DESIGN AND DEVELOPMENT OF A NEW MODEL FOR AUTOMATIC CHANGE DETECTION OF BUILDINGS FROM AERIAL IMAGES

Simintaj ziraksaz [1] ,Hamid Ebadi [2] ,Farshid Farnood Ahmadi [3] , saeed sadeghian 4
simin_ziraksaz@yahoo.comM.Sc. student, E-mail:1.
ebadi@kntu.ac.ir Assistant Professor, E-mail:2.
farshid_farnood@yahoo.com 3.Ph.D student, E-mail:
Department of Photogrammetry and Remote Sensing
K.N. Toosi university,Faculty of Geodesy and Geomatics Engineering
4.Ph.D, E-mail: sedeghian@ncc.neda.net.ir
National Cartographic Center of Iran (NCC)

Commission VI, WG VI/4

KEY WORDS: Aerial photos, Building, Change detection, Gis, Matching, Spatial database

ABSTRACT:

One of the most important tools in order to model real world phenomena is Geographic Information System. Real world phenomena are changing continuously, so it is necessary to detect and introduce changes to system in order to update GIS databases and model real world correctly. Geometric stability and high spatial resolution of aerial images leads to detect changes precisely. It is necessary to consider that feature recognition and extraction processes are two important stages in all of the change detection methods in which accurate results in detecting geometrical changes depends on performance of theses two stages. Each of these stages needs to use image processing algorithms. On the other hand, these algorithms are complicated, time consuming and the performance of them depends on specific conditions such as image acquisition conditions and so on. Therefore using the methods with low dependency on image processing can reduce these problems and increase the accuracy and reliability of the results. To achieve this aim, the integration of aerial photos and GIS spatial databases is suggested. In this research, a new algorithm was designed and implemented for automatic change detection of buildings based on the development of least squared matching technique. The accuracy assessment showed that change percentage of the regions that the algorithm can detect their changes correctly is 70% and the precision of this novel approach to approximate extracted edges with the real ones is 0.45 pixels.

1. INTRODUCTION

In photogrammetry and remote sensing, matching can be defined as the establishment of the correspondence between various data sets (Geodaetischas seminar ss/2000). The matching problem is also referred to as the correspondence problem. The data sets can represent images, but also maps, or object models and GIS data. The first step of matching is primitive extraction. The distinction between different matching primitives is probably the most prominent difference between the various matching algorithms. The primitives fall into two broad categories: either windows composed of grey values or features extracted in each image a priori are used in the actual matching step. The resulting algorithms are usually called: Area based matching (ABM), and Feature based matching (FBM), respectively (Bohuslav, 2004). Besides of these two algorithms, the combination algorithm can be also considered. This algorithm combines ABM and FBM in order to take the advantages of these two algorithms (Gruen, 1985).

Correlation coefficient and least squares matching (LSM) are two approaches used in ABM. Cross-correlation works fast and well, if patches to be matched contain enough signal without too much high frequency content (noise) and if geometrical and radiometric distortions are kept at minimum. Both conditions are often not encountered in or with aerial images, while LSM minimizes differences in grey values between the template and search image patches in an adjustment process where geometric and radiometric corrections of one of matching windows are determined (Bohuslav, 2004). On the other hand FBM is implemented in three ways point-based matching edge-based matching and region-based matching based on the type of features extracted from images (Geodaetischas seminar ss/2000). Among of these approaches edge-based matching can presented accurate and reliable results because of its specific radiometric characteristics (Agouris, 1992), (Baltsavias, 1991). So the combination of LSM and edge-based matching is used in this article as an optimal method to obtain desired results in building change detection process.

The goal of our study is to develop LSM for the identification of changes in buildings outlines. In this paper change is extracted thorough comparisons of observations. The differences in general exposure conditions among two different images in two distinct time instances may affect substantially the performance of the above described matching method. In order to minimize the effects of variations on our solution, we have to allow edge pixels to influence the solution more than the rest of the template. This can be performed by manipulating the corresponding weight matrix P.

Therefore our approach is based on the use of least squares template matching, where prior data are analyzed to provide template information. The product of such a process is the identification of changes in object outlines. Our work is innovative in its use of prior information to provide templates for matching, and in its analysis of template information to assign proper weights in the least squares solution. In this paper we present theoretical models and implementation considerations behind our approach for change detection. The steps of implementation of this proposed algorithm is described as below.

2. PRESENTING A NOVEL APPROACH TO DEVELOP LSM IN ORDER TO BUILDING CHANGE DETECTION

2.1 Object decomposition

For the purpose of our approach, a building is expressed through a wireframe representation based on prior vector information (Habib, and Kelley, 2001). For decomposition of the lines, a new element is introduced, the Minimum Spatial Element (MSE). The MSE describes the resolution of spatial change that the user is interested in. Using this information, we perform a segmentation of outlines, and lines are essentially substituted the corresponding points along the outline (Figure. 1).

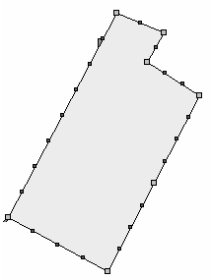

Figure 1: object decomposition

2.2 Providing template

In order to provide template, first the image coordinate of the point along the building edges is computed for each image by colinearity equation. The image coordinates is considered as the centre of the template. Determining the size of the templates is the second step. This allows us to optimize computational performance without compromising accuracy potential. From an accuracy and reliability point of view, small image patches, e.g. 3 x 3 pel^2 are not suitable for LSM due to small redundancy number. An optimal size of matched window also differs with a pixel size and scale of the original images. A size of 15 x 15 pel^2 for good quality images with a lot of details and at least 21 x 21 pel^2 for noisy images or when significant differences in brightness between a template and a search area occur are recommended in (Kraus, 1997). So, the appropriate template size for our images used in this research is 21 x 21 pel^2 .

2.3 Development of least square template matching

A window depicting an edge pattern is introduced as a reference template that is subsequently matched to digital image patches in the vicinity of actual edge segments. The concept behind the method is simple yet effective: by matching the edge template window to an image window, we can identify edge locations in the image as conjugate to the a priori known template edge positions. Assuming *f(x,y)* to be the reference edge template and *g(x,y)* to be the actual image patch, a matching correspondence is established between them when

$$f(x,y)=g(x,y) \quad (1)$$

However, considering the effects of noise in the actual image, the above equation becomes

$$f(x,y)-g(x,y)=e(x,y) \quad (2)$$

With *e(x,y)* being the error vector.
In a typical least squares matching method, observation equations can be formed relating the gray values of corresponding pixels, and they are linearized as(Grien, and Akca, 2005):

$$f(x,y)-e(x,y)=g^o(x,y)+\frac{\partial g^o(x,y)}{\partial x}dx+\frac{\partial g^o(x,y)}{\partial y}dy \quad (3)$$

The derivatives of the image function in this equation express the rate of change of gray values along the *x* and *y* directions, evaluated at the pixels of the patch. Depending on the type of edge, the geometric relationship describing the two windows may be as complex as an affine transformation, or as simple as a simple shift and/or rotation. To facilitate solutions we can resample template and/or actual image to have edges lying along one of the major coordinate axes. For example, we can have resampled edges oriented along the y axis of the corresponding windows. In this case, the above equation may be reduced to:

$$f(x,y)-e(x,y)=g^o(x,y)+\frac{\partial g^o(x,y)}{\partial x}dx \quad (4)$$

Where the two patches are geometrically related through a shift along the x direction:

$$X_{NEW}=X_{OLD}+dX \quad (5)$$

The dX parameter is the unknown that allows the repositioning of the image window to a location that displays better radiometric resemblance to the reference template. Regardless of the choice of geometric transformation, the resulting observation equations are grouped in matrix form as

$$-e = Ax - l;\ P \quad (6)$$

Where, *l* is the observation vector, containing gray value differences of conjugate pixels. The vector of unknowns *x* comprises the shift at the x direction, while *A* is the corresponding design matrix containing the derivatives of the observation equations with respect to the parameters, and *P* is the weight matrix. A least squares solution allows the determination of the unknown parameters as

$$\hat{x}=(A^TPA)^{-1}A^TPl \quad (7)$$

Based on the formulation of the above model, we have to face the major challenge about formulation of the weight matrix described in the next section.
In order to better handle noise through the analysis of local edge information within the matching windows, we have to allow edge pixels to influence the solution more than the rest of the template. This can be performed by manipulating the corresponding weight matrix P. Indeed, by assigning high weight values to certain pixels, we allow them to influence the solution more than pixels with low weight values and the effects of undesired variations on our solution are kept at minimum.

In order to take advantages of the raster and vector representation that preexists and enhances the algorithm's performance and liability, first a geometric representation of the edge is extracted and then included in the LSM solution.

2.3.1. Extraction of geometric representation of the edge

We categorize edges as having two, three or four levels, depending on their type, based on radiometric representation (Phalke, 2005) (Fig. 2).

Figure 2: two levels (left), three levels (middle), four levels (right)

This corresponds to one, two or three edges on the image. We resample our window so that the edge is perpendicular to the x-axis of the window under examination. Then, we calculate the average of the first derivative of each column, at the X direction:

$$g_x(x,y) = \frac{\partial g(x,y)}{\partial x} \tag{8}$$

$$\bar{g}_x(x) = \frac{\sum_{1}^{y} g_x(x,y)}{y} \tag{9}$$

Clearly, for our set-up, the derivatives along the y-axis will be minimal. An analysis of the $\bar{g}_x(x)$ graph in Fig. 3, by using the first and the second derivatives, reveals 3 (or less) maxima. Three criteria are introduced in the decision process to accept or reject a maximum that in essence corresponds to an edge. If we define $\max_0$ as the maximum of the three maxima found, then we compare this value with the existing maxima.

. . If $\max_i > T_1 \times \max_0$ then that edge is accepted because the high value of relative sharpness reveals a strong edge. This constant was set empirically to the value of 70%.

. If $\max_i < T_2 \times \max_0$ then that edge is rejected because the low value of relative sharpness reveals a weak edge. This constant was set empirically to the value of 30%.

. If $T_2 \times \max_0 < \max_i < T_1 \times \max_0$ then we compare the second derivative, before and after the candidate maximum, by using the following criterion:

$$\left|\overline{\nabla}_x(x)\right| + \left|\overline{\nabla}_x(x+1)\right| \le T_3 \max_0 \tag{10}$$

$(\bar{g}_x(x))$

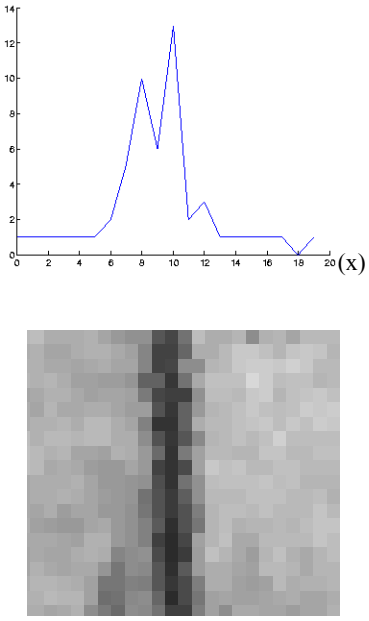

Figure 3: averages of the first derivative in the x direction (top), edge template (bottom)

If the above condition is satisfied, then the point is rejected, otherwise, it is included in the analysis. T_3 was set empirically to the value of 50%.

The above mentioned percentage values may be viewed as general parameters of our approach. The values included here are results of empirical analysis.

With the three above criteria, a "relative" check based on maximum value, compares the dominant edge with other gradient maxima to eliminate false responses. Before we finalize the levels, we perform a last check by introducing the first derivative in the *Y* direction. We claim that an edge should have high gray value variations perpendicular to itself, but at the same time low variations along itself. So in this step, we check for consistency along the Y-direction, but only where the edges (accepted maximums) that came from the previous step are not strong enough. The difference with the criteria used before is that this time absolute gray values are utilized, acting as thresholds, and not relative ones. During this process two thresholds are introduced:

- The Δgx, showing the difference in the gray values that might not imply an edge,
- The Δgy, expressing the difference in the gray values that might not imply homogeneity

First we compare the accepted maximums of the previous step, with our Δgx. If the condition

$$\max_i < \Delta gx \tag{11}$$

is satisfied, there is the suspicion that local noise might exist or the radiometric representation of the edge is not strong. To distinguish these two cases, we compute the first derivative at the *Y* direction, at the position where the candidate maximum is, and if it is smaller than the Δgy, then that point is accepted. Trial and error Experiments showed that a value of 40 gray values should be assigned to Δgx and a value of 7 gray values to Δgy.

2.3.2. Expressing Geometry Semantics through the Weight Matrix

In the previous section, the analysis on prior raster information provided a description of the geometry of the edge that we are trying to match. This geometric information is inserted in our mathematical model by formulating accordingly the weight matrix. We claim that a combination of Gaussian distributions can be applied in order to assign higher weights at the accepted maxima. The formula of the Gaussian distribution is expressed as:

$$G(x) = \frac{1}{\sqrt{2\pi\sigma}} e^{\frac{(x-\mu)^2}{-2\sigma^2}} \tag{12}$$

In this formula μ is the mean, s is the standard deviation and x is the coordinate of the pixel on the axis perpendicular to the edge. The main goal considered in designing the weight matrix is minimizing the effects of variations on our solution. In order to achieve this goal, we have to allow edge pixels to influence the solution of Equation 6 more than the rest of the template. This can be performed by manipulating the corresponding weight matrix P (of Equations 6 and 7). Indeed, by assigning high weight values to certain pixels, we allow them to influence the solution more than pixels with weight values approaching 0. Accordingly, we enhance the solution of the model presented earlier by incorporating local edge analysis in it. This transforms our template matching from a common area-based matching scheme to a content-based matching process, improving its performance potential.

The formula used to formulate GD of the weights in the case of an edge represented by two levels corresponds to equation 12. In this equation, the mean expresses the position of the edge as defined by the older vector information. The standard deviation defines the uncertainty for existence of the edge on the position of the mean. The standard deviation depends on the resolution of an image, because in higher resolution the edge is expected to look sharper and in coarser resolution more blurry. In our case, for an average resolution of 1/10000, a threshold of 99% is considered. So the standard deviation should be one third of d/2 in which d is the template size. Therefore, a standard deviation value of 4 pixels is assigned for template size of 21×21.

When the edge is composed of three levels, we define d as the distance between two μ. The GDs are used with their means μ_1 and μ_2, respectively. The mathematical representations for these distributions are:

$$G(x) = e^{\frac{(x-\mu_1)^2}{-2\sigma_1^2}}, \quad G(x) = e^{\frac{(x-\mu_2)^2}{-2\sigma_2^2}} \tag{13}$$

In order to formulate the GDs and in essence of the weight matrix, we need to define the two values for the standard deviation s_1 and s_2 .So, we need two equations. The first one is extracted from the fact that a point L should exist, where the two GDs intersected. Let d_1 and d_2 be the two distances from μ_1 and μ_2, respectively and *a* is the GD value of point L (Fig. 4). The first equation is:

$$\sigma_1 + \sigma_2 = \frac{d}{\sqrt{-2\ln a}} \tag{14}$$

In order to compute the standard deviations of the Gaussians, another constraint should be introduced. Due to the fact that point B is the pre-extracted edge and point A is a computed peak, we do not want point A to influence the solution beyond point B. mathematically this is ensured if:

$$s_2 > s_1 \tag{15}$$

On the other hand, that can statistically ensured by restricting the standard deviation. For a threshold of 99% for possibility of the observation to fall inside an interval, interval should be three times the standard deviation. So another equation is expressed as:

$$3\, s_1 = d \tag{16}$$

We substitute that in the first equation and we calculate s_2 .

$$\sigma_2 = d\,\frac{3 - \sqrt{-2\ln a}}{3\sqrt{-2\ln a}} \tag{17}$$

Since ($-2\ln a$) should be positive, the value of *a* should always satisfy the equation:

$$a \le 1 \tag{18}$$

If the equations (15), (16), (17) and (18) are combined, we find:

$$0.32 < a < 1 \tag{19}$$

Constant *a* expresses the weight of the intersecting point. If that weight is low, then the standard deviation of the second GD will be small, and weights will be denser near the point B. This will be applicable in the case of higher resolution, where the edges are distinct. On the other hand, if we choose a high intersecting value, then the GDs get wider and increase the contribution of surrounding pixel to point B. This is desirable in low resolution imagery, where edges are not so distinctive. Experiments showed that for an average resolution of 1/10000, a value close to 0.6 should be assigned to *a*. Finally, the weight distribution P(x) on the axes perpendicular to the vector edge is given by the formula:

$$P(x) = Max\{G_1(x), G_2(x)\} \tag{20}$$

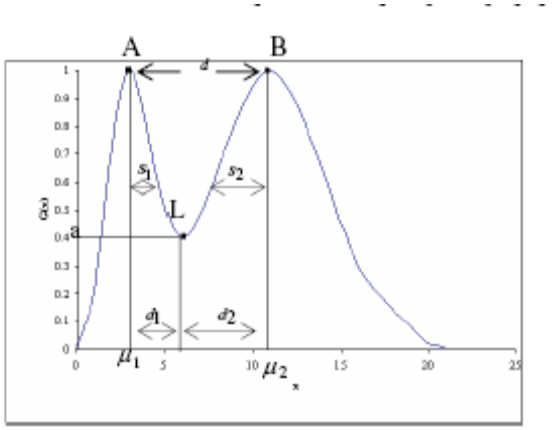

Figure 4: GDs in the case of an edge represented by three levels

When multiple edges exist, they can result from the actual shape of the edge or random condition such as noise, shadow, different projection, etc. We compensate for theses effects by introducing a scaling factor describing the expected geometry of the edge. Two new GDs are created on the both side of the vector edge. The means of these GDs are assigned the values of $\mu_* \pm d_i$ depend on which side they are (Fig. 5). The standard deviation of the expected geometry GDs expresses the uncertainty of this information and it is defined as a constant based on the distance of each edge to vector edge d_1 and d_2. The value of standard deviation corresponds to value of $d_i / 3$, for a threshold of 99%. The experiments showed that the standard deviation should be limited to 40% at the distance of standard deviation to compensate random condition. The mathematical expression of scaling factor is:

$$K_i(x) = Max\{e^{\frac{(x-\mu_*-d_1)^2}{-2\sigma_i^2}}, e^{\frac{(x-\mu_*+d_2)^2}{-2\sigma_i^2}}\}, \qquad (21)$$

$$G_i(x) = e^{\frac{(x-\mu_i)^2}{-2\sigma_i^2}}$$

The weight distribution P(x) on the axes perpendicular to the vector edge is given by the following formula:

$$P(x) = Max\{K_1(x) \times G_1(x), G_2(x), K_3 \times G_3(x)\} \quad (22)$$

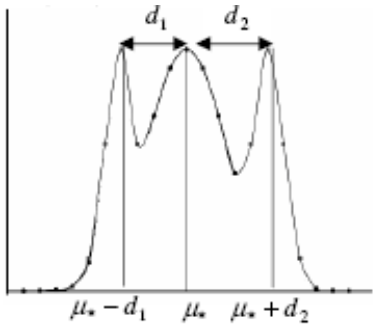

Figure 5: Gaussian distribution in the case of an edge represented by four levels

2.4. Decision process

The statistical tools of least squares adjustments provide the mathematical foundation necessary to come with a valid analysis of the obtained results and automatically decide whether change occurred or not. During the execution of the matching loop, several criteria are considered such as Shift dX. A threshold of 0.5 pixels is set to classify a matching as successful. During the application of the least square matching process, a significant amount of visually successful matches was not correctly identified, because the value of the shift was not close to our threshold. The distinction between a successful match and an unsuccessful one is the range of dX. If the variation is less than 0.5 pixels, then the match is considered successful, otherwise rejected.

3. RESULT AND DISCUSSION

The developed algorithm in this paper is implemented using Microsoft's visual C# Net ،Visual Basic 6 and Esri's MapObjects ActiveX component. The datasets and assessment process is described as follow. The available datasets to test change detection algorithm performance is described in table 1.

datasets	properties
Digital image of city	• IslamAbad • 1/10000 • 1988 • TIFF format
Digital image of city	• The same area(IslamAbad) • 1/10000 • 2002 • TIFF format
Vector information	• 1988 • Esri's shapefile format

Table 1: datasets and their characteristics to test proposed methodology

3.1 Evaluation of implemented system

3.1.1 Accuracy assessment of developed least square matching algorithm

The accuracy of the measurement refers to how close the measured value is to the true or accepted value. So, we have considered 50 buildings with known information in the test area, and compared the implemented system results with expected ones, in order to evaluate the accuracy of developed building change detection algorithm**.** The result of this evaluation has been shown in table 2.

Comparison between expected and computed result	Percentage (%)
Buildings that algorithm can detect their changes correctly	Upper 70%
Buildings have changed but the algorithm can not detect their changes	Under 10%
Buildings have not changed but the algorithm detect changes	Approximately 20%

Table 2: the accuracy assessment of implemented system

The statistical analysis of obtaining result showed that the performance of implemented system is desirable to detect building changes, while its performance have some limitations in buildings with no changes. This is caused by detecting unreal edge because of shadow, difference in exposure condition or losing actual edge made by shadow or vegetation coverage.

3.1.2 Precision assessment of developed least square matching algorithm

Precision refers to how close together a group of measurements actually are to each other. So the accepted criteria to precision assessment of proposed algorithm results, is shift dX. This parameter is calculated from the equation 7. The output result of this equation for two matched point after several iteration, is residual shift which is considered as algorithm precision. Indeed, the shift value is represented the ability of implemented system to separate two edge with no change. So the shift value of more precise system is smaller than the imprecise one. In an ideal case of a perfect match dX=0, therefore the shift value differences between real case and ideal one is defined the precision of our developed system. So the precision of our developed algorithm is calculated by averaging 10 matched points shift values for convergence case. Experiments showed the shift value of 0.45 pixels as algorithm precision.

4. CONCLUSIONS

The algorithm we presented in this paper extends the concept of least squares template matching to identify object outline changes. By using image orientation parameters and positional data we can reduce the problem of 3-D object monitoring to a 2-D image-space matching problem. Analysis of semantics within a template, before the actual matching taken place, improves the accuracy and reliability of the presented technique.

The accuracy assessment showed that the change percentage of the regions that the algorithm can detect them correctly is 70%, the change percentage of the regions that the algorithm can not detect their changes is 10% and the change percentage of the regions with no changes while our algorithm detects changes in these regions is 20%. The statistical analysis of obtained result showed that the performance of implemented system is desirable to detect building changes, while its performance has some limitations in buildings with no changes. This is caused by detecting unreal edges because of shadow, difference in exposure condition or losing actual edges made by shadow or vegetation coverage.

The ability of implemented system to separate two edges with no change is considered as precision criteria. Indeed, the average of shift values represents the precision of our implemented system. Experiments showed that the algorithm precision is 0.45 pixels.

The capability of this novel approach is representing the image and object coordinates and change percentages of urban area and also each building in it.

REFERENCES

Geodaetischas seminar ss/2000. "Matching methods for automatic DTM generation". www.photogrammetry.ethz.ch/general/person/maria/matching.pdf((accessed December 7,2007).

Bohuslav,V.,2004, "Image matching and its applications in photogrammetry", 15-53.

Gruen A.,1985,"Adaptive Least Squares Correlation: A Powerful Image Matching Technique", South African Journal of Photogrammetry, Remote Sensing & Cartography, Vol. 14, No. 3, pp. 175-187.

Agouris P.,1992, "multiple image multiple matching for aerotriangulation". The Ohio state university.Colubbus.ohio.

Baltsavias , E.P. , 1991,",multiphoto geometrically constrained matching" 49. ETH-institute for geodesy and photogrammetry.Zurich.

Habib, A., and Kelley, D. ,2001,"Automatic relative orientation of large scaleImagery over urban areas using Modified Integrated Hough Transform", ISPRS Journal of Photogrammetry& Remote Sensing 56, pp. 29-41.

Kraus, K., 1997, "Photogrammetry", Vol.2, 4th edition, Fer. Dümmlers Verlag, Bonn, ISBN 3-427-78694-3.

Grien, A., and Akca, D., 2005, "least square 3D surface matching", ASPRS annual conference, March 7-11.

Phalke, S., May 2005, "Change Detection of Man-made Objects Using Very High Resolution Images", university of CALGARY, Department of Geomatics Engineering.