IEEE IEEE Catalog Number : CFP10786-PRT ISBN : 978-1-4244-9035-6

Proceedings

The 5th International Forum on Strategic Technology
Oct. 13 ~ 15, 2010
Ulsan, Korea

IFOST

Organized By

울산대학교 UNIVERSITY OF ULSAN
울산테크노파크 Ulsan Technopark
울산광역시 ULSAN METROPOLITAN CITY

Vector Processing Contours

Andrey Kirsanov
Department of Automation and Control Processes
MAMI Moscow State Technical University
Moscow, Russia
AndKirsanov@yandex.ru

A.Vavilin and K-H. Jo
Department of Electrical Engineering
University of Ulsan
Ulsan, Korea
{andy|jkh2010}@islab.ulsan.ac.kr

***Abstract*— process of object recognition in satellite images of high resolution is a complex task associated with a large work on time and complexity of the operator. This paper describes an innovative approach for solving this problem.**

Based on monochromatic high-resolution satellite images (in the process of using data from the QuickBird satellite with a maximum resolution of 0.6 meters per pixel) geo data bitmap and vectorized output are received (shape files). The principle of object recognition in a satellite image is based on the allocation of edges in the gradient transition using a threshold filter. Obtained data is then transformed to a vector output using straight line detection and connected components analysis. The proposed method allows to process satellite images of large size with high performance. Performance of the proposed method could be improved by using GPU-based computations.

Keywords-filter; vector data; 3D maps; GIS;

I. INTRODUCTION

One of the challenging problems in aero and space geodesy is the building contour identification for the transfer of these contours on a map and further transfer cards in 3D-image. Reconstruction of three-dimensional models of buildings and objects on the map is a necessary solution to some problems, such as: GIS, a database for flight simulator, analysis and solution of critical situations prevailing in the city, the study the earth's surface layout of urban areas and cities development. Vectorization of satellite images is a laborious and time-consuming task which requires the presence of the human operator. The amount of work depends directly on the time of the operator. Also, in addition to the process of recognition of buildings from satellite images needed vectorized third-party sites, such as: trees, roads, walls, fences, power lines, railroad tracks. As a result, must recognize not only the geometric shapes and colors, too (for example recognition of trees). However, the main task is set in the drawing outlines of such objects as buildings, the difficulty lies in the complexity of the geometric shape of the object and the number of buildings. To create a complete database containing not only the three-dimensional objects are necessary input data to the grid heights of land, geo binding, virtual texture. In this paper we have listed the data were not considered because at this time there are many methods of obtaining this information, and methods of treatment. Following date is necessary for building a 3D scene:

- Elevation Data - for constructing 3D models of the Earth's surface
- Imagery Data - to create a virtual textures
- Map Data - Identification of geodetic data
- Culture Data - to build 3D models of objects

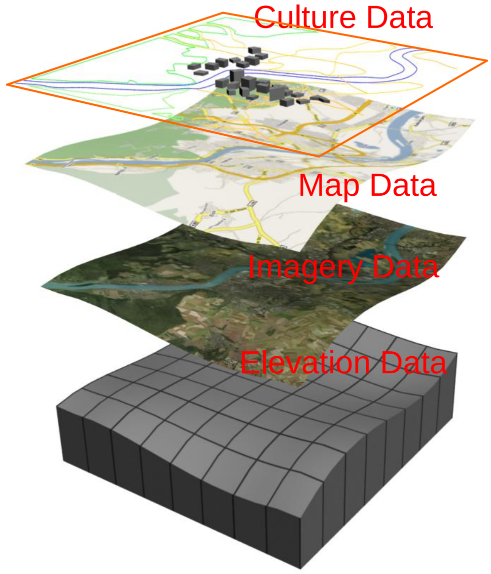

Figure 1. The necessary data to create 3D maps

The main task in the 3D reconstruction is the vectorization of raster images for Culture Data. Currently, there are a number of ready solutions provided by GIS companies, such as: ESRI, Presagis, MultiGen, etc. Most of the software does not use post-processing of satellite images and use only the clustering of data by color [1], [2]. However, this method does not allow handling bitmaps with sufficient accuracy and high performance. When clustering the color does not identify the objects. This makes it impossible to identify the type of vector objects: buildings, roads, fences, trees. As a result of this problem to be addressed to the human operator. The solution to this problem is the preparation of satellite imagery to 3D reconstruction by Threshold Filter.

II. SOBEL FILTER AND THRESHOLD FILTER

A. The Algorithm

In this work is considered the problem of 3D map reconstruction from raster image. The system is designed with the following algorithms:

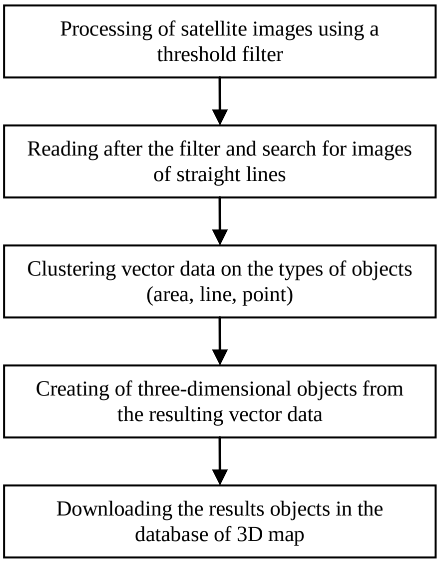

Figure 2. The flowchart of the automatic building detection

The proposed method is based on grayscale image processing. However, it could be extended to use color information as an additional segmentation criterion.

The first part is allocated bitmap contour image. Algorithms such programs are known. Traditionally, effective, in particular filters Sobel (Sobel) and Scharra (H. Scharr) [1]. The disadvantage of these filters is a significant computational time and noise resulting image. Therefore, in this paper, we propose a threshold filter, revealing the contours of grayscale images thickness of not more than one pixel. Filter algorithm as follows. Let each pixel of the image is encoded number from 0 to 1. The black color corresponds to 0, white - 1. In the first phase the whole image is replaced by the maximum possible contrast to some threshold h. Pixels with a value smaller than h are replaced with black, large - white. Next by horizontal scan images and identifies the left and right boundaries of solid black areas. The internal parts of these regions are replaced with white borders remain black. The same scanning and vertically, then the received images are blended (multiplies of pixel values of 0 and 1).

Figure 3. The necessary data to create 3D maps

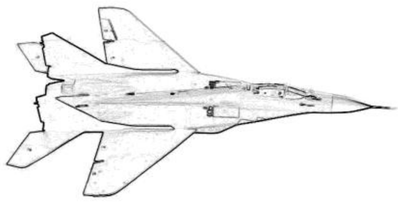

Figure 4. Sobel Filter

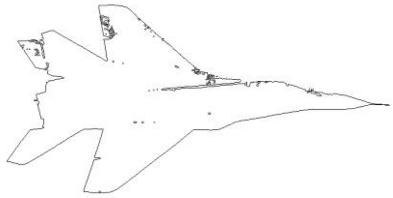

Figure 5. Threshold Filter

A comparison of the Sobel filter and proposed filter. Processing images in Figure 3 Sobel filter gives an image containing halftone and sharp contours (Fig. 3). Figure 4 shows the result of data processing threshold filter. Obviously, for tracing the second result is preferable. In addition to image quality in such problems plays an important role during recognition. In some cases, such as the problem of orientation, recognition process must have high performance, such as in tracking fast moving objects. This is particularly sensitive to fast moving objects (location planes, missiles and. Etc.). Graph comparing the speed of filters to identify the contours (Fig. 6) indicates that an increase in image size advantage of the proposed filter, the threshold increases. The calculations were performed in the Maple 11, used the operators of package “ImageTools”.

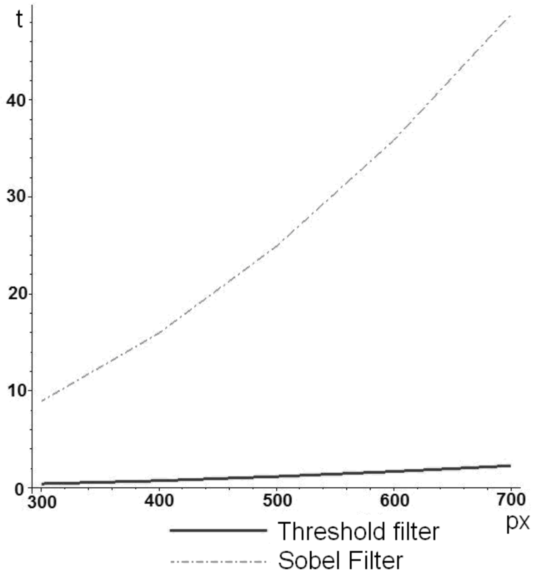

Figure 6. Demonstration of performance of the filter Sobel and threshold filter

It is noticed that the result of the threshold filter depends on the choice of values *h*. Depending on the brightness and contrast of the image, this value was varied around the average value of 0.5 and was chosen manually or as the average value $h = \sum_{i=1}^{N} \mu_i / N$, where μ_i - pixel value, $N = n \cdot m$ - the number of pixels the image size $n \times m$.

Another approach - building density curve (Figure 7). The graph on the *x*-axis - the value of a pixel (from black 0 to white 100), the *y*-axis - the number of pixels. The threshold is taken as the expectation:

$$h = \sum_{i=1} \mu_i k_i / K \qquad (1)$$

$$K = \int_0^1 k(\mu) d\mu \qquad (2)$$

In fact, *K* - area of the curve of intensity distribution of pixels. In the latter case as a dedicated circuit is obtained above. To speed up calculations in the calculation of the threshold scan was carried out not on all points of the image, but with some step, the threshold in this case has changed slightly.

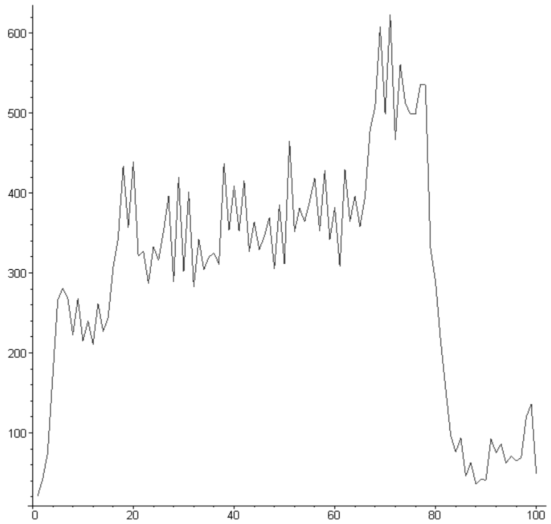

Figure 7. Graph the curve distribution density threshold

The abscissa shows the intensity of pixels of the minimum (black) to maximum (white) in the percentage content.

As a result we get a preview for the subsequent vectorization, an example can be considered for the resulting data:

Figure 8. Input Image

Figure 9. The image after processing filter

III. AUTOMATIC GENERATION OF VECTOR DATA FROM RASTER

After preparing the raster image using a threshold filter, the resulting image is converted into vector data and scanned image of vertical stripes of width k_h pixels. In each line consisting of black pixels marked the beginning and end of line. An indication of the beginning m_{begin} of pixels taken in a horizontal layer of height to 1 pixel, a sign of the end - m_{end} of white pixels in the layer height n_{end} . In the simplest case, the image is scanned stripe in two pixels, $m_{begin} = 1$, $m_{end} = 2$, $n_{end} = 1$. The obtained coordinates of start and end of the vertical segment are the output of the program, according to which the corresponding image. As an example, the common language postscript. Similarly, the horizontal scan using the same parameters are allocated horizontal segments of images. Example of tracing the vertical and horizontal scan is shown in Figure 8. Improve the quality of the image allows oblique scan. In the simplest case of selected slope in $\pm\pi/4$, it's especially useful on a rectangular grid of pixels. Each layer consists of a

k_h pixel width of the form - $P_{i+k,j+k}$, where $0 \le k \le k_h$ the scanning angle $+\pi/4$ and the pixel type $P_{i+k,j-k}$ of scan angle $-\pi/4$. The choice of the orthogonal grid defined object - houses and city buildings in aerial photographs are generally rectangular in shape.

IV. EXPERIMENTAL RESULTS

Experience with the proposed converter confirms its speed and quality. The disadvantage of the converter include difficulty arising from the recognition of small objects round. To enhance the quality will increase the frequency of scanning angles, the optimization of parameters k_h, m_{begin}, m_{end}, n_{end}. Another way to improve the quality - use multithreshold filter, which is constructed path is not only black, but gray field images. For more accuracy work can be applied multithreshold filter, but experience shows that in this case, firstly, the choice of thresholds is difficult, and secondly the result of increased noise and the traditional Sobel filter is preferable, despite its low speed. After processing the input image filters was analyzed two outcomes:

- Vectorization of raster images without a filter
- Vectorization of raster images with the filter

To convert a raster imagery into vector data used the following system:

TABLE I. THE USED SYSTEM

Type of system components	Model
CPU	Intel Core i7 960
GPU	Nvidia GeForce 470GTX
RAM	16 GB
Chipset	iX58

Time for processing images using this system have been spent 7 seconds. The size of the processed raster image was 1060 pixels horizontally and 800 pixels vertically.

Both experiments were carried out using the tool ArcGis (ESRI). The results were as follows:

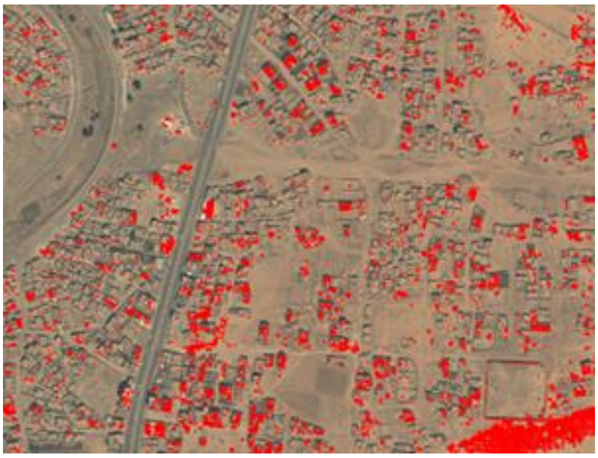

Figure 10. Vectorization of raster images without a filter

Figure 11. Vectorization of raster images with the filter

After tracing the input data image using a threshold filter, it is possible to build a 3D scene with a light raster and vector data:

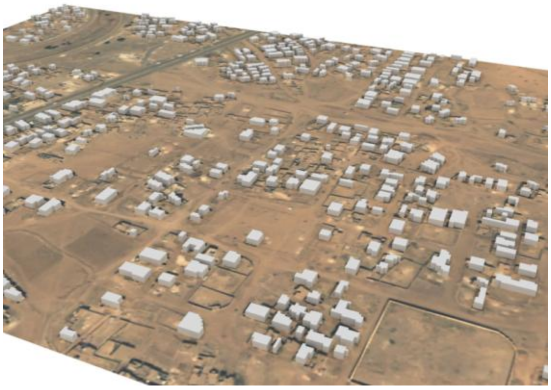

Figure 12. 3D model of the map

V. CONCLUSION

This paper describes a system for converting bitmap imagery into vector data with the construction of 3D maps. The system consists of several parts: preparation of satellite images using the filter Edge Detection, conversion of the received bitmap image into vector data, the construction of 3D maps. This system showed high performance and accuracy in comparison with other methods of processing satellite data. The proposed method has drawbacks: the restrictions on the identification of 3D models, recognition of only rectilinear objects.

The plan for the future work is to improve the current system taking into account all these drawbacks. Create a parallel rendering data on the GPU using the technology Cuda. This method will greatly improve system performance.

VI. REFERENCES

[1] Jianbo Shi and Jitendra Malik. Normalized Cuts and Image Segmentation, IEEE Transactions on pattern analysis and machine intelligence, pp 888—905, Vol. 22, No. 8, 2000

[2] Erick López-Ornelas. High Resolution Images: Segmenting, Extracting Information and GIS Integration, World Academy of Science, Engineering and Technology, 2009

[3] Jahne, H. Scharr, and S. Korkel. Principles of filter design. In Handbook of Computer Vision and Applications. Academic Press, 1999.

Author Index