

Terrain objects edge detection in noisy GPS images

Natalya N. Grinchenko^{1,a}, Aleksey I. Baranchikov¹ and Anton V. Tokarev¹

¹Ryazan state radio engineering university, Electronic computing machines department, 390005 Ryazan, Russia

Abstract. This paper presents the algorithm of terrain objects edge detection in GPS images. This objective is relevant for two reasons. Firstly, in case of distortions occurring in the process of data transmission through communication channel, and secondly, in many cases preliminary data processing requires edge detection. To solve the first issue we utilize quasi-2D filtration, and to solve the second we use an algorithm based on Canny edge detection algorithm.

1 Introduction

Nowadays the industry of Earth remote sensing (ERS) devices grows very rapidly. As a consequence, the amount of GPS information increased as well as the number of tasks connected with processing of such information. Apart from filtration, visual augmentation and restoration of missing areas, there are analytical tasks such as evaluation of different parameters (probabilistic characteristics, position and shape of objects etc.), object detection and identification etc. A lot of analytical tasks require preprocessing of images including edge detection.

Since this task is uncommon and multistep it causes certain problems. The most wide spread problem is the noise in images. In such case edge detection may give invalid results spoiled by vague edges or gaps. It often occurs during terrain (different terrain objects such as lakes, rivers, mountains, buildings etc.) observation.

As long as most of edge detection methods employ gradient methods, the distortions will negatively affect the objects detection and identification on GPS images. To solve the task effectively it is necessary to eliminate the described distortions, which may be completed by utilizing quasi-two-dimensional (Q2D) filtration methods. Since group and single distortions has a fairly predictable structure, they are clearly seen in spectral representations of the image.

2 Theoretical grounds

Any raster image is a matrix, the monolithic atomic unit of which is a pixel. Edge detection is mostly influenced by group noises, which can occur in two forms: the noise that affects the whole line or the noise that affects only a part of the line, which corresponds to the model of a noise that distorts a line in an image with a

certain probability p (Figure 1). The following distortion process may be described as follows (1)[1]:

$$g(x,y) = \begin{cases} f(x,y), & \text{if } R(x,y) = 0, \\ 0, & \text{if } R(x,y) = 1. \end{cases} \quad (1)$$

Where $R(x,y)$ describes distortion spot depending on values p and p_l , where p – probability of noise in the line and p_l – random value equal to noise length.

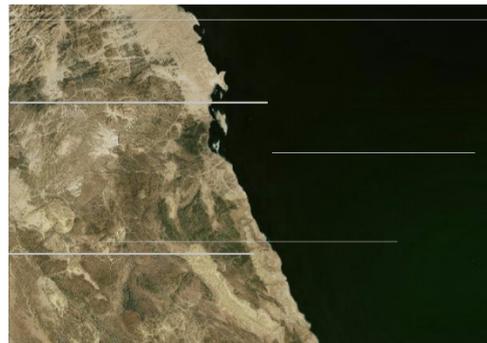


Figure 1. Types of Group noises.

Quasi-two-dimensional filtration method is as follows. Direct Walsh transform is performed according to the formula (2):

$$F = H_W g, \quad (2)$$

where H_W – Walsh-Adamar matrix; g – Image matrix.

Investigation of quasi-two-dimensional spectrum of group noises on a smooth background is likely to reveal sufficient runouts, connected with the presence of noise. On this basis we may conclude that the following run outs influence quasi-two-dimensional spectrum as well.

The elements of the spectrum may be divided into distorted and not distorted. The division is based on noise

^a Corresponding author: grinchenko_nn@mail.ru

runouts in quasi-two-dimensional spectrum. The algorithm checks the relation of the central element to certain local environment (3):

$$\begin{cases} 0, \text{ if } |F(u-1, y) - F(u, y)| \leq s, \\ 0, \text{ if } |F(u+1, y) - F(u, y)| \leq s, \\ 1, \text{ in any other case.} \end{cases} \quad (3)$$

where $M(u)$ – line of number of lines of possible location of a group noise, S – some threshold.

Threshold is selected for the whole spectrum but it can be adjusted for different images. Then you may use the following approach: interpolation of spectrum lines related to the noise with the nearest undistorted values of the spectrum. (4)[2]:

$$\hat{F}(u, y) = \begin{cases} F(u, y), u \notin M \\ \frac{F(u-1, y) + F(u+1, y)}{2}, u \in M \end{cases}, \quad (4)$$

Reverse Walsh transform (5):

$$I = H_w \hat{F}, \quad (5)$$

where I – restored image.

The second type of noises is the impulsive noise or dotty noises («salt and pepper» Figure 2). Impulsive noises may be described in the following manner: if the value in point $f(i, j) = 0$, it corresponds to black noise, if it is $f(i, j) = 255$ – then it is white noise (6). Therefore

$$g(x, y) = \begin{cases} f(x, y) \text{ if } R(x, y) = 0, \\ b, \text{ if } R(x, y) = 1, \end{cases} \quad (6)$$

where $R(x, y)$ describes noise spot, b – noise value in the spot, which can be only 0 or 255.

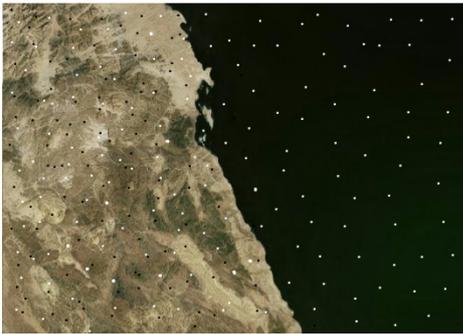


Figure 2. Dotty noises.

The most effective method for eliminating this noise is applying median filter (dimensionality of 3×3 [3, 4, 5]) to the noisy image (7).

$$f(x, y) = \underset{k=1, n}{\text{med}} \{g(x, y, k)\}, \quad (7)$$

where $\hat{f}(x, y)$ – processed image, $g(x, y, k)$ – filter mask with dimension of 3×3 pixels.

In general terms the quasi-two-dimensional algorithm consists of the following steps [6]:

1. Applying of non trigonometric spectrum restrain theorem to quasi-two-dimensional spectrum;
2. Applying of median theorem (8) to the obtained image:

$$\hat{f}(x, y) = \underset{k=1, n}{\text{med}} \{g(x, y, k)\}, \quad (8)$$

3. Subtraction of the obtained image from the image with impulse noises (9):

$$M = f - \hat{f}, \quad (9)$$

where f – image matrix, \hat{f} – new image matrix, M – impulse noise value mask;

4. Using the values of the mask, apply the formula (10):

$$b = f + M, \quad (10)$$

where b – restored image matrix.

Since size of GPS survey objects is small, edge detection must be very precise and the most appropriate algorithm that may be used here is the one proposed by John Canny. This algorithm has the following advantages

1. Fine detection
2. Fine localization
3. One response for one edge

These features allow the use of this algorithm for detecting wide range of edges with high precision.

This is a multistep algorithm that includes the following steps.

2.1 Use Sobel operator to define edge points

Sobel operator is a differential operator, which calculates gradient value in each point as well as its vector direction [2, 3].

The calculation involves germ folds of the following form:

$$G_x = \begin{bmatrix} +1 & 0 & -1 \\ +2 & 0 & -2 \\ +1 & 0 & -1 \end{bmatrix}, G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}. \quad (11)$$

With the help of the germs (11) initial image is folded for horizontal and vertical evaluation of approximate values of the derivatives (12).

$$b_{ij} = \sum_{i=1}^3 \sum_{j=1}^3 a_{ij} g_{ij}, \quad (12)$$

where b_{ij} – element of matrix B , which contains pixels of the final image; a_{ij} – element of matrix A , which contains pixels of the initial image; g_{ij} – element of matrix G .

Approximate gradient value in each point of the image is calculated as follows (13)

$$B = \sqrt{B_x^2 + B_y^2}; \quad C = \arctg\left(\frac{B_y}{B_x}\right), \quad (13)$$

where B – matrix of gradient values, C – matrix of gradient direction.

This method is very sensitive to different noises (Figure 3a,b)

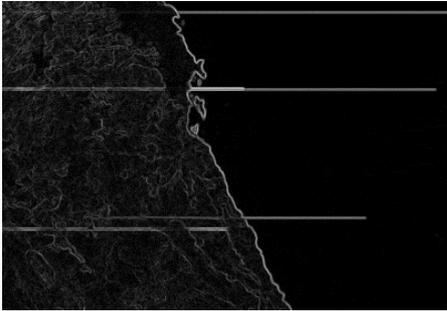


Figure 3. a) Sobel operator applied to image with group noises.

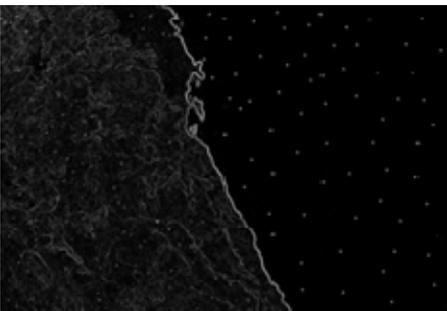


Figure 3.b) Sobel operator applied to image with dotted noises.

2.2 Operation proposed by the author

According to Canny algorithm the next step is an operation proposed by the author, which is reduction of no maximums. At this step each pixel is checked whether its gradient value is the local maximum in the gradient direction on the condition that the angle of the direction vector is rounded to the nearest multiple of 45 degrees.

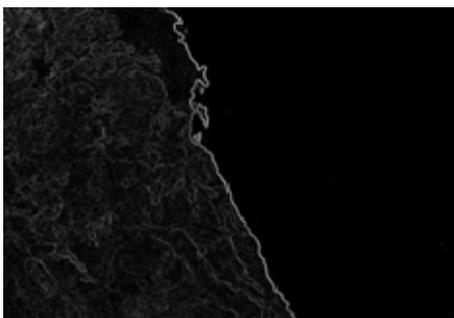


Figure 3. c) Sobel operator applied to the image after using quasi-two-dimensional filtration algorithm.

E.g. if the rounded angle of the gradient direction vector is equal to zero degrees (the edge is running from north to south), the pixel is considered to be the edge point, if its gradient value is higher than that of pixels which are to the north or to the south from it. The operation results in suppression of no maximum values, only local maximums marked as the borders and the edges become thinner and less blurry.

2.3 Threshold filtration

The next step is threshold filtration, which is carried out to define whether the edge is really situated in the chosen spot. The distinctive feature of filtration in Canny algorithm is that it uses two thresholds: if the pixel value is higher than the upper limit, it takes maximum value, if it is lower, then it is suppressed, points with values that are between the thresholds take the average value. This step is a little dubious due to the fact that if to reduce threshold the number of detected edges will increase, however, the result will be «noisier». From the other hand, high threshold may ignore weak edges, which may cause fragmented borders. As we eliminated all the noises in the beginning of the algorithm, the range of values that we consider as borders, may be wider (Figure 3c).

2.4 Highlighting groups of pixels

At the last step the task reduces to highlighting groups of pixels that gained average values at the previous step and relating them to the border or their suppression. Suppression takes place when pixel is not bound to the determined (strong) edge (Figure 4).

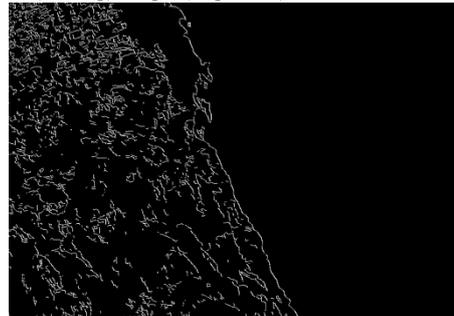


Figure 4. Final borders (edges) of terrain objects.

3 Conclusion

As the result, in images processed with the present algorithm, firstly, various distortions have been eliminated, and secondly, contours of terrain objects are sharp and thin.

References

1. N. Ahmed and K.R. Rao, *Orthogonal Transforms for Digital Signal Processing* (Springer, 1975)
2. R.C. Gonzalez and R.E. Woods, *Digital Image Processing* (Prentice Hall, 2002)
3. V.N. Ruchkin, B.V. Kostrov and E.V. Ruchkina *Communication Technologies 2015 4th Mediterranean Conference on Embedded Computing*, pp. 254-257 (2015)
4. A. Kolesenkov, B. Kostrov, E. Ruchkina and V. Ruchkin, *Proceedings 2014 3rd Mediterranean Conference on Embedded Computing, MECO 2014 - Including ECyPS 2014*, pp. 166-168 (2014)
5. E.V. Ruchkina, N.N. Grinchenko, V.N. Ruchkin, B.V. Kostrov and D.S. Stepanov *2015 4th*

Mediterranean Conference on Embedded Computing, 134-138 (2015)

6. A.N. Kolesnikov, B.V. Kostrov and V.N. Ruchkin, *Proceedings – 2013 2nd Mediterranean Conference on Embedded Computing, MECO–2013*, pp. 263–265 (2013)