

東京大学大学院新領域創成科学研究科
環境システム学専攻

2021 年度
修士論文

InSAR 画像の勾配に基づくノイズの識別と定量的推定

(Identification and quantitative estimation of noise based on
gradients in InSAR images)

2021 年 7 月 21 日 提出

指導教員 愛知 正温 講師

張開元

47-196784

Content

1. Introduction.....	4
2. Background	6
2.1. The history of land subsidence in Kujukuri Plain of Chiba Prefecture.....	6
2.2. The theoretical basis of InSAR	7
2.3. Specific problems of InSAR images and solutions.....	10
3. Study area and materials.....	11
3.1. Study area	11
3.2. Study materials.....	12
3.2.1. Land subsidence model.....	12
3.2.2. Noise model	13
4. Methodology	15
4.1. Method 1: Identify noise based on gradient direction.....	15
4.1.1. Reasons for using 2D image processing methods.....	15
4.1.2. Source of inspiration for Method 1	15
4.1.3. The steps of Method 1	16
4.2. Method 2: Based on the gradient angle map, reduce the amount of irrelevant points	22
4.2.1. The origin of inspiration	22
4.2.2. Create planes, compare normals, and find abnormal points in the gradient angle map	27
4.2.3. Find the noise point by superimposing the value of the neighborhood of the abnormal point	27
4.3. Find the best coefficients for Method 1 and Method 2	29
4.3.1. Find the best coefficient for Method 1	29
4.3.2. Find the best coefficients for Method 2	30
4.3.3. Apply the best coefficients to Method 1 and Method 2, and compare the results	31
4.4. Infer the real amount of noise	32
4.5. Smooth the noise and verify.....	34
4.5.1. Traditional Gaussian filtering.....	34
4.5.2. Traditional Median filtering.....	35
4.5.3. My Gaussian filtering and Median filtering.....	36
4.6. Apply methods and strategies to InSAR in the study area.....	38
5. Results and discussion	40
5.1. Method 1's results	40
5.1.1. Use condition 1 to process model	40
5.1.2. Use condition 2 to process model	41
5.1.3. Use condition 1 and condition 2 to process model.....	43
5.2. Method 2's results	44
5.2.1. The result of the first step of Method 2.....	44
5.2.2. The result of the second step of Method 2	45

5.3.	Results of Method 1 && Method 2	46
5.4.	Results of Best coefficients of Method 1 and Method 2	47
5.4.1.	Results of Best coefficient of Method 1	47
5.4.2.	Results of Best coefficient of Method 2	50
5.4.3.	Results of Best coefficients of Method 2 and Method 1	53
5.5.	The result of inferring the real amount of noise	55
5.6.	RMS comparison of the four methods and data cross-section comparison	56
5.7.	The result after processing through the smoothing strategies	62
5.8.	The result of applying methods and strategies in InSAR of study area	73
6.	Additional discussion	77
6.1.	About the accuracy of the method of deriving the amount of noise	77
6.1.1.	Change the scope of noise	77
6.1.2.	Change the standard deviation of noise	78
6.2.	About the magic number “60%”	79
7.	Conclusions	84
8.	Acknowledgments	85
9.	Reference	86

1. Introduction

Technological progress in the 20th century led to a significant increase in the world's population, prompting a rapid concentration of population in urban areas. This in turn led to a new environmental problem that occurred entirely in the city. In many coastal cities in Asia, the population is rapidly concentrating in urban areas, leading to excessive exploitation of groundwater, causing large-scale problems such as land subsidence and salt damage(Hayashi et al, 2009).

After ground subsidence was first discovered in Tokyo in 1910, a large amount of groundwater was pumped into the factory to adapt to the growing population. Over the next few decades, the groundwater level dropped to 58 meters below sea level. The pumping volume continued to increase until 1970, reaching a peak of nearly 1.5 million cubic meters (m³/d) per day. Over the past few decades, the depth of subsidence has continued to increase, and the disaster-stricken areas have continued to expand(Fig. 1). Although land subsidence exceeded 10 cm (cm/year) per year and reached approximately 24 cm/year in 1968, the Tokyo Metropolitan Government has introduced pump regulations for thousands of wells in the area. Slow down and reverse the pace of land subsidence. The amount of water pumped is reduced, and the sinking rate drops sharply. The groundwater level began to rise again in the early 1970s and is now 6-10 meters (m) below sea level. Even in the worst-hit areas, the sinking rate has slowed to about 1 cm/year in the last five years(Sato et al, 2006).

In the Kujukuri area in Chiba Prefecture, Japan, underground brine has been mined for more than 50 years. The dissolved methane and iodine in the water caused serious ground subsidence(MURAI et al, 2010).

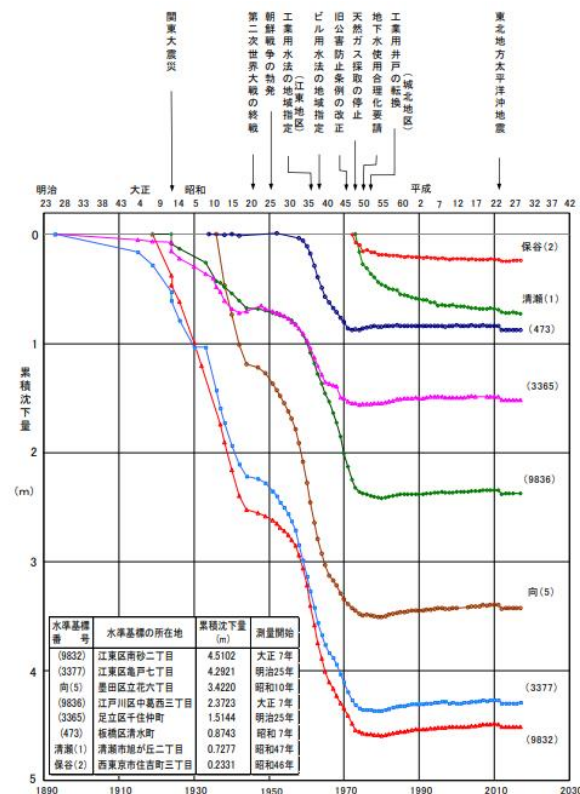


Fig. 1 Cumulative subsidence map of major level indicators (平成 28 年地盤沈下調査報告書)

Continuous land subsidence causes remarkable economic losses in the form of building damages leading to high maintenance costs(Raspini et al, 2014). Thus, identifying land deformation trends is a crucial task to maintain the sustainability of coastal urban areas(Aimaiti et al, 2018) .

In the past two decades, through the use of interferometric synthetic aperture radar (InSAR) technology (Cianflone et al, 2015), land subsidence monitoring has been significantly improved. Traditional methods (Global Positioning System (GPS) and leveling) can also be used for accurate measurement, but they cannot obtain large-scale coverage and high-density ground displacement measurement in a short time and at low cost(Hsieh et al, 2011). Advanced time series InSAR technology, such as continuous scattering interferometry (PSI) and small baseline subscript (SBAS) technology, can obtain better spatio-temporal resolution results with higher accuracy(Ferretti et al, 2010). In addition, the increase in available Synthetic Aperture Radar (SAR) satellites with different temporal and spatial resolutions provides researchers with a good opportunity to combine the observations of these satellites for long-term geological disaster monitoring(Armas et al, 2017).

In the synthesis process of InSAR images, it uses two high-resolution complex SAR images of the same scene to generate an interferogram. Then, the phase information contained in this interferogram is extracted to generate a digital-elevation model (DEM). As this phase information is wrapped within the interval of $[-\pi, \pi)$, it needs to be unwrapped before the estimation of height information(Meng D in 2007). However, the presence of phase noise not only interferes with the phase-unwrapping process, but it also affects the quality of topographic height information obtained from the interferogram. The spatial noise includes atmospheric conditions, ground surface conditions, leaves and so on. Some simple filtering can reduce spatial noise, such as median filtering, but the effect is not good enough. Several techniques have been proposed in the literature to reduce interferometric phase noise. One approach, applied to the complex interferometric signal, is the multilook filter(M. S. Seymour et al, 1994), which averages the values of neighboring pixels in the phase image. This technique, however, reduces noise at the expense of spatial resolution. Lee et al(J. S. Lee et al, 1998). uses a local statistics filter applied to the real and the complex interferometric phase signal as well. Techniques for phase-noise reduction in the Fourier domain have also been proposed(R. Goldstein et al, 1997). All these techniques, however, involve the loss of image detail to a certain extent.. Some trend analysis techniques for time series analysis have removed time noise(Ansari et al, 2017).

There is a close relationship between spatial noise and spatial location, and gradient can reflect this relationship. If the noise can be identified, finding the location of the noise will have a positive effect on smoothing the noise. Based on the direction of gradient descent, this research proposes a method that can identify the location of noise and infer the amount of noise, and combines and improves some traditional smoothing methods to smooth InSAR images

2. Background

2.1. The history of land subsidence in Kujukuri Plain of Chiba Prefecture

Land subsidence causes various damages to civil society, and due to the irreducibility of ordinary land subsidence, it may permanently deteriorate environmental conditions (Deming et al,2002). Due to the extraction of underground fluid resources such as dissolved methane and groundwater, land subsidence has occurred in many areas of Japan (Ministry of the Environment, Environmental Administration, 2009).

The Kujukuri Plain in Chiba Prefecture (Fig. 2) is one of the places where 100.7 cm of land subsidence was observed from 1969 to 2007 (Chiba Prefecture Environmental Bureau, 1970-2007). This coastal plain consists of alternating beach ridges and back swamps (Moriwaki et al, 1979); (Fig. 2). Most areas are located below 10 m above sea level, which may increase the possibility of continuous ground subsidence due to the extraction of groundwater and future global warming leading to sea level rise, thereby increasing the risk of flood disasters and coastal erosion. I am worried. However, understanding the changes in the surface environment caused by land subsidence is not enough to predict the degree of environmental impact and propose effective countermeasures(NAKAMURA T in 2010).

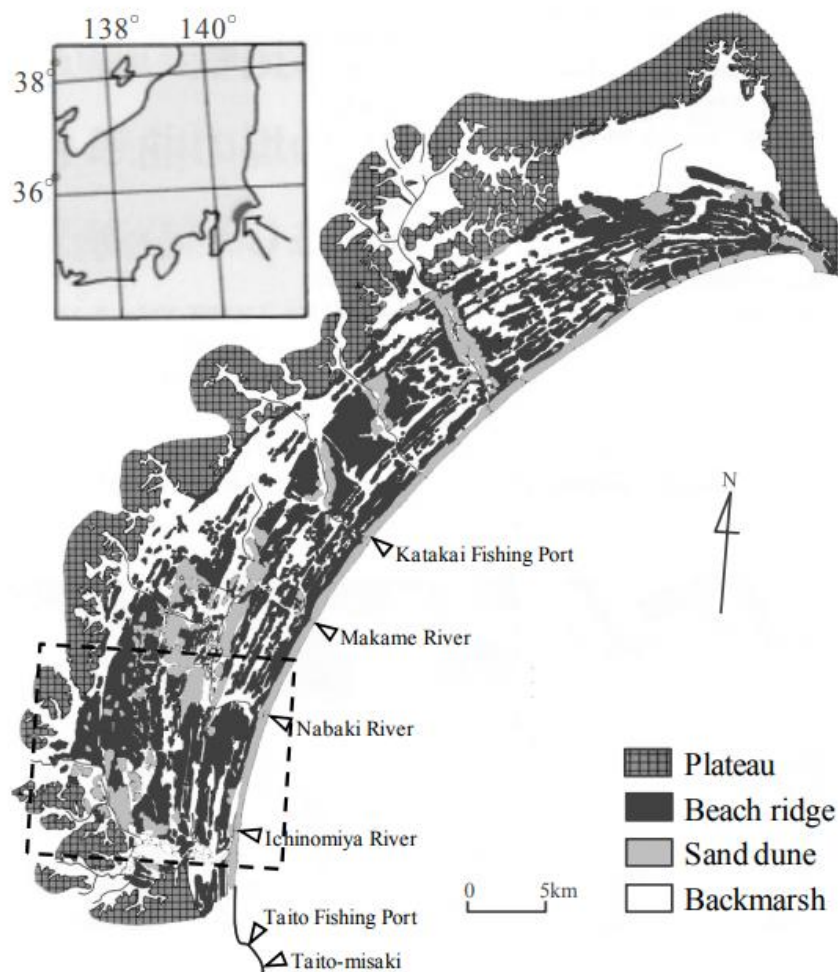


Fig. 2 Geomorphological map of the Kujukuri Plain (modified after Moriwaki, 1979).

Fig. 3 explains the lowland area of square marked by dashed line in Fig. 2. Fig. 3 (a) shows the expansion of the lowlands over time. The area less than 1 m has increased from 2.4% of the total study area in 1969 to 5.5% in 2009. The area less than 2-6 m grows faster. For example, between 1969 and 2009, the area less than 6 m increased by 5.6%. The lowlands extend along the Nabaki River, partly in the wetlands behind the beach ridge of the coastline (Fig. 3 (b)). However, the overall spatial pattern of topography such as beach ridges and back wetlands has been maintained. In other words, the regional differences in land subsidence in the past 40 years have not caused significant changes in the overall topography.

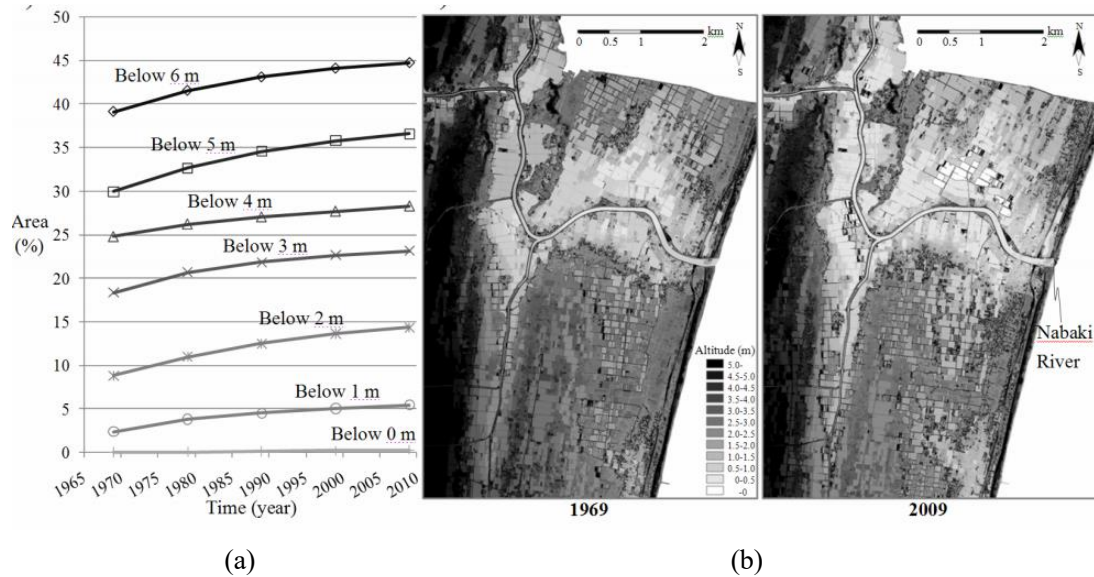


Fig. 3 Temporal change of landform in the past 40 years. (a) Altitude-area relations; and (b) an example of the landform change at the lower reach of the Nabaki River.

2.2. The theoretical basis of InSAR

Interferometric synthetic aperture radar (InSAR), is a radar technique used in remote sensing. This geodetic method uses two or more synthetic aperture radar (SAR) images to generate maps of surface deformation or digital elevation, using differences in the phase of the waves returning to the satellite.

The technique can potentially measure millimetre-scale changes in deformation over spans of days to years. It has applications for geophysical monitoring of natural hazards, for example earthquakes, volcanoes and landslides, and in structural engineering, in particular monitoring of subsidence and structural stability (Fig. 4).

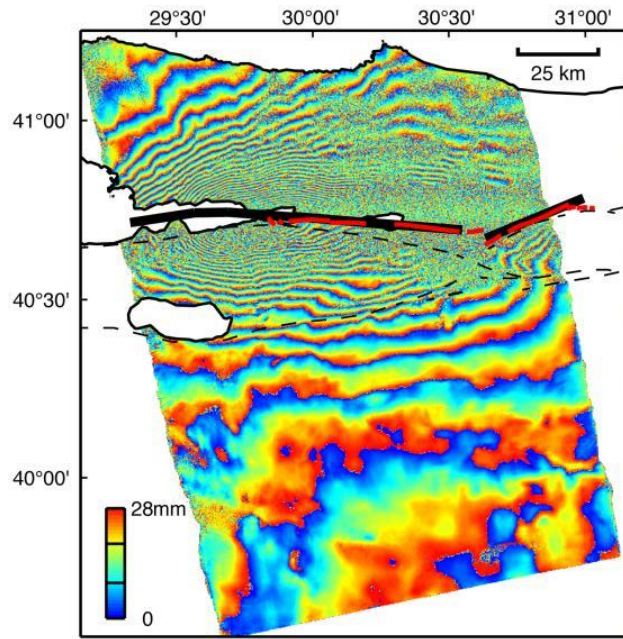


Fig. 4 An example of InSAR An example of

(https://en.wikipedia.org/wiki/Interferometric_synthetic-aperture_radar)

The acquisition of InSAR images can be divided into 4 steps:

- 1.Data preprocessing.This step is mainly to obtain the master image and slave image and read their information.
- 2.SAR image registration.In the process of interferometry, the coherent pixels of the two images always have a certain offset in the distance. So we need to register the two images and pre-filter.
- 3.Calculate the interference phase of SAR image.After calculating the phase, remove the additional phase caused by the flat effect^[1] and filter the phase.
- 4.Phase unwrapping.Unwrap the phase and calculate the height of the pixel points to reconstruct the elevation model.

The Fig. 5 shows how the satellite-borne radar measures a displacement and the explanation is as follows:

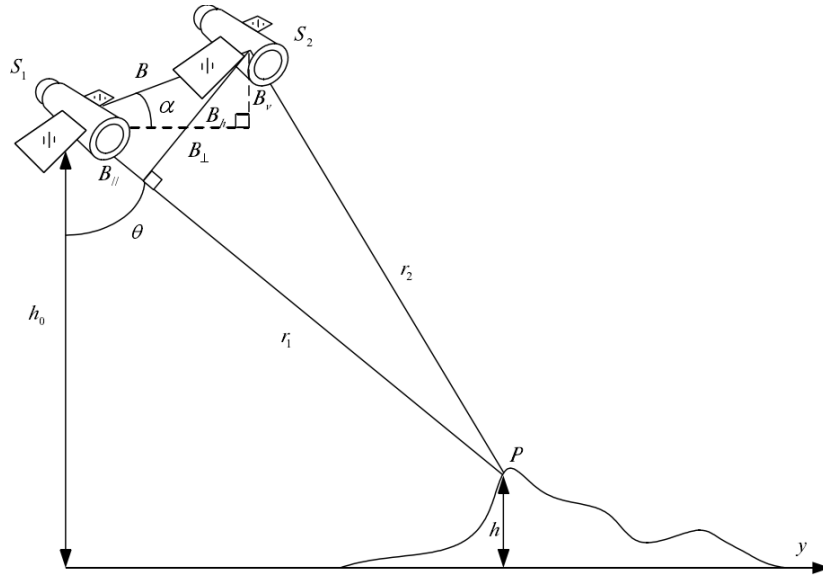


Fig. 5 Use satellite-borne radar to acquire InSAR

Satellite-borne radar is usually used to obtain InSAR images. The satellite is loaded with a master radar and a slave radar. The two radars emit lasers to the same area to obtain interference images. The principle is as follows(formula (1)~(8)):

$$h = h_0 - r_1 \cos \theta \quad (1)$$

$$r_2^2 = r_1^2 + B^2 - 2r_1 B \cos\left(\frac{\pi}{2} - \theta + \alpha\right) \quad (2)$$

$$r_2 - r_1 = \frac{B^2 - 2r_1 B \cos\left(\frac{\pi}{2} - \theta + \alpha\right)}{r_2 + r_1} \quad (3)$$

$$r_2 - r_1 \approx -B \cos\left(\frac{\pi}{2} - \theta + \alpha\right) \quad (4)$$

$$\theta = \alpha + \frac{\pi}{2} - \cos^{-1}\left(\frac{r_1 - r_2}{B}\right) \quad (5)$$

$$h = h_0 - r_1 \cos\left[\alpha + \frac{\pi}{2} - \cos^{-1}\left(\frac{-\lambda\phi}{4\pi B}\right)\right] \quad (6)$$

$$\phi = \varphi + \phi_c = \frac{4\pi}{\lambda}(r_2 - r_1) \quad (7)$$

$$\varphi = \arg(s_1 \times s_2^*) = \text{mod} \left[\frac{4\pi}{\lambda} (r_2 - r_1), 2\pi \right] \quad (8)$$

Combining the formula and Fig. 3, we can see that we need to get h through some known variables. Formulas (1)~(5) can intuitively calculate the angle value, and through formulas (6)~(8), we can calculate the phase of entanglement according to the angle. Then the entanglement is resolved to obtain the interferogram.

The above principles and steps will not be used in this study. In this study, the InSAR displacement map after unwrapped is used.

2.3. Specific problems of InSAR images and solutions

The first problem is the spatial noise in the InSAR image. The InSAR images we get are all noisy, but we don't know the distribution, amount, and location of the noise. Traditional methods cannot identify the distribution, location and amount of noise. It just smooths all the points of the InSAR image, so the smoothing accuracy is not enough, and it may cause over-smoothing. The main purpose of this article is to identify the location and distribution of noise and infer the amount of noise.

The second question is about verification. Because we only have noisy InSAR images and no original InSAR model without noise, we cannot verify the pros and cons of the method.

The solution for this article is to use virtual authentication. In other words, the digital land subsidence model and the artificially determined noise model are used to verify the method. It will be explained in detail in the following chapters.

3. Study area and materials

3.1. Study area

My study area is the Kujukuri plain in Chiba Prefecture, Japan, with an area of 23.72km^2 . The geographical location is as follows(Fig. 6):



Fig. 6 The location of Kujukuri plain in Chiba Prefecture

The range of the InSAR image I got covers the entire Chiba Prefecture, so when applying my method to the InSAR image, I will use the InSAR image of the entire Chiba Prefecture. The InSAR image of Chiba Prefecture is as follows(Fig. 7):

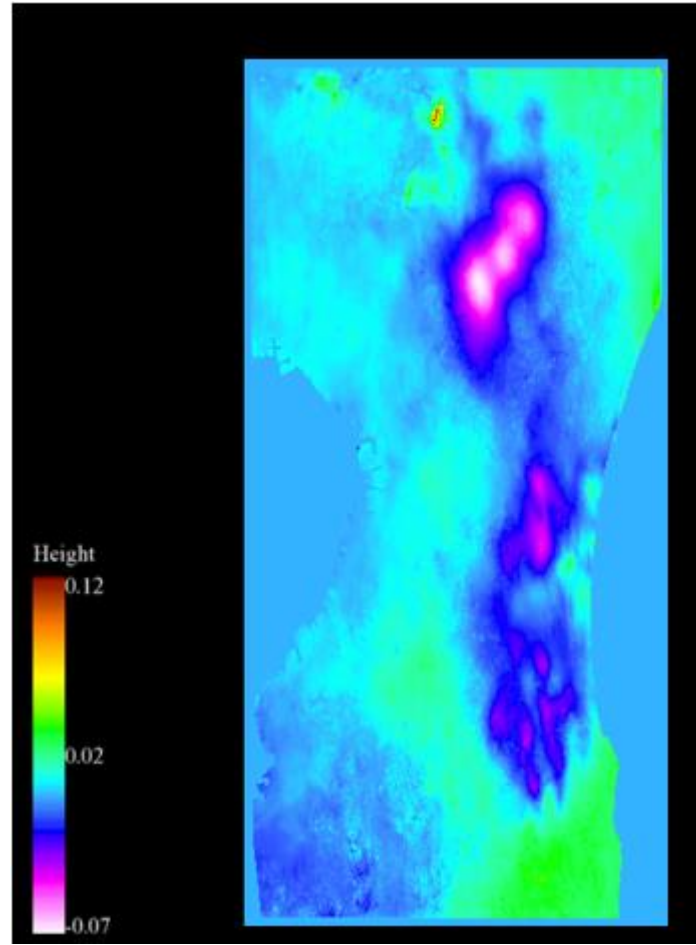


Fig. 7 InSAR displacement image of the Kujukuri Plain in Chiba Prefecture on May 7, 2019
(Compared with the InSAR image on January 15, 2015, the displacement at this time is 0)

This InSAR image contains 10 million points, and each point contains x, y, and z coordinate information. The x and y values represent the geographic location, and the z value represents the displacement of each point.

3.2.Study materials

Since I use a virtual verification method, I need a numerical model of land subsidence and a noise model.

3.2.1. Land subsidence model

The numerical model used in this study is only one well in uniform isotropic half-space/vertical displacement(Aichi ,2017), the model is as follows(formulas (9)~(10)):

$$u_z \approx \int_0^\infty S_s h dz = \frac{Q}{4\pi c} \int_0^t \frac{1}{\tau} \exp\left(-\frac{r^2}{4c\tau}\right) d\tau \quad (9)$$

where,

$$\begin{aligned} h &: \text{drawdown} & u_z &: \text{subsidence} \\ K &: \text{hydraulic conductivity} & S_s &: \text{specific storage} \\ Q &: \text{pumping rate} & z' &: \text{depth of screen} \\ c &: \text{hydraulic diffusivity} & & (= K / S_s) \end{aligned} \quad (10)$$

Formula (9) shows the formula for calculating land subsidence, and (10) explains the meaning of these variables. Below I list the parameters of these variables in this study.

Conditions of “observation” (Aichi ,2017):

Wave length:0.24m

10 images of 48 days cycle

10m * 10m resolution,200 * 200 grids

And as for the fitting “model parameter”:

Hydraulic diffusivity ($0.1 \text{ m}^2/\text{s}$)

Pumping rate ($1000 \text{ m}^3/\text{day}$)

In this study, I will use three wells as the source of ground subsidence. Their positions are P1 (1000, 1000), P2 (500, 800) and P3 (1714, 444). The numerical InSAR model obtained by the formula is as follows(Fig. 8):

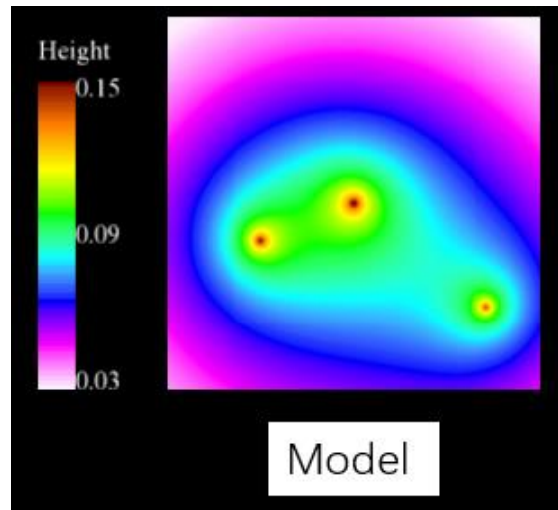


Fig. 8 Numeric land subsidence model

3.2.2. Noise model

For the noise model, I will use normally distributed noise, from 0% to 100%. The most important parameter is the standard deviation. Here I will set it as:

Standard Deviation=Well length * 0.005

Well length=0.24 m

The noise model of 100% noise amount is as follows(Fig. 9):

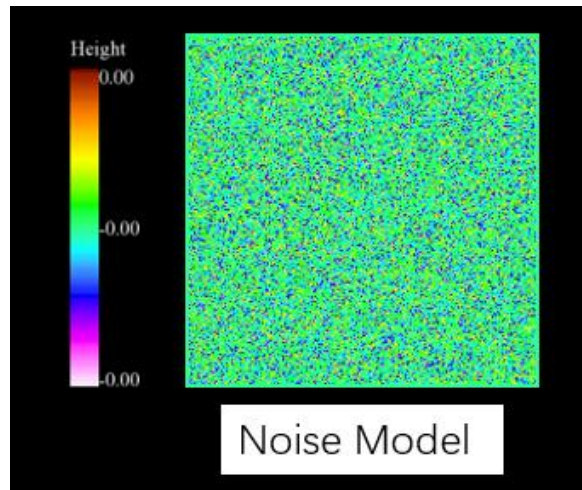


Fig. 9 Noise model(100% amount of noise)

Superimpose the noise model on the numerical model of land subsidence to get the noise InSAR model(Fig. 10):

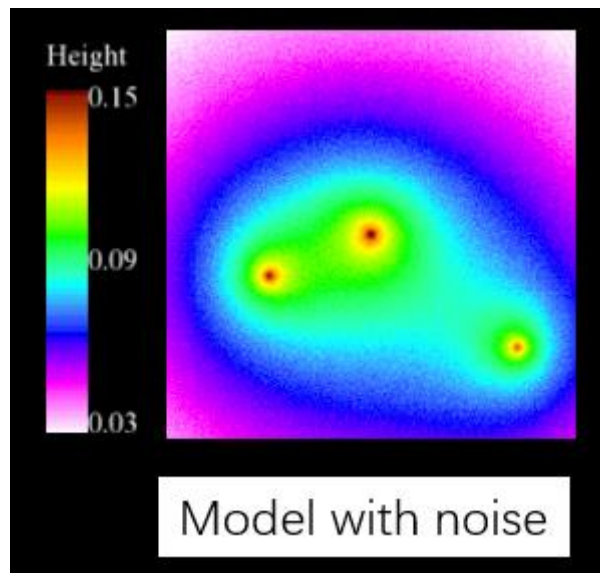


Fig. 10 InSAR model with noise

4. Methodology

4.1. Method 1: Identify noise based on gradient direction

I named this part of the method as “**Method 1**”, because there will be other methods to combine later. Method 1 is the most important way to identify noise. I will explain in detail “why I use 2D image processing method”, “where I got the inspiration” and “the steps of method1”.

4.1.1. Reasons for using 2D image processing methods

Compared with traditional lidar, InSAR is more like 2D images, so I think I could learn more from 2D image processing and get inspiration instead of 3D images.

Lidar images are usually acquired by equipment carried by drones, so their x and y coordinates are usually disorderly and chaotic. Lidar also lacks data for a long time due to Iather, clouds and liquids.

For InSAR, InSAR images are usually acquired by satellite-mounted equipment to measure the displacement of a certain area in a certain period of time. Therefore, the x and y coordinates of the InSAR image are in order. In an ordered coordinate system, I can also treat it as a 2D image for processing.

The difference between InSAR and 2D images is the z coordinate. The z coordinate of a 2D image represents RGB or other color spaces, has a specific range, and the value type is an integer. The z coordinate of the InSAR image represents displacement, no range, the value type is a floating point number, and the magnitude of data is usually very small.

For 2D images, I think the two concepts of noise and boundary have very similar characteristics. The similarity between noise and boundary is that they are significantly different from the values of surrounding points.

The difference between noise and edge is that noise points usually exist independently, and the value of each point is different, and for an edge, the boundary points should be continuous and similar in value.

4.1.2. Source of inspiration for Method 1

One can get inspiration from the edge extraction algorithm of 2D images. There are many classic edge extraction algorithms, such as "Canny edge extraction algorithm", "Sobel algorithm", and "Laplace algorithm".

What these algorithms have in common is that they all use some operators to perform calculations, such as the sobel operator(Fig. 11), the Prewitt operator(Fig. 12), and the Laplacian operator(Fig. 13). It can be found that the commonality of these operators is that they all obtain the result by calculating the value of the target point and the value of the neighborhood point, and the difference lies in their calculation method and subsequent processing.

-1	0	1
-2	0	2
-1	0	1

-1	-2	-1
0	0	0
1	2	1

Fig. 11.Sobel operator

1	1	1
0	0	0
-1	-1	-1

-1	0	1
-1	0	1
-1	0	1

Fig. 12. Prewitt operator

0	1	0
1	-4	1
0	1	0

Fig. 13. Laplacian operator

So I can know that the concept of "neighborhood" is essential to extract edges or find noise, and I think that in time domain space, I can only find noise through neighborhood.

In terms of InSAR gradient calculation, I used the "Canny Edge Extraction Algorithm(Kanopoulos et al, 1988)" for reference. It uses the sobel operator to calculate the gradient value of each point, and its steps are as follows:

1. Perform Gaussian smoothing on the input image to reduce the error rate.
2. Calculate the gradient magnitude and direction to estimate the edge strength and direction at each point.
3. Perform non-maximum suppression on the gradient amplitude according to the gradient direction. Essentially, it is a further refinement of the results of operators such as Sobel and Prewitt.
4. Use dual thresholds to process and connect edges.

Here, I only used the second step "Calculate the gradient magnitude and direction to estimate the edge strength and direction at each point". The reason is that the purpose of the Canny algorithm is to extract edges, and its other steps will eliminate noise, but my purpose is the opposite, I am to extract noise, so I only need to learn from the second step here, and then perform my own processing.

4.1.3. The steps of Method 1

The process of Method 1 is as follows:

1. I apply the second step of Canny's algorithm to InSAR images. Calculate dx and dy like the formula (13),(14),through the convolution of the soble horizontal(11) and vertical operators(12) with the input image.

$$Sobel_X = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} * \begin{bmatrix} 1 & 2 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad (11)$$

$$Sobel_Y = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (12)$$

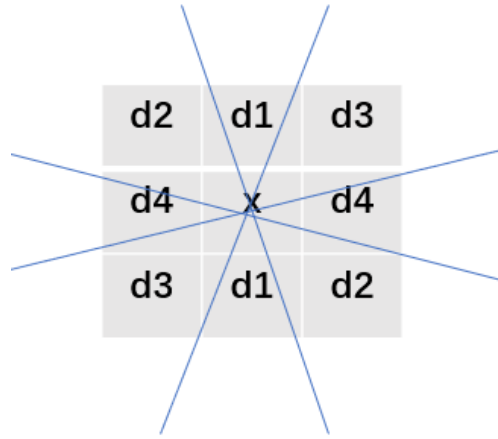
$$d_x = f(x, y) * Sobel_x(x, y) \quad (13)$$

$$d_y = f(x, y) * Sobel_y(x, y) \quad (14)$$

2. Calculate the gradient amplitude through formula (15) and angle through formula (16) of each point, and simplify the angle direction(Fig. 14(a)) into 4 directions according to different angles of points(Fig. 14(b)).

$$M(x, y) = \sqrt{d_x^2(x, y) + d_y^2(x, y)} \quad (15)$$

$$\theta_M = \arctan(d_y/d_x) \quad (16)$$



(a)

$$\mathbf{d1:} \quad \theta_M \in [0, 22.5) \cup (-22.5, 0] \cup (157.5, 180] \cup (-180, 157.5]$$

$$\mathbf{d2:} \quad \theta_M \in [22.5, 67.5) \cup [-157.5, -112.5)$$

$$\mathbf{d3:} \quad \theta_M \in [67.5, 112.5) \cup [-112.5, -67.5]$$

$$\mathbf{d4:} \quad \theta_M \in (112.5, 157.5] \cup [-67.5, -22.5]$$

(b)

Fig. 14.(a) Simplified gradient angle.(b) The angle of the simplified gradient angle map

3. The above steps 1 and 2 are the contents of the Canny algorithm, and the next step is my own processing. Let me use two examples to illustrate the difference in gradients between noise-free InSAR images and noisy InSAR images. Example 1(Fig. 15(a)) is an InSAR image without noise. It can be seen that the data decreases regularly from left to right in the horizontal direction. According to the Sobel operator, it can be calculated that in the X direction(Fig. 15(b)), the gradient value of all points is -8; in the Y direction(Fig. 15(c)), the gradient value of all points is 0. According to the gradient direction formula, the gradient direction(Fig. 15(d)) of all points is 0 degrees , which is directly on the right. The magnitude of the gradient of all points is 8(Fig. 15(e)).All points have the same characteristics.

7	6	5	4	3
7	6	5	4	3
7	6	5	4	3
7	6	5	4	3
7	6	5	4	3

(a)

-8	-8	-8	-8	-8
-8	-8	-8	-8	-8
-8	-8	-8	-8	-8
-8	-8	-8	-8	-8
-8	-8	-8	-8	-8

(b)

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

(c)

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

(d)

8	8	8	8	8
8	8	8	8	8
8	8	8	8	8
8	8	8	8	8
8	8	8	8	8

(e)

Fig. 15 Example 1.(a) InSAR image(without noise).(b)Gradient in x direction.(c) Gradient in y direction.(d) Gradient angle direction.(e)Gradient modulus length

Next, I use another example to illustrate and compare. The InSAR image of Example 2(Fig. 16(a)) is similar, except that the value of the center point is added with a noise value. According to the Sobel operator, the gradient value in the x direction(Fig. 16(b)) can be calculated. From the result, it can be seen that the gradient value on both sides of the center point has changed. Regarding the gradient value in the y direction(Fig. 16(c)), the gradient value on the vertical sides of the center point changes. After calculating the result of the gradient angle(Fig. 16(d)), it can be found that the gradient angle only changes on the vertical sides of the center point. I think this is because the overall data of the InSAR image is decreasing from left to right in the horizontal direction. For the gradient magnitude(Fig. 16(e)), the values of all 8 points around the center point have changed, except for the center point itself.

7	6	5	4	3	-8	-8	-8	-8	-8
7	6	5	4	3	-8	-3	-8	-13	-8
7	6	10	4	3	-8	2	-8	18	-8
7	6	5	4	3	-8	-3	-8	-13	-8
7	6	5	4	3	-8	-8	-8	-8	-8

(a)

(b)

0	0	0	0	0	0	0	0	0	0
0	5	10	5	0	0	-57	-51	-20	0
0	0	0	0	0	0	0	0	0	0
0	-5	-10	-5	0	0	57	51	20	0
0	0	0	0	0	0	0	0	0	0

(c)

(d)

8	8	8	8	8
8	5.8	12.8	13.9	8
8	2	8	18	8
8	5.8	12.8	13.9	8
8	8	8	8	8

(e)

Fig. 16 Example 2.(a) InSAR image(without noise).(b)Gradient in x direction.(c) Gradient in y direction.(d) Gradient angle direction.(e)Gradient modulus length

By comparing the data of Example 1 and the data of Example 2, the following conclusions can be drawn:

1. The gradient angle of the center point will not change even if the noise value is added, because in the Soble operator, the calculation of the gradient is only related to the neighborhood value. So I can use this conclusion to find out the normal data flow direction around the noise point.

2. For the direction of gradient descent in Example 2, combined with the value of the InSAR image and the displacement angle, I can find that the gradient direction of the center point is horizontal to the right. In other words, the magnitude of the data value should decrease horizontally from left to right, so the value of the left point of the center point is greater than the value of the right point. HoIver, due to the noise value, their values are less than the value of the center point or both are greater than the value of the center point(Fig. 17).

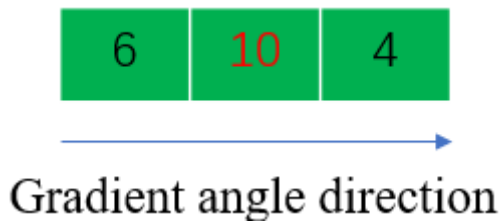


Fig. 17 The center point and the points on the horizontal sides of the center point in Example 2

3. For the orthogonal direction of the gradient descent in Example 2, combined with the value of the InSAR image and the displacement angle, I can find that the orthogonal direction is the vertical direction. In other words, the data values in this direction should be equal or similar (very slight noise should be accepted). HoIver, due to the noise, the values on both sides of the noise value are obviously less than the noise value or lgreater than the noise value(Fig. 18).

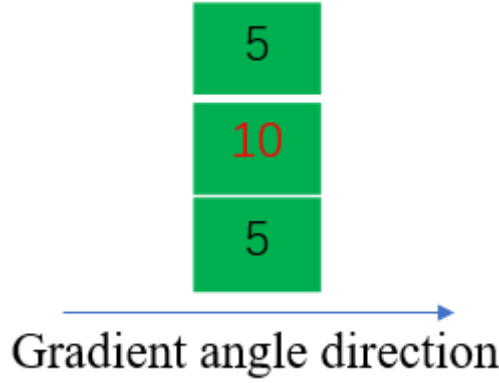


Fig. 18 The center point and the points on the vertical sides of the center point in Example 2

Based on the above conclusions, I summarized two characteristics of noise points. One feature is in the direction of gradient descent, and the other is in the direction of gradient descent. The two features are similar in nature, but in different directions. The following is based on these two characteristics to create two formulas to find the noise.

Formula 1. In the direction of gradient descent, referring to fig 7, the following formula (17)~(19) will find the noise:

$$V_{average} = 0.5 \times (V_{left} + V_{right}) \quad (17)$$

$$V_{keypoint} < (1 - k) \times V_{average} \text{ Or } M_{keypoint} > (1 + k) \times V_{average} \quad (18)$$

$$k \in R, k \geq 0 \&\& k \leq 1 \quad (19)$$

V_{left} and V_{right} represent the value of the center point on both sides of the gradient descent direction, take the average, and multiply it by the coefficient related to K , and compare the result with the value of the center point. The value of K is between 0 and 1.

Formula 2. Formula 2 is almost the same as Formula 1, the difference is that the direction of formula 2 is the orthogonal direction of gradient descent, refer to fig 8. It is expressed by the following formulas (20)~(22):

$$V_{average} = 0.5 \times (V_{up} + V_{down}) \quad (20)$$

$$V_{keypoint} < (1 - K_1) \times V_{average} \text{ || } V_{keypoint} > (1 + K_1) \times V_{average} \quad (21)$$

$$K_1 \in R, K_1 \geq 0 \&\& K_1 \leq 1 \quad (22)$$

V_{up} and V_{down} represent the values on both sides of the center point in the orthogonal direction of the gradient descent. The remaining content is consistent with formula 1.

It is worth mentioning that the value of K in Method 1 and Method 2 is used here. I use the same value of k to apply to the two methods. The reason is that other formulas and coefficients will appear in the subsequent processing. In order to simplify the calculation and reduce the uncertainty and complexity of the calculation, I use the same K value. The specific value of K will be discussed in the 4.3 chapter

Another point to note is that there will be boundary problems when using operators to perform convolution operations. In other words, how to deal with the points on the boundary is a problem (because the amount of neighbor points of the boundary point is less than inner points). In this article, my approach is to ignore the points of the outermost layer. So in Method 1, I ignored the points on the outermost layer.

4.2. Method 2: Based on the gradient angle map, reduce the amount of irrelevant points

The content of this part will be collectively referred to as “**Method 2**”. In the processing results of method 1, it can be seen that it accurately finds about 90% of the noise points (Can be viewed from the “**Results**” section), but it also includes some irrelevant points. In order to further improve the accuracy of the results, I use Method 2 to reduce the number of irrelevant points.

4.2.1. The origin of inspiration

From example 1 (Fig. 5) and example 2 (Fig. 6), I have reached the conclusion and method of Method 1. In addition, I found that there is other information available in Example 1 and Example 2. This information is related to the gradient angle, I will use Example 3 (Fig. 19) and Example 4 (Fig. 20) to illustrate.

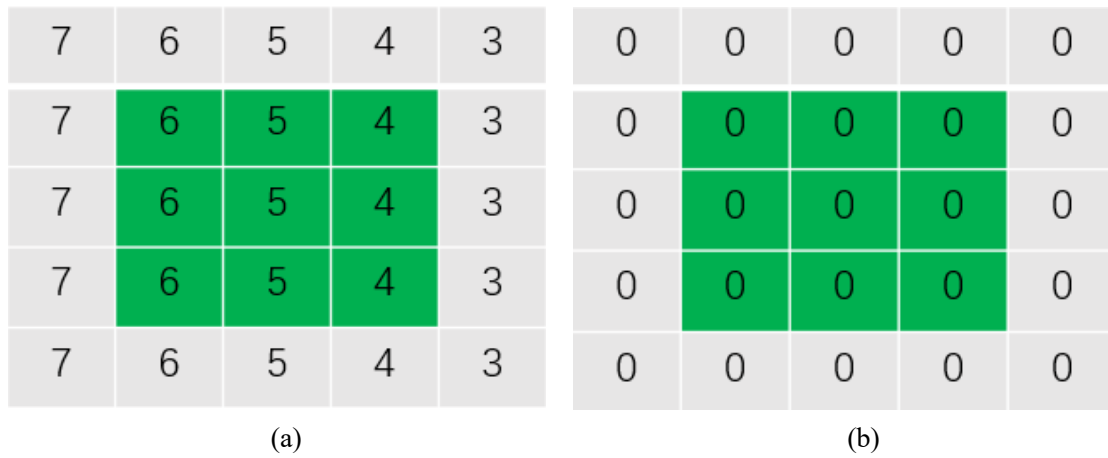


Fig. 19 Example 3.(a) InSAR image(without noise).(b)Gradient angle map

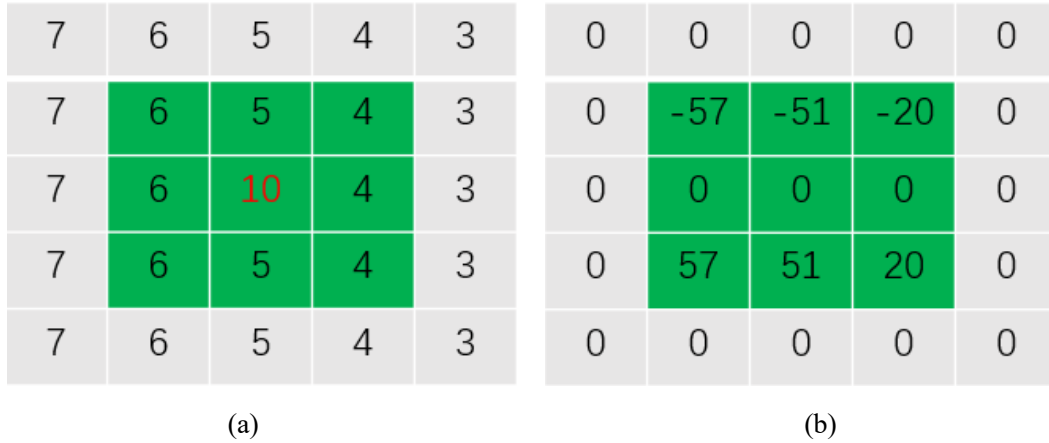


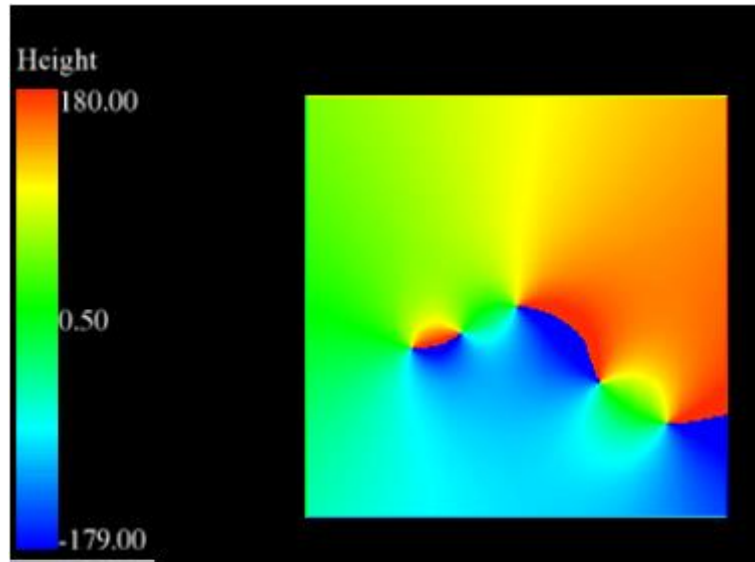
Fig. 20 Example 4.(a) InSAR image(with noise).(b)Gradient angle map

The InSAR image of Example 3 (Fig. 19(a)) is noise-free, and the InSAR image of Example 4 (Fig. 20(a)) is noisy. All of them are calculated using the sobel operator in method 1, and the gradient angle graph of example 3 (Fig. 19(b)) and the gradient angle graph of example 4 (Fig. 20(b)) are obtained respectively.

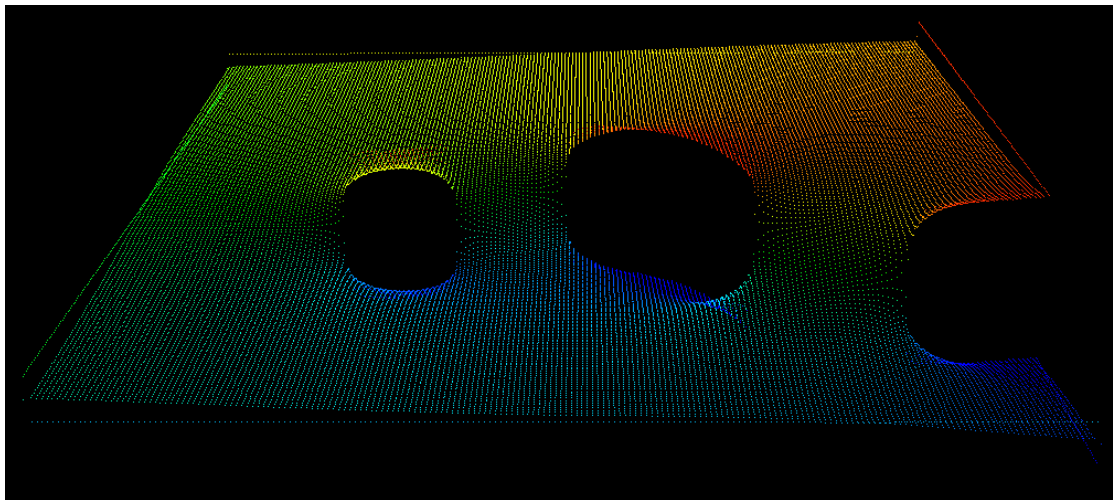
It can be seen that when there is no noise and the value of the InSAR image changes regularly, the gradient angles are all the same, and they are all 0 (only in this example). When a noise value is added to the center point, the gradient angles of the 6 points around the center point change drastically.

Compared with the subtle changes in the InSAR image itself, I think this change is magnified in the gradient angle map. So I think the point of change can be found through the gradient angle map.

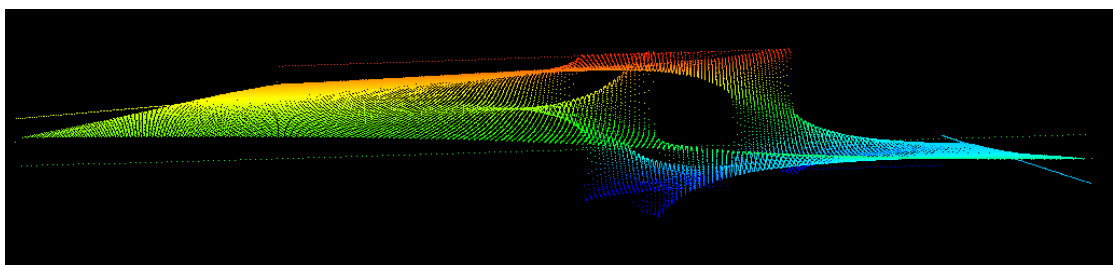
Next, I will manually add different amounts of noise (0%,5%,20%,100%) randomly to show the 3D image with gradient angles(Gradient angle range :-180°~180°):



(a)



(b)



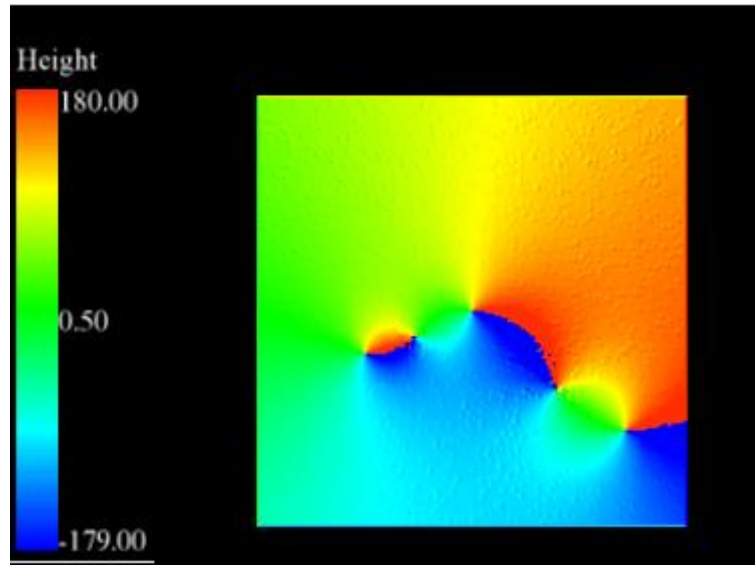
(c)

Fig. 21 Gradient angle diagram with 0% noise.(a) Top view (b)Front view(c)Side view

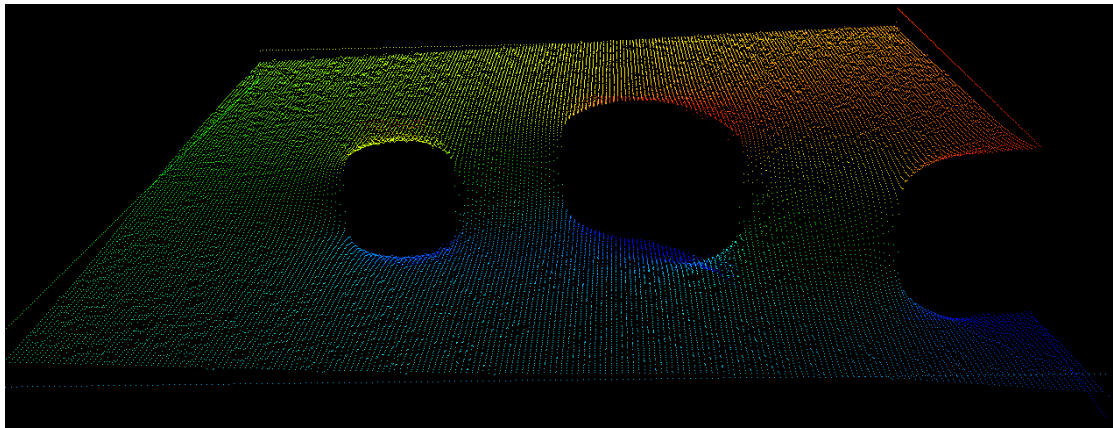
Fig. 21 shows the gradient angle diagram when the noise amount is 0%. From the above multiple perspective diagrams(Fig. 21(a)~(c)), it can be seen that when the noise amount is 0%. The gradient

angle between the points changes very smoothly. The three huge holes represent the location of the "III" in the land subsidence model, which is the origin of the numerical change, so the gradient angle changes very obviously.

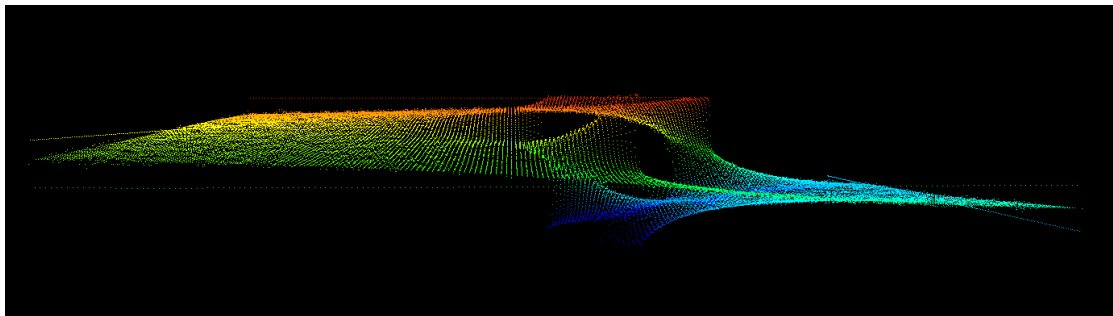
There is one thing to note. I can find that the outermost layer of the gradient angle image is obviously inconsistent with the inner layer points. The reason is that when dealing with the outermost point in Function 1, I chose to ignore it.



(a)



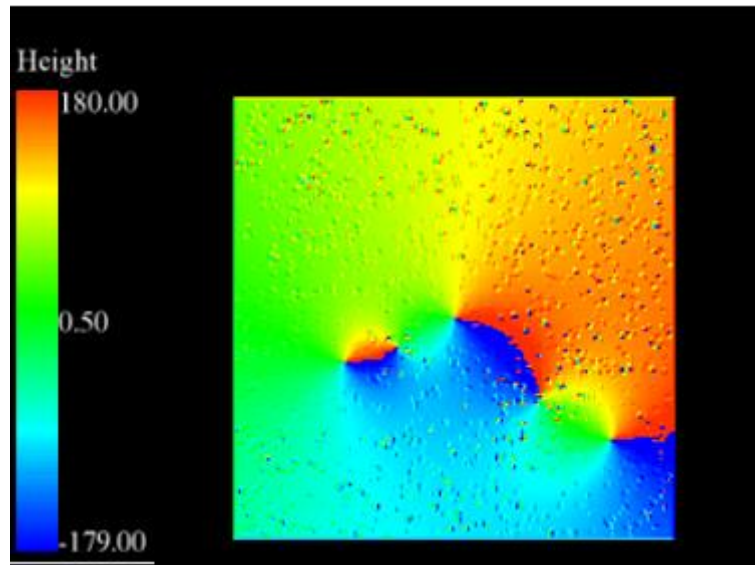
(b)



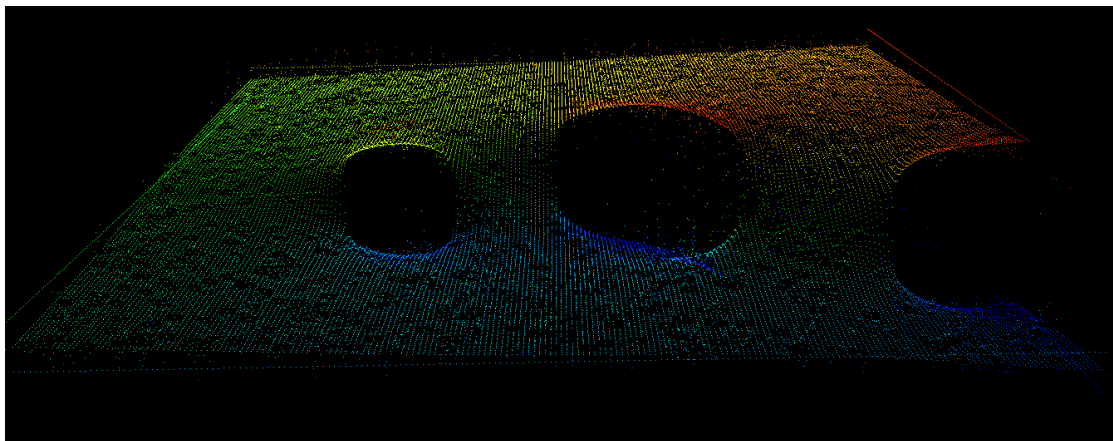
(c)

Fig. 22 Gradient angle diagram with 5% noise.(a) Top view (b)Front view(c)Side view

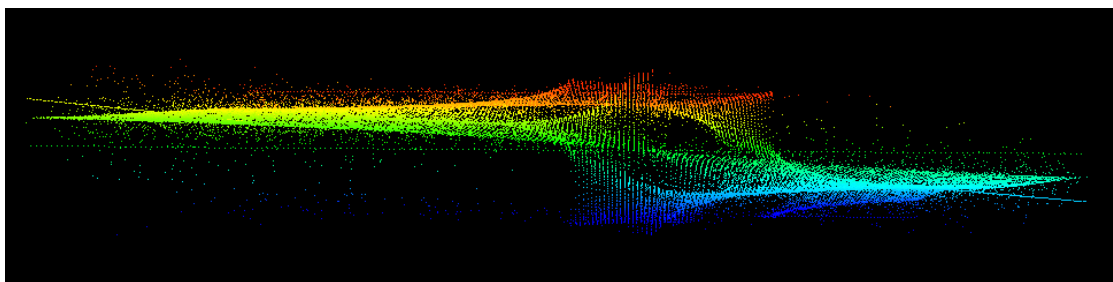
Fig. 22 shows the gradient angle diagram when the noise amount is 5%. From the above multiple perspective diagrams, it can be seen that when the noise amount is 5%. The gradient angle changes between the points are still smoother, and there are few points with changes.



(a)



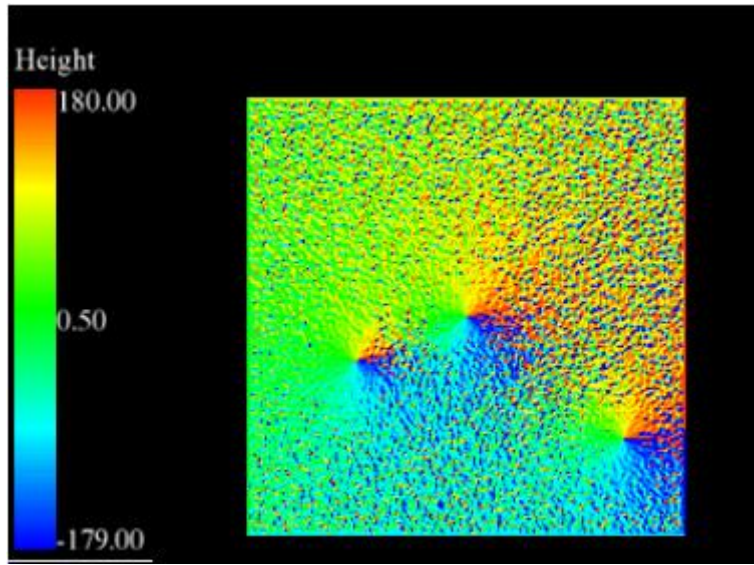
(b)



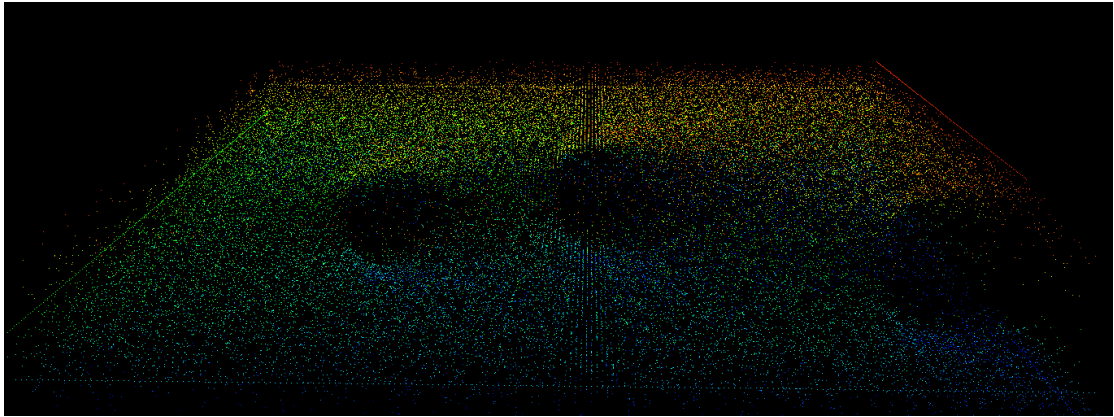
(c)

Fig. 23 Gradient angle diagram with 20% noise.(a) Top view (b)Front view(c)Side view

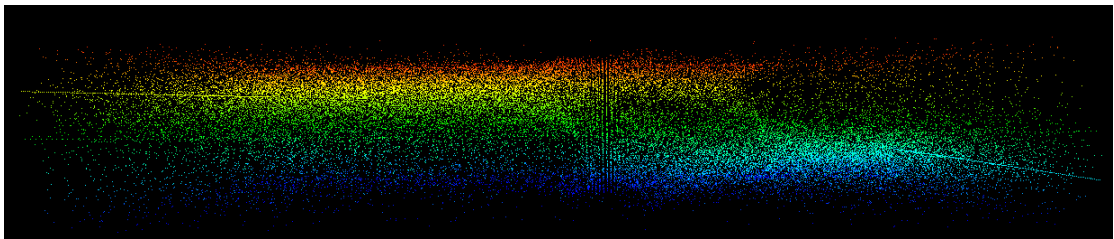
Fig. 23 shows the gradient angle diagram when the noise amount is 20%. From the above multiple perspective diagrams, it can be seen that when the noise amount is 20%. There are more points bulging or sinking, indicating that there are more points that are not smooth.



(a)



(b)



(c)

Fig. 24 Gradient angle diagram with 100% noise.(a) Top view (b)Front view(c)Side view

Fig. 24 shows the gradient angle diagram when the noise amount is 100%. From the above multiple perspective diagrams, it can be seen that when the noise amount is 100%. Almost all points are not on the same plane, and the relationship between points is very chaotic and disordered.

From Fig. 21 to Fig. 24, it can be seen that the gradient angle changes with the amount of noise in accordance with the law obtained in Example 3(Fig. 19) and Example 4(Fig. 20): In the gradient angle diagram, a noise will change the value of the surrounding 6 points, and this This change is not weak and easy to be caught.

In the next chapter, I will explain how to create conditions to find abnormal points based on this rule.

4.2.2. Create planes, compare normals, and find abnormal points in the gradient angle map

It can be concluded from the previous chapter that in the gradient angle map, a noise point will change the values of the surrounding 6 points, but the noise point itself will not change. According to this rule, I think that I can combine noise points to create multiple planes and compare the normal similarity between the planes. The steps are as follows:

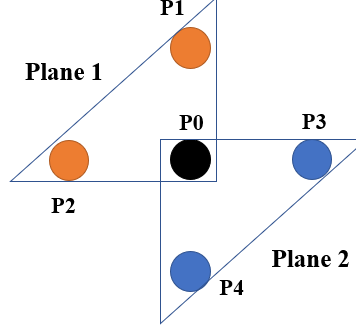


Fig. 25 Build planes

1. With one point as the center, construct two planes. P0 is the target point, and plane 1 is composed of P0, P1 and P2. Plane 2 is composed of P0, P3 and P4. Then get the normal(a,b,c) from (23)~(25).

$$a = (y_2 - y_1) * (z_3 - z_1) - (y_3 - y_1) * (z_2 - z_1) \quad (23)$$

$$b = (z_2 - z_1) * (x_3 - x_1) - (z_3 - z_1) * (x_2 - x_1) \quad (24)$$

$$c = (x_2 - x_1) * (y_3 - y_1) - (x_3 - x_1) * (y_2 - y_1) \quad (25)$$

$$\theta = \arccos(a_1 * a_2 + b_1 * b_2 + c_1 * c_2) / (|Normal_1| * |Normal_2|) \quad (26)$$

$$\theta > K_\theta \&\& \theta < 180 - K_\theta \quad (27)$$

$$K_\theta \in N, K_\theta \geq 0 \&\& K_\theta \leq 90 \quad (28)$$

Calculate the similar angle of the two normals by(26). Then compare the angle with K(27) to determine whether the point is an abnormal point. The specific value of K will be discussed in the next chapter

Then traverse the entire gradient angle graph to get all the abnormal points. The range of K is shown in (28). In order to simplify the complexity of the coefficient, only integer values are used here.

There is one more thing to note. In this step, the calculation of the neighborhood value is still involved, so I need to ignore the outermost points of the layer again. In other words, so far, I have neglected the calculation of the outer 2 layers' points.

4.2.3. Find the noise point by superimposing the value of the neighborhood of the abnormal point

In the previous chapter, I found outliers by constructing a plane and comparing the similarity of normal vectors. However, according to the conclusions drawn in example 3(Fig. 19) and example 4(Fig. 20), these abnormal points are not the noise points themselves, but points around the noise points, so I need to further process the data to find the noise points. The steps are as follows:

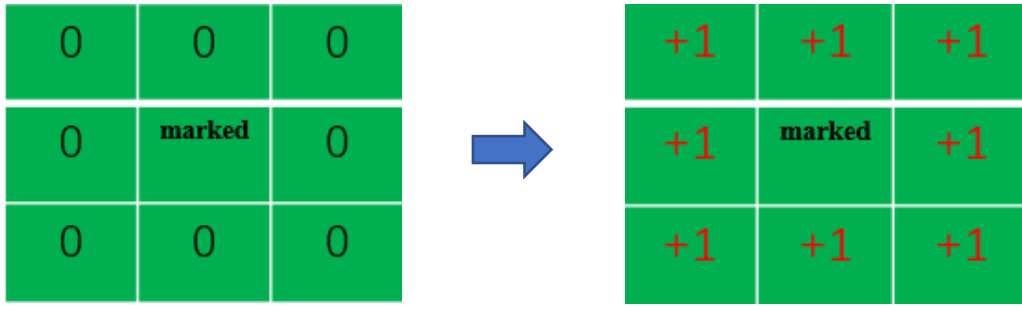


Fig. 25. Superimpose the value of the points around the target point

1. The principle of this method is shown in Figure 15. The initial value of all points is 0. For each marked point, the value of the surrounding 8 points is +1 (the marked point value is still 0). Then I will use an example 5(Fig. 26) to explain in detail:

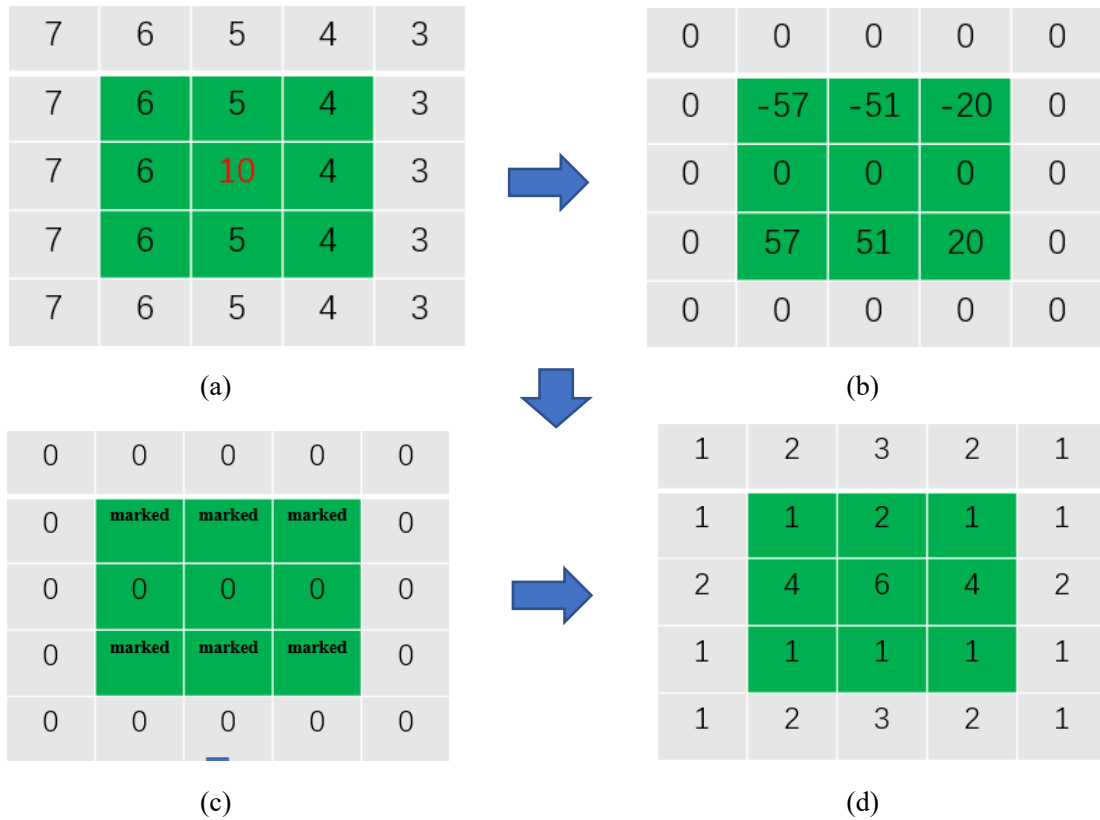


Fig. 26. Example 5. (a) InSAR image (with noise). (b) Gradient angle map. (c) Marked points map. (d) Superimposed value map

In example 5, first add a noise value at the midpoint (Fig. 26(a)), and then calculate the surrounding gradient angle (Fig. 26(b)) according to Method 1. Through the method of 4.2.1, the abnormal points can be found and marked (Fig. 26(c)), and finally the superimposed value (Fig. 26(d)) can be calculated according to the method provided in Fig. 25. It can be found that the superposition value of the center is the highest, which is the noise point I need to find.

2. After getting the graph of the superimposed value, some conditions like formula (29) can be used to filter out the normal points and get the noise points.

$$V \geq K_{\text{Superimpose}} \quad (29)$$

$$K_{\text{Superimpose}} \in N, K_{\text{Superimpose}} \geq 0 \& K_{\text{Superimpose}} \leq 8 \quad (30)$$

In the superimposed value graph, if the value of a certain point is greater than or equal to K, this point is regarded as a noise point. Here, the range of K is shown in formula (30). The specific value of K will be discussed in the next chapter.

4.3. Find the best coefficients for Method 1 and Method 2

In the above 2 chapters, I elaborate on the origin, principles and steps of the inspiration for Method 1 and Method 2. There are several coefficients involved, and these coefficients currently require manual input, so it is a problem to find the most suitable coefficients. Next I will find the best coefficients for method 1 and method 2 respectively by a loop method.

4.3.1. Find the best coefficient for Method 1

In method 1, there is a coefficient k to be determined, which is a natural number in the range [0,1]. According to the theory of method 1, the smaller the value of k, the more points can be found by method 1; the larger the value of k, the fewer points can be found by method 1. On this basis I set the range of k to [0.0001,1] and I will loop 10000 times with k taking the value from 0.0001 and increasing 0.0001 in each loop until the value is 1.

In addition to this, the amount of noise points to be placed needs to be discussed. Here I first put 5%, 50% and 100% of the amount of noise, and observe the law of k through these groups of noise, and the flow figure is shown (Fig. 27):

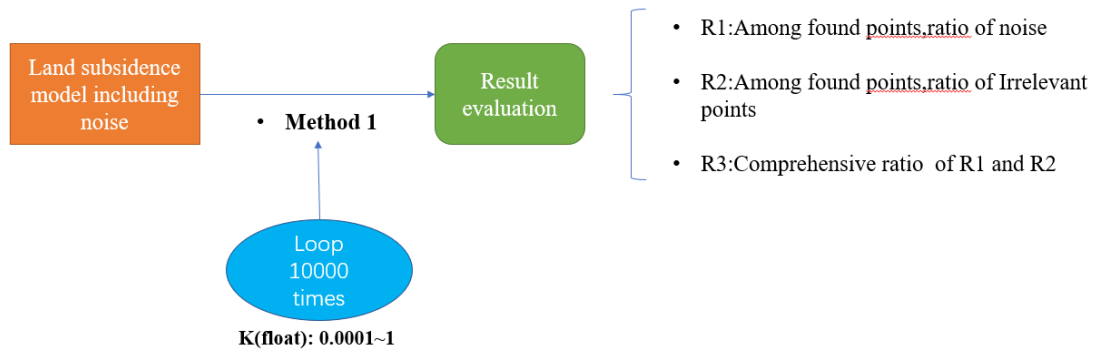


Fig. 27 Processing flow of method 1(5%,50%,100%amount of noise)

From the flow in the above figure, you can see that for the evaluation of the results, I used the values R1,R2 and R3 to evaluate the results. I will explain the meaning of these values and their specific formulas in the "Results" section.

After getting the best k-value results at 5%, 50% and 100% amount of noise, I will analyze from putting 1% to 100% amount of noise, and the process is shown in the figure below(Fig. 28):

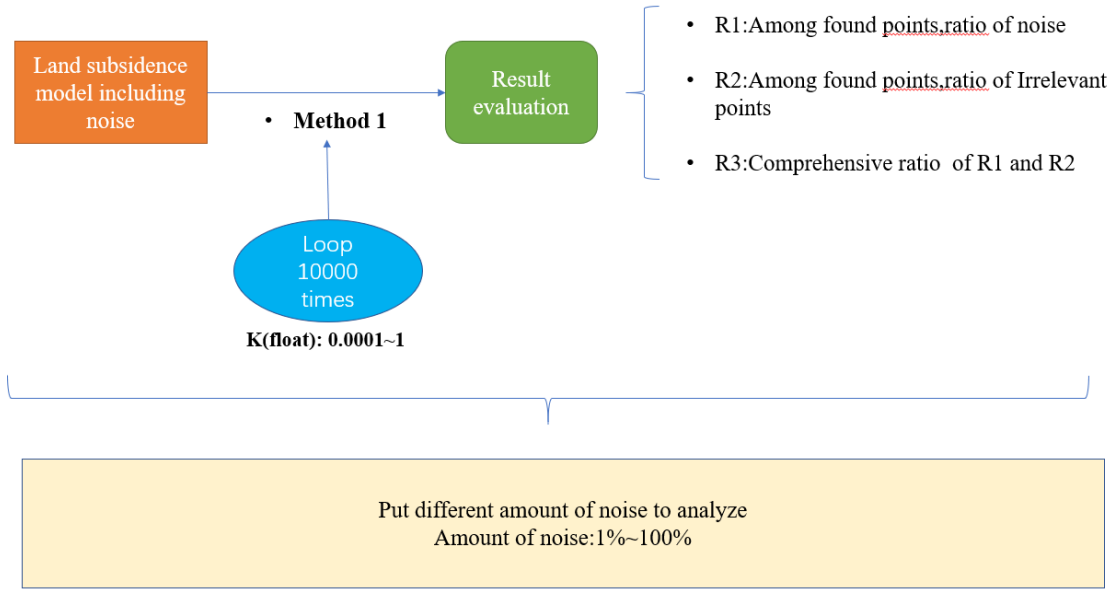


Fig. 28 Processing flow of method 1(1%~100% amount of noise)

It should be noted that my final assessments are based on the value of R3.

4.3.2. Find the best coefficients for Method 2

Now the optimal k value for method 1 in the case of 1% to 100% noise count is obtained, and based on this, I will explore how to find the optimal k value for method 2. Method 2 has two k values, and the principles are referred to sections 4.2.2 and 4.2.3.

K_θ denotes the angle (Section 4.2.2), which is used to evaluate the similarity of the normal vectors, that is $COS\theta$. K_θ ranges from $[0^\circ, 90^\circ]$, where K_θ is taken as an integer to simplify the computation and the complexity of the coefficients.

The $K_{\text{Superimpose}}$ denotes the threshold of the superposition value (Section 4.2.3), and the range of $K_{\text{Superimpose}}$ can be known from the principle as $[0, 8]$, and $K_{\text{Superimpose}}$ is taken as an integer.

The above analysis leads to a flow shown in Fig. 29.

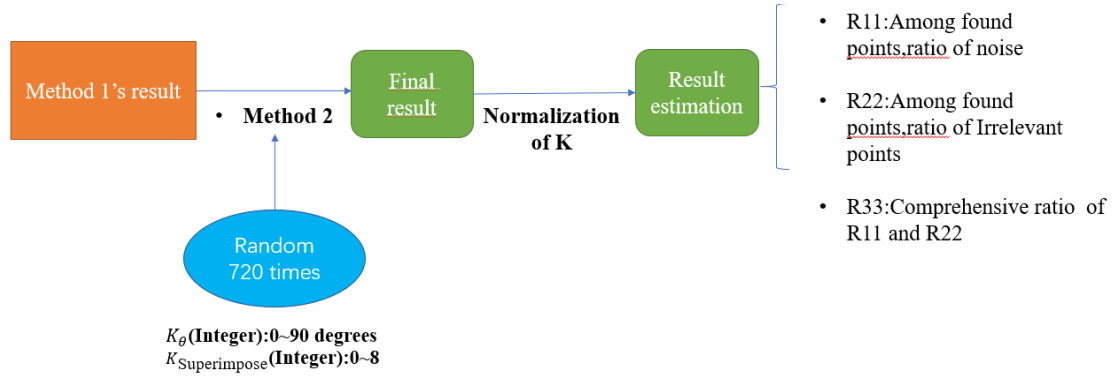


Fig. 29 Processing flow of method 2(5%,50%,100%amount of noise)

From the flow in the above figure, you can see that for the evaluation of the results, I used the values R11, R22 and R33 to evaluate the results. I will explain the meaning of these values and their specific formulas in the "Results" section.

In contrast to finding the best coefficients for method 1, there is a normalization step before evaluating the best coefficients for method 2. The reason for this is that method 2 has two coefficients, which do not have the same range, and to facilitate the evaluation, I use normalization.

After getting the best k-value results at 5%, 50% and 100% amount of noise, I will analyze from putting 1% to 100% amount of noise, and the process is shown in the figure below(Fig. 30):

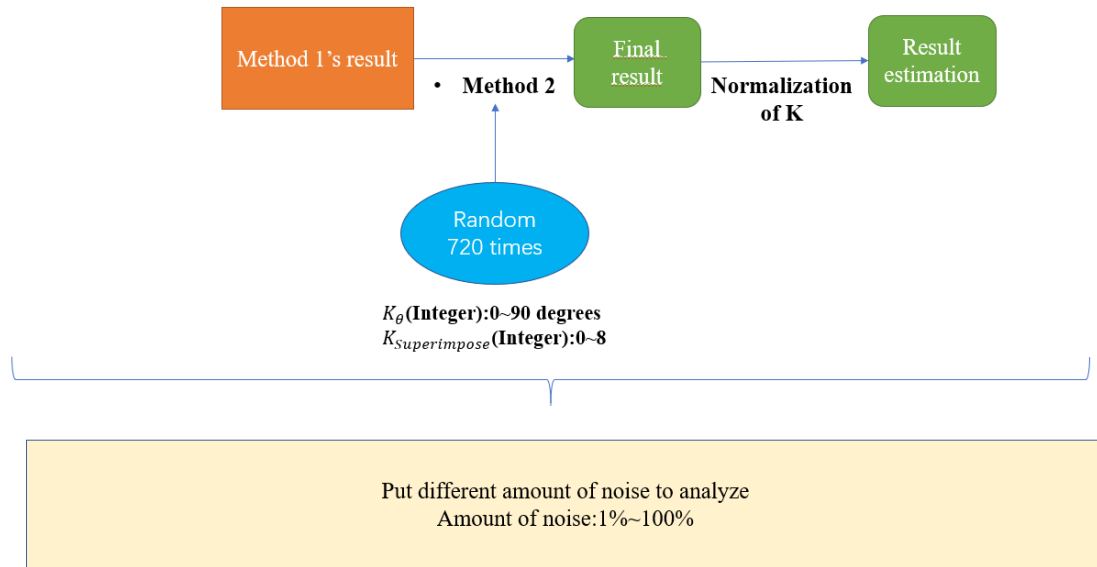


Fig. 30 Processing flow of method 2(1%~100% amount of noise)

Through the above two chapters, I found the optimal coefficients for method 1 and method 2, respectively, and obtained the table of optimal coefficients for noise quantities from 1% to 100%. The next chapter will show if the amount of noise is deduced from this list of data.

4.3.3. Apply the best coefficients to Method 1 and Method 2, and compare the results

Now I get a list of the best coefficients for method 1 and method 2 with the amount of noise from 1% to 100%. Different noise quantities have different coefficients. Now I will use this data sheet to apply the best coefficients according to different noise quantities to get the best results and compare them. The purpose of comparison is twofold:

1. Observe the law of the results obtained by applying the two methods from 1% to 100% of the noise amount.
2. Observe the number of noises from 1% to 100%, and only apply method 1 and apply method 1 and method 2 at the same time. In other words, in the case of different amounts of noise, method 2 improves the results. Here is the flow chart(Fig. 31):

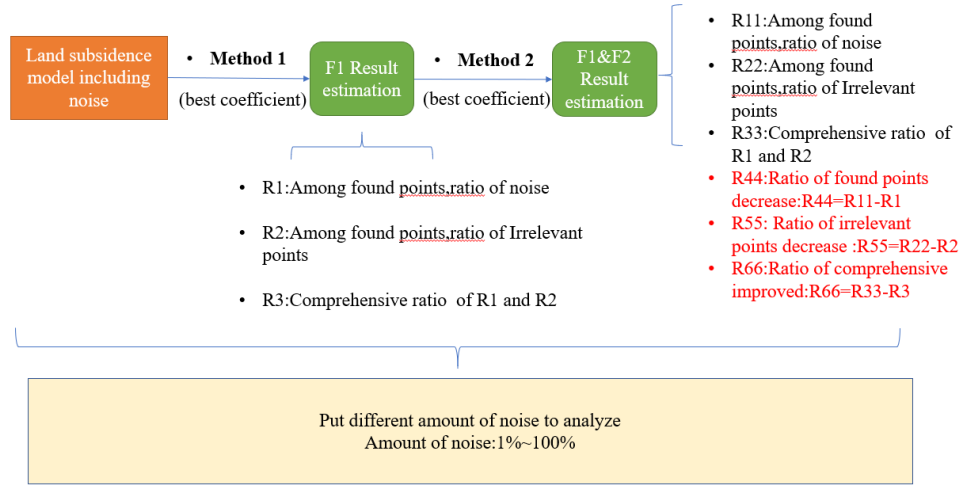


Fig. 31 Processing flow of method 1&&method 2(1%~100% amount of noise)

In the process in the above figure, I used several new indicators R44, R55 and R66 to compare the results of Method 1 and the results of the combination of Method 1 and Method 2.

4.4. Infer the real amount of noise

Using the above section, I found the optimal coefficients and the evaluated values R1, R2, and R3 for different amounts of noise (Chapter 4.3). I use the results to organize into a list of data, part of which is shown below (Fig. 32):

1	0.0006	8	4	0.963731	0.0147074
2	0.0006	11	5	0.94717	0.0128332
3	0.0006	11	3	0.954466	0.025328
4	0.0006	10	5	0.958831	0.0204342
5	0.0006	11	5	0.95543	0.0294669
6	0.0006	11	4	0.958073	0.0359225
7	0.0006	11	5	0.95685	0.040504
8	0.0006	12	5	0.949518	0.043914
9	0.0006	12	4	0.960613	0.0496668
10	0.0005	11	5	0.959959	0.0638015
11	0.0005	12	5	0.954961	0.0653374
12	0.0006	34	2	0.948579	0.064166
13	0.0006	11	4	0.95406	0.0758278
14	0.0005	17	3	0.961812	0.0889473
15	0.0006	23	3	0.950723	0.0871772
16	0.0006	22	3	0.951376	0.0903269
17	0.0006	17	4	0.952068	0.0994638
18	0.0006	23	3	0.948229	0.105008
19	0.0006	33	1	0.948515	0.111464
20	0.0006	19	4	0.952625	0.115499
21	0.0005	37	1	0.956095	0.132341
22	0.0005	24	2	0.957008	0.137417
23	0.0006	31	2	0.946354	0.132653
24	0.0006	29	2	0.951163	0.137036

Amount of noise(%)

K1

K_θ

$K_{superimpose}$

R1(Ratio of noise)

R2(Ratio of irrelevant points)

Fig. 32 Data list of noise quantities, optimal coefficients and evaluation values

So how do you derive the number of noises from this list of data? The derivation process is shown in(Fig. 33):

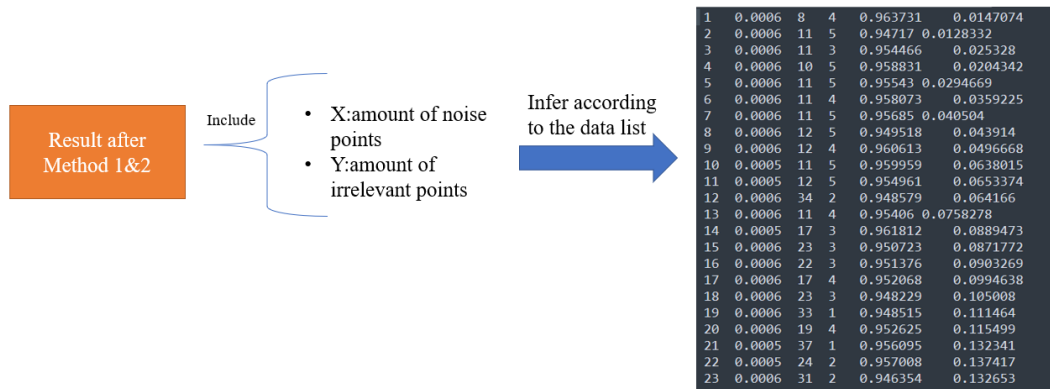


Fig. 33 Process of deriving noise quantities

In the flowchart, after method 1 and method 2, I will be able to find a certain amount of noise, but it contains an unknown number of points that really belong to the noise and an unknown number of irrelevant points, so the next step is to deduce a more accurate amount of noise by way of derivation.

In this derivation, the most important thing is the list of data above. The method of derivation can be summarized as "hypothesis" and "loop", and the data list is used to verify the best hypothesis.

You may find it difficult to understand how to derive the number of noises with just a flow chart and these words. So an example to illustrate is provided in Fig. 34.

Explanation of the example

A1(Amount of total points):40000
A2(Amount of noise):Unknown
P1(Found points):**Known**
 $P1 = X + Y$
X(Among found points, the amount of noise):Unknow
Y(Among found points, the amount of Irrelevant points):Unknown
According to data list:
Assume amount of noise is 1%:
 $\Rightarrow A2 = A1 * 1\% = 400$
 $\Rightarrow Y = 0.0147074 * A1 = 590$
 $\Rightarrow X = A2 * 0.963731 = 386$
 $\Rightarrow Total1 = X + Y = 946$
 \Rightarrow log Total1 , and compare Total1 & P1
 \Rightarrow Cycle from 1% to 100%, and find the nearest one

Fig. 34 Example of deriving noise quantities

I will explain this example in words. In this example, the total number of points is A1, which is 40,000 points. A2 is the number of noise, and A2 is the unknown value. p1 is the number of points found, which is the number of points obtained after processing by method 1 and method 2. This quantity contains the unknown number of noisy points and the unknown number of irrelevant points, that is, X and Y.

The derivation starts next. First, I assume that there is a total of 1% of the number of noise, which can be calculated as A2, that is, the number of noise is $40,000 * 1\% = 400$. At this time, according to the existing data list query, I know that when the number of noise is 1%, $Y/A1 = 0.0147074$, that is, the proportion of irrelevant points to the total number of points is 0.0147074, I can calculate $Y = A1 * R2 = 40,000 * 0.0147074 = 590$ points. It can also be learned that when the number of noise is 1%, $X/A2 = 0.963731$, that is, the proportion of noise points to the points already found is 0.963731, and $X = A2 * R1 = 400 * 0.963731 = 386$ points can be calculated.

Total1=X+Y=976 points can be calculated, because P1 is known. So theoretically, if the number of noise is 1%, the value of Total1 and the value of P1 should be equal or approximate. So I record the value of Total1 and compare the value of Total1 with P1.

The first derivation is over. Next, assume that the noise quantity is 2% and repeat the above process until 100%. Then compare the value of Total obtained each time and find the value closest to P1 among them. Then this case is the real amount of noise.

4.5.Smooth the noise and verify

In terms of noise smoothing, I have no essential innovations. I only made some improvements to the two traditional smoothing methods. The following will introduce these two traditional smoothing methods and my improvement.

4.5.1. Traditional Gaussian filtering

Gaussian smoothing, also called Gaussian blur, is usually used to reduce image noise and reduce the level of detail. The visual effect of the image generated by this blur technology is like observing the image through a translucent screen, which is obviously different from the out-of-focus imaging effect of the lens and the effect in the shadow of ordinary lighting. Gaussian smoothing is also used in the pre-processing stage of computer vision algorithms to enhance the image effect of images at different scales (see scale space representation and scale space implementation). From a mathematical point of view, the Gaussian blur process of an image is the convolution of the image and the normal distribution. Since the normal distribution is also called "Gaussian distribution", this technique is called Gaussian blur. Convolution of the blurred image with the circular box will produce a more accurate out-of-focus imaging effect. Since the Fourier transform of the Gaussian function is another Gaussian function, the Gaussian blur is a low-pass filter for the image.

The two-dimensional Gaussian distribution function is as formula (31):

$$f(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right) \quad (31)$$

where x and y represent spatial coordinates, and σ is a coefficient that needs to be manually input. In theory, the Gaussian distribution has non-negative values in all domains, which requires an infinite convolution kernel. In fact, you only need to take the value within 3 times the standard deviation around the mean, and just remove the other parts.

The two important steps of Gaussian filtering are to find the Gaussian template first and then perform convolution. Let me give an example to illustrate. Assuming that the coordinates of the center point are (0,0), then take the coordinates of the 8 points closest to it. For calculation, the value of σ needs to be set. Assuming $\sigma=1.5$, the Gaussian template with blur radius of 1 is as follows(Fig. 35):

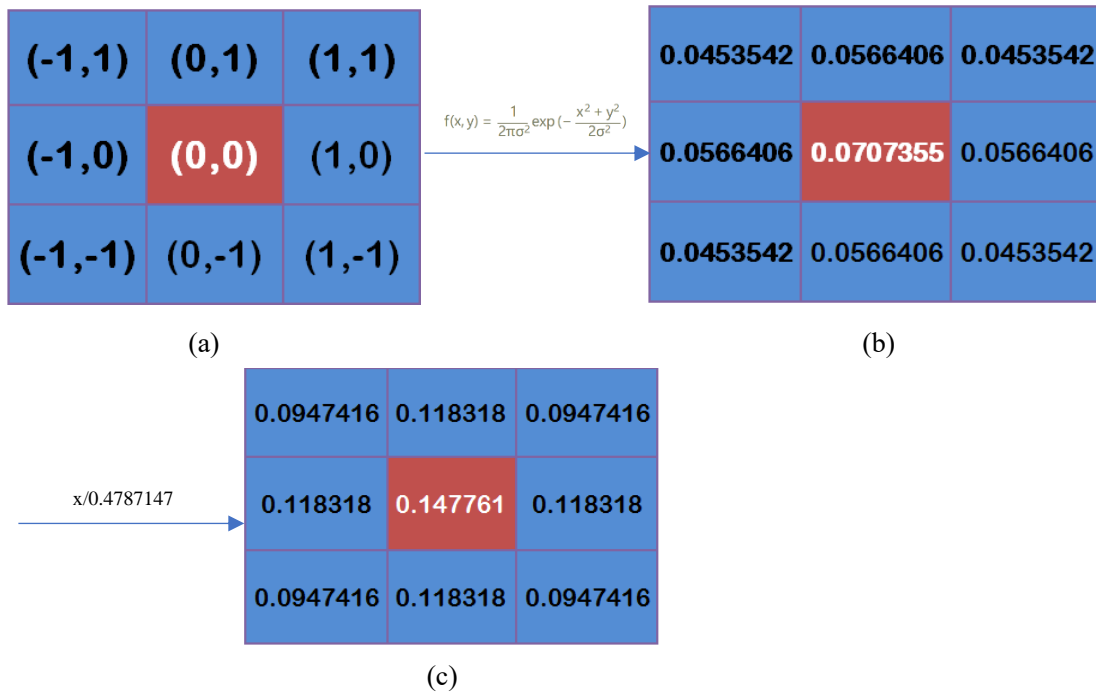
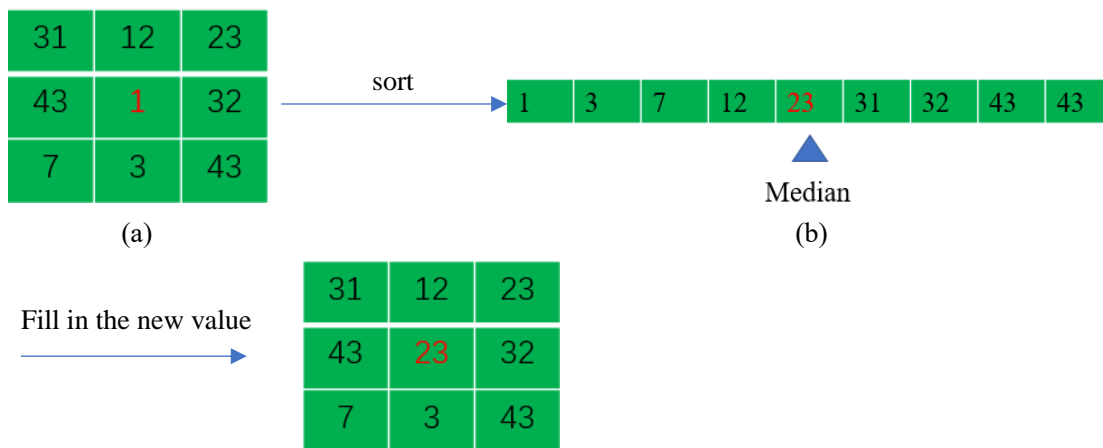


Fig. 35 Gaussian template.(a) Coordinate graph.(b) Coefficient graph after Gaussian function processing(c) Coefficient graph after weight processing

First, use Gaussian function to process the 9 coordinates(Fig. 35(a)) to get the coefficient map(Fig. 35(b)). When adding up all the coefficients, it is found that it is not equal to 1, but equal to 0.4787147. In order to make the weight add up to 1, so for all coefficients, I have to divide by 0.4787147, and then Get the final Gaussian template(Fig. 35(c)). Use this template to convolve the InSAR image (the outermost layer needs to be ignored), and the filtered result will be obtained.

4.5.2. Traditional Median filtering

The median filter is a non-linear digital filtering technique, often used to remove noise from an image or signal. Such noise reduction is a typical pre-processing step to improve the results of later processing (for example, edge detection on an image). Median filtering is very widely used in digital image processing because, under certain conditions, it preserves edges while removing noise, also having applications in signal processing. I will give an example to illustrate the median filter function(Fig. 36):



(c)

Fig. 36 Example of Median Filtering.(a) Original graph.(b) Sort the values(c) Graph after filling in new values

In the example, (Fig. 36(a)) is the original data graph. After sorting, I get (Fig. 36(b)), where the middle value is 23, and 23 is used as the new value instead of the original middle value to get (Fig. 36(c)).

4.5.3. My Gaussian filtering and Median filtering

In the subsequent processing, I improved the two traditional smoothing methods and used them. The difference between my method and the traditional method is shown in the table below(Fig. 37):

Method	Difference
Traditional Gaussian filter	Smooth all points
Traditional Median filter	Smooth all points
My Gaussian filter	Smooth found points
My Median filter	Smooth found points

Fig. 37 Method comparison table

So my processing method does not essentially improve the traditional method, but is closely related to method 1 and method 2.

4.5.4 Verification

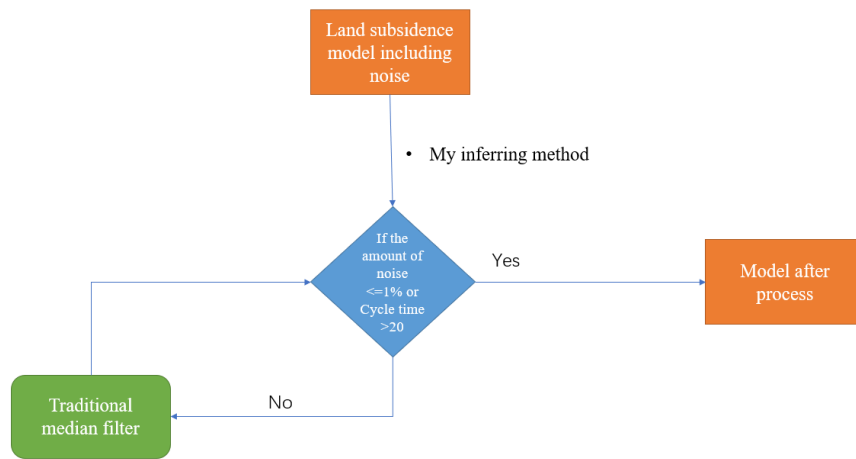
In order to quantify the result after smoothing, I need to use a certain index to judge and quantify the result. Here I choose RMS.

In mathematics and its applications, the root mean square (RMS or rms or rms) is defined as the square root of the mean square (the arithmetic mean of the squares of a set of numbers).The RMS is also known as the quadratic mean and is a particular case of the generalized mean with exponent 2. RMS can also be defined for a continuously varying function in terms of an integral of the squares of the instantaneous values during a cycle like formula (32):

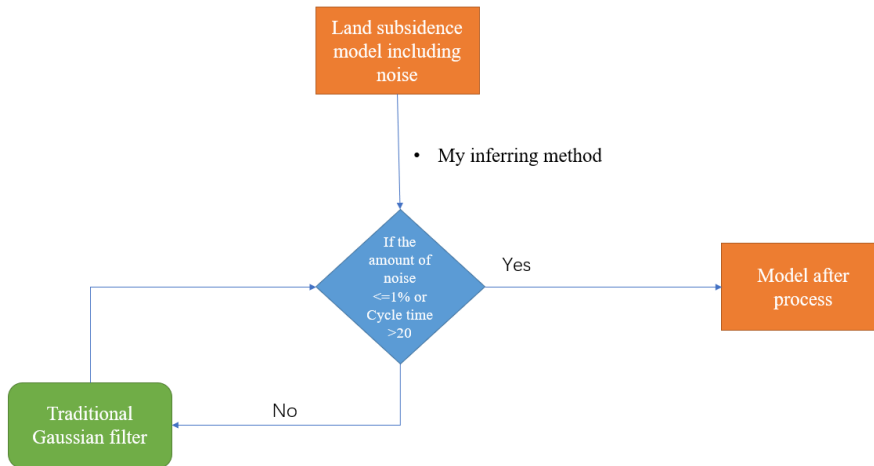
$$RMS = \sqrt{\frac{\sum (\phi_o(i,j) - \phi_p(i,j))^2}{N - 1}} \quad (32)$$

4.5.5 Make smoothing strategies

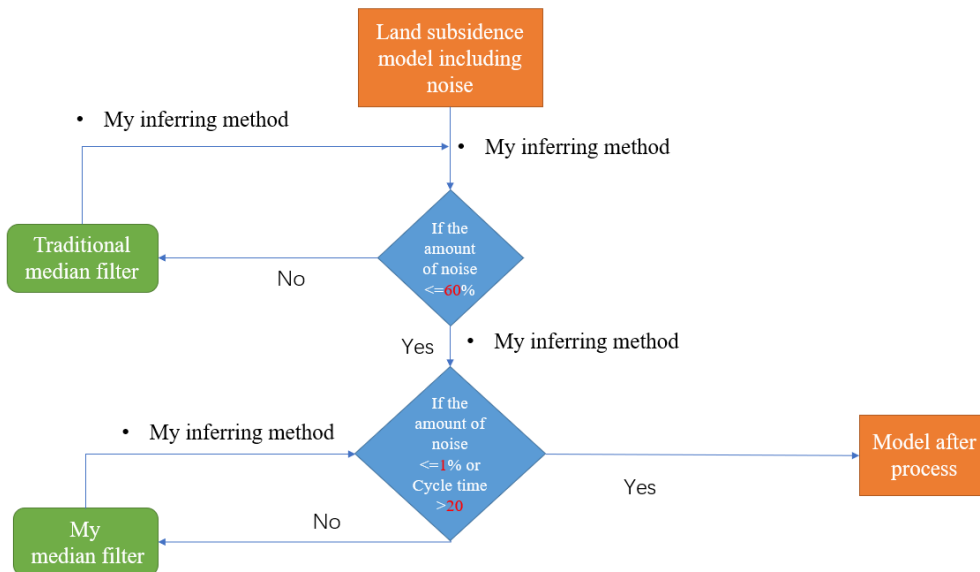
In the following "Results" chapter, you can find that when the amount of noise is limited, the method of combining method 1 and method 2 with smoothing noise can smooth the noise easily and achieve better results. However, when the amount of noise is very large, only smoothing once may not achieve the desired result, and smoothing for multiple times is required. If I want to achieve multiple smoothing, the indispensable method is to infer the amount of noise. I have described this method in detail in Chapter 4.4, so let me show the processing strategy flow for 4 different smoothing methods(Fig. 38):



(a)



(b)



(c)

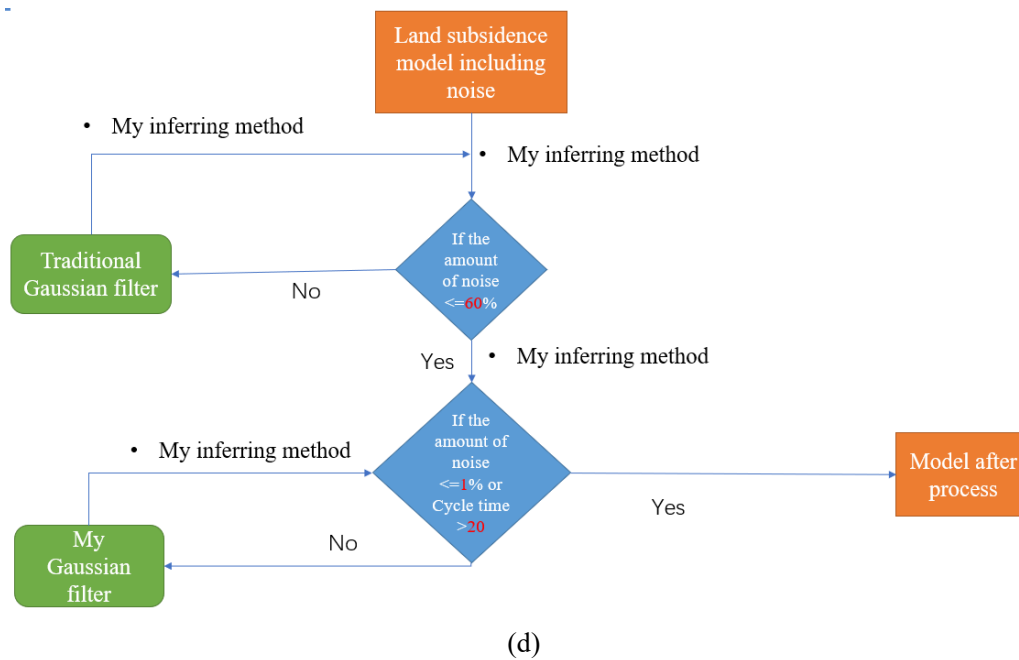


Fig. 38 Model processing flowchart.(a) Flow chart of traditional median filter processing.(b) Flow chart of traditional Gaussian filter processing.(c) Flow chart of my median filter processing.(d) Flow chart of my Gaussian filter processing.

As can be seen from the above figure, the maximum number of cycles for the four methods is 20. This is to save computing resources, and I think 20 cycles are enough to reduce the amount of noise to a certain range.

For the traditional smoothing method (Fig. 38(a)(b)), when the number of cycles is less than 20 and the amount of noise is more than 1% through the inference method, the traditional smoothing method is used for smoothing, otherwise exit the loop.

For my smoothing method (Fig. 38(c)(d)), the process is slightly different from the traditional method. When the amount of noise exceeds 60%, use the corresponding traditional method for smoothing. When the number of noises is less than 60%, judge whether the number of cycles is less than 20 and the number of noises is more than 1% through the inference method. If the result is "True", use my smoothing method for smoothing, otherwise exit cycle.

Among them, it can be found that 60% is a magic number. The reason why I use 60% as the basis for judgment can be seen in chapter 5.6.

1% is another critical point. I think if the noise number is less than or equal to 1%, it can be ignored and Get the final result.

4.6. Apply methods and strategies to InSAR in the study area

After verifying my methods and strategies using numerical models, I applied these methods and strategies to the actual research area.

Since the InSAR data I got in practice covers the entire Chiba Prefecture, its scope is too wide, the amount of data is very large, and it takes a lot of time to process. So I intercepted the part of the Kujukuri plain as follows(Fig. 39):

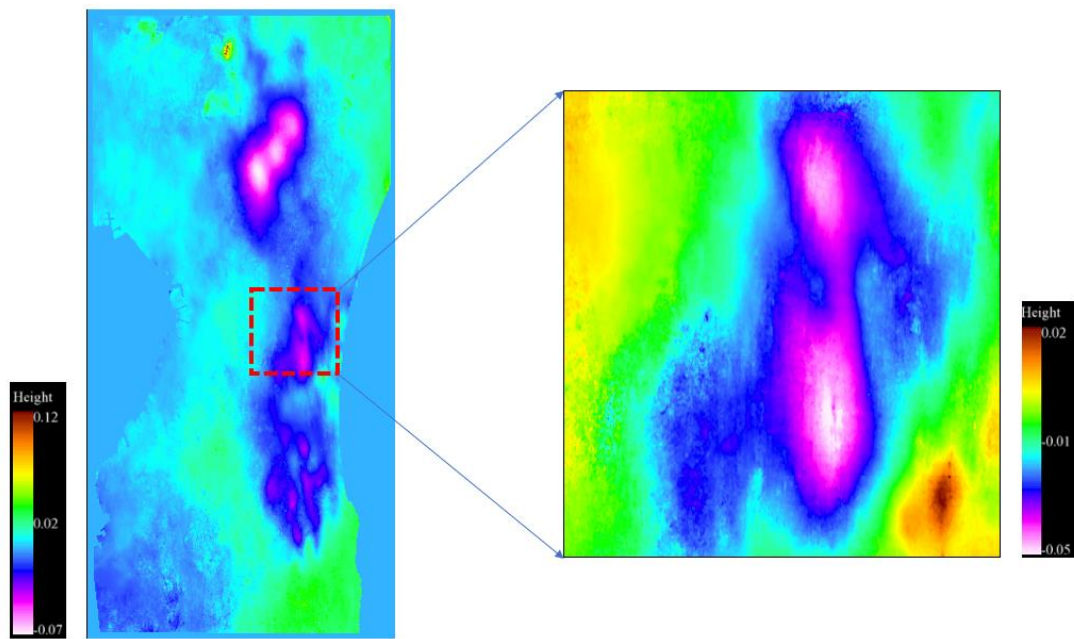


Fig. 39 InSAR image of Kujukuri area

For the smoothing strategy to be used, the same as in the previous chapter, I chose to loop 20 times and use my Gaussian smoothing method.

5. Results and discussion

5.1.Method 1's results

In Method 1, several evaluation indicators are used to measure the results, see section 4.3.1. In this chapter, I still need to use these indicators, so I will give examples to explain in detail the meaning of these indicators. They are R1, R2 and R3 respectively. R1 represents the ratio of the number of noise points among the points found. R3 represents the ratio of the number of irrelevant points among the points found. R3 represents the comprehensive ratio, $R3=R1*(1-R2)$.

One thing to note is that the denominators of R1 and R2 are different. The denominator of R1 is the number of points found, and the denominator of R2 is the number of points in the entire InSAR. An example is shown below(Fig. 40):

Explanation of the example	
A1(Amount of total points):	40000
A2(Amount of noise I drop):	6000
P1(Found points):	5000
P2(Among found points,amount of noise):	3000
P3(Among found points,amount of Irrelevant points):	$P1-P2=2000$
R1(Among found points, ratio of noise):	$P2/A2=0.5$
R2(Among found points, ratio of Irrelevant points):	$P3/A1=0.05$
R3(Comprehensive ratio):	$R1*(1-R2)=0.475$

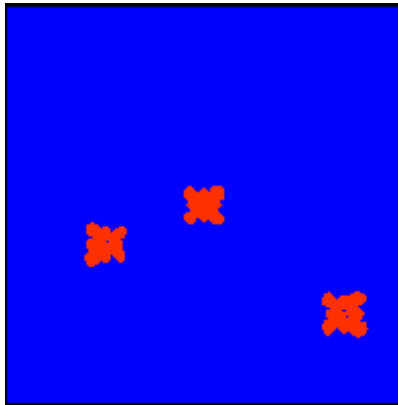
Fig. 40 Example of R1,R2 and R3

In Method 1, there is a undetermined coefficient k1, which needs to be entered manually. Before finding the best coefficient, I set $k1=0.001$. For the amount of noise, I used 5% of the noise and 100% of the noise for the experiment.

From section 4.1.2, I can find that I created two conditions to find noise. They are very similar, but they work in different directions. Now I define formulas (7)~(9) as condition 1, formulas (10)~(22) as condition 2. Condition 1 acts in the direction of gradient descent. Equation 2 acts on the orthogonal direction of the gradient descent.

5.1.1. Use condition 1 to process model

Next, I will show the results of using condition 1 on the original InSAR image, 5% noise image and 100% noise image(Fig. 41):



A1(Amount of total points):	40000
A2(Amount of noise I drop):	0
P1(Found points):	1045
P2(Among found points,amount of noise):	0
P3(Among found points,amount of Irrelevant points):	$P1-P2=1045$
R2(Among found points, ratio of Irrelevant points):	$P3/A1=0.026125$

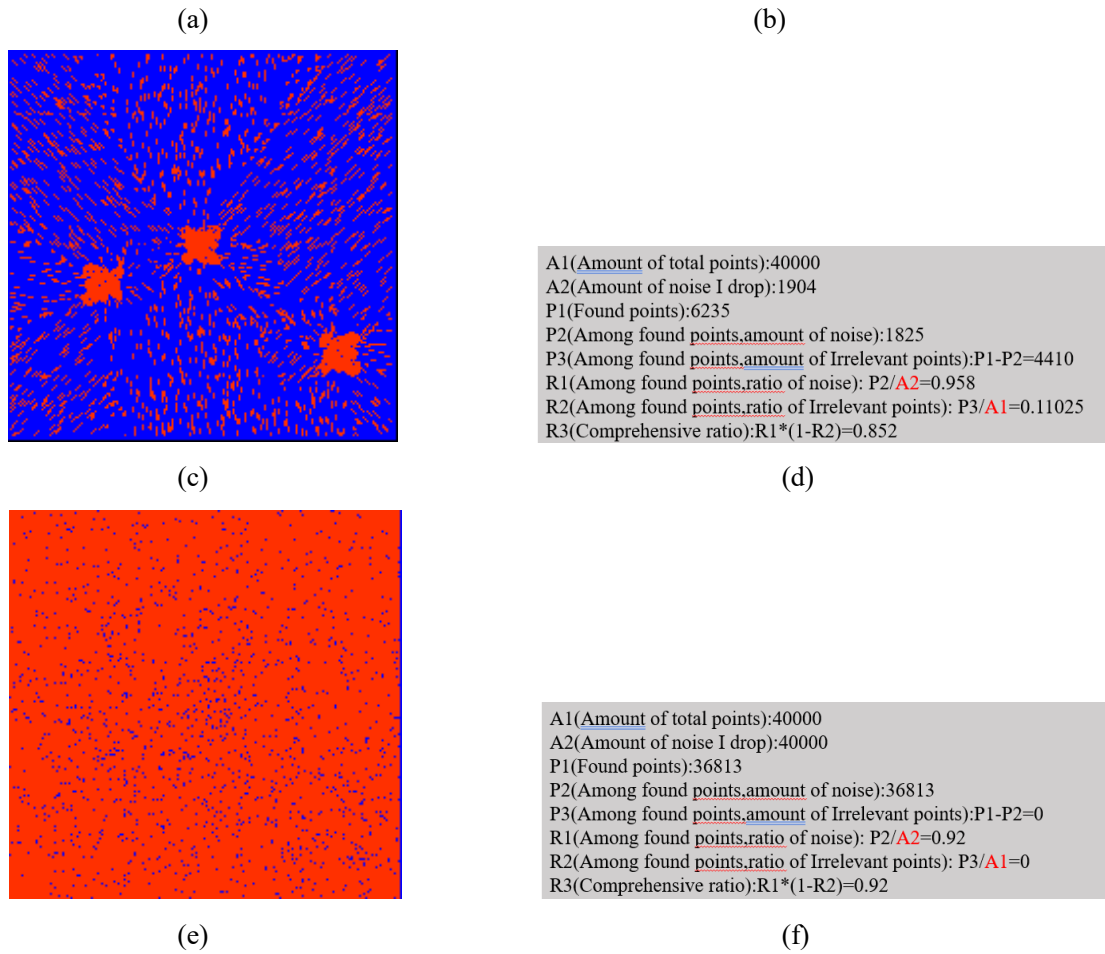
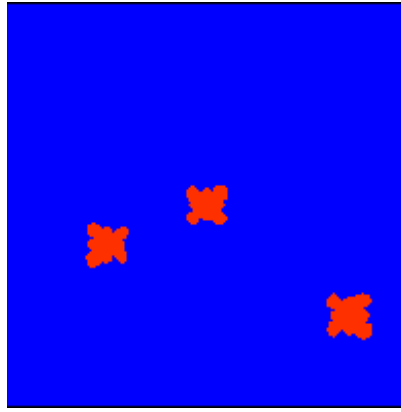


Fig. 41 Found points by condition 1.(a)Found points of original model.(b) indicators data of original model. (c)Found points of 5% noise model.(d) indicators data of 5% noise model. (e)Found points of 100% noise model.(f) indicators data of 100% noise model.

In the image above, the red dots represent the found points, and the blue dots represent other points. It can be seen that method 1 will still find about 1000 irrelevant points(Fig. 41(a)(b)) when processing the original model. When the amount of noise is 5%, most noise points can be found(Fig. 41(c)(d)), but many irrelevant points will also be found. When the amount of noise is 100%, most of the noise points can also be found(Fig. 41(e)(f)).

5.1.2. Use condition 2 to process model

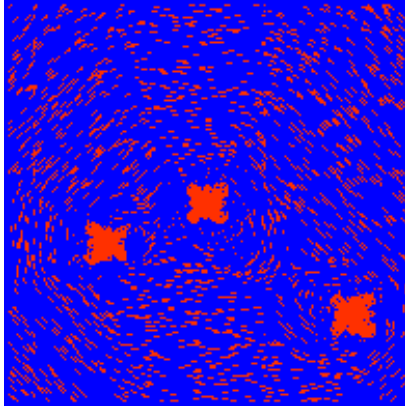
Next, I will show the results of using condition 2 on the original InSAR image, 5% noise image and 100% noise image(Fig. 42):



(a)

```
A1(Amount of total points):40000
A2(Amount of noise I drop):0
P1(Found points):1033
P2(Among found points,amount of noise):0
P3(Among found points,amount of Irrelevant points):P1-P2=1033
R2(Among found points, ratio of Irrelevant points): P3/A1=0.025825
```

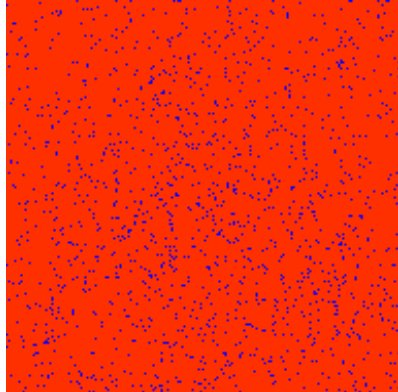
(b)



(c)

```
A1(Amount of total points):40000
A2(Amount of noise I drop):1860
P1(Found points):5824
P2(Among found points,amount of noise):1854
P3(Among found points,amount of Irrelevant points):P1-P2=3970
R1(Among found points, ratio of noise): P2/A2=0.9967
R2(Among found points, ratio of Irrelevant points): P3/A1=0.09925
R3(Comprehensive ratio):R1*(1-R2)=0.897
```

(d)



(e)

```
A1(Amount of total points):40000
A2(Amount of noise I drop):40000
P1(Found points):36814
P2(Among found points,amount of noise):36814
P3(Among found points,amount of Irrelevant points):P1-P2=0
R1(Among found points, ratio of noise): P2/A2=0.920
R2(Among found points, ratio of Irrelevant points): P3/A1=0
R3(Comprehensive ratio):R1*(1-R2)=0.92
```

(f)

Fig. 42 Found points by condition 2.(a)Found points of original model.(b) indicators data of original model. (c)Found points of 5% noise model.(d) indicators data of 5% noise model. (e)Found points of 100% noise model.(f) indicators data of 100% noise model.

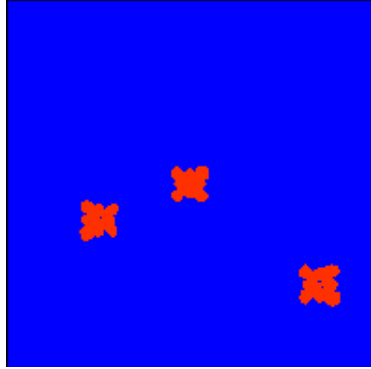
It can be seen that method 1 will still find about 1000 irrelevant points(Fig. 42(a)(b)) when processing the original model. When the amount of noise is 5%, most noise points can be found(Fig. 42(c)(d)), but many irrelevant points will also be found. When the amount of noise is 100%, most of the noise points can also be found(Fig. 42(e)(f)).

Comparing the results of Condition 1 and Condition 2, I found that the results they got are very similar.

Comparing Fig. 51(c) and Fig. 42(c), it can be found that the point found using condition 1 is very similar to the point in the direction of gradient descent, while the point found using condition 2 is very similar to the boundary point of the point.

5.1.3. Use condition 1 and condition 2 to process model

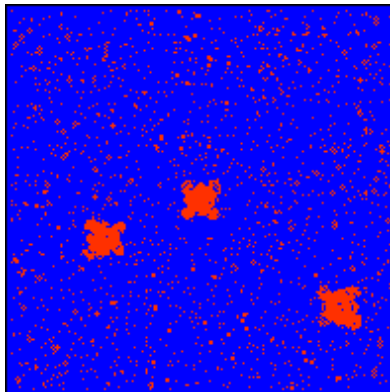
Next, I will show the results of using condition1 and condition 2 on the original InSAR image, 5% noise image and 100% noise image(Fig. 43):



(a)

A1(Amount of total points):40000
A2(Amount of noise I drop):0
P1(Found points):1028
P2(Among found points,amount of noise):0
P3(Among found points,amount of Irrelevant points):P1-P2=1028
R2(Among found points, ratio of Irrelevant points): P3/A1=0.0257

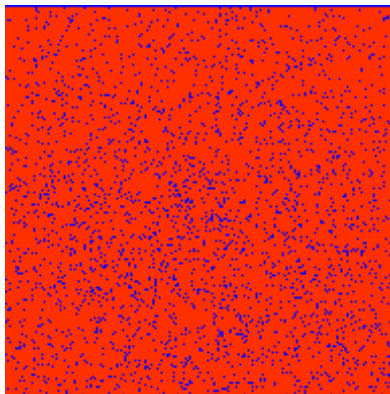
(b)



(c)

A1(Amount of total points):40000
A2(Amount of noise I drop):1953
P1(Found points):3088
P2(Among found points,amount of noise):1898
P3(Among found points,amount of Irrelevant points):P1-P2=1190
R1(Among found points, ratio of noise): P2/A2=0.9718
R2(Among found points, ratio of Irrelevant points): P3/A1=0.02975
R3(Comprehensive ratio):R1*(1-R2)=0.9429

(d)



(e)

A1(Amount of total points):40000
A2(Amount of noise I drop):40000
P1(Found points):35306
P2(Among found points,amount of noise):35306
P3(Among found points,amount of Irrelevant points):P1-P2=0
R1(Among found points, ratio of noise): P2/A2=0.882
R2(Among found points, ratio of Irrelevant points): P3/A1=0
R3(Comprehensive ratio):R1*(1-R2)=0.882

(f)

Fig. 43 Found points by condition 1 and condition 2.(a)Found points of original model.(b) indicators data of original model. (c)Found points of 5% noise model.(d) indicators data of 5% noise model.

(e)Found points of 100% noise model.(f) indicators data of 100% noise model.

Compared with condition 1 or condition 2 alone, for the original model, I found that the results of combining the two conditions are similar. But for the 5% noise quantity model, fewer irrelevant points

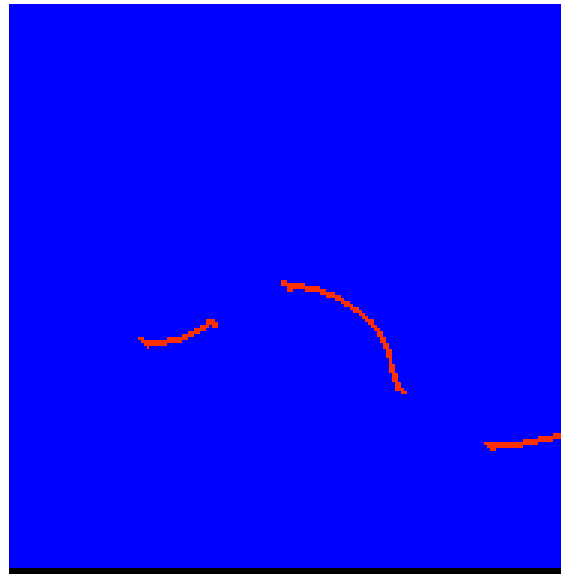
are found, almost half less, which improves the accuracy. For Yu Team's 100% noise quantity model, the result is similar to that of condition 1 and condition 2 alone.

5.2.Method 2's results

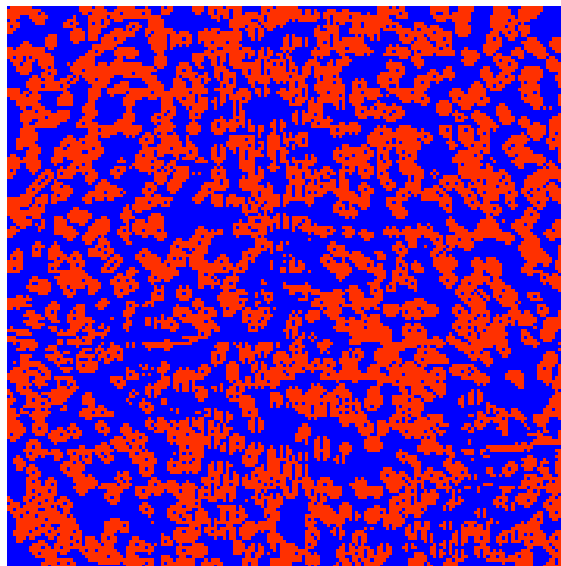
Through the processing of Method 1, I found that the number of irrelevant points has a greater impact on the results, so I used Method 2 to reduce the irrelevant points (Section 4.2).

5.2.1. The result of the first step of Method 2

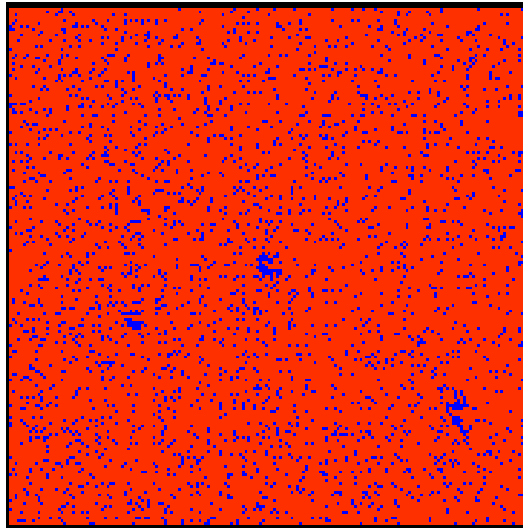
In Method 2, there are also two steps to deal with. The first step is to establish two planes based on the target point, and then calculate the similarity of the normals. There is a coefficient $K\theta$. The best coefficient has not been found yet. Here I define $K\theta=30^\circ$, the following are the results of the first step for the original model, 5% noise quantity model, and 100% noise quantity model(Fig. 44):



(a)



(b)



(c)

Fig. 44 Found points by the step1 of method 2.(a)Found points of original model.(b)Found points of 5% amount of noise model.(c)Found points of 100% amount of noise model

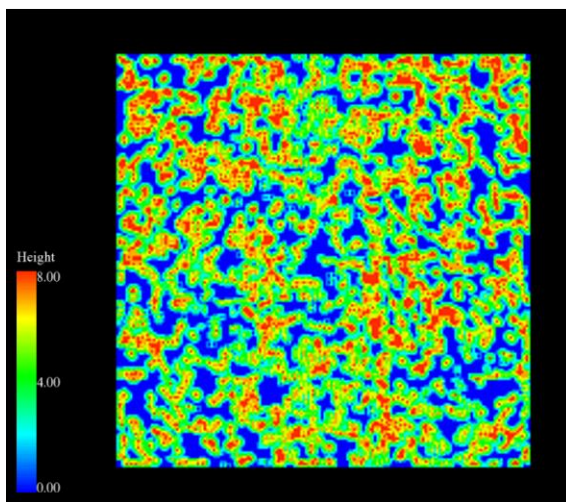
For the original model(Fig. 44(a)), the number of points found after processing in step 1 of method 2 is 235. For the 5% noise amount model(Fig. 44(b)), the number of points found is 18282. For the 100% noise amount model(Fig. 44(c)), the number of points found is 34225.

It can be found that compared with the processing of method 1, there are significantly less irrelevant points found after step 1 processing of method 2.

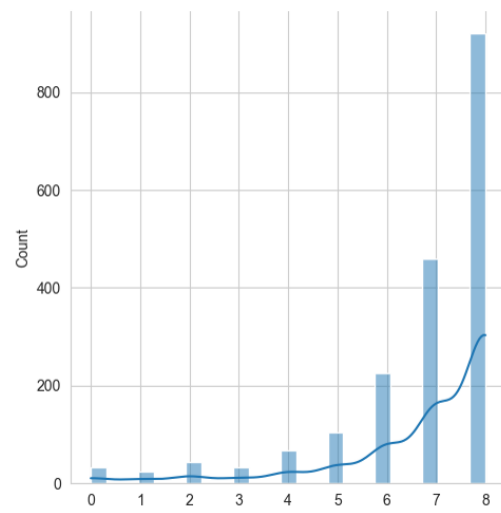
5.2.2. The result of the second step of Method 2

But one thing to note is that this is not the final result, because according to the theory I provided (Chapter 4.2.1), these points are points around the noise, so the result after step 1 processing requires step 2 (Chapter 4.2.2) Processing.

Next, As for the 5% amount of noise model and 100% amount of noise model, the result of step 2 will be shown(Fig. 45):



(a)



(b)

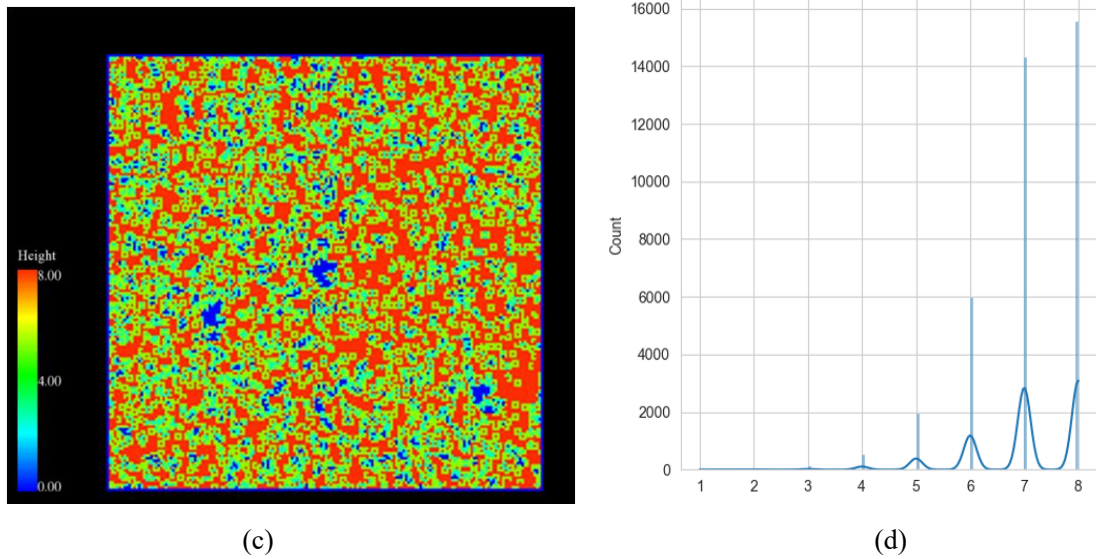


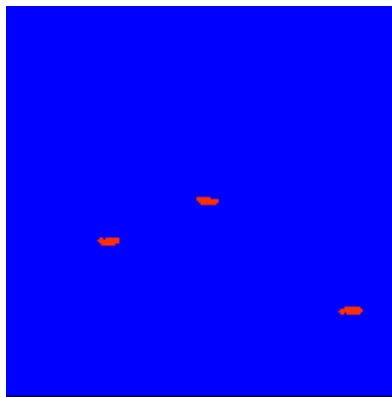
Fig. 45 Found points by the step2 of method 2.(a)Found points of 5% amount of noise model.(b) In the point graph found by the 5% noise model, the proportion of the noise placed in each value. (c)Found points of 100% amount of noise model.(d) In the point graph found by the 100% noise model, the proportion of the noise placed in each value.

Regardless of whether it is a model with 5% noise amount or a model with 100% noise amount, it can be found that the distribution law of the noise value is more in line with the theory of chapter 4.2.1.

5.3.Results of Method 1 && Method 2

In order to reduce the number of irrelevant points, I introduced method 2.In the step 2 of method 2,according to formulas (29)~(30), there is another undetermined coefficient $K_{\text{Superimpose}}$, where I set $K_{\text{Superimpose}}=3$.

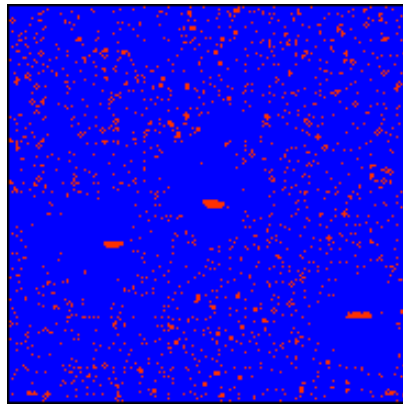
Next, I will show the results of Method 1 and Method 2 for the original model, 5% noise model and 100% noise model(Fig. 46):



(a)

A1(Amount of total points):40000
A2(Amount of noise I drop):0
P1(Found points):113
P2(Among found points,amount of noise):0
P3(Among found points,amount of Irrelevant points):P1-P2=113
R2(Among found points, ratio of Irrelevant points): P3/A1=0.0028

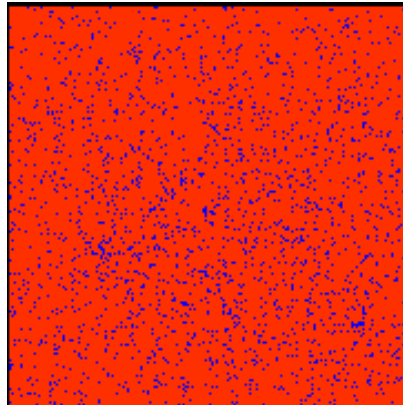
(b)



(c)

A1(Amount of total points):40000
A2(Amount of noise I drop):1910
P1(Found points):2017
P2(Among found points,amount of noise):1650
P3(Among found points,amount of Irrelevant points):P1-P2=367
R1(Among found points, ratio of noise): $P2/A2=0.8638$
R2(Among found points, ratio of Irrelevant points): $P3/A1=0.0091$
R3(Comprehensive ratio): $R1*(1-R2)=0.8558$

(d)



(e)

A1(Amount of total points):40000
A2(Amount of noise I drop):40000
P1(Found points):35249
P2(Among found points,amount of noise):35249
P3(Among found points,amount of Irrelevant points):P1-P2=0
R1(Among found points, ratio of noise): $P2/A2=0.881$
R2(Among found points, ratio of Irrelevant points): $P3/A1=0$
R3(Comprehensive ratio): $R1*(1-R2)=0.881$

(f)

Fig. 46 Found points by method 1 and method 2.(a)Found points of original model.(b) indicators data of original model. (c)Found points of 5% noise model.(d) indicators data of 5% noise model. (e)Found points of 100% noise model.(f) indicators data of 100% noise model.

It can be found that the combined treatment effect of method 1 and method 2 is better. For the original model(Fig. 46(a)(b)), only about 100 irrelevant points are found. I think this is an acceptable value. For the 5% noise quantity model(Fig. 46(c)(d)) and the 100% noise quantity model(Fig. 46(e)(f)), only most of the noise can be found, but not all the noise, but the number of irrelevant points found has also decreased.

5.4.Results of Best coefficients of Method 1 and Method 2

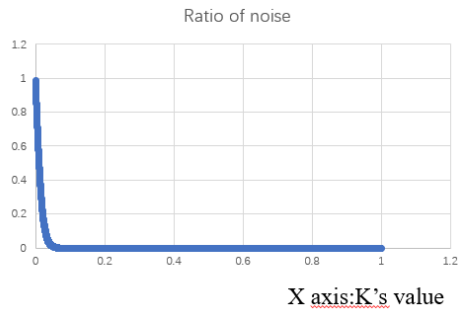
Because method 1 and method 2 are processed in a sequential order, I will show the best coefficient of method 1 first.

5.4.1. Results of Best coefficient of Method 1

R1, R2 and R3 are still used here as indicators. For the specific meaning, please refer to section 5.1. First, I apply the method 1 with different k values (k ranges from 0.0001 to 1) to the 5% noise quantity model, and 50% noise quantity Model and 100% noise quantity model. First observe whether there is a certain pattern and connection between them.

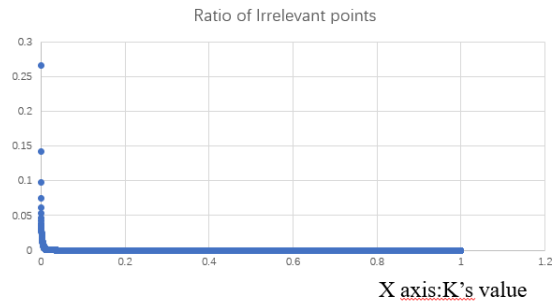
The first model to process is the 5% noise model(Fig. 47):

Y axis:R1(Among found points, ratio of noise)



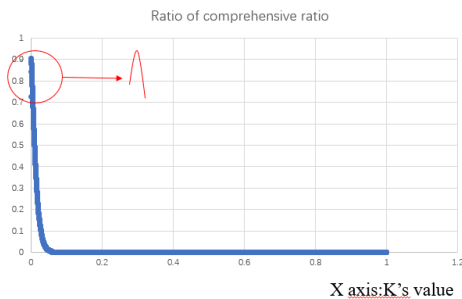
(a)

Y axis:R2(Among found points, ratio of irrelevant points)



(b)

Y axis:R3(Comprehensive ratio of R1 and R2) :R3=R1*(1-R2)



(c)

Best value

K's Value	R1	R2	R3	Column5	Column6	Column7	Column8
0.0006	0.954545	0.053233	0.903732	0.813807	0.952846	0.0411027	0.913682
0.0007	0.949448	0.0470117	0.904813	0.449293	0.947749	0.0361568	0.913481
0.0005	0.960493	0.061693	0.901237	0.531083	0.958794	0.0477666	0.912995
0.0008	0.942226	0.042196	0.902468	0.444746	0.940527	0.0325222	0.910193
0.0004	0.966665	0.0753332	0.894028	0.192114	0.965166	0.0577624	0.909415
0.0009	0.936703	0.0383955	0.900738	0.193457	0.935429	0.0293367	0.907987
0.001	0.93288	0.0352457	0.9	0.919492	0.932031	0.0270981	0.906774
0.0003	0.976211	0.0977457	0.88079	0.185888	0.973237	0.0733288	0.901871
0.0011	0.925234	0.0326166	0.895056	0.793359	0.924384	0.0252759	0.901019
0.0012	0.92141	0.0303519	0.893444	0.611164	0.920561	0.0234798	0.898946
0.0013	0.917162	0.0286079	0.890924	0.553728	0.916313	0.0223344	0.895847
0.0014	0.912489	0.0266556	0.888166	0.894986	0.91164	0.0209027	0.892584
0.0015	0.904843	0.0250156	0.882208	0.793573	0.903993	0.0197574	0.886133
0.0016	0.899745	0.0241045	0.878057	0.543321	0.898895	0.0191587	0.881674
0.0002	0.983432	0.141451	0.844324	0.000762963	0.97791	0.104904	0.875323
0.0017	0.892099	0.022803	0.871756	0.820093	0.891249	0.0182997	0.874939
0.0018	0.884452	0.0217097	0.865251	0.323283	0.883602	0.0175968	0.868054
0.0019	0.880629	0.0206424	0.86245	0.578387	0.880204	0.0167899	0.865425
0.002	0.874681	0.0200958	0.857104	0.723655	0.874257	0.0163734	0.859942
0.0021	0.870008	0.0195491	0.853001	0.442366	0.869584	0.0159829	0.855605
0.0022	0.865336	0.0190546	0.848847	0.663991	0.864911	0.0155404	0.85147
0.0023	0.859863	0.0179352	0.843558	0.841975	0.858539	0.0149157	0.845733
0.0024	0.853866	0.0172064	0.839174	0.351421	0.853441	0.0144471	0.841111
0.0025	0.850042	0.0163474	0.836147	0.477462	0.849618	0.0137703	0.837918
0.0026	0.845794	0.0158528	0.832386	0.282022	0.84537	0.0133017	0.834125

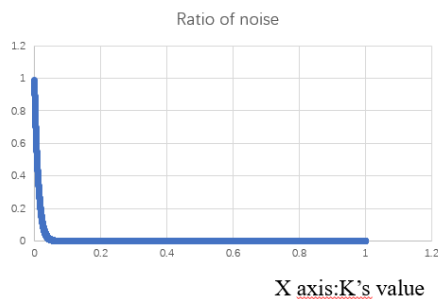
(d)

Fig. 47 Method 1 coefficient result graph.(a) R1 indicator.(b) R2 indicator. (c) R3 indicator.(d) Data sheet and optimal K value.

It can be seen that as the value of K increases, both R1 and R2 are decreasing because the points found decrease again. It should be noted that for R3, it first increases and then decreases, because there are too many points to observe this phenomenon. For a 5% noise model, the best K value is 0.0006.

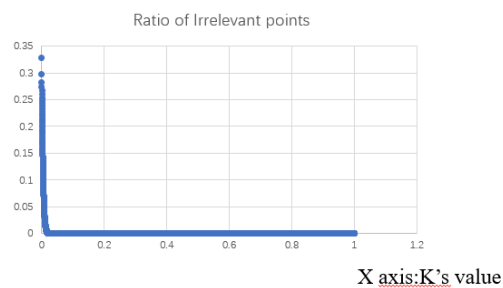
Next, process the 50% noise quantity model(Fig. 48):

Y axis:R1(Among found points, ratio of noise)



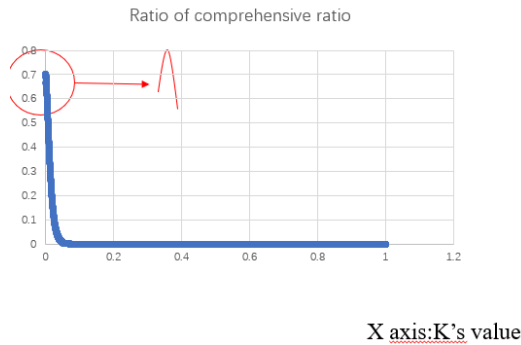
(a)

Y axis:R2(Among found points, ratio of irrelevant points)



(b)

Y axis:R3(Comprehensive ratio of R1 and R2) : $R3=R1*(1-R2)$



(c)

Best value

K's value	R1	R2	R3	Column5	Column6	Column7	Column8
0.0007	0.9404	0.253774	0.701751	0.925657	0.940093	0.253696	0.701595
0.0006	0.94533	0.26049	0.701299	0.156177	0.948023	0.260412	0.701146
0.0005	0.956362	0.266842	0.701164	0.137669	0.956055	0.266764	0.701014
0.0004	0.96511	0.274131	0.700544	0.823481	0.964803	0.274052	0.700396
0.0003	0.930936	0.248126	0.699947	0.481429	0.930629	0.248048	0.699788
0.0002	0.97735	0.282486	0.698499	0.042752	0.971193	0.282408	0.698355
0.0009	0.922853	0.243388	0.698241	0.70394	0.922546	0.24331	0.698081
0.001	0.916202	0.238104	0.698051	0.473678	0.915895	0.238026	0.697888
0.0011	0.908324	0.232976	0.697706	0.0317896	0.908017	0.232896	0.697542
0.0012	0.90189	0.238108	0.694849	0.717856	0.899802	0.23803	0.694602
0.0013	0.892157	0.223657	0.69262	0.191321	0.89185	0.223579	0.692452
0.0002	0.98189	0.296986	0.690283	0.466018	0.981563	0.296908	0.690144
0.0014	0.883665	0.219127	0.69003	0.83758	0.883358	0.219049	0.689859
0.0015	0.875991	0.214884	0.687754	0.214057	0.875684	0.214806	0.687582
0.0016	0.868215	0.211188	0.684858	0.0934782	0.867908	0.21111	0.684684
0.0017	0.860849	0.206015	0.682612	0.0645466	0.865411	0.206737	0.682396
0.0018	0.852407	0.202728	0.6796	0.91467	0.8521	0.20265	0.679422
0.0019	0.845296	0.198537	0.677473	0.800653	0.844989	0.198459	0.677293
0.002	0.837827	0.194424	0.674933	0.155553	0.83752	0.194346	0.674751
0.0021	0.830102	0.190484	0.671973	0.58661	0.829795	0.190415	0.671789
0.0022	0.822605	0.186459	0.670036	0.701102	0.822298	0.186381	0.669851
0.0023	0.816903	0.181721	0.668454	0.946287	0.816596	0.181643	0.668267
0.0001	0.996894	0.328092	0.665789	0.14832	0.995586	0.328014	0.665626
0.0024	0.809434	0.177843	0.665482	0.510575	0.809127	0.177764	0.665293
0.0025	0.802732	0.174172	0.662918	0.226081	0.802425	0.174094	0.662727
0.0026	0.79516	0.17084	0.659315	0.420067	0.794853	0.170762	0.659123
0.0027	0.789666	0.1673	0.657572	0.539659	0.789379	0.167222	0.657378
0.0028	0.782831	0.164333	0.654186	0.227363	0.782524	0.164254	0.653991

(d)

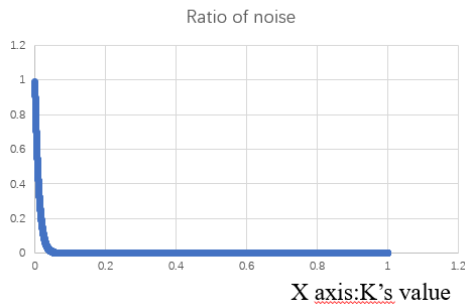
Fig. 48 Method 1 coefficient result graph.(a) R1 indicator.(b) R2 indicator. (c) R3 indicator.(d) Data sheet and optimal K value.

Similar to the 5% noise quantity model, as the value of K increases, both R1 and R2 decrease because the points found decrease. It should be noted that for R3, it first increases and then decreases, because there are too many points to observe this phenomenon. For a 50% noise model, the best K value is 0.0007.

Compared with the 5% noise quantity model, the overall R3 of the 50% noise quantity model is smaller than the 5% noise quantity model.

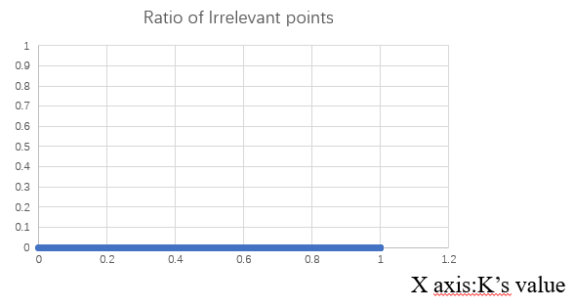
Next, process the 100% noise quantity model(Fig. 49):

Y axis:R1(Among found points.ratio of noise)



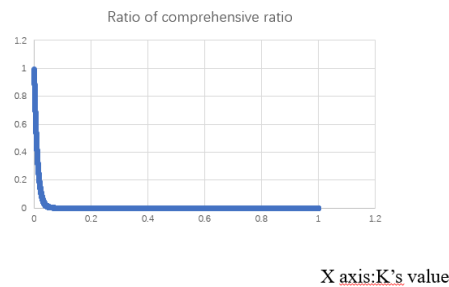
(a)

Y axis:R2(Among found points.ratio of irrelevant points)



(b)

Y axis:R3(Comprehensive ratio of R1 and R2) : $R3=R1*(1-R2)$



Best value

K's value	R1	R2	R3	Column5	Column6	Column7	Column8
0.0001	0.991696	0	0.991696	0.0484634	0.991176	0	0.991176
0.0002	0.983887	0	0.983887	0.67452	0.983366	0	0.983366
0.0003	0.975921	0	0.975921	0.0745872	0.975401	0	0.975401
0.0004	0.968268	0	0.968268	0.607748	0.967748	0	0.967748
0.0005	0.960772	0	0.960772	0.604419	0.960251	0	0.960251
0.0006	0.952702	0	0.952702	0.936868	0.952181	0	0.952181
0.0007	0.944867	0	0.944867	0.139592	0.944346	0	0.944346
0.0008	0.936693	0	0.936693	0.928709	0.936172	0	0.936172
0.0009	0.929665	0	0.929665	0.876827	0.929144	0	0.929144
0.001	0.921725	0	0.921725	0.0643941	0.921205	0	0.921205
0.0011	0.914254	0	0.914254	0.971648	0.913734	0	0.913734
0.0012	0.907018	0	0.907018	0.642109	0.906497	0	0.906497
0.0013	0.898766	0	0.898766	0.547502	0.898246	0	0.898246
0.0014	0.890879	0	0.890879	0.301218	0.890358	0	0.890358
0.0015	0.883304	0	0.883304	0.469802	0.882783	0	0.882783
0.0016	0.875911	0	0.875911	0.656636	0.87539	0	0.87539
0.0017	0.868414	0	0.868414	0.57271	0.867894	0	0.867894
0.0018	0.861204	0	0.861204	0.232093	0.860709	0	0.860709
0.0019	0.853837	0	0.853837	0.0188604	0.853342	0	0.853342
0.002	0.846522	0	0.846522	0.900357	0.846054	0	0.846054
0.0021	0.83965	0	0.83965	0.420637	0.839182	0	0.839182
0.0022	0.832648	0	0.832648	0.195227	0.832179	0	0.832179
0.0023	0.825724	0	0.825724	0.876247	0.825255	0	0.825255
0.0024	0.818825	0	0.818825	0.285287	0.818357	0	0.818357
0.0025	0.811953	0	0.811953	0.802911	0.811485	0	0.811485
0.0026	0.805498	0	0.805498	0.212531	0.805029	0	0.805029

(c)

(d)

Fig. 49 Method 1 coefficient result graph.(a) R1 indicator.(b) R2 indicator. (c) R3 indicator.(d) Data sheet and optimal K value.

Compared with the 5% noise quantity model and the 50% noise quantity model, the 100% noise model quantity is very special, because it has no irrelevant points, so R2 is always 0. As the value of K increases, the sum of R1 decreases, because it finds The points are reduced again. For the 100% noise amount model, R1 and R3 are the same. For a 100% noise model, the best K value is 0.0001.

Compared with the 5% noise quantity model and the 50% noise quantity model, the overall R3 of the 100% noise quantity model is greater than the 5% noise quantity model and the 50% noise quantity model.

Next, I will put the amount of noise from 1% to 100%, and find the best coefficient every time, and use this method to view the change law of the best K value(Fig. 50):

Y axis:K's value

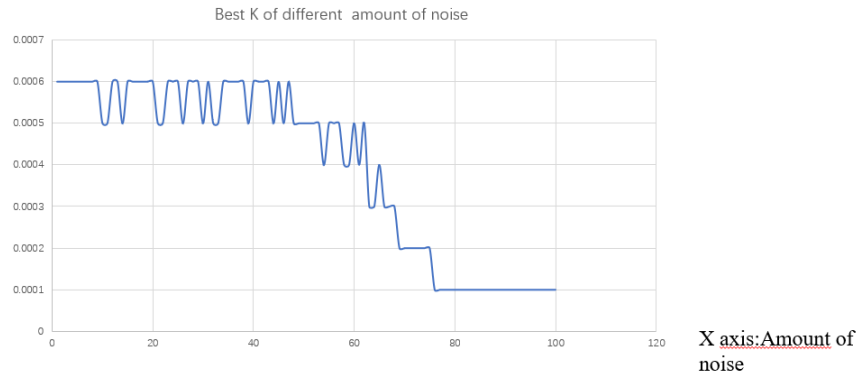


Fig. 50 Best k of different amount of noise

The overall data shows a downward trend.

I use the formula $R3=R1*(1-R2)$ to get the best K value. When the amount of noise increases, R2 drops faster, and the weight is lower; and the less the K value, the more points can be found, and the larger R1 is, so the larger R3 is.

5.4.2. Results of Best coefficient of Method 2

R33 is still used here as indicators which is same as R3. For the specific meaning, please refer to section 5.1. First, I apply the method 2 with different k values (K_{θ} ranges from 0° ~ 90° ; $K_{\text{Superimpose}}$ ranges from 0 ~8) to the 5% noise quantity model, and 50% noise quantity Model and 100% noise quantity model. First observe whether there is a certain pattern and connection between them.

The first model to process is the 5% noise model(Fig. 51):

Z axis: R33(Comprehensive ratio)

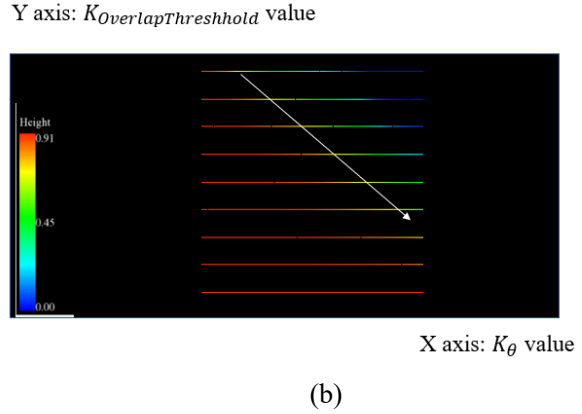
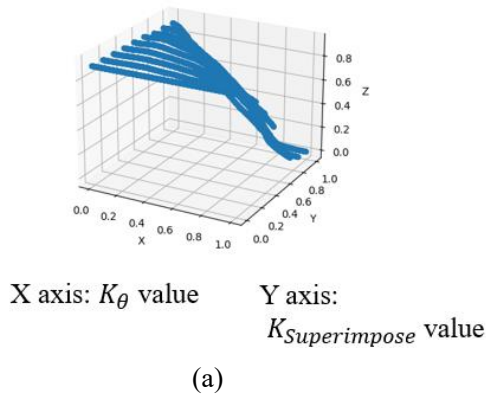


Fig. 51 Method 2 coefficient result graph.(a) R33 indicator.(b) R33 top view

As shown in Fig. 51(a), both coefficients are normalized here. K_θ and $K_{Superimpose}$ are in inverse proportion. The reason is that when the value of K_θ is smaller, more points will be found. In this case, the overlap value of noise will be higher, so it is necessary to increase $K_{Superimpose}$ to increase the value of R3.

When the values of K_θ and $K_{Superimpose}$ are very small, R3 is very high, because in this case method 2 is almost ineffective, no increase nor decrease.

Next, process the 50% noise quantity model(Fig. 52):

Z axis: R33(Comprehensive ratio)

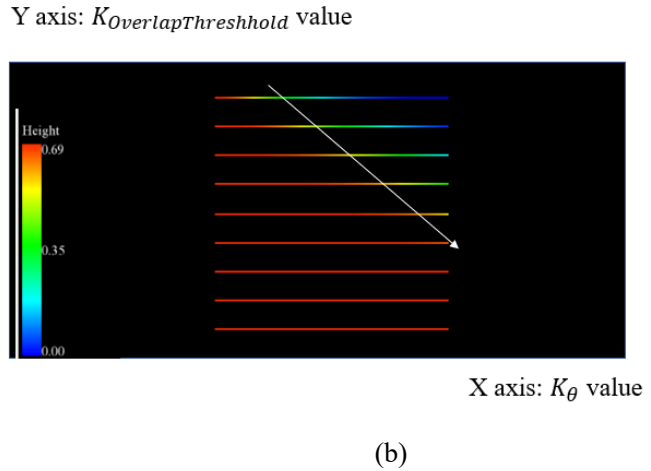
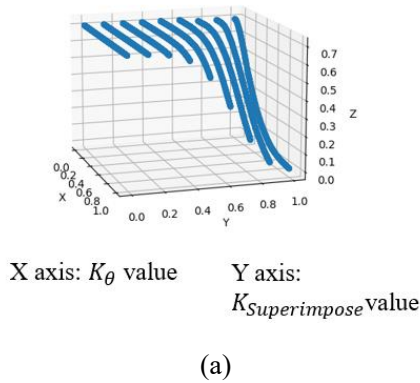


Fig. 52 Method 2 coefficient result graph.(a) R33 indicator.(b) R33 top view

It can be found that the result of the 50% noise amount model processing is similar to the 5% noise amount model. Next, process the 100% noise quantity model(Fig. 53):

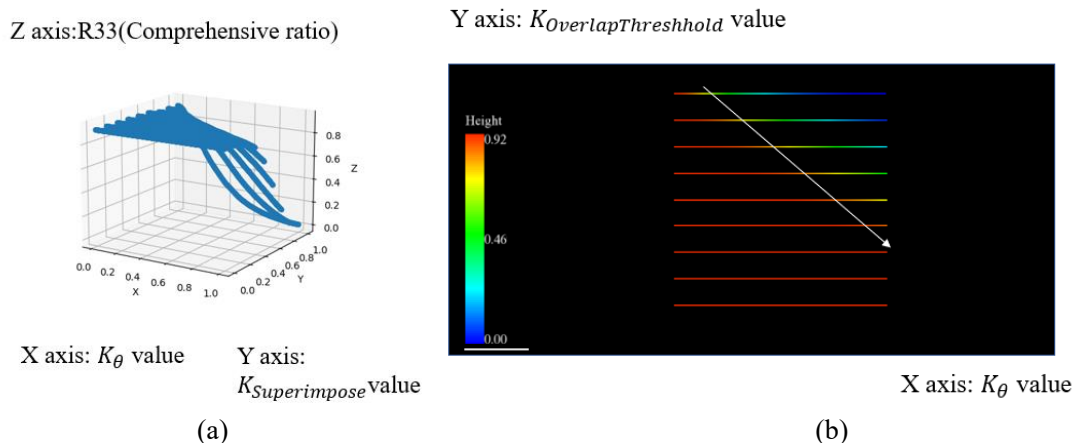


Fig. 53 Method 2 coefficient result graph.(a) R33 indicator.(b) R33 top view

Combining the 5% quantity noise model, 50% noise quantity model and 100% noise quantity model, I found that their laws are similar.

Next, I will put the amount of noise from 1% to 100%, and find the best coefficient every time, and use this method to view the change law of the best K value(Fig. 54):

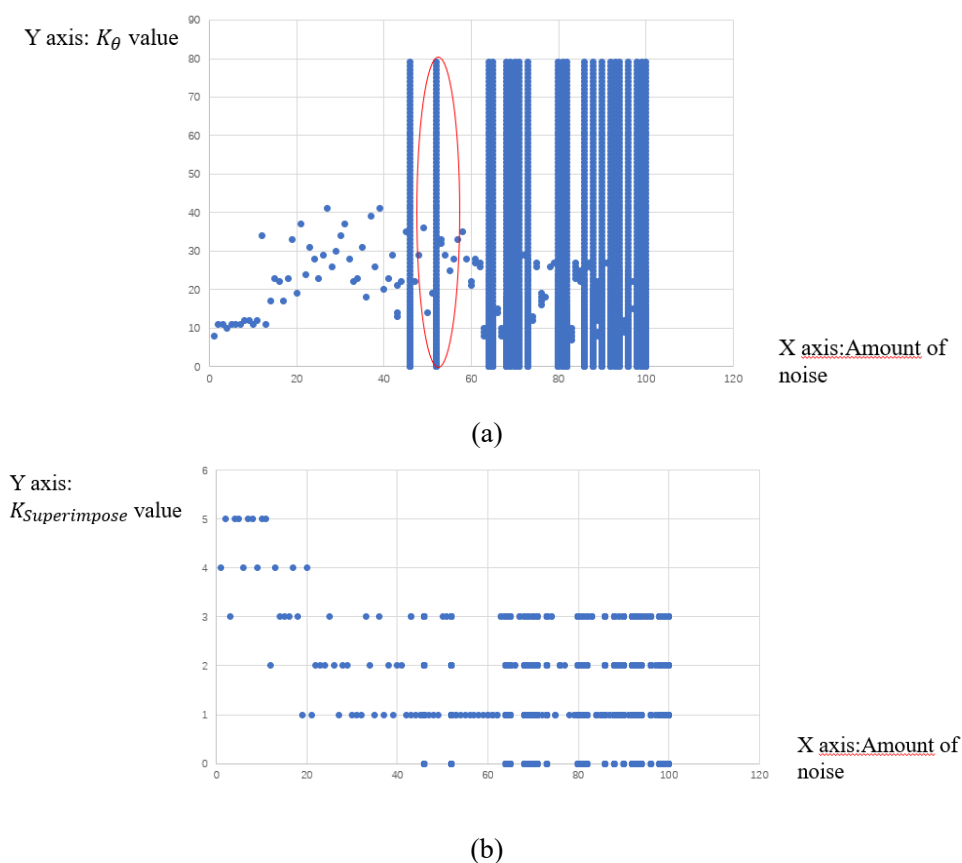


Fig. 54 Best k of different amount of noise.(a)Best K_{θ} of different amount of noise.(b)Best $K_{Superimpose}$ of different amount of noise

Different K_{θ} and $K_{Superimpose}$ may also get the same best R33 value.

The value in the red circle means that function 2 has no effect here. No matter what the values of K_{θ} and $K_{Superimpose}$ are, the best R33 value that can be obtained is R3 itself, without any improvement.

The more the amount of noise points, the more cases the method 2 is invalid. The reason is that the principle of function 2 is to minimize irrelevant points. When the noise is more, the irrelevant points are less, and the effect of function 2 is naturally worse.

As for the $K_{Superimpose}$, the overall data shows a downward trend. When the amount of noise points is limited, the points found in the function2 are also limited, so the $K_{Superimpose}$ value does not need to be too large.

5.4.3. Results of Best coefficients of Method 2 and Method 1

Now that I have obtained the best coefficients of Method 1 and Method 2 under different amounts of noise, I will now apply these coefficients to the method, get the results and compare them.

The first comparison is R1, R11 and R44, in other words, the ratio of the amount of noise. The result is shown in the figure below(Fig. 55):

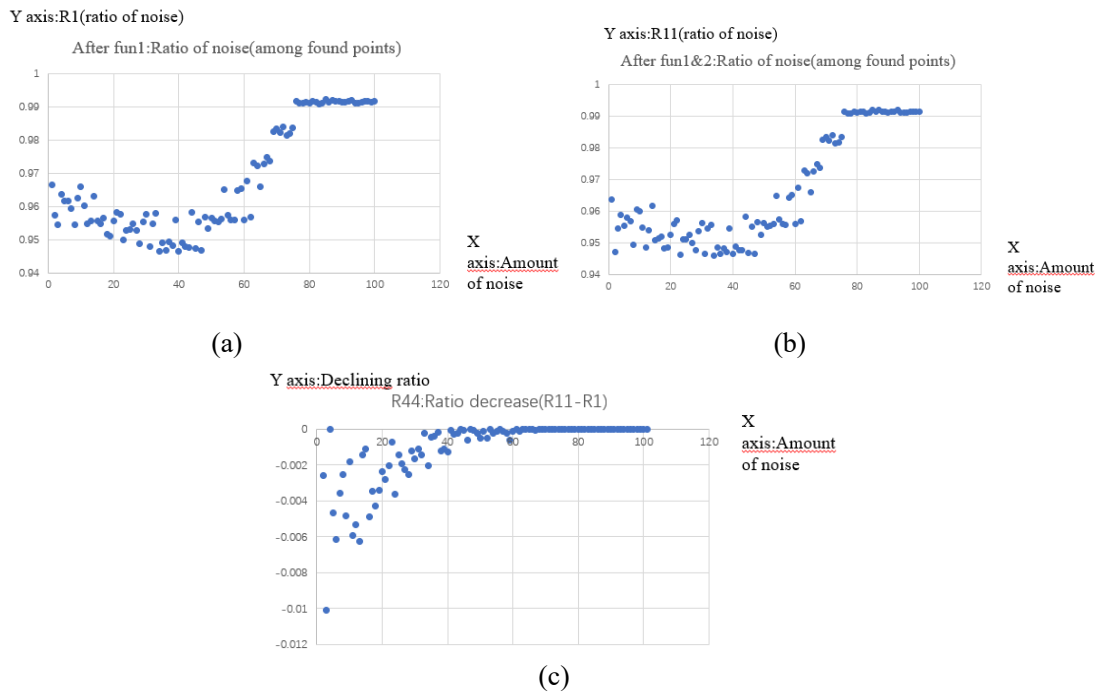


Fig. 55 Ratio of amount of noise by method 1 and method 1&&method 2.(a)Ratio of amount of noise by method 1.(b)Ratio of amount of noise by method 1&&method 2.(3)Comparison of (a) and (b)

From the above figure, it can be found that for method 1, when the amount of noise is less than 80%, the accuracy of R1 decreases first, and then rises, because method 1 is very sensitive. When the amount of noise is very small, the noise can be found more accurately. When the amount of noise is large, multiple noises affect the judgment of Method 1, so there is a slight decrease. When the amount of noise is greater than 80%, since almost all points are noise, the accuracy rate is very high, almost 100%. On the whole, the accuracy of R1 is above 90%, and this result is acceptable.

For method 2, after the comprehensive processing of method 1 and method 2, the result obtained is similar to that of method 1 only. The purpose and essence of Method 2 is to reduce irrelevant points. For R1, Method 2 sometimes removes noise points by mistake, resulting in R11 accuracy rate sometimes lower than R1.

By comparing R44, it can be found that method 2 will reduce the proportion of the amount of noise found in method 1, but this situation can be accepted because the degree of reduction is very low.

The next thing to compare is the results of R2, R22 and R55, which is the ratio of irrelevant points. The result is shown in the figure below(Fig. 56):

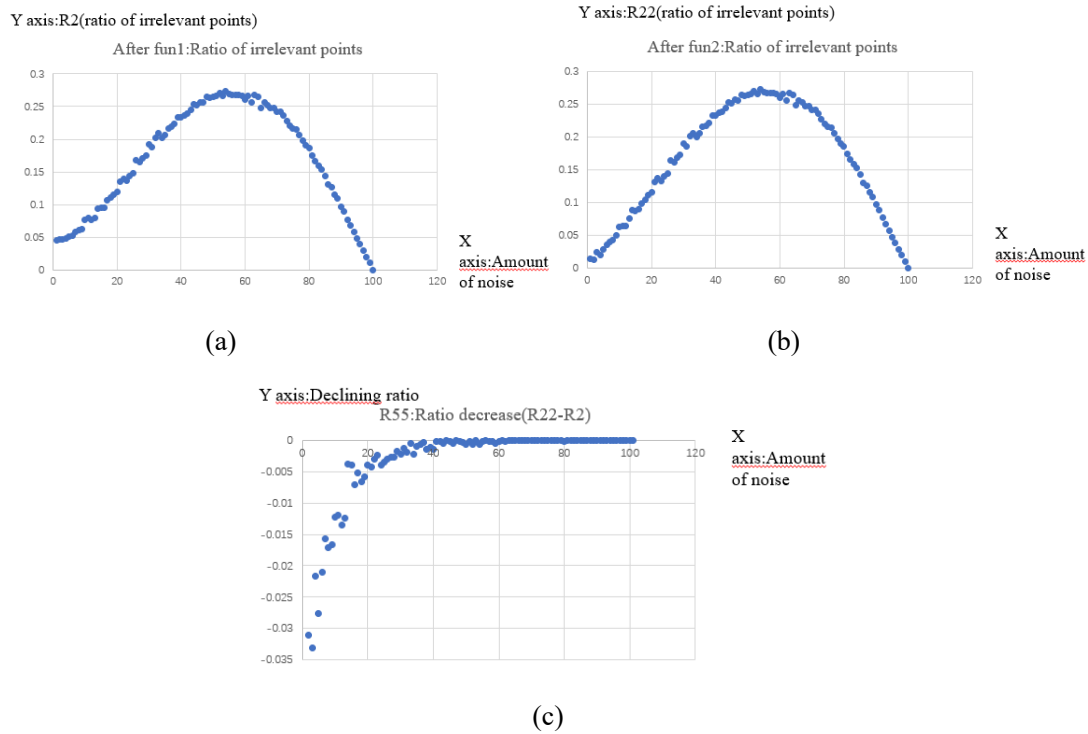


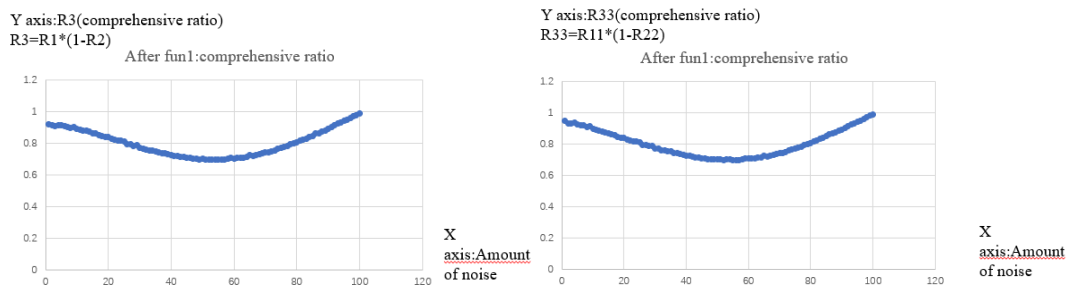
Fig. 56 Ratio of amount of irrelevant points by method 1 and method 1&&method 2.(a)Ratio of amount of irrelevant points by method 1.(b)Ratio of amount of irrelevant points by method 1&&method 2.(3)Comparison of (a) and (b)

From the above figure, it can be found that for method 1, when the number of noise is less than 60%, the number of irrelevant points is increasing, and when the number of noise is higher than 60%, the number of irrelevant points found is decreasing. This is because when the amount of noise increases, the number of irrelevant points itself decreases.

For Method 2, the overall trend of the data is very similar to Method 1. When the number of noise points is less than 60%, the number of irrelevant points increases, and when the number of noise points is higher than 60%, the number of irrelevant points found decreases. The difference is that when the number of noise points is small, the effect of method 2 is very significant.

It can be seen from R55 that when the amount of noise is less than 60%, method 2 has a significant improvement over method 1. In other words, it can effectively reduce the number of irrelevant points. When the amount of noise is higher than 60%, method 2 has almost no effect.

The next comparison is R3, R33 and R66, in other words, the comprehensive accurate value. The result is shown in the figure below(Fig. 57):



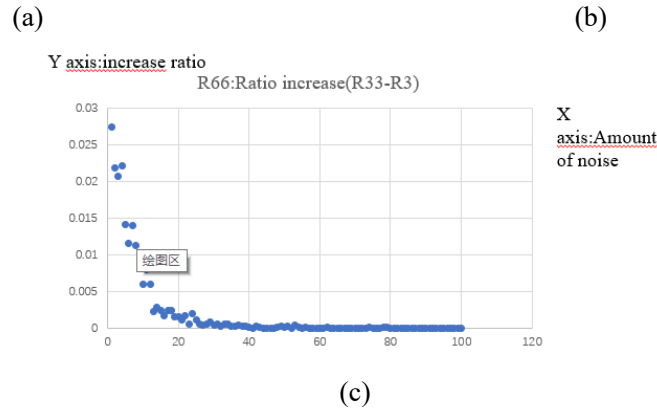


Fig. 57 Comprehensive ratio by method 1 and method 1&&method 2.(a) Comprehensive ratio by method 1.(b) Comprehensive ratio by method 1&&method 2.(3)Comparison of (a) and (b)

As can be seen from the above figure, for Method 1, when the number of noise points is less than about 60%, R3 is decreasing, and when the number of noise points is more than 60%, R3 is increasing. The data trend of method 2 is similar to that of method 1. When the number of noise points is less than about 60%, R33 decreases, and when the number of noise points exceeds 60%, R33 increases. The difference is that when the number of noise points is less than 60%, R33 drops more gently. It can be seen from R66 that when the amount of noise is less than about 60%, method 2 improves the overall result again, and the improvement effect is more obvious. When the amount of noise exceeds about 60%, method 2 has almost no effect because the number of irrelevant points is decreasing.

5.5.The result of inferring the real amount of noise

Through the results of the previous chapter, I got the data table of the best results obtained by the best coefficient when the amount of noise is from 1% to 100%. From this data sheet, I can deduce the real amount of noise, and the results are as follows(Fig. 58):

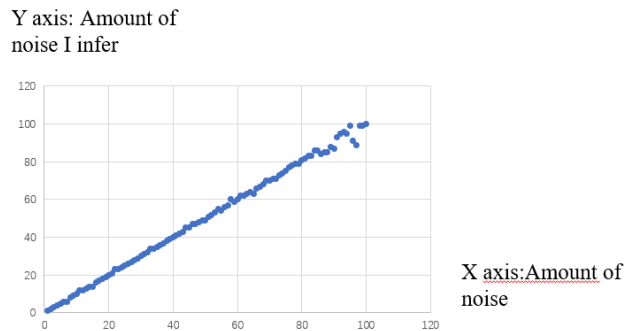


Fig. 58 The accuracy of deriving the amount of noise

The process of deriving the amount of noise can be found in section 4.4. In the previous chapters, I got the best coefficients of method 1 and method 2 through the loop method. Then apply the best coefficients under the conditions of different amounts of noise to get the best results. Reverse the actual amount of noise through the result.

As can be seen from the above figure, before 80% of the noise amount, the result is very accurate, but after 80% of the noise amount, the result is not accurate. The reason can be referred to Fig. 55. It can be seen that the noise ratio is almost 99% after 80%. In this case, the result of the reverse inference may be inaccurate. But I think this result is acceptable overall.

And I think the most important thing is that the application of this method does not need to use the original model (model without noise). It is possible for us to verify the smoothing noise in the real model

Use:

- Used to make strategies for smoothing noise
- Used to infer the distribution of noise

Both of these uses do not need to use the original model (no noise model), in other words, it can be applied under real situation.

5.6.RMS comparison of the four methods and data cross-section comparison

I will use 4 methods to smooth the noise, and use RMS to evaluate. Among the four methods, two of them are traditional methods, traditional Gaussian filtering methods and traditional median filtering methods. There are two others that I improved the traditional method to obtain, which I call "My Gaussian Filtering Method" and "My Median Filtering Method". The traditional filtering method will smooth all the points of the InSAR image, but my improvement is to combine method 1 and method 2, and only smooth the points found. This is the only difference.

First, I applied the four methods to the 5%, 20% and 100% noise amount models, and then observed the RMS results, as shown in the following figure(Fig. 59):

Methods	RMS(less,better)	Methods	RMS(less,better)
Traditional Median filtering	0.000133232	Traditional Median filtering	0.000196438
Traditional Gaussian mean filter	0.000122948	Traditional Gaussian mean filter	0.000195219
My Median filter	0.000115858	My Median filter	0.000172552
My Gaussian filter	9.64388e-05	My Gaussian filter	0.000154954

(a)

(b)

Methods	RMS(less,better)
Traditional Median filtering	0.00052553
Traditional Gaussian mean filter	0.000409563
My Median filter	0.000543443
My Gaussian filter	0.000440655

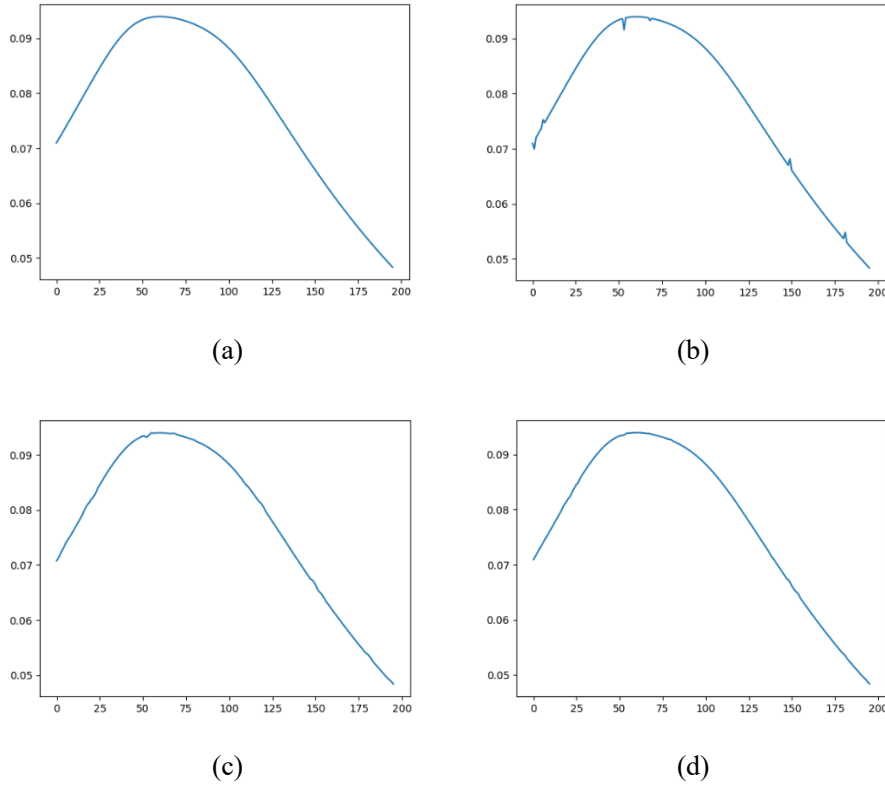
(c)

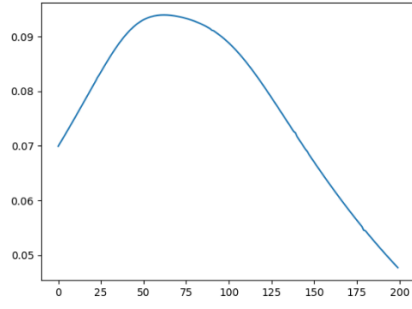
Fig. 59 Comparison of the RMS values obtained by the four methods under 5%, 20%, and 100% noise amount(a) 5% of the amount of noise(b) 20% of the amount of noise(c) 100% of the amount of noise

It can be seen from Fig. 58(a)(b) that when the amount of noise is small, the Gaussian filtering method is better than the median filtering method, both for the traditional method and for my method. Secondly, my two methods are better than traditional methods. So my Gaussian filtering method works best.

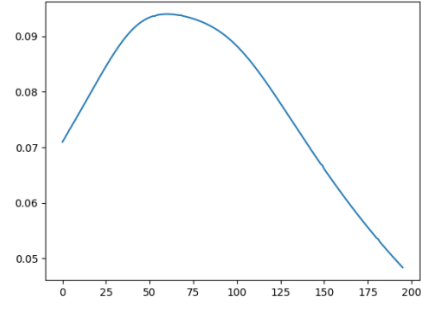
It can be seen from Fig. 58(c) that when the amount of noise is 100%, both the traditional method and my method, the Gaussian filtering method is better than the median filtering method. But the traditional methods are better than mine at this time. So the traditional Gaussian filtering method has the best effect. The reason is that my method cannot find all the noise, so when the amount of noise is large, the effect is not as good as the traditional method.

In addition to RMS, I also used cross-sectional data to face the opposite, because the cross-sectional data is more intuitive. In this article, the cross-sections I chose are all the cross-sectional data in the 100th row. The cross-sectional view of the 5% noise amount is shown below(Fig. 60):





(e)

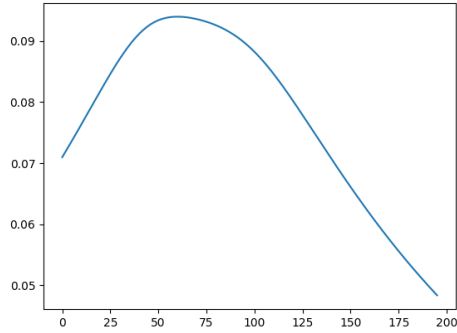


(f)

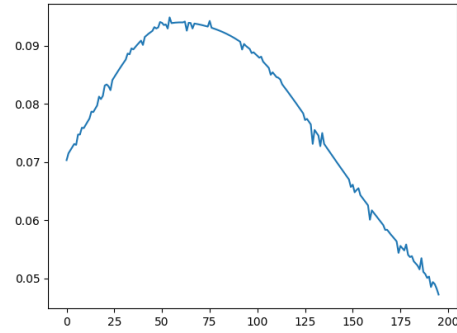
Fig. 60 Cross-sectional data of the 5% noise model in line 100.(a)Original model(without noise).(b)Noise model.(c) Model after traditional Gaussian filtering.(d) Model after traditional Median filtering.(e) Model after my Gaussian filtering.(f) Model after my median filtering.

It can be seen from Fig. 60(b) that the amount of noise of 5% has little effect on the data, and almost all noises can be very effective in noise reduction. Compared with the traditional method, my method has slight fluctuations in the data processed by the traditional method, and my method is almost the same as the original model data after processing.

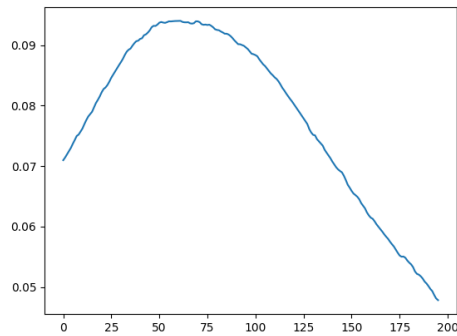
Next, the cross-sectional view of the 20% noise amount is shown below(Fig. 61):



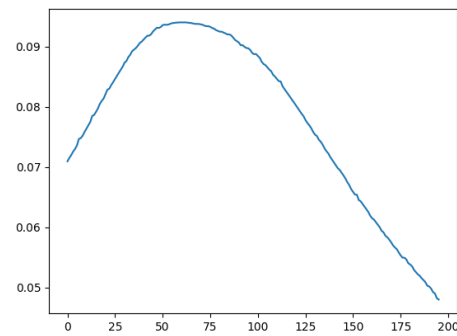
(a)



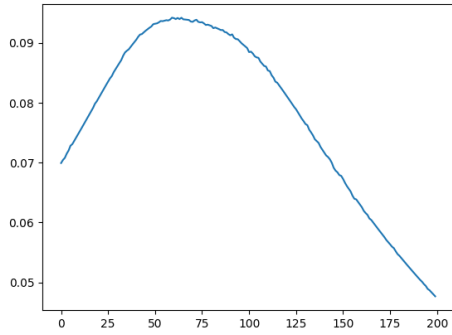
(b)



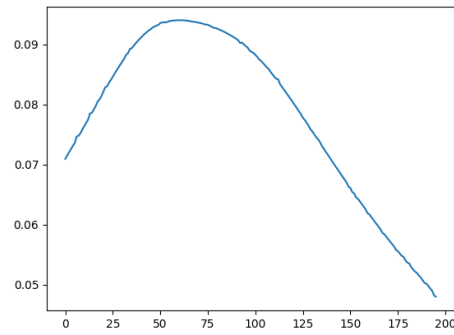
(c)



(d)



(e)

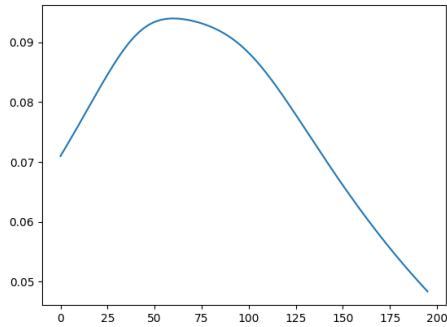


(f)

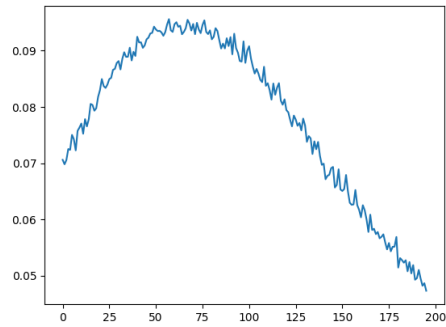
Fig. 61 Cross-sectional data of the 20% noise model in line 100.(a)Original model(without noise).(b)Noise model.(c) Model after traditional Gaussian filtering.(d) Model after traditional Median filtering.(e) Model after my Gaussian filtering.(f) Model after my median filtering.

As can be seen from the above figure, the amount of 20% noise has a greater impact on the original model data, and the data fluctuates more obviously. From the processing results, the traditional filtering method can filter some noise, but the effect is not particularly good. My method is obviously better than the traditional method of smoothing. Judging from the cross-sectional data, the median filtering method is better than the Gaussian filtering method. This is contrary to the conclusion obtained by RMS. I think the reason is that RMS belongs to the traditional evaluation method. For InSAR images, the evaluation conclusion of RMS is only Occupying one aspect, there are other aspects, but no other evaluation method has been found so far.

Next, the cross-sectional view of the 100% noise amount is shown below(Fig. 62):



(a)



(b)

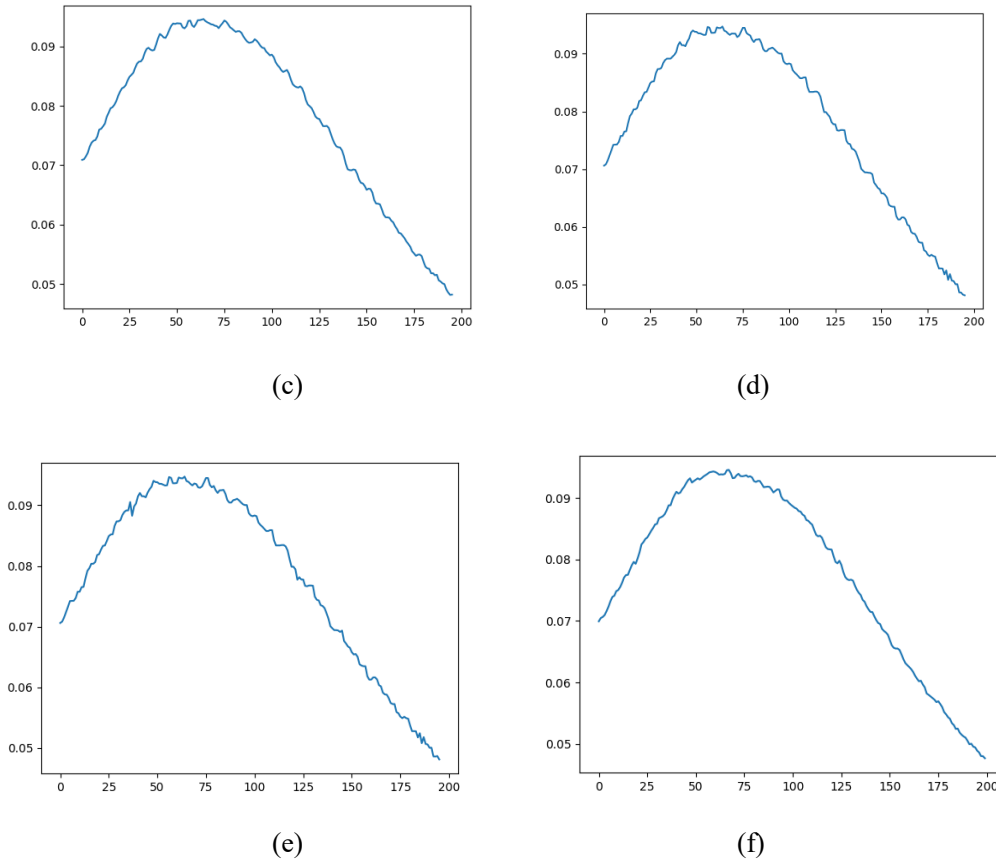


Fig. 62 Cross-sectional data of the 100% noise model in line 100.(a)Original model(without noise).(b)Noise model.(c) Model after traditional Gaussian filtering.(d) Model after traditional Median filtering.(e) Model after my Gaussian filtering.(f) Model after my median filtering.

It can be seen from the above figure that the amount of 100% noise completely changes the shape of the original model data and has a very serious impact on the original model data. Judging from the processing results, the smoothing results of all the smoothing methods are not ideal when only one smoothing is performed, and it can be easily seen that there is still more noise. Comparing the four methods, it can be found that the effect of the Gaussian filtering method is significantly better than the median filtering method. And my Gaussian filtering method also looks better than the traditional Gaussian filtering method. This conclusion is also contrary to the conclusion obtained by RMS. I think the reason may be that my method ignores some noises with limited values.

I can find some rules through these three examples, but I cannot explain and explain the relationship between the amount of noise and the pros and cons of the method in detail. Below I will record the relationship between the amount of noise and the method in detail from 1% to 100%.

The following will show the results after filtering(Fig. 63):

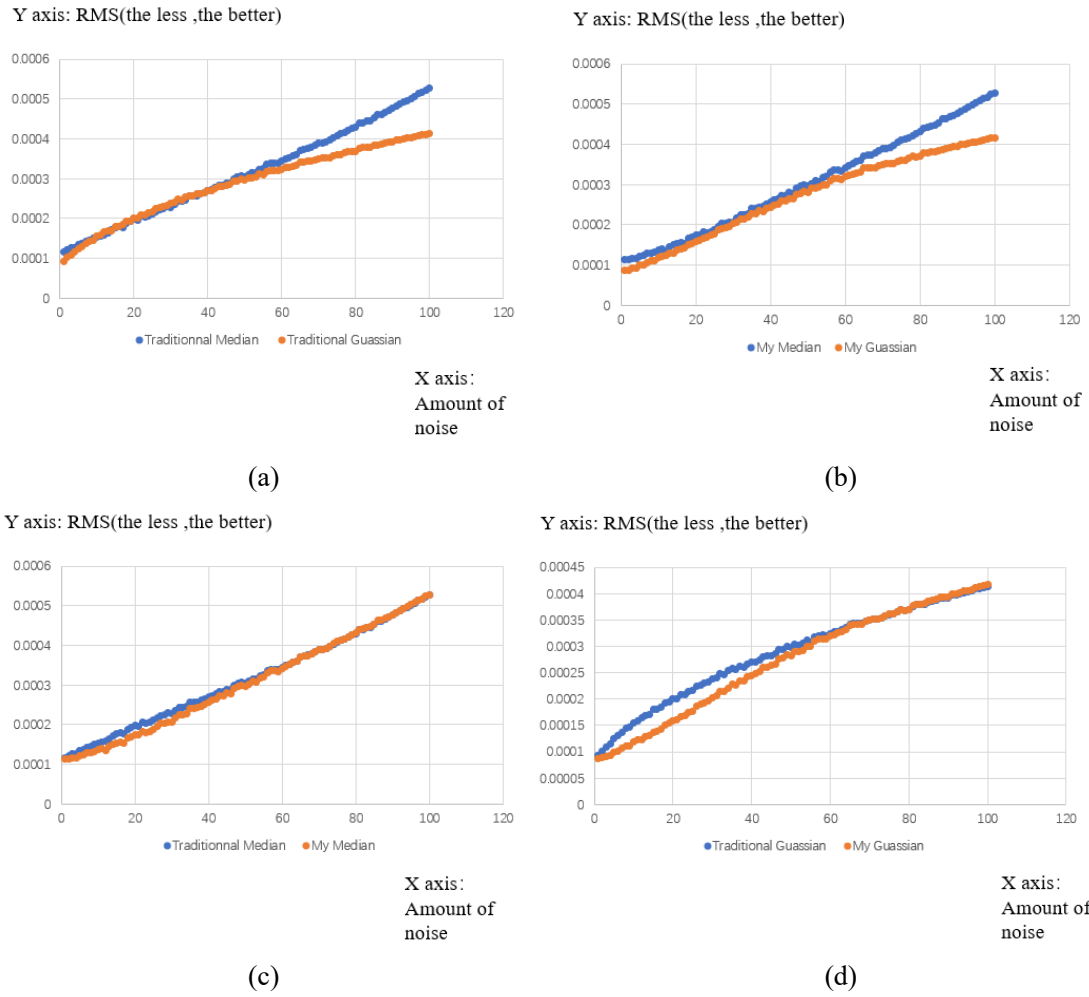


Fig. 63 RMS graph after processing by 4 methods.(a)RMS Comparison chart of traditional methods.(b)RMS Comparison chart of my method(c) RMS comparison chart of traditional median filtering method and my median filtering method.(d) RMS comparison chart of traditional Gaussian filtering method and my Gaussian filtering method.

It can be seen from Fig. 63(a) that for traditional methods, regardless of the amount of noise, the results of traditional Gaussian filtering are better than traditional median filtering. When the amount of noise is very small, the traditional median filtering effect is better.

It can be seen from Fig. 63(b) that for my method, the results obtained are similar to those of the traditional method, and the traditional Gaussian filtering is better than the traditional median filtering. Of course, this is only the result of the RMS evaluation.

It can be seen from Fig. 63(c) that compared with the traditional median filtering method, when the number of noise points is less than 60%, the processing result of my median filtering method is slightly better than the traditional median filtering method. . When the amount of noise is more than 60%, the result of my median filtering method is similar to that of the traditional median filtering method.

It can be seen from Fig. 63(d) that compared with the traditional Gaussian filtering method, when the number of noise points is less than 60%, the processing result of my Gaussian filtering method is significantly better than that of the traditional Gaussian filtering method. . When the amount of noise is more than 60%, the result of my Gaussian filtering method is similar to that of the traditional Gaussian filtering method.

60% is a magic number, and I will discuss this in the "Additional discussion" section.

5.7.The result after processing through the smoothing strategies

It can be known from the previous chapter that when the amount of noise is very large, no matter which smoothing method is used, smoothing once is not enough to get the desired result. So I introduced the smoothing strategy. In other words, it is a strategy that determines whether to smooth again by judging the amount of noise (Chapter 4.5.5). Next, I will conduct an experiment. The experimental data are set as follows:

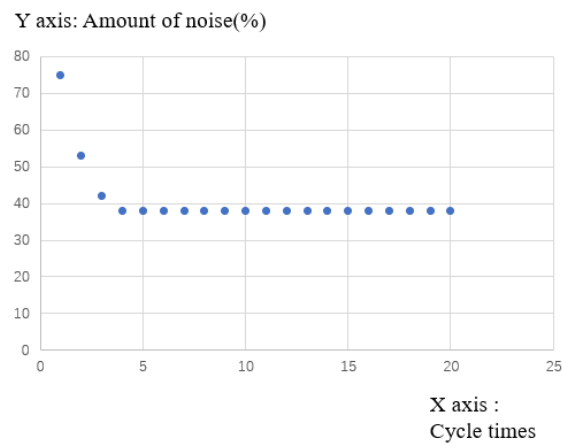
The number of noise input: 100%.

Maximum number of cycles: 20.

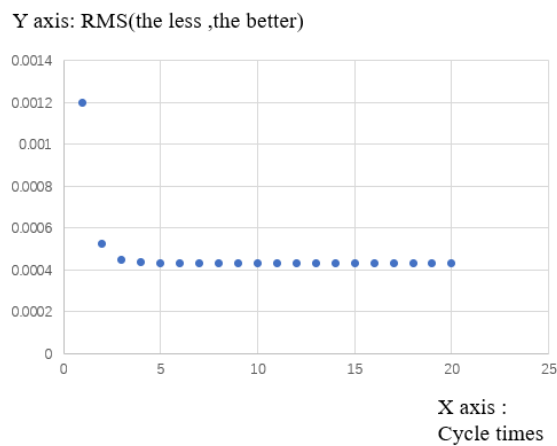
The number of critical noise exiting the loop: 1%.

I used 4 different methods to conduct experiments, and the results of the experiments are shown in terms of RMS and changes in the amount of noise.

The first to show is the traditional median filtering method(Fig. 64):



(a)



(b)

Fig. 64 The result of traditional median filter processing.(a)result of amount of noise.(b)result of RMS

It can be seen from Fig. 64(a) that the amount of noise drops rapidly in the first few cycles, and when it reaches about 38%, there is no more change. It can be seen from Fig. 64(b) that, similar to

Fig.52(a), the RMS drops rapidly at the beginning of the cycle, and then no longer changes. I guess the reason is that the median filtering method itself does not generate new values, but only finds intermediate values to replace the original multiple values. When the amount of noise is large, the effect is not ideal.

Next, the cross-sectional data will be displayed, which can more intuitively reflect the quality of the smoothing effect(Fig. 65):

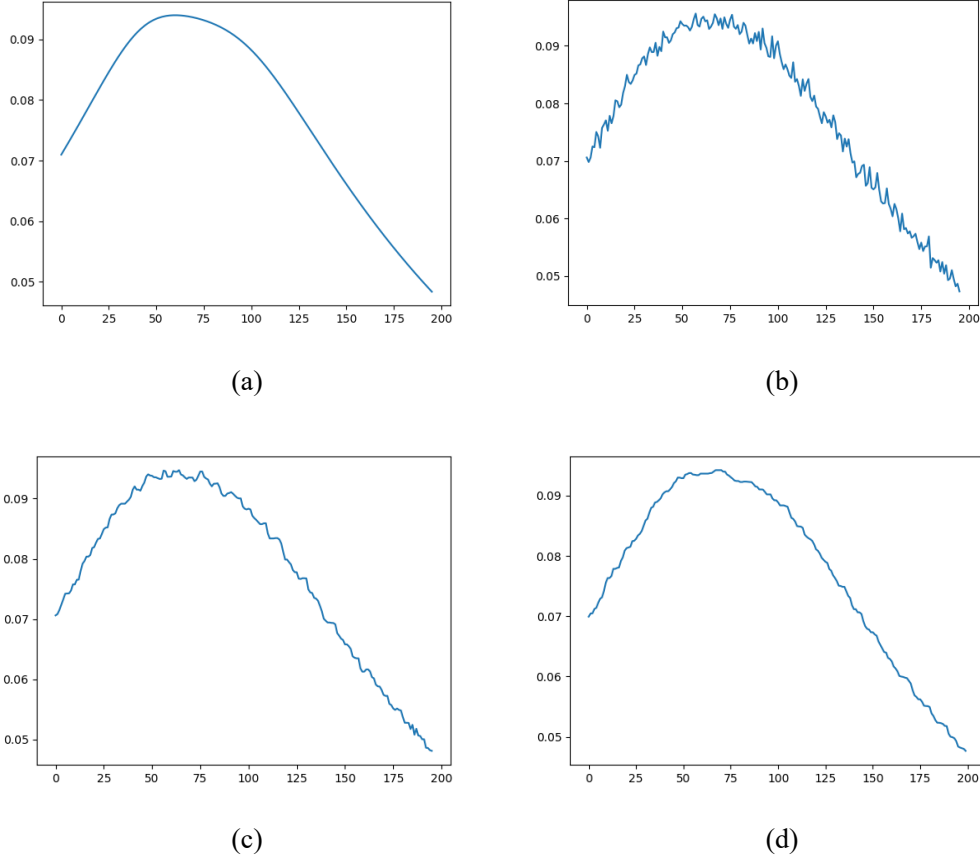
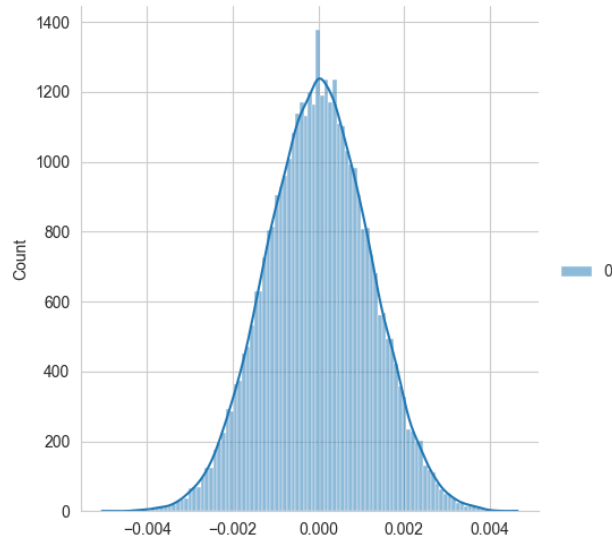


Fig. 65 Cross-sectional data of the 100% noise model in line 100.(a)Original model(without noise).(b)Noise model.(c) Model after traditional median filtering(1 time).(d) Model after traditional median filtering(20 times).

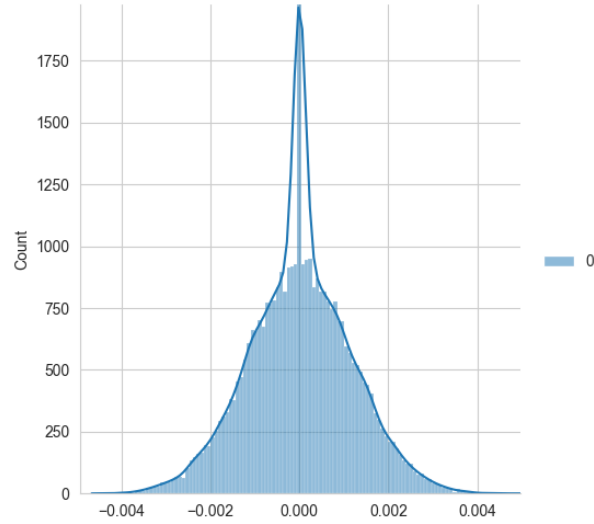
It can be seen from Fig. 65(b) that the amount of 100% noise has a serious impact on the value and shape of the original model. Fig. 65(c) shows the result of filtering only once using the traditional median filtering method. It can be seen that the noise has been reduced, but the overall data is not smooth enough. After using the strategy I developed (Fig. 65(d)), I cycled a total of 20 times, that is, smoothed 20 times. It can be seen that the data is smoother than the result of only smoothing once. But it is still not ideal. The reason is that the median filtering method itself does not generate new values.

In section 5.5, I said that the method of inferring the amount of noise has two uses. One is to provide a basis for the strategy of smoothing multiple times; the other is to reverse the distribution of real noise. The principle is: $\phi_{ModelwithNoise}(x, y) - \phi_{AfterProcessing}(x, y)$

Next, I will show the noise distribution inversely deduced by traditional median filtering strategy processing (Fig. 66):



(a)

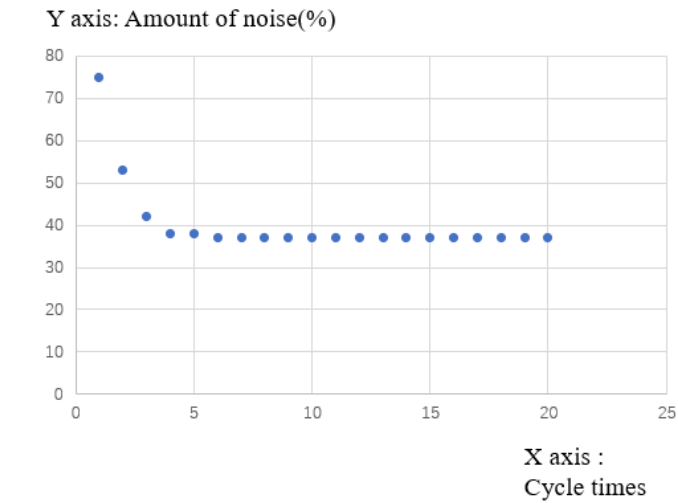


(b)

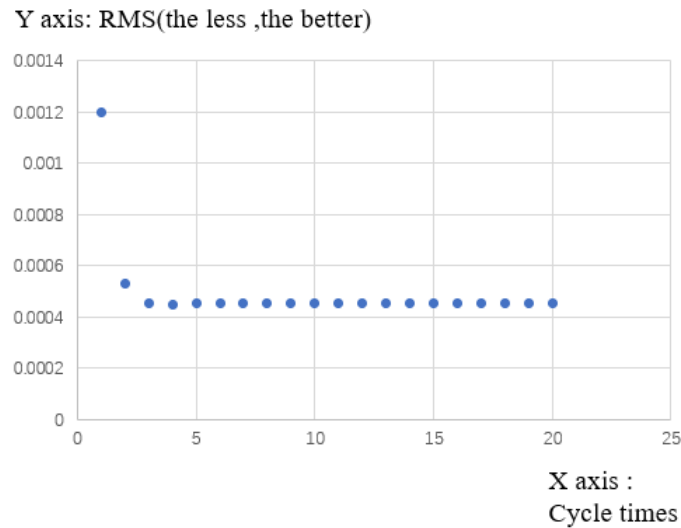
Fig. 66 Noise Distribution.(a) Artificial noise distribution.(b) The noise distribution derived from the result of multiple smoothing through the strategy

In this strategy experiment, the noise I put in was a normally distributed random noise with a standard deviation of $2.4E-04$. It can be seen from Fig. 66(a) that the noise distribution is indeed a normal distribution. Fig. 66(b) shows the noise distribution obtained by inverse deduction. It can be seen that the number of points with a value of 0 is very large. I think this is very consistent with the characteristics of median filtering. The overall shape is still somewhat similar to the normal distribution.

Next, I will show the application of my median filtering method to the strategy(Fig. 67):



(a)



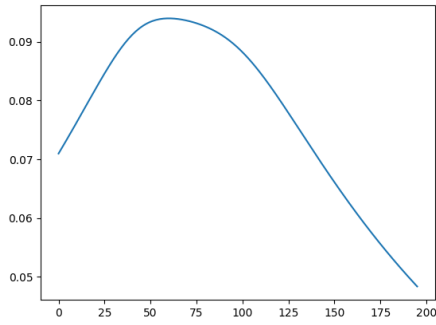
(b)

Fig. 67 The result of my median filter processing.(a)result of amount of noise.(b)result of RMS

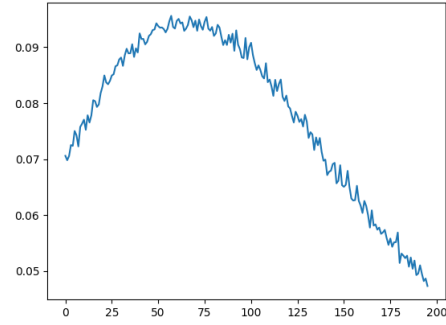
It can be seen from Fig. 67(a) that the amount of noise drops rapidly in the first few cycles, and when it reaches about 38%, there is no more change. It can be seen from Fig. 67(b) that, similar to Fig.55(a), the RMS drops rapidly at the beginning of the cycle, and then no longer changes. I guess the reason is that the median filtering method itself does not generate new values, but only finds intermediate values to replace the original multiple values. When the amount of noise is large, the effect is not ideal.

In the same strategy, compared with the traditional median filtering method, the results obtained by my median filtering method are basically the same. This shows that it is probably because when the amount of noise is large, the effect of median filtering itself is limited.

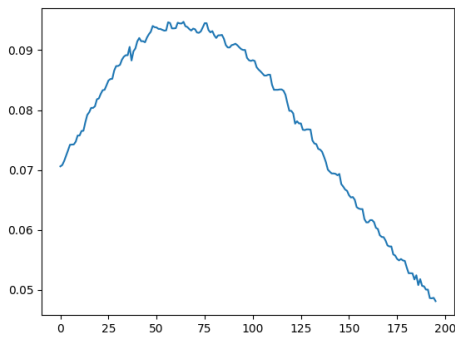
Next, the cross-sectional data will be displayed, which can more intuitively reflect the quality of the smoothing effect(Fig. 68):



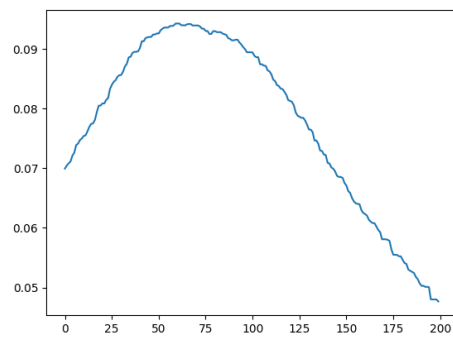
(a)



(b)



(c)

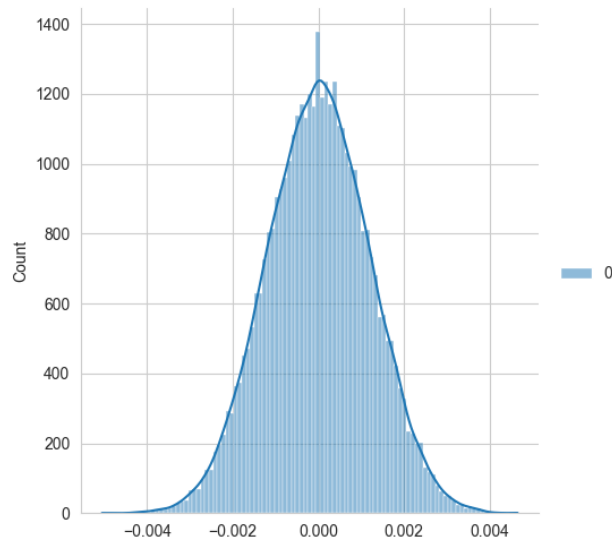


(d)

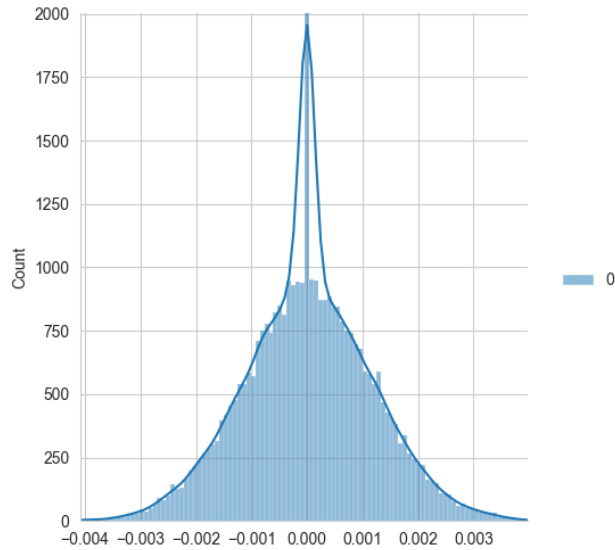
Fig. 68 Cross-sectional data of the 100% noise model in line 100.(a)Original model(without noise).(b)Noise model.(c) Model after my median filtering(1 time).(d) Model after my median filtering(20 times).

From Fig. 68(c), it can be seen that even though some noises can be smoothed by using my median filter method to smooth once, the data is not smooth enough. Fig. 68(d) compared with (c), intuitively the data is smoother, but it is still not smooth enough. It can be seen that using my smoothing noise will still fall into a local optimal solution, and then the effect will no longer improve, indicating that the median filtering method itself does have defects.

Next, I will show the noise distribution inversely deduced by my median filtering strategy processing(Fig. 69):



(a)

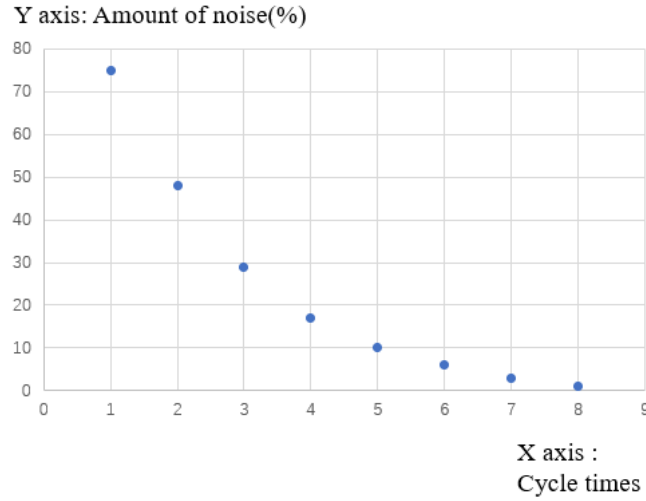


(b)

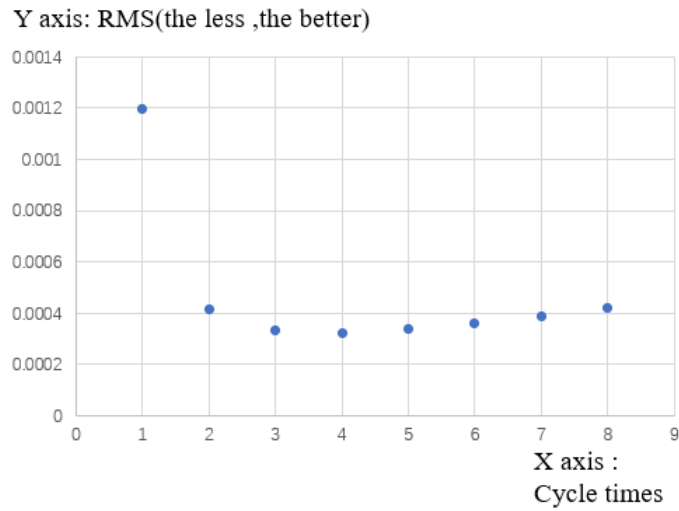
Fig. 69 Noise Distribution.(a) Artificial noise distribution.(b) The noise distribution derived from the result of multiple smoothing through the strategy

It can be seen from Fig. 69(b) that the result obtained by the traditional median filter method is almost identical. Using the result obtained by my median filter method, the noise distribution graph obtained by inversely deduces the point where the value is equal to 0. The amount is very large, in line with the characteristics of median filtering, but also shows that the effect of median filtering is not ideal.

Next, I will apply the traditional Gaussian filtering method to the strategy, and the results are as follows(Fig. 70):



(a)



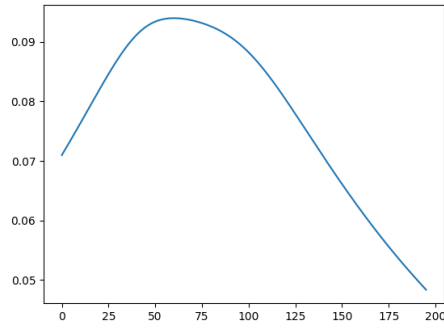
(b)

Fig. 70 The result of traditional Gaussian filter processing.(a)result of amount of noise.(b)result of RMS

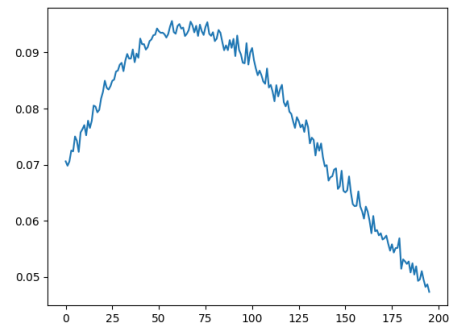
It can be seen from Fig. 70(a) that the traditional Gaussian filtering method only used 8 times to reduce the amount of noise to 1%. It can be seen that the Gaussian filtering method has a very good effect, and each filtering can effectively reduce the noise.

It can be seen from Fig. 70(b) that the RMS value obtained after the traditional Gaussian filtering method processes the data first decreases and then increases. I think this shows that using traditional Gaussian filtering multiple times will cause the InSAR image to be over-smooth. In other words, InSAR images have lost some feature points and textures.

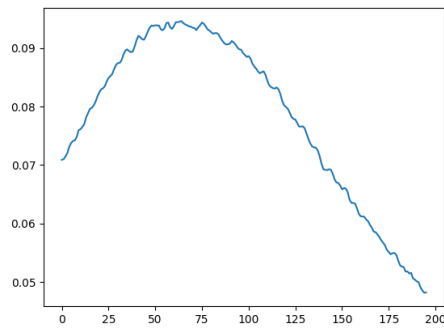
Next, the cross-sectional data will be displayed, which can more intuitively reflect the quality of the smoothing effect(Fig. 71):



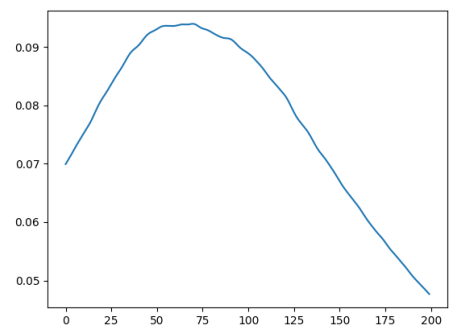
(a)



(b)



(c)

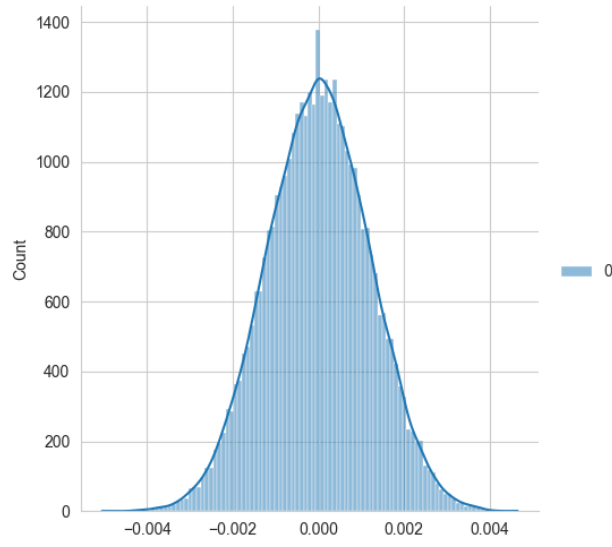


(d)

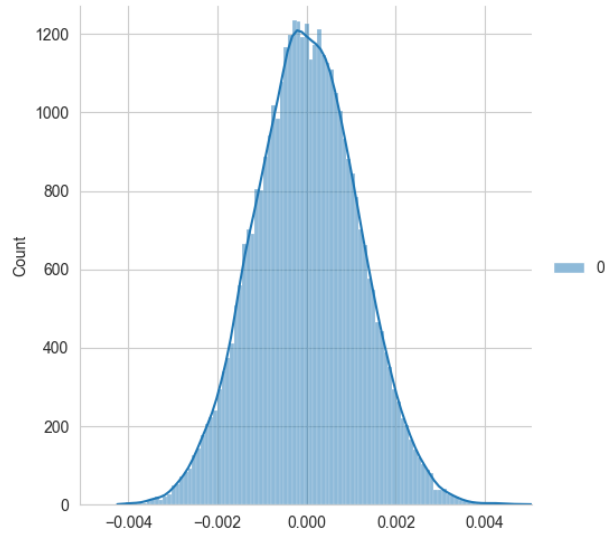
Fig. 71 Cross-sectional data of the 100% noise model in line 100.(a)Original model(without noise).(b)Noise model.(c) Model after traditional Gaussian filtering(1 time).(d) Model after traditional Gaussian filtering(8 times).

It can be seen from Fig. 71(c) that only once use of traditional Gaussian filtering can smooth some noise. Compared with the result of traditional median filtering, the effect is better, but it is not smooth enough. It can be seen from Fig. 71(d) that, by making the strategy, the result of traditional Gaussian filtering is very smooth, almost exactly the same as the original model, so it can be considered that Gaussian filtering is very effective in smoothing noise. But there will be over-smooth situations.

Next, I will show the noise distribution inversely deduced by traditional Gaussian filtering strategy processing(Fig. 72):



(a)

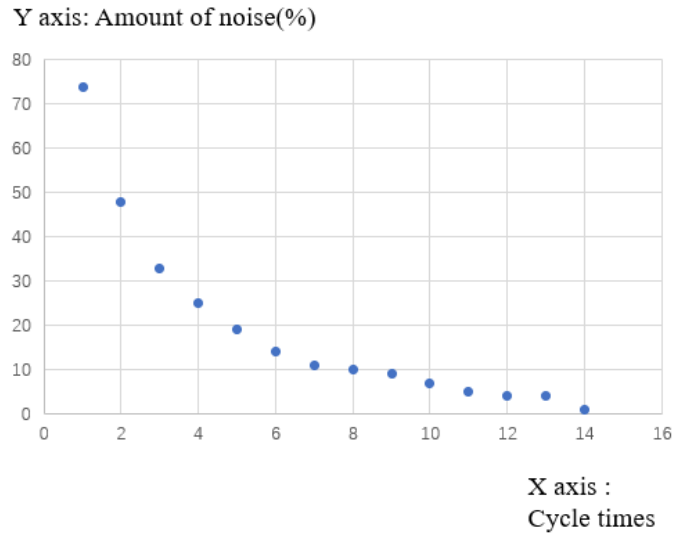


(b)

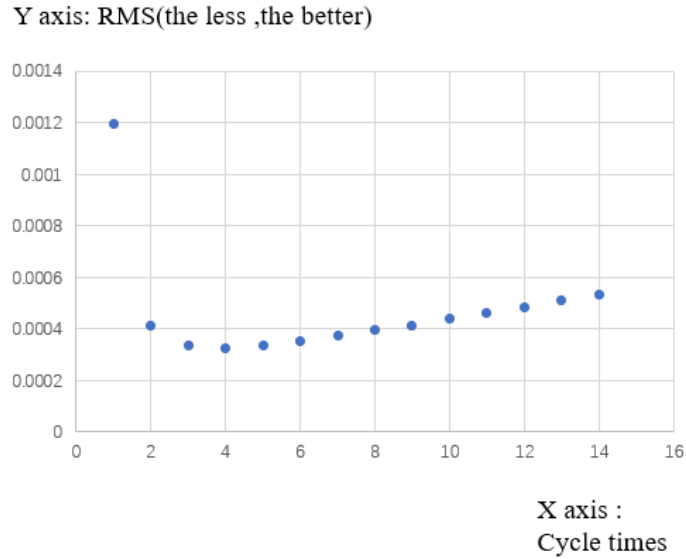
Fig. 72 Noise Distribution.(a) Artificial noise distribution.(b) The noise distribution derived from the result of multiple smoothing through the strategy

Comparing Fig. 72(a) and Fig. 72(b), it can be found that the shapes of the two noise distributions are almost the same, that is, the shape of the normal distribution, which shows that the noise distribution corrected by the traditional Gaussian filter is almost completely in line with the artificial input. Noise distribution. But it can also be found that the shape of Fig. 72(b) is thinner than the shape of Fig. 72(a), and the peak value is also lower, indicating that the noise removal is still not complete.

Next, I will apply the my Gaussian filtering method to the strategy, and the results are as follows(Fig. 73):



(a)



(b)

Fig. 73 The result of my Gaussian filter processing.(a)result of amount of noise.(b)result of RMS
The results obtained by my Gaussian filtering method are similar to those obtained by the traditional Gaussian filtering method.

It can be seen from Fig. 73(a) that the traditional Gaussian filtering method only used 14 times to reduce the amount of noise to 1%. It can be seen that the Gaussian filtering method has a very good effect, and each filtering can effectively reduce the noise.

It can be seen from Fig. 73(b) that the RMS value obtained after my Gaussian filtering method processes the data first decreases and then increases. I think this shows that using traditional Gaussian filtering multiple times will cause the InSAR image to be over-smooth. In other words, InSAR images have lost some feature points and textures.

The reason why my Gaussian filtering method needs to be used 14 times, while the traditional Gaussian filtering method only needs to be used 8 times is that my method can not find all the noise, only about 90% to 99% of the noise, so it needs more Smooth noise.

It can be seen that whether it is the traditional Gaussian filtering method or my Gaussian filtering method, there will be an over-smooth situation, indicating that this is a common defect of the Gaussian filtering method.

Next, the cross-sectional data will be displayed, which can more intuitively reflect the quality of the smoothing effect(Fig. 74):

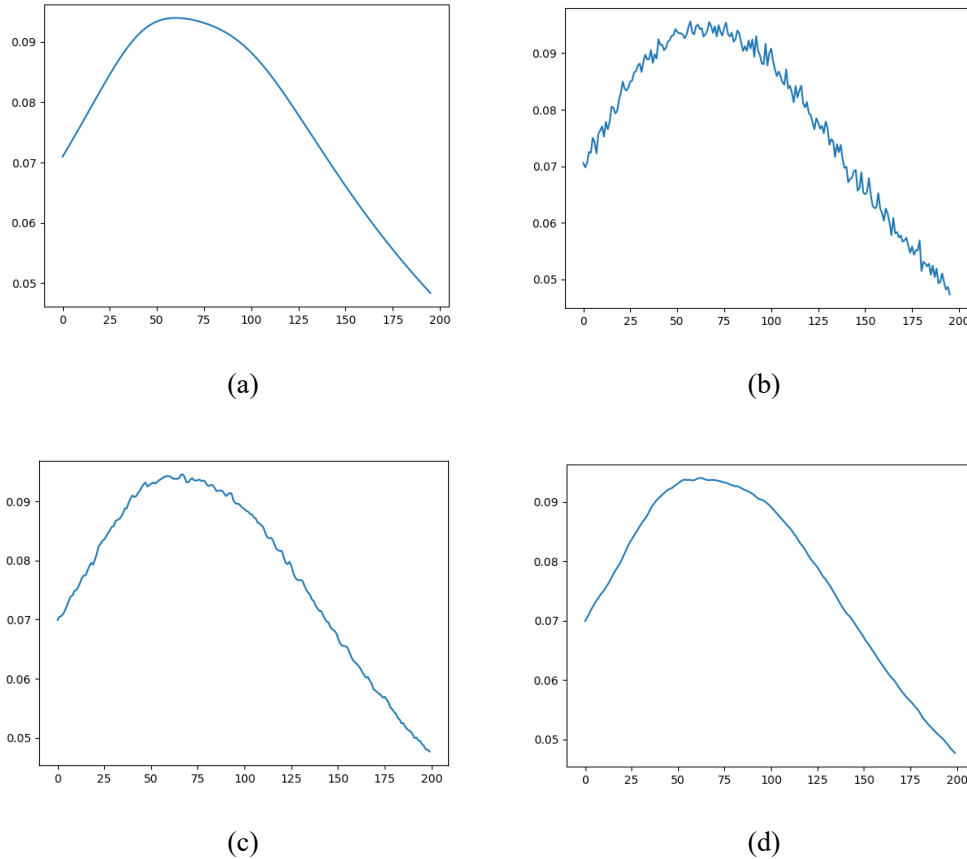
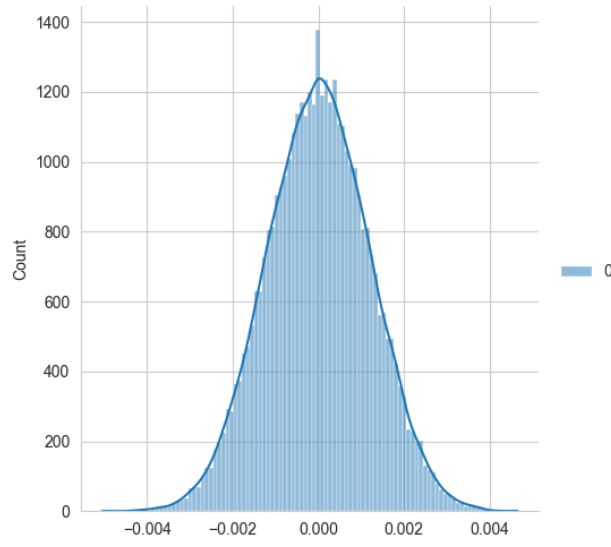


Fig. 74 Cross-sectional data of the 100% noise model in line 100.(a)Original model(without noise).(b)Noise model.(c) Model after my Gaussian filtering(1 time).(d) Model after my Gaussian filtering(14 times).

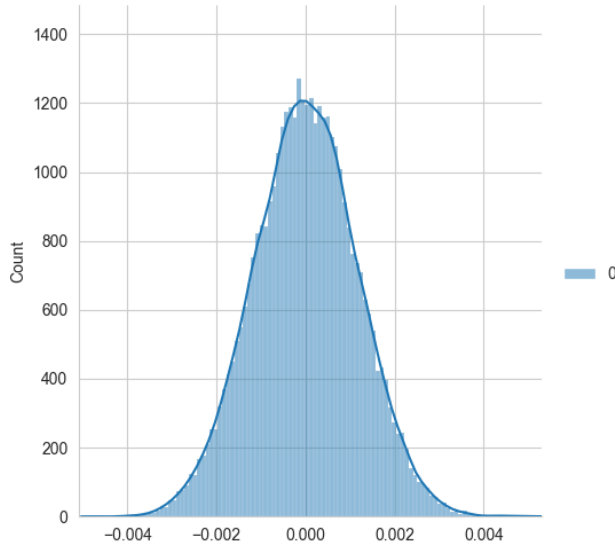
It can be found that the results obtained by using my Gaussian filtering method and the traditional Gaussian filtering method are similar.

It can be seen from Fig. 74(c) that only once use of my Gaussian filtering can smooth some noise. Compared with the result of my median filtering, the effect is better, but it is not smooth enough. It can be seen from Fig. 74(d) that, by making the strategy, the result of my Gaussian filtering is very smooth, almost exactly the same as the original model, so it can be considered that Gaussian filtering is very effective in smoothing noise. But there will be over-smooth situations.

Next, I will show the noise distribution inversely deduced by my Gaussian filtering strategy processing(Fig. 75):



(a)



(b)

Fig. 75 Noise Distribution.(a) Artificial noise distribution.(b) The noise distribution derived from the result of multiple smoothing through the strategy

It can be seen that the noise distribution obtained by my Gaussian filtering method is almost identical to the noise distribution artificially put in by me. At the same time, comparing the noise distribution obtained by my Gaussian filtering method with the noise distribution obtained by the traditional Gaussian filtering method, the shape of the noise distribution obtained by my method is more similar to the original noise distribution. From this perspective, I think my Gaussian filtering method is better.

5.8.The result of applying methods and strategies in InSAR of study area

First I will show how the amount of noise changes, as shown below(Fig. 76):

Y axis: Amount of noise(%)

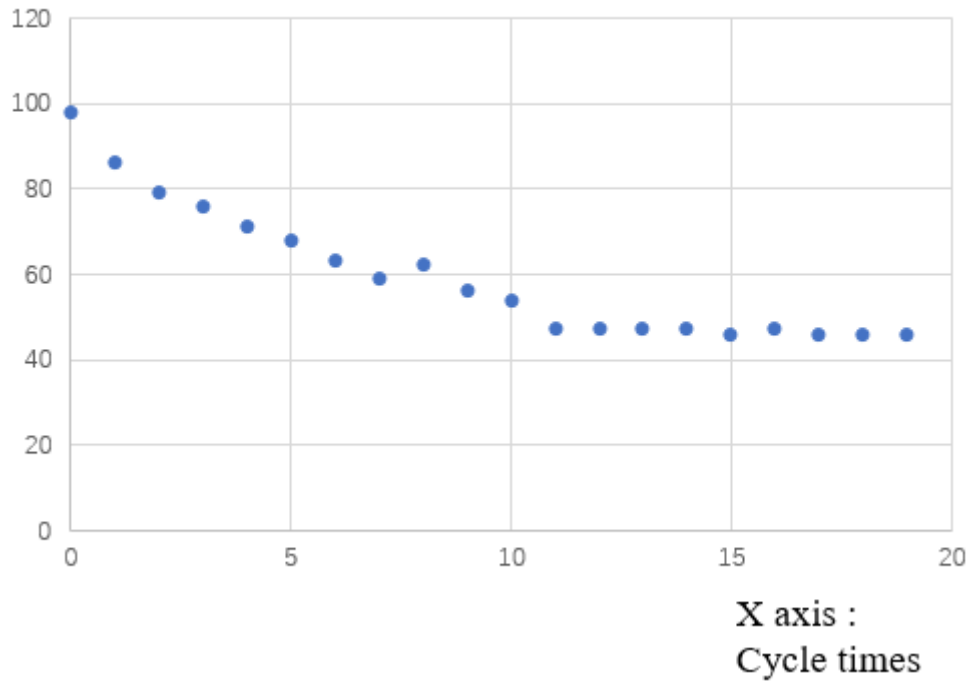


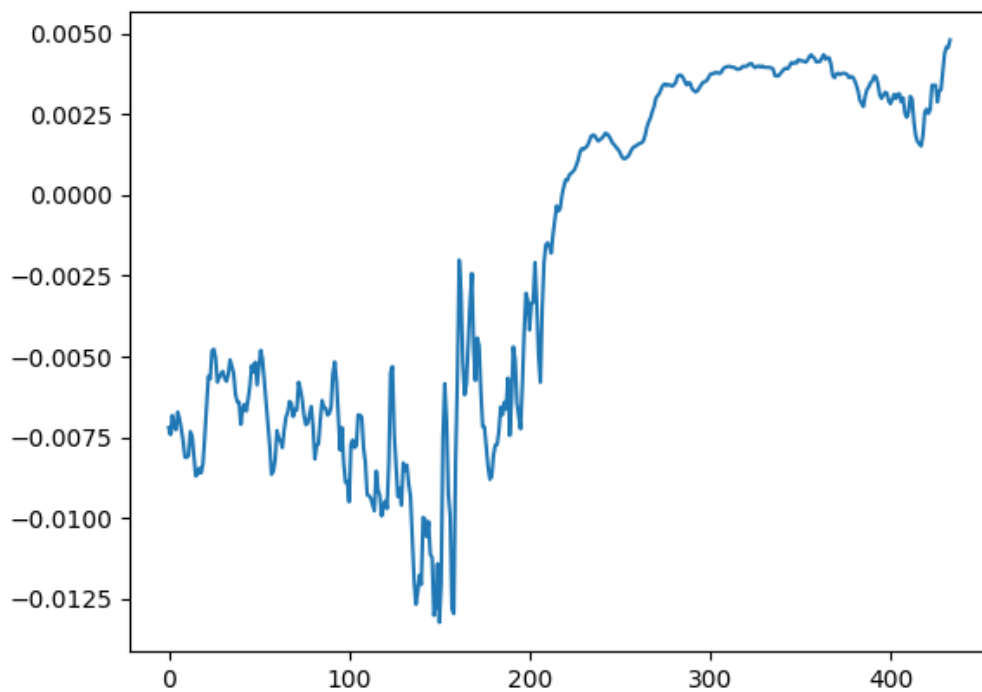
Fig. 76 The change of amount of noise

It can be seen that the amount of noise first steadily drops to about 46%, and then does not change. In virtual verification, using normally distributed noise can reduce the amount of noise to 1%. However, there are limitations in the actual situation. I think there are several possible reasons:

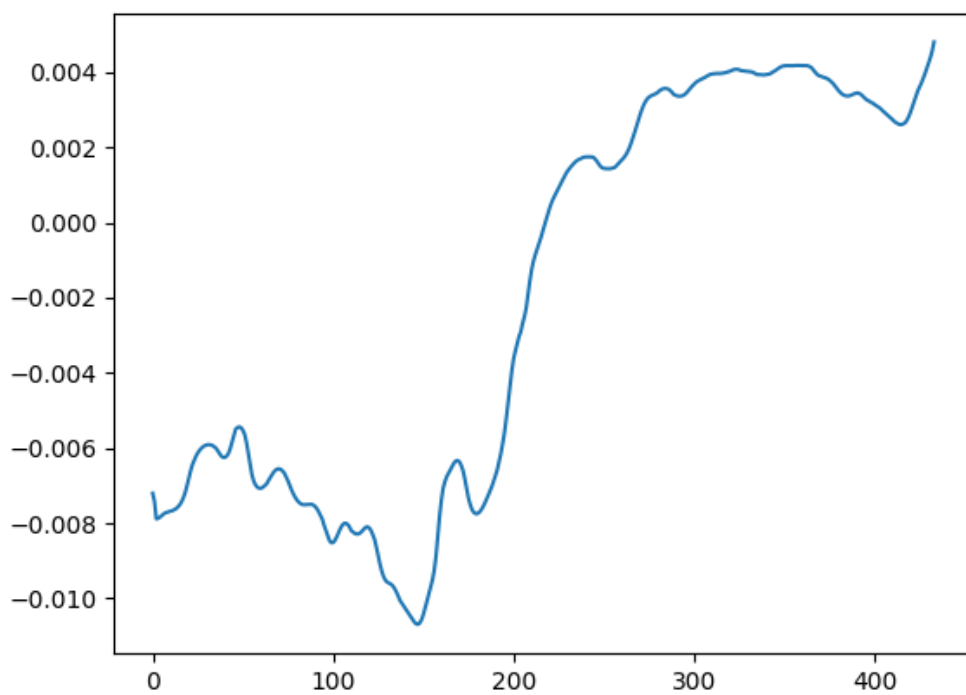
1. The method of inferring the amount of noise is limited. The method of inferring the amount of noise has a certain relationship with the standard deviation of the noise. In actual situations, we cannot know the standard deviation and distribution of noise, so this method has limitations.

2. The smoothing method is not effective enough. The smoothing methods I use in virtual verification are median filtering and Gaussian filtering. In practical applications, I use Gaussian filtering. I think that when the distribution and composition of noise are very complicated, Gaussian filtering alone is not enough to achieve the desired effect.

Next, the cross-sectional data will be displayed. At this time, the cross-sectional data in row 100 is still selected(Fig. 77):



(a)



(b)

Fig. 77 Cross-sectional data of the study area's InSAR model in line 100.(a) Study area's InSAR model.(b) Model after my Gaussian filtering(20 times)

From the cross-sectional data, the model after smoothing is relatively smooth, and my method can be considered effective, but I cannot verify whether it has been smoothed. Combined with the change in the amount of noise (Fig. 76), the reason why the amount of noise no longer decreases can be considered to be that my method of inferring the amount of noise is limited, because the smoothing method is effective.

Next, the noise distribution will be shown(Fig. 78):

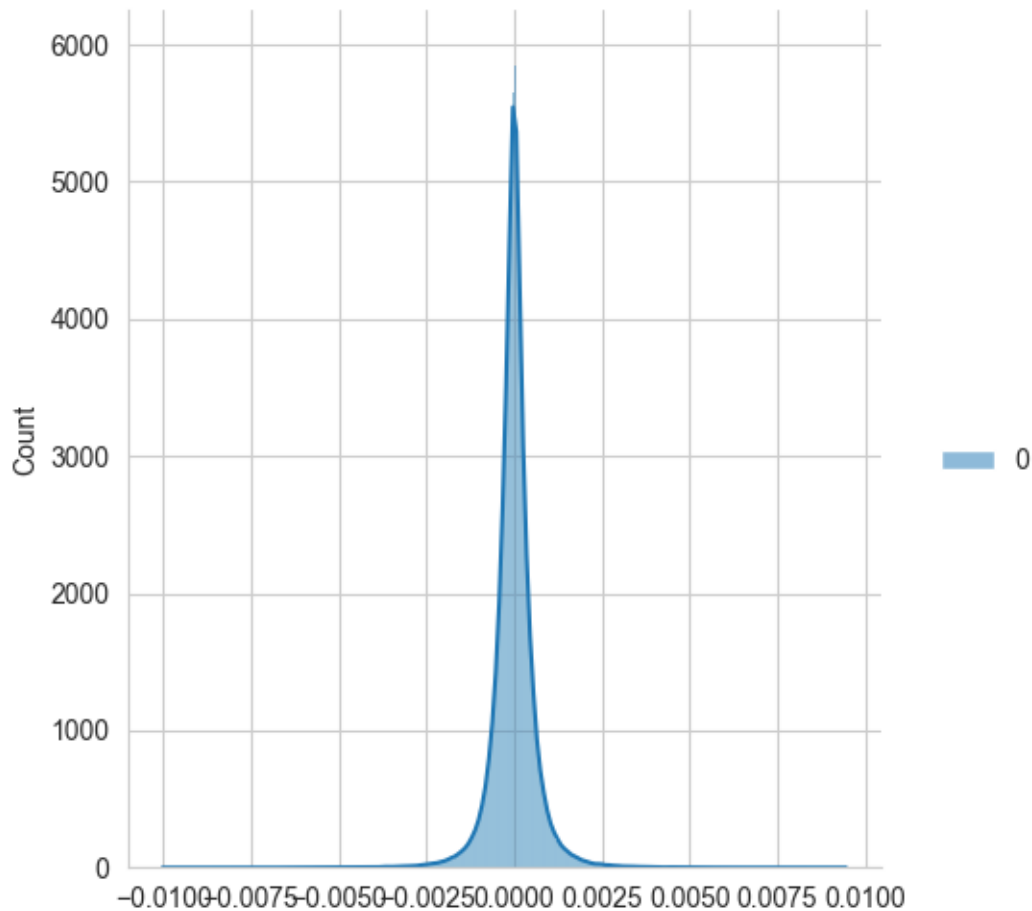


Fig. 78 The noise distribution derived from the result of multiple smoothing through the strategy

It can be seen that the distribution of noise is very similar to the normal distribution, and they are both symmetric about $y=0$. But the shape of this noise distribution map is very thin and tall, indicating that the value of the standard deviation is very limited. The calculation shows that the standard deviation is 0.000629. The standard deviation of the noise model I used in virtual verification is 0.00024. I think this may be the reason for the abnormal number of noise distributions.

6. Additional discussion

6.1.About the accuracy of the method of deriving the amount of noise

For the accuracy of inferring the amount of noise, I will use several experiments to verify.

6.1.1. Change the scope of noise

The first experiment: changing the scope of noise delivery(Fig. 79):

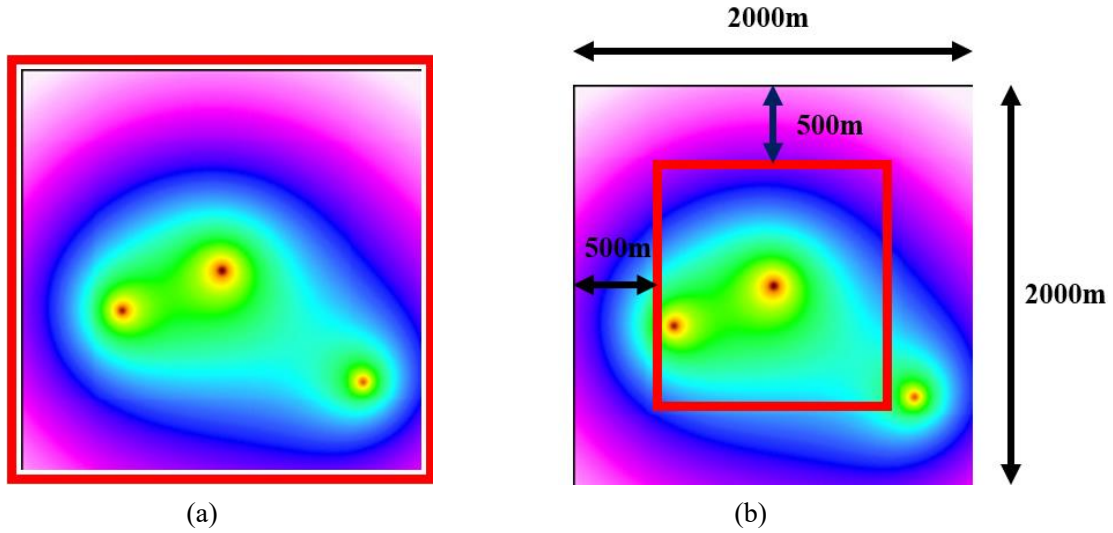


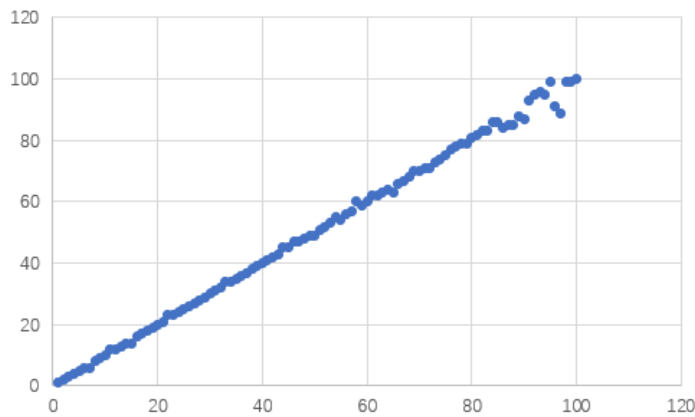
Fig. 79 The range of noise points.(a) Put noise on the entire InSAR image range.(b) Put noise on a part of the InSAR image

The red box represents the range of noise. Fig. 79(a) represents the range of noise in the entire InSAR image. In the virtual verification, I used this range to get the result. Fig. 79(b) indicates that noise is injected in a part of the middle of the InSAR, and the area of the injection range is about one-fourth of the entire InSAR area.

So in terms of the proportion of the number of injections, for the entire InSAR image, the number of noises ranges from 1% to 25%, which is mapped to the noise injection area, and the range is 4% to 100%.

The result is as follows(Fig. 80):

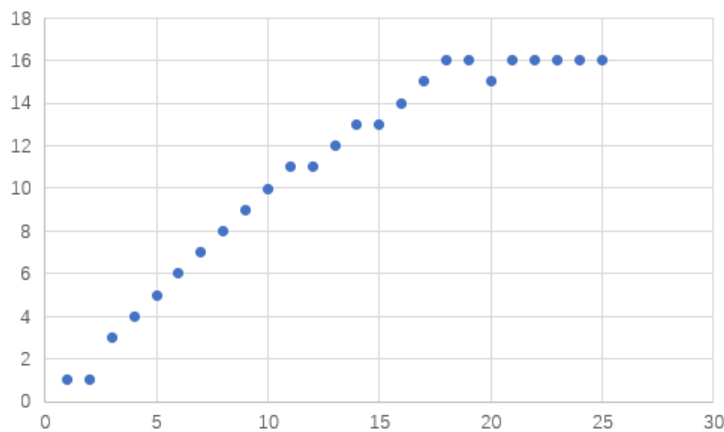
Y axis: Amount of
noise I infer



X axis: Amount of
noise

(a)

Y axis: Amount of
noise I infer



X axis: Amount of
noise

(b)

Fig. 80 The accuracy of deriving the amount of noise.(a)The result of entire InSAR range.(b)The result of part of InSAR range.

It can be found that the inference result of putting only part of the range of noise is similar to the result in virtual verification. When the amount of noise is greater than 80% (for Fig. 80(b), this should be 20%), the inference result is biased. When the amount of noise is less than 80%, the inference result is more accurate.

6.1.2. Change the standard deviation of noise

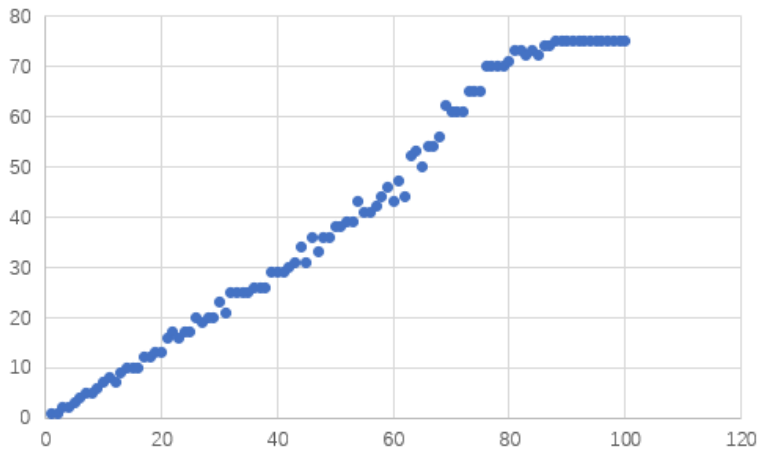
In the second experiment, change the standard deviation of noise. In the virtual verification, I used the noise standard deviation of 0.005, and then I changed the standard deviation to the following two numbers to verify:

1.standard deviation=0.001

2.standard deviation=0.01

The result is as follows(Fig. 81):

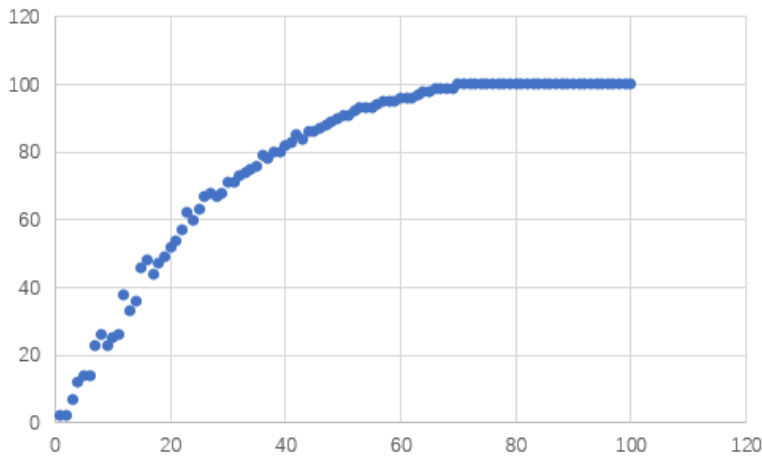
Y axis: Amount of
noise I infer



X axis: Amount of
noise

(a)

Y axis: Amount of
noise I infer



X axis: Amount of
noise

(b)

Fig. 81 The accuracy of deriving the amount of noise.(a)The result of standard deviation=0.001.(b) The result of standard deviation=0.01.

It can be seen that changing the standard deviation will change the accuracy of the inference result. When the value of the standard deviation is increased, the slope of the accuracy equation of the inference result will increase; when the value of the standard deviation is decreased, the slope of the accuracy equation of the inference result will decrease. .

This shows that the standard deviation is directly related to the accuracy of the inference method. If the standard deviation of the noise can be obtained, the results can be improved.

6.2.About the magic number “60%”

In the results obtained in the above multiple chapters, I found that the number 60% appears very frequently, and it often means a turning point in the trend of the data. Next, I will change the number of wells in the digital model. In other words, use different digital models to see if it affects the 60%

figure.

First use the digital model of a single well:

Number of wells: 1

The coordinates of the well: (1000,1000)

Smoothing method: traditional Gaussian filter, traditional median filter, my Gaussian filter, my median filter

Evaluation method: RMS

The digital model and the results are as follows (Fig. 82 and Fig. 83):

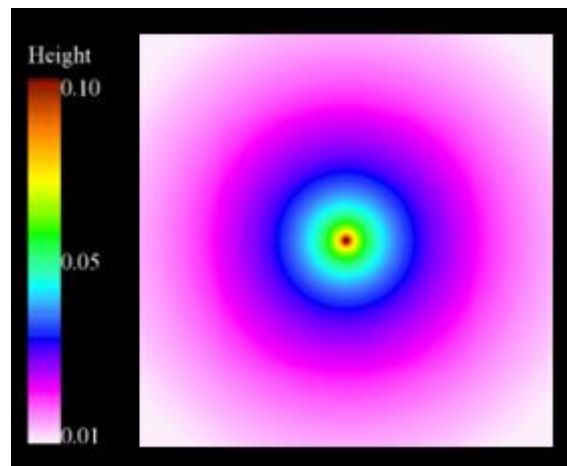
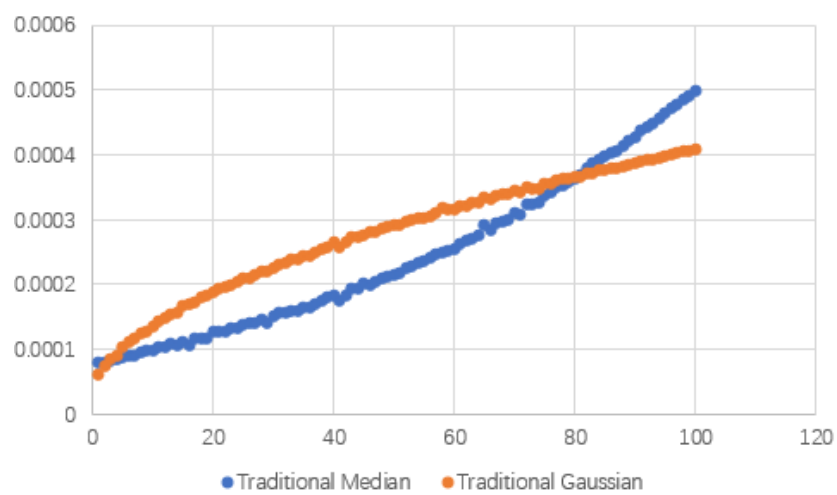


Fig. 82 Numeric InSAR image of single well

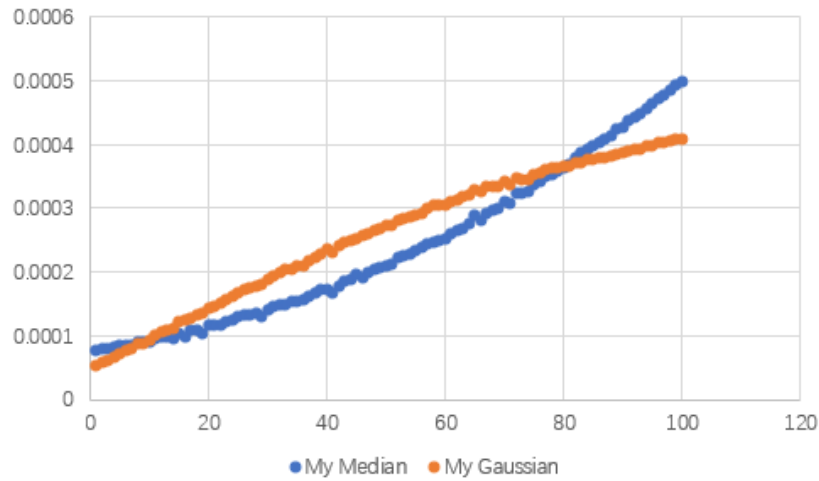
Y axis: RMS(the less ,the better)



X axis:
Amount of
noise

(a)

Y axis: RMS(the less ,the better)



X axis:
Amount of
noise

(b)

Fig. 83 RMS graph after processing by 4 methods.(a)RMS Comparison chart of traditional methods.(b)RMS Comparison chart of my method

It can be found that whether it is the comparison of the traditional method or the comparison of my method, the position of the inflection point has changed, no longer 60%, but about 80%. This shows that the position of the inflection point is indeed related to the number of wells.

Then use the digital model of 2 wells:

Number of wells: 2

The coordinates of the well: (500, 800) ,(1714, 444)

Smoothing method: traditional Gaussian filter, traditional median filter, my Gaussian filter, my median filter

Evaluation method: RMS

The digital model is as follows(Fig. 84):

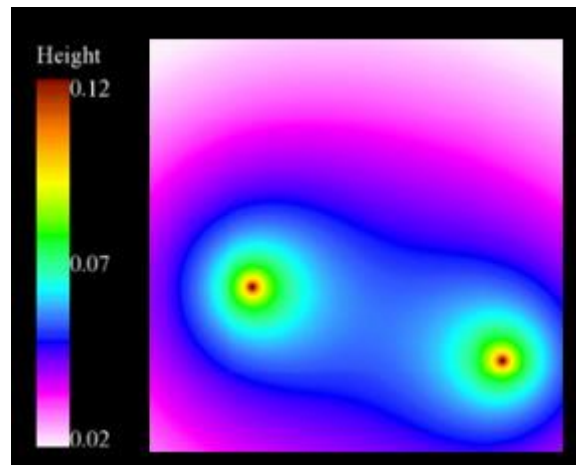
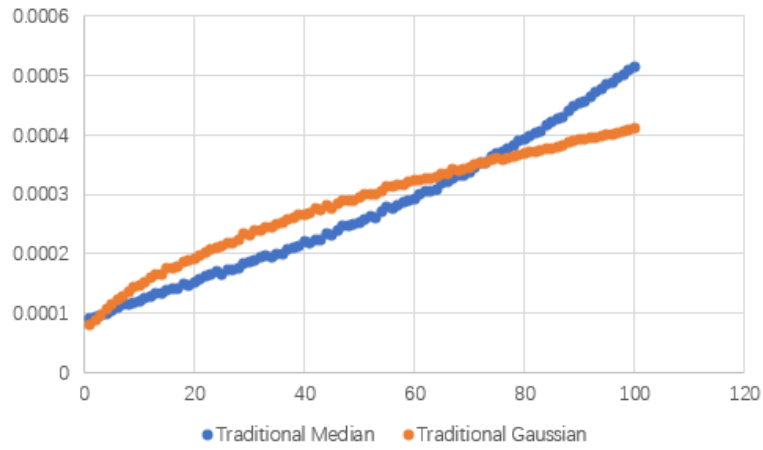


Fig. 84 Numeric InSAR image of 2 wells

The result is as follows(Fig. 85)

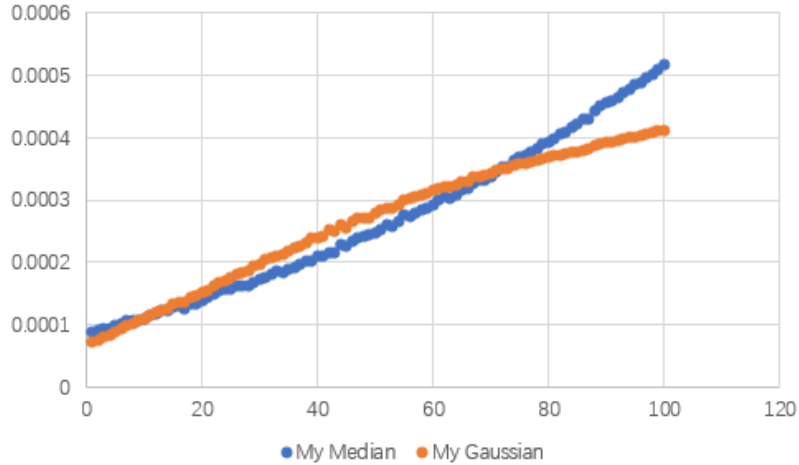
Y axis: RMS(the less ,the better)



X axis:
Amount of
noise

(a)

Y axis: RMS(the less ,the better)



X axis:
Amount of
noise

(b)

Fig. 85 RMS graph after processing by 4 methods.(a)RMS Comparison chart of traditional methods.(b)RMS Comparison chart of my method

Similar to the numerical model of a single well, whether it is a comparison of the traditional method or the comparison of my method, the position of the inflection point has changed, not 60%, but about 70%. This shows that the position of the inflection point is indeed related to the number of wells. And

with the increase in the number of wells, the position of the hanging point is decreasing, from 80% to 60%.

7. Conclusions

This research uses Method 1 (Chapter 4.1) and Method 2 (Chapter 4.2) to identify the noise of InSAR images and find the location and distribution of the noise. In section 4.3, the best coefficient values of method 1 and method 2 under different noise levels are found through looping and comparison, and a data list is obtained. In section 4.4, through the data list obtained in section 4.3, a method to infer the amount of noise in the current InSAR image is summarized, and the results show that the accuracy of this inference method is relatively accurate. In Chapter 4.5, formulate a smoothing strategy through the method of Chapter 4.4, and smooth the noise multiple times by judging the current amount of noise.

Based on the results and discussion, the following conclusions can be drawn:

1. The combination of Method 1 and Method 2 of this study can identify most of the noise and find the distribution and location of about 90% of the noise. The accuracy will be affected by the amount of noise.

Its advantage is that the traditional smoothing method cannot identify the noise and find the location of the noise, but combined with my method, it can smooth the noise more accurately, improve the accuracy of the smoothing, and reduce the possibility of over-smoothing.

Its disadvantage is that this method cannot find and identify all noises. When the amount of noise approaches 100%, its efficiency is not as good as the traditional smoothing method.

2. When the amount of noise is large, one smoothing is not enough to smooth all the noises, so my method of inferring the amount of noise provides the possibility and a basis for judgment for multiple smoothing. But this judgment method is closely related to Method 1 and Method 2, and it is affected by the noise standard deviation. Therefore, how to calculate the noise standard deviation of the actual InSAR image is a problem and direction to be studied.

3. Regarding the smoothness of noise, this research did not make essential improvements to the traditional methods, but only combined the traditional methods with methods 1 and 2. So I think the method of smoothing noise is worthy of improvement and research.

4. The most important innovation of this research is to identify noise through my method, find the location and distribution of the noise, and then infer the amount of noise. This method provides the possibility and judgment basis for the strategy of multiple smoothing noise. However, the accuracy of the method in this study is affected by the noise standard deviation, so I think this content can be studied in the next step.

8. Acknowledgments

I am grateful to Masatsu Aichi for guiding my research direction, providing me with research data and research equipment, and providing valuable research suggestions as my supervisor. I am grateful to Yoshito Oshima for providing valuable advice for my research as my sub-supervisor. I am grateful to Akitaya Kento for providing research suggestions as a senior in the laboratory.

I am grateful to all the friends who have helped me.

9. Reference

- Aichi M, Rokugawa S. Decomposition of land surface displacement observed by InSAR into land subsidence caused by groundwater abstraction and natural motion of crust[C]//EGU General Assembly Conference Abstracts. 2017: 11383.
- Aimaiti Y, Yamazaki F, Liu W. Multi-sensor InSAR analysis of progressive land subsidence over the Coastal City of Urayasu, Japan[J]. *Remote Sensing*, 2018, 10(8): 1304.
- Ansari H, De Zan F, Bamler R. Sequential estimator: Toward efficient InSAR time series analysis[J]. *IEEE Transactions on Geoscience and Remote Sensing*, 2017, 55(10): 5637-5652.
- Armas, I.; Mendes, D.A.; Popa, R.G.; Gheorghe, M.; Popovici, D. Long-term ground deformation patterns of Bucharest using multi-temporal InSAR and multivariate dynamic analyses: A possible transpressional system? *Sci. Rep.* 2017, 7, 43762
- Cianflone, G.; Tolomei, C.; Brunori, C.A.; Dominici, R. InSAR time series analysis of natural and anthropogenic coastal plain subsidence: The case of sibili (Southern Italy). *Remote Sens.* 2015, 7, 16004–16023.
- Deguchi, T., Rokugawa, S. & Matsushima, J. (2009) Long-term ground deformation measurement by time series analysis for SAR interferometry. *J. Remote Sensing Soc. Japan* 29, 418–428 (in Japanese with English abstract).
- Deming, D. (2002) *Introduction to Hydrogeology*. McGraw-Hill, New York, USA.
- Environmental and Community Affairs Department, Chiba Prefecture (1970–2007) Levelling results in Chiba Prefecture [reference date: February 1970–January 1st, 2007], (in Japanese).
- Environmental and Community Affairs Department, Chiba Prefecture (2007) State of land subsidence in Chiba Prefecture (in Japanese).
- Ferretti, A.; Prati, C.; Rocca, F. Nonlinear Subsidence Rate Estimation Using Permanent Scatterers in Differential SAR Interferometry. *IEEE Trans. Geosci. Remote Sens.* 2000, 38, 2202–2212
- Hayashi, T., Tokunaga, T., Aichi, M., Shimada, J., & Taniguchi, M. (2009). Effects of human activities and urbanization on groundwater environments: An example from the aquifer system of Tokyo and the surrounding area. *Science of The Total Environment*, 407(9), 3165–3172.
- Hsieh, C.S.; Shih, T.Y.; Hu, J.C.; Tung, H.; Huang, M.H.; Angelier, J. Using differential SAR interferometry to map land subsidence: A case study in the Pingtung Plain of SW Taiwan. *Nat. Hazards* 2011, 58, 1311–1332
- J. S. Lee, K. P. Papathanassiou, T. L. Ainsworth, M. H. Grunes and A. Reigber, "A new technique for noise filtering of SAR interferometric phase images", *IEEE Trans. Geosci. Remote Sens.*, vol. 36, no. 5, pp. 1456-1465, Sep. 1998.
- Kanopoulos, N, Vasanthavada, et al. Design of an image edge detection filter using the Sobel operator[J]. *Solid-State Circuits, IEEE Journal of*, 1988.
- M. S. Seymour and I. G. Cumming, "Maximum likelihood estimation for SAR interferometry", *Proc. IGARSS*, pp. 2272-2275, 1994-Aug.
- Meng D, Sethu V, Ambikairajah E, et al. A novel technique for noise reduction in InSAR images[J]. *IEEE Geoscience and Remote Sensing Letters*, 2007, 4(2): 226-230.
- MURAI D, NAKAMURA M, IKEDA S, et al. Land subsidence observation using GPS on the Kujukuri Plain[J]. *IAHS-AISH publication*, 2010, 339: 377-379.

- NAKAMURA T. Land subsidence at the Kujukuri Plain in Chiba Prefecture, Japan: evaluation and monitoring environmental impacts[J]. 2010.
- R. Goldstein and C. Werner, "Radar ice motion interferometry", Proc. 3rd ERS Symp. Space Service Environ., pp. 969-972, 1997
- Raspini, F.; Loupasakis, C.; Rozos, D.; Adam, N.; Moretti, S. Ground subsidence phenomena in the Delta municipality region (Northern Greece): Geotechnical modeling and validation with Persistent Scatterer Interferometry. Int. J. Appl. Earth Obs. Geoinf. 2014, 28, 78–89.
- Sato, C., Haga, M., & Nishino, J. (2006). International Review for Environmental Strategies Special Feature on Groundwater Management and Policy Land Subsidence and Groundwater Management in Tokyo. 6(2), 403–424.
- 平成 28 年地盤沈下調査報告書 - 東京都建設局
(<https://www.kensetsu.metro.tokyo.lg.jp/content/000028631.pdf>)