LEAST SQUARE MATCHING 1 TO COMBINE THE TWO IMAGES OF NCU AREA

Hepi Hapsari Handayani

Department of Geomatics Engineering, FTSLK-ITS, Kampus ITS Sukolilo, Surabaya, 60111, Indonesia
e-mail: hepihapsari@gmail.com

Abstract

Least square matching technique is included in area-based digital matching method. Conceptually, least square matching is closely related to the correlation method, with the added advantage of being able to obtain the match location to a fraction of a pixel. Least square matching (LS1)1 has merit to minimize the sum of squares for grayscale differences, so the result will be more accurate.

The images covering the National Central University (Taiwan) area are aerial images taken from digital camera with sensor ultracam-D. Interior orientation parameter consist of focal length in 101.400000mm, principal point offset (0.000000e+000, 0.000000e+000)mm, and principal point symmetry (-2.110000e-001, 0.000000e+000)mm.

The experimental result shows that the best accuracy of x direction is reached when the rotation angle is 9 degree, then those of y direction is reached when the rotation angle is 3 degree. The accuracy of both directions is getting worse when the scale of image is less than 0.8. The success rate 100% is reached in all of window size except 51 and 101. Then, the best accuracy of x direction is showed in 3x3 window size, those of y direction is employed when the work used the window size 11x11. Based on the experimental result, it can be concluded that using different rotation and scale can get the different result that it will be worse or better. Thus, to get the better result in matching image and better accuracy, the work should use the orthorectified image as base image to do rotation scheme and use small window size to minimize the number iteration, but it will be not significant with RMSe.

Keywords: Least square matching, aerial images, NCU area

INTRODUCTION

One tool for making automatic parallax measurements in digital images is the correlation of the image grey levels using the least squares technique. The least squares image matching method has been investigated practically by many researchers. The results show in all cases that matching with subpixel accuracy is possible. The purpose of this project is to empirically investigate a method for finding the optimal window size using the least squares matching method, with regard to precision, and to investigate optimal window sizes, with regard to both precision and reliability, in this particular material. In this article two terms describing accuracy are used, precision and success rate.

The terminology is related to the result of the matching, not to the matching method. The precision has been computed using manually selected gross error free check points. The precision measure is the root mean square deviation between the manually measured parallaxes and the automatically measured parallaxes. Points with typical indications of gross errors, such as too large deviations, too large standard deviation of the parallax or too slow convergence, were excluded. The matching windows should also visually look similar, a matching should a priori be possible.

METHOD

Least square matching (LS1)1 has merit to minimize the sum of squares for grayscale differences, so the result will be more accurate. Using LS1, we should do point prediction using resampling for first initial point. This step has aim to limit size of the area for selecting the corresponding point.

Image normalization is for keeping the window with the same mean and variance (μ, σ^2). It means that the two lines that we want to calculate the distance have been in the same level. The equations for image normalization are

$$\left. \begin{aligned} g_T &= r_0 + r_1 . G_T \\ r_1 &= \frac{\sigma_S}{\sigma_T} \\ r_0 &= \mu_S - r_1 . \mu_T \end{aligned} \right\} \qquad (1)$$

Where,

G_T = original grayscale
g_T = normalized grayscale
r_0 = shift factor of grayscale
r_1 = scale factor of grayscale
σ_T = standard deviation of grayscale in target window
σ_S = standard deviation of grayscale in search window
μ_T = mean grayscale of target window
μ_S = mean grayscale of search window

The equations of displacement in x and y direction are

$$\Delta G = G'_x . \Delta x + G'_y . \Delta y$$

Where,

$$\left. \begin{aligned} \Delta G &= G_S(x,y) - G_T(x,y) \\ G'_x &= \frac{dG_T}{dx} \approx (G_S(x+1,y) - G_S(x-1,y))/2 \\ G'_y &= \frac{dG_T}{dy} \approx (G_S(x,y+1) - G_S(x,y-1))/2 \end{aligned} \right\} \quad (2)$$

Δx is unknown displacement in x direction and Δy is unknown displacement in y direction.

$$\left. \begin{aligned} \Delta x &= \frac{\sum\sum(G'_x.\Delta G)*\sum\sum(G'_y)^2 - \sum\sum(G'_y.\Delta G)*\sum\sum(G'_x.G'_y)}{\sum\sum(G_x')^2 \sum\sum(G_y')^2 - \sum\sum(G'_x.G_y')^2} \\ \Delta y &= \frac{\sum\sum(G'_y.\Delta G)*\sum\sum(G'_x)^2 - \sum\sum(G'_x.\Delta G)*\sum\sum(G'_y.G'_x)}{\sum\sum(G_y')^2 \sum\sum(G_x')^2 - \sum\sum(G'_y.G_x')^2} \end{aligned} \right\} \quad (3)$$

The procedures of the least square matching1 are:
1. Specify the coordinate of pixel in the left image that is corresponding with the coordinate of pixel in the right image. This is for the initial value as approximation at the beginning of iteration. The pixel of subarray B (along with a 1-pixel-wide border around B which is needed for derivative estimates) are resampled. The resampling method of this experiment is nearest neighbor. The nearest neighbor resampling involves assigning the

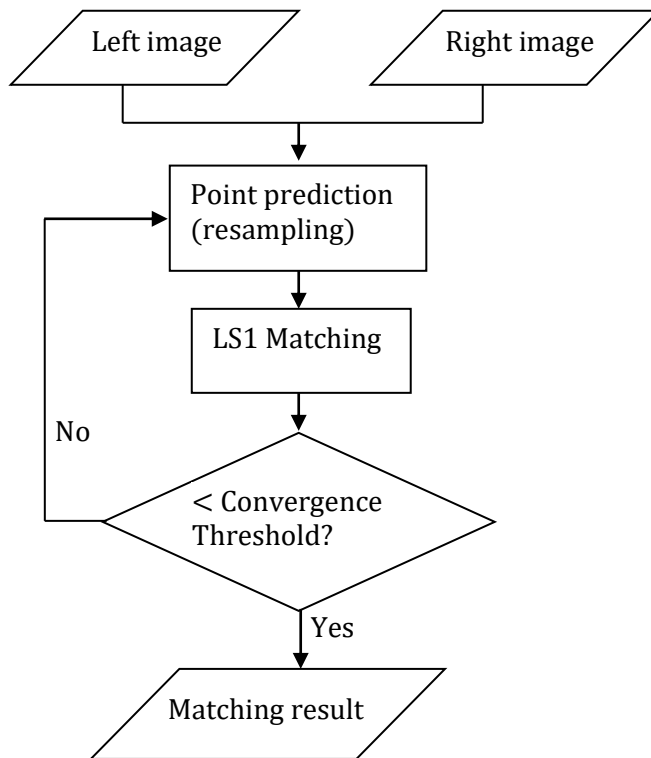digital number from the nearest input pixel center to the output coordinates.

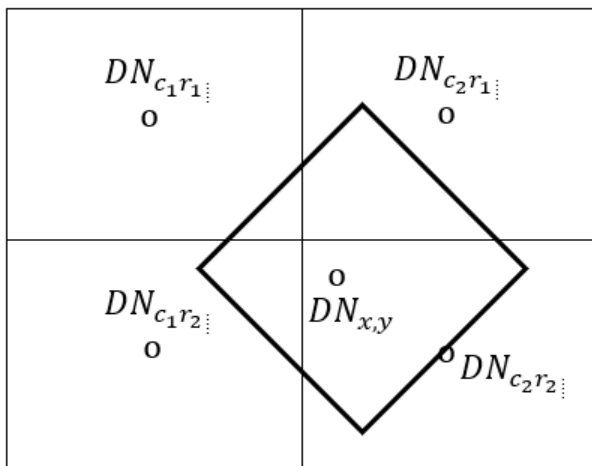Figure 1. Flowchart of least square matching1 process

Figure 2. Four image pixels surrounding the location of an output pixel, resulting from the transformation of ground (for map) coordinates (X, Y) back to image column and row coordinates (x, y)

In Figure 2, the output pixel would be assigned the value of $DN_{2,2}$ because the center of the input pixel at the location is closest to the desired output pixel coordinates. Implementing nearest neighbor resampling

can be using the equation below to compute the image row and column coordinates for the desired output pixel location and then rounding to the nearest whole row and column number.

$$x = a_0 + a_1X + a_2Y$$
$$y = b_0 + b_1X + b_2Y \qquad (4)$$
Where,

(x, y) = image coordinates (column, row)
(X, Y) = ground (or map) coordinates
a_0, a_1, a_2, b_0, b_1, b_2 = transformation parameters

For example, in Figure 2, the computed coordinates for the output pixel location are (1.67, 1.58), which round to (2, 2).

2. The iteration will obtain Δx and Δy. If $\Delta x_{n-1} = \Delta x_n$ so the iteration will stop, so the result will converge. This rule is used to stop for Δy. Δx and Δy are correction to the approximations and adding the corrections to the approximation.

3. In this experiment, it's used several window size, scale, and rotation angle. The iteration for each type will stop if the result reach converge. So, it can be concluded that if the result is convergence the image will match.

The analysis for accuracy can be presented from the RMSE value of each type.

EXPERIMENTAL RESULT

RMSE of several rotation angles

1. The best accuracy of x direction is reached when the rotation angle is 9 degree, then those of y direction is reached when the rotation angle is 3 degree. So, it means that to get the closest matching between the 2 images, the image should be rotated 9 degree for x direction and 3 degree for y direction.

2. The worst accuracy of both x and y directions is occurred when the rotation angle is so big, the orientation is fully changed angle is 30 degree. Because the rotation. Then the result will be worse and the images will not match each other.

3. The accuracy of x direction is getting worse when the rotation angle is more than 15 degree. Then the accuracy of y direction is

getting worse when the rotation angle is more than 9 degree.

4. To get the better result in matching image and better accuracy, the orthorectified image should be used as base image to do the rotation.

Table 1. RMSE (pixel) of several rotation angle

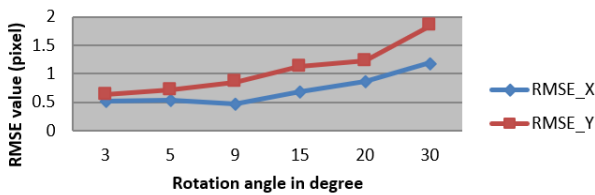|	Rotation angle (in degree)					
	3	5	9	15	20	30
RMSE_X	0.511962	0.527624	0.47013	0.67712	0.8623	1.1883
RMSE_Y	0.633187	0.712135	0.85254	1.12016	1.2274	1.8413

Figure 3. Graphic of RMSE of several rotation angles

RMSE of several scales

1. The best accuracy of both x and y directions when the scale of image is 1. Because there is no change of 2 images. It means that the two images are in same scale.

2. The worst accuracy of both x and y directions when the scale of images is 0.4. When the image's scale 0.4, it's too small. The original image will generalize from the scale 1 to scale 0.4. The DN value changes and gives effect for resampling. Even if it use the nearest neighbor, it will create the wrong output result. It will yield not matching images because the DN values involved are different with the original ones.
Then many information contents are loss because of generalization.

3. Comparing scale 0.4 and 1.6, the scale 1.6 creates a better accuracy than those of scale 0.4. When it uses scale 1.6, DN value change but still preserve the trademark and originality. It's different with the scale 0.4 that it changes more because of generalization.

4. The accuracy of both x and y directions are getting worse when the scale of image less than 0.8.

Table 2. RMSE (pixel) of several scale

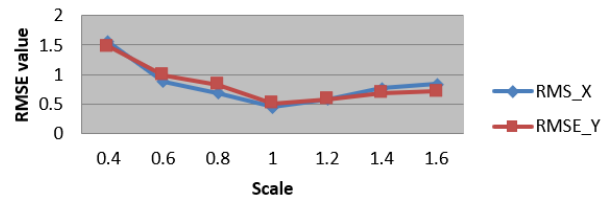|	Scale value						
	0.4	0.6	0.8	1	1.2	1.4	1.6
RMSE_X	1.551214	0.879544	0.67859	0.44876	0.5864	0.7684	0.837614
RMSE_Y	1.477182	0.983817	0.83172	0.51234	0.5858	0.6761	0.711672

Figure 4. Graphic of RMSE of several scales

Success rate

1. Almost using all number of window size, we can reach the success rate 100, except window size 51 and 101. Because the window size is so large, the DN value involved the matching process is many. The probability of matching reduces than success rate will be lower.

2. The success rate will be low when we using the window size more than 41. The window size less than 41 reach the 100% success rate. It means that the window size 3 to 41 has the DN value which preserves the originality. It can be concluded that the two images can match each other exactly.
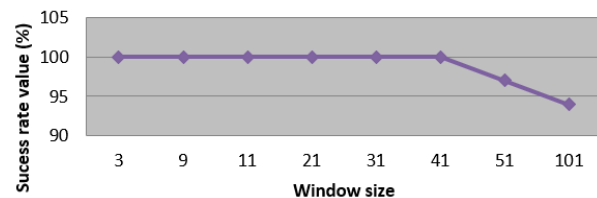

Figure 5. Success rate vs window size

Number of iterations

1. Using window size 3 to 31, the number of iterations to reach the converging result is only 3 times. Using window size 41 and 51, the number of iteration which is needed to reach the converging result is 4 times. Then, using window size 101, the number of iterations which is needed increase sharply is 10 times.

2. When it uses window size from 3 to 31, the area of subarray is small enough. The pixel number involved the matching process is few, the number of DN for resampling process is few. The matching process didn't need more computational to complete the process.
3. When it uses window size 41 and 51, the area of subarray is bigger. It will be involved more number of pixel and DN. The matching process needs more time of computation to complete until reach the converging result.
4. Then, using window size 101 the area of subarray is too large. Many pixels and DN must be involved to the matching process. The process must consider many DN. It makes the process is longer than another and of course giving effect to the computational time. The number of iteration which is needed to reach the converging result increase.
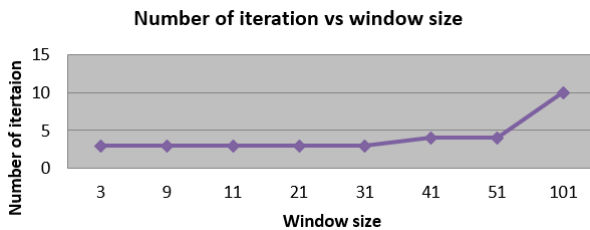

Figure 6. Number of iteration vs window size

RMSE of several window size

1. The best accuracy of x direction is reached when we used the window size 3, then those of y direction is reached when we used the window size 11. There is different window size which is needed in x direction and y direction to reach best accuracy. It presents that the DN of x direction is more various than those of y direction. Then, it should use the small window size to match image in x direction. But it can use larger window size to do matching process in y direction.
2. The accuracy of x direction is good enough (< 1 pixel) using the window size from 3 to 51. So, the DN involved in the area of subarray is close each other. Then the different of accuracy is not significant. However, using the window size 101 the RMSE_X is worst. There is too large area of subarray for matching process. Then the number of pixel is too many that are involved in processing.

3. The accuracy of y direction is good enough (< 1 pixel) using the window size from 3 to 41. The different of accuracy is not significant. However, using the window size 51 and 101 the RMSE_X are high. So that even if we can achieve the good result using window size 11 (number 1), but we should not use too big window size for subarray.
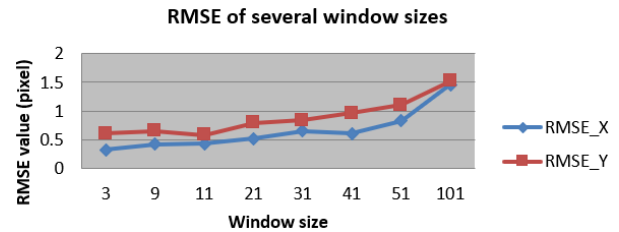

Figure 7. RMSE of several window sizes

CONCLUSIONS

The conclusions that we can present from this experiment are:
1. Using different rotation and scale can get the different result that it will be worse or better. So, to get the better result in matching image and better accuracy, it should use the orthorectified image as base image to do the rotation.
2. The experimental results show that the ideal window size for the matching is in range 11 until 30.

REFERENCES

Chen, L. C., 2009, The handbook of Digital Photogrammetry course, Center for Space and Remote Sensing Research, National Central University.

Gruen, A., 1985a., Adaptive least squares correlation: a powerful image matching technique. South African Journal of Photog., Remote Sensing and Cartography, 14(3), pp. 175-187.

Masuda, T., 2002. Registration and integration of multiple range images by matching signed distances fields for object shape modeling. Computer Vision and Image Understanding, 87 (1-3), pp. 51-65

Mikhail, E., 1976. Observation and least squares. New York, NY University Press, New York.

Mikhail, E., Bethel, J., and McGlone, J., 2001. Introduction to modern photogrammetry. Join Wiley & Sons Inc., New York.

Wolf, R. P., Dewitt, A., B., 2000, Elements of Photogrammetry with Application in GIS, The University of Wisconsin, Madison

Lillesand, T. M., Kiefer, R. W., Chipman, J.W., 2004, Remote Sensing and Image Interpretation. Wiley : USA