國防科技學術合作協調小組研究計畫成果報告

影像關聯性比對演算法則之精度提昇研究

Improvement on the Accuracy of Image Correlation Matching

計畫編號：NSC91-2623-7009-004

執行時間：91 年 1 月 1 日至 91 年 12 月 31 日

計畫主持人：陳 稔　教　授

共同主持人：賈叢林 教　授

執行單位：國立交通大學資訊工程系

目錄

第一章 緒論

本計畫係為上年度計畫「應用於航照圖之關聯性比對演算法的效能評估」之延伸，目的在探討如何提高前計畫中直接用影像灰度值作相關度量測的比對精確度。

由於拍攝當時攝影機視點的不同，二張影像間存在的幾何關係通常以透視投影轉換(perspective transformation)來描述，但由於在相機未校正的情況下並不知道相機與物體間的距離及方位，處理此問題並不容易，因此，許多研究將此問題簡化為仿射轉換(affine transformation)[1-4]，在仿射轉換下，所要處理的是旋轉(rotation)、 平移(translation)、尺度(scale)及 歪斜(shear)問題；然而，這種簡化對於真實拍攝具有較大變形的影像並無法處理，因此，為求能處理更多樣的影像並獲得較精確的結果，我們有必要將變形的問題考量的更周延一點，對於空照圖而言，由於具有與地面景物距離遠、景物複雜等特性，使得在此應用範疇內，可以假設影像間的幾何關係為平面透視投影(planar perspective transformation)。在沒有解決幾何變形問題前就直接作相關度量測(Correlation measurement)，將使得比對視窗內點與點間的對應關係不正確而造成比對的失敗或錯誤，因此，為提高比對的正確性及精確度，有必要對初始影像作前置處理(Pre-processing)以消除上述之影響因素，也就是先求取二張影像間的相對幾何關係，將其中一張影像依此相對幾何關係作轉換後再來作相關度的量測。

一般而言，求取影像間相對幾何關係的做法，首先在二張影像上利用特定的運算元(operator)來尋找特徵，在建立這些特徵的對應關係後，即可藉以求取這二張影像間的相對幾何關係。較常用的特徵為輪廓線(contour),邊緣(edge),邊角(corner)及特徵點(point)等，然而由於以輪廓為基礎的(contour-based)方法需要有一完整的輪廓線，因此，當比對物件間其中一個有被遮蔽的情形時並無法處理，此方式另有一缺點為需要先將影像作分割(segmentation) 後再

以中介表示法(intermediate representation)(如 Fourier descriptor)來表示輪廓線[3-6]，然而針對複雜背景的影像作分割並不是一件容易的事；相對而言，由於在影像上偵測邊緣, 邊角及特徵點等特徵受到比較小的限制，因此，成為許多研究用來作比對的選擇。

由於二張影像間存在著平面透視投影關係，在作特徵比對時不能再直接以影像的灰度值作相關度量測，取而代之的，是以對平移、旋轉、尺度及視點不同所造成的幾何變形不變(invariant)的特徵向量(feature vector)來描述其週遭的灰度值變化，並定義一比對準則來據以進行比對。然而，初始比對的結果通常具有混淆(ambiguous)或不正確的問題，因此，必須引入一個機制來確保比對的結果具有一對一的對應關係及比對的正確性。

綜合言之，本計畫的目的在給予二張存在著平面透視投影關係的影像，希望提出一個對平移、旋轉、尺度及視點不同所造成的幾何變形不變的特徵比對演算法，以找出足夠數量的特徵，據以求取影像間的轉換參數，再將影像轉換到相同的視點後作為前計畫的相關度量測之用，最後則會引入評估比對結果精確度的機制，以量化表示比對結果之優劣。

本研究計畫的進行，首先將以 Gabor filter 為核心，深入探討如何利用其對區域(local)訊號變化的優異量測能力來進行特徵點的偵測，及利用其多重解析度分析能力特性，來進行特徵點的粹取及用以輔助比對的進行。其次，在二張影像中分別獲得一組特徵點後，必須建立這二組特徵點間的對應關係以進一步求取二張影像間的相對幾何關係。此部分，常見的做法是先將所有點作初始比對，再引入一個離散鬆弛法(discrete relaxation)的機制，在定義一些限制(constrain)及相容函數(compatibility function)後，以分群(clustering) 的方法來確保比對的結果具有一對一的對應關係及比對的正確性。在本研究中，我們將提出一快速有效同時決定點對應及轉換矩陣的方法。

本研究報告章節之安排為:第二章進行 Gabor filter 的特性探討及介紹特

徵點粹取的方法，第三章說明特徵點比對及轉換矩陣之求取，第四章包含一些實驗驗證，最後，第五章為結論。

第二章 Gabor filter 的重要特性及特徵點粹取

2.1 Gabor filter 的重要特性

A Gabor function is a Gaussian-modulated complex exponent function that provides the best spatial and frequency information of the signal. The general form of the Gabor function is given as [7-8]

$$g^{s,N,l}(x,y)=\frac{1}{2\pi\alpha\sigma_s^{\,2}}\exp\{-[(\frac{x'}{\sigma_s})^2+(\frac{y'}{\alpha\sigma_s})^2]\}\exp(jw_s x') \tag{1}$$

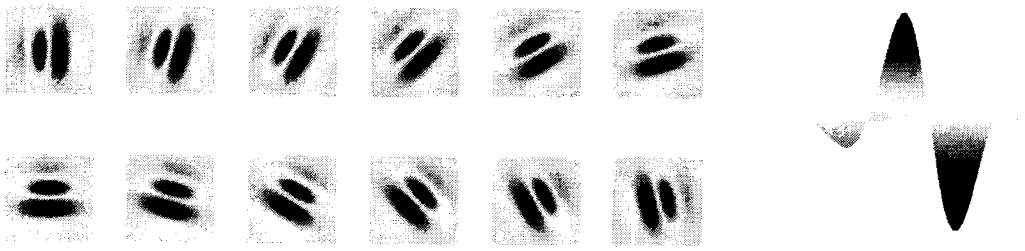
where

$$\begin{bmatrix}x'\\y'\end{bmatrix}=\begin{bmatrix}\cos\theta_l & \sin\theta_l\\-\sin\theta_l & \cos\theta_l\end{bmatrix}\begin{bmatrix}x\\y\end{bmatrix}$$

with θ_l being the orientation parameter; σ_s and $\alpha\sigma_s$ are the Gaussian window size parameters (α = 1 is assumed here); ω_s is the spatial frequency parameter. A normalization condition is usually imposed on the parameters σ_s and ω_s such that $\sigma_s w_s = N\pi/4$ (Here $4\sigma_s$ is used to approximate the Gaussian window size, which is also called the filter size) for all scale indices *s*.

Gabor functions form a complete but non-orthogonal basis set. Signal expansion using this basis provides a localized frequency description that is useful for image encoding and compression [9]. Gabor functions can be extended to Gabor wavelets used for image representation [10], and object recognition [11]. If Gabor function is used for extracting the object features, then the even-symmetric component $g_{even}^{s,N,l}(x,y)$ can be used to extract the bar (or line) features and the odd-symmetric component $g_{odd}^{s,N,l}(x,y)$ is for extracting the edge features [12].

Since we want to extract feature point along the object boundary, we shall use the odd-symmetric Gabor function from now follow on. Fig. 1 depicts such a set of Gabor filters.



(a) (b)

Fig. 1 (a) A set of odd-symmetric Gabor filters. **(b)** The plot of a typical odd-symmetric Gabor filter with $N = 4$.

Let $I(x,y)$ be the input image function. For multiple scales s, $s \in \{1, 2, 3, ..., S\}$, and multiple orientations $\theta_l, \theta_l = l \times \Delta\theta$, $l = 1, 2, ..., L$ (π is a multiple of $\Delta\theta$), the filter responses or outputs are given by the convolution operations:

$$R^{s,N,l}(x,y) = I(x,y) * g_{odd}^{s,N,l}(x,y) \tag{2}$$

In the following, we give the important properties of Gabor-filtered image that lead to the robust feature points.

We first consider the effect of 2-D rotation on the filter response.

Property 1: Let $\hat{R}^{s,N,l}(x,y)$ and $R^{s,N,l}(x,y)$ be the filter responses to the images $\hat{I}(x,y)$ and $I(x,y)$. If $\hat{I}(x,y)$ is obtained from $I(x,y)$ by a rotation through an angle ϕ in the counter-clock direction, i.e., $\hat{I}(x,y) = I(x',y')$ for points (x,y) and (x',y') that are related by

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \tag{6}$$

Then $$\hat{R}^{s,N,l}(x,y) = R^{s,N,\theta_l+\phi}(\bar{x},\bar{y}) \tag{7}$$

where $$\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

Proof:

$$\hat{R}^{s,N,l}(x,y) = \iint \hat{I}(u,v) g_{odd}^{s,N,l}(x-u, y-v)\,du\,dv$$

$$= \iint I(u',v') g_{odd}^{s,N,l}(x-u, y-v)\,du\,dv$$

where

$$\begin{pmatrix} u' \\ v' \end{pmatrix} = \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} u \\ v \end{pmatrix}$$

and

$$g^{s,N,l}(x-u, y-v) = \frac{1}{2\pi\sigma_s^2} \exp\{-\frac{1}{2}[(\frac{x'}{\sigma_s})^2 + (\frac{y'}{\sigma_s})^2]\} \cdot \exp(jw_s x')$$

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta_l & \sin\theta_l \\ -\sin\theta_l & \cos\theta_l \end{bmatrix} \begin{bmatrix} x-u \\ y-v \end{bmatrix}$$

$$= \begin{bmatrix} \cos\theta_l & \sin\theta_l \\ -\sin\theta_l & \cos\theta_l \end{bmatrix} \begin{pmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{pmatrix} \left\{ \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} - \begin{pmatrix} u' \\ v' \end{pmatrix} \right\}$$

$$= \begin{bmatrix} \cos(\theta_l + \phi) & \sin(\theta_l + \phi) \\ -\sin(\theta_l + \phi) & \cos(\theta_l + \phi) \end{bmatrix} \begin{pmatrix} \bar{x} - u' \\ \bar{y} - v' \end{pmatrix}$$

$$\hat{R}^{s,N,l}(x,y) = \iint I(u',v') g_{odd}^{s,N,l'}(\bar{x}-u', \bar{y}-v')\,du'\,dv'$$

$$= R^{s,N,l'}(\bar{x}, \bar{y}) \quad with \quad \theta_{l'} = \theta_l + \phi$$

where

$$\begin{pmatrix} \bar{x} \\ \bar{y} \end{pmatrix} = \begin{pmatrix} \cos\phi & -\sin\phi \\ \sin\phi & \cos\phi \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}$$

The orientation associated with the maximum Gabor filter response among the L orientations is called the principal orientation. It is perpendicular to the direction of the highest changing rate of the intensity in the neighbor around the object point.

Property 2: Let $\hat{R}^{s,N,l}(x,y)$ and $R^{s,N,l}(x,y)$ be the Gabor filter responses to input images $\hat{I}(x,y)$ and $I(x,y)$. If the two input images are related by a scale factor such that $\hat{I}(x,y) = I(kx,ky)$, then there is a correspondence between their responses given by

$$\hat{R}^{s,N,l}(x,y) = R^{s',N,l}(kx,ky) \quad \text{with} \quad \sigma_{s'} = k\sigma_s \tag{8}$$

Proof: $\hat{R}^{s,N,l}(x,y) = \iint \hat{I}(u,v) g_{odd}^{s,N,l}(x-u, y-v)\,du\,dv$

$$= \iint I(ku, kv) g_{odd}^{s,N,l}(x-u, y-v)\,du\,dv$$

$$= \iint I(u', v') g_{odd}^{s,N,l}(\frac{\bar{x}}{k} - \frac{u'}{k}, \frac{\bar{y}}{k} - \frac{v'}{k}) d(\frac{u'}{k}) d(\frac{v'}{k})$$

(with $u' = ku$, $v' = kv$, $\bar{x} = kx$, $\bar{y} = kv$)

$$= \iint I(u', v') g_{odd}^{s',N,l}(\bar{x} - u', \bar{y} - v')\,du'\,dv' \quad \text{with} \quad \sigma_{s'} = k\sigma_s$$

$$= R^{s',N,l}(kx, ky)$$

We compute the responses $R^{s,N,l}(x,y)$ at each point (x, y) for $s = 1, 2, \ldots, S$ and $l = 1, 2, .., L$ (N is fixed). Then the total energy value at point (x, y) at a scale s is defined as

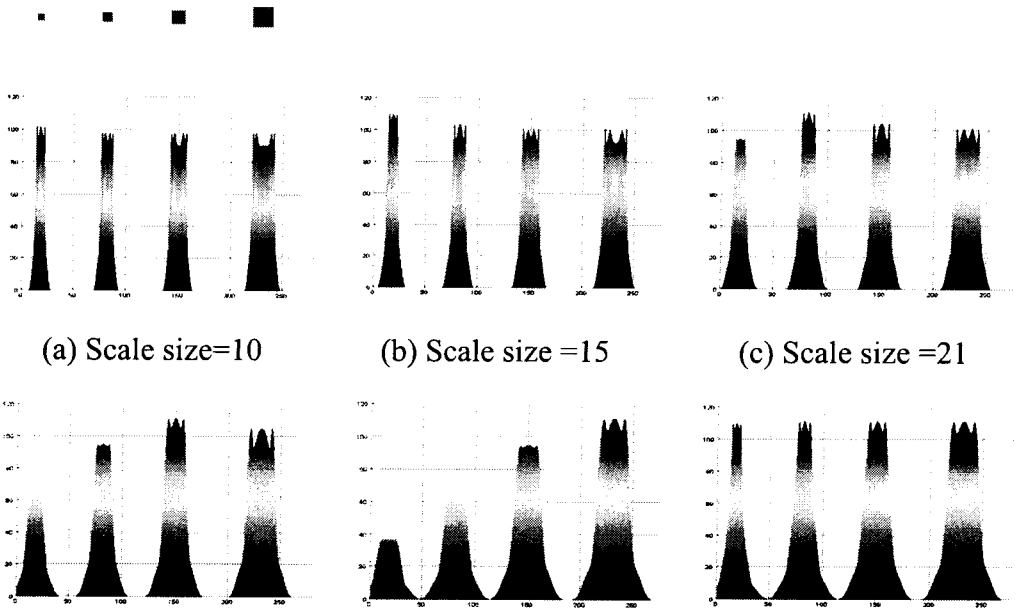
$$E^{s,N}(x,y) = \sum_{l=1}^{L} | R^{s,N,l}(x,y) | \tag{9}$$

Then the maximum energy value is calculated at point (x,y) with respect to all scales $\{1,2,...,S\}$ (N is fixed)

$$E^{N}(x,y) := E^{s^*,N}(x,y) = \max_{s} \{E^{s,N}(x,y)\} \tag{10}$$

The 2D array of the maximum energy values constitutes a maximum energy map. The point with a strictly local maximum energy is called a dominant point and the scale s^* associated with the maximum energy is called the principal scale. Under proper filter design condition, it will be shown that there exist the dominant points in the filtered image.

In Fig. 2 the multi-scale energy maps are obtained through the application of the same set of Gabor filters with different filter scales to four different squares. Notice the squares yield different energy maps at the multiple filter scales. However, the dominant points in the maximum energy maps obtained from the multi-scale energy maps have the nearly equal energy values, as indicated by Properties 2.



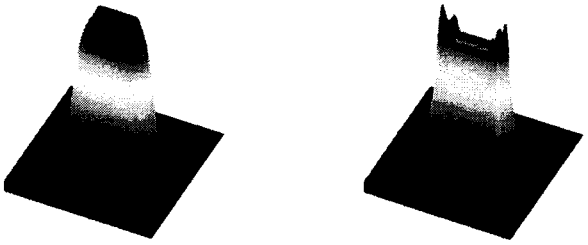
(a) Scale size=10 (b) Scale size =15 (c) Scale size =21

(d) Scale size =30 (e) Scale size =42 (f) Maximum energy maps

Fig. 2 (a)-(e) The side view of the energy maps at five different scales. **(f)** The maximum energy maps at the principal scale (The image containing four squares has 200×100, the sizes of four squares : 7, 10, 14, 20. Filter size = 10, 15, 21, 30, 42. N = 3).

The Gabor filter output energy $E^{s,N}(x,y)$ for an input object image $I(x, y)$ is governed by the filter size parameter σ_s and the filter sinusoidal lobe parameter N. We shall study the behaviors of the filtering with respect to each of these two parameters below. First, for a fixed filter size parameter σ_s, we want to know how the relative magnitudes of the filter output energy at the object points vary with the lobe parameter N ? To gain some insight, let us consider a particular case in the following property.

Property 3. Consider the Gabor filter output energy $E^{s,N}(x,y)$ for a binary rectangular object image with a fixed filter size parameter σ_s and a varying lobe parameter N. Along a longer rectangle side, the maximum energy value occurs at consecutive points near the middle point of the side for $N \leq 2$; while for $N > 2$, the maximum energy value occurs at a single point near the vertex of the side. (Please refer to Fig. 3 for an illustration)

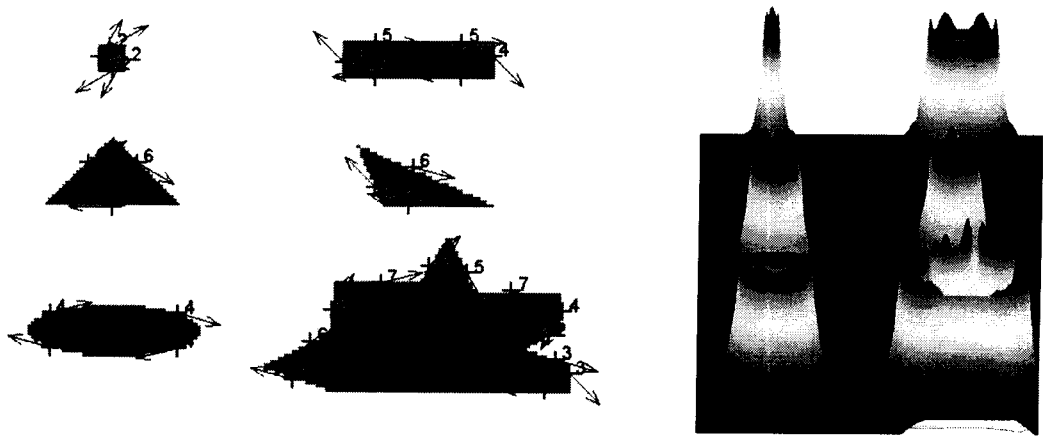


(a) (b)

Fig. 3 (a) There are no dominant points in the maximum energy map for a rectangular object image of size 8×33 for N=1.5. **(b)** There are dominant points in the maximum energy map for N=3.

Next, let the value of N be fixed. Then the filter output energy at an object edge point changes as filter size parameter σ_s changes, since the energy is a continuous function of the filter size σ_s. When σ_s is too large, the output energy value becomes small. When σ_s is equal to a proper value, which depends on the local object structure around the object point under consideration, the output energy value becomes maximum. When the parameter N is greater than 2, the filter contains more lobes and it is more responsive to a varying local object structure. Thus, the filter will reveal the object points with the local maximum output energy. These points are often isolated and can be used as the feature points for object representation. Several objects of simple polygonal shape are shown in Fig. 4(a) with isolated dominant points. Fig. 4(b) shows the associated maximum energy maps. From these figures, we can see the maximum output energy values occur at different principal size scales.

When the object shape becomes more complex, then all structure patterns in the neighborhood of an object contour point will jointly determine the principal scale and the existence of the isolated dominant point.



(a) (b)

Fig. 4 (a) Various simple polygonal objects with the detected dominant points marked with individual principal scales and principal orientations. **(b)** The associated maximum energy maps.

2.2 特徵點粹取

Based on the above properties of the Gabor filtered image, we outline below an algorithm for extracting dominant points from a given image.

Algorithm for dominant point extraction:

(1) Choose a set of Gabor filters with filter parameters N = 3 and L = 12.

(2) Determine a set of filer size scales based on the structure pattern of the object under consideration. Apply the multi-scale Gabor filters to the image to obtain the energy maps at the multiple scales and merge them into a single maximum energy map by retaining the maximum value among them.

(3) Construct the maximum energy histogram and choose a lower bound on the energy value for a candidate dominant point in the upper T %. (T = 10 in our case)

(4) Partition the image into non-overlapping blocks of m×m pixels each (m=7 in

the experiments).

(5) Find one candidate dominant point having the maximum energy in each block. Then check if each candidate dominant point obtained so far is also a local maximum in a neighborhood of size m×m centered at the point.

第三章 特徵點比對及轉換矩陣之求取

In general, there are two classes of algorithms for point matching. The algorithms in the first class use the neighborhood around the feature points to derive some properties for use in the initial point matching stage. These methods can reduce the search space for final point matching, but the viewing parameters, including rotation, scaling, translation and perspective deformation, need to be solved in the initial point matching stage. Then, a mechanism, for example, relaxation [13-15] and maximum clique [16], is imposed to eliminate the false matching pairs. It is generally not clear whether the relaxation process will be converged. Still, the exponential complexity of finding maximum clique is unacceptable.

On the other hand, the algorithms in the second class are the point pattern matching method without performing the initial matching process. These cover the relaxation-based [17-18], graph matching [19] and neural network [20] method. Most of techniques are computationally exhaustive in the worst-case theoretical sense. Some researchers attempt to circumvent the high complexity by estimating the matching pairs and the transform parameters simultaneously. Their basic idea is that three pairs of non-collinear points determine a unique affine transformation between two images of plane (or five points in projective space [21], depending on the number of parameters of the transformation space). Thus, by choosing a sufficient number of triplets of points, and enumerating all possible matching triplets in the other point set, one is possible to find a good match. The intuitive algorithm is to use the full search of all possible combinations of triplets of points. If there are *m* points in the sensed image and *n* points in the reference image, it needs to examine a total of $C_3^m \times C_3^n \times 3!$ possibilities. Although Lamdan [22] used a hashing table to store the pre-computed possibilities of C_3^m item such that the possibilities can be reduced to $C_3^n \times 3!$.

However, in addition to the exhaustive memory requirement, it is also sensitive to noise. Cheng [23] combined the relaxation technique by computing the compatibility coefficient of the triplet. The concept is also based on a triplet of points.

In chapter 2, a multi-scale Gabor filtering technique is used to detect feature points in the scale-orientation space. The feature points extracted are quite invariant to rotation, scaling, translation, and robust to the image noise and minor perspective deformation. The individual feature point also accompanies with the information of principal scale, principal orientation, and output responses to a bank of filters. A feature vector $\vec{V}(x,y)$ computed at the principal scale and cyclically reordered base on the principal orientation describs the feature point located at (x,y).

$$\vec{V}(x,y) = [R_{odd}^{(s_d,l_d)}(x,y), R_{odd}^{(s_d,l_d+1)}(x,y), \ldots R_{odd}^{(s_d,l_d+L)}(x,y)] \tag{11}$$

where s_d is the principal scale with the maximum energy in a set of filter scale sizes, l_d is the label of the principal orientation with the maximum absolute response among L line orientations at the principal scale. The feature vector representation is invariant to rotation and scaling, and thus can be used to measure the similarity between points with which the above viewing parameters be changed.

In this chapter, a computational efficient method for point-based image registration will be proposed. First, two Gabor feature points in the reference image together with their counterparts in the sensed image, each with its principal orientations, is employed to initially estimate the transformation. Then, an iterative closest matching pairs (ICMP) algorithm based on point-to-point alignment is iteratively applied to establish the corresponding point sets that are then used to refine the transformation result. Comparing with the algorithms using three pairs of point correspondences in most methods, the numbers of point sets need to be examined are reduced dramatically. Moreover, to avoid the full search of all possible combinations

of two selected points, some search strategies based on the pre-analyzed information of the points in the reference image are employed.

3.1 Estimation of transform parameters and determination of the matching pairs simultaneously

In this section, we will propose an iterative closest matching pairs registration algorithm to estimate the correct matching pairs and the transform parameters simultaneously. First, by using only two pairs of corresponding points together with their individual principal orientation, an initial estimation of transform parameters is determined. Then, an iterative refinement algorithm is imposed to refine these results. At each iteration, the algorithm determines the corresponding pairs by finding the closest matching pairs and uses them to re-compute the transform parameters. Comparing with the triplet-based methods, the computational complexity can be reduced dramatically.

3.1.1 Initial estimation of the transform parameters

Assume the distance between the object and the camera is far greater than the object size; the spatial transformation $\vec{X}' = M\vec{X}$ between corresponding points $\vec{X}, \vec{X}'$ in two partially overlapped images can be described by affine transformation, consisting of rotation, translation, scaling and shear, of the following form:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \tag{12}$$

Conventionally, at least three pairs of point correspondence are needed to solve the six affine parameters. Instead, we will show that only two feature point correspondences plus their individual principal orientations can also carry out this purpose. Thus, the computational complexity for point matching can be reduced dramatically.

As shown in Fig.5, considering given two pairs of point correspondence $(\vec{p}_1, \vec{q}_1)$ and $(\vec{p}_2, \vec{q}_2)$ with their individual principal orientations $\vec{e}_{p_1}, \vec{e}_{p_2}, \vec{e}_{q_1}, \vec{e}_{q_2}$, respectively. To derive the relationship between principal orientations of two corresponding points under affine transformation, let us first consider a new pairs of corresponding points $(\vec{p}_3, \vec{q}_3)$ intersected by the two principal orientations $(\vec{e}_{p_1}, \vec{e}_{p_2})$ and $(\vec{e}_{q_1}, \vec{e}_{q_2})$. Then, the directions of the two pairs of points $(\vec{p}_1, \vec{q}_1)$ and $(\vec{p}_3, \vec{q}_3)$ can be written as Eq.(13) and Eq.(14)

$$\vec{p}_1 - \vec{p}_3 = \vec{d}_{p_1 p_3} = \left|\vec{d}_{p_1 p_3}\right| \vec{e}_{p_1} \tag{13}$$

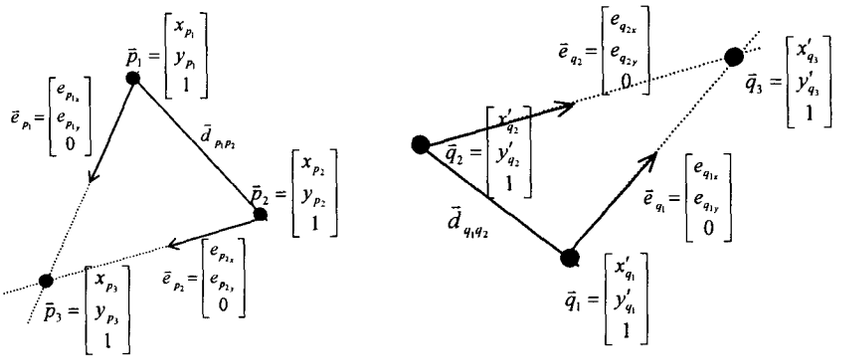


Fig.5 two pairs of point correspondence $(\vec{p}_1, \vec{q}_1)$ and $(\vec{p}_2, \vec{q}_2)$ with their individual principal orientations $\vec{e}_{p_1}, \vec{e}_{p_2}, \vec{e}_{q_1}, \vec{e}_{q_2}$ intersected at $(\vec{p}_3, \vec{q}_3)$

$$\vec{q}_1 - \vec{q}_3 = \vec{d}_{q_1 q_3} = \left|\vec{d}_{q_1 q_3}\right| \vec{e}_{q_1} \tag{14}$$

Under affine transform, we can write

$$\vec{q}_1 - \vec{q}_3 = M(\vec{p}_1 - \vec{p}_3) \tag{15}$$

Substituting Eq.(5) and (6) into (7)

$$\left|\vec{d}_{q_1 q_3}\right| \vec{e}_{q_1} = M \left|\vec{d}_{p_1 p_3}\right| \vec{e}_{p_1} \tag{16}$$

$$\frac{\left|\vec{d}_{q_1q_3}\right|}{\left|\vec{d}_{p_1p_3}\right|}\vec{e}_{q_1} = h\vec{e}_{q_1} = M\vec{e}_{p_1} \tag{17}$$

So, we can write Eq.(7) as the following matrix form

$$h\begin{bmatrix} e_{q_{1x}} \\ e_{q_{1y}} \\ 0 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} e_{p_{1x}} \\ e_{p_{1y}} \\ 0 \end{bmatrix} \tag{18}$$

Thus, from Eq.(2) and Eq.(8), the two pairs of corresponding points and their principal orientations form the following system of 8 linear equations.

$$\begin{bmatrix} x_{p_1} & y_{p_1} & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_{p_1} & y_{p_1} & 1 & 0 & 0 \\ x_{p_2} & y_{p_2} & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & x_{p_2} & y_{p_2} & 1 & 0 & 0 \\ e_{p_{1x}} & e_{p_{1y}} & 0 & 0 & 0 & 0 & -e_{q_{1x}} & 0 \\ 0 & 0 & 0 & e_{p_{1x}} & e_{p_{1y}} & 0 & -e_{q_{1y}} & 0 \\ e_{p_{2x}} & e_{p_{2y}} & 0 & 0 & 0 & 0 & 0 & -e_{q_{2x}} \\ 0 & 0 & 0 & e_{p_{2x}} & e_{p_{2y}} & 0 & 0 & -e_{q_{2y}} \end{bmatrix}\begin{bmatrix} m_{11} \\ m_{12} \\ m_{13} \\ m_{21} \\ m_{22} \\ m_{23} \\ h \\ h' \end{bmatrix} = \begin{bmatrix} x'_{q_1} \\ y'_{q_1} \\ x'_{q_2} \\ y'_{q_2} \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{19}$$

The system of simultaneous linear equations can be solved to reveal the eight unknowns $m_{11}, m_{12}, m_{13}, m_{21}, m_{22}, m_{23}, h, h'$ using SVD. The solution M is called as T_0.

3.1.2 Iterative Closest Matching Pairs (ICMP) algorithm

Let Q_c be a set of N_q points $\{q_1, q_2, \cdots, q_{N_q}\}$ in the sensed image, P_c be a set of N_p points $\{p_1, p_2, \cdots, p_{N_p}\}$ in the reference image. Starting from the initial estimate of the transformation T_0 described in section 3.1, the goal of *iterative closest matching pairs (ICMP)* is to iteratively build the corresponding point pairs and refine the transformation that best align the set of corresponding point. The algorithm transforms each point $q_i \in Q_c$ to the reference image coordinate system and seeks for a closest point $p_j \in P_c$ such that (q_i, p_j) is a matching pair by similarity measurement. Then, the matching pair (q_i, p_j) will be added to the *corresponding*

point set (CPS). The *CPS* is then used to compute the new transformation T_k where k is iterative number. The steps will be iteratively executed to refine the transformation T_k until no new point is added to the corresponding set. The details of the algorithm are as the follows:

Algorithm ICMP

Input:

1. *Two point set* $Q_c = \{q_1, q_2, \cdots, q_{N_q}\}$ *and* $P_c = \{p_1, p_2, \cdots, p_{N_p}\}$ *with their feature vectors.*
2. *The two selected matching pair* (q_{i_1}, p_{j_1}) *and* (q_{i_2}, p_{j_2}) *with their principal orientations* $(\vec{e}_{q_{i_1}}, \vec{e}_{p_{j_1}})$ *and* $(\vec{e}_{q_{i_2}}, \vec{e}_{p_{j_2}})$.

Output:

1. *The corresponding point set* CPS_f
2. *The transformation* T_f

Initial: k=0; $CPS_0 = \{(q_{i_1}, p_{j_1}), (q_{i_2}, p_{j_2})\}$

Begin

1. *Determine the initial estimate of the transformation* T_0 *computed by the two selected matching pairs* (q_{i_1}, p_{j_1}) *and* (q_{i_2}, p_{j_2}) *plus their principal orientations* $(\vec{e}_{q_{i_1}}, \vec{e}_{p_{j_1}})$ *and* $(\vec{e}_{q_{i_2}}, \vec{e}_{p_{j_2}})$ *using Eq.(9).*
2. *Repeat until* $CPS_k \bigcup CPS_{k-1} = CPS_{k-1}$
   a. *Build up the set of corresponding points* CPS_k

$$CPS_k = \bigcup_{i=1}^{N_q} (q_i, CC(T_{k-1}(q_i), P_c)$$

where $CC(T_{k-1}(q_i), P_c) = p_j$

if

(a) $\min_{p_j} |T_{k-1}(q_i) - p_j|$ *with* $|T_{k-1}(q_i) - p_j| \le d_k$ *where* $d_k = \frac{d_0}{2^k}$

and $S(q_i, p_j) >$ Threadshold

where $S(\vec{V}_{q_i}, \vec{V}_{p_j}) = \dfrac{\vec{V}_{q_i} \cdot \vec{V}_{p_j}}{\left\|\vec{V}_{q_i}\right\| \cdot \left\|\vec{V}_{q_j}\right\|}$ *with the vector factors* $\vec{V}_{q_i}, \vec{V}_{p_j}$ *of* q_i, p_j

(b) $(\vec{e}_{Q_k} \cdot \vec{e}_{Q_{k_1}})(\vec{e}_{P_j} \cdot \vec{e}_{P_{j_1}}) \geq 0$ *(Convexity compatibility check)*

b. Compute the new transformation T_k *using all points in* CPS_k

c. $k = k + 1$

end

3. $T_f = T_k$; $CPS_f = CPS_k$

End

3.2 Registration process

3.2.1 Learning stage (Off-line)

As described in the previous section, starting with the initial estimation of the transformation by two pairs of matching points together with their principal orientations, we can obtain the transformation by further applying the ICMP algorithm. However, when given *m* feature points in the reference image, there are C_2^m possible combinations of the two points. The full search is computational exhaustive. Therefore, we will propose some search strategies to accelerate the search time. The search strategies are planned using the following four priori-derived information. As a result, the two selected points (p_i, p_j) used to estimate the initial transform parameters T_0 could be ranked for choosing in advance.

1. Choose the points from the size of clusters as small as possible:

   Usually there are many points in the sensed image that are highly similar with the selected point in the reference image. This will result in the higher matching ambiguity, and thus reduce the matching performance. The merit of the clustering process is to group the points with highly similarity into clusters. As a result, we can use the size of clusters to adjust the distinctness of the selected points.

   For the clustering process, we first compute the similarity measurement,

defined as the normalized cross correlation computed by their feature vectors (Eq.10), between any two points p_i and p_j.

$$S(\vec{V}_{p_i}, \vec{V}_{p_j}) = \frac{\vec{V}_{p_i} \cdot \vec{V}_{p_j}}{\left\|\vec{V}_{p_i}\right\| \cdot \left\|\vec{V}_{p_j}\right\|} \tag{20}$$

A graph is constructed based on the similarity measurement. The graph node is the point label and the node (p_i, p_j) are jointed by an edge if $S(\vec{V}_{p_i}, \vec{V}_{p_j}) \geq Threshold$. Then, all the cliques are searched in the graph. The nodes in each clique belong to a cluster.

Suppose we have clustered the feature point set $\{p_i\}_{i=1,2,.....,m}$ into K clusters $C_1, C_2, ...C_k$, with size of $s_1, s_2, ...s_k$, respectively. The points are ranked by the associated cluster size in ascending order. Moreover, in the case of the same size of cluster, the rank is determined by the between class distances. Let OI_c_i be the rank index for point p_i.

2. The efficacy of the triangle formed by the two points and their principal orientations:

The points (p_i, p_j) and their principal orientations (e_{p_i}, e_{p_j}) can form a triangle with area size A_{ij}, as shown by solid line in Fig.6. Since the farer-separation of the three vertex points can determine the better accuracy of the transform parameters, we will provide a mechanism to measure the efficacy of the triangle.

The equilateral triangle gives a good point separation with equally distance of the three side lengths. To measure the degree of similarity of the formed triangle with the equilateral triangle, we can simulate a equilateral triangle with the three

equally side length $l_{\max}$ that is the longest side length of the original triangle, as shown by dash line in Fig.6. The area of the simulated equilateral triangle can be computed as $A_{rt} = \frac{\sqrt{3}}{4} l_{\max}{}^2$. Then, the degree of similarity S_{rt} of the original triangle with the simulated equilateral triangle is defined as the ratio of the area between the two triangles.

$$S_{rt} = \frac{A_{ij}}{A_{rt}} = \frac{A_{ij}}{\frac{\sqrt{3}}{4} l_{\max}{}^2} \in [0,1]$$

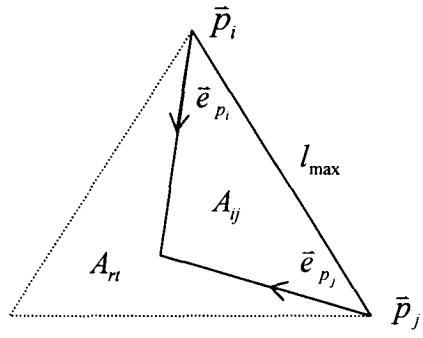


Fig. 6 The triangle formed by (p_i, p_j), (e_{p_i}, e_{p_j}) with area A_{ij} and the simulated equilateral triangle with area A_{rt}.

The equilateral triangle similarity S_{rt} embeds the information of the relative size of the area of the triangle. The analogous triangles with different scale have the same S_{rt}. As a result, we further need to consider the area of the triangle. However, if the two principal orientations e_{p_i} and e_{p_j} are nearly parallel, this will result in a huge area A_{ij} and a tiny S_{rt}. When A_{ij} multiplied by S_{rt}, the result is not preferred. Therefore, instead of using the area, the metric of the appearance of the triangle is defined as multiplying the side length by S_{rt} as the follows.

$$TA_{ij} = S_{rt} \left\| \vec{p}_i - \vec{p}_j \right\|$$

We rank the combinations of the two selected points by TA_{ij} in descending order.

Let OI_a_{ij} be the rank index for the pair of point p_i and p_j.

3. The stableness of the principal orientation:

Due to the triangle mentioned above is determined by the positions of the two feature points and their principal orientations, the accuracy of the principal orientations will acutely affect the determination of initial transformation T_0. The principal orientations are determined by finding the maximum absolute filter response among a bank of filter output. If the filter response at the principal orientation is farer discriminate from the others, the determined principal orientation is more stable. The discriminate capability can be measured by the normalized absolute difference of the filter response at the principal orientation with the two side neighbors.

$$DC_i = \frac{\left|R_i^{l_d} - R_i^{l_d+1}\right| + \left|R_i^{l_d} - R_i^{l_d-1}\right|}{2\left|R_i^{l_d}\right|}$$

where $R_i^{l_d}$ means the filter response of point P_i computed at principal orientation l_d.

We rank the point by DC_i in descending order. Let OI_o_i be the rank index for point P_i.

4. The energy value

The main factor resulted in the feature points do not appear in both of the two images is that the energy value is in the critical range of the threshold value defined in the feature detection process. Due to the points with higher energy will have the higher probability to appear in both of the two images, those points are preferred to have the priori order. We rank the points by the energy value in descending order with OI_e_i represent the rank index for point P_i.

We combine the above four order indexes into a single order index for the two selected points P_i and P_j.

$$OI_{ij} = w_1(OI\_c_i \times OI\_c_j) + w_2 OI\_a_{ij} + w_3(OI\_o_i \times OI\_o_j) + w_4(OI\_e_i \times OI\_e_j)$$

where w_1, w_2, w_3 and w_4 are the weights.

We rank OI_{ij} in a descending order and build a linking list, in which the entries store the pairs of points.

3.2.2 Registering stage(On-line)

In the registering process, we take a pair of points from the link list one by one to perform the initial estimate of the transformation T_0 and the *ICMP* to determine the *CPS* and the transformation *T*. The size of *CPS* and the *RMS* (*Root Mean Square*) of the distances of the points in *CPS* between their transformed versions will be used to measure the quality of the determined transformation. The searching is ended when it exceeds a suitable threshold. The algorithm of the registering process is the follows:

Algorithm registering_ process

Begin

While (size of $CPS_i <$ *threshold*$_{CPS}$*) or (* $RMS_i >$ *threshold*$_{RMS}$*)*

1. *Select the entry* e_i *from the link list. Assume the entry* e_i *corresponding to the point pair* p_k, p_l.
2. *To search the matching points in the sensed image by computing the correlation equation defined in Eq.10 using their feature vectors, assume the result shows that* q_m, q_n *are matching with* p_k, p_l.
3. *Check the concavity compatibility of the matching pairs* (p_k, q_m) *and* (p_l, q_n).

$$(\vec{e}_{p_k} \cdot \vec{e}_{p_l})(\vec{e}_{q_m} \cdot \vec{e}_{q_n}) \geq 0$$

where $\vec{e}_{p_k}, \vec{e}_{p_l}, \vec{e}_{q_m}, \vec{e}_{q_n}$ *are the principal orientations at* p_k, p_l, q_m, q_n.

If the compatibility check passes, go to step.4, otherwise, go to step 6.

4. *Apply* p_k, p_l, q_m, q_n *and* $\vec{e}_{p_k}, \vec{e}_{p_l}, \vec{e}_{q_m}, \vec{e}_{q_n}$ *to perform the initial estimation of the transformation* T_0 *and the ICMP algorithm described in Section.2 for determining the* CPS_i *and the transformation* T_i.

5. *Compute the* RMS_i *of the* CPS_i

$$RMS_i = \sqrt{\frac{1}{N_{cps_i}} \sum_{j=1}^{N_{cps_i}} \left\| p_j - T_i(q_j) \right\|^2}$$ *where* p_j, q_j *are a matching pair.*

6. *i=i+1*

End

$CPS_{final} = CPS_i$

$T_{final} = T_i$

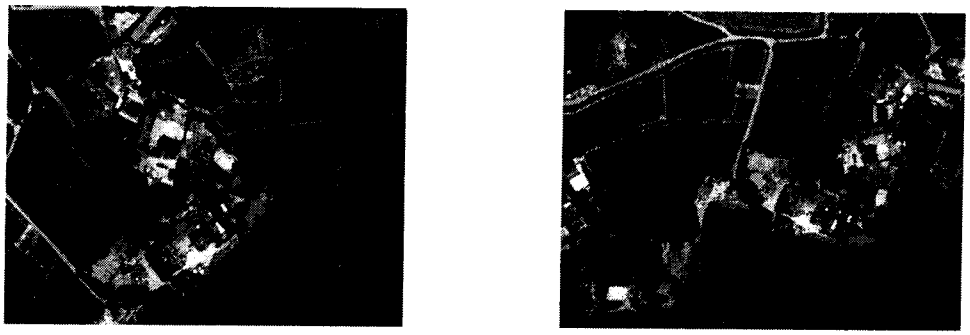
End

第四章 實驗分析

4.1 影像間轉換矩陣之求取

實驗一:

圖 7 所示為存在著旋轉及尺度差異之二張實驗空照影像，影像大小均為 500×400 pixels，其 overlap 區域約為 70%。



(a) (b)

圖 7 實驗影像 (a)Reference image (b)Sensory image

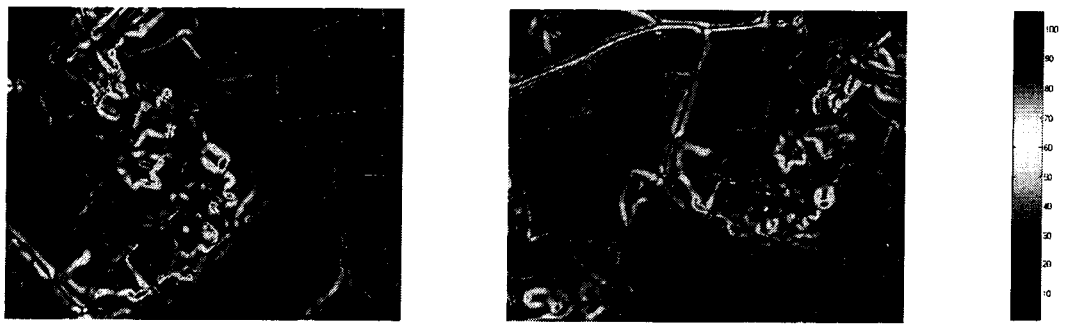
Gabor filter 的參數設計:

(1) N=4。

(2) 12 個方向(0~π)，每個方向差異為π/12。

(3) 共 8 個 scale(filter size 分別為 10×10, 12×12, 14×14, 17×17, 20×20, 24×24, 29×29, 34×34 及 40×40) 。

此二張影像與 Gabor filter set 作 convolution，其 maximum energy map 分別如圖 8(a)及(b) 。

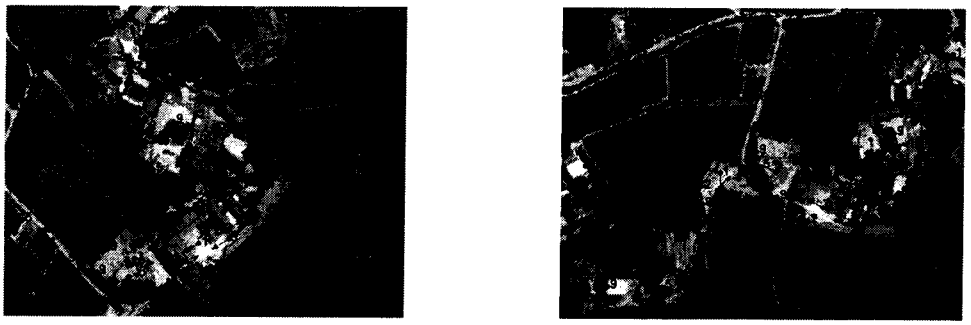


(a) (b)

圖 8 (a)Reference image (b)Sensory image 之 maximum energy map

在定義 block size 為 20×20 後，依第二張所述的特徵點粹取演算法，Reference image 粹取到 65 點，sensory image 則取到 57 點，特徵點位置及其 principal scale

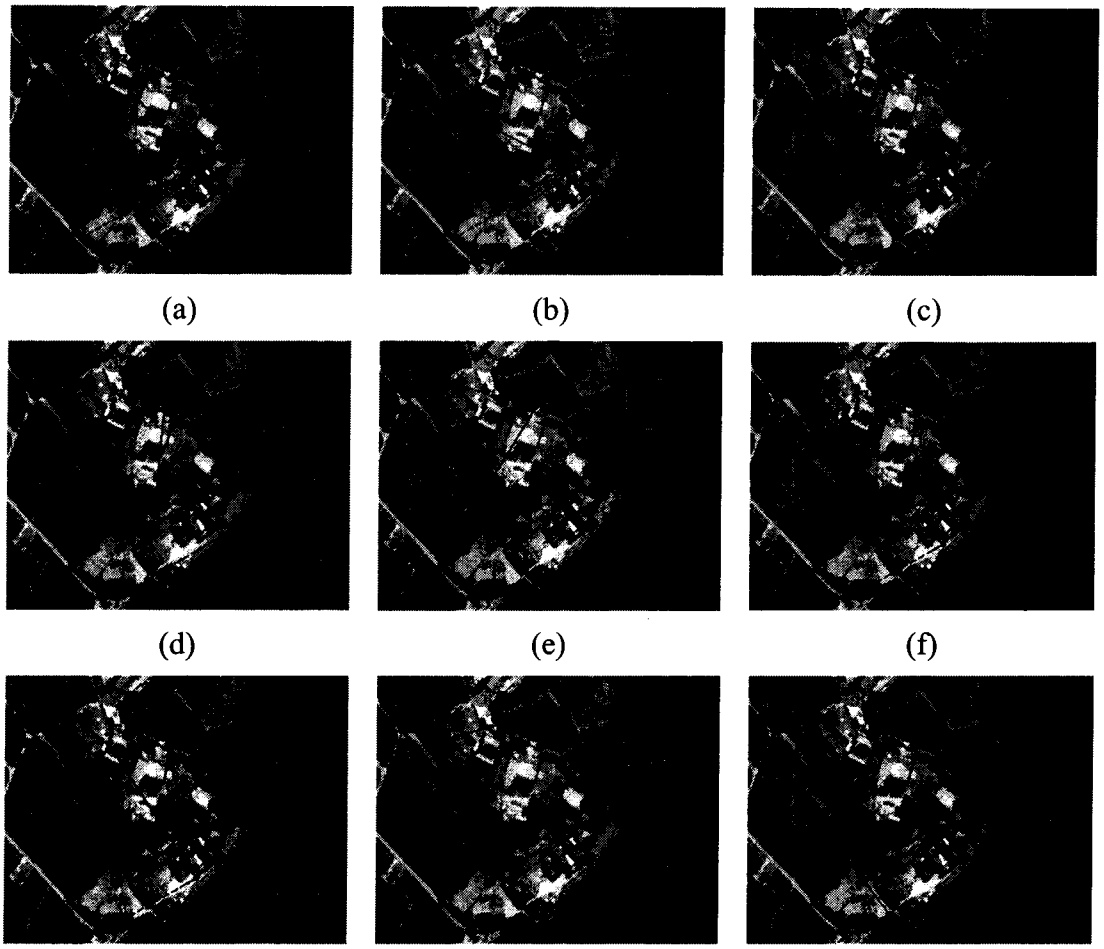
與 principal orientation index 分別如圖 9 (a)(b)所示。



(a) (b)

圖 9 (a)Reference image (b) sensory image 中所偵測到之特徵點位置及其 principal scale 與 principal orientation

參考影像之特徵點預先規劃求轉換矩陣之組合順序，圖 10 顯示前九個順序之點與其 principal orientation 所組成的三角形。



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

圖 10 Reference image 中特徵點預先規劃結果 (a)-(i) rank 1-9 之特徵點組合及其 principal orientation 所組成之三角形。

圖 11(a),(b)分別顯示 Reference image 及 Sensory image 中用來求取初始轉換矩陣之 2 點及其 principal orientation 所夾出之三角形，表 1 為求得之 $m_{3\times3}$ 轉換矩陣，圖 11(c)則顯示所找到對應到的特徵點之疊合結果，其中藍色圓圈表示 Reference image 之特徵點，紅色十字標則顯示 Sensory image 經乘上轉換矩陣後之對應特徵點位置，其中計找到 27 對特徵點對應。圖 12 則顯示 Reference image 與 Sensory image 之轉換影像疊合結果。

(a) (b) (c)

圖 11 (a)Reference image (b)Sensory image 中用來求取轉換矩陣之 2 點特徵點及其 principal orientation 所夾出之三角形 (c)所找到對應特徵點之疊合結果

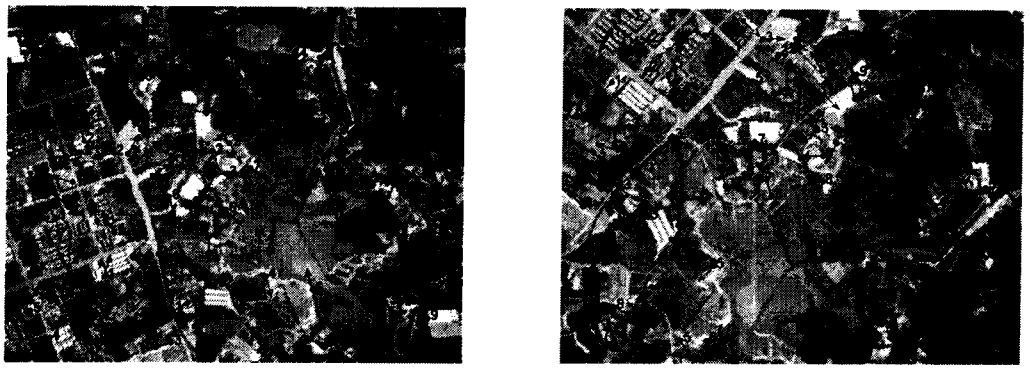
表 1 $m_{3\times3}$ 轉換矩陣

0.497306	0.869413	-175.136
-1.01355	0.477846	450.7193
-0.00035	-0.00015	1

圖 12 Reference image 與 Sensory image 之轉換影像疊合結果

實驗二:

圖 13 所示為存在著旋轉及尺度差異之二張實驗空照影像所找到的特徵點(Gabor filter 的參數設計與實驗一相同),影像大小均為 500×375 pixels,其 overlap 區域約為 90%,圖 14(a),(b)分別顯示 Reference image 及 Sensory image 中用來求取初始轉換矩陣之 2 點及其 principal orientation 所夾出之三角形,表 3 為求得之 $m_{3\times3}$ 轉換矩陣,圖 14(c)則顯示所找到對應到的特徵點之疊合結果,圖 15 則顯示 Reference image 與 Sensory image 之轉換影像疊合結果。



(a) (b)

圖 13 (a)Reference image (b) sensory image 中所偵測到之特徵點位置及其 principal scale 與 principal orientation

(a) (b) (c)

圖 14 (a)Reference image (b)Sensory image 中用來求取轉換矩陣之 2 點特徵點及其 principal orientation 所夾出之三角形 (c)所找到對應特徵點之疊合結果

表 3 實驗 2 之 $m_{3\times3}$ 轉換矩陣

0.397609	0.689224	15.09605
-0.69308	0.400835	289.3091
-0.000005	-0.000009	1

圖 15 實驗 2 Reference image 與 Sensory image 之轉換影像疊合結果

4.2 前處理後關聯性比對演算法則之精度改良評估

我們在 reference image, sensed image 及 transformed image 上分別以共同點為中心，取 121×121 pixles 之 template window，運用前計畫所提之六種關聯性比對演算法則來評估轉換處理前後影像之比對精度:

(1)絕對差值和(Sum of Absolute Differences: SAD)

$$\sum_{(u,v)\in W}\left|I_1(u,v)-I_2(x+u,y+v)\right|$$

(2)絕對差零均和(Zero mean Sum of Absolute Differences: ZSAD)

$$\sum_{(u,v)\in W}\left|(I_1(u,v)-\bar{I}_1)-(I_2(x+u,y+v)-\bar{I}_2)\right|$$

(3)差值平方和(Sum of Squared Differences: SSD)

$$\sum_{(u,v)\in W}\left(I_1(u,v)-I_2(x+u,y+v)\right)^2$$

(4)差值平方零均和(Zero mean Sum of Squared Differences: ZSSD)

$$\sum_{(u,v)\in W}\left((I_1(u,v)-\bar{I}_1)-(I_2(x+u,y+v)-\bar{I}_2)\right)^2$$

(5)正規交相聯(Normalized Cross Correlation: NCC)

$$\sum_{(u,v)\in W} I_1(u,v)\cdot I_2(x+u,y+v) \Big/ \sqrt{\sum_{(u,v)\in W} I_1^2(u,v)\cdot \sum_{(u,v)\in W} I_2^2(x+u,y+v)}$$

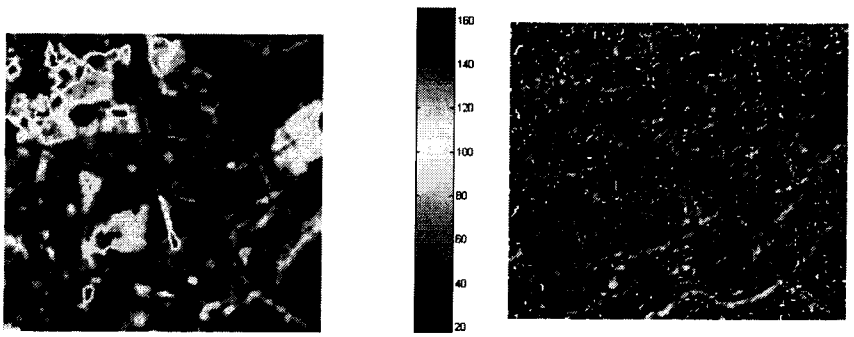
(6)零均正規交相聯(Zero mean Normalized Cross Correlation: ZNCC)

$$\sum_{(u,v)\in W}(I_1(u,v)-\bar{I}_1)\cdot(I_2(x+u,y+v)-\bar{I}_2) \Big/ \sqrt{\sum_{(u,v)\in W}(I_1(u,v)-\bar{I}_1)^2\cdot\sum_{(u,v)\in W}(I_2(x+u,y+v)-\bar{I}_2)^2}$$

圖 16(a)(b)(c)分別為實驗一中 reference image, sensed image 及轉換影像以某一共同點為中心所切割之 121×121 pixles template window，圖 17(a) 為 reference image 與 sensed image 之 template window 直接灰度值相減之差值圖，圖 17(b) 則為 reference image 與 transformed image 之 template window 直接灰度值相減之差值圖，這三個 template window 分別計算其六種關聯性比對演算法之係數如表 4.

(a) (b) (c)

圖 16 (a) Reference image (b) sensory image (c)轉換影像 中以特徵點為中心切割之 121×121 pixles 之 template window



(a) (b)

圖 17 (a) Reference image 與 sensed image 之 template window 直接灰度值差 (b) Reference image 與 transformed image 之 template window 直接灰度值差

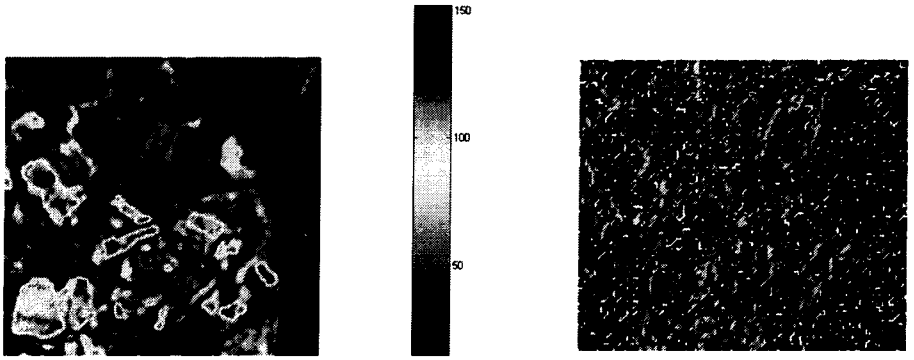
表 4 轉換處理前後之關聯性比對演算法精度比較

比對演算法	SAD	ZSAD	SSD	ZSSD	NCC	ZNCC
轉換影像	8.1975	96.6190	4.9352	45.0248	0.9989	0.9682
原影像	42.2062	2881.6	38.2646	2441.4	0.9207	0.1669

圖 18(a),(b)(c)分別為實驗二中 reference image, sensed image 及轉換影像以某一共同點為中心所切割之 121×121 pixles template window，圖 19(a) 為 reference image 與 sensed image 之 template window 直接灰度值相減之差值圖，圖 19(b) 則為 reference image 與 transformed image 之 template window 直接灰度值相減之差值圖，這三個 template window 分別計算其六種關聯性比對演算法之係數如表 5.

(a) (b) (c)

圖 18 (a) Reference image (b) sensory image (c)轉換影像 中以特徵點為中心切割之 121×121 pixles 之 template window



(a) (b)

圖 19(a) Reference image 與 sensed image 之 template window 直接灰度值差 (b) Reference image 與 transformed image 之 template window 直接灰度值差

表 5 轉換處理前後之關聯性比對演算法精度比較

比對演算法	SAD	ZSAD	SSD	ZSSD	NCC	ZNCC
轉換影像	4.1302	31.5342	3.8682	28.7037	0.9994	0.9844
原影像	36.8880	2387.1	37.0433	2355.6	0.9514	0.0963

第五章 結語

本計畫之目的在探討如何提高前計畫中直接用影像灰度值作相關度量測的比對精確度。首先，我們以 multi-scale 與 multi-orientation 的方向來深入分析 Gabor filter 的特性，我們提出 3 個重要之 Gabor filtering image 的 properties，並認為經由適當的 Gabor filter 參數設計，影像上會有 isolated feature points 存在，且任一點之 local pattern，具有其 natural scale 與 natural orientation，這些特性在特徵點之粹取與特徵表現上扮演關鍵的角色。在特徵點的粹取方面， 傳統上只用單一 scale 的方法，會使得使用不同 scale 的 output energy 會有不同的結果表現，造成特徵粹取不一致之困擾。我們的做法是利用 multi-scale 的特性及 principal scale 的觀念，將影像上每一點在其 principal scale 所產生的 output energy 組成 maximum energy map，這個 maximum energy map 具有對 scaling 及 rotation 不變量之特性，而特徵點的粹取是在 maximum energy map 找 local maximum，因此，更能處理影像間存在之 scaling 變化問題。我們以特徵點在其 principal scale 之 multi-orientation Gabor filter output responses 所組成之特徵向量來描述此特徵點之 local pattern structure，此特徵向量再以 principal orientation 為起始作 cyclic rotation，此作法將使得特徵向量具有對平移、旋轉、尺度不變量之特性， 而且對於 perspective deformation，如果主要變形不在 principal orientation 方向上的話，此特徵向量之變化並不大，因此，此特徵向量亦 robust to minor perspective deformation。特徵點間的相似程度是以其特徵向量作 cross-correlation 的相關度係數來量度， 由於 cross-correlation 有作 normalization 處理， 所以可以處理影像間存在之 uniform intensity 的變化。為建立特徵點間之 1-1 正確對應關係，我們發展出一個較快速的方法，這種方法是以 2 個對應點(如何決定用哪二個點可以經由 off-line 事先規劃)及其附帶之 principal orientation 所求得之 affine transformation 當作 initial solution，再利用 ICMP 以 iterative 方式作 refinement 並處理 perspective deformation 之問題。最後，我們使用二對存在著平移、旋轉、

尺度及幾何變形之影像，利用前計畫所提之六個關聯性比對演算法則，來分析處理前後之精度改良評估。

參考文獻

1. K. Arbter, W. E. Snyder, H. Burkhardt and G. Hirzinger, "Application of Affine-Invariant Fourier Descriptors to Recognition of 3D Objects", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.12, No.7, pp.640-647, July 1990.
2. Q. M. Tieng, W. W. Boles, "Wavelet-Based Affine Invariant Representation: A Tool for Recognizing Planar Objects in 3D Space", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.19, No.8, pp.846-857, Aug. 1997.
3. Y. Lamdan, J. T. Schwartz and H. J. Wolfson, "Affine Invariant Model-Based Object Recognition", *IEEE Trans. Robotics and Automation*, Vol.6, No.5, pp.578-589, Oct. 1990.
4. Z. Yang and F. S. Cohen, "Image Registration and Object Recognition Using Affine Invariants and Convex Hulls", *IEEE Trans. Image Processing*, Vol.8, No.7, pp.934-946, July 1999.
5. R. K. K. Yip, P. K. S. Tam and D. N. K. Leung, "Application of Elliptic Fourier Descriptors to Symmetry Detection Under Parallel Projection", *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol.16, No.3, pp.277-286, March 1994.
6. K. C. Hung, "The Generalized Uniqueness Wavelet Descriptor for Planar Closed Curves", *IEEE Trans. Image Processing*, Vol.9, No.5, pp834-845, May 2000.
7. J.G. Daugman, Two-dimensional spectral analysis of cortical receptive field profile. *Vision Research*, vol. 20(1980), pp. 847-856.
8. J.G. Daugman, Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters. J. *Optical Soc. Amer.*, vol. 2, no.7(1988), pp. 1169-1179.
9. J.G. Daugman, Computing discrete 2-D Gabor transforms by neural networks for image analysis and compression. *IEEE Trans. Acous., Speech, Signal Processing*, vol. 36 (1988), pp. 1169-1179.
10. T. S. Lee, Image representation using Gabor wavelets. *IEEE Trans. PAMI*, vol 18, no. 10(1996), pp. 959-97.
11. X. Wu and B. Bhanu, Gabor wavelet representation for 3D object recognition. *IEEE Trans. Image Processing*, vol.6, no. 1(1997), pp. 47-64.
12. Z. Wang and M. Jenkin, Using complex Gabor filters to detect and localize edges and bars.

In: C. Archibald and E. Petriu, (eds.): *Advanced in Machine Vision: Strategies and Applications*, vol. 32, River Edge, NJ: World Scientific ,1992, pp. 151-170.

13. Z. Zhang, R. Deriche, O. Faugeras, Q. T. Luong, "A robust technique for matching two uncalibrated images through the recovery of the unknown epiploar geometry ", *Artificial Intelligence*, Vo.78, pp.87-119, 1995.
14. J. Zhou and J. Shi, "A robust algorithm for feature point matching", *Computers & Graphics*, Vol.26, pp.429-436, 2001.
15. Fonseca, L.M.G.; Costa, M.H.M., "Automatic registration of satellite images", *Computer Graphics and Image Processing*, 1997. Proceedings., X Brazilian Symposium on ,pp. 219 –226,1997.
16. F. Ola and J. A. Marchant, "Matching feature points in image sequences through a region-based method", *Computer Vision and Image Understanding*, Vol.66, No.3, pp.271-285,1997.
17. M. C. Ibison, L. Zapalowski, "On the use of relaxation labeling in the correspondence problem", *Pattern Recognition Letter*, Vol.4 pp.103-109,1986.
18. J. Ton and A. K. Jain, "Registering Landsat images by point matching", *IEEE Trans. Geoscience and Remote Sensing,* Vol.27, No.5, pp.642-651,1989.
19. Z. Yang and F. S. Cohen, "Image registration and object recognition using affine invariants and convex hulls", *IEEE Trans. Image Processing*, Vol.8, No.7, pp.934-946, July 1999.
20. A. Branca, E. Stella and A. Distante, "An high order relaxation labeling neural network for feature matching", *Neural Networks Proceedings, IEEE World Congress on Computational Intelligence*, Vol.2, pp.1590-1595, 1998.
21. T. Suk and J. Flusser, "Point-based projective invariants", *Pattern Recognition*, Vol.33 pp.251-261, 2000.
22. H. Lamdan, J. T. Schwartz and H. J. Wolfson, "Affine Inavriant Model-Based

Object Recognition", *IEEE Trans. Robotics and Automation*, Vol.6, No.5, pp. 578-589, Oct. 1990.

23. F. H. Cheng, "Point pattern matching algorithm invariant to geometrical transformation and distortion". *Pattern Recognition Letter*, Vol.17 pp.1429-1435, 1996.