

SEMI-AUTOMATIC EXTRACTION OF FEATURES FROM DIGITAL IMAGERY

O. Eker^{a,*}, D. Z. Seker^b

^aGeneral Command of Mapping, Photogrammetry Department, 06100 Dikimevi Ankara, Turkey - oeker@hgk.mil.tr

^bITU, Civil Engineering Faculty, 34469 Maslak Istanbul, Turkey - seker@itu.edu.tr

KEY WORDS: Image Segmentation, Semi-Automatic, Feature Extraction, Fast-Marching, Level-Set

ABSTRACT:

Aerial photographs have been evaluated manually by the operators for a long time for the extraction of the vector data. The development of computer technology and digital image processing technologies provide to perform these extraction processes automatically or semi-automatically. Image segmentation can be used for automatic and semi-automatic feature extraction and classification of images. In recent years, image segmentation and the front propagation of the segments have been carried out successfully by the Level Set and Fast Marching methods. In this study, a semi-automatic line extraction method, based on the segmentation of the images using color-differences of the pixels and the propagation of the fronts by the Level Set algorithms, is developed. An object oriented application software is also developed to test the capabilities of the developed method. Some semi-automatic feature extraction applications are made by the help of the developed software using a 1:16000 scale black/white aerial photograph for determining the capabilities of this method. Another application with 1:5000 scaled two ortho images which have 0.50m resolution. These ortho images are generated from 1:16000 scale color aerial photographs. In this test area, an accuracy test is also carried out to find the accuracy of the developed method. The accuracy test is carried out in two groups. In the first group, on 422 road check points, measurements are made and the root square mean is found as $\pm 0.66\text{m}$. In the second group, buildings are used and 281 check points are measured and the root mean square of this group is found as $\pm 0.46\text{m}$.

1. INTRODUCTION

Aerial photographs have been evaluated manually by the operators for a long time for the extraction of the vector data. Computer technology and digital image processing technologies have been developed and this development provides to perform these extraction processes automatically or semi-automatically. The aim of making the processes automatic is to increase the speed of collecting the data and to reduce the cost. Automatic feature extraction studies are firstly motivated to carry out the extraction of roads from digital images because roads have characteristic attributes like width, surface type and geometrical shape which can be modelled more easy than the others.

These studies are showed that the resolution of the images has a very important role in the automatic and semi-automatic extraction of the roads. The developed methods can be classified in three groups: road extraction in low resolution, road extraction in high resolution and multi-resolution road extraction. Most known methods are based on the road tracing and the snakes algorithms.

Another method of automatic and semi-automatic feature extraction and classification of images is the image segmentation. Image segmentation is commonly used the interpretation of the medical imagery like mr-scans (Adalsteinsson and Sethian, 1995).

In recent years, image segmentation and the font propagation of the segments have been carried out successfully by the Level Set and Fast Marching methods (Malladi, 1994).

The goal of this study is to develop a semi-automatic extraction method of the line and area features like roads, rivers and lakes from digital aerial photographs and updating of these features in Geographic Information Systems (GIS) and also to develop an application software to test the capabilities of this method.

According to this goal, a semi-automatic line extraction method, based on the segmentation of the images using color-differences of the pixels and the propagation of fronts by the Level Set algorithms, is developed. An object oriented application software is also developed using Borland C++ and Visual C++.

2. SEMI – AUTOMATIC FEATURE EXTRACTION APPROACH AND DEVELOPED SOFTWARE

The approach that has been developed for semi-automatic feature extraction has been based on level set and image segmentation algorithms. According to this method, three problems had to be solved. The first one was how the algorithm would be started. This problem was solved by marking any point (pixel) on the feature wanted to be digitized by the operator. Thus, level set algorithm would start to work from the point that operator chooses. This solution brought out the approach to be semi-automatic.

Second problem was which criteria would be taken in whether the digitizing feature process would progress or not. This problem was solved by making use of color values of each pixel. The color value of marked point or determined neighborhood level, color values of neighboring pixels and computed color value by getting average, is compared with color value of neighboring pixels and if the color difference is within the

* Corresponding author.

tolerance value determined before, algorithm is continued otherwise it is stopped. In other words, segmentation is carried out according to color differences.

After completing the determination of the feature wanted to be digitized and marking process it is needed to get the feature as vector data. Thus the data can be used in a Geographic Information System (GIS). This was the third problem. Raster to vector transformation was carried out to solve the problem. Semi-automatic digitizing approach that solves those three problems mentioned above was configured in five main steps as the integrity of processes. This process steps are:

- The selection of starting point or pixel of the feature wanted to be digitized by the operator.
- Performing the image segmentation of the selected pixel by making use of color difference with neighboring pixels.
- Propagation of the segmentation by using the level set method and storing as a heap sort.
- Getting 1 bit (1 color) masked image from those structured pixels.
- Getting features' vector data in a known format after raster to vector conversion from masked image.

First four process steps were compiled using Borland C++ programming language and using no library files except for system libraries. Fifth step was carried out by the help of a raster to vector conversion open code that is shared in the internet. This code was developed and compiled by using Visual C++ programming language. Some functions like weed tolerance and coordinated transformation tools were added to this code.

2.1 Image Segmentation Algorithm

Image segmentation has been based on color difference and this algorithm had been made flexible by adding a tolerance value that could be set by operator. Thus operator can detect and digitize large areas at one click of the mouse by giving high tolerance value in high contrasted images. Segmentation algorithm threshold value was determined as two times of multiplied value of image width and length.

Segmentation is started from the selected feature point. Value of neighboring pixels and their average value are computed as the reference, according to pixel value and/or neighboring level of the feature point. If neighboring value is zero, pixel's color value is taken on the reference neighboring level is one, color value computed by averaging with the eight pixels around it is taken as the reference value (Eker, 2006).

Here, color differences of the pixels in three bands (red, green, blue) are computed one by one and searched whether each is under the threshold or not.

2.2 Storing and Propagating With Level Set Algorithm

Starting value needed for propagation with level set or going along with the feature is coped with by means of the semi-automatic nature of developed method. Propagation of the surface would start from the pixel which is selected by the

operator and zero level set would be defined by the location of starting pixel.

Another component that is needed for level set algorithm is the limit value providing the control of propagation. Solution for the limit was carried out by computing the color differences as mentioned above.

Latter there had been one problem to carry out that was which neighboring pixel would be chosen to continue fitting the limit value. As a matter of fact in this solution of the problem Fast Marching method may be used. The function would go on to propagate hence the pixel that has the least value (Sethian, 1998). However, what would the least value be?

In the developed method for the question of what would the least value represent to, the first pixel selected by the operator is admitted as zero level set function and each neighboring pixel (the first pixels in east, west, north, south direction) is checked by the color difference value algorithm mentioned above and the distance from the zero level set (first selected pixel) is computed for the pixels fit the condition, thus diffusing is carried on hence the pixel having the least distance.

Completing the propagation updating process should be carried out (Sethian, 1998). For updating process quadratic equations were used required for computing the difference (Sethian, 1998).

Minimum heap sort had been used in order to store and access the pixels. In the structure of minimum heap sort, the image cell having the least distance from level 0 would be at the top. Certain structure is required for preserving the heap sort. Adding the image cell to the heap, fixing the image cell in the heap, being removed the image cell having the least distance from the heap and updating the structure on the event of every new image cell is added is required.

By means of this storing structure, effectiveness has been increased in large volume processes such as attaining to the image cells, testing and computing the propagation, marking the image cells and storing the marked image cells (Eker, 2006).

2.3 Raster To Vector Conversion

Raster to vector conversion is valid for 1 bit image files which include two types of data (0 or 1), (ESRI, 1997). By the help of the developed software, extracted pixels from the image are marked as colored and the others are black. This new image is saved as a masked image in bmp format.

Open Visual C++ code shared in the internet, was developed and compiled with adding additional functions to convert the acquired mask image file to vector data. The center and the border lines of the features are being converted from raster data to vector data by the help of an interactive interface by setting the connection with the main interface. A coordinated vector data is gained by entering the lower left corner coordinates and image resolution in both two dimensions (x, y) on the main interface. Besides, entering the weed tolerance with an interactive interface, it is possible to get vector data in desired smoothness (Eker, 2006).

If weed tolerance is 0 all the pixels are included in calculation without making smoothness. When the weed tolerance is increased, pixels with increasing intervals are taken into consideration instead of all pixels and vector data becomes

smooth. However, it is probable the geometry of vector data may get worse if higher weed tolerance is given (Eker, 2006).

3. TEST STUDIES

Test studies were carried out with the program developed in two groups in order to determine the usefulness and effectiveness of this new approach exposed.

In the first group test study a 1:16000 scale black/white digital aerial photograph was used. Thus it was investigated whether the software would be used or not to extract some features in large scaled mapping.

In the second group study Another application with 1:5000 scaled two ortho images which have 0.50m resolution. These ortho images are generated from 1:16000 scaled color aerial photographs. In this test area, an accuracy test is also carried out to find the accuracy of the developed method.

3.1 Test 1

In the first test work that a 1:16000 scaled white/black aerial photo which was scanned at 20 microns was used. This image includes a village center. Roads were selected as the features that were extracted semi-automatically from the image. It is shown in the Figure 1 that vector data which were obtained from the result of feature extraction studies. Roads were vectorized from their edge lines. The roads' surfaces are not homogeneous. Because of this in this image the extraction of the roads were a hard work. In this image, if we selected a big tolerance value the extraction extended out of the roads. On the other hand if we selected small tolerance values we saw that we had to make small propogations and select a lot of starting points. This increased the time of the extraction.

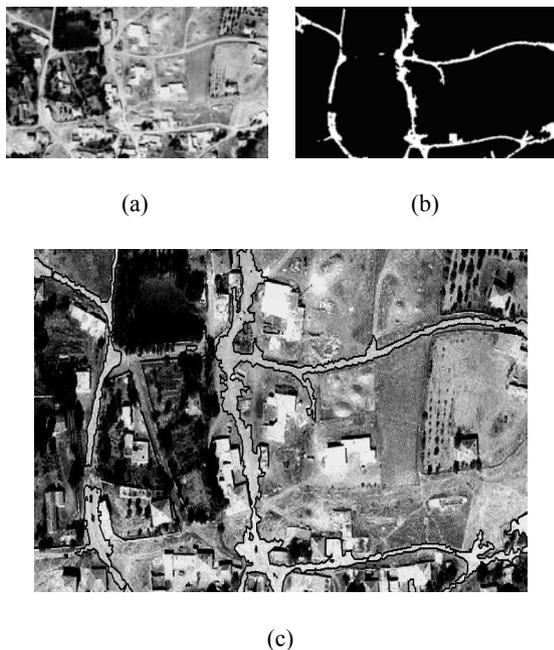


Figure 1. (a) The original image (b) Mask image of the extracted roads (c) Extracted vectors.

3.2 Test 2

It is thought that this software can be used in producing maps and in collecting the vector data for GIS. Another application with 1:5000 scaled two ortho images which had 0.5m resolution of Ayazağa Campus of İstanbul Technical University. These ortho images were generated from 1:16000 scaled color aerial photographs. In this test area, an accuracy test was also carried out to find out the accuracy of the developed method. In this accuracy test, vector data of the roads and buildings were collected semi-automatically with the developed software (Figure 2) and also manually by an experienced operator (Figure 3). The data collected by the operator were assumed as the correct ones and they were compared with the others collected by the software. The accuracy test was carried out in two groups. In the first group, roads were used and on 422 road check points were measured. In the second group, buildings were used and 281 check points were measured. The results are shown in Table 1.

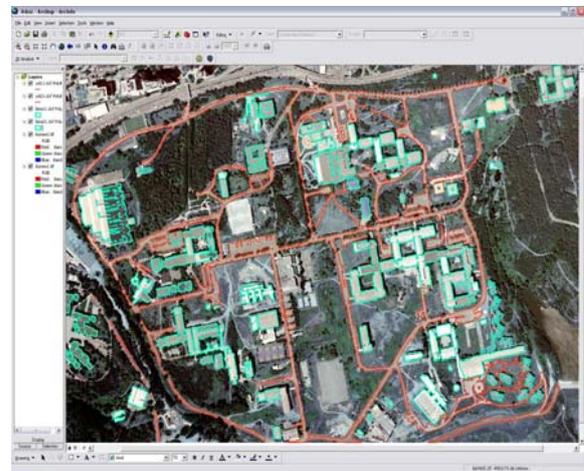


Figure 2. Semi-automatically extracted features.



Figure 3. Manually extracted features by the operator.

	Mean	RMS	Min.-Max. Errors
Roads	0.93 m.	±0.66 m.	0.10 m. - 4.18 m.
Buildings	0.71 m.	±0.46 m.	0.03 m. - 2.29m.

Table 1. Results of the accuracy test.

4 CONCLUSIONS

As the results of the applications and tests, it can be said that the accuracy of this developed method is ± 1 pixel size of the used imagery. It can be used correctly for producing maps and collecting vector data for GIS. Especially for lakes, rivers and buildings can be collected very efficiently. Different classifications and segmentations, which an operator's can not see, can be made also with the adjusting of the tolerance value. Roads which have good quality can be vectorized from their center lines and/or boundaries according to the scale of the image used.

Some weak sides of this developed method and software are also found out. These are:

- Especially on big scale aerial photographs, the obstacles on the features, as trees, cars and shadows, effects the extraction of the features negatively. Effects of this factor are reduced whether the scale of the image gets smaller.
- If the tolerance value is not be adjusted to the correct values, wrong features can be extracted.
- When a big size image is used, the software gives back some errors because the size of the arrays are directly proportional to the number of the pixels.
- The quality, contrast and noise of the image effect the feature extraction process.

- The surface attributes of the features also effect the success degree of the feature extraction.

If the noise and the contrast of the images are eliminated by the image process algorithms like edge detection algorithms and filters as anisotropic diffusion and the blanks that are generated by the obstacles on the feature can be enterpolated by the different kinds of entepolation methods, more good results can be achieved by the developed method and the software. Also, for the image segmentation different types of segmentation like snakes, instead of color difference and for big size images pyramid levels can be used to increase the success degree of this method.

REFERENCES

- Adalsteinsson, D., Sethian, J.A., 1995. Fast Level Set Method for Propagating Interfaces, *Journal of Comp.Phys.*, 118, pp 269-277.
- Eker, O., 2006. Semi-Auotmatic Extraction of Linear Feature From Aerial Photograps, *Ph.D. Thesis*, İTÜ, İstanbul.
- ESRI, 1997. ARC/INFO User's Guide Cell-Based Modelling With GRID, Redlands, USA.
- Malladi, R., Sethian, J.A. ve Vemuri, B.C., 1994. Evolutionary Fronts for Topology-Independent Shape Modeling and Recovery, *Proceedings of Third European Conference on Computer Vision, Stockholm, Sweden, Lecture Notes in Computer Science*, 800, pp 3-13.
- Sethian, J.A., 1998. Fast Marching Methods and Level Set Methods for Propogating Interfaces, *von Karman Institute Lecture Series*, Computational Fluid Mechanics, Belgium.
- URL (Raster to vector conversion open code), <http://www.xmailserver.org/davide.html> , 28 February 2006.