A Condition Number for Point Matching with Application to Registration and Post-Registration Error Estimation

C. S. Kenney[†], B. S. Manjunath[†], M. Zuliani[†], G. Hewer[‡], A. Van Nevel[‡]

[†] Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA
{kenney, manj, zuliani}@ece.ucsb.edu

[‡] Research Department
Naval Air Warfare Center Weapons Division, Code 4T4100D
China Lake, CA 93555-6100
{hewerga, vannevelaj}@navair.navy.mil

Abstract

Selecting salient points from two or more images for computing correspondence is a well studied problem in image analysis. This paper describes a new and effective technique for selecting these tiepoints using condition numbers, with application to image registration and mosaicking. Condition numbers are derived for point-matching methods based on minimizing windowed objective functions for 1) translation, 2) rotation-scaling-translation (RST) and 3) affine transformations. Our principal result is that the condition numbers satisfy $K_{Trans} \leq K_{RST} \leq K_{Affine}$. That is, if a point is ill-conditioned with respect to point-matching via translation then it is also unsuited for matching with respect to RST and affine transforms. This is fortunate since K_{Trans} is easily computed whereas K_{RST} and K_{Affine} are not. The second half of the paper applies the condition estimation results to the problem of identifying tiepoints in pairs of images for the purpose of registration. Once these points have been matched (after culling outliers using a RANSAC-like procedure) the registration parameters are computed. The post-registration error between the reference image and the stabilized image is then estimated by evaluating the translation between these images at points exhibiting good conditioning with respect to translation. The proposed method of tiepoint selection and matching using condition number provides a reliable basis for registration. The method has been tested on a large number of diverse collection of images–multi-date Landsat images, aerial images, aerial videos, and infra-red images. A web site where the users can try our registration software is available and is being actively used by researchers around the world.

Index Words: Registration, Conditioning, Feature representation, Motion

This research was supported by the Office of Naval Research under ONR Grant Number N00014-02-1-0318.

I. Introduction

The problem of selecting image points for reliably determining optical flow, image registration parameters and 3D reconstruction has been extensively studied over the last 30 years, and many good schemes for selecting feature points have been advanced. In this paper we show that condition theory can be applied to this problem with the result that we recover and extend time-tested feature selection procedures. The formulation we present has the advantage of being derived from first principles rather than heuristics.

As early as 1987, with the work of Kearney et al. [34] it was realized that the normal matrix associated with locally constant optical flow is critical in determining the accuracy of the computed flow. This matrix has the form

$$A^T A \equiv \begin{bmatrix} \sum g_x g_x & \sum g_x g_y \\ \sum g_x g_y & \sum g_y g_y \end{bmatrix} \tag{1}$$

where $g = g(x, y)$ refers to the image intensity, subscripts indicate differentiation, and the summation is over a window about the point of interest. Kearney et al. report that ill-conditioning in the matrix $A^T A$ and large residual error in solving the equations for optical flow can result in inaccurate flow estimates. This was supported by the work of Barron et al. [3] who looked at the performance of different optical flow methods; see also [4].

More recently, Shi and Tomasi [45] presented a technique for measuring the quality of local windows for the purpose of determining image transform parameters (translational or affine). For local translation they argued that in order to overcome errors introduced by noise and ill-conditioning, the smallest eigenvalue of the normal matrix $A^T A$ must be above a certain threshold: $\lambda \leq \min(\lambda_1, \lambda_2)$ where λ is the prescribed threshold and λ_1, λ_2 are the eigenvalues of $A^T A$. When this condition is met the point of interest has good features for tracking [46]. The paper [46] builds on the work of Shi and Tomasi [45] by examining the statistics of the residual difference between a window and a computed backtransform of the corresponding window in a second image with the goal of deriving conditions for rejecting a putative match. The papers [8] and [33] give further consideration to the importance of the $A^T A$ in estimating vision parameters.

Schmidt et al. [44] present results for the problem of evaluating interest point detectors from the standpoint of repeatability (i.e., whether the point is repeatedly detected in a series or sequence of images) and information content (which measures the distinctiveness of the interest point as measured by the likelihood of the local greyvalue descriptor). This paper provides a good survey of interest point detectors for contour-based methods such as [56] and [38], corner detection [20], intensity-based methods and parametric model methods. Evaluation of computed video georegistration for aerial video is considered by [52].

The work we present below uses condition theory to formulate the sensitivity of matching feature points with respect to image transforms, such as translation, rotation-scale-translation (RST), and affine transforms. As such it bears a close relationship to the work of earlier investigators especially that of Shi and Tomasi [45] (see Section 2 for more details). Note that in this paper we focus on point feature only. This can be further extended by considering geometrical features such as lines (see [5], [32], [52]) but such an extension would require defining an objective function to determine these features as well as a corresponding conditioning based analysis. Aside from this agreement with time-tested results, the condition theory approach shows how to define point sensitivity with respect to other image transformation models such as the RST and affine transforms. This generality allows us to compare condition numbers for different transforms (Theorem 2).

A. Problem Formulation

We start by casting the matching problem as a minimization of an objective function that measures the match between windows in both images.

Given two images g and $\hat{g}$ and a point (x, y) we define the point-matching objective function for a transformation T as

$$f(T) = \frac{1}{2}\,\Sigma\left(g(T(x', y')) - \hat{g}(x', y')\right)^2 \tag{2}$$

where the summation is over (x', y') in a window about (x, y).

We consider three types of transformations with associated parameter vectors p and we seek the best parameter vector for minimizing the objective function.

1) Translation: we look for the best shift $p = (a, b)$ minimizing the objective function (2) with T given by

$$T(x', y') = \begin{bmatrix} x' + a \\ y' + b \end{bmatrix} \tag{3}$$

2) Rotation-Scale-Translation (RST): find the best rotation-scale-translation values $p = (\theta, r, a, b)$ minimizing the objective function (2) with T given by

$$T(x', y') = r \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x' - x \\ y' - y \end{bmatrix} + \begin{bmatrix} a \\ b \end{bmatrix} \tag{4}$$

3) Affine: find the best affine parameter values $p = (m_{11}, m_{12}, m_{21}, m_{22}, a, b)$ minimizing the objective function (2) with T given by

$$T(x', y') = \begin{bmatrix} m_{11} & m_{12} \\ m_{21} & m_{22} \end{bmatrix} \begin{bmatrix} x' - x \\ y' - y \end{bmatrix} + \begin{bmatrix} a \\ b \end{bmatrix} \tag{5}$$

To measure the sensitivity of the minimizing solutions we use the following definition.

Definition. Let $p = p(x, y, g, \hat{g}, T)$ denote the minimizer to the objective function (2). The condition number K_T measures the sensitivity of p to perturbations $(\Delta g, \Delta\hat{g})$ and is defined by

$$K_T \equiv \lim_{\delta \to 0} \max_{\|(\Delta g, \Delta\hat{g})\| \leq \delta} \frac{\|\Delta p\|}{\|(\Delta g, \Delta\hat{g})\|} \tag{6}$$

where $\|(\Delta g, \Delta\hat{g})\| = \left(\|\Delta g\|^2 + \|\Delta\hat{g}\|^2\right)^{1/2}$ and Δp denotes the perturbation in the parameter vector p corresponding to the perturbation $(\Delta g, \Delta\hat{g})$.

In the next section we describe the background and standard theory of condition measurement. We then derive computable expressions for the condition numbers for point-matching with respect to the three types of transforms (Theorem 1). Our principal result is the inequality (Theorem 2)

$$K_{Trans} \leq K_{RST} \leq K_{Affine} \tag{7}$$

This makes a lot of sense: minimizing the three types of objective functions corresponds to trying to extract more information from a fixed data set and should give results that are increasingly uncertain.

Computationally K_{Trans} is easy to evaluate, taking about 24 arithmetic operations per pixel (see Appendix C), whereas K_{RST} and K_{Affine} are much more expensive. Because of this and in light of (7), we concentrate in the remainder of the paper on applications of K_{Trans} in the areas of optical flow, registration and post-registration error estimation. As explained in the applications section, we use the translation condition number to cull the points used in the matching process; however once this is done the transformation parameters (in our case the RST values) are determined from the geometry of the feature points so that the use of the translation condition number does not limit the accuracy of our RST estimates.

In Section 3 we compare the minimization of the translation objective function with the problem of determining the optical flow. In Section 4 we describe using the translation condition number to select tiepoints for image registration. We also look at estimating the offset error between the first image and a stabilized image obtained by back-transforming the second image to the reference frame of the first image. The local offset error is determined by computing the best local translation at points in the image that are well conditioned with respect to matching by translation.

Before proceeding it is helpful to make a few remarks about the goal of our work. We do not advocate computing the best local translation for arbitrary image pairs because the limitation to a simple translation model is too restrictive. On the other hand the translation condition number is a useful tool in image analysis for several reasons. First the expression for the condition number turns out to be invariant with respect to rotation as well as independent of the actual translation. This is a property that one would expect of any good measure of the matching quality of an image point. For example a well defined corner remains a good point for matching when rotated or shifted.

Second there are image pairs for which the local translation model is appropriate. For example, after computing registration parameters one might want to transform the second image into the frame of reference of the first in order to detect target motion. Assuming that the computed registration parameters are reasonably accurate we might want to calculate the best local translation between images as an estimate of the offset error in the back-transform.

II. Condition Theory

A. Historical Background

The current viewpoint on condition estimation can trace its roots to the era of the 1950's with the development of the computer and the attendant ability to solve large linear systems of equations and eigenproblems. The question facing investigators at that time was whether such problems be solved reliably. The work of Wilkinson [54], [55] and Rice [43] in the early 1960's established a general theory of condition for computed functions. In the 1970's this led to power method condition estimates for matrix inversion by Cline et al. [10] as well as condition estimates for the exponential matrix function by Ward [51], Moler and Van Loan [42], invariant subspaces associated with eigenvalues by Stewart [47] and other problems. The 1980's saw the extension of this work (especially the ideas of Cline et al. [10]) to the matrix square root function by Björck and Hammarling [7], the Lyapunov and Riccati equations by Hewer and Kenney [22], [35] and the distance to the nearest unstable matrix by Van Loan [53] and by Hinrichsen [27]. At the same time, Kenney and Laub provided computational procedures for estimating the sensitivity of general matrix functions [36], [37] and Demmel developed a global rather than local theory of sensitivity based on the distance to the nearest ill-posed problem [11], [12], [13].

This latter work represents a step toward more realistic condition results since it goes beyond the assumptions of sensitivity based on smoothness and differentiability. This is especially appropriate for image processing since many perturbation effects, such as quantized gray level intensity changes, are discrete rather than continuous. Unfortunately the more realistic condition results for discrete and global changes are usually very difficult to evaluate. Because of this for the purposes of our work we will restrict our attention to continuous condition estimates as described below.

B. General Condition Measures

The solution of a system of equations can be viewed as a mapping from the input data $D \in \mathbb{R}^n$ to the solution or output $X = X(D) \in \mathbb{R}^m$. If a small change in D produces a large change in $X(D)$ then X is ill-conditioned at D. Following Rice [43], we define the δ-condition number of X at D by

$$K_\delta = K_\delta(X, D) \equiv \max_{\|\Delta D\| \leq \delta} \frac{\|X(D + \Delta D) - X(D)\|}{\|\Delta D\|} \tag{8}$$

where $\|\cdot\|$ denotes the vector 2-norm: $\|D\|^2 = \Sigma|D_i|^2$. For any perturbation ΔD with $\|\Delta D\| \leq \delta$, the perturbation in the solution satisfies

$$\|X(D + \Delta D) - X(D)\| \leq \delta K_\delta \tag{9}$$

The δ condition number inherits any nonlinearity in the function X and consequently is usually impossible to compute. For this reason the standard procedure is to take the limit as $\delta \to 0$. If X is differentiable at D we can define the (local or differential) condition number

$$K = K(X, D) \equiv \lim_{\delta \to 0} K_\delta(X, D) \tag{10}$$

Using a first order Taylor expansion, we have

$$X(D + \Delta D) = X(D) + X_D\ \Delta D + O(\|\Delta\|^2) \tag{11}$$

where X_D is the $m \times n$ gradient matrix with entries

$$(X_D)_{ij} = \frac{\partial X_i}{\partial D_j} \tag{12}$$

This expansion shows that the local condition number is just the norm of the matrix X_D

$$K(X, D) = \|X_D\| \tag{13}$$

and

$$\|X(D + \Delta D) - X(D)\| \leq K\|\Delta D\| + O(\|\Delta D\|^2) \tag{14}$$

Large values for $K(X, D)$ indicate that X is ill conditioned in D.

C. Objective Function Conditioning

The general theory of condition as applied to the sensitivity of the minimizer of the objective function (2) gives the definition (6). This condition number incorporates all possible perturbations to an arbitrary pair of images $(g, \hat{g})$. Unfortunately this generality makes the analysis difficult. To overcome this we consider the condition number for a more reasonable class of problems in which g is related to $\hat{g}$ via the transform T and we allow $\hat{g}$ to be perturbed by noise.

Theorem 1. *Define the 'transform plus noise' problem as* $\hat{g}(x', y') = g\left(T(x', y')\right) + \eta(x', y')$ *with perturbations* $\|\hat{\eta}\| \le \delta$ *and* (x', y') *varies over the window centered at* (x, y)*. In the limit as* $\delta \to 0$ *we find that*

$$K_T^2 = \| \left(A^T A\right)^{-1} \| \tag{15}$$

where

$$A = \begin{bmatrix} v^1 \\ \vdots \\ v^n \end{bmatrix} \tag{16}$$

with row vectors v^i *depending on the type of transformation* T*. The row vectors for translation, RST, and affine transformations are given by*

1) *Translation:* $v^i = \begin{pmatrix} \hat{g}_x^i & \hat{g}_y^i \end{pmatrix}$
2) *RST:* $v^i = \begin{pmatrix} \hat{g}_x^i & \hat{g}_y^i & \hat{g}_x^i(x^i - x) + \hat{g}_y^i(y^i - y) & \hat{g}_x^i(y^i - y) - \hat{g}_y^i(x^i - x) \end{pmatrix}$
3) *Affine:* $v^i = \begin{pmatrix} \hat{g}_x^i & \hat{g}_y^i & \hat{g}_x^i(x^i - x) & \hat{g}_x^i(y^i - y) & \hat{g}_y^i(x^i - x) & \hat{g}_y^i(y^i - y) \end{pmatrix}$

where (x^i, y^i) *is the* i *the point in the window centered at* (x, y)*,* $\hat{g}_x^i = \hat{g}_x(x^i, y^i)$*,* $\hat{g}_y^i = \hat{g}_y(x^i, y^i)$ *and subscripts denote differentiation.*

Proof. See the Appendix A. □

Remark. The condition numbers are given in terms of $\hat{g}_x$ and $\hat{g}_y$ and do not require knowledge of the minimizing parameters for the transformation. For the translation condition number, the matrix $A^T A$ is 2×2 and given by

$$A^T A = \begin{bmatrix} \Sigma\hat{g}_x^2 & \Sigma\hat{g}_x\hat{g}_y \\ \Sigma\hat{g}_x\hat{g}_y & \Sigma\hat{g}_y^2 \end{bmatrix} \tag{17}$$

where the summation is over (x', y') in a window centered at (x, y).

The connection between conditioning and the matrix $A^T A$ is not surprising considering that the eigenvalues of this matrix have commonly been used in the analysis of local image structure, in particular, for distinguishing flat, line and corner structures [20], [49].

Since we are dealing with a 2×2 matrix we can write out the inverse explicitly. As a technical point it is numerically easier to deal with the modified condition number

$$\tilde{K}^2_{Trans} \equiv \| \left(A^T A + \epsilon I \right)^{-1} \| \tag{18}$$

where ϵ is some small number (see [18] for the relationship with the pseudo-inverse). This formulation avoids problems associated with inverting singular or nearly singular matrices.

In our experiments we used $\epsilon = 10^{-8}$. Here we are using the 2-norm. If we switch to the Schatten 1-norm $\|M\|_S = \Sigma \sigma_i$ where $\sigma_1, \ldots, \sigma_n$ are the singular values of M (see [30] p. 199) then we have the closed form expression:

$$\begin{aligned} K^2_{Trans,Schatten} &\equiv \| \left(A^T A + \epsilon I \right)^{-1} \|_S \qquad (19) \\ &= \frac{2\epsilon + \Sigma \hat{g}_x^2 + \Sigma \hat{g}_y^2}{\left(\Sigma \hat{g}_x^2 + \epsilon \right) \left(\Sigma \hat{g}_y^2 + \epsilon \right) - \left(\Sigma \hat{g}_x \hat{g}_y \right)^2} \qquad (20) \end{aligned}$$

Here we have used the fact that if M is a symmetric positive definite matrix and its eigenvalues are λ_1 and λ_2, then

$$\|M\|_S = \Sigma \sigma_i = \Sigma \lambda_i = \text{Trace}(M) \tag{21}$$

Applying this with $M = (A^T A + \epsilon I)^{-1}$ gives the expression for $K_{Trans,Schatten}$. As a side note we have switched to the Schatten norm for ease of computation only; the Schatten 1-norm is essentially equivalent to the regular 2-norm because the 2-norm of a matrix M is equal to the largest singular value: $\|M\|_2 = \sigma_2$. Thus for a 2×2 matrix M the following chain of inequalities holds: $\|M\|_2 \leq \|M\|_S \leq 2\|M\|_1$. We also note that

$$\| \left(A^T A + \epsilon I \right)^{-1} \|_S = \frac{\text{Trace} \left(A^T A + \epsilon I \right)}{\det \left(A^T A + \epsilon I \right)} = \frac{\mu_1 + \mu_2}{\mu_1 \mu_2} \geq \frac{1}{\mu_2}$$

where μ_1 and μ_2 are the eigenvalues of the matrix $A^T A + \epsilon I$, and $0 \leq \mu_2 \leq \mu_1$. This provides nice connection with the work of Shi-Tomasi-Kanade [45] in which a point is considered good for feature tracking if μ_2 is large or equivalently if $1/\mu_2$ is small. The connection arises from the inequality

$$\frac{1}{\mu_2} \leq \frac{\mu_1 + \mu_2}{\mu_1 \mu_2} \leq \frac{2}{\mu_2}$$

for $0 \leq \mu_2 \leq \mu_1$. That is

$$\frac{1}{\mu_2} \leq K^2_{Trans,Schatten} = \| \left(A^T A + \epsilon I \right)^{-1} \|_S = \frac{\mu_1 + \mu_2}{\mu_1 \mu_2} \leq \frac{2}{\mu_2}$$

Moreover, since we are working with the matrix 2-norm, we can write $K^2_{Trans} = \|(A^TA)^{-1}\| = \frac{1}{\min(\lambda_1,\lambda_2)}$ where λ_1, λ_2 are the eigenvalues of A^TA. Thus the Shi-Tomasi requirement that $\lambda < \min(\lambda_1, \lambda_2)$ for some threshold value λ is equivalent to requiring that $K_{Trans} \leq \frac{1}{\sqrt{\lambda}}$; that is the Shi-Tomasi feature condition is equivalent to specifying a maximum condition value for translation.

We now state our main result.

Theorem 2. *Under the assumptions of Theorem 1 we have*

$$K_{Trans} \leq K_{RST} \leq K_{Affine} \tag{22}$$

Proof. See the Appendix B. □

Figure 1 shows a noisy IR image containing an urban street scene that we will use to illustrate the condition number for the three types of matching: translation, RST and affine.

Figure 2, gives the condition surface for the three types of transforms. In this image dark points have good conditioning. The striking feature of Figure 2 is the overall similarity of the condition numbers for translation, RST and affine point matching for the urban street image. In part this similarity is a consequence of (22): well conditioned points are only a small percentage of the overall image pixels and (7) implies that the well conditioned points for translation, RST and affine transformations form a nested series of subsets. This is excellent because the computation of the translation condition number requires the inversion of a 2×2 matrix and can be calculated very rapidly. In contrast both the RST and affine condition numbers are much more expensive to compute because they involve inverting respectively a 4x4 or a 6x6 matrix at each point in the image, as per Theorem 1.

Example 1. A point may be well-conditioned with respect to translation and ill-conditioned with respect to an RST transformation. As an illustration consider a dark point at (x, y) on a white background. This point is well-conditioned for translational matching but the rotational symmetry means that we can not extract rotational information: it is ill-conditioned with respect to RST matching. In Section 4, we use points that are well-conditioned with respect to translation for computing the best RST transformation between two images. One might reasonably ask why we don't use points that are well-conditioned with respect to RST matching rather than translational matching. The reason is provided by this example: a point may be ill-conditioned for local RST matching while being part of a group of points that is well-conditioned for global RST matching. Consider an image of three dark points on a white background, and a second image that is a rotated-scaled-translated version of the first image. Each dark point is well-conditioned for translation matching and ill-conditioned for RST matching. On the other hand the problem of extracting the global RST parameters from the three points as a group in each image is very well-conditioned, assuming of course a reasonable spread in the points. Although we do not pursue it, similar remarks apply to multiple line-like features, each of which can provide constraints on global transformations.

III. Discussion of Related Work

A. *Minimizing the Objective Function Via Newton's Method*

An efficient method of solving for the best translation using Newton's method is discussed by Vemuri et al. [50], which also has a nice survey of work in the areas of registration and optical flow. In order to discuss the work in [50] and to see how their method could benefit by using conditioning, it is helpful to describe Newton's method for minimizing an objective function.

To minimize a general function $F(X)$ where $X = (X_1, \cdots, X_n)$, Newton's method is applied to the problem $F_X = 0$. This gives [1], p. 108-111, the iteration

$$X^{k+1} = X^k - F_{XX}^{-1} F_X \tag{23}$$

where subscripts denote differentiation: $(F_X)_i = \partial F / \partial X_i$ and $(F_{XX})_{ij} = \partial^2 F / \partial X_i \partial X_j$.

For Newton's method for minimizing the translation objective function we set $X = (a, b)$ and $F(X) = f(a, b)$ and find that from (1) and (2),

$$F_X = \begin{bmatrix} f_a \\ \\ f_b \end{bmatrix} = \begin{bmatrix} \Sigma \left(g(x'+a, y'+b) - \hat{g}(x', y')\right) \; g_x(x'+a, y'+b) \\ \\ \Sigma \left(g(x'+a, y'+b) - \hat{g}(x', y')\right) \; g_y(x'+a, y'+b) \end{bmatrix} \tag{24}$$

$$F_{XX} = \begin{bmatrix} f_{aa} & f_{ab} \\ f_{ab} & f_{bb} \end{bmatrix} \tag{25}$$

where

$$\begin{aligned} f_{aa} &= \Sigma g_x^2(x'+a, y'+b) + \left(g(x'+a, y'+b) - \hat{g}(x', y')\right) \; g_{xx}(x'+a, y'+b) \\ f_{ab} &= \Sigma g_x(x'+a, y'+b) \; g_y(x'+a, y'+b) + \left(g(x'+a, y'+b) - \hat{g}(x', y')\right) \; g_{xy}(x'+a, y'+b) \\ f_{bb} &= \Sigma g_y^2(x'+a, y'+b) + \left(g(x'+a, y'+b) - \hat{g}(x', y')\right) \; g_{yy}(x'+a, y'+b) \end{aligned}$$

and the summations are taken over the local window as in (1).

Noting that the expression for F_{XX} requires computing the gray scale Hessian at each point, Vemuri et al. [50] found that they could achieve greater efficiency in solving for the best translation by approximating F_{XX} locally by using the form of F_{XX} at the minimizing point, in which case the second derivatives of g drop out. The pleasant result from [50] is that this can be done without knowing the minimizing point explicitly.

This avoids repeatedly computing the Hessian as the translation of neighboring points is found. Further efficiency is obtained by taking a hierarchical approach in which the best translation is found on a sparse set of points and then extended to the entire set of image pixels using spline interpolation. However, this approach has the drawback that some points on the sparse grid may not be suitable for point matching. Hence it seems reasonable that the method of [50] could benefit by taking into consideration the conditioning of the grid points, possibly with the conditioning determining a weighting used in the spline interpolation process.

B. Optical Flow

The area of optical flow is closely related to the problem of determining the best local translation for matching points between images. Unfortunately, many existing strategies for evaluating the reliability of optical flow computations can lead to unstable estimates.

Let $g = g(x, y, t)$ be the intensity of a time varying image. If a point $(x, y) = (x(t), y(t))$ in the image maintains constant brightness $g(x(t), y(t), t) = c$ with respect to time then its time derivative is zero [29]:

$$0 = g_x x_t + g_y y_t + g_t \tag{26}$$

where subscripts denote differentiation. We would like to know (x_t, y_t) in order to track the approximate motion of the point

$$(x(t), y(t)) \approx (x(0), y(0)) + t\ (x_t, y_t) \tag{27}$$

If we identify g_t with the difference between the first and second images

$$g_t = \hat{g} - g \tag{28}$$

and estimate g_x and g_y via finite differences then we have one equation for the two unknowns x_t and y_t at each point in the image. The vector (x_t, y_t) is referred to as the optical flow at (x, y). Several approaches have been advocated for overcoming the underdetermined nature of optical flow equations.

Horn and Shunck [28] impose a smoothness constraint on the optical flow by casting the optical flow as a minimization problem over the entire image. This results in a large system of elliptic linear equations which is usually solved iteratively starting with the ‘normal flow’

$$(x_t, y_t)_{nor} = (-g_t g_x, -g_t g_y)/(g_x^2 + g_y^2)^{1/2} \tag{29}$$

as an initial guess. Unfortunately this approach is computationally costly and has problems at points in the image where the optical flow is discontinuous such as may be induced by the motion of occluding objects [31].

To avoid these problems various alternatives have been suggested. For example, Hildreth [26] proposed an optical flow computation of edges in an image sequence, which is a 1-D version of Horn and Schunck’s algorithm. Sundareswaran and Mallat [48] combine Hildreth’s approach with multiscale wavelet information to regularize the optical flow computation. See also [23].

One of the more successful alternative methods assumes that local to the point (x, y) the optical flow is constant [6], [39], [41]. For example, if the optical flow is constant in a window about (x, y) then we have a system of equations for the two unknowns (x_t, y_t):

$$A \begin{bmatrix} x_t \\ y_t \end{bmatrix} = v \tag{30}$$

where

$$A = \begin{bmatrix} g_x^1 & g_y^1 \\ \vdots & \\ g_x^n & g_y^n \end{bmatrix} \qquad v = \begin{bmatrix} -g_t^1 \\ \vdots \\ -g_t^n \end{bmatrix} \tag{31}$$

where the superscript denotes the function value ranging over the window centered at (x, y). Under the assumption that A is full rank the least squares solution is given by

$$\begin{bmatrix} x_t \\ y_t \end{bmatrix} = \left(A^T A\right)^{-1} A^T v \tag{32}$$

The assumption that the optical flow is constant over a window is used to overcome the underdetermined nature of optical flow equations. Unfortunately this approach may fail. For example if the image consists of a linear edge moving from left to right then the least squares system of equations is rank deficient and does not have a unique solution. This problem is well known and leads to a loss of accuracy in the optical flow vector estimate. A variety of related effects can also cause loss of accuracy in the computed optical flow as described by Kearney et al. [34]. This has prompted some investigators such as Irani [31] et al. to assign a reliability measure to the flow estimates based on K_L the condition number with respect to inversion of the least squares system. For a square matrix L, the condition number with respect to inversion is $K_L = \|L\| \; \|L^{-1}\|$; for the optical flow problem $L = A^T A$ as in (1). Values of K_L near 1 indicate that L is well-conditioned with respect to inversion. However, values of K_L near 1 do not ensure accurate optical flow computation, as seen by a simple example. Consider a flat background point in the image. The least squares matrix L in this case is doubly rank deficient: it has rank zero. Now add a slight amount of noise to the image. Since K_L is the ratio of the largest singular value of L to the smallest singular value of L, we see that K_L can be close to 1 under the addition of even arbitrarily small noise, as shown by the following experiment.

Example 2. Starting with an image of constant intensity, we added Gaussian random noise of mean zero and standard deviation $\sigma = 10^{-8}$ at each point. Using a 3x3 window we then formed the least squares matrix

$$L = A^T A = \begin{bmatrix} \sum g_x g_x & \sum g_x g_y \\ \sum g_x g_y & \sum g_y g_y \end{bmatrix} \tag{33}$$

where the summation extends over the 3x3 window and evaluated both K_L (the condition number of L with respect to inversion) and $K_{trans} = \|L^{-1}\|^{1/2}$ (the condition number with respect to matching via translation, see Theorem 1). Running this test 100 times we found that the inversion condition number stayed in the range $1.2 \leq K_L \leq 7.4$ indicating excellent conditioning with respect to inversion for each of the 100 test samples! At the same time the translation matching condition number was always near 10^4 denoting bad conditioning, and in general behaves like $K_{Trans} \approx 1/\sqrt{\sigma}$ for this example. Clearly something is wrong with using K_L as a condition measure for optical flow for this problem: K_L stays near 1 even though for the underlying problem (i.e., without the noise) L is doubly rank deficient.

C. Least Squares Corner Detection

The appearance of the work of Fonseca et al. [16], [17] was the motivation for the current investigation of selecting tiepoints based on conditioning with respect to translation. In [17] the question was raised of how translation conditioning compares with other point selection procedures. One standard approach to corner detection is to use the least squares procedure discussed in Haralick and Shapiro [19]. We can summarize this approach as follows. An idealized corner consists of the intersection of lines associated with regions of constant intensity. Let $X_c = (x_0, y_0)^T$ be the corner location. At any point $X = (x, y)^T$ on an edge, the intensity gradient points perpendicular to the edge. At the same time, both X and X_c

lie along the edge (by assumption for both points). This means that the line connecting X to X_c is perpendicular to the gradient at X:

$$\nabla g \cdot (X - X_c) = 0 \tag{34}$$

This equation is also true away from the edges if we assume that the image intensity is constant within regions (i.e., $\nabla g = 0$ inside regions). Rewrite this equation as $\nabla g X_c = \nabla g X$. Letting X vary over all the points X_i in the window gives a system of over determined equations for the unknown corner location X_c:

$$AX_c = b \tag{35}$$

where A is a 2 column matrix with row i equal to the gradient of g at the point X_i. The ith entry of the vector b is equal to $\nabla g(X_i)X_i$.

This overdetermined system has the least squares solution

$$X_c = (A^T A)^{-1} A^T b \tag{36}$$

where we assume that the normal matrix $A^T A$ is invertible.

Note that there are two limiting assumptions used in deriving the least squares system of equations. First the corner is the intersection of straight edges. This limits the size of the window that can be used since most edges are not straight over long distances. Second, the regions are assumed to have constant intensity, so that $\nabla g = 0$ inside regions. This assumption can be violated in images subject to speckle. The least squares form of the corner equations has some resemblance to the form of the LS operator for optical flow but with one significant difference: the term b in the corner equations has entries $b_i = \nabla g_i X_i$. This means that the equations of the corner are homogeneous of order zero and hence invariant with respect to rescaling. That is if g is replaced by sg for any scale factor s then we have the same set of equations. In many situations such invariance is desirable but in this circumstance it is not since it means that tiny amounts of noise in an otherwise uniform region can produce the effect of a strong corner signal. This is similar to the problem of measuring the reliability of optical flow by the condition number of the least squares system as discussed above in Example 2.

Two other methods of selecting points for matching should be mentioned. The first looks for points (x, y) with maximal curvature of the intensity level lines through the point as measured by

$$\kappa = \frac{g_{xx} g_y^2 - 2 g_{xy} g_x g_y + g_{yy} g_x^2}{\left(g_x^2 + g_y^2\right)^{3/2}} \tag{37}$$

The curvature of the level line has been shown [24] to be to be inversely related to the conditioning of the normal matrix L as in Example 2 above. Thus high curvature indicates L is well conditioned but as in Example 2 this may not indicate anything more than the presence of noise. The other popular method is selecting points for matching potential is to pick points where the gradient norm

$$\|\nabla g\| = \sqrt{g_x^2 + g_y^2} \tag{38}$$

is maximized. Unfortunately this can easily occur along an edge with an aperture problem.

IV. Application to Image Registration

A. Selecting Tiepoints for Registration

The translation condition number K_{Trans} provides a measure of the suitability of a point (x, y) for matching purposes. Large condition numbers indicate bad matching potential so we define a set $S = S(g)$ of potential matching points as the set of points (x, y) in an image g where the translation condition number achieves a local minimum. This is illustrated in Figures 3 and 4 for a pair of agricultural images.

Not all points in $S(g)$ can be matched to points in $S(\hat{g})$. This might be because the points have moved out of frame or because of sensor noise or temporal changes in the image. To eliminate mismatched points we use a two pass approach that is somewhat similar to the RANSAC method [15] as described by Hartley and Zisserman [21]. In the first pass of our procedure, points from $S(g)$ are tentatively matched with points in $S(\hat{g})$. This preliminary matching is accomplished by comparing features at each tiepoint in the first image with features at each tiepoint in the second image. To guard against rotation effects, we have taken the feature vectors to be windows about each tiepoint that have been rotated so that their central gradient points downward. The windows have also been rescaled to have zero mean and unit variance in order to protect against contrast changes that might occur between images such as that produced by a bright object appearing in the second image but not the first. Subsequent to this the feature vectors are compared by computing the mean square error. We also note that in order to limit the computer effort in registration we have pruned the potential tiepoints aggressively with respect to conditioning; thus many obviously good matches are discarded and do not appear in the figures.

The initial feature matching can be used to cull points from both images that have no corresponding point in the opposing image. At the same time a preliminary correspondence between points in the two images is obtained. The second pass refines this correspondence by using a purely geometric matching procedure in which the location of points and their relationship to each other determine the matching.

In this geometric matching phase we use the assumption that the images are related by a rotation-scale-translation (RST) transformation as described below[1] . In this type of transformation we assume the image $\hat{g}$ is related to g by

$$\hat{g}(\hat{x}, \hat{y}) = Ag(x, y) + B$$

where A and B are constants (possibly induced by contrast changes due to a bright object moving in or out of frame) and

$$\begin{bmatrix} \hat{x} \\ \hat{y} \end{bmatrix} = r \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} dx \\ dy \end{bmatrix}$$

This type of transformation applies in many settings of interest such as frame to frame registration of video sequences with sufficiently high frame rate relative to the camera motion. However the techniques described for RST transformations can also be applied to more general transformations.

In the geometric culling of the tiepoints, pairs of points are selected in each image. The associated RST transformation for these pairs is computed and then applied to the entire group of tiepoints. The number of tiepoints in the first image that land on a tiepoint in the second under the RST transformation then gives a measure of the accuracy of the RST parameters for the initial pairs of points. Selecting the RST with the highest score provides a means of eliminating outliers as those points without good matching tiepoints after the transform is applied. Once the RST parameters r, θ, dx, dy are computed we then can determine the contrast constants A, B by using the RST parameters to find the image-overlap region under back-transformation. Then we select A, B so that the backtransformed image has the same mean and variance as g in the overlap region.

Because the first and second culling passes are based on different matching criteria they provide a safeguard against mismatched points. Moreover the preliminary culling in the first pass reduces the computational burden of the second pass, resulting in a fast combined matching algorithm.

Figures 3 and 4 illustrate the two pass matching procedure for a pair of agricultural images. Figure 3 shows the original images and the 169 initial tiepoints. Of these only 69 tiepoints have matching points in the opposite image. The number of matches was determined by first using hand picked tiepoints to determine the registration parameters and then transforming the first set of tiepoints into the frame of the second followed by comparison of the tiepoint indices. Any index pair with difference less than one pixel was considered a matching pair.

[1] Note however that RST transformation cannot cope with out of plane rotations of the scene: even the orthographic projection of a planar scene would require an affine model.

Figure 4a,b shows the tiepoints after pass 1 (feature matching) has culled out some of the worst fits. There are now 32 tiepoints in each image and 28 of these are matching tiepoints. This is a match rate of 88 percent.

Figure 4c,d shows the tiepoints after pass 2 (both feature matching and geometric matching). There are now 26 tiepoints in each image and all of these are matching tiepoints. This is a match rate of 100 percent.

We also illustrate the performance of the two-pass tiepoint matching on several sets of images in Table 1. The associated images are shown in Figure 5 (coastline images), Figure 6 (urban images formed synthetically from one larger image using RST transformations), Figure 7 (noisy IR urban images).

Note that for the coastline images the first pass achieves only moderate matching success. In part this is due to temporal differences in the images at the shoreline (waves) and the large translation reducing the tiepoint overlap. Large temporal differences are evident in the Amazon forest images (Figure 8) which are separated in time by 2 years and the Brazilian agricultural images (Figure 9) which are separated in time by 4 years.

In the Amazon images the feature matching is not very successful (16% match rate) but the geometric matching is able to overcome this. In the Brazilian agricultural images the temporal variations are so large that both the feature matching and the geometric matching fail. This indicates the limits of the two pass matching. On the positive side, however, the Brazilian images provide a nice challenge for the fit assessment tests which readily detect the registration failure (see below and Tables 2 and 3).

B. Fit Assessment and Post-Registration Error Estimation

Once the match between tiepoints has been computed, we find the associated transform parameters (in our work the rotation angle, scale factor and translation shift vector for the RST transform) for the matched points using a standard least squares procedure (see [18]) applied to the set of equations relating the tiepoints via the RST transformation. The transform parameters allow us to associate with each pixel (x_2, y_2) in image 2 a corresponding pixel (x_1, y_1) in image 1 as per (4). This association allows us to back transform the intensity at (x_2, y_2) to (x_1, y_1) resulting in a 'back-transformed' image $\hat{g}^{back}$ which we can then compare to the first image g. The difference between these images gives us a measure of the acceptability of the computed transform parameters.

We have employed two kinds of tests in determining acceptability. The first group measures the pointwise difference between the first image and the back-transform of the second image. (To calculate the difference in such a way as to avoid intensity changes between frames we find the common overlap of the first image and the back transformed image. Then we subtract the respective intensity means and divide by the intensity standard deviations before the pointwise subtraction over the common overlap region. The sum of these pointwise differences is then normalized by dividing by the number of pixels in the overlap region.) This is done by using three independent tests. In the first two tests, a statistical procedure is used to characterize the back-transform difference for good and bad registration parameter sets. To estimate the 'bad' fit, we generate random values for the transform parameters and then backtransform the second image and take the norm of the difference with the first image. Repeating this q times to generate a sample of values gives us an estimate of the mean and standard deviation of the norm of the image difference for a random (or bad) transform. (We used $q = 32$ which gave reasonably accurate estimates for the mean and variance for the images we tested without excessive computational effort.) We then compare the backtransform difference $d_{measured}$ (from the tiepoint transform parameters) with the mean d_{bad} and standard deviation σ_{bad} of the bad fit. The value $k = (d_{measured} - d_{bad})/\sigma_{bad}$ tells us how many standard deviations the computed value $d_{measured}$ is from d_{bad}. For example if $k >> 3$ then we reject the hypothesis that the computed transform parameters are no better than randomly selected transform parameters.

For the good fit test we employ a similar procedure. In this case we want to know the expected value and standard deviation of the norm of the image difference when the backtransformed image is near the true image. This can be done in several ways; the easiest (which we used in the numerical studies for this paper) is to select transform parameters randomly from ranges that are very near the identity (we limited the transform ranges to produce pixel offsets of one pixel or less). Applying the resulting transform to the first image and then differencing with the first image mimics the effect of having a nearly exact set of transform parameters for the second image. This approach tends to give conservative results for the reason that temporal differences in the second image are not measured in setting up the mean and standard deviation of the good fits (because the good fit estimates are derived from the first image only). Thus this test occasionally rejects computed transform parameters which actually are acceptable.

This pair of "good fit-bad fit" statistical tests is supplemented with a bootstrap parameter variation test in which subsets of the matched tiepoints are used to recompute the transformation parameters. Large parameter variations over the subsets indicates the presence of mismatched tiepoints [14]. Altogether this suite of three acceptability tests provides a powerful mechanism for detecting improper registrations.

The second type of error detection procedure that we use estimates the post-registration error between the first image and the back-transformation of the second image. This is useful for problems of motion detection and target identification and is found by minimizing the translation objective function at the set of points $S(g)$ in the image that have good conditioning with respect to translation. The minimization in done with respect to the first image g and the back-transform image $\hat{g}^{back}$:

Post-Registration Error Estimation Algorithm: At each point (x, y) in $S(g)$ determine the minimizing value of $(a, b) = (a(x, y), b(x, y))$ for the objective function

$$f(a,b) = \frac{1}{2}\,\Sigma\left(g(x',y') - \hat{g}^{back}(x'+a, y'+b)\right)^2 \tag{39}$$

where the summation is taken over the window about (x, y).

We then estimate the overall post-registration error either as the maximum or average of $\sqrt{a^2 + b^2}$ over $S(g)$. Offset error uncertainty ellipses can be determined by finding the covariance matrix for (a, b) over $S(g)$.

Note however that if we use an overall error assessment we may be merging two different types of errors; an illustration of this is given in Example 3 below in which some of the errors are due to registration parameter uncertainty (these generate small errors since the computed registration parameters are nearly correct) and the remaining errors are due to model mis-match (large disparity produced by projective geometry effects that are not included in the RST transform).

Also note that in this offset error estimation procedure, we use a translation objective function that reverses the roles of the first and second images. This is done to avoid having to compute $S(\hat{g}^{back})$.

Remark. The K_{Trans} was derived for image pairs related locally by shift plus noise. If the computed registration parameters (rotation, scale, translation) are reasonably accurate then this assumption will hold for comparing the original image with the back-transformation of the second image since the back-transformation removes the effect of rotation and scaling (zoom) and intensity contrast changes. Because of this we might expect good accuracy in our offset error estimates; this is supported by our numerical tests as reported in Table 4. For more details see report [25].

C. Experimental Results

The programs for registration described in this paper can be accessed on the www [57] where one register user supplied pairs of images or just try out some of the examples of this paper.

Table 2 reports the results for the sets of images considered earlier in Table 1. In this table the images have been normalized to have zero mean and unit variance. There are several remarks to make about these results. The first is that the two pass matching procedure for the coastline images 1-4 produces excellent results (as compared to "ground-truth" transformations determined by hand-picked tiepoints) but the predicted fit error for a good match is about half of the computed fit error. The reason for this is that the temporal variation in the coastline images (waves at the shoreline) is not included in the model for the good fit error estimate. This kind of model mismatch can produce false negatives for the good fit error test. For this reason the good fit test can be somewhat conservative. This kind of problem does not occur for the bad fit test since minor model mismatch is overshadowed by the extent of the maximum pixel error for the bad fit error estimate.

For the coastline images 1-4 the bad fit error estimate is well removed from the computed fit (by more than 3 standard deviations). This allows us to use the bad fit test to reject the hypothesis that the computed fit is no better than a random fit.

For the Brazilian agricultural images, both the good and bad fit tests indicate that the computed registration is no better than a random fit. This is supported by the bootstrap uncertainty estimates in Table 3.

A second point is that sometimes the computed fit is much smaller than the estimated good fit error as in the urban images 1-4 and the IR urban images 1-4. This simply indicates that the computed transformation is almost exact and the computed fit error consists mostly of the residual interpolation error. For example, for urban images 1 and 2, the computed fit error was 0.0405 and the interpolation error estimate using the Laplacian approximation was 0.0329.

A third point is that as a test we tried to register two coastline images with no overlap (i.e. an impossible registration). In this case the problem is readily detected by comparing the computed fit error (1.06) with the estimated bad fit error (1.10 ± 0.03) as seen in the last line of the table. In this case the bad fit test indicates that the computed registration is no better than a random fit - exactly what we would expect when trying to register two images with no overlap.

To illustrate the bootstrap tests we have determined the true RST parameters for several images by first selecting 16 tiepoints in each image by hand, then computing the associated RST parameters and refining them using gradient descent on the fit error function. The results are reported in Table 3 below.

Table 4 compares the results of applying the offset error estimation algorithm (39) to those determined by using the exact transformation parameter values. Note that for the coastline images temporal change gives a large value for the maximum offset error although the average error is small (subpixel). This illustrates that the offset error estimation algorithm (39) can be quite useful in detecting target motion not associated with the dominant background motion. The very small error for the urban images is probably due to the fact that the images in this sequence were formed synthetically from one larger image using RST transformations. The very large temporal variations for the Amazon and Brazil images made it impossible to assign exact offset error by handpicked tiepoints. Additionally the very bad fit of the computed parameters for the Brazil images (detected successfully by the good-bad fit test) made it impossible to apply the offset error estimation (39). This points out the importance of having multiple reliability tests for computed registration parameters and transforms.

Example 3. We conclude this section with an example that shows the effects of model mismatch. Figure 10 presents two images taken from a helicopter undergoing pure translational motion above an urban street. The motion induces parallax as seen in the foreshortening of the building. The projective geometry transformation between images is not an RST transformation. To see how this impacted our procedure we ran the program and computed the RST parameters.

Once we computed the RST transformation from the tiepoints, we then stabilized the second image by back-transforming to the frame of reference of the first image. Figure 11 (left) shows the difference between the first image and the stabilized second image. Notice the large error in the part of the image corresponding to the side of the building. The parallax effects of the side of the building are mismodeled by the RST transformation. In order to detect post-registration errors we select error estimation points (Figure 11 right) that are well conditioned for translational matching. These points are not the same as the tiepoints used in the registration because the registration model (RST) culls out those points that don't fit the model (i.e., the points on the side of the building showing parallax).

Figure 12 (left) is a plot of the post-registration error vectors (v_x, v_y) as calculated at the points in Figure 11. Notice that the errors cluster into two groups. The smaller errors are due to inaccuracy in the computed registration parameters. These errors are on the order of 0.6 pixels. (Note: 21 of the 55 points are right at (0,0).) This is subpixel accuracy for the points not showing parallax. The other cluster has an average offset error of 9.4 pixels; this group of 8 points are the ones on the side of the building and do not fit the RST model. This is seen in Figure 12 (right) which shows the location and magnitude of the offset errors: darker is larger error. All the dark points cluster on the building. This illustrates that we shouldn't lump all the error vectors together to form a covariance error matrix since this may be merging two kinds of errors.

V. Conclusion

Condition numbers for point-matching have been derived for translation, rotation-scaling-translation and affine window matching between images, under a restricted but reasonable problem type: 'transform plus noise' as described in Theorem 1. It is shown in Theorem 2 that ill-conditioning with respect to translation implies ill-conditioning with respect to RST and affine transforms. This means that the translational condition number may be used to remove from consideration ill-conditioned points in the image for the purpose of point-matching. The translational condition number exhibits the desirable features of rotational invariance, ease of computability, and single image definition. Moreover, for the group of images that we tested, tiepoints selected on the basis of this condition number provide a reliable basis for registration when coupled with post-registration tests for fit assessment and post-registration error estimation. In particular the nature of the offset error estimation conforms closely with the original assumption of translation plus noise and hence is quite accurate. A web site is available [57] at where one may try our registration program on user supplied image pairs.

Acknowledgements: The authors would like to thank Leila Fonseca, INPE, Brazil for her many fruitful discussions, and Dmitry Fedorov for his help in implementing the web registration demo.

Appendix

A. Proof of Theorem 1

Proof. We derive the condition number for translational matching; the condition numbers for RST and affine objective functions can be found by the same procedure.

When the noise is nonzero, the best shift is given by $(a + \Delta a, b + \Delta b)$. The condition number for the restricted problem is

$$K_{Trans} = \lim_{\delta \to 0} \max_{\|\eta\| \le \delta} \|(\Delta a, \Delta b)\| / \|\eta\|$$

In order to evaluate K_{Trans} we need to find an expression for $(\Delta a, \Delta b)$ in terms of the perturbation η. Denoting the partial derivative of f with respect to x with f_x and setting the gradient with respect to

(a, b) of the objective function to zero at the minimizer, we obtain

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} f_a(a+\Delta a, b+\Delta b) \\ f_b(a+\Delta a, b+\Delta b) \end{bmatrix}$$

$$= \begin{bmatrix} \Sigma\left(g(x'+a+\Delta a, y'+b+\Delta b) - \hat{g}(x',y') - \Delta\hat{g}(x',y')\right)\ g_x(x'+a+\Delta a, y'+b+\Delta b) \\ \\ \Sigma\left(g(x'+a+\Delta a, y'+b+\Delta b) - \hat{g}(x',y') - \Delta\hat{g}(x',y')\right)\ g_y(x'+a+\Delta a, y'+b+\Delta b) \end{bmatrix}$$

$$= \begin{bmatrix} \Sigma\left(g(x'+a+\Delta a, y'+b+\Delta b) - g(x'+a, y'+b) - \eta(x',y')\right)\ g_x(x'+a+\Delta a, y'+b+\Delta b) \\ \\ \Sigma\left(g(x'+a+\Delta a, y'+b+\Delta b) - g(x'+a, y'+b) - \eta(x',y')\right)\ g_y(x'+a+\Delta a, y'+b+\Delta b) \end{bmatrix}$$

where all the summations are defined for x' and y' varying over the window centered at (x, y) and $\hat{g}(x',y') = g(T(x',y')) + \Delta\hat{g}(x',y') = g(x'+a, y'+b) + \hat{\eta}(x',y')$. Expanding the previous expression we obtain

$$\begin{bmatrix} 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \Sigma g_x^2(x'+a, y'+b)\Delta a + g_x(x'+a, y'+b)g_y(x'+a, y'+b)\Delta b - g_x(x'+a, y'+b)\eta(x',y') \\ \\ \Sigma g_x(x'+a, y'+b)g_y(x'+a, y'+b)\Delta a + g_y^2(x'+a, y'+b)\Delta b - g_y(x'+a, y'+b)\eta(x',y') \end{bmatrix}$$

$$+ \quad \text{higher order terms}$$

Thus

$$\begin{bmatrix} \Delta a \\ \Delta b \end{bmatrix} = \left(A^T A\right)^{-1} A^T \tilde{\eta} + \quad \text{higher order terms}$$

where

$$A = \begin{bmatrix} g_x(x^1+a, y^1+b) & g_y(x^1+a, y^1+b) \\ \vdots & \\ g_x(x^n+a, y^n+b) & g_y(x^n+a, y^n+b) \end{bmatrix} \qquad \tilde{\eta} = \begin{bmatrix} \eta(x^1, y^1) \\ \vdots \\ \eta(x^n, y^n) \end{bmatrix}$$

where the superscripts denote ranging over the window centered at (x, y).

The expression for the condition number K_{Trans} now follows from taking the limit as $\eta \to 0$ in the above and replacing $g(x'+a, y'+b)$ by $\hat{g}(x', y')$ to get

$$A = \begin{bmatrix} \hat{g}_x^1 & \hat{g}_y^1 \\ \vdots & \\ \hat{g}_x^n & \hat{g}_y^n \end{bmatrix}$$

which completes the proof of the theorem. □

B. *Proof of Theorem 2*

Proof. The form of A means that

$$A^T A = \Sigma \left(v^i\right)^T v^i$$

where the v^i are determined by T as given in Theorem 1. Since we are working with the 2-norm we use the fact that

$$\begin{aligned} \| \left(A^T A\right)^{-1} A^T \|^2 &= \| \left(A^T A\right)^{-1} A^T \left(\left(A^T A\right)^{-1} A^T\right)^T \| \\ &= \| \left(A^T A\right)^{-1} \| \\ &= \frac{1}{\lambda_{min} \left(A^T A\right)} \end{aligned}$$

Now note that

$$v_{Trans} = v_{RST}\ Q_1 = v_{RST} \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad v_{RST} = v_{Affine}\ Q_2 = v_{Affine} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

This gives

$$\left(A^T A\right)_{Trans} = \Sigma \left(v^i_{Trans}\right)^T v^i_{Trans} = Q_1^T \left[\Sigma \left(v^i_{RST}\right)^T v^i_{RST}\right] Q_1 = Q_1^T \left(A^T A\right)_{RST} Q_1$$

Since the 2-norm of Q_1 is one, all of the eigenvalues of $\left(A^T A\right)_{Trans}$ satisfy the interlacing bound [30]

$$\lambda_{min} \left(A^T A\right)_{RST} \le \lambda \left(A^T A\right)_{Trans} \le \lambda_{max} \left(A^T A\right)_{RST}$$

In particular we have

$$K_{Trans} = \frac{1}{\lambda_{min}\left(A^T A\right)_{Trans}} \leq \frac{1}{\lambda_{min}\left(A^T A\right)_{RST}} = K_{RST} \tag{40}$$

We complete the proof by noting that

$$\left(A^T A\right)_{RST} = Q_2^T \left(A^T A\right)_{Affine} Q_2$$

Now let w be an eigenvector of unit norm for $\left(A^T A\right)_{RST}$ corresponding to $\lambda_{min}\left(A^T A\right)_{RST}$. Then

$$\begin{aligned} \lambda_{min}\left(A^T A\right)_{RST} &= w^T \left(A^T A\right)_{RST} w \\ &= w^T Q_2^T \left(A^T A\right)_{Affine} Q_2 w \\ &\geq \|Q_2 w\|^2 \lambda_{min}\left(A^T A\right)_{Affine} \\ &\geq \lambda_{min}\left(A^T A\right)_{Affine} \end{aligned}$$

since $1 \leq \|Q_2 w\|^2 \leq 2$. From this we have

$$K_{Affine} = \frac{1}{\lambda_{min}\left(A^T A\right)_{Affine}} \geq \frac{1}{\lambda_{min}\left(A^T A\right)_{RST}} = K_{RST} \tag{41}$$

Therefore from (40) and (41) it follows that $K_{Trans} \leq K_{RST} \leq K_{Affine}$. □

C. Sensitivity Operation Count

In order to evaluate the condition number with respect to point matching for translation we first need to form the matrix

$$A^T A = \begin{bmatrix} \Sigma \hat{g}_x^2 & \Sigma \hat{g}_x \hat{g}_y \\ \Sigma \hat{g}_x \hat{g}_y & \Sigma \hat{g}_y^2 \end{bmatrix}$$

where the summation is over (x', y') in a window centered at (x, y). The norm of the inverse of this matrix is the desired condition number.

The arithmetic operations (multiplication, addition, subtraction, or division) needed to compute the condition number can be summarized as follows:

1) Form g_x and g_y (using differences) at a cost of 2 differences per pixel and then form the products $\hat{g}_x^2,\ \hat{g}_x \hat{g}_y,\ \hat{g}_y^2$. Each product costs one arithmetic operation per pixel. The total cost for this step is 5 operations per pixel.

2) Now we need to form the block sums in A^TA. Block sums are efficiently computed by first calculating rows sums (with a moving window); the cost for each window row sum is two operations since we can update the previous row sum by adding and subtracting elements:

$$\sum_{i}^{i+n} a_j = a_{i+n} - a_{i-1} + \sum_{i-1}^{i+n-1} a_j \tag{42}$$

where a_j is the quantity being summed. Neglecting the start up costs we see that the cost per pixel of a windowed row sum is 2 operations. After row summing, we column sum to get the block sums. This costs an additional 2 operations per pixel. Thus each block sum can be computed at a cost of 4 operations per pixel. Since we need 3 block sums the total cost for this step is 3x4=12 operations per pixel.

3) To finish the condition evaluation we use

$$\begin{aligned} K^2_{Trans,Schatten} &\equiv \| \left(A^TA + \epsilon I\right)^{-1} \|_S \\ &= \frac{2\epsilon + \Sigma\hat{g}_x^2 + \Sigma\hat{g}_y^2}{\left(\Sigma\hat{g}_x^2 + \epsilon\right)\left(\Sigma\hat{g}_y^2 + \epsilon\right) - \left(\Sigma\hat{g}_x\hat{g}_y\right)^2} \end{aligned}$$

This can be evaluated in 7 arithmetic operations per pixel.

The total count is 5+12+7=24 operations per pixel.

REFERENCES

[1] K. Atkinson, *An Introduction to Numerical Analysis*, 2nd Edition, John Wiley and Sons, New York, 1989.

[2] H. S. Baird, *Model-Based Image Matching Using Location*, MIT Press, Cambridge, Massachusetts, 1985.

[3] J.L. Barron, D.J. Fleet and S. Beauchemin, "Performance of Optical Flow Techniques," *International Journal of Computer Vision*, 12(1):43-77, 1994.

[4] S.S. Beauchemin and J.L. Barron, "The Computation of Optical Flow," *ACM Computing Surveys*, 27(3):433-467, 1996.

[5] M. Ben-Ezra, S. Peleg and M. Werman,"Robust Real-Time Motion Analysis," In Proc. DARPA IUW, 207-210, 1988.

[6] J. Bergen, P. Anandan, K. Hanna and R. Hingorani, "Hierarchical Model-Based Motion Estimation," *European Conf. Comp. Vis.*, ,(1992), pp. 237–252.

[7] A. Björck and S. Hammarling, "A Schur Method for the Square Root of a Matrix," *Lin. Alg. Appl.*, 52/53 (1983), pp. 127–140.

[8] M. Brooks, W. Chojnacki, D. Gaeley and A. can den Hengel, "What Value Covariance Information in Estimating Vision Parameters?," *Proc. IEEE ICCV*, 302-308, 2001.

[9] L. G. Brown, "A survey of image registration techniques," *ACM Computing Surveys* **24**(4), pp. 325–376, 1992.

[10] A. Cline, C. Moler, G. Stewart and J. Wilkinson, "An Estimate for the Condition Number of a Matrix," *SIAM J. Num. Anal.*, 16 (1979), pp. 368–375.

[11] J. Demmel, “On Condition Numbers and the Distance to the Nearest Ill-Posed Problem,” *Numer. Math.*, 51 (1987), pp. 251–289.

[12] J. Demmel, “The Probability that a Numerical Analysis Problem is Difficult,” *Math. Comp.*, 50 (1988), pp. 449–480.

[13] J. Demmel, B. Diament and G. Malajovich, “On the Complexity of Computing Error Bounds,” *Foundations of Computation Math.*, 1 (2001), pp. 101–125.

[14] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans*, SIAM CBMS-NSF Regional Conference Series in Applied Mathematics, Sixth Prining, 1994.

[15] M. A. Fischler and R. C. Bolles, ”Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography,” *Comm. Assoc. Comp. Mach.* 24(6), pp. 381-395, 1981.

[16] L. Fonseca, G. Hewer, C. Kenney, and B. Manjunath, “Registration and Fusion of Multispectral Images Using a New Control Point Assessment Method Derived from Optical Flow Ideas,” *Proc. SPIE* , Vol. 3717, pp. 104–111, April 1999, Orlando, FLA.

[17] L. Fonseca and C. Kenney, “Control Point Assessment for Image Registration,” *Proc. XII Brazilian Symposium on Computer Graphics and Image Processing*, Campinas, Brazil, October 1999, pp. 125-132.

[18] G. H. Golub and C. F. Van Loan, *Matrix Computations*, John Hopkins Univ. Press, Baltimore, 2nd Edition, 1989.

[19] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision* Vol. 2, Addison-Wesley, New York, 1993.

[20] C. Harris and M. Stephens, “A Combined Corner and Edge Detector,” *Proc. Alvey Vision Conf.*, 147-151, 1988.

[21] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.

[22] G. Hewer and C. Kenney, “The Sensitivity of the Stable Lyapunov Equation,” *SIAM J. Control and Optim.*, 26 (1988), pp. 321–344.

[23] G. Hewer, C. Kenney and W. Kuo, “A Survey of Optical Flow Methods for Tracking Problems,” *Proceedings SPIE Conference*, April 1994, Orlando, Florida, vol. 2242, pp. 561-572.

[24] G. Hewer, C. Kenney, W. Kuo and L. Peterson, “Curvature and Aggregate Velocity for Optical Flow,” *Proceedings SPIE Conference*, July 1995, San Diego, California.

[25] G. Hewer, C. Kenney, B. Manjunath and A. Van Nevel, “Condition Theory for Image Processing,” ECE Techincal Report, University of California, Santa Barbara, CA 93106, March 2002.

[26] E. C. Hildreth, “Computations Underlying the Measurement of Visual Motion,” *Artificial Intelligence*, vol. 23 (1984), pp. 309–354.

[27] D. Hinrichsen and A. J. Pritchard, “Stability Radii of Linear Systems,” *Syst. Contr. Lett.*, 7 (1986), pp. 1–10.

[28] B. K. P. Horn and B. Schunck, “Determining Optical Flow,” *Artificial Intelligence*, 17 (1981), pp. 185-203.

[29] B. K. P. Horn, *Robot Vision*, MIT Press, Cambridge, MA, 1986.

[30] R. A. Horn and C. R. Johnson, *Topics in Matrix Analysis*, Cambridge Univ. Press, Cambridge, 1991.

[31] M. Irani, B. Rousso and S. Peleg, “Computing Occluding and Transparent Motion,” *Intern. J. Comp. Vision*, 12 (1994), pp. 5–16.

[32] M. Irani and P. Anandan, “Robust Multi-Sensor Image Alignment,” *Proc. IEEE ICCV*, 959-966, 1998.

[33] Y. Kanazawa and K. Kanatani, “Do We Really Have to Consider Covariance Matrices for Image Features?,”*Proc. IEEE ICCV*, 301-306, 2001.

[34] Joseph K. Kearney, William B. Thompson and Daniel L. Boley, “Optical Flow Estimation: An Error Analysis of Gradient-Based Methods with Local Optimazation,” *IEEE Trans Pat. Anal. Mach. Intell.*, Vol PAMI-9, No. 2, March, (1990), pp. 229–244.

[35] C. Kenney and G. Hewer, "The Sensitivity of the Algebraic and Differential Riccati Equations," *SIAM J. Control and Optim.*, 28 (1990), pp. 50–69.

[36] C. S. Kenney and A.J. Laub, "Condition Estimates for Matrix Functions," *SIAM J. Matrix Anal. Appl.*, 10 (1989), pp. 191–209.

[37] C. S. Kenney and A.J. Laub, "Small-Sample Statistical Condition Estimates for General Matrix Functions," *SIAM J. Sci. Comp.*, 15 (1994), pp. 36–61.

[38] H. Li, B.S. Manjunath and S.K. Mitra, " A Contour Based Approach to Multisensor Image Registration," *IEEE IP*, 320-334, 1995.

[39] B. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. Image Understanding Workshop*, (1981), pp. 121–130.

[40] D. G. Luenberger, *Linear and Nonlinear Programming*, 2nd Edition, Addison-Wesley, Menlo Park, California, 1984.

[41] J. Tanner and C. Mead, "Optical motion sensor," in *Analog VLSI and Neural Systems* (Carver Mead, ed.), chapter 14, pp. 229–255, Reading, MA: Addison-Wesley Publishing Company, 1989.

[42] C. B. Moler and C. F. Van Loan, "Nineteen Dubious Ways to Compute the Exponential of a Matrix," *SIAM Review*, 20 (1978), pp. 801–836.

[43] J. R. Rice, "A Theory of Condition," *SIAM J. Num. Anal.*, 3 (1966), pp. 287–310.

[44] C. Schmidt, R. Mohr and C. Baukhage, "Evaluation of Interest Point Detectors," *IJCV*, 27: 151-172, 2000.

[45] J. Shi and C. Tomasi, "Good Features to Track," CVPR, 593-600, 1994.

[46] T. Tommasini and A. Fusiello and E. Trucco, and V. Roberto, "Making good features track better," CVPR, 178-183, 1998.

[47] G. W. Stewart, "Error and Perturbation Bounds for Subspaces Associated with Certain Eigenvalue Problems," *SIAM Review*, 15 (1973), pp. 727–764.

[48] V. Sundareswaran and S. Mallat, "Multiscale optical flow computation with wavelets," Preprint. Courant Institute of Mathematical Sciences, New York University, 1992.

[49] E. Trucco and A. Verri, *Introductory Techniques for 3D Computer Vision*, Prentice-Hall, Upper Saddle River, NJ, 1998.

[50] B. C. Vemuri, S. Huang, S. Sahni, C. M. Leonard, C. Mohr, R. Gilmore and J. Fitzsimmons, "An efficient motion estimator with application to medical image registration," Medical Image Analysis, Oxford University Press, Vol.2, No. 1, pp. 79-98, 1998

[51] R. C. Ward, "Numerical Computation of the Matrix Exponential with Accuracy Estimate," *SIAM J. Numer. Anal.*, 14 (1977), pp. 600–614.

[52] R. Wildes, D. Horvonen, S. Hsu, R. Kumar, W. Lehman, B. Matei and W. Zhao, "Video Georegistration: Algorithm and Quantitative Evaluation," *Proc. ICCV*, 343-350, 2001.

[53] C. Van Loan, "How Near is a Stable Matrix to an Unstable Matrix?," *Contemporay Math.*, 47 (1985), pp. 465–477.

[54] J. H. Wilkinson, "Error Analysis of Direct Methods of Matrix Inversion," *J. ACM*, 10 (1961), pp. 281–330.

[55] J. H. Wilkinson, *The Algebraic Eigenvalue Problem*, Clarendon Press, Oxford, England, 1965.

[56] Q. Zheng and R. Chellappa, "A Computational Vision Approach to Image Registration," *IEEE IP*, 2(3): 311-326, 1993.

[57] **http://vision.ece.ucsb.edu/demos.html**. Follow the link to the Image Registration demo.

TABLE I

PERCENT OF CORRECT POINT-MATCHING FOR BOTH FEATURE MATCHING AND GEOMETRIC MATCHING.

Images	Percent Correct Point Match	Percent Correct Point Match	Remarks
	Pass One (Features)	Pass Two (Geometry)	
Coastline 1-2	69%	100%	Large x-shift
Coastline 2-3	66%	100%	Large x-shift
Coastline 3-4	59%	100%	Large x-shift
Agricultural 1-2	88%	100%	Medium x-shift
Agricultural 2-3	91%	100%	Medium x-shift
Agricultural 3-4	75%	100%	Medium x-shift
Urban 1-2	97%	100%	15 degree rotation
Urban 2-3	97%	100%	15 degree rotation
Urban 3-4	100%	100%	15 degree rotation
IR Urban 1-2	100%	100%	Very noisy
IR Urban 2-3	100%	100%	Very noisy
IR Urban 3-4	97%	100%	Very noisy
Amazonia 1-2	16%	100%	Large temporal variation
Brazil 1-2	13%	13%	Large temporal variation

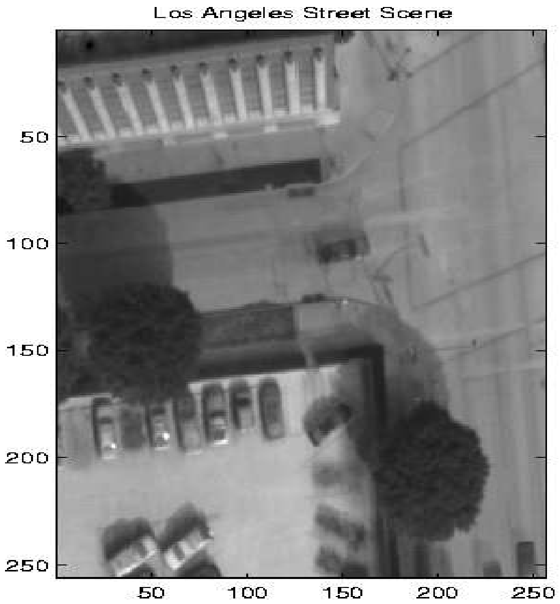

Fig. 1. Image Selected to Illustrate Point-Matching Condition Numbers for Translation, RST and Affine Transforms.

TABLE II

COMPARING COMPUTED FIT ERROR WITH ESTIMATED GOOD AND BAD FIT VALUES (UNCERTAINTIES IN THE ESTIMATED VALUES CORRESPOND TO ONE STANDARD DEVIATION). ALL IMAGES HAVE BEEN NORMALIZED TO HAVE ZERO MEAN AND UNIT VARIANCE.

Images	Computed Match Fit Error	Good Match Fit Error	Bad Match Fit Error
Coastline 1-2	0.51	0.24 ± 0.04	1.10 ± 0.09
Coastline 2-3	0.41	0.26 ± 0.07	1.04 ± 0.16
Coastline 3-4	0.37	0.23 ± 0.05	1.00 ± 0.05
Agricultural 1-2	0.23	0.25 ± 0.05	0.96 ± 0.15
Agricultural 2-3	0.25	0.28 ± 0.05	0.96 ± 0.13
Agricultural 3-4	0.37	0.29 ± 0.07	0.95 ± 0.13
Urban 1-2	0.04	0.10 ± 0.03	0.93 ± 0.14
Urban 2-3	0.03	0.09 ± 0.03	0.93 ± 0.15
Urban 3-4	0.03	0.09 ± 0.03	0.94 ± 0.16
IR urban 1-2	0.14	0.39 ± 0.07	1.05 ± 0.07
IR urban 2-3	0.13	0.41 ± 0.09	1.00 ± 0.10
IR urban 3-4	0.20	0.39 ± 0.09	1.05 ± 0.04
Amazonia 1-2	0.51	0.34 ± 0.05	0.97 ± 0.08
Brazil 1-2	0.96	0.35 ± 0.07	1.04 ± 0.04
Coastline images with no overlap	1.06	0.26 ± 0.05	1.10 ± 0.03

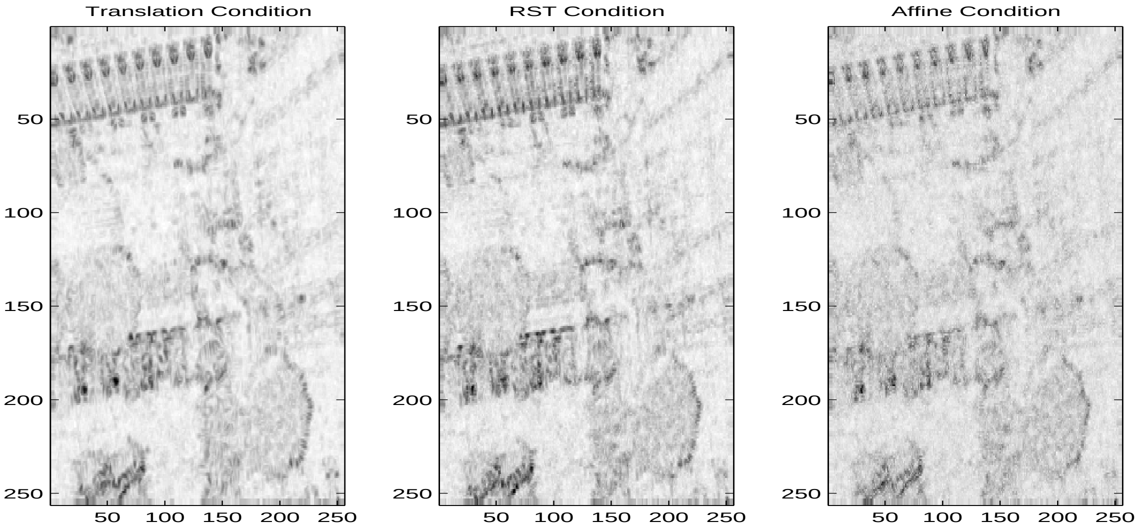

Fig. 2. Condition Numbers for Translation, RST and Affine Transformation for the LA Street Scene: Dark is Better Conditioned.

TABLE III

BOOTSTRAP PARAMETER UNCERTAINTY ESTIMATES.

Images		Scale Factor	Rotation Angle	X Translation	Y Translation
Coastline	Computed	0.99 ± 0.002	-1.09 ± 0.20	-193.6 ± 1.7	-2.1 ± 0.8
	Exact	0.99	-1.04	-193.9	-2.0
Agricultural	Computed	1.00 ± 0.001	2.99 ± 0.09	19.7 ± 0.2	0.2 ± 0.2
	Exact	1.00	3.03	19.7	0.3
Urban	Computed	1.00 ± 0.0002	-15.00 ± 0.02	51.6 ± 0.15	-89.6 ± 0.07
	Exact	1.00	-15.00	51.2	-89.5
Amazonia	Computed	1.00 ± 0.003	0.03 ± 0.10	36.6 ± 0.4	-183.3 ± 0.6
	Exact	1.00	-0.07	37.5	-183.7
Brazil	Computed	0.40 ± 0.12	-95 ± 31	317 ± 49	116 ± 60
	Exact	1.00	0	0	0

Images	Estimation Method	Average Offset Error	Maximum Offset Error
Coastline	Estimated via (39)	0.9	5.4
	Exact	0.9	5.4
Agricultural	Estimated via (39)	0.5	1.4
	Exact	0.5	1.4
Urban	Estimated via (39)	0.0	0.0
	Exact	0.0	0.0
Amazonia	Estimated via (39)	1.2	7.2
	Exact	–	–
Brazil	Estimated via (39)	–	–
	Exact	–	–

TABLE IV

AVERAGE AND MAXIMUM POST-REGISTRATION ERROR FOR THE BACK-TRANSFORMED IMAGES.

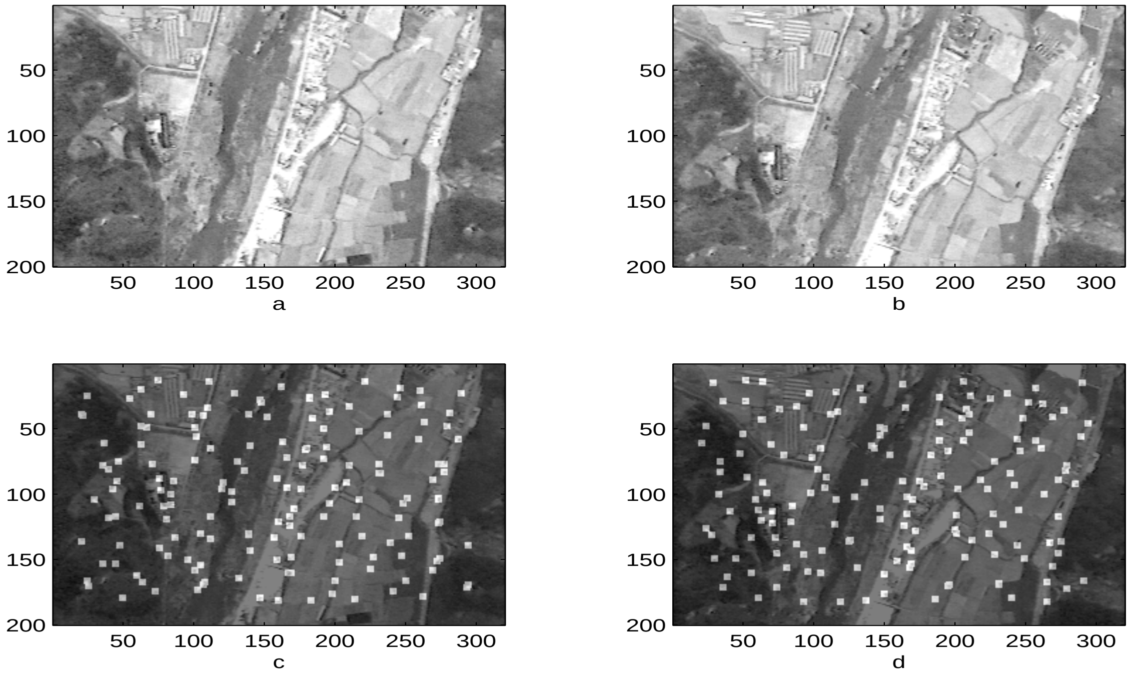

Fig. 3. Korean agricultural images (a and b), original tiepoints (c and d)

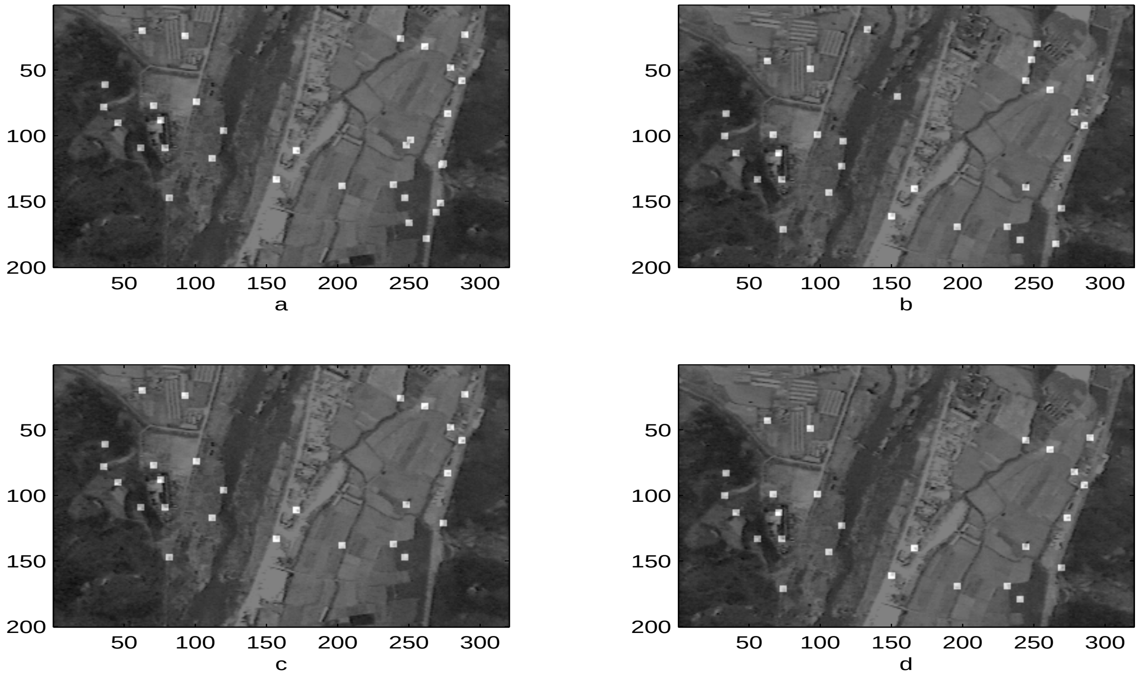

Fig. 4. Korean agricultural images: tiepoints have a 88% match rate after feature matching pass (a and b), tiepoints have 100% match rate after both feature and geometric matching (c and d).

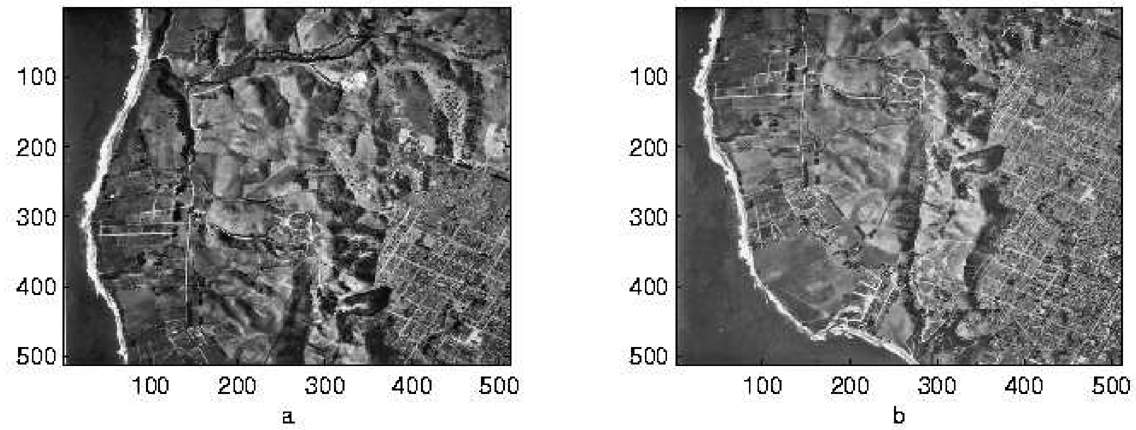

Fig. 5. Coastline images, tiepoints have a 100% match rate after both feature matching and geometric matching.

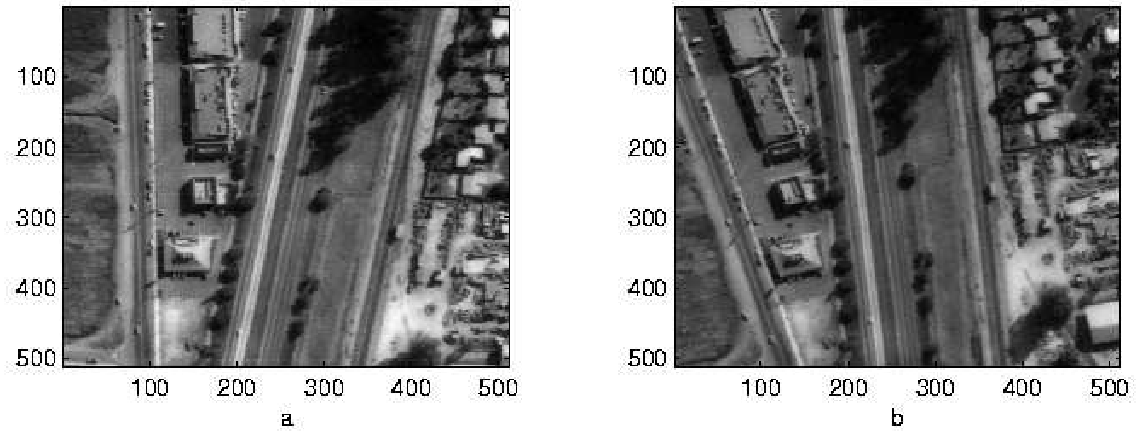

Fig. 6. Urban images, tiepoints have a 100% match rate after both feature matching and geometric matching.

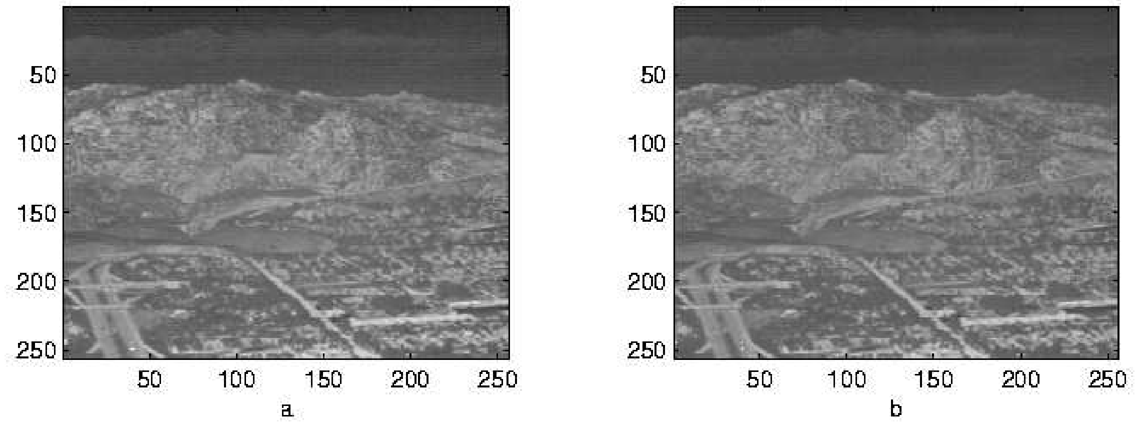

Fig. 7. Noisy IR urban images, tiepoints have a 100% match rate after both feature matching and geometric matching.

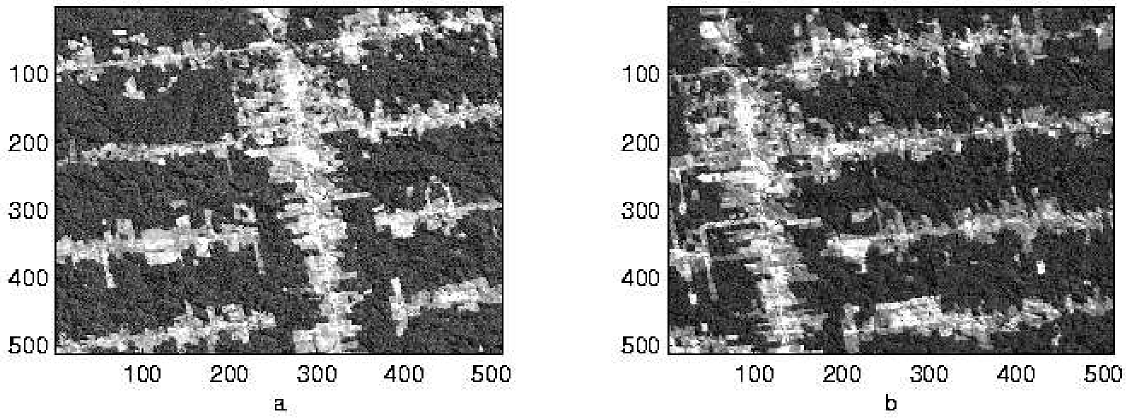

Fig. 8. Amazon forest images taken 2 years apart, tiepoints have a 100% match rate after both feature matching and geometric matching.

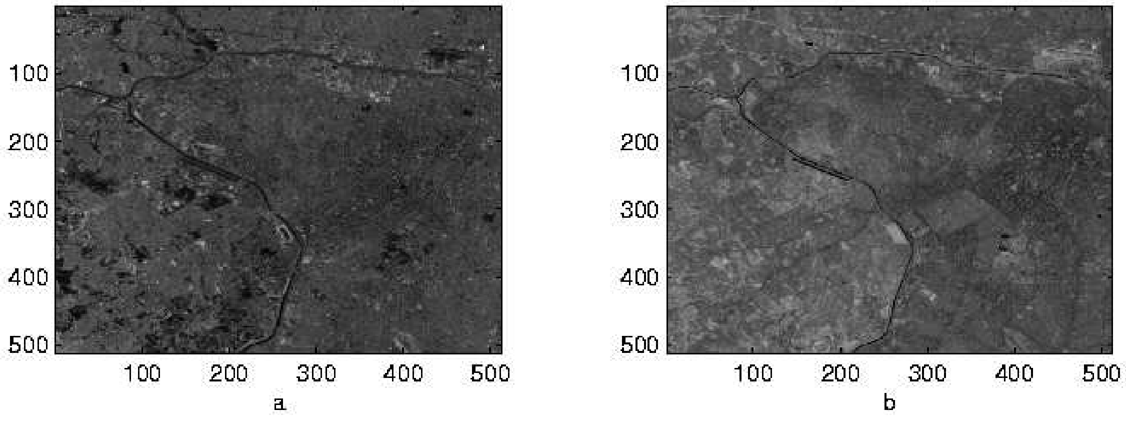

Fig. 9. Brazilian agricultural images taken 4 years apart, tiepoints have a 13% match rate after both feature matching and geometric matching.

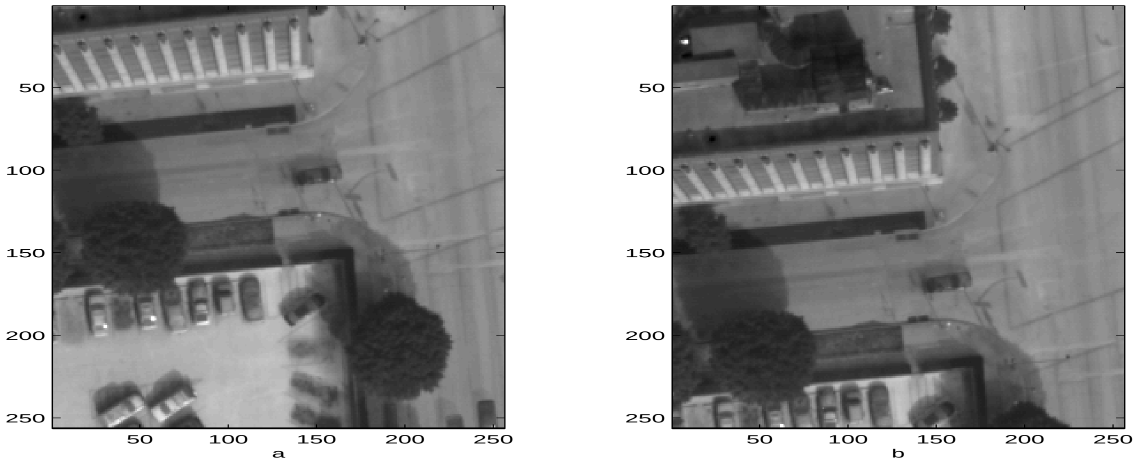

Fig. 10. Pair of images Showing Projective Geometry Effects

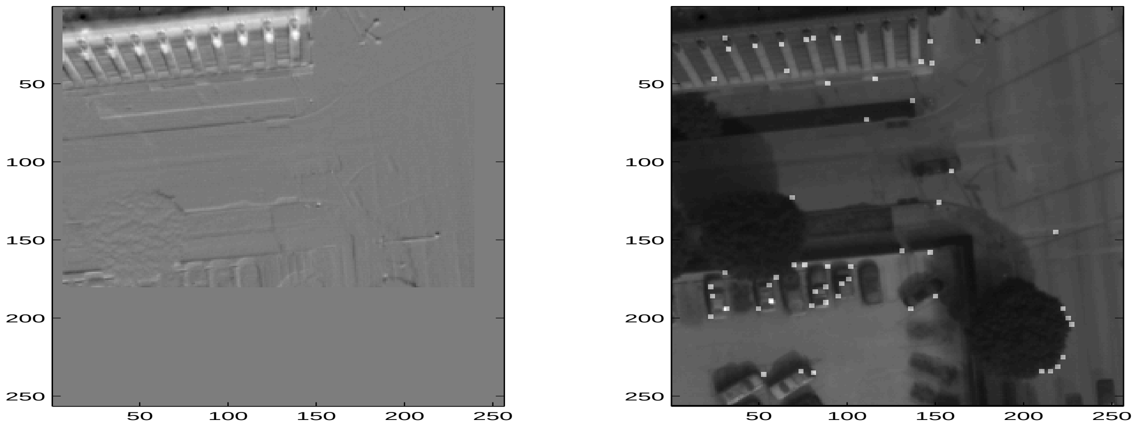

Fig. 11. Left: Difference Between Original and Stabilized Image; Right Points Selected for Post-Registration Error Analysis

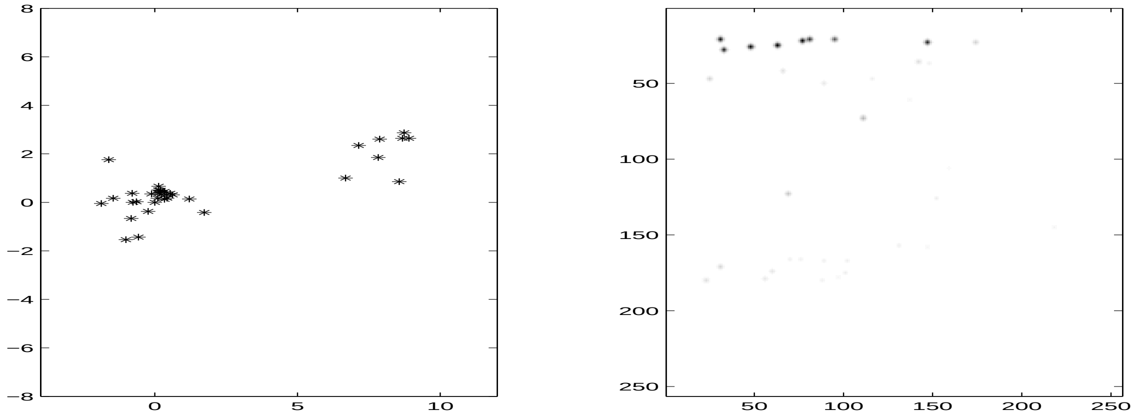

Fig. 12. Left: Post-Registration Errors Fall Into Two Clusters; Right: Points Correspond to Projective Geometry Errors Due to Using RST Registration Model.