Double Interpolation in GIS Tasks

Tarasov Andrey, Potapova Valentina
Ryazan State Radioengineering University
Ryazan, Russian Federation
vb2005@yandex.ru, valentina2008.91@mail.ru

Abstract— **In this paper considers the development of a project module "Mobile GIS", providing the possibility of building surfaces and contours on the basis of a set of information about the altitude in certain points. The variants of the acceleration of the operation of the double interpolation algorithm for the processing of heterogeneously distributed data are considered. The application of various interpolation functions of two variables is considered and the quality of recovery of the result is analyzed.**

Keywords-component; GIS; Interpolation; Approximation; Point cloud; Maps

I. INTRODUCTION

When composing thematic maps, one often has to deal with the problem of comparing to a point cloud A_n of a function $z = F_A(x, y)$. Point cloud is a set of data obtained in the course of measuring works, represented as a three-dimensional point $A_i = (x_i\ y_i\ z_i)$, where x_i and y_i – are the coordinates in which the measurement is obtained, z_i – is the numerical value of the measured parameter. In most cases, we can say that the function is piecewise and can hardly be described mathematically for an adequate time.

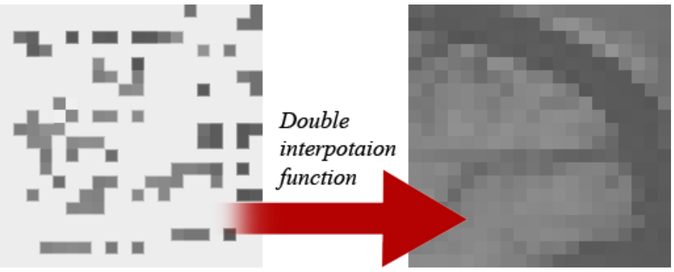

Figure 1. Task of double interpolation.

In this case, the optimal solution is to find not the function itself, but the set of values required for analysis. The method of finding intermediate values of a quantity from the available discrete set of known values is usually called interpolation [1]. Double interpolation is a subset of the interpolation problem, where the value of the intermediate value is calculated for a function of two variables. In the future, this article is about this.

Double interpolation can be used to obtain a set of values of any parameters (weather data, geological structure of the soil, etc.). Of this article, we will talk about finding heights based on LiDAR survey, barometric measurements and SRTM data [2].

II. GRID STEP AND REGION SIZE

Before proceeding with the construction of matrices, it is necessary to select the regions and the grid step, indicating the degree of detail of the data obtained.

The grid spacing is selected in such a way as to provide maximum informative value, while avoiding the calculation of an excessive number of values. The boundaries of the value calculation area are selected in such a way that sufficient data is available near the points to calculate an adequate value.

The selection of interpolation region boundaries should be approached very attentively. If it is chosen too small, then not all places on the map will be able to calculate the values. If it is too large, then they will have a very low degree of adequacy.

In the course of the experiment, the following dependence of the SD on the distance to the nearest points in which it is possible to calculate the value was found. It should be noted that the SD values are very conditional, and depend on the behavior of the function on the plane. However, the general form of the dependence corresponds to the following figure:

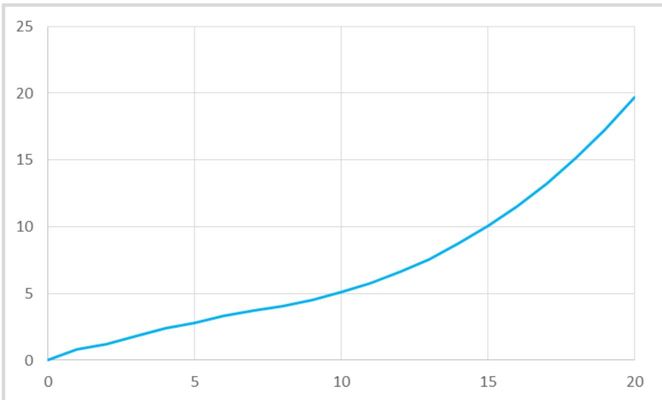

Figure 2. SD dependence on distance to pixel.

III. NEAREST POINTS SEARCH ALGORITHMS

Based on double interpolation for each point of the matrix of heights, the value is calculated as follows:

1. For the selected point, there are n-nearest values.

ISSN 2337-0343

2. Filtering and sorting based on the selected values.
3. The interpolation function is applied to the remaining values.

The selection of the nearest values to a point can be based on the search of all existing points. However, the use of this option is inexpedient, because when working with a large number of points the processing time multiplies [3]. As a solution to this problem, two variants of searching for neighboring values were considered.

A. Spiral search

If all the existing values already represent nodes in the final matrix, then a variant of spiral search is allowed. The search algorithm is as follows:

- For a selected point, the neighborhood of 3x3 is analyzed for the presence of initial values in them.
- In the case that a sufficient number of values is collected, the algorithm stops its work, otherwise the next neighborhood is analyzed – 5x5, and excluding the neighborhood 3x3 from it (it turns around the perimeter of the square).
- This algorithm is executed until a sufficient number of points n is collected, or the boundary of the search area is reached.

This algorithm allows an average of 100 times faster to find neighboring values in the matrix, but it is only applicable if the coordinates of the initial cloud of points are discrete and already represent nodes of the finite matrix.

B. Search based on square-divisioning

This option is one of the ways of indexing values in the matrix. The assignment of the index to the square is performed in the same way as in Figure 3.

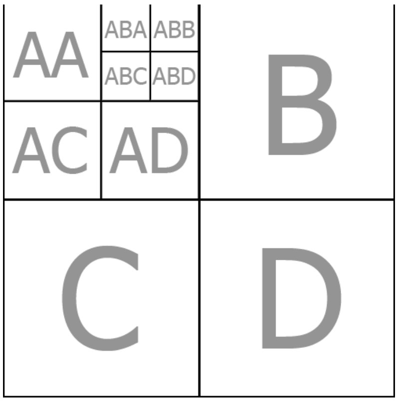

Figure 3. Split the matrix into squares.

The whole image area is divided into 4 squares conventionally called A, B, C and D. We will assume that all points with coordinates x smaller than the center of the image and the coordinates of y smaller than the center of the image belong to the square A. Similarly for B, C, D. For each square, the index in the sub-square is defined.

For each element from the cloud of points, its index is based on the location of the point. For example, for the point located in the upper right corner, the "AAAA" index will be assigned. The degree of fragmentation of the matrix (the number of letters) is chosen in such a way that each square has an average of no more than 10 points. This partitioning makes it possible to find the neighboring elements most effectively.

It should be noted that the use of this method, unlike the previous one, allows storing the coordinates of a cloud of points not only in integers but also in fractional coordinates. In addition, in matrices with a very rare arrangement of elements, this method significantly improves the performance in comparison with the previous algorithm.

In case, you need to find neighboring elements, the values in the current square are analyzed. If the distances to these elements are greater than to the boundary of the square, then adjoining squares are connected to the analysis, and so on, until a sufficient number of values are collected.

To store data, the Dictionary class is used, in which the index is a string parameter, the point coordinate according to the principle described above, and the index value is a set of points corresponding to a given square [4]. There can be several points in the square, and not even one.

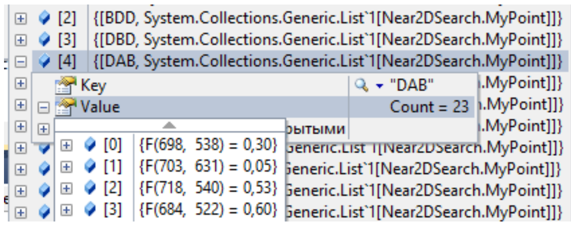

Figure 4. A dictionary with squares, and the first 3 points for a square "DAB".

IV. Double Interpolation Formula

When solving the problem of constructing a matrix of values from a cloud of points, the choice of the distance function is an important factor. It establishes a relationship between the intensity of the color and the distance to the pixel:

$$W = \sum_{i=0}^{n} w_i * F_l(l_i) \,/\, \sum_{i=0}^{n} F_l(l_i).$$

Here w_i – color value in the point, l_i – the distance from the determined point to the i point, $F(l_i)$ – the distance function, n – count of used points (n nearest points).

The form of the distance function was selected manually based on the change in the SD in various cases. The most reliable function is $F_l(l) = l^{-4}$. It should be noted that for these data (Weather maps, maps of deposits, etc.), the function may be different.

V. QUALITY CONTROL

In order to evaluate the algorithm efficiency, a series of SRTM images was selected. Some number of pixels (in %) was removed from the images, after which they were executed by the algorithm of double interpolation. The resulting image was compared to the original one based on the SD characteristic.

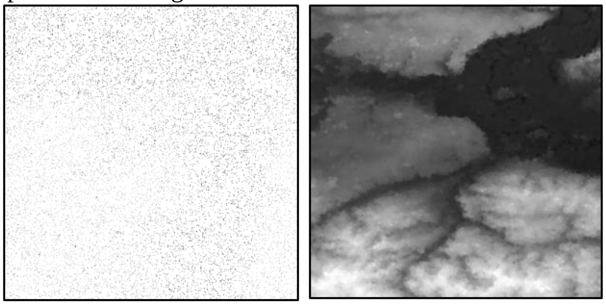

Figure 5. Recovered image from 5% of pixels. Used double interpolation. SD = 6.2.

In order to test the ability to interpolate both with a negligibly small amount of data, and at a high, the following values were chosen: 0.01; 0.1; 0.5; 1; 2; 5; 20; 50; 75; 90; 99. For each of them, 1000 experiments were performed on different sets of images. The following results were obtained:

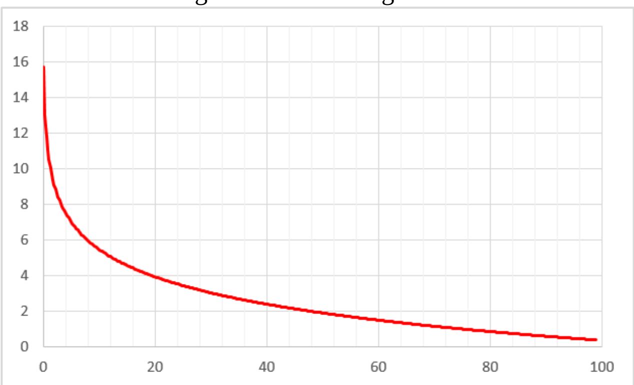

Figure 6. SD dependence on number of pixels (in %).

It should be noted that the result depends on the number of selected points for analysis. Too small a quantity will result in the values being too rough, too - to the fact that the result will be very blurred, and many times will increase the calculation time.

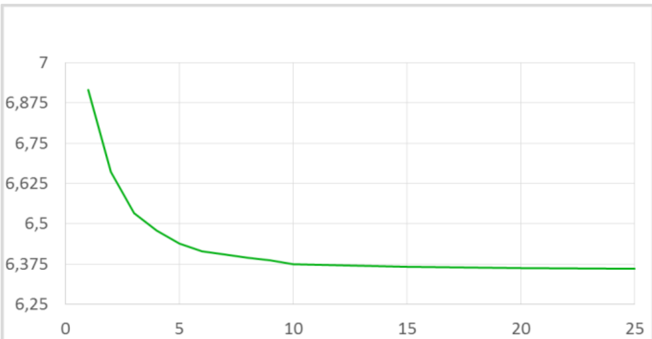

Figure 7. SD dependence on n (number of selected points).

VI. USE IN OTHER AREAS

In addition to GIS tasks, this algorithm can be successfully used in data compression and recovery tasks.

A. Recovering of damaged image

For example, in case of damage to m percent of points on the sensor, this algorithm will allow getting an image close to an adequate one, even though the losses are significant.

Figure 8a – Original Image. Figure 8b – recovered image from 20% of original data (1.6 bits per pixel, SD = 6.2).

B. Image compression

The double interpolation algorithm can be applied to image compression problems. For example, if you save 5% of the pixels from the image, and then transfer them through the channel, the picture size is reduced by 20 times in compared to the original image [5]. When it is restored to the final device, it is possible to achieve high accuracy. It has been experimentally established that such a recovery algorithm gives 30% better results than orthogonal transformation based on Walsh function:

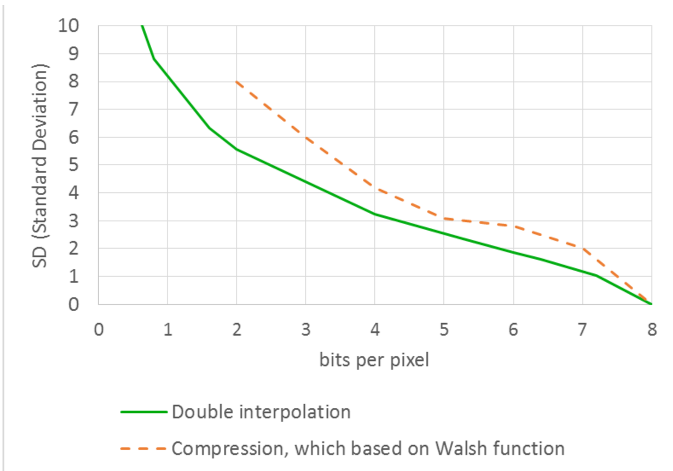

Figure 9. SD dependence of bits per pixel.

VII. CONCLUSION

As it was demonstrated in the article, the application of double interpolation algorithms can help in solving problems of finding values, as well as in image compression and reconstruction problems. The results obtained indicate low error rates. The use of algorithms for finding the closest elements and distance functions made it possible to significantly shorten the time of obtaining the result and improve its accuracy.

REFERENCES

[1] Sakhnovich L.A. Interpolation Theory and Its Applications, Germany, 1997.

[2] Tarasov A., Potapova V., Svirina A. Mobile geoinformation system development 6th Mediterranean Conference o Embedded Computing, Montenegro, 2016.

[3] Kormen T.H. Introduction to Algorithms Text. 2th ed. Moscow, Williams, 2005, 1296 p. (Russ. ed. Krasinikov I.V., Kormen T.H.)

[4] Albahari J., Albahari B. C# 5.0 in a Nutshell, Sebastopol, USA, 2012. pp 282-320.

[5] Kostrov B.V., Grinchenko N.N., Gerashenko E.S., Potapova V. Yu., Tarasov A.S., Tokarev A.V. MODELING OF THE TRANSMISSION CHANNEL OF AEROSPACE IMAGES USING A CONCATENATED CODE Izvyestia Tulskogo Gosudarsvennogo Universiteta. Tekhnicheskie nauki [Tidings of TulGU. Technical Sciences]. Tula, 2006, pp.124-129 (In Russian).