Universität Stuttgart

Schriftenreihe der Institute des Studiengangs Geodäsie und Geoinformatik

Technical Reports
Department of Geodesy and Geoinformatics

J. Banks, K. Kubik, Y.H. Lu

Investigation into Digital Image Matching

ISSN 0933-2839

Report Nr. 2000.1

Introduction

This volume documents some of the work of the research group of Prof. Kubik on Digital Image Processing and Image Matching. Part of the work was done while Prof. Kubik was under the Alexander von Humboldt Programme guest at the Department of Geodetic Science, University of Stuttgart.

The formation of photographic images of 3D objects is a well-understood process. Its inversion, however, namely the reconstruction of the 3D shape from a series of (stereo) images is a non-trivial task. The problem is under-determined, and usually assumptions of continuity and piecewise smoothness of the object surface are made during reconstruction. This works well in small-scale aerial photogrammetry. In large-scale photogrammetry and terrestrial scenes these assumptions are not justified, which makes that the existing systems for digital photogrammetry either do not work at all or give poor results in these applications.

Our research concentrates on those environments where there are large disparities (parallaxes) in the images. We wish to develop methods, which give accurate 3D reconstructions in these environments, and also to derive accuracy and integrity measures for the operation. This publication gives some of the results of our ongoing research.

The research was is supported by the following grants:

- Australian Research Council Large Grant 1997-99
- Australian Postgraduate Award 1997-1999
- CMTE Postgraduate Award 1997-1999
- Alexander von Humboldt Prize 1999-2000

Chapter 1 deals with robot vision and in particular with automatic collision avoidance in the underground mining environment. Here, we can not assume continuity in the object space, and any disparity is possible. Rank correlation is proposed for image matching due to its favourable properties for hardware implementation in a real time environment. We investigate the integrity of the proposed image matching method for this mining environment, in function of the window size.
It shows that a poor integrity is obtained when no assumptions can be made regarding continuity. Our work on this topic is still ongoing, and chapter 1 should be seen as an encouragement for other researchers to address this problem, possibly along the lines here described.

In Chapter 2 we look at large scale photogrammetric mapping. The before-named problems manifest themselves in poor results for digital elevation models. Houses and trees are not recognised as such and are integrated as bumps into the digital elevation model. Our approach to improve on this situation is to combine results from image segmentation and image matching to recognise houses and trees. These are then either eliminated in the surface reconstruction, or dealt with separately. In that way the continuity assumption could be avoided along the boundaries of these object types.

We hope to combine our different research streams, which also include groundel based matching and matching in compressed space, into a theory for object reconstruction where we clearly understand the underlying assumptions, and hopefully can relax on the current stringent assumptions in image matching. This would enable us to come closer to the very flexible 3D experience of human vision.

Jasmine Banks[1], Kurt Kubik[2] and Yi Hui Lu[2]

1. Fraunhofer Institut for Computer Graphics, Rundeturmstr. 6, D-64283 Darmstadt, Germany, email: jbanks@igd.fhg.de
2. Department of Geographical Sciences and Planning, The University of Queensland, Brisbane Qld 4072, Australia, email: k.kubik@mailbox.uq.edu.au

Chapter 1

Estimating the Probability of an Invalid Match for Matching Using the Rank Transform

Abstract

A fundamental problem faced by stereo matching algorithms is the *matching* or *correspondence* problem. A wide range of algorithms have been proposed for the correspondence problem. For all matching algorithms, it would be useful to be able to compute a measure of the probability of correctness, or *reliability* of a match.

This paper focuses in particular on one class of matching algorithms, which are based on the rank transform. The interest in these algorithms for stereo matching stems from their invariance to radiometric distortion, and their amenability to fast hardware implementation.

This work differs from previous work in that it derives, from first principles, an expression for the probability of a correct match. This method was based on an enumeration of all possible symbols for matching. The theoretical results for disparity error prediction, obtained using this method, were found to agree well with experimental results. However, disadvantages of the technique developed in this chapter are that it is not easily applicable to real images, and also that it is too computationally expensive for practical window sizes. Nevertheless, the exercise provides an interesting and novel analysis of match reliability.

1.1 Introduction

Stereo vision is a method of depth perception, in which depth information is inferred from two (or more) images of a scene, taken from different perspectives. There are a wide range of applications for stereo vision, including aerial photogrammetry[17, 18], autonomous vehicle guidance[15], robotics and industrial automation.

A fundamental problem faced by stereo matching algorithms is the *matching* or *correspondence* problem. This involves establishing correspondences between points in two images. If the parameters of camera orientation are known, it is then possible to perform triangulation and compute the 3D coordinates of the original scene point.

A wide range of algorithms have been proposed for the correspondence problem[8]. Matching algorithms may be categorised according to the primitives used for matching. In *area-based* techniques[1, 9, 11], regions of actual pixel values in two images are compared in order to locate the best match. Matching metrics such as Normalised Cross Correlation (NCC) or Sum of Absolute differences (SAD) may be used to measure the similarity between two pixel regions. In contrast, *feature-based* techniques[10, 16] locate interesting features such as edges, and then compare the symbolic representations of these features to find the best match.

Matching algorithms may furthermore be categorised according to the matching strategy employed. Typically matching algorithms select a point in the first image, and then compute the similarity between this point and all candidate points in the second image. The most basic of all matching strategies would merely select the most similar candidate as the correct match. More sophisticated matching strategies include dynamic programming and relaxation labeling. Dynamic programming [2, 7, 13] is one such technique, where all points along a pair of corresponding epipolar lines are matched at once. A two-dimensional array of similarity scores is generated from every possible point and candidate combination from a pair of corresponding epipolar lines. The match problem is then solved by finding the optimum path through this two-dimensional space. Relaxation algorithms[6, 12, 14], compute the probabilities for an initial set of matches, and then iteratively update these probabilities according to assumptions such as similarity between neighbouring matches. This iterative process should result in a set of optimal matches.

The more sophisticated matching strategies tend to depend on being able to compute a measure of the probability of correctness, or *reliability* of a match. Often this measure of reliability is derived from assumptions made about the images and/or scene topology. For example, it is often assumed that depth varies gradually, therefore neighbouring matches will be similar to each other. In addition, bland areas in the images are unlikely to be correctly matched, therefore these regions may be flagged as having a low match reliability. A number of existing methods for estimating match reliability are outlined in [3].

This paper focuses on computing match reliability for one class of matching algorithms, which use the rank transform[19]. The advantages of the rank transform include its invariance to radiometric distortion and its amenability to fast hardware implementation[4].

The process of matching using the rank transform is outlined in Section 1.2. The notation used and overall motivation behind the direction taken by this paper are then described in Section 1.3. The work presented here differs from previous approaches to estimating match reliability, in that it derives, from first principles, an expression for the probability of a correct match. This derivation was based on an enumeration of all possible symbols for matching. Sections 1.4 and 1.5 describe two separate approaches to the problem of estimating the probability of a particular disparity being returned by the matching process. The first of these proved to be a difficult and laborious process, so was discontinued in favour of the second. The theoretical results of the second approach were compared with the results obtained from test image pairs consisting of uniformly distributed random pixel values. Section 1.6 then uses the results of Section 1.5 to derive the probability of an invalid disparity. In Section 1.7 the predicted disparity errors were compared with the actual disparity errors obtained from matching contrived stereo pairs. Finally, a discussion of these results and conclusion are presented in Section 1.8.

1.2 Matching Using the Rank Transform

The rank transform is a non-parametric transform which has been proposed for the matching problem[19]. To perform the rank transform, a square or rectangular window is passed over the image, and at each image location, the number of pixel values which are less than the centre pixel are counted. The images will therefore be transformed into an array of integers, whose possible values range from 0 to $N-1$, where N is the number of pixels in the window. An example of the rank transform, for a 3×3 pixel window, is as follows:

$$\begin{array}{ccc} 114 & 87 & 42 \\ 96 & 74 & 51 \\ 23 & 18 & 77 \end{array} => 4. \tag{1.1}$$

In this example, there are four pixels in the transform window which are less than the centre pixel, therefore, the rank transform at this location is 4. The possible values from 0 to $N-1$ which comprise the rank transformed images are referred to as *rank symbols*.

The process of matching using the rank transform is illustrated in Figure 1.1. This process accepts an epipolar stereo pair[5] as input and produces a disparity map as output. The images comprising the stereo pair are first of all rank transformed. Epipolar constrained area-based matching is then used to establish correspondences between the images. As shown in Figure 1.2, a template window is selected in one image, and this is compared with a series of candidate windows along the epipolar line in the second image. The two windows are compared using the

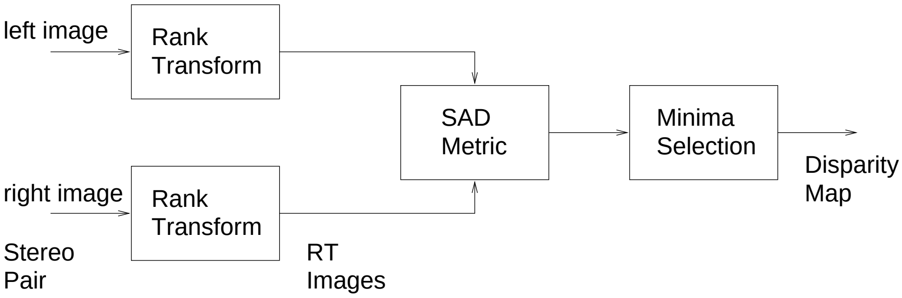


Figure 1.1: Matching using the rank transform.

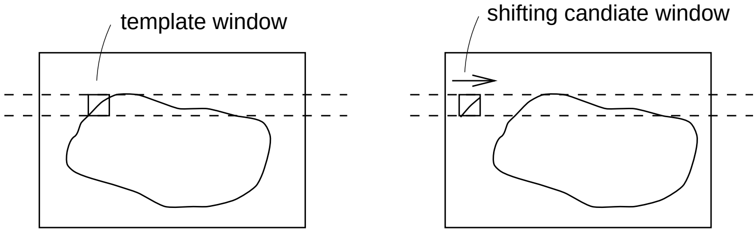


Figure 1.2: Epipolar constrained area-based matching.

Sum of Absolute Differences (SAD) metric, given as follows:

$$SAD(I_1(x,y), I_2(x+d,y)) = \sum_{(m,n)\in M} |I_1(x+m, y+n) - I_2(x+d+m, y+n)| \tag{1.2}$$

where I_1 and I_2 are the two images, d is the test disparity and $(m,n) \in M$ are indices into the match window. The SAD metric will be smaller when the template and candidate windows are more similar, and will be zero if the two windows are identical. The difference in location of corresponding points in two images is called the *disparity.* The disparity having the smallest value of the SAD matching metric is selected as the correct match. The disparity values thus obtained for every location in the images are collectively referred to as the disparity map.

1.3 Problem Formulation

The objective of this work is to derive an expression for the probability of an incorrect match, for matching using the rank transform. In order to derive this expression, the problem is formulated in the following manner. First of all, the images are rank transformed, as described in the previous section. The rank symbols then consist of integers in the range 0 to $N-1$, N being the number of pixels in the rank window.

The rank transformed images are then matched using the SAD metric. The process of epipolar constrained area-based matching is simulated by taking a template window, T, and computing the SAD metric for a series of candidate windows, denoted $C(k)$ in Figure 1.3. The disparity k having the smallest value of SAD metric is then chosen. If more than one disparity value has an equally low SAD score, then the first disparity encountered, ie, the smallest one, is selected.

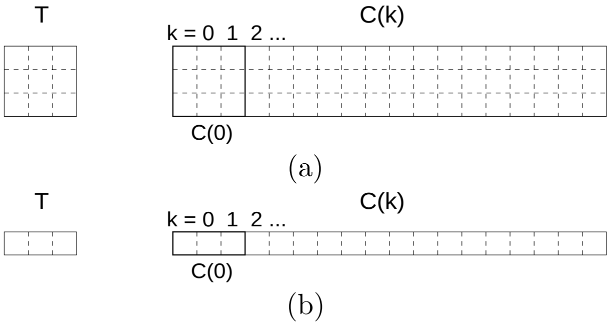


Figure 1.3: Matching a template, T, with a series of candidate windows, $C(k)$. (a) 2-dimensional template window is matched with a series of candidate windows (b) 1-dimensional template window is matched with a line of candidate windows.

Hopefully this will in fact correspond to the correct disparity. However, if there happens to be an arrangement of pixels which results in a smaller SAD score, then an erroneous disparity will result.

The approach taken for this analysis of match errors is to assume the series of candidate windows $C(k)$ consists of random pixel values. In this case, no correct match exists for the template T. Therefore, when the template T is matched with candidate windows $C(k)$, the disparity result is analogous to an erroneous disparity. Subsequent work, therefore, focussed on deriving an expression for the probability of a disparity k being returned by the matching process. This is considered equivalent to the probability of that disparity being returned in error.

1.4 Permutation Analysis Approach

The initial approach taken to this problem was to enumerate every possible permutation of rank symbols in the array of candidate windows $C(k)$. In order to simplify the problem, 1-dimensional rank and match windows of size $1 \times R$ and $1 \times M$ respectively were used. As illustrated in Figure 1.3(b), the matching process compares a 1-dimensional template window, T with a series of candidate windows. The series of candidate windows $C(k)$ in this case comprise a line of rank symbols. The possible rank symbols consist of integers $0 \ldots R - 1$, and it was initially assumed that the rank symbols occur with equal probability. This analysis first seeks to find the probability of each disparity k being chosen.

1.4.1 Single element match window

First of all, a match window consisting of a single element, ie, $M = 1$, was considered. This single element match window is compared with each location k along the line of rank symbols $C(k)$ until a perfect match, ie $C(k) = T$ is found. The location of this match is the disparity value returned by the matching process.

k	0	1	2	3	4	$\cdots$	N
P(k)	$\frac{1}{3}$	$\frac{2}{3^2}$	$\frac{4}{3^3}$	$\frac{8}{3^4}$	$\frac{16}{3^5}$	$\cdots$	$\frac{2^N}{3^{(N+1)}}$

Table 1.1: Probability of a match at disparity k along a line of rank symbols, for $R = 3$, $M = 1$.

k	0	1	2	3	$\cdots$	N
P(k)	$\frac{1}{5}$	$\frac{4}{5^2}$	$\frac{4^2}{5^3}$	$\frac{4^3}{5^4}$	$\cdots$	$\frac{4^N}{5^{(N+1)}}$

Table 1.2: Probability of a match at disparity k along a line of rank symbols, for $R = 5$, $M = 1$.

Figure 1.4 illustrates this concept for a template window $T = 0$. Possible permutations of a line of rank symbols have been enumerated, and displayed in a tree-like structure. The template window, in this case $T = 0$ is compared with every location in the line of rank symbols $C(k)$, starting with $k = 0$, until a perfect match, ie, $C(k) = T = 0$ is encountered. The locations of these matches have been highlighted in Figure 1.4. Only the first location of a match is considered, therefore, possible permutations of the line of rank symbols which occur after the match have been shown with dashed lines in Figure 1.4. This is because the process of area-based matching returns the first match encountered, ie, that with the lowest disparity, when there is more than one match with the equally lowest SAD score.

Given that all rank symbols are assumed to occur with equal probability, it is possible to use Figure 1.4 to compute the probability of a match at each disparity k, An expression for the probability of each disparity can then be inferred. These results are shown in Table 1.1.

A similar enumeration of possible permutations for a line of rank symbols, was performed for a rank window size of $R = 5$. In this case the probability of a match at each location is given by Table 1.2.

From Tables 1.1 and 1.2, it is possible to infer an expression for the probability that the template $T = 0$ will first occur at position k. This is the probability of a disparity k being returned by the matching process. This expression is as follows:

$$P(k) = \frac{(R-1)^k}{R^{(k+1)}} \tag{1.3}$$

Equation 1.3 gives the probability that a match will occur at a particular disparity k. However, this method was derived using a match window of only a single element, $M = 1$, and would need to be extended to be applicable to larger sized windows.

1.4.2 Multi-element match window

The next step in this analysis was to consider a 1-dimensional match window of size $1 \times M$. A match window and rank window of size $M = 3$ and $R = 3$ respectively were used. Given a template window, $T = \{0, 0, 0\}$, and a line of rank symbols, we wish to find the probability of a perfect match at a certain disparity. Figure 1.5 shows expansions of possible permutations of a

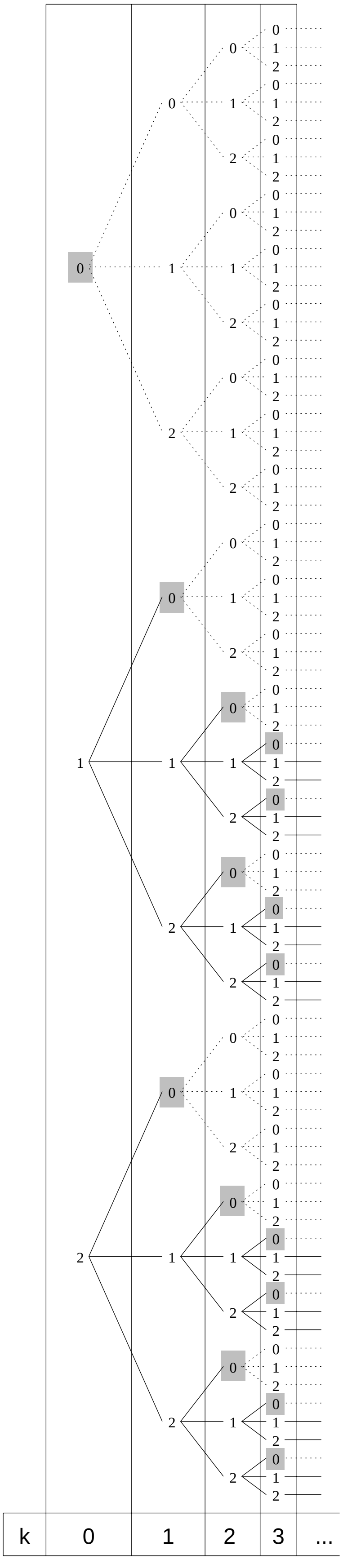


Figure 1.4: Matching the template $T = 0$ to possible permutations of a line of rank symbols.

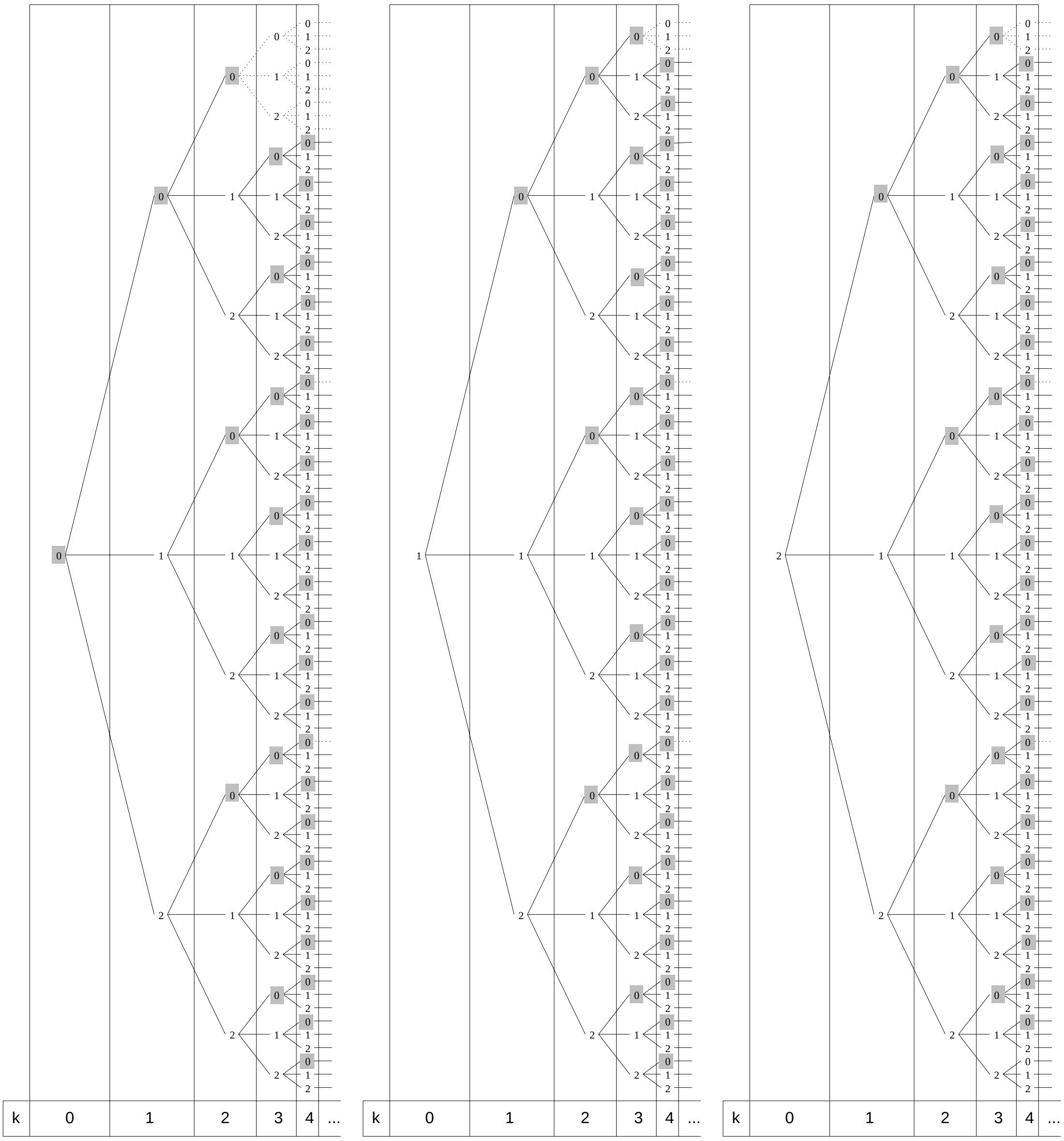


Figure 1.5: Probability of a match at disparity k along a line of rank symbols, for $R = 3$, $M = 3$, and a template window of $T = \{0, 0, 0\}$.

line of rank symbols, and the first occurrence of the pattern $\{0, 0, 0\}$.

Figure 1.5 may be used to deduce the probability of a match at each disparity. The disparity results are given in Table 1.3. Similar experimentation with different values of rank window size R and match window size M leads to the following expression for the probability of a match at disparity, k:

$$P(k) = \frac{(R-1) * (\sum \text{previous M numerators})}{R^{(k+R)}} \tag{1.4}$$

Equation 1.4 gives an expression for computing the probability of matching a template window consisting of M zeros, at each disparity k. However, if a different template window, for example, $T = \{0, 0, 1\}$ is considered, then a different expression for computing the probability of a match will result. Figure 1.6 shows the first occurrence of the template $T = \{0, 0, 1\}$, along a line of

k	0	1	2	3	4	5	$\cdots$	N
P(k)	$\frac{1}{3^3}$	$\frac{2}{3^4}$	$\frac{6}{3^5}$	$\frac{18}{3^6}$	$\frac{52}{3^7}$	$\frac{152}{3^8}$	$\cdots$	$\frac{2*\sum(\text{previous 3 numerators})}{3^{(N+3)}}$

Table 1.3: Probability of a match at disparity k along a line of rank symbols, for $R = 3$ $M = 3$, and a template window of $T = \{0, 0, 0\}$.

k	0	1	2	3	4	$\cdots$	N
P(k)	$\frac{1}{3^3}$	$\frac{3}{3^4}$	$\frac{9}{3^5}$	$\frac{26}{3^6}$	$\frac{75}{3^7}$	$\cdots$	$\frac{3*\text{Numerator}_{k-1}-\text{Numerator}_{k-3}}{3^{(N+3)}}$

Table 1.4: Probability of a match at disparity k along a line of rank symbols, for $R = 3$ $M = 3$, and a template window of $T = \{0, 0, 1\}$.

rank symbols, where $R = 3$ and $M = 3$. Figure 1.6 may be used to deduce the probability of first encountering this template, the results are shown in Table 1.4.

It can be seen from Tables 1.3 and 1.4 that the probability of a match at a given disparity k will differ depending on the template window. To find an overall expression for the probability of a match at a certain disparity, it is necessary to find the probability of each template. The probability of a match at disparity k is then given as

$$P(k) = \sum_{T\in\phi}^{T} P(T) \cdot P(k|T) \tag{1.5}$$

where ϕ is the set of possible template windows. For $M = 3$, and $R = 3$, this would become

$$\begin{aligned} P(k) \quad = \quad & P(\{0,0,0\}) \cdot P(k|\{0,0,0\}) + \\ & P(\{0,0,1\}) \cdot P(k|\{0,0,1\}) + \\ & \vdots \\ & P(\{2,2,2\}) \cdot P(k|\{2,2,2\}) \end{aligned} \tag{1.6}$$

From Equation (1.6), it can be seen that in order to compute the probability of a match a certain disparity, we need to derive the probabilities of a match at each disparity for each template. This was done for templates $\{0, 0, 0\}$ and $\{0, 0, 1\}$ in Figures 1.5 and 1.6. This is a laborious task, and would need to be done for every possible template for Equation (1.6) to be used.

Another point to note is that in this analysis, only the probability of a perfect match, ie, $T = C(k)$ is derived. This differs from what would actually take place in matching. If there is no match having $T = C(k)$, matching algorithms will select the next best match. In the case of the algorithm of Figure 1.1, this will be the match having the next lowest SAD score. Therefore, instead of merely computing the probability of $T = C(k)$ at a given disparity, it is necessary to compute the probability of a disparity having the lowest SAD score. This issue will be addressed in the following section.

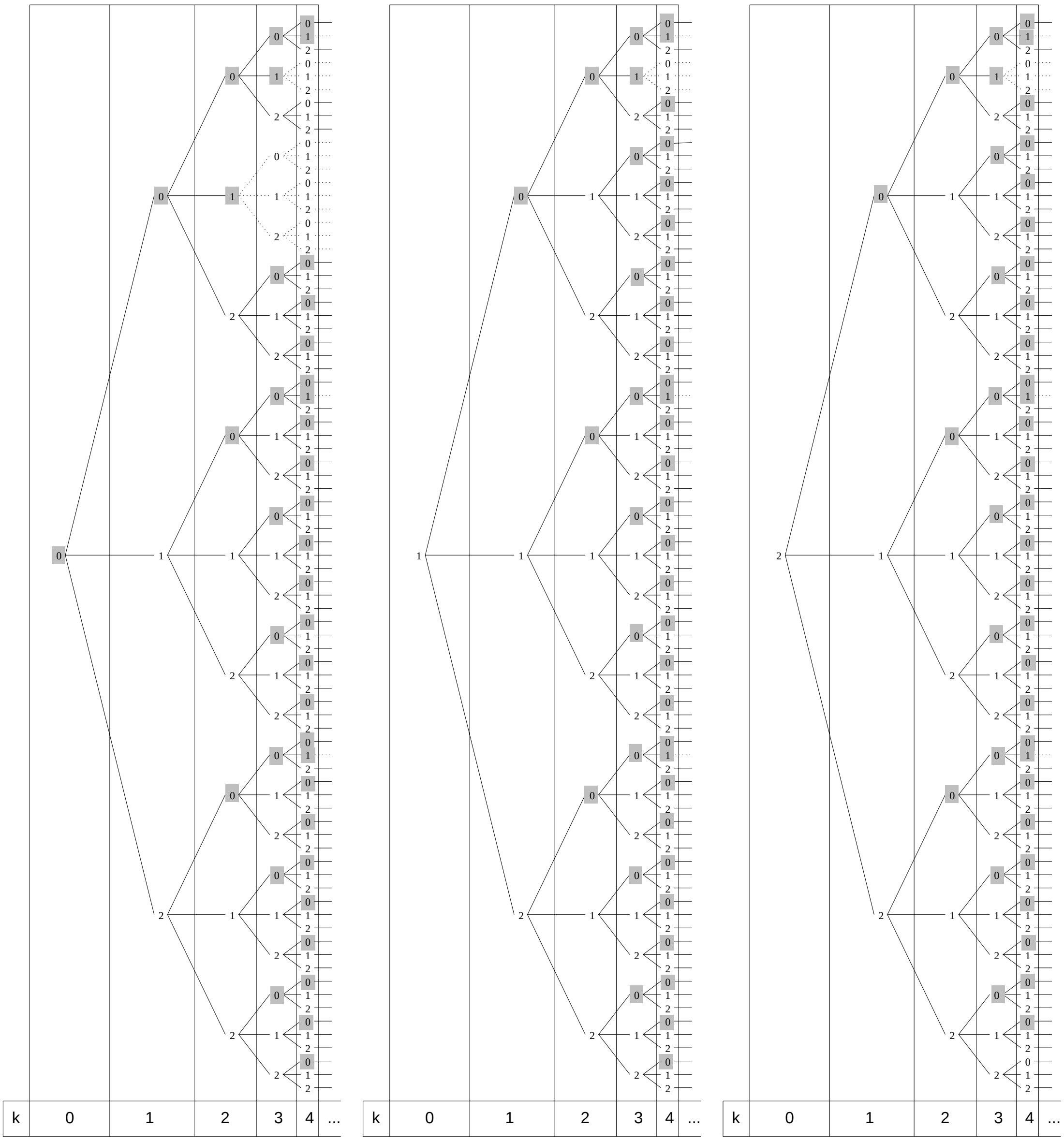


Figure 1.6:]
Probability of a match at disparity k along a line of rank symbols, for $R = 3$, $M = 3$, and a template window of $T = \{0, 0, 1\}$.

1.5 Disparity Probability Approach

The previous section found an expression for the probability that a particular disparity would be selected by the matching process. However, this result depended on finding the probability of selecting each disparity for each template. Computation of these results for each template was a difficult and laborious process. Furthermore the previous method considered only the case of a perfect match between template and candidate, ie, $T = C(k)$, and not the general case where the match having the lowest SAD score is selected.

The method used in this section begins with a enumeration of possible template windows. The probabilities of every template window and candidate window are used to compute the probabilities of every SAD score. The SAD score probabilities are then used to compute the probability of a match at a given disparity.

It was initially assumed that all possible rank symbols and therefore all possible match windows occur with equal probability. However, this assumption was found to be incorrect. Subsequent analysis was able to account for different match window probabilities, and experimental results were generated which were in agreement with those obtained theoretically.

1.5.1 Equal Match Window Probabilities

As with the previous section, 1-dimensional rank and match windows of size $1 \times R$ and $1 \times M$ respectively were used. The set of possible match windows, ϕ are enumerated. This is expressed as follows:

$$\phi = \{\phi_i\} \tag{1.7}$$

where $1 < i < q$. The number of possible templates, q, is given by:

$$q = R^M \tag{1.8}$$

Each possible match window ϕ_i consists of a permutation of rank symbols, which are integers in the range 0 to $R - 1$. For a rank window of size $R = 3$, the rank symbols consist of the set $0, 1, 2$. Furthermore, when a match window of size $M = 3$ is used, the set ϕ consists of all possible permutations of the symbols $0, 1, 2$, ie:

$$\phi = \{(0,0,0)(0,0,1)\ldots\ldots(2,2,2)\} \tag{1.9}$$

there being $R^M = 27$ permutations in all. The next step involves computing the probabilities of occurrence of each of the match windows, ϕ_i, which are denoted by $P(\phi_i)$. An initial estimate could assume that all match windows occur with equal probability, ie:

$$P(\phi_i) = \frac{1}{R^M} \tag{1.10}$$

Given the possible match windows, ϕ_i, it is possible to compute all possible SAD scores arising from every combination of template and candidate match windows. This set of SAD scores is given by:

$$S = \{S_i\} \tag{1.11}$$

where the SAD scores S_i range from 0 to $M(R-1)$. Table 1.5 shows the SAD scores arising from every template-candidate window combination, given the set of match windows of Equation 1.9. It can be seen from Table 1.5 that the set of possible SAD scores for $R = 3$ and $M = 3$ is comprised of integers in the range 0 to 6.

The probability of every entry in Table 1.5 can be computed as the product of the probabilities of the template window $P(\phi_i)$ and the candidate window $P(\phi_j)$. By summing the probabilities of entries which have the same SAD score, the probability of each SAD score, $P(S_i)$ can be found. This is expressed as follows:

$$P(S_x) = \left\{ \sum_{i=1}^{q} \sum_{j=1}^{q} P(\phi_i) \cdot P(\phi_j) \big| SAD(\phi_i, \phi_j) = x \right\} \tag{1.12}$$

where ϕ_i = template window, ϕ_j = candidate window and q = number of possible match windows, calculated from Equation (1.8). Substituting the template and candidate window probabilities from Table 1.7 into Equation (1.12), the probabilities of every possible SAD score are computed. These are shown in Table 1.6.

Finally, after calculating the probability of every possible SAD score, it is possible to compute the probability that a particular disparity d_i has the smallest SAD score, and thus will be chosen by the minima selection process. A disparity will be selected if it is the minimum value of the match function. If more than one minima exist with the same value, the one with the smallest disparity will be chosen. The probability of a disparity d_i being selected is given as follows:

$$\begin{aligned} P(d_i) = \quad & P\Big\{ \big(SAD(x) > 0, \forall x < d\big) \cdot \big(SAD(d) = 0\big) \cdot \big(SAD(x) \geq 0, \forall x > d\big) \Big\} \\ & \oplus P\Big\{ \big(SAD(x) > 1, \forall x < d\big) \cdot \big(SAD(d) = 1\big) \cdot \big(SAD(x) \geq 1, \forall x > d\big) \Big\} \\ & \qquad \vdots \\ & \oplus P\Big\{ \big(SAD(x) > S_n, \forall x < d\big) \cdot \big(SAD(d) = S_n\big) \cdot \big(SAD(x) \geq S_n, \forall x > d\big) \Big\} \end{aligned} \tag{1.13}$$

where n is the number of possible SAD scores, and $\cdot$, $\oplus$ denote the AND and OR operations respectively. Equation (1.13) may be summarised as:

$$P(d \text{ selected}) = \sum_{s=0}^{S_n} \left\{ \Big(\prod_{x=0}^{d-1} P\big(SAD(d) > s\big) \Big) P\big(SAD(d) = s\big) \Big(\prod_{s=d+1}^{d_{max}} P\big(SAD(d) \geq s\big) \Big) \right\} \tag{1.14}$$

The SAD probabilities of Table 1.6 were substituted into Equation (1.14), in order to compute the probabilities of each disparity being chosen. Equation (1.14) was evaluated for disparity values from $d_{min} = 0$ to $d_{max} = 200$. The results are plotted in Figure 1.7(a).

	Template Window																										
	000	001	002	010	011	012	020	021	022	100	101	102	110	111	112	120	121	122	200	201	202	210	211	212	220	221	222
000	0	1	2	1	2	3	2	3	4	1	2	3	2	3	4	3	4	5	2	3	4	3	4	5	4	5	6
001	1	0	1	2	1	2	3	2	3	2	1	2	3	2	3	4	3	4	3	2	3	4	3	4	5	4	5
002	2	1	0	3	2	1	4	3	2	3	2	1	4	3	2	5	4	3	4	3	2	5	4	3	6	5	4
010	1	2	3	0	1	2	1	2	3	2	3	4	1	2	3	2	3	4	3	4	5	2	3	4	3	4	5
011	2	1	2	1	0	1	2	1	2	3	2	3	2	1	2	3	2	3	4	3	4	3	2	3	4	3	4
012	3	2	1	2	1	0	3	2	1	4	3	2	3	2	1	4	3	2	5	4	3	4	3	2	5	4	3
020	2	3	4	1	2	3	0	1	2	3	4	5	2	3	4	1	2	3	4	5	6	3	4	5	2	3	4
021	3	2	3	2	1	2	1	0	1	4	3	4	3	2	3	2	1	2	5	4	5	4	3	4	3	2	3
022	4	3	2	3	2	1	2	1	0	5	4	3	4	3	2	3	2	1	6	5	4	5	4	3	4	3	2
100	1	2	3	2	3	4	3	4	5	0	1	2	1	2	3	2	3	4	1	2	3	2	3	4	3	4	5
101	2	1	2	3	2	3	4	3	4	1	0	1	2	1	2	3	2	3	2	1	2	3	2	3	4	3	4
102	3	2	1	4	3	2	5	4	3	2	1	0	3	2	1	4	3	2	3	2	1	4	3	2	5	4	3
110	2	3	4	1	2	3	2	3	4	1	2	3	0	1	2	1	2	3	2	3	4	1	2	3	2	3	4
111	3	2	3	2	1	2	3	2	3	2	1	2	1	0	1	2	1	2	3	2	3	2	1	2	3	2	3
112	4	3	2	3	2	1	4	3	2	3	2	1	2	1	0	3	2	1	4	3	2	3	2	1	4	3	2
120	3	4	5	2	3	4	1	2	3	2	3	4	1	2	3	0	1	2	3	4	5	2	3	4	1	2	3
121	4	3	4	3	2	3	2	1	2	3	2	3	2	1	2	1	0	1	4	3	4	3	2	3	2	1	2
122	5	4	3	4	3	2	3	2	1	4	3	2	3	2	1	2	1	0	5	4	3	4	3	2	3	2	1
200	2	3	4	3	4	5	4	5	6	1	2	3	2	3	4	3	4	5	0	1	2	1	2	3	2	3	4
201	3	2	3	4	3	4	5	4	5	2	1	2	3	2	3	4	3	4	1	0	1	2	1	2	3	2	3
202	4	3	2	5	4	3	6	5	4	3	2	1	4	3	2	5	4	3	2	1	0	3	2	1	4	3	2
210	3	4	5	2	3	4	3	4	5	2	3	4	1	2	3	2	3	4	1	2	3	0	1	2	1	2	3
211	4	3	4	3	2	3	4	3	4	3	2	3	2	1	2	3	2	3	2	1	2	1	0	1	2	1	2
212	5	4	3	4	3	2	5	4	3	4	3	2	3	2	1	4	3	2	3	2	1	2	1	0	3	2	1
220	4	5	6	3	4	5	2	3	4	3	4	5	2	3	4	1	2	3	2	3	4	1	2	3	0	1	2
221	5	4	5	4	3	4	3	2	3	4	3	4	3	2	3	2	1	2	3	2	3	2	1	2	1	0	1
222	6	5	4	5	4	3	4	3	2	5	4	3	4	3	2	3	2	1	4	3	2	3	2	1	2	1	0

Table 1.5: SAD scores arising from every template and candidate window combination when rank window size $R = 3$ and match window size $M = 3$.

SAD Score	Probability
0	0.0370
1	0.1481
2	0.2716
3	0.2853
4	0.1811
5	0.0658
6	0.0110

Table 1.6: Probability of SAD scores for a rank window $R = 3$ and a match window $M = 3$, assuming all possible template and candidate windows are equally probable, ie, have the probability of Equation (1.10).

Next, it was attempted to verify these results through actual experimentation. Two lines of random pixel values were generated, and rank transformed using a 1-dimensional rank window of size $R = 3$. A template matching window was selected from one line, while the second line was used as a series of candidate windows. The template was matched with the set of candidate windows, and the disparity value having the smallest SAD score selected. The experiment was repeated a large number of times, and the disparity results were plotted in the normalised histogram of Figure 1.7(b).

It was initially expected that Figure 1.7(a) and (b) would yield a graph of the same shape and magnitude. This was found not to be the case. This is because the assumption that the template and candidate windows occur with equal probability is not correct. The next step of this analysis therefore examines the case of variable template and candidate window probabilities.

1.5.2 Variable Match Window Probabilities

The assumption of Equation (1.10) that all possible template and candidate windows occur with equal probability, cannot hold for any real images. This is illustrated in Figure 1.8. If an image of pixel values is rank transformed, the resulting rank symbols are partially dependent on their neighbours within half a rank window width. In fact, some sequences of rank symbols, and therefore some match windows cannot occur at all. For example, for 1-D rank and match windows of size $R = 3$ and $M = 3$, it is not possible to have two 2's in succession in the rank transformed image.

It is possible, however, to estimate the match window probabilities $P(\phi_i)$ experimentally. In order to compute these estimates, an image consisting of a uniform random distribution of pixel values between 0 and 255 was generated. The rank transform of this image was then computed using a rank window of size $R = 3$. The number of instances of each possible match window were then counted, using a match window of size $M = 3$. The experiment was repeated a large number of times, and an estimate of the probability of each possible template, $P(\phi_i)$ derived. The results are shown in Table 1.7.

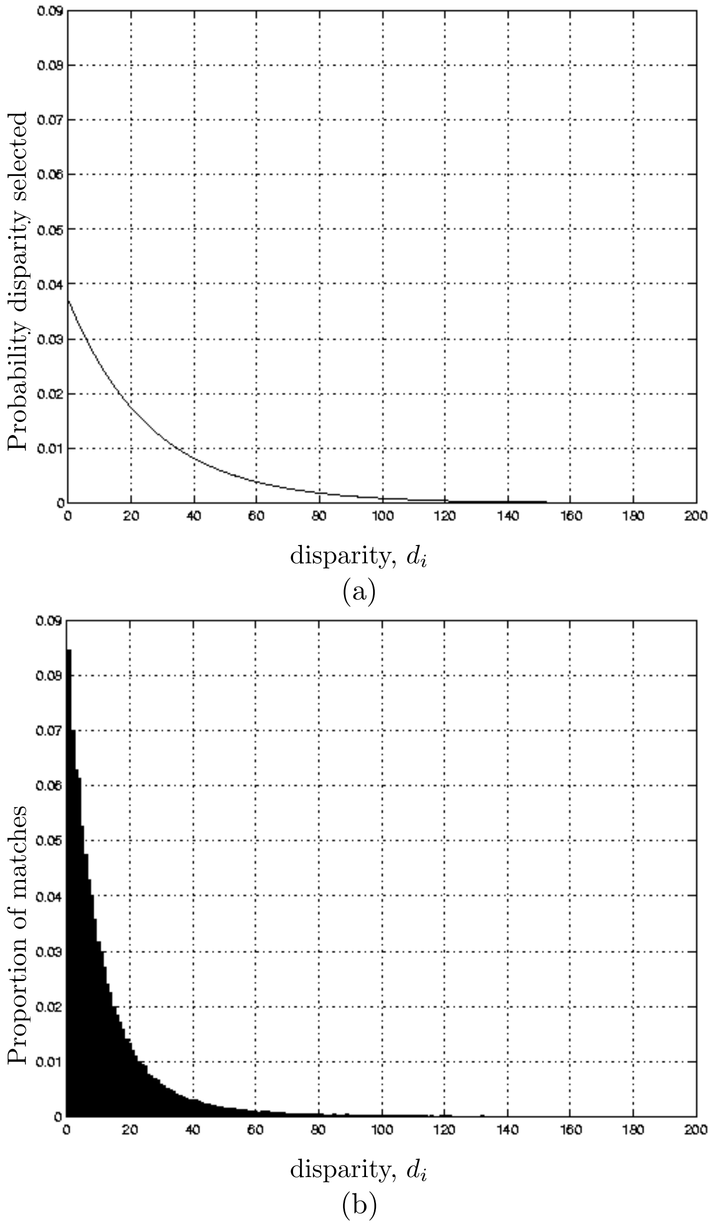


Figure 1.7: Probability of a particular disparity being chosen by the matching process (a) Computed theoretically from Equation (1.14), assuming equal match window probabilities. (b) Computed experimentally, by generating lines of random pixel values, rank transforming, and matching.

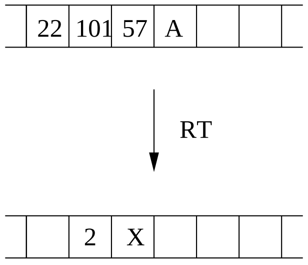


Figure 1.8: Example where $R = 3$ and $M = 3$, showing that it is not possible to generate a match window which contains two 2's in succession. Whatever the value of pixel A, it is not possible for rank symbol X to be 2.

template ϕ_i	probability $P(\phi_i)$
000	0
001	0.0005
002	0.0008
010	0.0011
011	0.0337
012	0.0907
020	0.1349
021	0.0734
022	0
100	0.0005
101	0.0497
102	0.0744
110	0.0350
111	0.0189
112	0.0322
120	0.0743
121	0.0485
122	0
200	0.0008
201	0.0755
202	0.1329
210	0.0885
211	0.0335
212	0
220	0
221	0
222	0

Table 1.7: Probabilities of template windows, $P(\phi_i)$, for $R = 3$ and $M = 3$.

SAD Score	Probability
0	0.0838
1	0.1630
2	0.2029
3	0.2564
4	0.1775
5	0.0806
6	0.0359

Table 1.8: Probability of SAD scores for a rank window $R = 3$ and a match window $M = 3$, assuming the template and candidate window probabilities of Table 1.7.

The SAD probabilities of Table 1.8 were substituted into Equation (1.14), in order to compute the probabilities of each disparity being chosen. Equation (1.14) was evaluated for disparity values from $d_{min} = 0$ to $d_{max} = 200$. The results are plotted in Figure 1.9(a).

The graph of disparity probabilities obtained through experimentation, shown in Figure 1.7(b), has been reproduced in Figure 1.9(b). It can be seen that the estimated probability of choosing each disparity, given by Figure 1.9(a), compares well with the experimentally generated results of (b), both graphs having a similar shape and magnitude.

The disparity probability graph shows that the lower the disparity, the higher the probability of that disparity being chosen by the matching process. This can be explained in that if more than one disparity has the lowest SAD score, the matching process will select the smallest disparity. Therefore, the matching process will favour selection of lower valued disparities.

The graph of Figure 1.9(a) plots the probability that a particular disparity is selected by the matching process. In the next section this will be used to predict the probability of an erroneous disparity, given a known true disparity.

1.6 Probability of Erroneous Disparity

The graph of Figure 1.9(a) plots the probability that a particular disparity d_i is selected by the matching process. Since the input “images” in this case consist of random pixel values, there is no correct match, and the disparity results from matching are considered analogous to an erroneous disparity. This section first of all uses Figure 1.9(a) to predict the probability of an erroneous disparity, given a true disparity, D. Next, experiments were carried out using contrived stereo pairs of random pixel values displaced by a known amount. It was found it was possible to predict the probability of the correct disparity being chosen, in each case.

1.6.1 Predicting the Probability of Erroneous Disparity

Given that the true disparity, D is known, it is then possible to use the result of Figure 1.7(a) to compute the probability of an incorrect disparity, ie, a disparity other than D being chosen. The matching process searches for a match starting at disparity d_{min} and stepping in integer increments to d_{max}. The probability of encountering an erroneous disparity before disparity D is the sum of the probabilities of choosing disparities d_{min} to $D - 1$. Computing this for every disparity results in the cumulative probability graph of Figure 1.10. This graph can be interpreted as follows. Given a true disparity D, the graph shows the probability of choosing an erroneous disparity instead.

1.6.2 Error Prediction for Random Images

This experiment involved generating an image of uniform random pixel values, then from this, extracting two images displaced by a known disparity, D. The two images were then rank

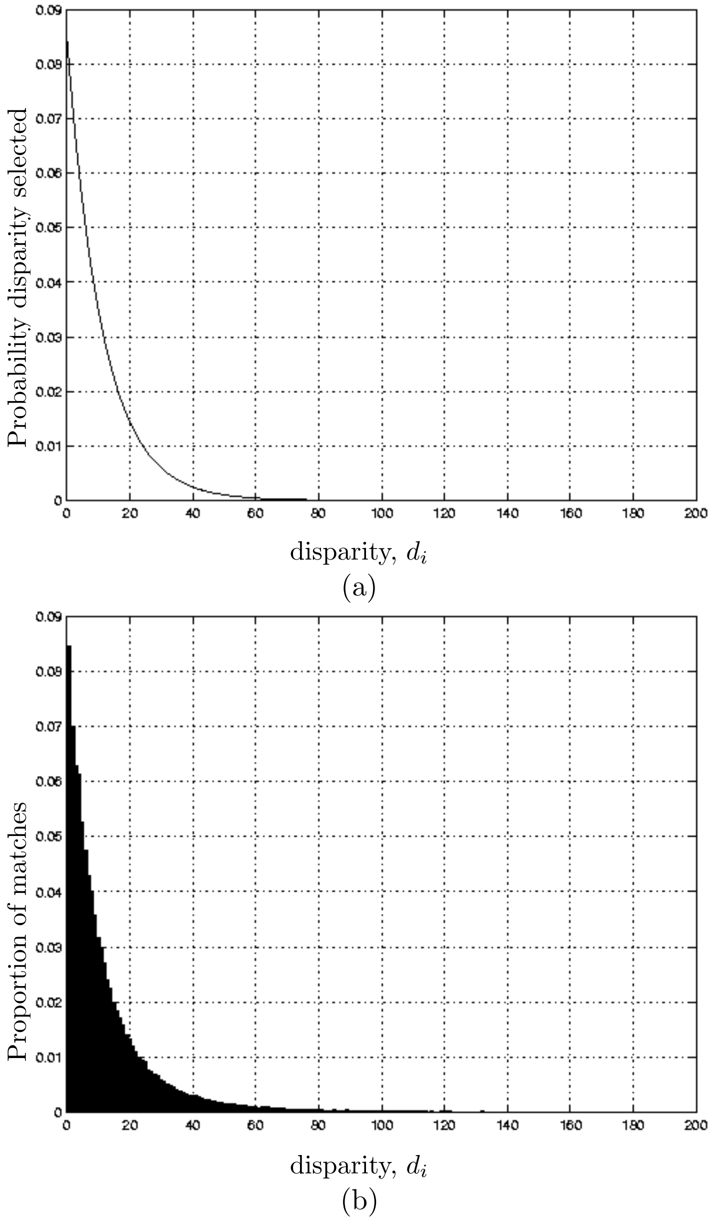


Figure 1.9: Probability of a particular disparity being chosen by the matching process (a) Computed theoretically from Equation (1.14), assuming the variable match window probabilities of Table 1.7. (b) Computed experimentally, by generating lines of random pixel values, rank transforming, and matching.

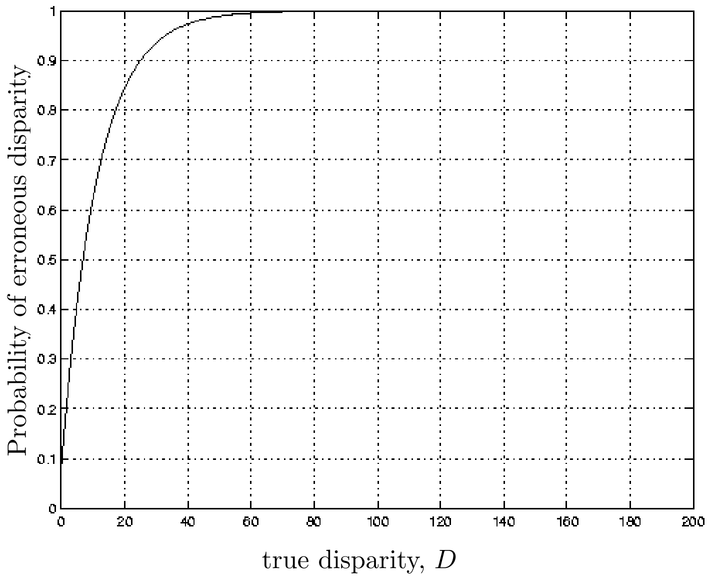


Figure 1.10: Probability of an erroneous disparity being chosen, given a known true disparity D.

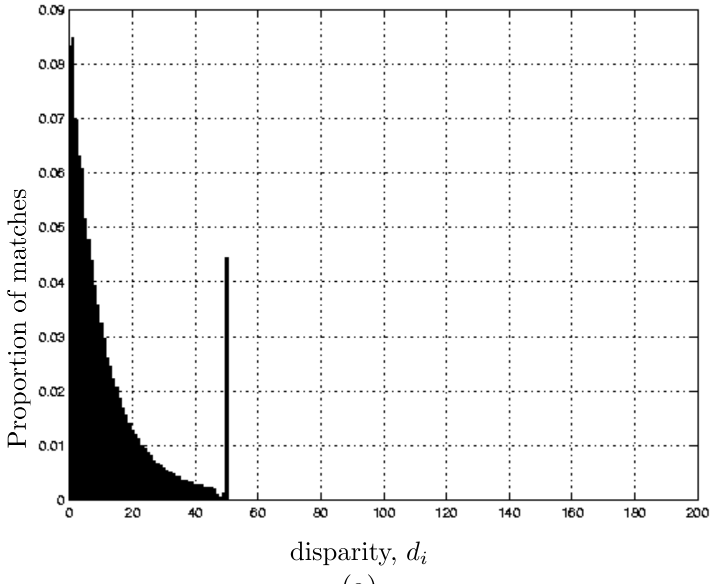
transformed and matched, and the disparity results displayed as a normalised histogram. Figure 1.11(a) and (b) show disparity results where the true disparity D is 50 and 100 respectively. Again, rank and match windows of size $R = 3$ and $M = 3$ were used. These results show that for disparities less than D, the shape of the histogram is the same as that of Figure 1.7(b). The remainder of the disparity values occur at the true disparity, D. The proportion of the disparity values which are in error, ie $\neq D$, is equivalent to the probability of erroneous disparity given by the graph of Figure 1.10, for disparity D. Therefore, the graph of Figure 1.10 may be used to predict the probability of an erroneous match, for input scenes consisting of random pixel values. The next section will show experiments with contrived stereo pairs derived from actual images.

1.7 Experiments with Contrived Stereo Pairs

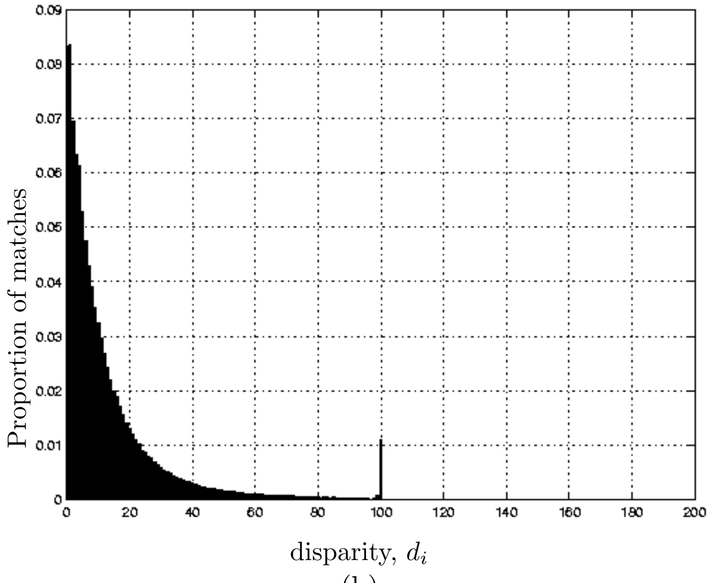
This section first of all shows a number of contrived stereo pairs which were generated from actual images, then uses these contrived pairs to produce normalised histograms of disparity results. These could then be compared with the theoretical disparity error graph of Figure 1.9(a).

1.7.1 Contrived Stereo Pairs

A number of contrived stereo pairs were generated, and are shown in Figures 1.13—1.18. These were constructed in order to simulate stereo pairs with ground truth, where the disparity of every pixel is known. For the pairs shown in this section, the images have a constant disparity of 50 pixels.



(a)



(b)

Figure 1.11: Disparity results obtained by matching a contrived stereo pair consisting of random pixel values, where the images are displaced by a known disparity of (a) $D = 50$ and (b) $D = 100$. The total proportion of erroneous disparities is equivalent to the erroneous disparity probability of Figure 1.10, for disparity D.

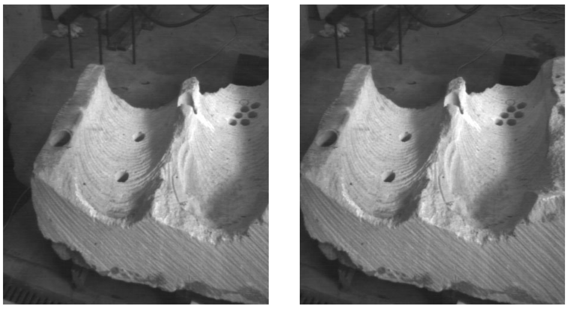


Figure 1.12: Contrived stereo pair 1.

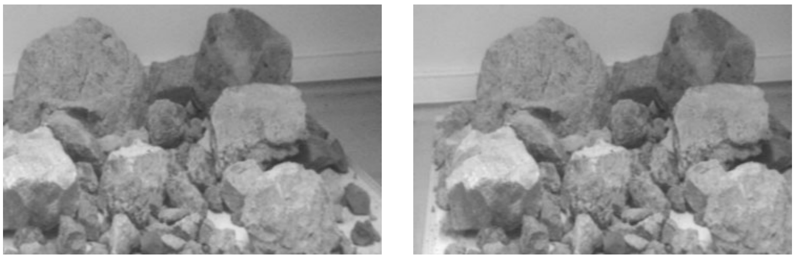


Figure 1.13: Contrived stereo pair 2.

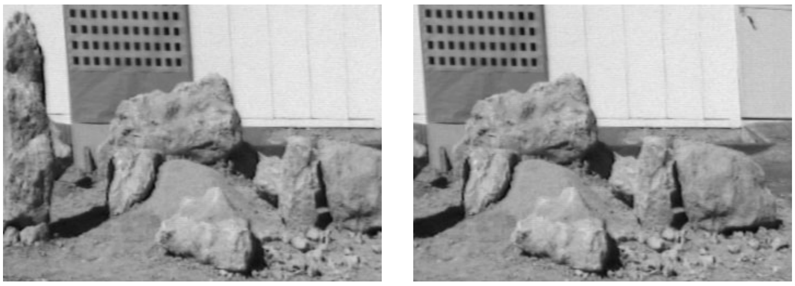


Figure 1.14: Contrived stereo pair 3.

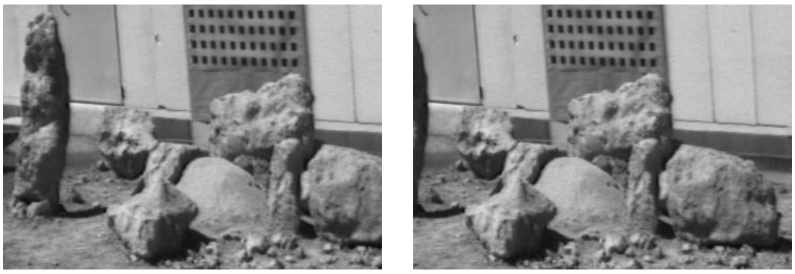


Figure 1.15: Contrived stereo pair 4.

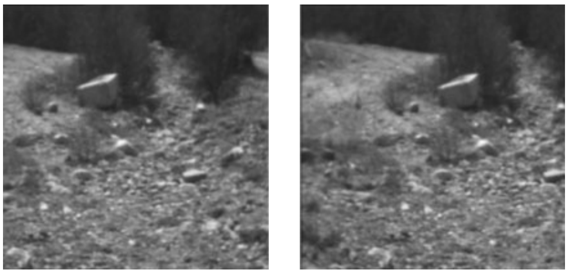


Figure 1.16: Contrived stereo pair 5.

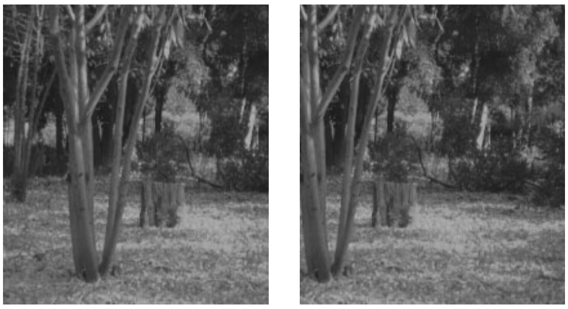


Figure 1.17: Contrived stereo pair 6.

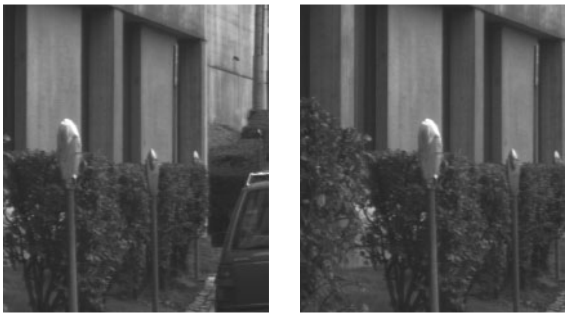


Figure 1.18: Contrived stereo pair 7.

1.7.2 Experimental Results

In this experiment, the contrived stereo pairs of Section 1.7.1 were rank transformed and matched. For all these pairs, the true disparity D is known and has a value of 50 pixels. These stereo pairs were matched using 1D rank and match windows of size $R = 3$ and $M = 3$. The disparity results were again displayed in the form of normalised histograms, in Figures 1.19–1.25. These results show that a number of matches occur at the correct disparity of $D = 50$ pixels. The remainder of the matches occurring at disparity $\neq 50$ are incorrect.

The shape of the graph for these erroneous matches at disparity $\neq 50$ was then compared with the shape of the predicted error graph of Figure 1.9(a). It can be seen that the graphs for the erroneous matches are similar in overall shape to the predicted error graph, in that lower values of disparity are more likely to be erroneously selected. However, looking in more detail, these graphs differ in shape and magnitude to the predicted error graphs. The reason for this is that the computation of the predicted error graph of Figure 1.9(a) assumes that images consist of uniformly distributed random pixel values, which is not the case for real images.

1.8 Discussion and Conclusion

This work has focussed on the rank transform for stereo matching. In particular, a method of predicting the probability of false matches has been devised. This method is based on enumerating every possible template and candidate window, and estimating their probability of

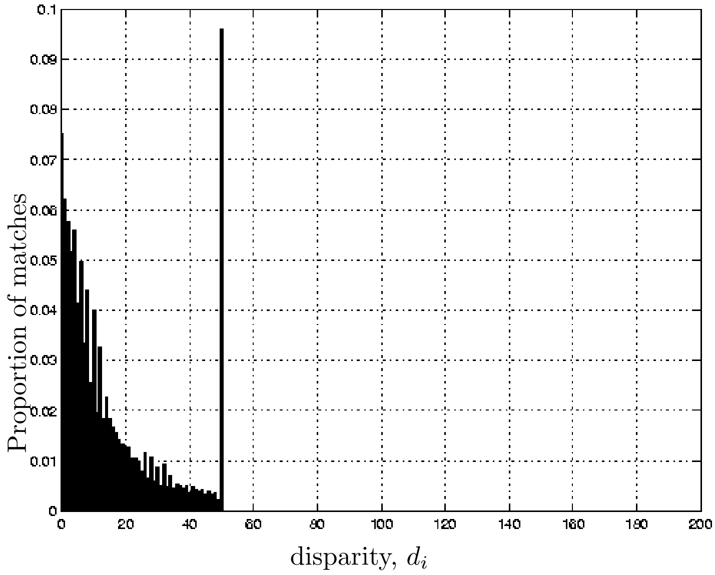


Figure 1.19: Disparity results obtained by matching contrived stereo pair of Figure 1.12, where images are displaced by a known disparity of $D = 50$.

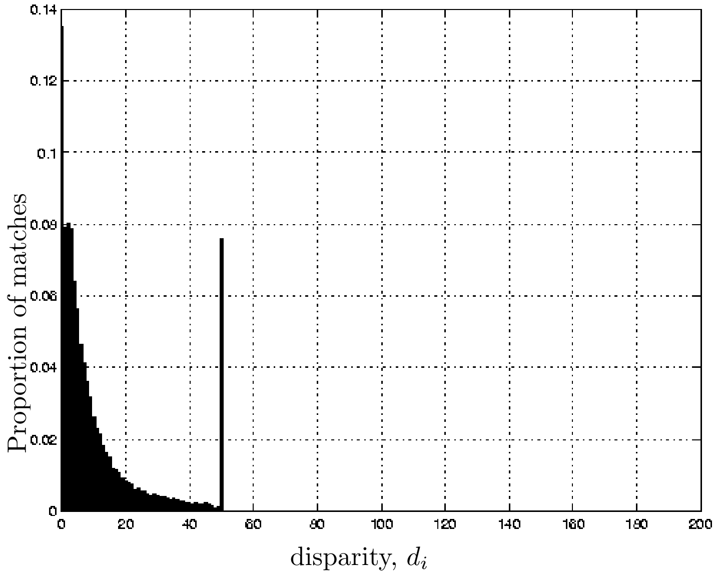


Figure 1.20: Disparity results obtained by matching contrived stereo pair of Figure 1.13, where images are displaced by a known disparity of $D = 50$.

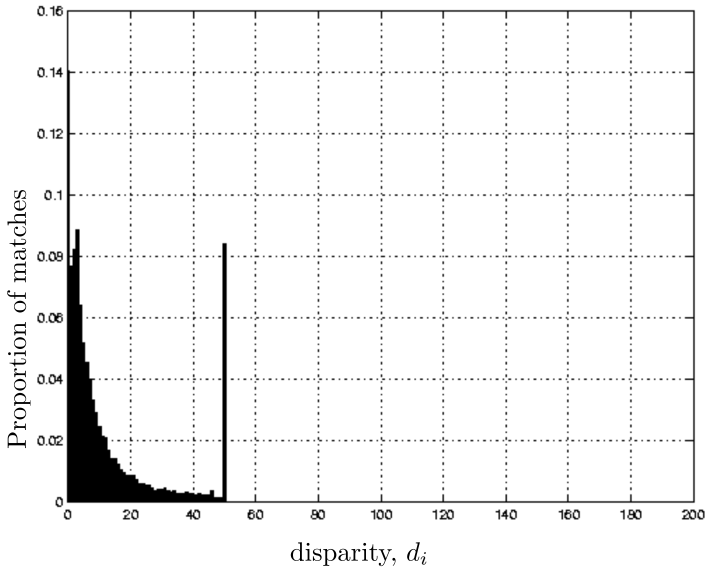


Figure 1.21: Disparity results obtained by matching contrived stereo pair of Figure 1.14, where images are displaced by a known disparity of $D = 50$.

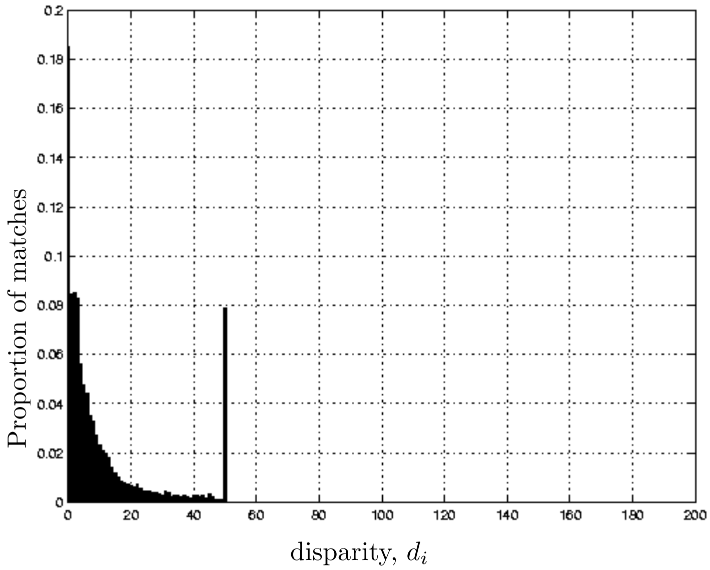


Figure 1.22: Disparity results obtained by matching contrived stereo pair of Figure 1.15, where images are displaced by a known disparity of $D = 50$.

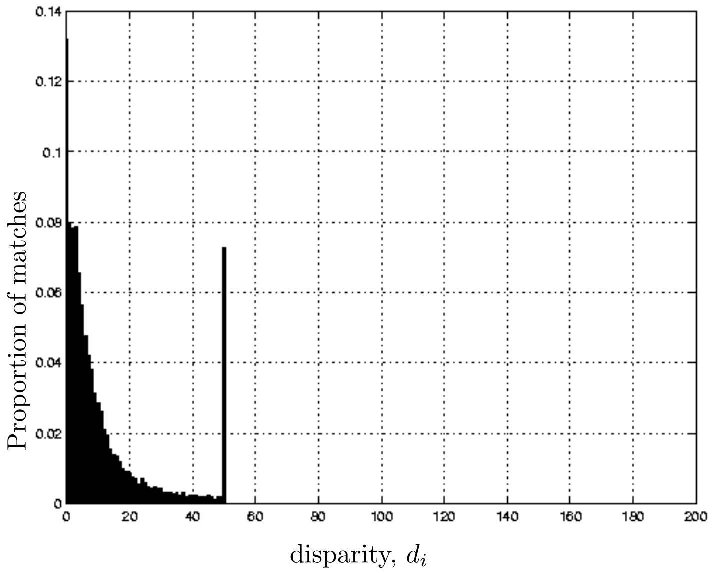


Figure 1.23: Disparity results obtained by matching contrived stereo pair of Figure 1.16, where images are displaced by a known disparity of $D = 50$.

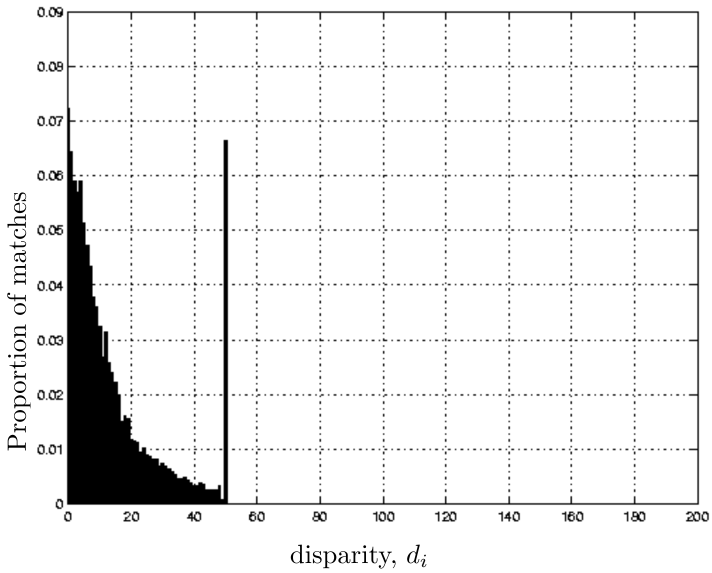


Figure 1.24: Disparity results obtained by matching contrived stereo pair of Figure 1.17, where images are displaced by a known disparity of $D = 50$.

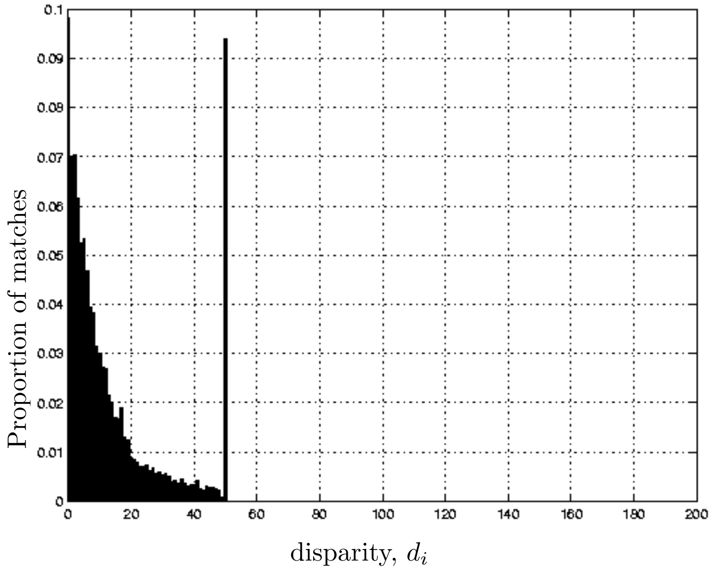


Figure 1.25: Disparity results obtained by matching contrived stereo pair of Figure 1.18, where images are displaced by a known disparity of $D = 50$.

occurrence. It was then possible to compute the probability of each possible SAD score, and subsequently the probability of each disparity being selected by the matching process. This information could be used to derive the probability of an invalid disparity being returned by the matching process, given a certain true disparity.

As can be seen by Figure 1.11, it was possible to predict matching errors when the input images were assumed to consist of uniformly distributed random pixel values. However in practice, the pixel distribution of real images will differ from this assumption. Figures 1.19–1.25 show disparity error graphs computed using stereo pairs contrived from actual images. These are similar in overall shape to the predicted disparity errors of Figure 1.7(a). However, in each case, there are some differences. This can be explained in that a different pixel distribution will result in different match window probabilities, $P(\phi_i)$, which in turn will result in different SAD score probabilities, $P(S_i)$. This in turn will alter the probabilities of a disparity being chosen, $P(d_i)$. For an image with a pixel distribution that differs from the assumed uniform random distribution, it would still be possible to use the method outlined in this work, as long as the match window probabilities, $P(\phi_i)$ can be predicted. This may well be possible for some images, where the statistical properties of the pixel values and therefore $P(\phi_i)$ are constant over the whole image. However, $P(\phi_i)$ will usually vary according to the image content at different locations in the image. Therefore, in order to apply the disparity error prediction techniques outlined in this work, it would be necessary to first predict the template probabilities $P(\phi_i)$ at all image locations, which is difficult to do in practice.

A further difficulty in applying this disparity error prediction technique in practice is the amount of computation required for practical window sizes. For simplicity, this study has assumed 1-D rank and match windows of size $R = 3$ and $M = 3$. This results in $R^M = 27$ possible match windows, for which a minimum amount of computation is required. However, the amount of computation required increases exponentially with window size. For example, setting $R = 5$ and $M = 11$ results in 5^{11} possible match windows. It is necessary to compute the probabilities $P(\phi_i)$ of each of these windows, which are in turn used in Equation (1.12) to compute the probability of each SAD score. As a result, the computations become too lengthy to be completed within a reasonable amount of time. The situation is considerably worse for 2-D window sizes used in practice such as 5×5 for the rank window and 11×11 for the match window.

In conclusion, this paper has performed an analysis of match error prediction, and has devised a method of predicting disparity errors. However, this method depends on being able to estimate the probabilities of all possible template and candidate windows, which is difficult to do in practice. Also, the computations become intractable for practical window sizes. The exercise has nonetheless provided an interesting and novel analysis of match reliability.

Bibliography

[1] P. Aschwanden and W. Guggenbühl. Experimental results from a comparative study on correlation-type registration algorithms. In Förstner and Ruwiedel, editors, *Robust Computer Vision*, pages 268–289. Wickmann, 1993.

[2] H. Baker and T. Binford. Depth from edge and intensity based stereo. In *7th Joint Conference on Artificial Intelligence*, pages 631–636, 1981.

[3] J. Banks, M. Bennamoun, P. Corke, and K. Kubik. Evaluation of new and existing confidence measures for stereo matching. In *Image and Vision Computing New Zealand (IVCNZ)'98*, pages 252–261, Auckland, New Zealand, November 16–18 1998.

[4] J. Banks, M. Bennamoun, P. Corke, and K. Kubik. Suitability of non-parametric stereo matching techniques for mining automation. *Australian Journal of Intelligent Information Processing Systems*, 5(2):111–119, 1998.

[5] S. Barnard and M. Fischler. Computational stereo. *Computing Surveys*, 14(4):553–572, December 1982.

[6] S. Barnard and W. Thompson. Disparity analysis of images. *IEEE Transactions On Pattern Analysis and Machine Intelligence*, PAMI-2(4):333–340, July 1980.

[7] I. Cox, S. Hingorani, S. Rao, and B. Maggs. A maximum likelihood stereo algorithm. *Computer Vision and Image Understanding*, 63(3):542–567, May 1996.

[8] U. Dhond and J. Aggarwal. Structure from stereo – a review. *IEEE transactions on systems, man, and cybernetics*, 19(6):1489–1510, 1989.

[9] P. Fua. A parallel stereo algorithm that produces dense depth maps and preserves image features. *Machine Vision and Applications*, 6:35–49, 1993.

[10] J. Greenfeld and A. Schenk. Experiments with edge-based stereo matching. *Photogrammetric Engineering and Remote Sensing*, 55(12):1771–1777, December 1989.

[11] M. Hannah. A system for digital stereo image matching. *Photogrammetric Engineering and Remote Sensing*, 55(12):1765–1770, December 1989.

[12] R. Hummel and S. Zucker. on the foundations of relaxation labeling process. *IEEE Transactions on pattern analysis and machine intelligence*, 5(3):267–287, May 1983.

[13] R. Koch. Automatic reconstruction of buildings from stereoscopic image sequences. In *Proceedings of Eurographics '93*, Barcelona, Spain, September 1993.

[14] Y. Lu, K. Kubik, and M. Bennamoun. Stereo image matching based on probability relaxation. In *IEEE TENCON'97*, Brisbane, December 1997.

[15] L. Matthies. Stereo vision for planetary rovers: stochastic modelling to near real-time implementation. *International Journal of computer vision*, 8(1):71–91, 1992.

[16] G. Medioni and R. Nevatia. Segment-based stereo matching. *Computer vision, graphics and image processing*, 31:2–18, 1985.

[17] C. Slama, editor. *Manual of Photogrammetry.* American Society of Photogrammetry, 4th edition, 1980.

[18] Z. Wang. *Principles of Photogrammetry (with Remote Sensing).* Press of Wuhan Technical University of Surveying and Mapping, 1990.

[19] R. Zabih and J. Woodfill. Non-parametric local transforms for computing visual correspondence. In *3rd European Conference on Computer Vision*, Stockholm, 1994.

Chapter 2

Image Matching and Image Analysis Techniques for DEM Generation

Abstract

Digital Elevation Models (DEM) produced by digital photogrammetry workstations often used as a component in complex Geographic Information Systems (GIS) modeling. Since the accuracy of GIS databases must be within a specified range for appropriate analysis of the information and subsequent decision making, an accurate DEM is needed. Conventional image matching techniques may be classified as either area-based or feature-based methods. These image matching techniques could not overcome the disparity discontinuities problem and only supply a Digital Surface Model (DSM). This means that matching may not occur on the terrain surface, but on the tops of man-made objects such as houses, or on the top of the vegetation. In order to get more accurate DEM from overlapping digital aerial images and satellite images, a 3D terrain reconstruction method using compound techniques is proposed. Area-based image matching method is used to supply dense disparities. Image edge detection and texture analysis techniques are used to find houses and tree areas. The final DEM comes from the two parts of image matching and image analysis and hence overcomes errors in the DEM caused by matching on tops of trees or man-made objects.

2.1 Introduction

A major research area in computer vision and digital photogrammetry is image matching for the reconstruction of a Digital Elevation Model (DEM). This process, which is a fundamental problem in stereo vision, involves the determination of corresponding points in a stereo image pair. From the image coordinates of these corresponding points, their 3D positions can be computed by triangulation, from the known camera geometry, and additional points on the terrain surface can be obtained by interpolation [1]. However, 3D terrain reconstruction from aerial or satellite images will be subject to errors in built-up and treed areas [2, 3, 4]. In order to obtain a more accurate 3D terrain model, it is necessary to develop better methods to overcome these problems. In this paper, procedures are described that combine image analysis and image matching methods in an attempt to ensure that the elevation points are measured only on the natural terrain surface, and not on the top of vegetation or man made features such as houses [5]. This chapter is organized as follows. Section 2 introduces the proposed system. Section 3 and Section 4 describe the stereo image processing procedure and the single image processing procedure respectively. Section 5 gives experimental results, and conclusions are drawn in Section 6.

2.2 General Description of the Terrain Reconstruction System

Figure 1 illustrates the architecture of the proposed 3D reconstruction system. The goal of this technique is to achieve more accurate reconstruction of elevations from overlapping aerial or satellite images over a wide variety of terrain types and ground cover.

The key functions of data acquisition and pre-processing, are to acquire the images in digital form and improve the output for the subsequent processes by the production of epipolar images from the original left and right images.

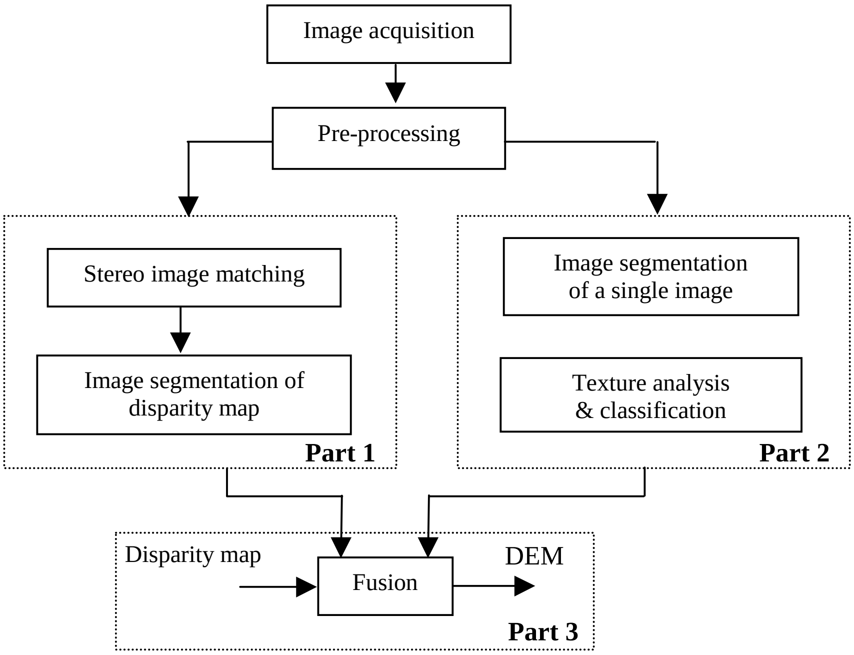


Image acquisition
Pre-processing
Stereo image matching
Image segmentation of disparity map
Part 1
Image segmentation of a single image
Texture analysis & classification
Part 2
Disparity map
Fusion
DEM
Part 3

Figure 1: Architecture of the proposed reconstruction system

The system developed for DEM determination consists of three main parts. Part 1 performs the matching of the stereo image pair, derives a disparity map, and produces a digital surface model (DSM). An analysis of the disparity map then reveals possible house and tree areas.

Part 2 applies standard image segmentation and texture analysis techniques to the left image to recognize houses and to locate trees. Based on a combination of the 3D information extracted from the disparity map and the 2D image segmentation, the elevations derived in regions which do not appear to represent the terrain surface can be removed from the DSM in Part 3, thus leading to a more accurate DEM. In the case of the houses, the elevations can then be interpolated from the surrounding terrain. Where trees exist, the DSM heights can be reduced by the tree heights.

2.3 Processing of Stereo Image Pair

2.3.1 Derivation of the disparity map

The first step in the recovery of 3D terrain information from overlapping aerial or satellite images is based on the matching of corresponding pixels in the stereo images. From the matched points, the 3D coordinates of a point can be obtained by triangulation using information of the image capturing geometry. Many computational algorithms have been used to solve the stereo matching problem. Conventional image matching techniques may be classified as either feature-based or area-based. Each of these approaches has advantages and disadvantages. Feature-based matching generally produces good results, is less expensive and is more tolerant of illumination differences and geometric distortions. However, only a few points may be matched in some regions due to the scarcity of the features, which leads to large areas being subjected to inaccurate interpolations [6, 7, 8]. Area-based matching algorithms can provide denser disparity maps. However, they are intolerant to geometric distortions caused by steep terrain slopes or imaging geometry [9, 10, 11].

In order to produce a dense, reliable matching result, commercial software packages normally employ hierarchical area-based matching. Since this paper concentrates on the process of recognizing houses and trees in images, and correcting for their effects on derived elevations from image matching, the disparity values obtained from matching in the software package VirtuoZo have been directly used in the subsequent stages of the system in Figure 1. For these developments, a dense sample of points in the disparity map is required in order to avoid some of structures being missed. Hence, a matching grid interval of 5 pixels in column and row directions has been used. The derived disparity map is then interpolated to the same size as the original image for further processing.

2.3.2 Edge Detection Applied To The Disparity Map

Figure 2 illustrates the stereo image processing procedure of Part 1 in detail.

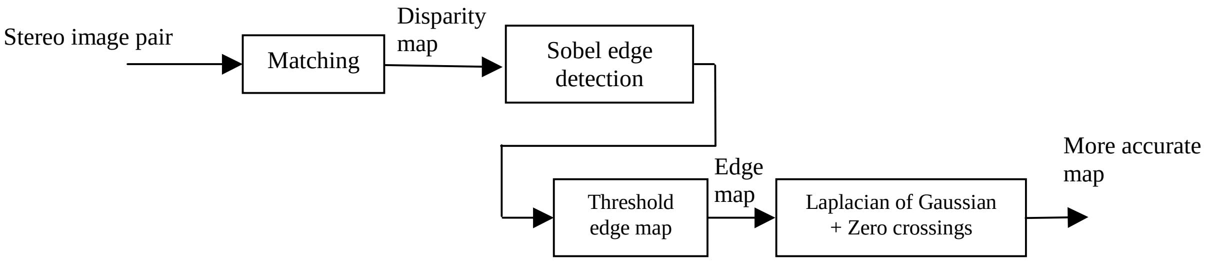


Figure 2: Stereo image processing procedure of Part 1 in Figure 1

Although automatic matching algorithms used by the standard software packages are not able to distinguish between the terrain surface and objects on and above this surface, the output of stereo image matching can supply significant information to identify man-made structures such as houses, and trees. When houses and trees exist in the images, the disparity values of these areas are locally larger, often with discontinuities occurring in the disparity values at the edges of the features [3, 4]. Hence, edge extraction methods, in which relatively distinct changes in grey level properties between two regions in an image are located by changes in local derivatives, are used to define the discontinuities in the disparities values by treating the disparity map as an image. Common methods used to calculate these derivatives are the gradient and Laplacian operators [12]. The Sobel gradient operator has the advantage of providing both a differencing and a smoothing effect. Since the derivatives enhance noise, the smoothing effects are a particularly attractive feature of this operator.

The process of edge detection in the disparity map by the Sobel operator involves a repeated convolution of a 3×3 operator over the disparity map, by rotating the operator in eight positions about its center. A maximum response will occur for one of the eight orientations of the operator. This response value and the operator's orientation are stored as values in a 'Sobel image', defining the edges in the disparity map. In order to reduce the effects of noise in extracting the discontinuities, a threshold must be set for recognizing an edge in the disparity map, as a change of disparity. In general, threshold selection is one of the key issues in edge detection [13]. If the threshold level is set too high, it will not permit detection of low magnitude edges, while if it is set too low, it will cause noise to be falsely detected as edges. In this work, the standard deviation (σ) of the variation in the Sobel image magnitudes is used for threshold definition. The pixels in the Sobel image, whose magnitudes satisfy the following threshold formula are deemed to be edge pixels in the disparity image. Equation (1) was obtained by tests to determine an optimal edge image.

$$mag \geq 3 \times \sigma \tag{1}$$

The Sobel operator supplies edges which are more than one pixel wide. To reduce edges to one pixel wide, an edge thinning algorithm is applied to the Sobel image, by computing second derivatives to determine zero-crossings in the image. This is obtained by convolving the disparity image with the Laplacian of a 2D Gaussian function. The pixels which have local maxima in the Sobel image and a zero value for the second derivative can be used to identify the edges in the disparity map and hence the edges of tree and house areas.

The processing of the left image to locate houses and tree areas, in the next section, involves the steps of: region growing to extract homogeneous regions in the image; an analysis of geometric and radiometric parameters of the regions; determination of the texture of the extracted regions; and the application of morphological operators to confirm the regions.

2.4 Processing of Single Image

2.4.1 Single image processing for house extraction

The left image is processed to separate house and tree areas. Figure 3 illustrates the implementation steps.

2.4.1.1 Dynamic Region Growing

A dynamic region growing technique is used to determine homogeneous areas in the images in which the intensities of the pixel values are within a given threshold value. The threshold is modified dynamically according to the mean and standard deviation of the pixels in the region while it is being grown, based on Equation (2) [14]. The adaptive threshold will never be larger than the pre-defined threshold (*T*), but may be smaller.

$$th = (1\text{-min}(0.8, \text{standard-deviation/mean}))*T \tag{2}$$

The first region is chosen at the lower left corner of the image and processed until that region can no longer be grown. The next region starts at a pixel that has not been incorporated into the previous region. This process continues until all pixels have been grouped into separate regions which represent homogeneous areas in the input image.

This dynamic region growing technique will locate most homogenous regions, but sometimes the process may cross from one region to the next. To limit this occurrence, the region growing needs to be constrained using an edge detection process. Sobel edge masks are used to produce *x* and *y* direction edge images, which define region boundaries between adjacent pixels in the *x* and *y* directions. These *x* and *y* edge images are used as input for the region growing and are combined with the dynamic threshold in order to locate the regions more precisely.

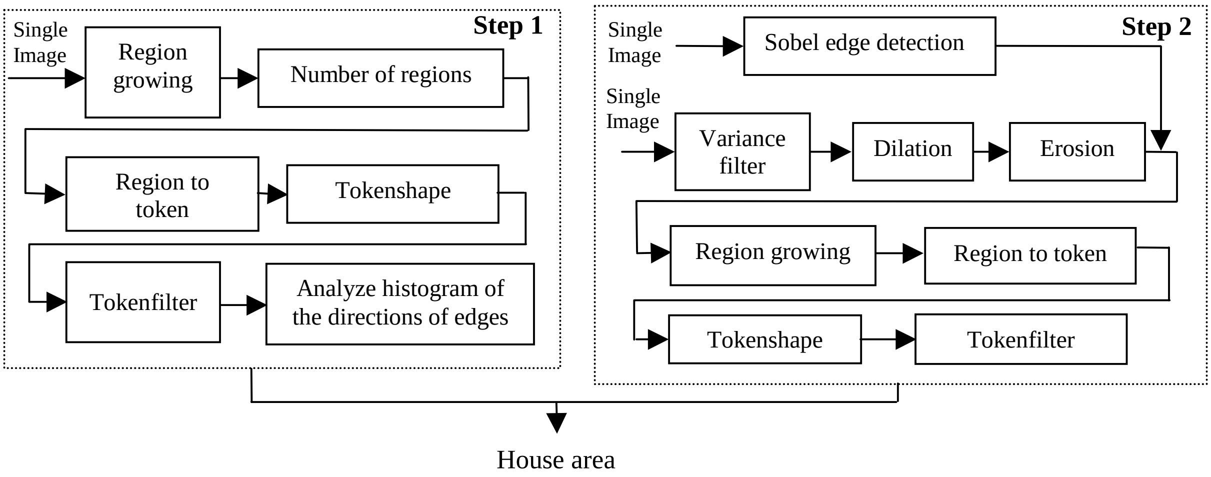


Figure 3: A single image processing procedure of Part 2 in Figure 1

2.4.1.2 Analyzing region parameters

The number of regions in the image are calculated in the Figure 3. Regions are represented by tokens which also describe the features of that region [15]. The task “Region to token” implements the transformation from region to token. “Tokenshape” is used to calculate a series of feature values for the regions, as follows [15]:

1) perimeter,
2) intensity_mean,
3) br_to_perimeter= $\left(\frac{perimeter}{2height + 2width}\right)$
4) log_h_to_w= $\log_{10}\left(\frac{height}{width}\right)$
5) pixel_count.

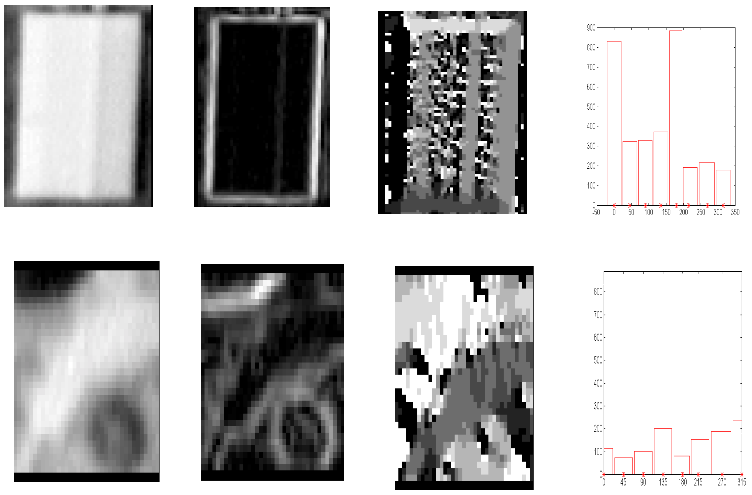
The third feature value measures region convexity, which is larger for regions with boundaries which double back on themselves. The fourth feature value gives a symmetric measure of aspect ratio.

“Tokenfilter” in Figure 3 is used to filter out extracted regions which are not houses, based on the five feature values for every region. For each image, the minimum sizes of houses, perimeter and pixel_count can be defined. Intensity_mean is based on a special case and helps to extract the houses which have a bright roof. However, houses with dark roofs will have similar grey values as the ground cover, so it is difficult to locate them. Step 2 in Figure 3 can be used to recognize the area of the dark roof house as described below.

2.4.1.3 Analysing the histogram of orientations of edges

After analysing their parameters, most extracted regions can be eliminated, but some regions will be described incorrectly as houses. In order to eliminate these false areas, an analysis is made of the histogram of the orientations of edges of the regions.

Figure 4 illustrates the procedure for distinguishing houses from other objects. There are eight compass directions of region edge pixels in this small test image. Table 1 shows that for regularly shaped houses, the histogram contains a greater number of edges whose orientations are mutually orthogonal such as here at 0° and 90°. This does not occur for all images. However, because the directions of the edge pixels of the houses in this test image are significantly different from those of trees, which are obviously not square, this method can assist in differentiating between houses and trees.



(1) (2) (3) (4)

(1) original images, (2) edge magnitude images, (3) edge orientation images, (4) histogram of the orientation images

Figure 4 Analysis the difference between two regions

Table 1: The number of different orientation pixels
(h: house, t: tree)

	0○ *	45○	90 ○ *	135○	180○ *
h	832	325	331	371	883
t	115	73	103	201	80
	225○	270○ *	315○	Sum of *	Sum of other
h	192	217	179	2263	1067
t	153	187	235	485	662

2.4.1.4 Texture analysis

Locating houses with dark roofs has been addressed in [16] by texture analysis. A variance filter is determined from a texture algorithm capable of distinguishing uniform intensity areas in images. Although the dark roofs have similar intensity to the ground cover in the image, they have different textures. The variance filter, which provides a measure of local homogeneity of the intensities in an image, can also be regarded as a non-linear non-directional edge detector [17, 18]. The variance filter involves replacing a central pixel value with the variance of a specified set of pixel values surrounding it in a window on the image, which does not need to be square. The variance of such a set is given as follows:

$$\bar{x} = \frac{1}{n^2}\sum_{r=1}^{n}\sum_{c=1}^{n} x_{rc}$$

$$v_w = \frac{1}{n^2}\sum_{r=1}^{n}\sum_{c=1}^{n}(x_{rc} - \bar{x})^2 = \frac{1}{n^2}\sum_{r=1}^{n}\sum_{c=1}^{n}(x_{rc}^2) - \bar{x}^2 \tag{3}$$

Where n *x* n is the total number of pixels in the window. *w* is the window in the image, x_{rc} is the value of the pixel at row r and column c in the windows. $\bar{x}$ is the mean of pixel values in the window.

2.4.1.5 The Morphological functions

Morphological transformations are powerful tools for extracting image components, that are useful for representing and describing region shapes [15, 19]. Dilation combines two image sets using vector addition of set elements, while erosion combines two sets by vector subtraction.

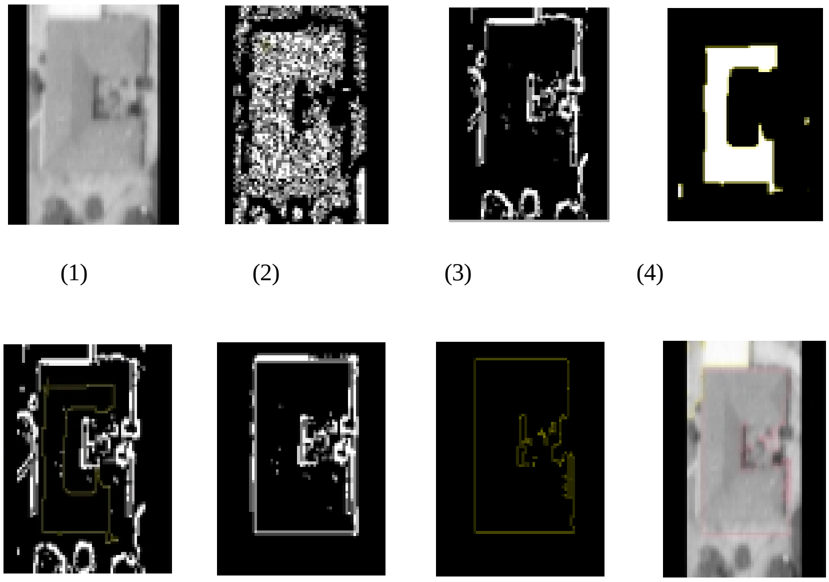
After calculating and thresholding the variance for each pixel derived by equation (3), the area of the dark roof can normally be recognized. If some of the points in the dark roofs have a higher variance, some small sections may not be correctly assigned as roof pixels. A dilation operation can be used to fill in these small areas. Since some of the boundary pixels of dark roof areas may be connected with other regions, an erosion operation can be used to separate them from other regions. Dark roofs can thus normally be extracted.

By combining the results of morphological transformation with the edges derived from the Sobel edge detection, a more accurate region boundary can be obtained. As shown in step 2 in Figure 3, the subsequent operations are similar to the ones described in step 1. After steps 1 and 2, the final house regions can be obtained. Figure 5 illustrates the results of processing step 2 in Figure 3.

2.4.2 Single Image Processing for Delineation Tree

2.4.2.1. Image segmentation

The image is processed to recognize tree areas. Figure 6 illustrates the implementation steps. Using combined results from step1 and step2, delineation of tree areas can be more accurate than either step1 or step 2.



(1) (2) (3) (4)

(5) (6) (7) (8)

(1) Image of dark roof house, (2) Result of threshold of variance, (3) Edges detected by the Sobel operator, (4) Result of the morphological transformation, (5) Combined result of (3) and(4), (6) New boundary location for dark roof, (7) Region of dark roof, (8) Region overlaid on the image

Figure 5: Results by step 2 in Figure 3

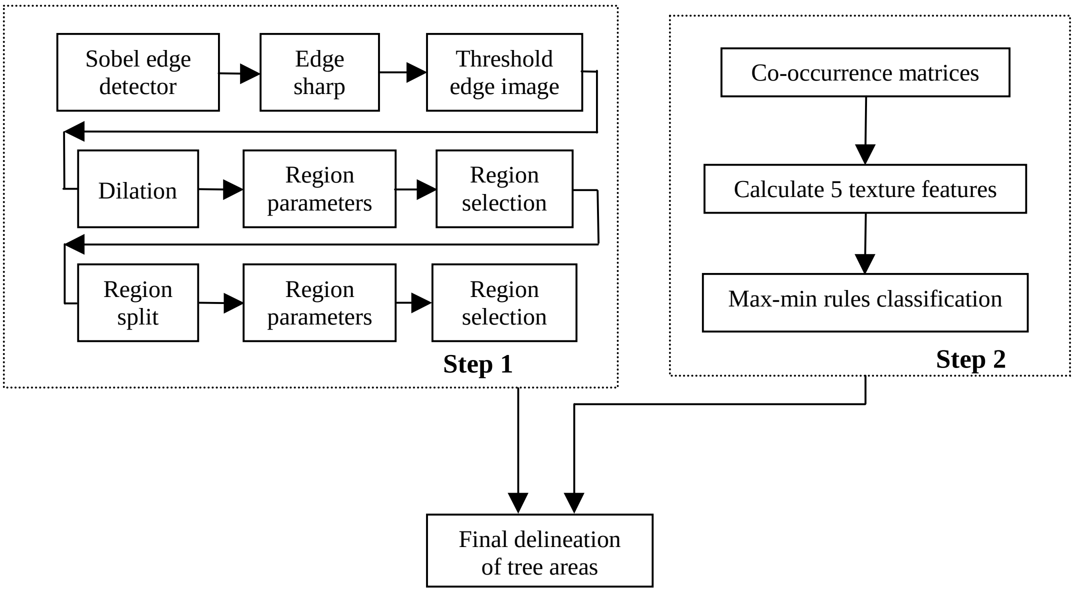


Figure 6: Image segmentation and classification for delineation tree areas

The steps of image segmentation are illustrated in the step1 in Figure 6. The process of edge detection, morphological functions and analyzing region parameters are described as in 2.4.1. Region split is an important step in delineation tree areas, since trees are usually close to each other or to other objects, some regions remain connected. The “region split” function is used to split the connected regions and eliminate undesirable regions. Detailed can be found in [20].

This task splits the regions into new smaller homogenous regions based on peak-valley analysis of the associated pixel intensity histogram. For each region, a pixel intensity histogram is created. Only pixels that lie within the region are used to compute the histogram. The task automatically selects values and use them to threshold the pixels in the region. Contiguous pixels that fall within the same threshold boundaries form new regions. The threshold selection is based on a histogram peak-valley analysis technique proposed by Beveridge for local histogram-based segmentation [21].

2.4.2.2 Texture analysis and image classification using co-occurrence matrices

Co-occurrence matrices has been described in image processing literature by a number of names including gray-tone spatial-dependence matrices [22, 23]. It has been widely used in texture analysis and classification. The co-occurrence matrix in it core is a two-dimensional histogram of the occurrence of pairs of intensity values in a given spatial relationship. Co-occurrence matrices are based on the relative frequencies $p(i, j)$ with which two pixels with a specified separation occur in the image, one with gray level I and another with gray level J. The separation is usually specified by distance vector $\phi(d,\alpha)$. Pixel distance *d* and angular orientation α are parameters of a particular co-occurrence matrices. Using different parameters, several matrices can be derived. The matrices obtained are then used for extraction of texture features.

For the image of *N* gray level, co-occurrence matrices *cm* can be obtained by estimating the pairwise statistics of pixel intensity. The size of *cm* is determined by the number of gray level *N* in the input image. The matrices *cm* are a functions of the angular relationship between the pixels as well as a function of the distance between them. The matrices can be illustrated as following:

$$cm(d,\alpha) = [\,p(i,j\,|\,d,a)]$$

After introducing the symmetry [24], we can only consider the angular α up to 180° rotation. The value *d* normally be chosen as 1. An example of a 4x4 image with four gray levels and the computation of the co-occurrence matrices for *d=1* and α varying from 0° to 135° by 45° increments are shown in Figure 7:

0	0	1	1
0	0	1	1
0	2	2	2
2	2	2	3

(a) image of 4 gray levels

$$\begin{array}{r|cccc} 0 & 1 & 2 & 3 & \text{(Gray levels)} \\ \hline 0 & \#(0,0) & \#(0,1) & \#(0,2) & \#(0,3) \\ 1 & \#(1,0) & \#(1,1) & \#(1,2) & \#(1,3) \\ 2 & \#(2,0) & \#(2,1) & \#(2,2) & \#(2,3) \\ 3 & \#(3,0) & \#(3,1) & \#(3,2) & \#(3,3) \\ \text{(gray levels)} & & & & \end{array}$$

(b) general form of any co-occurrence matrix (#(i,j) is the number of times gray level i and j have been neighbors)

$$cm(1,0^{\circ}) = \begin{bmatrix} 4 & 2 & 1 & 0 \\ 2 & 4 & 0 & 0 \\ 1 & 0 & 6 & 1 \\ 0 & 0 & 1 & 2 \end{bmatrix} cm(1,45^{\circ}) = \begin{bmatrix} 4 & 1 & 0 & 0 \\ 1 & 2 & 2 & 0 \\ 0 & 2 & 4 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} cm(1,90^{\circ}) = \begin{bmatrix} 6 & 0 & 2 & 0 \\ 0 & 4 & 2 & 0 \\ 2 & 2 & 2 & 2 \\ 0 & 0 & 2 & 0 \end{bmatrix} cm(1,135^{\circ}) = \begin{bmatrix} 2 & 1 & 3 & 0 \\ 1 & 2 & 1 & 0 \\ 3 & 1 & 0 & 2 \\ 0 & 0 & 2 & 0 \end{bmatrix}$$

Figure 7: Co-occurrence matrices for four given distance vectors (taken from Haralick[1973])

In texture classification, individual elements of the co-occurrence are rarely used. Instead, features are derived from the matrix. A large number of textural features have been proposed starting with the original fourteen features described by Haralick, however only some of these are in wide use. The features we used are listed as following:

1. Inverse Difference Moment

$$f_1 = \sum_{i,j} P(i, j) / 1 + (i - j)^2$$

2. Contrast

$$f_2 = \sum_{i,j} \delta_{ij}^2 P(i, j)$$

3. Entropy

$$f_3 = -\sum_{i,j} P(i, j) \log P(i, j)$$

4. Correlation

$$f_4 = \sum_{i,j} (i - \mu_i)(j - \mu_j) P(i, j) / \sigma_i \sigma_j$$

5. Energy (angular second moment)

$$f_5 = \sum_{i,j} P(i, j)^2$$

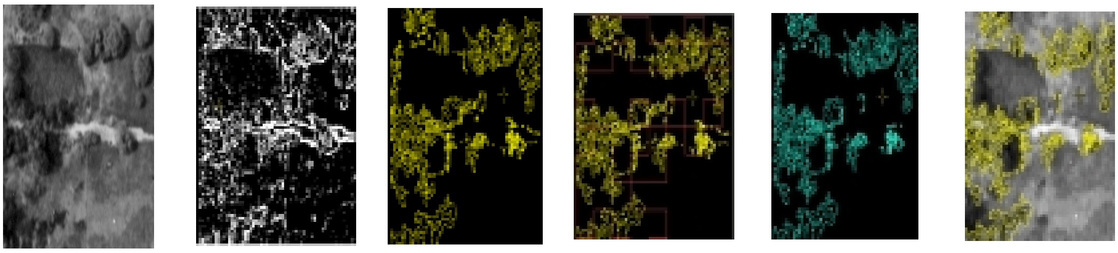
μ_i and μ_j are the means and σ_i and σ_j are the standard deviations of i and j respectively. Rotating α, there are 4 values for every texture feature. Then the minimum and maximum values of the texture features can be obtained.

Since we only want to find tree areas, the number of samples for tree category is small. We use a min-max decision rule for classification of the image based on their texture features. The procedure is repeated for all the image blocks in the image. The decision rule is described by the following equation. If the equation is satisfied, the processed image block *j* can be assigned as catalogue *k.*

$$\prod_{n=1}^{N} \left(\frac{1}{a_{nk} - b_{nk}}\right) \geq \prod_{n=1}^{N} \left(\frac{1}{a_{nj} - b_{nj}}\right) \tag{4}$$

b_{nk}, a_{nk}, b_{nj} and a_{nj} define the minimum and maximum texture feature values of the training samples and processed image blocks. n is the number of texture features.

Figure 8 illustrates the delineation tress areas using the steps in Figure 6. Using image segmentation method described in Section 2.4.2.1, most of the tree areas can be delineated. Some areas obtained are not correct. Using the texture analysis and image classification method introduced in Section 2.4.2.2, the original image can be classified to supply image blocks which are possible tree areas.



(1) (2) (3) (4) (5) (6)

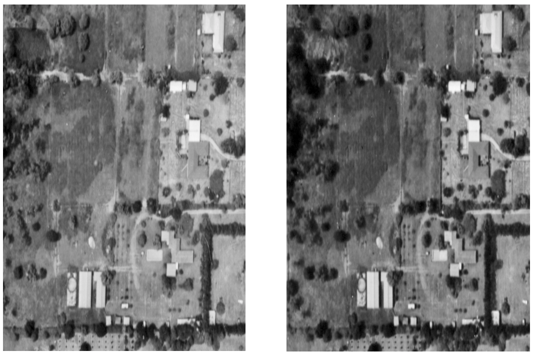
(1) Testing image area, (2) threshold edge image, (3) Tree areas from segmentation (4), Two results from Figure 6, (5) Final delineation of tree areas, (6) Extracted tree areas overlaid on image

Figure 8: Delineation tree areas using steps in Fig.7

The results from two steps are overlaid together and shown in Figure 8(4). Combing the results from two steps, the final tree areas can be illustrated in Figure 8(5). Some small areas which are extracted incorrectly, are eliminated. Figure 8(6) is the extracted tree areas overlaid on the image.

2.5 Test and Results

Figure 9 illustrates a pair of aerial images with 630 × 714 pixels in the row and column directions respectively. The scale of image is 1:3437. The flying height is 519 meter. The focal length of the camera is 153mm and pixel size is 100 μm . Figure 10 illustrates the disparity map obtained from stereo image matching in the digital photogrammetric workstation. The disparity map is further processed to obtain the house and tree areas illustrated in Figure 11, using the method described in Section 2. 3.



Left

Right

Figure 9: Stereo image pair

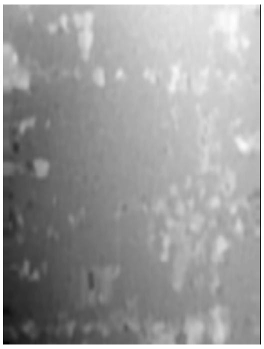


Figure 10: Disparity map

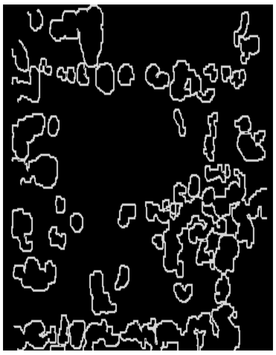


Figure 11: Outlines of house and tree areas

The left image has then been processed to locate houses and to separate them from trees by the method described in Section 2.4.1. Figure 12 shows the regions obtained by region growing in step 1 in Figure 3.

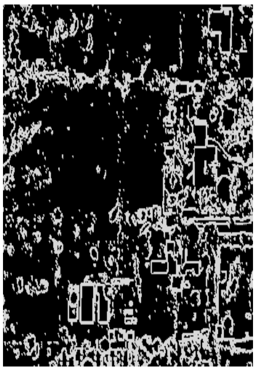


Figure 12: Results of region growing

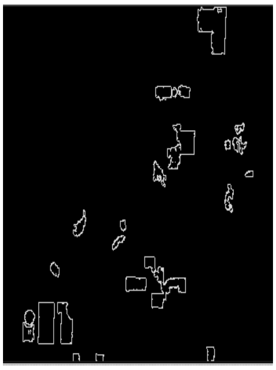


Figure 13: Definition of regions after the application of region parameters

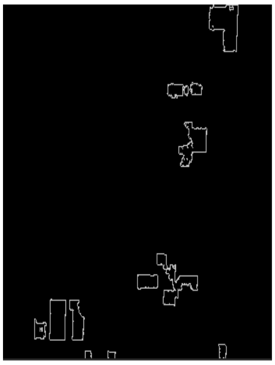


Figure 14: Houses obtained by step 1 in Figure 3

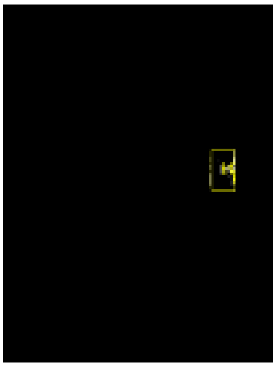


Figure 15: The dark roof area

Based on the five feature values of regions, most of the houses can be defined, as illustrated in Figure 13. For each region in Figure 13, the corresponding small image region in the original image can be identified.

Based on an analysis of the histogram of the orientations of edges, areas whose edges are not considered to be those of houses can be eliminated. The extracted house areas are again illustrated in Figure 14. Figure 15 shows the area of dark roof extracted by step 2 in Figure 3, while Figure 16 shows the final determination of the houses. Figure 17 is the image in which the final extracted house areas are overlaid on the original image. This figure shows that the houses are well delineated.

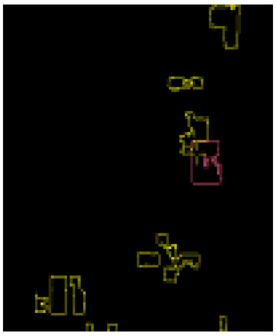


Figure 16: Final delineation of house

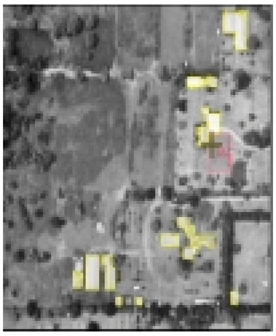
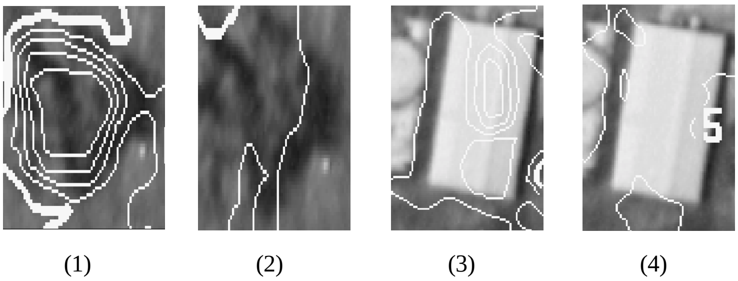


Figure 17: Extracted roofs overlaid on the image

Using the compound information from the analysis of image matching and 2D image segmentation, some digital elevation points which were initially located on the tops of houses and trees have been interpolated onto the ground. Figure 18 illustrates the contour changes in these areas.



(1) Contour in tree area from matching, (2) Corrected contours, (3) Original contours in house area, (4) Corrected contours

Figure 18 Contours obtained from the image matching and corrected contours based on the method described in the paper

Figures 19 and 21 illustrate the DEM and 3D perspective view derived directly from stereo image matching produced by VirtuoZo A more accurate DEM and 3D perspective view produced by the method in this paper are shown in Figures 20 and 22.

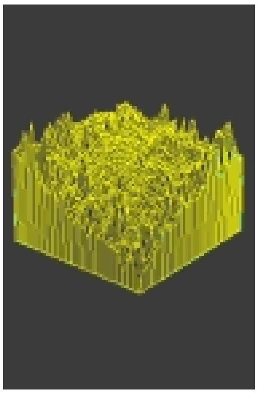


Figure 19: DEM from matching

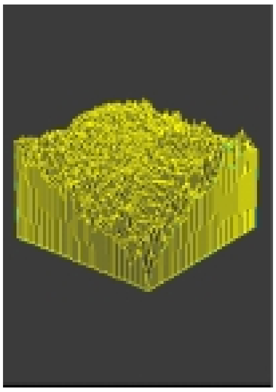


Figure 20: DEM from the combined method

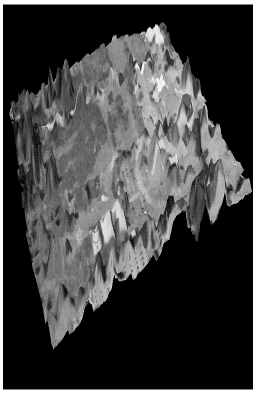


Figure 21: 3D perspective view from matching

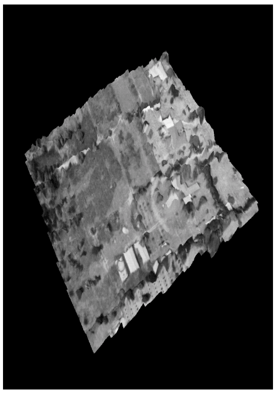


Figure 22: 3D perspective view derived by the combined method

2.6 Conclusion

The method described in this paper combines image matching and image analysis methods, which enables the location of most of the house and tree areas in the test images. The image segmentation and classification methods overcome the weakness of co-occurrence matrices that is it does not consider the shapes of gray level primitives. These extracted house and tree areas are important information for 3D terrain reconstruction and ensure that points are only measured on the natural terrain. The method leads to more accurate determination of elevations from overlapping digital aerial images than the DSM determined only by image matching, since it avoids errors caused by man-made or natural surface features.

The method can also locate dark roofs. The disadvantage is its inability to exactly locate the boundary of dark roofs in cases when the roof of a house is not of regular shape. Since the classification result of co-occurrence matrices are dependent on chosen training sample and the size of the processed image block, further research is needed to find a more reliable method for image classification. The method will also be tested on other scales and different images.

References

[1] Zong J., Li J. & Schenk T. (1992). Aerial Image Matching Based on Zero-Crossings. *International Archives of Photogrammetry and Remote Sensing,* 29(3):144-150.

[2] Tönjes R. (1996). Knowledge Based Modelling of Landscapes. *International Archives of Photogrammetry and Remote Sensing,* 31(3):868-873.

[3] Baltsavias E., Mason S. & Stallmann D. (1995). Use of DTMs/DSMs and Orthoimages to Support Building Extraction. Automatic Extraction of Man-made Objects from Aerial and Space Images. Birkhauser Verlag, Basel, pp 199-210.

[4] Henricsson O., Bignone F., Willuhn W., Ade F., & Kuebler O. (1996). Project AMOBE: Strategies, Current Status and Future Work. *International Archives of Photogrammetry and Remote Sensing,* 31(3):321-330.

[5] Miller S., Walker S. & Walsh M.C. (1994). Digital Photogrammetry at the Ordnance Survey of Ireland. Report by Helava and Associates.

[6] Marr D. & Poggio T. (1979). A Computational Theory of Human Stereo Vision. *Proceedings of Royal Society of London,* B204, pp. 301-328.

[7] Barnard S. T. & Thompson W. B. (1980). Disparity Analysis of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence,* PAMI-2(4):333-340.

[8] Medioni G. & Nevatia R. (1985). Segment-based stereo Matching. *Computer Vision, Graphics, and Image Processing,* 31: 2-18.

[9] Hannah M. J. (1989). A System For Digital Stereo Image Matching. *Photogrammetric Engineering and Remote sensing,* 55(12): 1765-1770.

[10] Li M. (1991). Hierarchical Multipoint Matching. *Photogrammetric Engineering and Remote Sensing,* 57(8): 1039-1047.

[11] Wu X. (1995). Grey-based Relaxation for Image Matching. *3^{rd} Conference on Digital Image Computing: Techniques and Applications*, December, 1995, Brisbane, Australia.

[12] Gonzalez R. C. & Woods R. E. (1992). *Digital Image Processing*. Addison-Wesley, U. S. A.

[13] Bennamoun M. & Boashash B. (1997). A Probabilistic Approach for Automatic Parameters Selection for the Hybrid Edge Detector. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, E80-A*(8): 1423-1429.

[14] Levine M. & Shaheen S. (1981). A Modular Computer Vision System for Image Segmentations. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI* 3(5): 540-554.

[15] KBVision System Task Reference Manual (1996). Amerinex Applied Imaging, Inc.

[16] Vohra V. K. & Dowman I. J. (1996). Automatic Extraction of Large Buildings From High Resolution Satellite Images For Registration with a Map. *International Archives of Photogrammetry and Remote Sensing,* 31(3):903-905.

[17] Wilson P. A. (1997). Rule-Based Classification of Water in Landsat MSS Images Using the Variance Filter. *Photogrammetric Engineering & Remote Sensing*, 63(5):485-491.

[18] Reed T. R. (1993). A review of Recent Texture Segmentation and Feature Extraction Techniques. *Computer Vision, Graphics and Image Processing: Image Understanding,* 57(3):359-372.

[19] Kanai Y. (1998). Image Segmentation Using Intensity and Color Information. *Visual Communications and Image Processing'98,* 3309:709-720.

[20] Lu, Y., Kubik, K. & Bennamoun, M. (1998). An Accurate Approach to Localize House Areas for 3D Terrain Reconstruction. *9th Australia Remote Sensing and Photogrammetry Conference,* Syndey, Australia , 20-24 July.

[21] Beveridge, J. R. et al. (1989). Segmenting Image Using Localized Histograms and Region Merging. *International Journal of Computer Vision*. 2:311-347.

[22] Haralick, R. M., Shanmugam, K. S., and Dinstein, I. (1973) Textural Features for Image Classification. *IEEE International Conference on Systems, Man, and Cybernetics,* Vol. SMC-3, No.6, pp610-621.

[23] Conners, R. W., Harlow, C. A. (1980). A Theoretical Comparison of Texture Algorithms, *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. PAMI-2, No. 3, pp.204-222

[24] Burns, I. and Smith, G. (1996) MeasTex Version 1.0: A Framework for Measuring the Perfomance of Texture Classification Algorithms. University of Queensland, http://www/cssip.elec.uq.edu.au/~guy/meastex/meastex.html