

NON-LINEAR SCALE AND ORIENTATION FREE CORRELATION MATCHING ALGORITHM BASED ON EDGE CORRESPONDENCE

Xiunguang Zhou*, Egon Dorrer**

*Environmental System Research Institute (ESRI) Inc.
Redlands, Ca 92373, USA

xzhou@esri.com

**Munich Bundeswehr University
Institute for Photogrammetry and Cartography
D-85577 Neubiberg, Germany
egon.dorrer@unibw-muenchen.de

KEY WORDS: Photogrammetry, Remote Sensing, Image Matching, Image Stereo, DEM Generation

ABSTRACT

A novel image matching algorithm is presented in this paper. One significant property of this algorithm is: free of large non-linear scale difference and orientation difference between left and right. The algorithm tries to match every pixel in the image pair to provide a very detailed matched point set for generating a very good quality DEM. The matching approach, which combines feature based and area based matching methods, consists sub-algorithms of edge extraction, edge description, edge matching, edge geometry constraint cross-correlation matching and Least Square Matching. The well-known From-Coarse-to-Fine matching approach is adopted. A Wavelet Transform was applied to generate image pyramids, extract edges and improve the cross-correlation.

1 INTRODUCTION

Stereo image matching is the key problem of automatic DEM generation. It is also one of the basic objectives of computer vision, photogrammetry and remote sensing. A large number of digital image matching algorithms have been proposed in digital photogrammetry and its related areas. The matching algorithms are usually classified into three classes:

- Signal matching (area based image matching);
- Feature matching (attribute based image matching);
- Combination of signal and feature matching.

In signal matching (area based matching), intensity values of the pixels in the selected window are taken into account in order to measure the disparities between two overlapping images. The well known cross-correlation algorithm takes the minimized disparity of the search area as the corresponding match pairs by measuring the maximization of correlation coefficients (Barnea and Silverman, 1972; Hannah, 1989). Another well-known technique Least Squares Matching (LSM) attempts to match windows of pixels by minimizing the differences of their gray values (Förstner, 1982; Ackermann, 1984; Grün, 1985; Rosenholm, 1986).

In feature matching (attribute based image matching), the predefined common features or attributes (normally are points, lines and areas) are detected in the conjugate windows. Similarity check and additional techniques, such as relaxation or robust statistics and dynamic programming, are employed for matching (Förstner, 1986; Papanikolaou and Derenyi, 1988; Ackermann and Hahn, 1991; Cucka and Rosenfeld, 1992; Murtagh, 1992). Also graph theory is used to describe the relations of the features in matching (Shapiro and Haralick, 1987).

Both area based and feature based matching have their advantages and disadvantages. Though LSM takes the radiometric and geometric differences into account to obtain very accurate correlation, it also requires quite accurate approximations for the corresponding image patches (e. g., to be accurate within 1–2 pixels). Usually, it is not easy to meet such requirements. Cross-correlation is an often used algorithm to get these initial approximations. Unfortunately, cross-correlation does not work properly in case where the stereo images contain larger geometric differences (i. c., large and nonlinear scale difference or large and nonlinear orientation difference or both). In order to handle the

geometry distortion, an algorithm was proposed by the authors (X.Zhou and E.Dorrer, 1994) to include orientation and scale in the cross-correlation algorithm. That is a time consuming algorithm.

Though the feature based matching is free of geometric distortion, only a small number of points in images are belong to features. The matched points in the image are quite sparse. Attempts were made to combine the feature based and area based matching. In signal and feature combined matching algorithm, both gray values and features are used (Otto and Chau, 1989; Zong, Li and Schenk, 1992). The algorithm that proposed in this paper is also such a kind of matching algorithm that makes use of feature and area based matching together.

A general problem in generating a DEM is that the obtained DEM loses a lot of topological details. This is because there are not enough matched points. Too many DEM points were actually obtained by interpolation using far neighbors. In order to get very detailed DEM, this algorithm tries to match every pixel in the image pair to provide a set of very detailed matched points for the DEM generation. In the following section, a general description of the algorithm is presented.

2 GENERAL DESCRIPTION OF THE MATCHING ALGORITHM

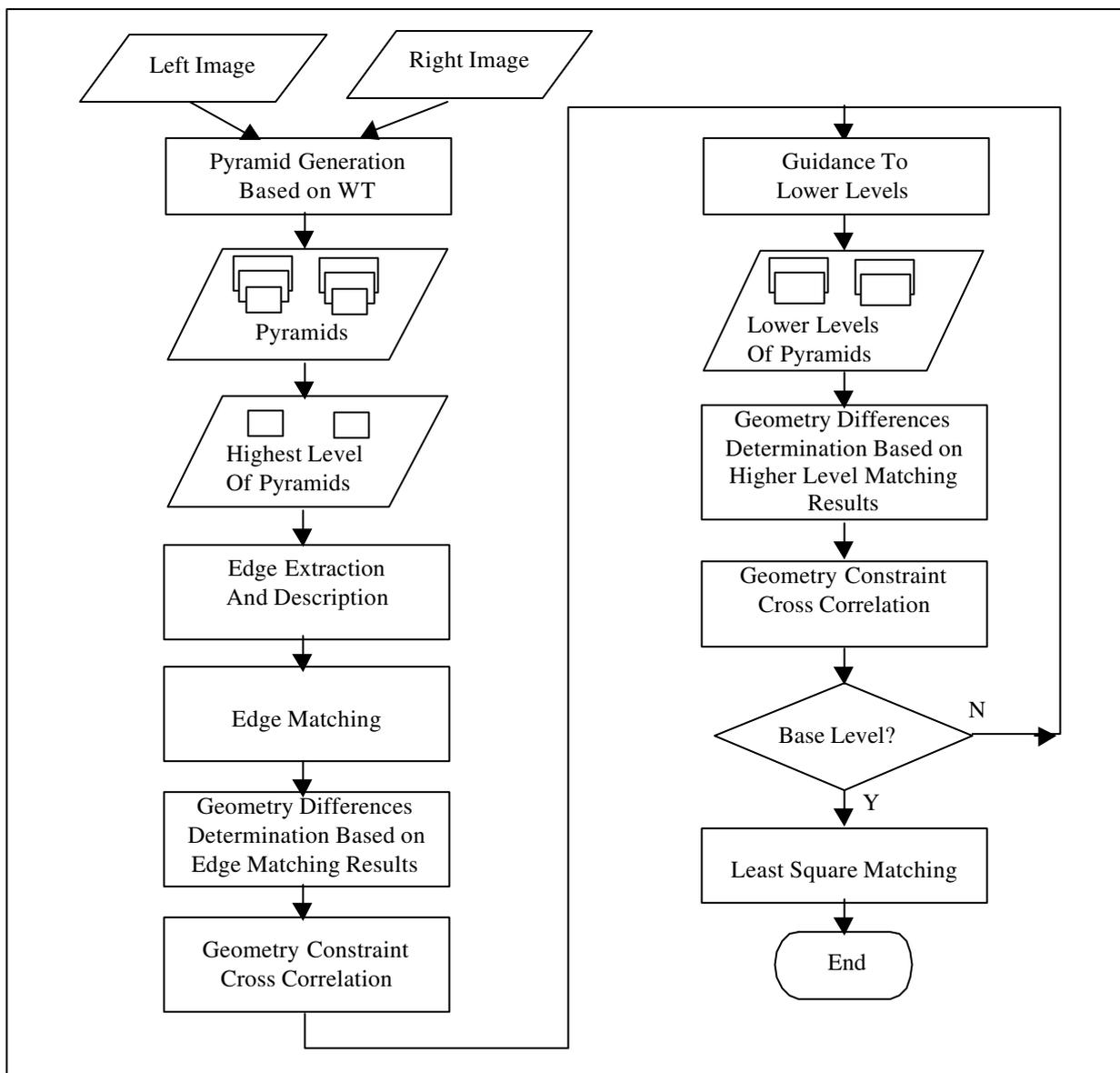


Fig. 1 Diagram of the algorithm

The goal of image stereo is to measure the coordinate disparities of the pixels in the stereo image pair (the left image and the right image). It needs to determine the conjugate point pairs in the image pair. The two points in a conjugate pair must represent the same ground point. Usually, due to the different perspective views and different characteristics,

the appearances of a ground point in the left and right images are quite different. For example, a point that appeared in the left image may not appear in the right image or vice versa, or the geometric aspect of a point in the left image may be different from that in the right image, such as orientation difference and scale difference. As is known, area based matching only works in case that there is not too much geometric distortion in the image pair. In order to handle the geometry difference problem, the feature based matching was selected for the first matching step to determine the local geometry differences. A geometry constraint cross correlation matching was then selected as the second matching step to get the pixel accurate matching results. Finally, we used the well-known Least Square matching to obtain the sub-pixel matching accuracy.

This matching algorithm consists of the following sub-algorithms:

- Wavelet Transform based image pyramid generation;
- Edge extracting sub-algorithm based on Wavelet Transform;
- Edge describing sub-algorithm;
- Edge matching sub-algorithm;
- Edge geometry constraint cross-correlation matching sub-algorithm;
- Least Square Matching sub-algorithm.

A diagram of the algorithm is shown in Figure 1. The left image pyramid and right image pyramid are first generated based on a wavelet transform. The algorithm can be subdivided into two main parts. One is in the left side of Fig. 1. The other is in the right side. The left side is carried out at the highest level of the image pyramid, while the right side is carried out at lower levels of the image pyramid.

At the highest level, edge extraction and edge matching were taken place alternately. Edges were extracted and described one by one in the left image. After each edge in the left image was obtained, the conjugate edge in the right image was determined by using the description of the left edge as an indicator. By using the conjugate edge pairs, the local geometric differences were determined. A geometric constraint cross correlation sub-algorithm was finally applied at the highest level to obtain the disparity of each point pair. The sub-algorithms performed at the highest level will be described in section 4, 5 and 6.

At the lower levels, matching was carried out through all levels except the highest level in the pyramids. The matching at the lower level was guided by the results obtained at the higher level. At each level, the local geometry difference was first calculated by using the results of cross correlation at the higher level. Then the obtained geometry was used in the geometry constraint cross correlation sub-algorithm. After these processes were done at the base level (or, level 0), we had the coarse matching results in pixel accuracy. In order to obtain the sub-pixel accuracy, the Least Square Matching was carried out to complete the matching algorithm.

3 IMAGE PYRAMID GENERATION BY USING WAVELET TRANSFORM

The Wavelet Transform (WT), which was developed by the authors (X.Zhou and E.Dorrer, 1994) was employed in building the image pyramids for both left and right images. Due to the limitation of the text space, in this paper we do not describe the WT transform algorithm.

One reason that we selected the WT algorithm is that it is efficient in storage space and in running speed. Another reason is that the four sub-bands of WT (the approximate band, the horizontal edge band, the vertical edge band and the diagonal edge band) at each level provide very good information for edge extraction and cross correlation. An example of image WT is shown in Figure 2.

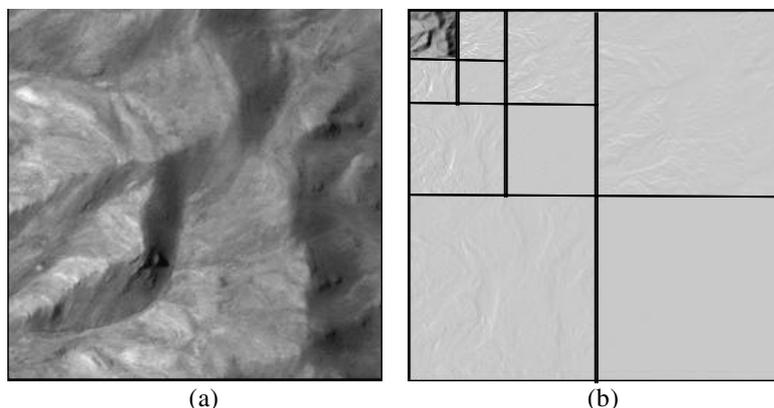


Fig. 2 An example of image WT.
(a) The original image
(b) Transformed image.

4 EDGE EXTRACTION

It can be proved that if a 2X2 average filter is applied to the original image and the approximate sub-band of each level of the pyramid, the Sobel edge can be directly obtained from the horizontal edge sub-band and vertical edge sub-band (X.Zhou and E.Dorrer, 1996). Let $D_{2^j}^h \bar{f}(x, y)$ and $D_{2^j}^v \bar{f}(x, y)$ be the horizontal edge sub-band and vertical edge sub-band at level j (or, at resolution 2^j), respectively, the Sobel value $g_{2^{j+1}}(2x+1, 2y+1)$ at level $j+1$ is

$$g_{2^{j+1}}(2x+1, 2y+1) = \frac{1}{2} \sqrt{D_{2^j}^h \bar{f}(x, y) + D_{2^j}^v \bar{f}(x, y)} \tag{1}$$

A Sobel edge image can be simply generated by using Eq. 1. Figure 3.(a)-(3) shows such a Sobel edge image generated at level 2 of the image WT pyramid

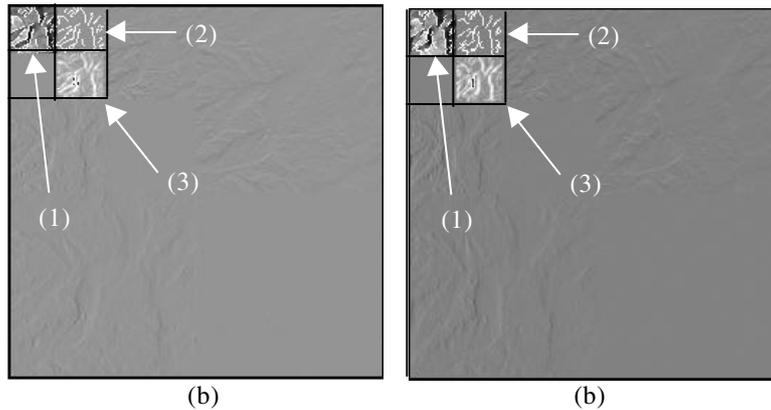


Fig. 3 An example of edge extraction and edge matching
 (a) Left image. (b) Right image.

- (1) Extracted edges superimposed on the approximate band at level 2.
- (2) Extracted edges at level 2.
- (3) Sobel edge image at level 2.

The edge extraction procedure was carried out on the highest level of the edge pyramid. Edges were extracted by following the local maximum Sobel value. The extracted edge points were recorded using the Freeman Direction Code. The Freeman Direction \vec{d}_i is shown in Figure 4. The present edge direction was used to guide the search for the next point. Let \mathbf{p}_i be i th point on the edge and $g(\mathbf{p}_i)$ be the Sobel value at point \mathbf{p}_i , we have the following equation for edge extraction.

$$\mathbf{p}_{i+1} = (x, y)_{i+1} = (x, y) \left| \text{Max}_{k \in \mathbf{s}(\vec{d}_i)} (g(\mathbf{p}_i + \Delta \mathbf{p}(\vec{d}_k))) \right. \tag{2}$$

where $\mathbf{s}(\vec{d}_i)$ is a set of searching directions guided by previous edge direction \vec{d}_i that

$$\mathbf{s}(\vec{d}_i) = \{ \vec{d}_j \mid j = i - 2, \dots, i + 2 \} \tag{3}$$

and $\Delta \mathbf{p}(\vec{d}_k)$ is the x and y offset from one edge point to its next point along the direction \vec{d}_k , that

$$\Delta \mathbf{p}(\vec{d}_k) = \begin{cases} (1, 0) & \text{for } \vec{d}_k = \vec{d}_0 \\ (1, 1) & \text{for } \vec{d}_k = \vec{d}_1 \\ (0, 1) & \text{for } \vec{d}_k = \vec{d}_2 \\ (-1, 1) & \text{for } \vec{d}_k = \vec{d}_3 \\ (-1, 0) & \text{for } \vec{d}_k = \vec{d}_4 \\ (-1, -1) & \text{for } \vec{d}_k = \vec{d}_5 \\ (0, -1) & \text{for } \vec{d}_k = \vec{d}_6 \\ (1, -1) & \text{for } \vec{d}_k = \vec{d}_7 \end{cases} \tag{4}$$

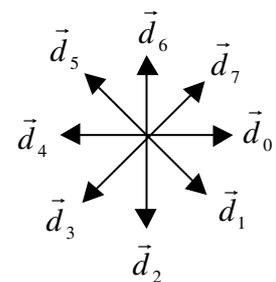


Fig. 4 Freeman

In equation 3, obviously, five directions were selected for the search of next point. For example, if the previous edge direction is \vec{d}_0 , \vec{d}_k for $k=0, 1, 2, 6, 7$ are the search directions. The following criteria were used in the edge extraction sub-algorithm:

- 1) The extracted edge must not be the neighbor of the other edges;
- 2) The length of an edge must be longer than a predefined threshold;
- 3) The procedure of tracing an edge stops when the edge meets the border of the image.

This edge extraction algorithm was only applied in the left image. The edge extraction in the right image is different than that in the left image. The extracted edge in the left image was used as a guide for the edge extraction in the right image. We will describe this in section 5. Figure 3 (a)-(2) shows an example of edges extracted using this edge extraction sub-algorithm. Figure 3 (a)-(1) shows the extracted edges superimposed on the approximate band at level 2 of the image pyramid.

5 EDGE DESCRIPTION AND EDGE MATCHING

5.1 Edge Description

An extracted edge is just a vector data. It is difficult to match the edges directly from the vector data. An efficient edge descriptor is necessary for the edge matching. Our edge descriptor was designed as following:

- 1) The length (number of the edge points) of the edge n
- 2) The coordinate of starting edge point \mathbf{ps}
- 3) The coordinate of ending edge point \mathbf{pe}
- 4) The coordinate of inflection edge point \mathbf{pi}_i having larger curvature v_i , denoted by $\{\mathbf{pi}_i, v_i\}$

Let **DES** be the edge descriptor, we have

$$\mathbf{DES} = \{ n, \mathbf{ps}, \mathbf{pe}, \{\mathbf{pi}_i, v_i\} \} \tag{5}$$

The edge descriptor was basically constructed by the feature points on the edge. The feature points consist of the starting point, the ending point and inflection points. Obviously, it is easy to get n , \mathbf{ps} and \mathbf{pe} directly from the extracted edge vector data. An edge point \mathbf{p}_i was selected as an inflection point \mathbf{pi}_i if its curvature v_i is larger than a predefined threshold T , i.e., $v_i \geq T$.

5.2 Edge Matching

The edge matching sub-algorithm is to determine the conjugate edges in the left image and the right image as well as each conjugate point pair on the conjugate edge pair. Given an extracted edge in the left image, the matching procedure contains three parts: right edge extracting (guided by the left edge), feature points matching and other points matching.

For the feature points matching, a “rotation and scale free cross-correlation” algorithm (X.Zhou and E.Dorrer, 1994) is employed. An n by n patch centered at the feature point is used as a template. Geometric affine transforms of different rotations and scales were applied to the template to take the geometric difference into account. The transformed template was then used in the cross-correlation to find the conjugate feature points in the right image. That is

$$(x, y)^r = (x, y)^l \left| \begin{array}{l} \mathbf{Max} \\ \mathbf{R}^{lr}((x, y)^l, (x, y)^r, \mathbf{q}, \mathbf{r}) \end{array} \right. \tag{6}$$

where $\mathbf{R}^{lr}((x, y)^l, (x, y)^r, \mathbf{q}, \mathbf{r})$ is the geometry transform involved cross-correlation. $(x, y)^l$ is the feature point on the left edge, \mathbf{q} and \mathbf{r} are rotation angle and scale, respectively.

Obviously, this is a time consuming algorithm. Fortunately, as the number of feature points at the highest pyramid level is very small and the searching area is also quite small, the calculating speed is acceptable.

After the corresponding feature points had been obtained, the edge extraction in the right image was carried out in the right Sobel image. The edge extraction sub-algorithm, same as that in section 4, was used except the searching direction was guided by the determined conjugate feature points in the right image. Let \mathbf{fp}_i and \mathbf{fp}_{i+1} be two feature points, the edge points among these two feature points, $\mathbf{p}_{1\dots n}$, were detected one by one from \mathbf{fp}_i to \mathbf{fp}_{i+1} . The search for the first point \mathbf{p}_1 was guided by the direction from \mathbf{fp}_i to \mathbf{fp}_{i+1} , The search for the second point \mathbf{p}_2 was guided by the direction from \mathbf{p}_1 to \mathbf{fp}_{i+1} , and so on. Let \mathbf{p}_k be one point in $\mathbf{p}_{1\dots n}$ and \vec{d}_k be the direction from \mathbf{p}_k to \mathbf{fp}_{i+1} , the set of searching directions $\mathbf{s}(\vec{d}_k)$ in equation 2 is

$$\mathbf{s}(\vec{d}_k) = \{\vec{d}_j | j = k - 2, \dots, k + 2\} \quad (7)$$

Figure 5 demonstrates the search procedure. In figure 5, \vec{d}_0 , \vec{d}_1 , \vec{d}_2 and \vec{d}_3 are the directions from \mathbf{fp}_i^r , \mathbf{p}_1 , \mathbf{p}_2 and \mathbf{p}_3 to \mathbf{fp}_{i+1}^r , respectively. The dashed lines are the tracing paths.

Now we have conjugate feature point pairs $(\mathbf{fp}_i^l, \mathbf{fp}_i^r)$, $(\mathbf{fp}_{i+1}^l, \mathbf{fp}_{i+1}^r)$. We also have the extracted points \mathbf{p}_k^l , for $k=1 \sim m$, between \mathbf{fp}_i^l and \mathbf{fp}_{i+1}^l , and the extracted points \mathbf{p}_k^r , for $k=1 \sim n$, between \mathbf{fp}_i^r and \mathbf{fp}_{i+1}^r . Note that the number of extracted points between two feature points on the left edge may be different with that on the right image edge. I.e., m may not equal to n . For these non-feature edge points, it is reasonable that a linear segmentation is used to determine their correspondence. That is

$$\mathbf{p}_k^l \Leftrightarrow \mathbf{p}_j^r | j = \mathbf{Round} \left(\frac{k}{m} \bullet n \right) \quad \text{for } k=1 \sim m \quad (8)$$

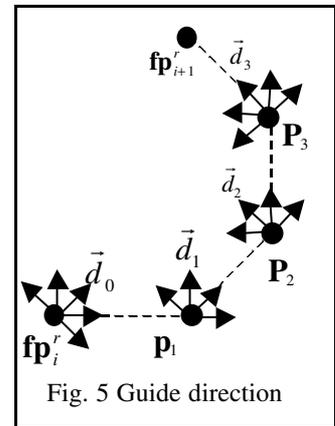


Fig. 5 Guide direction

The above procedure was applied to each two-feature-points (including the starting point and the ending point) to obtain all conjugate edge points on the conjugate edge pair. Figure 3 (a) and (b) show the matched edges.

6 GEOMETRY CONSTRAINT CROSS CORRELATION

The edge matching sub-algorithm only provides conjugate point pairs on edges. These matched points are only a small percentage of the whole image. In order to get conjugate point pairs that are not on edges, a sub-algorithm named Geometry Constraint Cross-correlation was applied. Cross-correlation is a simple matching algorithm. It only works in case that there is no geometry difference between the left and right images. In most remote sensing stereo images, geometry difference is the common case. We modified the traditional cross-correlation by introducing a local affine transform. The coefficients of the affine transform can be derived from the results of the edge matching sub-algorithm.

6.1 Local Geometry Correspondence Based On Edge Correspondence

The local geometry correspondence of the stereo images is described by six affine transform coefficients. Though the geometry difference is in general non-linear in the range of whole scene, it can be approximated to be linear if the local geometry difference is concerned. Just like an arbitrary curve can be approximated by piece wise lines, the general non-linear geometry difference can be described by all linear local differences. The local geometry correspondences of all matched edge points were first determined by using the edge correspondences. The Least Square Fitting was used to determine the affine transform coefficients. Let \mathbf{p}^l and \mathbf{p}^r be a conjugate edge point pair; let \mathbf{S}_{p^l} and \mathbf{S}_{p^r} be two edge point sets, each contains the edge point (\mathbf{p}^l or \mathbf{p}^r) as well as their neighbor edge points; let \mathbf{C}_p be local geometry affine transform coefficients matrix, we have

$$\mathbf{C}_p = (\mathbf{A}_p^T \cdot \mathbf{A}_p)^{-1} \cdot \mathbf{A}_p^T \cdot \mathbf{B}_p \quad (9)$$

where $\mathbf{A}_p = [\mathbf{1}, \mathbf{S}_{p^l}]$ is the coefficient matrix combining a unit vector $\mathbf{1}$ and the left edge point set \mathbf{S}_{p^l} , $\mathbf{B}_p = [\mathbf{1}, \mathbf{S}_{p^r}]$ is the observation matrix combining a unit vector $\mathbf{1}$ and the right edge point set \mathbf{S}_{p^r} .

In order to let \mathbf{S}_{p^l} and \mathbf{S}_{p^r} correctly represent the local geometry correspondence, the edge points in \mathbf{S}_{p^l} should have a good distribution surrounding \mathbf{p}^l . The best distribution is that there are edge points located in the upper-left, upper-right, lower-left and lower-right quarters of the location of \mathbf{p}^l . The selection of \mathbf{S}_{p^l} was automatically performed by scanning the neighbors of \mathbf{p}^l .

The local geometry correspondences of non-edge-points were obtained by interpolating the determined geometry correspondences on edge points. Therefore we had the local geometry correspondence at every point in the stereo images.

6.2 Geometry Constraint Cross Correlation

The traditional cross-correlation is to determine the similarity of two points by measuring the similarities pixel by pixel in two rectangular windows that centered at the two points. It is given in Equation 10.

$$R^{l,r} = \frac{\sum_{i=1}^m \sum_{j=1}^n (g^l(i, j) - \bar{g}^l)(g^r(i, j) - \bar{g}^r)}{\sqrt{\sum_{i=1}^m \sum_{j=1}^n (g^l(i, j) - \bar{g}^l)^2 \cdot \sum_{i=1}^m \sum_{j=1}^n (g^r(i, j) - \bar{g}^r)^2}} \quad (10)$$

where m and n give the rectangular window size, $g^l(i, j)$ and $g^r(i, j)$ are gray values in the left window and right window, respectively, \bar{g}^l and \bar{g}^r are the mean gray values of the two windows.

Equation 10 presumes that the geometry in the two windows is identical. Obviously, if the two points have different geometry properties (different orientation and/or different scale), we can not measure the similarity by two identical rectangular windows. In order to get a correct measuring result, we introduced a geometry constraint in the traditional cross correlation. That is

$$R^{l,r} = \frac{\sum_{i=1}^m \sum_{j=1}^n (g^l(T(i, j)) - \bar{g}_T^l)(g^r(i, j) - \bar{g}^r)}{\sqrt{\sum_{i=1}^m \sum_{j=1}^n (g^l(T(i, j)) - \bar{g}_T^l)^2 \cdot \sum_{i=1}^m \sum_{j=1}^n (g^r(i, j) - \bar{g}^r)^2}} \quad (11)$$

where $T(i, j) = C_p \cdot [i, j, 1]^T$ is the new coordinates of (i, j) after an affine transform, \bar{g}_T^l is the mean gray value in the transformed window.

Obviously, equation 11 takes geometry differences into account to measure the disparity of the two points. Equation 11 can be equivalently explained in the way that a geometric transform is applied to the left window before measuring the disparity. This will guarantee that the disparity measure is performed under the same geometry.

7 MATCHING AT LOWER LEVELS OF THE PYRAMIDS

The sub-algorithms given in above sections were only carried out at the highest level of the image pyramid. At levels other than the highest level, two matching procedures were performed. They are 1) Iteration of geometry constraint cross-correlation at lower levels; 2) Least Square matching at base level.

7.1 Iteration Of Geometry Constraint Cross-Correlation At Lower Levels

As the image size at the highest level is quite small, it is not a big performance issue if a time consuming algorithm is carried out on such a small image. When the matching goes through different levels of the image pyramids, the image size gets larger as the level of the pyramid gets lower. Therefore, the computational costs become a critical issue. Fortunately, we already had the matching results at the higher level. Alternatively say, we had the pre-knowledge about the matching at lower levels. The pre-knowledge can be used as the matching guide at the present level. This greatly simplifies the matching process. Two items of the pre-knowledge can be derived from the higher level. One is the searching range for conjugate point pairs at the present level. The other is the local geometry correspondence at the present level.

According to the WT algorithm (X. Zhou and E. Dorrer, 1994), a point at the higher level of the pyramid corresponds to four points at the present level. Let $(x, y)_{2^j}^l$ be a point at level j of the left image pyramid, the four corresponding points at level $j+1$ are $(2x, 2y)_{2^{j+1}}^l$, $(2x+1, 2y)_{2^{j+1}}^l$, $(2x, 2y+1)_{2^{j+1}}^l$ and $(2x+1, 2y+1)_{2^{j+1}}^l$. In the same way we have the conjugate point $(x, y)_{2^j}^r$ and its four corresponding points $(2x, 2y)_{2^{j+1}}^r$, $(2x+1, 2y)_{2^{j+1}}^r$, $(2x, 2y+1)_{2^{j+1}}^r$ and $(2x+1, 2y+1)_{2^{j+1}}^r$. The matching task at level $j+1$ is to determine the correspondences of the four points in the left and right images. Though the coordinates between two levels have the above relationship, the correspondence of the left and right points at level $j+1$ can not be simply obtained on this analogy. This means that $(2x, 2y)_{2^{j+1}}^l$ and $(2x, 2y)_{2^{j+1}}^r$ may be not a conjugate pair even if $(x, y)_{2^j}^l$ and $(x, y)_{2^j}^r$ are a conjugate pair. In pixel accuracy, one point in the left image may

correspond to more than one point in the right image. For the matching task at level $j+1$ the same geometry constraint cross-correlation sub-algorithm as described in section 6 was employed. In stead of using the edge correspondence to determine the local geometry correspondence in section 6, the local geometry affine transform coefficient matrix was derived from the coordinates of the matched points at the higher level. For the reliability, we expanded the searching range one pixel more both in x and y direction.

7.2 Least Square Matching

After the matching iteration procedure was finished at the base level of the image pyramid, for each point in the left image we obtained its conjugate point in the right image in pixel accuracy. In the final matching step, the Least Square Matching (LSM) was performed at the base level to get sub-pixel accuracy. As Least Square Matching is a very popular and well known algorithm, it is not necessary to describe the algorithm here. We added a constraint in LSM to guarantee that each matched pair has a logical geometric relationship to its neighbors. The constraint is: if a point is on the left (or, right, upper, lower) side of another point in one image, this characteristic must be kept in the other image. A conflict can easily happen, if the LSM is applied to each pixel in the image. For each output from LSM iteration we checked the logical relationship with its neighbors. If a conflict happened, we stopped the LSM and took the last iteration result as the output. If the check failed at the first iteration, we simply took the initial coordinates as the output of LSM.

8 RESULTS AND CONCLUSION

We used different MOMS stereo images in our experiments. The resolution of MOMS image is 13.5 m / pixel. Figure 7 shows a MOMS stereo image pair that we already used as examples to show the image pyramid and edge extraction. It is clear that there are non-linear geometry differences between the two images. Due to the different perspective view of the left and right image, the scale difference and orientation difference are quite large. Our technique produced very good matching results. The quality of the matching result can be checked by verifying the generated DEM and its shaded-relief. In this paper we only use this check method. Each point in the generated DEM was directly obtained by the forward intersection without any interpolation. This will show how good of each matched point pair is. Compare the shaded-relief image with the original image, you can see what details are obtained in the DEM. Figure 8 shows a DEM that generated in such a way. Figure 9 shows the shaded-relief image. Figure 10 shows the location of the shaded-relief image in the original image.

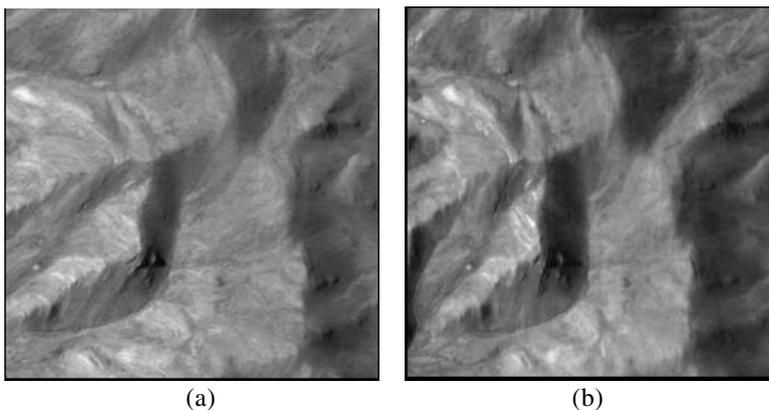


Fig. 7 A MOMS stereo image pair
(a) left image (b) right image



Fig.8 Generated DEM

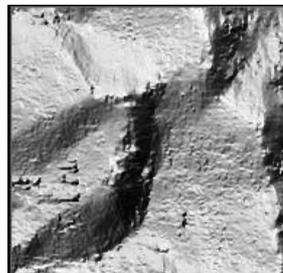


Fig.9 Shaded-relief image

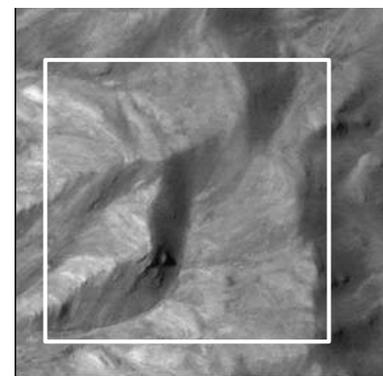


Fig 10 The location of the shaded-relief image.

The shaded-relief image was created using the Lambert photometric function (X. Zhou and E. Dorrer, 1995). Considering that the albedo is assumed constant in the shaded-relief image and the limitation of the Lambert model, the two images are quite similar. It is an overcritical method to verify the quality of the DEM by comparing its shaded-relief with its original image.

There are some errant points in the DEM (in the right area of the DEM), which can be seen in the shaded-relief image. The errors come from unmatched points. Errors such as these can be easily removed if an interpolation is used in the DEM generation like the usual way. An interpolation was not performed in this case to show the high accuracy and detail possible using this technique.

REFERENCES

- Ackermann F., Hahn M., 1991. Image pyramids for digital photogrammetry. *Digital Photogrammetric Systems*, Wichmann, Karlsruhe, pp. 43-58
- Barnea D., Silverman H., 1972. A class of algorithms for fast digital image registration. *IEEE Trans. on Computers*, Vol. C-21, No. 2, pp. 178-186
- Cucka P., Rosenfeld A., 1992. Linear feature compatibility for Pattern-matching relaxation. *Pattern Recognition*, Vol. 25, No. 2, pp. 189-196
- Förstner W., 1982. On the Geometric precision of digital correlation. *Inter. Arc. Photo. Remote Sensing*, 24(3), pp. 176-189
- Förstner W., 1986. A feature based correspondence algorithm for image matching. *Int. Arc. Phot. Remote Sensing*, 26(3), pp. 1-17
- Grün A., 1985. Adaptive least squares correlation: a powerful image matching technique. *S. Afr. J. of Phot., Remote Sensing and Cartography Cartography*, 14(3), pp. 175-187
- Hannah M., 1989. A system for digital stereo image matching. *Phot. Eng. and Remote Sensing*, 55(12), pp. 1765-1770
- Lemons M., 1988. A survey on stereo matching techniques. *Int. Arc. Phot. Remote Sensing*, Vol. 27, Part B 8, pp. v 11-23
- Murtagh F., 1992. A feature-based $O(N^2)$ approach to point matching. *Inter. Con. on Pattern Recognition*, B, pp. 174-177
- Otto G., Chau T., 1989. "Region-growing" algorithm for matching of terrain images. *Image and vision Computing*, Vol. 7, No. 2, pp. 83-94
- Papanikolaou K., Derenyi E., 1988. Structural matching of digital images and terrain models. *ISPRS*, Part B 27(3), pp. 669-678
- Rosenholm D., 1986. Accuracy improvement of digital matching for evaluation of digital terrain models. *Int. Arc. Phot. Remote Sensing*, 26(3), pp. 573-587
- Shapiro L., Haralick R., 1987. Relational matching. *Applied Optics*, Vol. 26, pp. 1845-1851
- Zhou X., Dorrer E., 1994. A non-error reconstruction of multiresolution discrete wavelet representation and its fast algorithm. *SPIE vol. 2242, Wavelet Applications*, pp.236-247
- Zhou X., Dorrer E., 1994. An Automatic Image Matching Algorithm Based on Wavelet Decomposition. *Proceedings of ISPRS, Commission III*, on "Spatial Information from Digital Photogrammetry and Computer Vision", pp.951-960
- Zhou X., Dorrer E., 1995. An Adaptive Algorithm of Shaded-relief Images from DEMs Based on Wavelet Transform", *Proceedings of ISPRS & SPIE*, on "Digital Photogrammetry and Remote Sensing '95", SPIE vol.2646, pp.212-224
- Zhou X., Dorrer E., 1996. De-shading: Integrated Approach to Photometric Model, Surface Shape and Reflectance Properties. *Proceedings of ISPRS Congress*, Vol XXXI, B3, WG II/2, PP.1028-1035
- Zhou X., Dorrer E., 1997. A New Edge Detection Algorithm Based on Wavelet Transform. *Proceedings of Second Turkish-German Joint Geodetic Days*, PP. 727-736
- Zong J., Li J., Schenk T., 1992. Aerial image matching based on zero-crossing. *Int. Arc. Phot. Remote Sensing, Commission III*, pp. 144-150