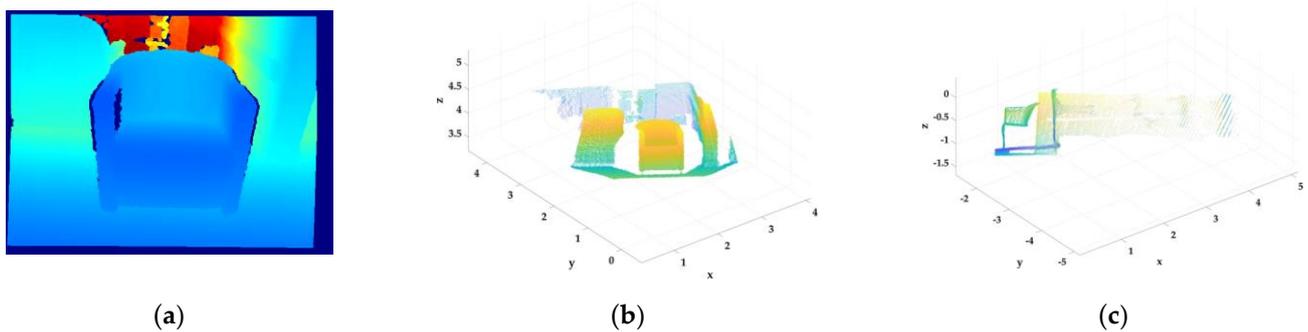


Figure 1. Depth image: original images. (a), depth image; (b), front view of the point cloud map of the depth image; (c), side view of the point cloud map of the depth image.

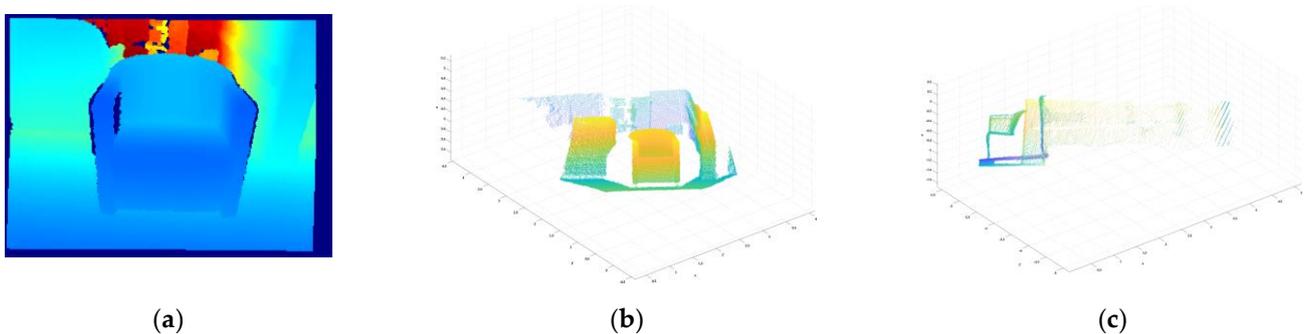


Figure 2. Nearest neighbour interpolation (`INTER_NEAREST`) results for depth images (PSNR = 29.795 dB). (a), depth image; (b), front view of the point cloud map of the depth image; (c), side view of the point cloud map of the depth image.

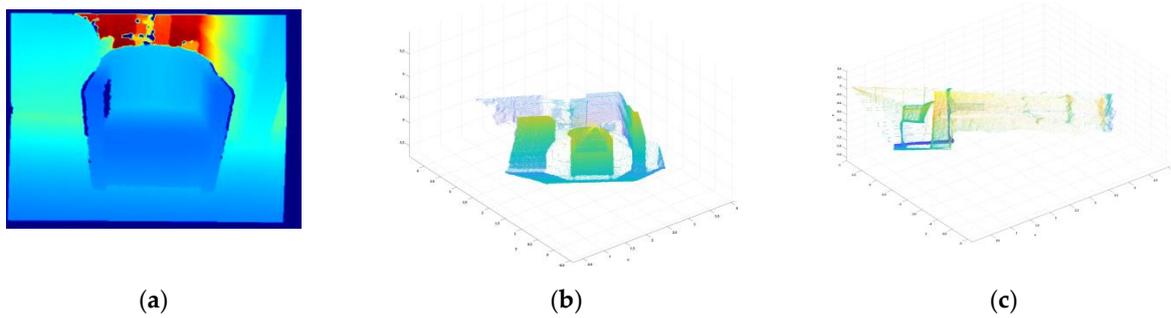


Figure 3. Bilinear interpolation (INTER_LINEAR) results for depth images (PSNR = 34.164 dB). (a), depth image; (b), front view of the point cloud map of the depth image; (c), side view of the point cloud map of the depth image.

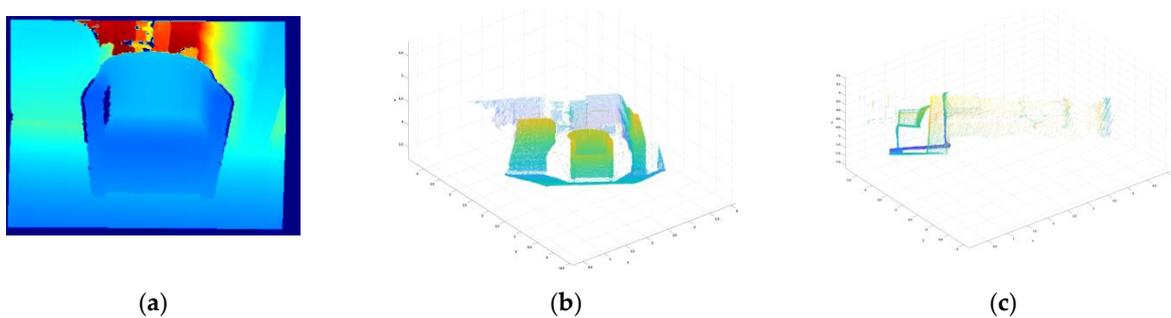


Figure 4. Resampling interpolation (INTER_AREA) results for depth images (PSNR = 33.570 dB). (a), depth image; (b), front view of the point cloud map of the depth image; (c), side view of the point cloud map of the depth image.

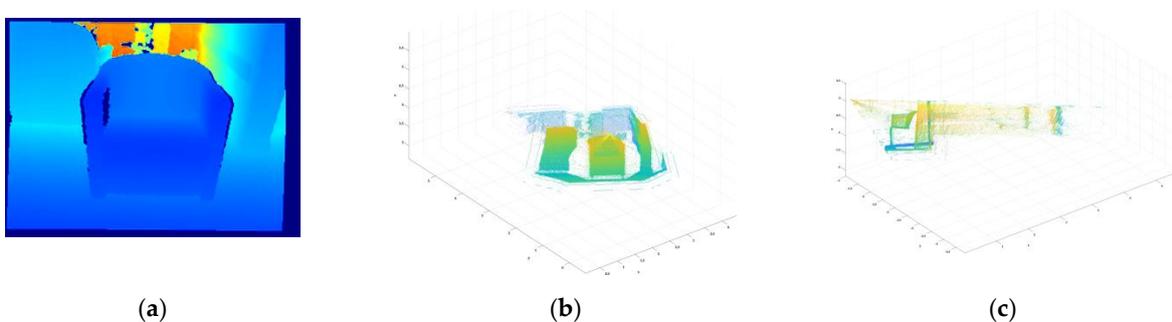


Figure 5. Results of cubic interpolation (INTER_CUBIC) of the depth image (PSNR = 34.515 dB). (a), depth image; (b), front view of the point cloud map of the depth image; (c), side view of the point cloud map of the depth image.

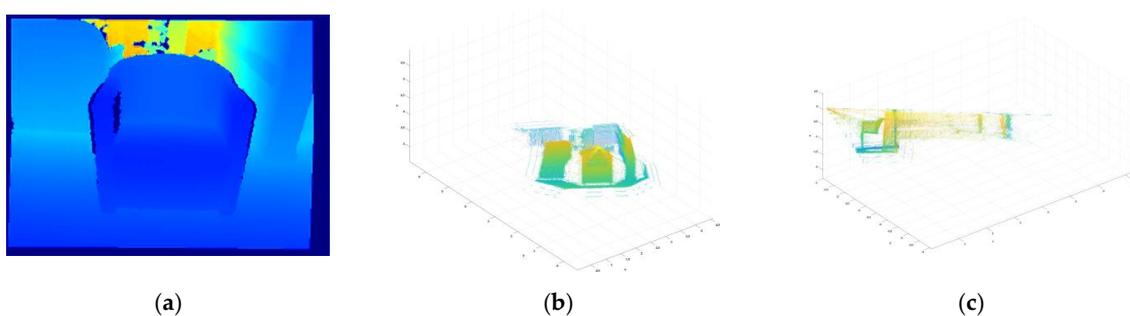


Figure 6. Lanczos interpolation (INTER_LANZOS24) results for depth images (PSNR = 33.654 dB). (a), depth image; (b), front view of the point cloud map of the depth image; (c), side view of the point cloud map of the depth image.

Table 1. Comparison of classical interpolation algorithms.

Interpolation Method	Nearest Neighbor Interpolation	Bilinear Interpolation	Resampling Interpolation	Cubic Interpolation	Lanczos Interpolation
PSNR (dB)	29.795	34.154	33.570	34.515	33.654

The five classical interpolation algorithms are tested experimentally with respect to their ability to meet the requirements of image appearance. In terms of image edge completeness and clarity, the cubic interpolation and Lanczos interpolation algorithms achieve the highest values. However, after conversion of the depth images into point clouds, it is evident that the cubic interpolation and Lanczos interpolation algorithms introduce a large amount of smoothed data to optimise the appearance of the image. From the point cloud or depth distance perspectives, these data can be seen as a source of additional noise. Overall, the above experimental results show that the classical image interpolation algorithms, except the nearest-neighbour interpolation algorithm, inevitably introduce noise that cannot be eliminated. If the images need to be converted to point clouds and reconstructed in 3D, the noise introduced will greatly affect the final result and be difficult to eliminate. If high-quality, super-resolution depth images are to be obtained, improvements are needed, starting with the interpolation algorithm.

2.2. Interpolation Algorithm Based on Kriging Interpolation

A depth image is itself a distance image and is similar to a geological/topographic image after conversion to a point cloud. Considering the concept of inverse distance used in adaptive weighting, an algorithm suitable for geology/topography is introduced. The algorithm is based on inverse distance interpolation and Kriging interpolation, both of which take into account the spatial relationship between the point to be interpolated and the valid points around them [20,22].

Kriging interpolation is a method for the unbiased optimal estimation of regionalised variables in a finite region (used to estimate values that are spatially correlated, such as air quality, where values at closely spaced locations are similar). ‘Unbiased’ means that the expectation of the difference between the estimated and actual values is equal to zero, and ‘optimal’ means that the variance between the estimated and actual values is minimised. Based on this feature, Kriging interpolation is much more effective than other interpolation methods. In simple terms, Kriging interpolation does the following. Given some coordinates and the real values at those coordinates, which we call sampling points, these points are fed into the Kriging algorithm to estimate the values at other unknown locations [22].

Kriging interpolation analyses the regional variation of variables to achieve an optimal linear unbiased estimation [20,22]. The method consists of two steps: (1) the variational function model is established using the values of the spatial attributes of the regional variables, which are used to express the structural and stochastic characteristics of those attributes; (2) on the basis of the analysis of the spatial attributes, the domain range is determined and the Kriging equation is solved to estimate the values of the attributes with interpolation points.

$$\hat{z}_o = \sum_{i=1}^n \lambda_i z_i \quad (1)$$

where z_i is the estimated value at the point (x, y) and λ_i is the weight coefficient, which is a weighted sum of the data at all known points in the space and is used to estimate the value of unknown points. However, the weight coefficients are not the inverse of the distance but are the optimal set of coefficients that can satisfy the minimum difference between the valuation \hat{z}_o of the point (x, y) and the true value of z_o , i.e.,

$$\min_{\lambda_i} \text{Var}(\hat{z}_o - z_o) \quad (2)$$

Additionally, the unbiased condition must be satisfied:

$$E(\hat{z}_o - z_o) = 0 \tag{3}$$

The main difference between different Kriging interpolation methods is the underlying assumptions. In this paper, only the assumptions and applications of ordinary Kriging interpolation are presented.

Ordinary Kriging interpolation assumes that the spatial properties z are homogeneous. All points in space (x, y) have the same expectation c with variance σ^2 . That is, for any point (x, y) , we have

$$E[z(x, y)] = E[z] = c \tag{4}$$

$$\text{Var}[z(x, y)] = \sigma^2 \tag{5}$$

In other words, the value at any point $z(x, y)$ is composed of the regional mean c and the random deviation at that point $R(x, y)$:

$$z(x, y) = E[z(x, y)] + R(x, y) = c + R(x, y) \tag{6}$$

where $R(x, y)$ denotes the deviations at the point (x, y) , and their variances are all constants.

$$\text{Var}[z(x, y)] = \sigma^2 \tag{7}$$

First, we analyse the unbiased estimation condition:

$$E(\hat{z}_o - z_o) = 0 \tag{8}$$

Substitute

$$\hat{z}_o = \sum_{i=1}^n \lambda_i z_i \tag{9}$$

We obtain

$$E\left(\sum_{i=1}^n \lambda_i z_i - z_o\right) = 0 \tag{10}$$

Furthermore, because $E[z] = c$ for any z , one of the constraints on λ_i can be obtained.

$$\sum_{i=1}^n \lambda_i = 1 \tag{11}$$

The analysis estimation error $\text{Var}(\hat{z}_o - z_o)$ is denoted here by J :

$$J = \text{Var}(\hat{z}_o - z_o) \tag{12}$$

Furthermore, we have

$$\begin{aligned} J &= \text{Var}\left(\sum_{i=1}^n \lambda_i z_i - z_o\right) \\ &= \text{Var}\left(\sum_{i=1}^n \lambda_i z_i - 2\text{Cov}\left(\sum_{i=1}^n \lambda_i z_i, z_o\right) - z_o\right) + \text{Cov}(z_o, z_o) \\ &= \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j \text{Cov}(z_i, z_j) - 2 \sum_{i=1}^n \lambda_i \text{Cov}(z_i, z_o) + \text{Cov}(z_o, z_o) \end{aligned} \tag{13}$$

By definition, $C_{ij} = \text{Cov}(z_i, z_j) = \text{Cov}(R_i, R_j)$, where $R_i = z_i - c$ denotes the deviation of the attribute value at point (x_i, y_i) from the regional average attribute value

$$J = \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j C_{ij} - 2 \sum_{i=1}^n \lambda_i C_{i0} + C_{00} \tag{14}$$

Defining the semivariance function $r_{ij} = \sigma^2 - C_{ij}$, which is substituted into J, we have the following:

$$\begin{aligned}
 J &= \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (\sigma^2 - r_{ij}) - 2 \sum_{i=1}^n \lambda_i (\sigma^2 - r_{i0}) + (\sigma^2 - r_{00}) \\
 &= \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (\sigma^2) - \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (r_{ij}) - 2 \sum_{i=1}^n \lambda_i (\sigma^2) + 2 \sum_{i=1}^n \lambda_i (r_{i0}) + (\sigma^2 - r_{00})
 \end{aligned}
 \tag{15}$$

Given that $\sum_{i=1}^n \lambda_i = 1$, after substitution we have

$$\begin{aligned}
 J &= \sigma^2 - \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (r_{ij}) - 2\sigma^2 + 2 \sum_{i=1}^n \lambda_i (r_{i0}) + (\sigma^2 - r_{00}) \\
 &= 2 \sum_{i=1}^n \lambda_i (r_{i0}) - \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (r_{ij}) - r_{00}
 \end{aligned}
 \tag{16}$$

The goal is to find a set of λ_i that minimises J. As J is a function of λ_i , we can directly calculate the partial derivative of J with respect to λ_i , and find the zero-valued point. That is, we solve the following for λ_i :

$$\frac{\partial J}{\partial \lambda_i} = 0; \quad i = 1, 2, 3, \dots, n
 \tag{17}$$

It is also necessary to satisfy $\sum_{i=1}^n \lambda_i = 1$. Additionally, the optimisation problem with this constraint is solved using the Lagrange multiplier method by first constructing a new objective function.

$$J + 2\phi \left(\sum_{i=1}^n \lambda_i - 1 \right)
 \tag{18}$$

where ϕ is the Lagrangian multiplier. We solve for the set of parameters $\phi, \lambda_1, \lambda_2, \dots, \lambda_n$ that minimises this cost function and thereby minimises J under the constraint $\sum_{i=1}^n \lambda_i = 1$, which yields

$$\begin{cases} \frac{\partial (J + 2\phi (\sum_{i=1}^n \lambda_i - 1))}{\partial \lambda_i} = 0; \quad i = 1, 2, 3, \dots, n \\ \frac{\partial (J + 2\phi (\sum_{i=1}^n \lambda_i - 1))}{\partial \phi} = 0 \end{cases}
 \tag{19}$$

Substituting into J, we obtain

$$\begin{cases} \frac{\partial (2 \sum_{i=1}^n \lambda_i (r_{i0}) - \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (r_{ij}) - r_{00} + 2\phi (\sum_{i=1}^n \lambda_i - 1))}{\partial \lambda_i} = 0; \\ \frac{\partial (2 \sum_{i=1}^n \lambda_i (r_{i0}) - \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (r_{ij}) - r_{00} + 2\phi (\sum_{i=1}^n \lambda_i - 1))}{\partial \phi} = 0 \\ (i = 1, 2, 3, \dots, n) \end{cases}
 \tag{20}$$

$$\begin{cases} \frac{\partial (2 \sum_{i=1}^n \lambda_i (r_{i0}) - \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (r_{ij}) - r_{00} + 2\phi (\sum_{i=1}^n \lambda_i - 1))}{\partial \lambda_i} = 0; \\ \frac{\partial (2 \sum_{i=1}^n \lambda_i (r_{i0}) - \sum_{i=1}^n \sum_{j=0}^n \lambda_i \lambda_j (r_{ij}) - r_{00} + 2\phi (\sum_{i=1}^n \lambda_i - 1))}{\partial \phi} = 0 \\ i = 1, 2, 3, \dots, n \end{cases}
 \tag{21}$$

For the reason that $C_{ij} = \text{Cov}(z_i, z_j) = C_{ji}$, the same semivariance function $r_{ij} = r_{ji}$, and further simplification shows that

$$\begin{cases} r_{i0} - \sum_{j=1}^n r_{ij} \lambda_j + \phi = 0; \quad i = 1, 2, 3, \dots, n \\ \sum_{i=1}^n \lambda_i = 1 \end{cases}
 \tag{22}$$

The equations to be solved for the above weighting factor λ_i are first written in the form of a linear system of equations, as follows.

$$\begin{cases} r_{11}\lambda_1 + r_{12}\lambda_2 + \dots + r_{1n}\lambda_n - \phi = r_{1o} \\ r_{21}\lambda_1 + r_{22}\lambda_2 + \dots + r_{2n}\lambda_n - \phi = r_{2o} \\ \vdots \\ r_{n1}\lambda_1 + r_{n2}\lambda_2 + \dots + r_{nn}\lambda_n - \phi = r_{no} \\ \lambda_1 + \lambda_2 + \dots + \lambda_n = 1 \end{cases} \tag{23}$$

The system of equations is written in matrix form, and the inverse matrix solution procedure is performed.

By the definition of semi-covariance, $r_{ij} = \sigma^2 - C_{ij}$ and its equivalent form $r_{ij} = \frac{1}{2}E[(z_i - z_j)^2]$.

By the previous definition (hypothetical conditions), we have $Z(x, y) = c + R(x, y)$, so we then have $R_i = z_i - c$. Further, we have $z_i - z_j = R_i - R_j$ under these conditions:

$$\begin{aligned} r_{ij} &= \frac{1}{2}E[(R_i - R_j)^2] \\ &= \frac{1}{2}E[R_i^2 - 2R_{ij} + R_j^2] \\ &= \frac{1}{2}E[R_i^2] + \frac{1}{2}E[R_j^2] - E[R_i R_j] \end{aligned} \tag{24}$$

This follows from the fact that

$$E[R_i^2] = E[R_j^2] = E[(z_i - c)^2] = \text{Var}(z_i) = \sigma^2 \tag{25}$$

$$E[R_i R_j] = E[(z_i - c)(z_j - c)] = \text{Cov}(z_i, z_j) = C_{ij} \tag{26}$$

After substitution, we obtain

$$r_{ij} = \frac{1}{2}\sigma^2 + \frac{1}{2}\sigma^2 - C_{ij} = \sigma^2 - C_{ij} \tag{27}$$

The above proof confirms that $r_{ij} = \frac{1}{2}E[(z_i - z_j)^2]$. The following procedure is then used to calculate r_{ij} . Here, we continue to use the principle of similarity of attributes in space. This spatial similarity is expressed by $r_{ij} = \frac{1}{2}(z_i - z_j)^2$ in terms of distance, defining the geometric distance between z_i and z_j as

$$d_{ij} = d(z_i, z_j) = d((x_i, y_i), (x_j, y_j)) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{28}$$

Kriging interpolation assumes that there is a functional relationship between r_{ij} and d_{ij} , which can be linear, quadratic, exponential or logarithmic. To identify this relationship, we perform the following steps on the observed data set $\{z(x_1, y_1), z(x_2, y_2), \dots, z(x_{n-1}, y_{n-1}), z(x_n, y_n)\}$.

First, we calculate the distance between any two points

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{29}$$

and semi-covariance

$$r_{ij} = \frac{1}{2}(z_i - z_j)^2 \tag{30}$$

It is possible to obtain n^2 (d_{ij}, r_{ij}) data pairs by plotting all the d and r values in a scatter plot to find an optimal curve to fit d and r and obtain a functional relationship.

$$r = r(d) \tag{31}$$

Furthermore, for any two points $z(x_i, y_i), z(x_j, y_j)$, we first calculate their distance, d_{ij} . Next, according to the functional relationship, we obtain the semi-covariance r_{ij} . Considering the computational complexity, simple Kriging is used for the interpolation of depth images. The ordinary Kriging formula is as follows.

$$\hat{Z}_o = \sum_{i=1}^n \lambda_i z_i \quad (32)$$

The simple Kriging formula is as follows.

$$\hat{Z}_o - c = \sum_{i=1}^n \lambda_i z_i - c \quad (33)$$

where c is the mathematical expectation of the attribute value, i.e., $E[z] = c$. That is, in ordinary Kriging interpolation, the attribute value of an unknown point is considered to be a weighted sum of the attribute values of known points.

The simple Kriging Formula (33) completes the derivation of the estimation of the values of depth images by interpolation, according to which different peripheral depth values can be selected for interpolation. Based on this theoretical derivation, a depth image interpolation algorithm is constructed to super-resolve depth images by Kriging interpolation. The method first down-samples the image by 50% and then up-samples it by 200%, where the sampling algorithm uses Kriging interpolation. The super-resolution results are shown in Figure 7 which shows that the Kriging interpolation algorithm is affected by edges and voids. It can also be seen from the output of the Kriging interpolation algorithm that rapid changes at edges can lead to highly discrepant interpolation results from the actual situation.

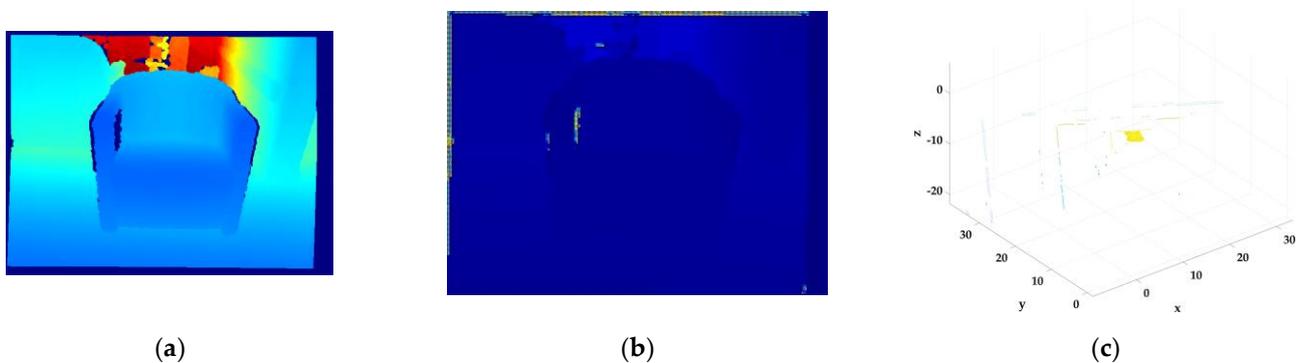


Figure 7. Kriging interpolation results for depth images (PSNR = 20.629 dB). (a), depth image; (b), depth image after Kriging interpolation; (c), point cloud map orthomosaic of the depth image.

3. Fractional-Order Differential Edge Recognition

Image enhancement processing is an important branch of digital image processing. Under real-world environmental conditions, the visual effectiveness of image capture is poor, which necessitates image enhancement techniques. Such techniques include highlighting certain characteristics of the target object in the image and extracting the characteristic parameters of the target object from the digital image. The goal is to improve the recognition, tracking, and understanding of the target [36]. The main objective of image enhancement processing is to highlight the parts of interest in an image and attenuate or remove unwanted information so that the useful information is enhanced to obtain a more useful image, or the image can be made more suitable for analysis and processing by humans or machines. The main features of an image (i.e., the main information) include edges, texture details, and other features. The enhancement of image features plays an important role in improving the visual effectiveness of an image and also lays a good foundation for subsequent image processing (e.g., image recognition, image segmentation, or super resolution).

Fractional-order differentiation significantly enhances the high- and medium-frequency components of the signal while nonlinearly strengthening the medium-frequency components of the signal to a certain extent as well as nonlinearly preserving the low-frequency and DC components of the signal. Therefore, fractional-order differentiation is beneficial in terms of both significantly enhancing the image edge and contour information and strengthening the texture details in the smooth regions of the image. Due to this property of fractional-order differentiation, the fractional calculus theory in image processing operations has received increasing attention [13–17].

It is generally believed that the main benefit of applying methods based on fractional-order calculus to digital images is to add a degree of freedom, i.e., the fractional-order ν ($0 < [\nu] < 1$). By choosing the appropriate order and constructing the corresponding mask operator, better image enhancement ($\nu > 0$) and denoising ($\nu < 0$) can be achieved. Huang G et al. derived fractional-order calculus and verified the application of the fractional-order integral in image processing [37]. They also proposed a fractional-order differential enhancement and fractional-order integral denoising model based on the Grünwald–Letnikov (G-L) definition of 2D images.

Introducing the G-L definition of the fractional-order calculus, the $-\nu$ order integral of the G-L expression [17] is given by the following equation:

$${}_a^G D_x^{-\nu} = \lim_{h \rightarrow 0} h^\nu \sum_{n=0}^{\lfloor \frac{t-a}{h} \rfloor} \binom{-\nu}{j} f(x - jh), \nu \in \mathbb{R} \tag{34}$$

The fractional-order integral can be approximately expanded by defining the one-dimensional (1D) signal $f(t)$ of the duration interval as $[a, t]$. According to the unit $h = 1$ equipartition, we have $m = \lfloor \frac{t-a}{h} \rfloor = [t - a]$, implying that there is a 1D signal of ν ($\nu \geq 0$). The equivalent expression of the fractional-order differentiation of order ν is

$$\frac{d^\nu t}{dt^\nu} \approx f(t) + (-1)^1 (\nu) f(t-1) + (-1)^2 \binom{\nu(\nu+1)}{2} f(t-2) + \dots + (-1)^j \frac{\Gamma(\nu+1)}{\Gamma(j+1)\Gamma(\nu-j+1)} f(t-j) \tag{35}$$

Define the 2D image signal $I(x, y)$, and assume that the fractional-order derivatives of $I(x, y)$ are separable for the x-axis and y-axis directions under certain conditions. Furthermore, we use the separability of the Fourier transform to extend the fractional-order calculus from one dimension to two dimensions. By equating the 2D image signal $I(x, y)$ according to the unit time with $h = 1$, the fractional-order calculus of the x-axis and y-axis can be obtained, and ν is the fractional-order calculus order ($\nu \geq 0$).

From Equation (35), the approximate solution of partial fractional-order calculus under the G-L definition can be derived, enabling us to solve the numerical expression of the fractional-order calculus operator along the x-axis and y-axis directions, namely

$$\frac{d^\nu I(x, y)}{dx^\nu} \approx I(x, y) + (-1)^1 (\nu) I(x-1, y) + (-1)^2 \binom{\nu(\nu-1)}{2} I(x-2, y) + \dots + (-1)^j \frac{\Gamma(\nu+1)}{\Gamma(j+1)\Gamma(\nu-j+1)} I(x-j, y) \tag{36}$$

$$\frac{d^\nu I(x, y)}{dy^\nu} \approx I(x, y) + (-1)^1 (\nu) I(x, y-1) + (-1)^2 \binom{\nu(\nu-1)}{2} I(x, y-2) + \dots + (-1)^j \frac{\Gamma(\nu+1)}{\Gamma(j+1)\Gamma(\nu-j+1)} I(x, y-j) \tag{37}$$

That is,

$$\frac{d^\nu I(x, y)}{dx^\nu} = \lim_{N \rightarrow \infty} \left[\sum_{m=0}^{N-1} (-1)^j \frac{\Gamma(\nu+1)}{\Gamma(j+1)\Gamma(\nu-j+1)} I(x-j, y) \right] \tag{38}$$

$$\frac{d^\nu I(x, y)}{dy^\nu} = \lim_{N \rightarrow \infty} \left[\sum_{m=0}^{N-1} (-1)^j \frac{\Gamma(\nu+1)}{\Gamma(j+1)\Gamma(\nu-j+1)} I(x, y-j) \right] \tag{39}$$

From Equations (38) and (39), we can obtain ν ($\nu \geq 0$). The coefficients of the fractional-order differential operator of order R are as follows.

$$R = (-1)^j \binom{\nu}{j} = (-1)^j \frac{\Gamma(\nu+1)}{\Gamma(j+1)\Gamma(\nu-j+1)} \tag{40}$$

Assuming a mask size of 3×3 , i.e., setting $N = 3$, the approximate solutions in both axis directions can be obtained from Equations (38) and (39).

$$\frac{d^v I(x, y)}{dx^v} \approx I(x, y) + (-1)^1(v)I(x - 1, y) + (-1)^2\left(\frac{v(v - 1)}{2}\right)I(x - 2, y) \tag{41}$$

$$\frac{d^v I(x, y)}{dy^v} \approx I(x, y) + (-1)^1(v)I(x, y - 1) + (-1)^2\left(\frac{v(v - 1)}{2}\right)I(x, y - 2) \tag{42}$$

Considering that the other six directions in the image have similar definitions, the approximate solution of the fractional-order differential in those six directions can be derived. Thus, a fractional-order differential operator filter with 8 directions can be obtained, and the operator is rotationally invariant.

From Equations (40)–(42), we can obtain the fractional-order differential mask and a 5×5 fractional-order differential mask template v ($v \geq 0$). The coefficients of the fractional-order differential convolution mask are

$$a_*^v = [R_0^v, R_1^v, R_2^v], (j = 0, 1, 2) \tag{43}$$

where

$$R_0^v = 1 \tag{44}$$

$$R_1^v = -v \tag{45}$$

$$R_2^v = \frac{v(v - 1)}{2} \tag{46}$$

The definition mask template has 8 directions, where by definition the x -axis positive and negative directions have a_0^v and a_{180}^v . In the anticlockwise direction, there are a_{45}^v , a_{135}^v , a_{225}^v , and a_{315}^v , and the positive and negative directions of the y -axis have a_{90}^v and a_{270}^v . The constructed masks are as follows.

Where the coefficients are as follows

$$a_0^v = R_0^v = 1 \tag{47}$$

$$a_1^v = R_1^v = -v \tag{48}$$

$$a_2^v = R_2^v = \frac{v(v - 1)}{2} \tag{49}$$

In any image, there is a certain similarity between neighbouring pixels. To reduce the unnecessary spatial and temporal complexity, the feature information of the local neighbourhood of the target pixel should be fully utilised, considering that the pixels closer to the target have higher similarity to the target. Given 8 directions of 3×3 masks $a * v$, for a 5×5 masks of image points $I(x, y)$, the convolution calculation is performed as follows:

$$I(x, y)_*^v = I(x, y) * a_*^v \tag{50}$$

The result of the convolution calculation is linearly weighted by the sum of the convolutions in each direction to obtain the final result:

$$I(x, y)_*^v = \frac{I(x, y)_0^v}{\text{sum}(x, y)} \times I(x, y)_0^{-v} + \frac{I(x, y)_{45}^v}{\text{sum}(x, y)} \times I(x, y)_{45}^v + \frac{I(x, y)_{90}^v}{\text{sum}(x, y)} \times I(x, y)_{90}^v + \frac{I(x, y)_{135}^v}{\text{sum}(x, y)} \times I(x, y)_{135}^v + \frac{I(x, y)_{180}^v}{\text{sum}(x, y)} \times I(x, y)_{180}^v + \frac{I(x, y)_{225}^v}{\text{sum}(x, y)} \times I(x, y)_{225}^v + \frac{I(x, y)_{270}^v}{\text{sum}(x, y)} \times I(x, y)_{270}^v + \frac{I(x, y)_{315}^v}{\text{sum}(x, y)} \times I(x, y)_{315}^v \tag{51}$$

where

$$\text{sum}(x, y) = I(x, y)_0^v + I(x, y)_{45}^v + I(x, y)_{90}^v + I(x, y)_{135}^v + I(x, y)_{180}^v + I(x, y)_{225}^v + I(x, y)_{270}^v + I(x, y)_{315}^v \tag{52}$$

After the fractional-order differential mask convolution calculation, there is a continuous process of enlargement or reduction. Hence, it is necessary to construct the fractional-order differential normalisation factor q to unify the final enlargement and reduction factors. The factor is calculated as follows:

$$q = \sum_{\theta=0}^{360} I(x, y)_{\theta}^{\nu} \quad (\theta = 0, 45, 90, 135, 180, 215, 270, 315) \quad (53)$$

Importing q into Equation (54), the new $I(x, y)_{\text{m}}^{\nu}$ after fractional-order differential filtering of order ν using a 5×5 mask is obtained as follows.

$$I(x, y)_{\text{m}}^{\nu} = \frac{I(x, y) * a_{*}^{\nu}}{q} \quad (54)$$

In 2D images, both edges and noise are discontinuous points with local characteristics, and both have rapid changes in the pixel values of the corresponding domain, so they are high-frequency signals. The edges have the characteristics of orderliness, directionality, and structure, and they have a larger energy and range relative to the noise. Edge detection models based on fractional-order differentiation have shown good results. The authors of [13] recommended the use of $\nu = 0.6$ in fractional-order differentiation based on their experiments, which enabled them to achieve richer edge information compared with the classical integer-order edge detection operator. From the depth image and the corresponding point cloud map after the fractional-order differential mask convolution calculation in Figure 8. It can be seen from Figure 9 that the fractional-order differential method can identify the edges accurately.

a_2^{ν}	0	a_2^{ν}	0	a_2^{ν}
0	a_1^{ν}	a_1^{ν}	a_1^{ν}	0
a_2^{ν}	a_1^{ν}	$8a_0^{\nu}$	a_1^{ν}	a_2^{ν}
0	a_1^{ν}	a_1^{ν}	a_1^{ν}	0
a_2^{ν}	0	a_2^{ν}	0	a_2^{ν}

Figure 8. Fractional-order differential mask.

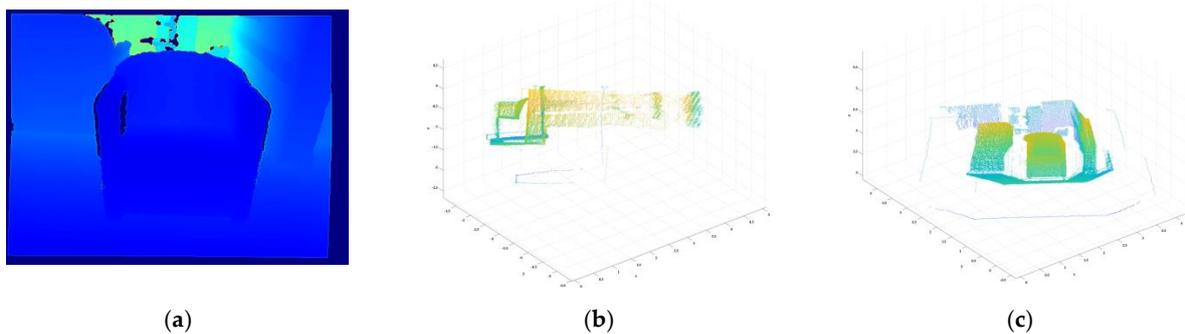


Figure 9. Fractional-order differential edge recognition for depth images. (a), depth image; (b), front view of the point cloud map of the depth image; (c), side view of the point cloud map of the depth image.

4. Kriging Interpolation Super-Resolution Algorithm Based on Fractional-Order Differentiation

Depth images consist of ordered distance data distributed spatially, and the orderliness makes them conducive to parallel computation, a technology that may realise efficient filtering and convolution computation in the future.

Depth images also contain peripheral correlation, and the high-frequency points of edges can be identified by fractional-order differentiation. The high-frequency points are interpolated using the spatial local neighbourhood, while the non-high-frequency points are interpolated by Kriging interpolation. The flowchart in Figure 10 depicts the fractional-order differential Kriging interpolation super-resolution model, which is divided into the following processes: acquiring the depth image, down-sampling the depth image, performing fractional-order differential edge detection, interpolating the down-sampled depth map using the fractional-order differential Kriging interpolation super-resolution model, and obtaining the high-resolution depth image. Table 2 shows the comparison between the kriging interpolation algorithm based on fractional order differentiation and the nearest neighbor interpolation algorithm. It can be seen that the PSNR of kriging interpolation based on fractional order differentiation is better than the nearest neighbor interpolation algorithm.

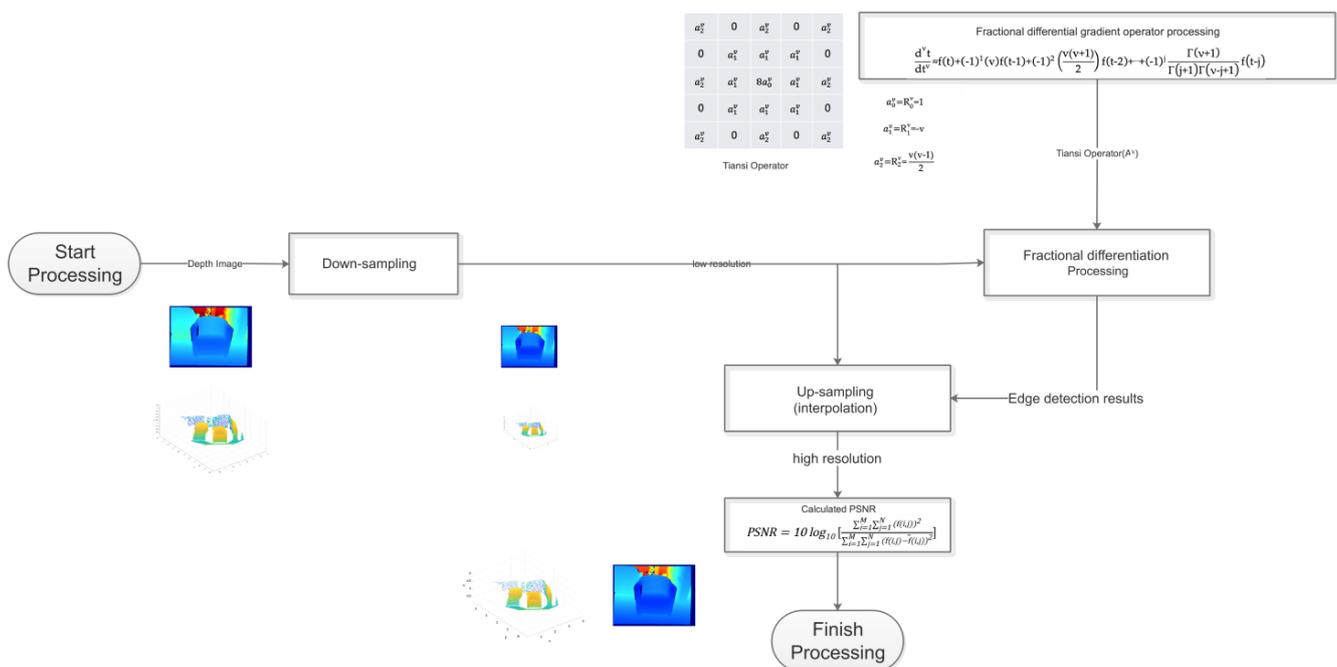


Figure 10. Super-resolution flow chart.

Using open-source datasets [35], a comparison of nearest-neighbour interpolation and Kriging interpolation is performed. In the Kriging interpolation method, the images are first up-sampled by 200% and then down-sampled by 50%. As shown in Table 2, using peak signal-to-noise ratio (PSNR) as the evaluation metric, Kriging interpolation outperforms nearest-neighbour interpolation in terms of reflecting texture details. Specifically, the super-resolution method based on fractional-order differential Kriging interpolation improves the PSNR by 3–18 dB, a significant enhancement. The super-resolved depth images are converted into point cloud maps for comparative analysis, and it can be seen that most of the edges are correctly recovered. These experimental results show that the fractional-order differential Kriging interpolation algorithm performs very well in the super-resolution of depth images and has significance for further research and application.

Table 2. Fractional-order differential Kriging interpolation super-resolution (200%).

Datasets [35] Number	Fractional-Order Differential Kriging Interpolation		Nearest Neighbor Interpolation
	ν	PSNR	PSNR
00033	0.6	33.457	29.795
03236	0.6	37.230	27.738
03528	0.6	38.598	29.868
04797	0.6	47.182	26.580
05989	0.6	39.888	29.468
08343	0.6	38.391	25.008

The super-resolution results are shown in Figure 11 and Table 2, it can be seen that the edge blurring problem can be solved by edge detection, but the adoption of the simple nearest-neighbour interpolation scheme also causes unwanted jagged edge effects. Subsequently, the super-resolution model based on Kriging interpolation is verified for larger-scale magnification factors using 25% down-sampling and 400% up-sampling. The data in Table 3 show that the super-resolution method based on Kriging interpolation via fractional-order differential edge detection achieves a greater improvement in super-resolution than that of nearest-neighbour interpolation.

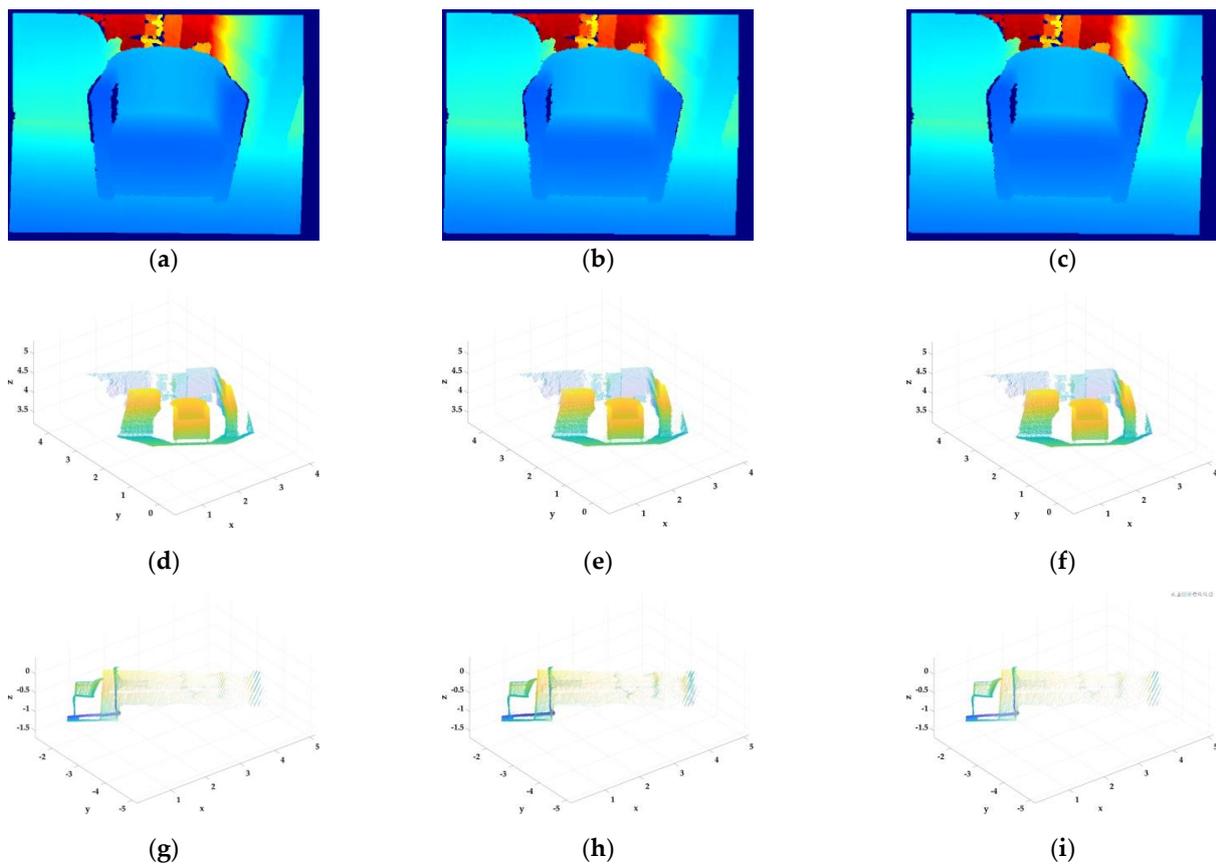


Figure 11. Validation of fractional-order differential Kriging interpolation super-resolution model (dataset no. 00333). (a,d,g) Original depth images; (b,e,h) Outputs of fractional-order differential Kriging interpolation super-resolution model ($\nu = 0.6$, PSNR = 33.299 dB); (c,f,i) Outputs of nearest-neighbour interpolation super-resolution model (PSNR = 29.795 dB).

Table 3. Super-resolution results of the method based on fractional-order differential Kriging interpolation (400% up-sampling) and the nearest-neighbour interpolation method.

Datasets [35] Number	Fractional-Order Differential Kriging Interpolation		Nearest-Neighbour Interpolation
	ν	PSNR	PSNR
00033	0.6	28.828	25.385
03236	0.6	32.290	23.383
03528	0.6	33.677	25.825
04797	0.6	40.595	22.456
05989	0.6	34.459	25.503
08343	0.6	33.819	21.051

The super-resolution results are shown in Figures 12–18 and Table 3. The results of Figure 12 are compared in side view in Figure 13 as an example. As can be seen from Figure 13, the results of the interpolation methods differ in the details. The edge detection method using fractional-order differentiation is implemented to solve the problem of blurred edge points. In terms of texture details, the Kriging interpolation yields more details than the nearest-neighbour interpolation. In particular, the Kriging interpolation yields results with more texture details and a certain level of randomness and fractal characteristics, indicating its superiority over inverse distance interpolation in terms of detail representation.

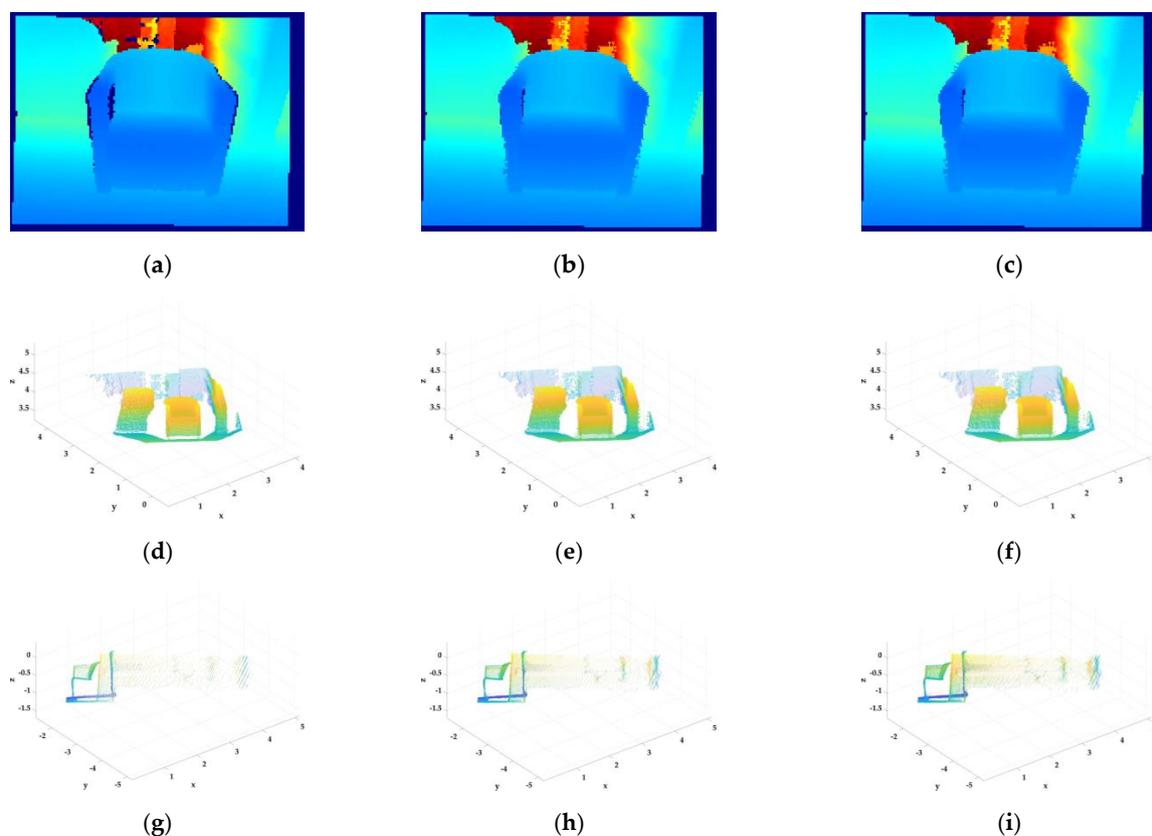


Figure 12. Validation of fractional-order differential Kriging interpolation super-resolution model (dataset no. 00333). (a,d,g) Original depth images; (b,e,h) Outputs of fractional-order differential Kriging interpolation super-resolution model ($\nu = 0.6$, PSNR = 28.828 dB); (c,f,i) Outputs of nearest-neighbour interpolation super-resolution model (PSNR = 25.385 dB).

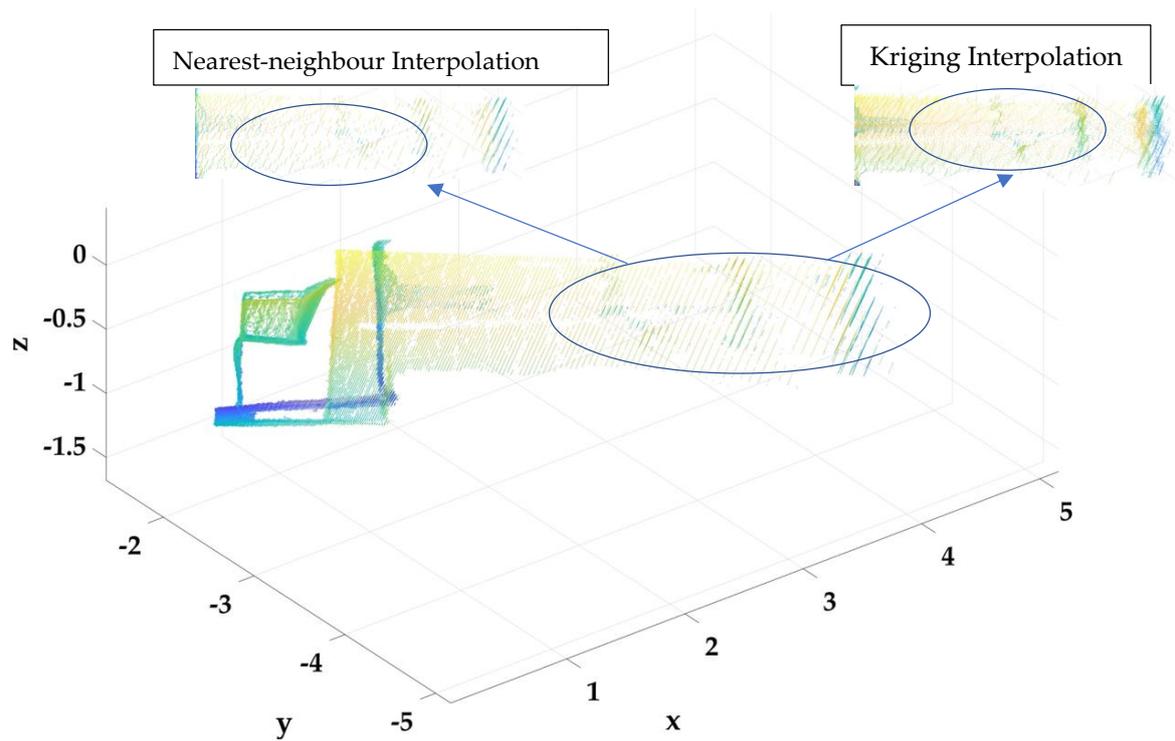


Figure 13. Comparison of details of the Kriging and nearest-neighbour interpolation algorithms.

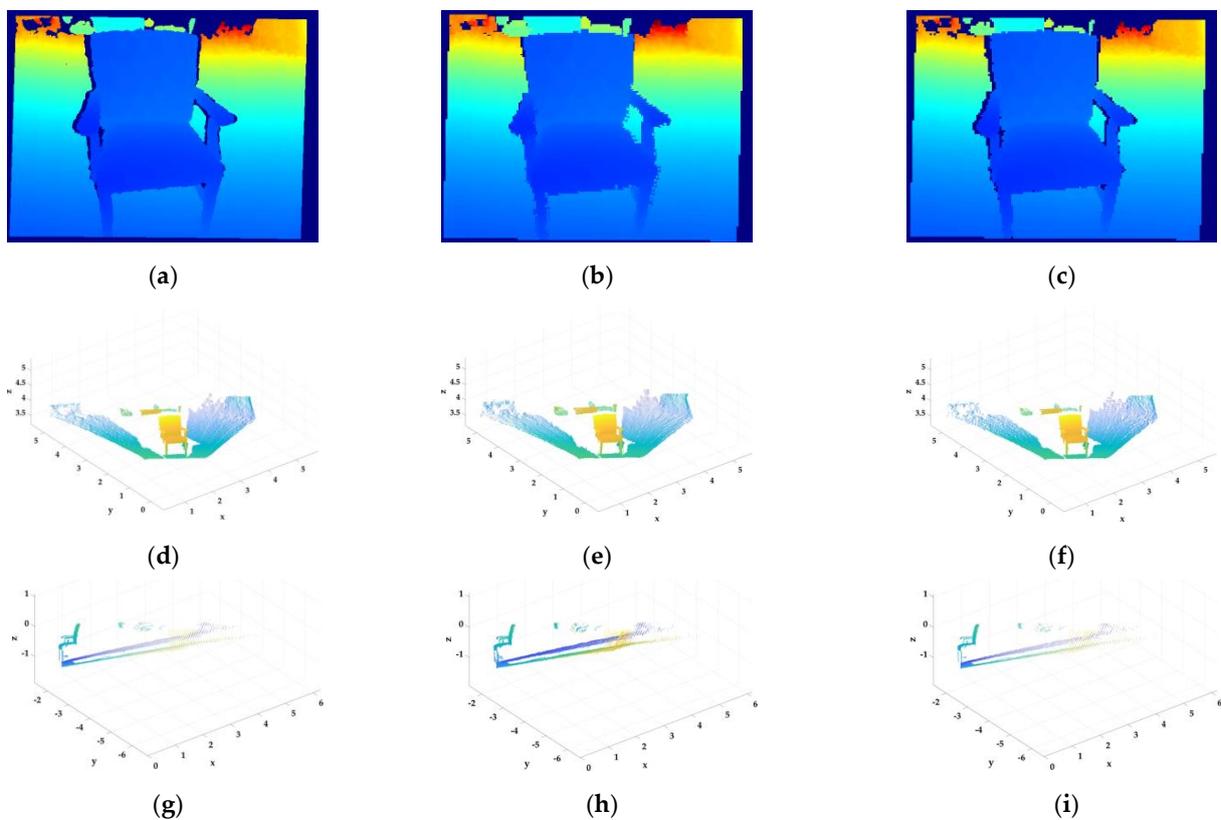


Figure 14. Validation of fractional-order differential Kriging interpolation super-resolution model (dataset no. 05989). (a,d,g) Original depth images; (b,e,h) Outputs of fractional-order differential Kriging interpolation super-resolution model ($\nu = 0.6$, PSNR = 34.459 dB); (c,f,i) Outputs of nearest-neighbour interpolation super-resolution model (PSNR = 25.503 dB).

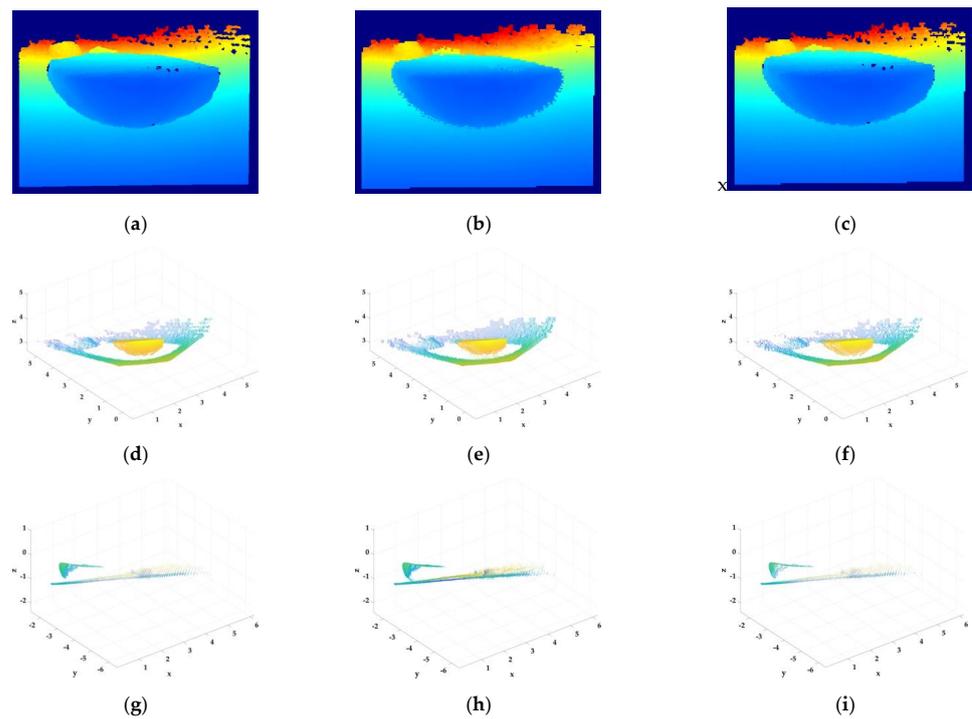


Figure 15. Validation of fractional-order differential Kriging interpolation super-resolution model (dataset no. 03236). (a,d,g) Original depth images; (b,e,h) Outputs of fractional-order differential Kriging interpolation super-resolution model ($\nu = 0.6$, PSNR = 32.290 dB); (c,f,i) Outputs of nearest-neighbour interpolation super-resolution model (PSNR = 23.383 dB).

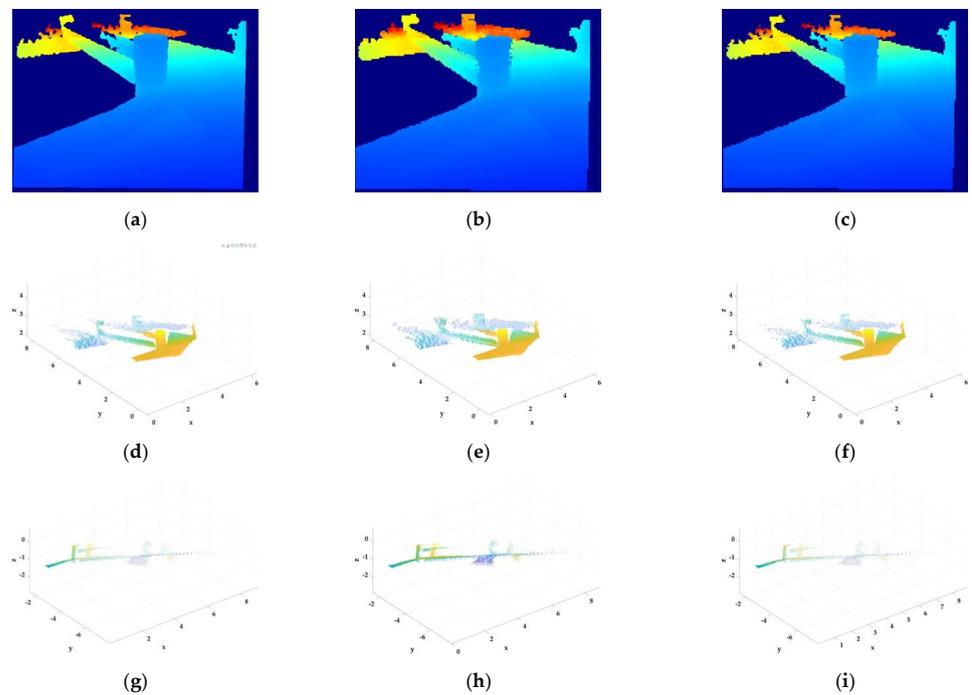


Figure 16. Validation of fractional-order differential Kriging interpolation super-resolution model (dataset no. 03528). (a,d,g) Original depth images; (b,e,h) Outputs of fractional-order differential Kriging interpolation super-resolution model ($\nu = 0.6$, PSNR = 33.677 dB); (c,f,i) Outputs of nearest-neighbour interpolation super-resolution model (PSNR = 25.825 dB).

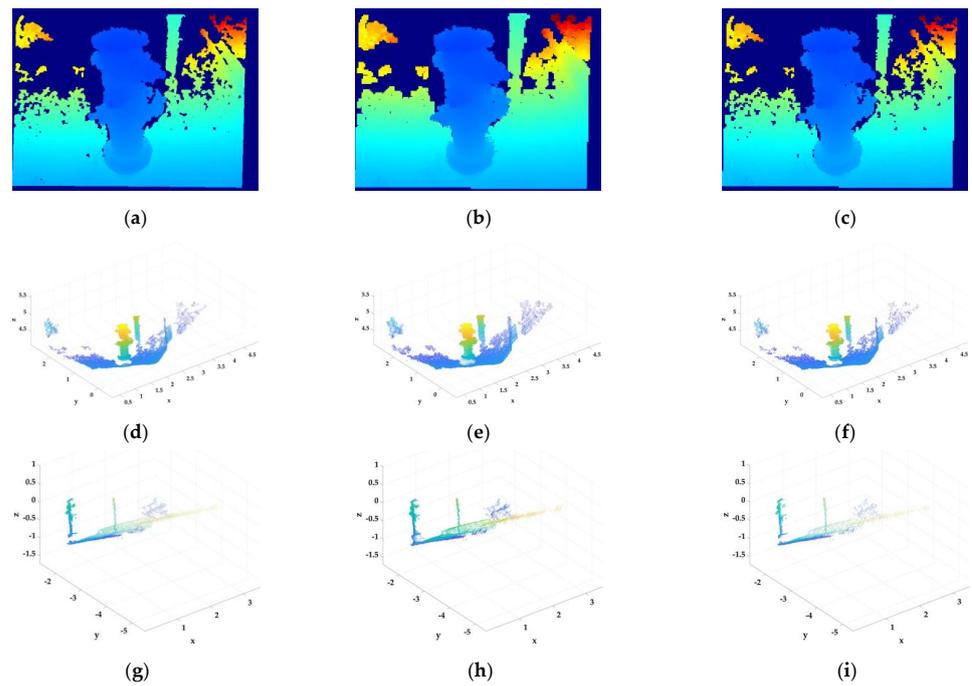


Figure 17. Validation of fractional-order differential Kriging interpolation super-resolution model (dataset no. 04797). (a,d,g) Original depth images; (b,e,h) Outputs of fractional-order differential Kriging interpolation super-resolution model ($\nu = 0.6$, PSNR = 40.595 dB); (c,f,i) Outputs of nearest-neighbour interpolation super-resolution model (PSNR = 22.456 dB).

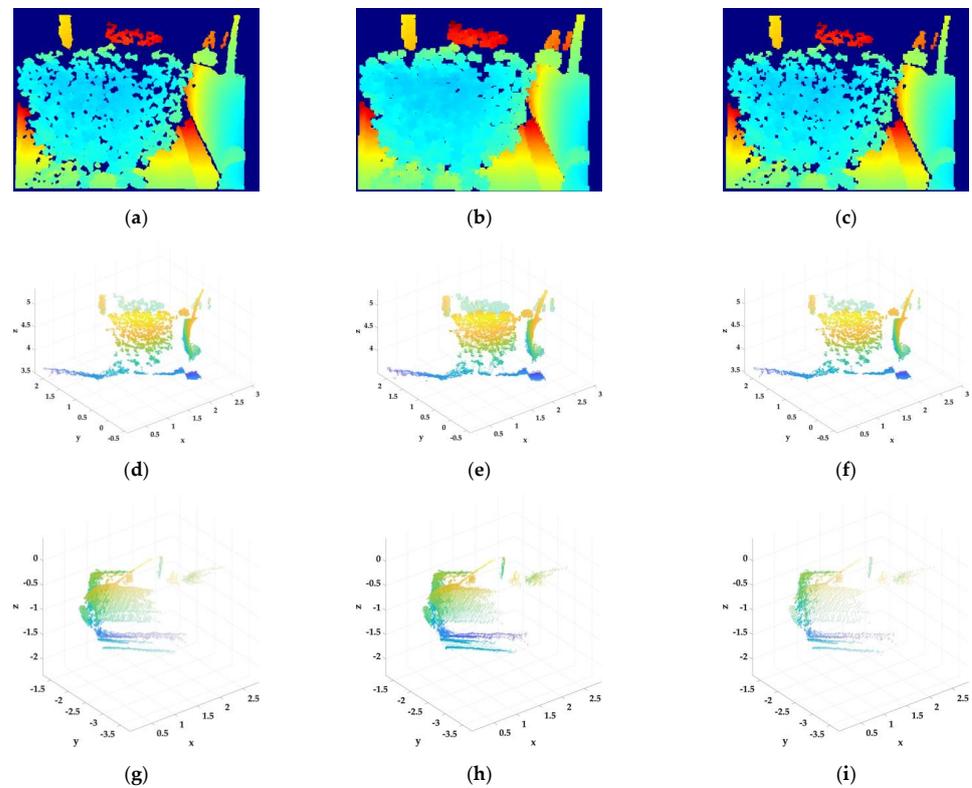


Figure 18. Validation of fractional-order differential Kriging interpolation super-resolution model (dataset no. 04797). (a,d,g) Original depth images; (b,e,h) Outputs of fractional-order differential Kriging interpolation super-resolution model ($\nu = 0.6$, PSNR = 33.819 dB); (c,f,i) Outputs of nearest-neighbour interpolation super-resolution model (PSNR = 21.051 dB).

Figures 14–18 show the super-resolution results after 25% down-sampling and 400% up-sampling, and it can be seen that the depth images after super-resolution meet the requirements for effective reconstruction and recognition. Moreover, for complex and variable data, such as the multiple bouquets in Figure 18 Kriging interpolation considerably outperforms nearest-neighbour interpolation.

5. Discussion

This study has shown that most of the classical image interpolation algorithms are not suitable for super-resolution of complex depth images with voids and rapid edge changes. Through comparisons of the point clouds of interpolated depth images, it is found that only nearest-neighbour interpolation of the classical image interpolation algorithms is applicable to depth images. The Kriging interpolation model, especially the variant based on spatial distance, can obtain better results that reflect the spatial characteristics to a greater extent. The introduction of fractional-order differential edge recognition successfully addresses the edge blurring problem of Kriging interpolation. The experimental results demonstrate the superiority of the improved Kriging interpolation based on fractional-order differentiation in terms of the super-resolution of depth images, which preserve texture information and edge properties. Objective evaluations show that the Kriging interpolation with fractional-order differential edge detection significantly improves the interpolation effect compared with the classical interpolation algorithm, with a PSNR improvement of 3–18 dB. Furthermore, the images produced by Kriging interpolation have more fractal characteristics, as would be expected for super-resolution based on interpolation. Overall, the Kriging interpolation super-resolution method based on fractional-order differential edge detection has better image reconstruction ability and stronger robustness than the classical interpolation algorithms.

While our proposed method shows promising results, there are still some limitations that should be addressed in future work. For instance, our current approach assumes that the depth images are noise-free, which may not be realistic in practise. Future work could explore the integration of denoising techniques with our approach to improve its robustness. Additionally, while we have demonstrated the effectiveness of our approach on synthetic depth images, its performance on real-world data should be further evaluated.

Overall, our work provides a valuable contribution to the field of depth image super-resolution and has the potential to be applied in a range of practical applications, such as 3D reconstruction, augmented reality, and medical imaging.

Author Contributions: Conceptualization, T.H., X.L. and C.W.; methodology, T.H. and X.W.; software, T.H. and Y.Y.; formal analysis, T.H. and X.W.; data curation, X.W. and Y.Y.; writing—original draft preparation, T.H. and X.W.; writing—review and editing, T.H., X.L. and C.W.; funding acquisition, X.L. and C.W. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by National Key R&D Program of China, grant number 2022YFC3803702.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kim, J.; Lee, H.; Kim, H.; Lee, J.; Yoo, H. Deep Laplacian Pyramid Networks for Fast and Accurate Super-Resolution. *IEEE Trans. Pattern Anal. Mach. Intell.* **2022**, *44*, 1569–1582. [[CrossRef](#)]
2. Dong, C.; Loy, C.C.; He, K.; Tang, X. Learning a Deep Convolutional Network for Image Super-Resolution. In *Computer Vision—ECCV 2014. ECCV 2014; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2014; p. 8692.* [[CrossRef](#)]
3. Li, Y.; Su, Y.; Guo, M.; Han, X.; Liu, J.; Vishwasrao, H.D.; Li, X.; Christensen, R.; Sengupta, T.; Moyle, M.W.; et al. Incorporating the image formation process into deep learning improves network performance. *Nat. Methods* **2022**, *19*, 1427–1437. [[CrossRef](#)]

4. Wang, X.; Chen, Z.; Jia, Y.; Yang, C. A novel interpolation-based image super-resolution algorithm for complex depth images. *Opt. Express* **2022**, *30*, 2292–2307. [[CrossRef](#)]
5. Eilers, P.H.C.; Ruckebusch, C. Fast and simple super-resolution with single images. *Sci. Rep.* **2022**, *12*, 11241. [[CrossRef](#)] [[PubMed](#)]
6. Mortazavi, M.; Gachpazan, M.; Amintoosi, M.; Salahshour, S. Fractional derivative approach to sparse super-resolution. *Vis. Comput.* **2022**, 1–18. [[CrossRef](#)]
7. Yarmohammadi, M.; Javadi, S. Piecewise Fractional Interpolation with Application to Fractional Differential Equation. *J. Sci. Comput.* **2021**, *86*, 18. [[CrossRef](#)]
8. Chen, X.X.; Qi, C. Single-image super-resolution via low-rank matrix recovery and joint learning. *Chin. J. Comput.* **2014**, *37*, 1372–1379.
9. Mei, K.; Jiang, A.; Li, J.; Liu, B.; Ye, J.; Wang, M. Deep residual refining based pseudo-multi-frame network for effective single image super-resolution. *IET Image Process.* **2019**, *13*, 591–599. [[CrossRef](#)]
10. Bätz, M.; Eichenseer, A.; Seiler, J.; Jonscher, M.; Kaup, A. Hybrid super-resolution combining example-based single-image and interpolation-based multi-image reconstruction approaches. In Proceedings of the 2015 IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015. [[CrossRef](#)]
11. Bengtsson, T.; Gu, I.Y.H.; Viberg, M.; Lindström, K. Regularized optimization for joint super-resolution and high dynamic range image reconstruction in a perceptually uniform domain. In Proceedings of the International Conference on Acoustics, Speech, and Signal Processing, Kyoto, Japan, 25–30 March 2012. [[CrossRef](#)]
12. Xiao, J.; Liu, E.; Zhao, L.; Wang, Y.F.; Jiang, W. Detail enhancement of image super-resolution based on detail synthesis. *Signal Process Image Commun.* **2017**, *50*, 21–33. [[CrossRef](#)]
13. Mathieu, B.; Melchior, P.; Oustaloup, A.; Ceyral, C. Fractional differentiation for edge detection. *Signal Process.* **2003**, *83*, 2421–2432. [[CrossRef](#)]
14. Pu, Y.F.; Zhou, J.L.; Yuan, X. Fractional Differential Mask: A Fractional Differential-Based Approach for Multiscale Texture Enhancement. *IEEE Trans. Image Process.* **2010**, *19*, 491–511. [[CrossRef](#)]
15. Yu, M.; Yu, K.; Han, T.; Wan, Y.; Zhao, D. Research on application of fractional calculus in signal analysis and processing of stock market. *Chaos Solitons Fractals* **2020**, *131*, 109468. [[CrossRef](#)]
16. Hu, J.R.; Pu, Y.F.; Zhou, J.L. Fractional Integral Denoising Algorithm. *J. Univ. Electron. Sci. Technol. China* **2012**, *41*, 706–711. [[CrossRef](#)]
17. Chen, L. Fractional integral denoising algorithm of image based on Grünwald-letnikov. *J. Shaanxi Univ. Technol.* **2018**, *2*, 39–44. [[CrossRef](#)]
18. Wang, X.Y.; Huang, P.C. Comparative Study of Interpolation Methods of Meteorological Factors Based on GIS. *Geomat. Spat. Inf. Technol.* **2020**, *43*, 4, 167–170. [[CrossRef](#)]
19. Wong, D.; Yuan, L.; Perlin, S. Comparison of spatial interpolation methods for the estimation of air quality data. *J. Expo Sci. Environ. Epidemiol.* **2004**, *14*, 404–415. [[CrossRef](#)]
20. Jin, G.D.; Liu, Y.C.; Niu, W.J. Comparison between Inverse Distance Weighting Method and Kriging. *J. Chang. Univ. Technol.* **2003**, *24*, 53–57. [[CrossRef](#)]
21. Xiao, H.; Zhang, Z.; Chen, L.; He, Q. An Improved Spatio-Temporal Kriging Interpolation Algorithm and Its Application in Slope. *IEEE Access* **2020**, *8*, 90718–90729. [[CrossRef](#)]
22. Gong, X.P.; Yang, Y.H. Ordinary Kriging algorithm applied for interpolation. *J. Northwest Univ.* **2008**, *27*, 2585–2592. [[CrossRef](#)]
23. Zhou, R.; Hu, W.; Luo, G.; Liu, X.; Fan, P. Quantum realization of the nearest neighbor value interpolation method for INEQR. *Quantum Inf. Process.* **2018**, *17*, 166. [[CrossRef](#)]
24. Mastylo, M. Bilinear interpolation theorems and applications. *J. Funct. Anal.* **2013**, *265*, 185–207. [[CrossRef](#)]
25. Li, P.; Liu, X. Bilinear interpolation method for quantum images based on quantum Fourier transform. *Int. J. Quantum Inf.* **2018**, *16*, 1850031. [[CrossRef](#)]
26. Malini, P. Interpolation Techniques in Image Resampling. *J. Int. J. Eng. Technol.* **2018**, *7*, 567–570. [[CrossRef](#)]
27. Keys, R. Cubic convolution interpolation for digital image processing. *IEEE Trans. Acoust. Speech Signal Process.* **1981**, *29*, 1153–1160. [[CrossRef](#)]
28. Zhang, S.; Yang, F.; Wan, S.; Di, P. Spherical Lanczos Interpolation in Planar Projection or Format Conversions of Panoramic Videos. *IEEE Access* **2020**, *8*, 9655–9667. [[CrossRef](#)]
29. Agathiyar, A.; Gowrisankar, A.; Priyanka, T.M.C. Construction of New Fractal Interpolation Functions through Integration Method. *Results Math. Results Math.* **2022**, *77*, 122. [[CrossRef](#)]
30. Wang, H.; Liu, J.; Liu, L.; Zhao, M.; Mei, S. Coupling technology of OpenSURF and Shannon-Cosine wavelet interpolation for locust slice images inpainting. *Comput. Electron. Agric.* **2022**, *198*, 107110. [[CrossRef](#)]
31. Lu, Y.; Zhang, X.; Pang, S.; Li, H.; Zhu, B. A robust edge-based template matching algorithm for displacement measurement of compliant mechanisms under scanning electron microscope. *Rev. Sci. Instrum.* **2021**, *92*, 033703. [[CrossRef](#)]
32. You, L.; Yang, X.; Pan, J.; Lee, T.Y.; Bian, S.; Qian, K.; Habib, Z.; Sargano, A.B.; Kazmi, I. Fast character modeling with sketch-based PDE surfaces. *Multimed. Tools Appl.* **2020**, *79*, 23161–23187. [[CrossRef](#)]
33. Sun, C.; Wang, N.; Jin, S.; Wang, M. Study on spatial interpolation method of marine sediment particle size based on geostatistics. *J. Coast. Res.* **2020**, *108*, 125–130. [[CrossRef](#)]

34. Xu, T.; Liu, Y.; Tang, L.; Liu, C. Improvement of Kriging interpolation with learning kernel in environmental variables study. *Int. J. Prod. Res.* **2022**, *60*, 1284–1297. [[CrossRef](#)]
35. Choi, S.; Zhou, Q.Y.; Miller, S.; Koltun, V. A Large Dataset of Object Scans. *arXiv* **2016**, arXiv:1602.02481. [[CrossRef](#)]
36. Liang, J.; Chen, P.X.; Wu, M. Research on an Image Denoising Algorithm based on Deep Network Learning. *J. Phys. Conf. Ser.* **2021**, *1802*, 032112. [[CrossRef](#)]
37. 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](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.