

Robust Outliers Detection in Image Point Matching

Simon Beckouche
Caltech, Pasadena, USA
ENS Cachan, Paris, France

simon.beckouche@ens-cachan.fr

Sébastien Leprince, Neus Sabater, François Ayoub
Caltech, Pasadena, USA

leprincs@caltech.edu

Abstract

Classic tie-point detection algorithms such as the Scale Invariant Feature Transform (SIFT) show their limitations when the images contain drastic changes or repetitive patterns. This is especially evident when considering multi-temporal series of images for change detection. In order to overcome this limitation we propose a new algorithm, the Affine Parameters Estimation by Random Sampling (APERS), which detects the outliers in a given set of matched points. This is accomplished by estimating the global affine transform defined by the largest subset of points and by detecting the points which are not coherent (outliers) with the transform. Comparisons with state-of-the-art methods such as GroupSAC or ORSA demonstrate the higher performance of the proposed method. In particular, when the proportion of outliers varies between 60% and 90% APERS is able to reject all the outliers while the others fail. Examples with real images and a shaded Digital Elevation Model are provided.

1. Introduction

1.1. Motivations and context

Image registration has many applications such as stereovision [18], mosaicing [19], or change detection [8]. We address the question of tie-points matching for change detection in multi-temporal series of airborne and spaceborne images. Relative image content changes are typically due to landscape or environment evolutions from man-made changes, vegetation growth, clouds, shadows, landslides, glaciers flowing, earthquakes, etc. Relative changes due to variations in the viewing geometry are also common. Classical registration techniques use local analysis of the images, that is points in one image are associated to corresponding points in another image using information on their neighborhoods. Because of the significant changes that may occur in multi-temporal images, state-of-the-art feature matching algorithms may produce abundant mis-

matches. We propose a method to robustly detect outliers in a set of tie-points under an affine constraint.

In feature detection algorithms like the Harris detector [4], SIFT [11], ASIFT [24], or SURF [1], interest points are found using local extrema of the image's laplacian. Interest points of the two images are then matched if their local neighborhoods are similar enough [11]. Feature matching techniques have been improved with the use of robust distances like the earth mover's distance [15] or using an *a contrario* criterion to match the patches [16], but they still rely on a local description of the images.

Another way to find corresponding points between pairs of images is to use block matching via image cross-correlation, *e.g.*, [26], [22], [5] or [6]. A sliding window scans the whole image and for each point in one image, a corresponding point is searched in the other image within an *a priori* region.

Both feature and block based matching methods produce sets of tie-points between pairs of images. As those matches are based on local texture analysis only, they do not necessarily follow any global geometrical consistency. When images have sophisticated structures or repetitive patterns, local description is therefore not robust enough to avoid mismatches (outliers).

The RANDOM SAMPLING Consensus algorithm (RANSAC) [3], and its modern variants [13], [23], [2], [25], [7], [14] can be used to remove outliers from a set of matches. However, RANSAC and several variants do not handle a proportion of inlier inferior to 50%. Other versions of RANSAC require abstract or arbitrary parameters that are hard to determine to ensure optimal results. Different improvements have been made, such as the multi *a contrario* RANSAC [17] that proposed to iteratively use RANSAC to detect parts of the image that have sustained different geometrical transforms. The Optimized Random Sampling Algorithm (ORSA) [12] uses random sampling to estimate the projective transform that maps the largest set of tie-points and detects the matches that best fit with this transform. However, ORSA is not able to detect every outlier, and works with a fixed

number of iterations that does not adapt to the outlier proportion. Hence it does not quickly process tie-points sets that present a small proportion of outliers. It also has no criterion to reject the estimated transform if it poorly fits to the tie-points.

Other matching techniques such as [7] and [9] have been developed, but they apply to video tracking and these images typically do not present drastic changes between pairs.

1.2. Problem statement

We work with a set of tie-points (matches) that could be produced by any local algorithm such as SIFT, ASIFT, SURF, or block matching. Those matches may contain many outliers, especially in multi-temporal registration for change detection. In addition, the set of matches can be large, as the number of interest points in an image representing a complex scene can easily exceed several thousands.

In remote sensing imagery, images are usually first pre-processed, *i.e.*, rectified, in order to approximately correct for viewing geometry effects. The residual mapping between the rectified slave and master images is here approximated by an affine transform, which accounts for inaccurate knowledge of the viewing geometries.

Our problem is to detect, among N interest points $(p_k)_{k=1,\dots,N}$ in a slave image and their assumed corresponding matches $m(p_k)_{k=1,\dots,N}$ in a master image, the largest subset of points that follows the same affine model, and to propose the best affine transform that maps these pairs of points. It is equivalent to finding the matrix A representing the affine mapping between the two images and the set of inliers that maximizes

$$\sum_{k=1}^N \delta_k \quad \text{s.t.} \quad \max_{k=1,\dots,N} \delta_k \|m(p_k) - A(p_k)\|_2 \leq \varepsilon, \quad (1)$$

where $(\delta_k)_{k=1,\dots,N} \in \{0,1\}^N$ is the set of boolean variables “match k is an inlier” and ε , in pixels, limits the tolerated distance between the matches and the geometrical model.

1.3. APERS

In this paper, we present the Affine Parameters Estimation via Random Sampling (APERS) algorithm. It estimates the affine transform that fits the largest set of matches, and detects the matches that can be considered as inliers according to the estimated affine transform. It iteratively uses random sampling of matches to perform an independent clustering in the six affine coefficient dimensions. It also gives *a posteriori* uncertainties on the matches. It only requires a coarse geometric threshold ε as the maximum distance allowed between the matches and the geometrical model.

This article is organized as follows: **Section 2** focuses on our mathematical model and on the different steps of

APERS, and we give its pseudo-code in **Section 3**. We present the results of our experiments in **Section 4**, where we show that we can correctly estimate the mapping between the two images up to 90% of outliers, while we systematically filter every single outlier.

2. Mathematical Model

In our model, outliers are seen as uniform random matches. Inliers are defined as the largest set of matches that follow the same transform. The key heuristic is that all the inliers follow the same affine transform, while outliers do not have any overall behavior. Using the RANSAC idea of random sampling [3], we perform a statistical analysis of the samples to discard outliers.

Our algorithm is based on identifying the most frequent affine transform defined by all potential triplet of matches. However, an exhaustive estimation of all possible transforms is out of reach when presented with a large amount of matches [12]. For instance, 500 matches would yield more than 20×10^6 triplets and cannot be processed by a standard computer. One cannot simply use an arbitrary subset of matches neither because there is no *a priori* criterion to select that subset. Excluding a part of the matches could lead to increasing the proportion of outliers, and make the inliers impossible to identify. Our solution is to iteratively randomly regroup the matches into smaller subsets to exhaustively estimate the best affine transform related to the inliers of each subset. Then, all these estimated transforms are compared with each other to find the one that fits with most subsets. Notice that the subsets have to be small enough to be quickly processed, but large enough to keep the affine transform estimation accurate and robust. Although three points are enough to define an affine transform, the statistical analysis must rely on larger sets of points.

Given a subset of matches, every affine transform defined by a triplet in the subset is estimated. We identify in the space of the affine transform coefficients the cluster where the concentration of transforms is the highest. Indeed, we will show that any triplet that contains at least one outlier yields an affine transform whose coefficients are uniformly distributed over an interval, while triplets of inliers give coefficients that follow a Gaussian distribution. The outliers’ variance is much larger than the variance of the coefficients of the affine transform estimated from a triplet of inliers. For the sake of speed, we perform a unidimensional, separable clustering in every six dimensions [21] of the affine transform coefficient space. To estimate the probability density function (PDF) of the coefficients, we use a Gaussian kernel method as we have no *a priori* knowledge of the location of the cluster nor of its width. The mode of the PDF is considered to be the best estimation of the coefficient.

Once we have estimated the best affine transform for

each subset, we identify the most likely global transform. Again, we use a PDF estimation with a kernel method to perform that task because we do not know at what scale the cluster of the best affine transforms is distinguishable. We also evaluate the accuracy of the transform estimation, which makes us able to reject a transform if it is not precise enough. The expected proportion of inliers is not assumed known and is gradually decreased until enough inliers are found.

2.1. Random sampling of triplets and affine transform estimation

Generally, interest points in the image are not detected with high spatial accuracy. Furthermore, the affine geometrical deformation model may not be accurate. For these reasons our model accounts for additive white Gaussian noise. In this section the variance formulas of the affine coefficients are given.

2.1.1 Expected variance of affine parameters

Let (x^s, y^s) be an interest point in the slave image I_s and let (x^m, y^m) be the theoretical matched point in the master image I_m . Then the observed match $(\widehat{x}^m, \widehat{y}^m) = (x^m + \widetilde{x}^m, y^m + \widetilde{y}^m)$ contains the residual error $(\widetilde{x}^m, \widetilde{y}^m)$.

Let $(x_i^s, y_i^s) \in I_s$ and $(\widehat{x}_i^m, \widehat{y}_i^m) \in I_m$ with $i = 1, \dots, 3$ be three matches between the slave and the master images where the points in the slave image are not aligned. We denote

$$S = \begin{pmatrix} x_1^s & x_2^s & x_3^s \\ y_1^s & y_2^s & y_3^s \\ 1 & 1 & 1 \end{pmatrix} \quad \text{and} \quad M = \begin{pmatrix} x_1^m & x_2^m & x_3^m \\ y_1^m & y_2^m & y_3^m \\ 1 & 1 & 1 \end{pmatrix} \quad (2)$$

Similarly, \widehat{M} and \widetilde{M} are defined with the triplets $(\widehat{x}_i^m, \widehat{y}_i^m)_i$ and $(\widetilde{x}_i^m, \widetilde{y}_i^m)_i$ respectively, but the last row of \widetilde{M} containing zeros. With this notation, the affine transform defined by this triplet, \widehat{A} , and the theoretical affine matrix A are the matrices such that

$$\widehat{A} \cdot S = \widehat{M} \quad \text{and} \quad A \cdot S = M. \quad (3)$$

Proposition 2.1. *With the notations above, if the residual errors $\widetilde{x}_i^m, \widetilde{y}_i^m$ are i.i.d. centered Gaussian variables, then the coefficients of the noise matrix*

$$\widetilde{A} = \widehat{A} - A := \begin{pmatrix} \widetilde{a} & \widetilde{c} & \widetilde{u} \\ \widetilde{b} & \widetilde{d} & \widetilde{v} \\ 0 & 0 & 0 \end{pmatrix} \quad (4)$$

are centered Gaussian variables and their expected variances are

$$\begin{aligned} \sigma_a^2 &= \frac{(y_1^s - y_2^s)^2 \cdot \sigma_{x_3^m}^2 + (y_2^s - y_3^s)^2 \cdot \sigma_{x_1^m}^2 + (y_3^s - y_1^s)^2 \cdot \sigma_{x_2^m}^2}{(\det(S))^2} \\ \sigma_c^2 &= \frac{(x_2^s - x_1^s)^2 \cdot \sigma_{x_3^m}^2 + (x_1^s - x_3^s)^2 \cdot \sigma_{x_2^m}^2 + (y_3^s - y_2^s)^2 \cdot \sigma_{x_1^m}^2}{(\det(S))^2} \\ \sigma_u^2 &= \frac{1}{(\det(S))^2} \cdot \left((x_1^s y_2^s - x_2^s y_1^s)^2 \cdot \sigma_{x_3^m}^2 \right. \\ &\quad \left. + (x_3^s y_1^s - x_1^s y_3^s)^2 \cdot \sigma_{x_2^m}^2 + (x_2^s y_3^s - x_3^s y_2^s)^2 \cdot \sigma_{x_1^m}^2 \right) \end{aligned} \quad (5)$$

and $\sigma_b^2, \sigma_d^2, \sigma_v^2$ are obtained with the same formulas than $\sigma_a^2, \sigma_c^2, \sigma_u^2$ respectively but using $\sigma_{y_i^m}^2$ instead of $\sigma_{x_i^m}^2$.

Proof. Since $\widetilde{A} = \widehat{M} \cdot S^{-1}$ we have

$$\widetilde{a} = \frac{(y_1^s - y_2^s) \cdot \widetilde{x}_3^m + (y_2^s - y_3^s) \cdot \widetilde{x}_1^m + (y_3^s - y_1^s) \cdot \widetilde{x}_2^m}{\det(S)}. \quad (6)$$

With the hypothesis that $\widetilde{x}_i^m \sim \mathcal{N}(0, \sigma_{x_i^m}^2)$, $i = 1, \dots, 3$ we have that $\widetilde{a} \sim \mathcal{N}(0, \sigma_a^2)$. By the same token, we obtain the result for the other coefficients. \square

Applying Eq. (5) to outliers, we show how the transform coefficients behave when not estimated from three inliers. We consider that outliers behave as if they had been matched randomly, thus the transform defined by a triplet composed by at least one outlier is also randomly defined.

Proposition 2.2. *Let A be an affine transform, $(x_i^s, y_i^s)_{i=1,2,3}$ be three points in I_s , and $(x_i^m, y_i^m)_{i=1,2,3}$ be their image by A . Let (\mathbf{X}, \mathbf{Y}) be a uniform random vector on I_m . Then each coefficient of the affine transform \widehat{A} related to the triplet $(\{(x_1^s, y_1^s), (\mathbf{X}, \mathbf{Y})\}, \{(x_2^s, y_2^s), (x_2^m, y_2^m)\}, \{(x_3^s, y_3^s), (x_3^m, y_3^m)\})$ is of the form:*

$$\widehat{a} = \mu_a + \lambda_a \mathbf{X} \quad (7)$$

and follows a uniform distribution where μ_a and λ_a are two deterministic constants.

Proof. Focusing on the coefficient \widehat{a} and ignoring the Gaussian noise, we have

$$\widehat{a} = \frac{1}{\det(S)} \cdot \left((y_1^s - y_2^s)x_3^m + (y_2^s - y_3^s)\mathbf{X} + (y_3^s - y_1^s)x_2^m \right) \quad (8)$$

Let

$$\begin{aligned} \mu_a &= a - \frac{1}{\det(S)} \cdot (y_2^s - y_3^s)x_1^m \quad \text{and} \\ \lambda_a &= \frac{1}{\det(S)} \cdot (y_2^s - y_3^s), \end{aligned} \quad (9)$$

where a is the first coefficient of the matrix A . It then yields Eq. (7). \square

This result remains true when more than one point are replaced by uniform random variables. The proof is similar, and for example if the second master point is replaced with a uniform variable, we have:

$$\begin{aligned}\mu_a &= a - \frac{1}{\det(S)} \cdot \left((y_2^s - y_3^s)x_1^m + (y_3^s - y_1^s)x_2^m \right) \\ \lambda_a &= \frac{1}{\det(S)} \cdot \sqrt{(y_2^s - y_3^s)^2 + (y_3^s - y_1^s)^2}.\end{aligned}\quad (10)$$

In practice, the scalar λ_a is large enough for the coefficients estimated from triplets containing outliers to be largely spread compared to the coefficients correctly approximated. For $I_s = [1\dots 1024]^2$ and $(x_i^s, y_i^s)_{i=1,2,3}$ uniformly generated on I_s , λ_a is on the order of 0.1. Thus the magnitude of the product $\lambda_a \widehat{x}_1^m$ varies between 0.1 and 100, when the standard deviation of the coefficients estimated from a triplet of inliers is usually below 0.01.

2.1.2 Random sampling of triplets: a group-experiment

During the random sampling, we take m matches uniformly drawn among all the matches. We compute the $\binom{m}{3}$ affine transforms related to the possible $\binom{m}{3}$ triplets from the sample via SVD. We use SVD to avoid ill-conditioned affine transforms matrices encountered for instance when the matches that define it are too similar. We also compute the six standard deviations associated to the six coefficients of each affine transform using Eq. (5), with the assumption that

$$(\sigma_{x_i^m})_{i=1,2,3} = (\sigma_{y_i^m})_{i=1,2,3} = 1 \text{ pixel}, \quad (11)$$

which is a good estimate for the precision of typical interest point detection methods. We call this step a group-experiment. It yields $\binom{m}{3}$ sets of six coefficients and standard deviations.

After the group-experiment, we have a list $(\widehat{A}_k)_{k=1,\dots,T}$ with $T = \binom{m}{3}$, and for each coefficient of each \widehat{A}_k , we have computed an estimation of its variance with Eq. (5). For the six coefficients, we estimate independently their empirical probabilistic distribution using the kernel method detailed further to detect the most likely value of each coefficient.

2.2. Meaningful cluster of affine transforms

A good estimation of the coefficients' value requires to identify the cluster of coefficients defined by triplets of inliers. As we do not *a priori* know the width of the cluster, we estimate the coefficients PDF using a kernel method that automatically adapts its scale as a function of the standard deviation σ_k of every coefficient [20]. A parameter estimation with a small σ and with a lot of points close to it (w.r.t.

σ) will have a high "score" according to the kernel method, while isolated points will be neglected. We give the definition of our Gaussian kernel function:

Definition 2.2.1. Let T be a number of experiments, let $(t_k)_{k=1,\dots,T}$ be a set of values and let $(\sigma_k)_{k=1,\dots,T}$ be their error terms. For all k , we define:

$$\text{score}(t_k) = \sum_{j=1}^T \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{(t_k - t_j)^2}{2\sigma_j^2}}. \quad (12)$$

The *score* function computes a histogram of $(t_k)_{k=1,\dots,T}$ with an adaptive scale. It gives a smooth curve that can be seen as an empirical probability density function.

After computing $(\widehat{A}_k, \sigma_{\widehat{a}_k}, \sigma_{\widehat{b}_k}, \sigma_{\widehat{c}_k}, \sigma_{\widehat{d}_k}, \sigma_{\widehat{u}_k}, \sigma_{\widehat{v}_k})_{k=1,\dots,T}$, the kernel method is used to detect the corresponding cluster. For each coefficient, we perform several tests to detect if the experiment is meaningful and to decide if we can correctly estimate the coefficient value.

Because the procedure is identical for all coefficients, we here only focus on the coefficient a and its estimation \widehat{a} . We define \widehat{a} , the most likely value of a as

$$\widehat{a} = \arg \max_{\widehat{a}_k, k=1,\dots,T} \text{score}(\widehat{a}_k) \quad (13)$$

and we define $\sigma_{\widehat{a}}^{\text{temp}}$ as being the error term associated to \widehat{a} . We restrain our search to the discrete values $(\widehat{a}_k)_{k=1,\dots,T}$ to reduce computation time. Although the *score* function could be maximized on \mathbb{R} , we do not need an accurate estimation of its maximum, and finding it in \mathbb{R} would be much longer.

As the triplets made of inliers produce coefficients that follow a normal distribution, we consider that all correct coefficients, that is estimated from inlier triplets, should be located in $[\widehat{a} - 3\sigma_{\widehat{a}}^{\text{temp}}, \widehat{a} + 3\sigma_{\widehat{a}}^{\text{temp}}]$. We define the neighborhood of \widehat{a} as being:

$$E_{\widehat{a}} = \{(\widehat{a}_k)_{k=1,\dots,T} / |\widehat{a} - \widehat{a}_k| \leq 3\sigma_{\widehat{a}}^{\text{temp}}\}. \quad (14)$$

The set $E_{\widehat{a}}$ is seen as the cluster of correct coefficients.

Then, the normalized score function

$$\text{score}(E_{\widehat{a}}) = \frac{\sum_{\widehat{a}_k \in E_{\widehat{a}}} \text{score}(\widehat{a}_k)}{\sum_{k=1}^T \text{score}(\widehat{a}_k)} \quad (15)$$

indicates the relative weight of the detected cluster with respect to the entire distribution of coefficients.

An experiment is meaningful if its corresponding neighborhood satisfies:

$$\text{card}(E_{\widehat{a}}) \geq 4 \quad \text{and} \quad \text{score}(E_{\widehat{a}}) \geq \text{score}_{\text{thresh}} \quad (16)$$

where $\text{card}(\Omega)$ is the number of elements that contains the set Ω . Indeed, if a group-experiment is made with a set of

matches that contains 3 inliers or less, we cannot correctly estimate the value of the coefficient. If the set of matches used for the group-experiment contains at least 4 inliers, the associated set of triplets will contain at least $\binom{4}{3} = 3$ triplets made by inliers, thus there will be at least three coefficients really close from each other, thus located in the cluster we were looking for. $score_{thresh} = 0.15$ is a good trade off between speed and robustness. This ensures that the cluster determined by the kernel method is concentrated enough relatively to the standard deviation of its center value.

If the experiment is meaningful, one can then re-estimate the standard deviation of \hat{a} :

$$\sigma_{\hat{a}} = \frac{1}{card(E_{\hat{a}})} \cdot \sqrt{\sum_{\hat{a}_k \in E_{\hat{a}}} (\hat{a} - \hat{a}_k)^2}. \quad (17)$$

We now have \hat{a} , an estimated value of a , and its precision term $\sigma_{\hat{a}}$. If the set $(\hat{a}_k)_{k=1, \dots, T}$ meets the requirements in Eq. (16), we consider that \hat{a} provides a meaningful estimation of a , and we store its value and its estimated standard deviation $\sigma_{\hat{a}}$. We perform several group-experiments to obtain a set of meaningful parameters.

2.3. Outliers filtering

We explain how we compute an affine transform with the stored parameters, and how outliers are rejected.

2.3.1 Global affine transform estimation

The results of a group-experiment are not directly used to detect outliers as we need a robust affine estimation. Indeed, some experiments could verify Eq. (16) without being a good approximation of the global affine mapping. This happens when a set of m matches contains too many outliers. If the number of inliers is low, outliers may unfortunately form a cluster. Therefore, the PDF estimation method is used again to compute a robust value for each coefficient, using all the stored meaningful values and their standard deviation. As the kernel method needs at least three or four points to work correctly, we only apply it every 10 group-experiments. The kernel method gives the six final coefficients $a_f, b_f, c_f, d_f, u_f, v_f$ of A_f , and their six related standard deviations $\sigma_{a_f}, \sigma_{b_f}, \sigma_{c_f}, \sigma_{d_f}, \sigma_{u_f}, \sigma_{v_f}$.

2.3.2 Outliers detection

We now have to check if A_f fits well with a significant proportion of matches $p_{min} \in [0, 1]$. This ‘‘significant proportion’’ evolves during the execution of APERS and is discussed later.

Definition 2.3.1. Let $\{(x^s, y^s), (\hat{x}^m, \hat{y}^m)\}$ be a pair of points. The geometrical deviation σ_M associated to this

pair is defined as the norm of the standard deviation of $\tilde{A}_f \cdot (x^s, y^s, 1)^t$, where \tilde{A} is defined by Eq. (4):

$$\sigma_M = \left((\sigma_{a_f}^2 + \sigma_{b_f}^2)x^{s2} + (\sigma_{c_f}^2 + \sigma_{d_f}^2)y^{s2} + \sigma_{u_f}^2 + \sigma_{v_f}^2 \right)^{1/2}. \quad (18)$$

As we modeled the inaccuracy of the inliers with a Gaussian noise, the probability that a point (\hat{x}^m, \hat{y}^m) in I_m and the point predicted by the geometric model $A_f \cdot (x^s, y^s, 1)^t$ are separated by a distance smaller than $3\sigma_M$ is above 99% if (\hat{x}^m, \hat{y}^m) is an inlier.

Definition 2.3.2. Let $\{(x^s, y^s), (x^m, y^m)\}$ be a pair of points. It is considered an inlier if

$$\|A_f \cdot (x^s, y^s, 1)^t - (x^m, y^m, 1)^t\|_2 \leq 3\sigma_M. \quad (19)$$

We check if we have found enough inliers, that is

$$\frac{card\{inliers\}}{card\{points\}} \geq p_{min}, \quad (20)$$

where p_{min} is the expected minimum proportion of inliers. We assume here that outliers have no affine consistency, which is coherent with our initial hypothesis. If A_f has been too inaccurately estimated, our model could accept all the matches as inliers, allowing them to be far from the geometrical model. We accept A_f and the set of inliers if the largest deviation among all inliers is below ε .

$$\max_{k \in \{inliers\}} \{\sigma_M^k\} \leq \varepsilon. \quad (21)$$

The ε parameter can be adapted according to the desired accuracy. We set its value to 5% of the image dimension, which was found appropriate in our practical examples.

If Eq. (20) and (21) are verified, the algorithm stops and outputs $A_f, \sigma_{a_f}, \sigma_{b_f}, \sigma_{c_f}, \sigma_{d_f}, \sigma_{u_f}, \sigma_{v_f}$ and the set of inliers. Otherwise, the parameter p_{min} is decreased and we proceed as described in the next section.

2.4. APERS convergence

The last step consists in automatically updating the parameters so that APERS converges. We begin with an expected inlier proportion $p_{min} = 90\%$. The algorithm proceeds with one series of 10 group-experiments. If the result does not satisfy Eq. (20) and Eq. (21), another series of 10 group-experiments is processed, and its results are added to the stored parameters of the first series. The algorithm performs up to 10 series of 10 group-experiments. If APERS does not find an inlier set containing 90% of the matches, we delete the stored values and reduce p_{min} .

The reason we delete stored values is first because the PDF estimation is not tractable if applied to a very large

set of points. Second, if the outlier proportion is large, keeping all stored values would lead to an incorrect inlier detection. Indeed, as all triplets would be sampled several times, many irrelevant clusters would appear and could be accepted. p_{min} is reduced until APERS converges or until it reaches 5%, where APERS enters a slow search regime. In this slow search regime, APERS stops storing values as it looks for a very small set of inliers. We perform 500 independent series of 10 group-experiments. If Eq. (20) is verified during the slow search, APERS stops and outputs the set of matches it detected as inliers. If the algorithm does not converge, the computation is stopped, and all matches are considered as outliers.

3. APERS algorithm

APERS pseudo-code is presented in Algo. 1. We use the same parameters for every pair of images, that is: $m = 10$, the number of matches that composes a group-experiment. $score_{thresh} = 0.15$, seen in Eq. (16). $\varepsilon = 5\%$ of the image size, which controls the distance between the geometrical model and the matches (all our images were downsampled to be approximately a thousand pixel wide). Parameters m and $score_{thresh}$ have been experimentally determined, being good trade-offs between robustness and speed.

Algorithm 1: APERS pseudo-code.

Data: A set of potential matches
 $\{(x_k^s, y_k^s), (x_k^m, y_k^m)\}_{k=1, \dots, N}$, a precision parameter ε , in pixels

Result: Set of inliers that fit with the estimated transform with a precision better than ε

for
 $p_{min} = [0.9, 0.8, 0.7, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1, 0.05]$
do

while (Eq. (19) or Eq. (21) is not satisfied) and the number of stored parameters is not too large **do**

Perform a group-experiment and store the result if meaningful;

if number of loop is a multiple of 10 **then**

Use the kernel method to estimate the transforms coefficients and compute all geometrical thresholds using Eq. (18);

Identify inliers and stop if the number of inliers exceeds $\lfloor p_{min} N \rfloor$;

end

if Number of stored values is too high **then**

Delete stored values and reduce p_{min} ;

end

end

4. Experiments

4.1. Theoretical experiments

To test APERS with different amounts of outliers we simulate several sets of matches. Every set has 512 matches where the proportion of outliers varies between 0% and 100%. The outliers proportion in our experiments is denoted by p_{out} . The matches are generated in a 1024×1024 pixel square and we generate an affine transform, with the a, b, c, d coefficients uniformly distributed in $[-20, 20]$ and the translation coefficients u and v are uniform in $[-100, 100]$. We compute the images of the slave points by the affine transform, and randomly reassign a proportion p_{out} uniformly inside the smallest rectangle that contains all the original master points. Finally, we add a white Gaussian noise with variance of 1 pixel on every match. Fig. 1 and Fig. 2 show the result of this synthetic benchmark. To compare APERS to other algorithms, we implemented RANSAC and ORSA for affine transform estimation. We also used the implementation of GroupSAC proposed in [13], as well as a recent implementation of RANSAC labeled HomogSAC on the figures, for finding a homography among pairs of points [23]. The homographic model applies to our case, although it is slightly less accurate than the affine model.

APERS correctly identifies a significant proportion of inliers up to 90% of outliers, while it is always capable of filtering every single outliers. For $p_{out} \leq 80\%$, all the inliers and outliers are correctly identified, and the proportion of identified inliers remains very high even for $p_{out} = 90\%$. The computation time increases approximately linearly until $p_{out} = 80\%$, because APERS always converges before entering the slow search regime. As ORSA and RANSAC algorithms have no stop criterion, we adapt the number of iterations so their computation time is the same as APERS i.e., 0.1, 0.1, 10, 25, 45, 55, 65, 80, 100, 160 and 220 seconds for the eleven values of p_{out} . The default values for the number of iterations proposed with the RANSAC variants were much smaller than the value we used, which made the algorithms find many more inliers. All implementations are done in Matlab and using the same computer. RANSAC and GroupSAC reject almost every outlier in each experiment, but they discard almost all inliers when p_{out} exceeds 50%. The rejection threshold being underestimated, a good estimation of the affine transform is performed at the cost of many inliers incorrectly identified. ORSA detects inliers as well as APERS, but starts to accept outliers for $p_{out} \geq 60\%$. The HomogSAC algorithm filters every outliers, but always misses a portion of the inliers because of its sensitivity to noise. The authors adjusted the parameters of the different algorithms to maximize the proportion of detected inliers without accepting any outliers, whenever possible.

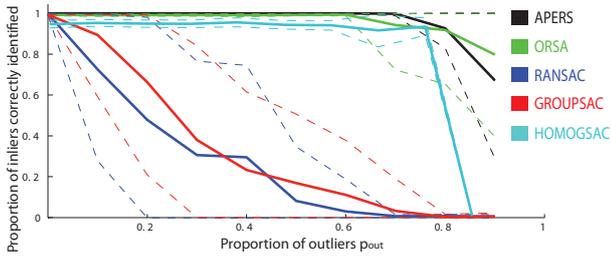


Figure 1. Theoretical experiments with APERS compared to RANSAC and ORSA showing detected inliers as a function of outlier proportion. Every 10% of p_{out} , we generate 40 similar experiments. The solid curves represent the proportion of inlier that were identified. The dashed curves represent the standard deviation of different series. We used a Matlab implementation on a PC with an Intel i7 2.8 Ghz processor.

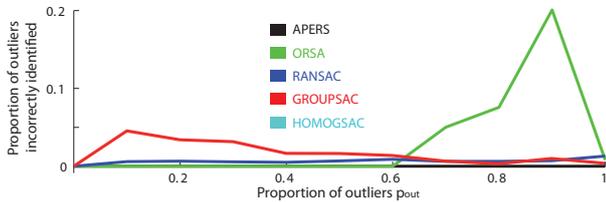


Figure 2. Theoretical experiments with APERS compared to RANSAC and ORSA showing detected outliers as a function of outlier proportion.

4.2. Application to remote sensing imagery

We present the results of APERS applied to different pairs of images. We use the SIFT algorithm to generate potential matches [11]. Inliers are displayed in green, outliers in red. We essentially use airborne and spaceborne optical images, but also a shaded Digital Elevation Model (DEM). The images are down-sampled with a Lanczos filter to be processed on a standard computer, to a resolution of few mega pixels. Figure 3 is an example where a part of the image has been masked to avoid tie-points in a specified area of the image (the assumed location of a seismic fault zone). Fig. 4 is an example of a difficult case with a small proportion of inliers, and Fig. 5 is an example with a mix of a shaded Lidar DEM as slave image, and an aerial image for the master.

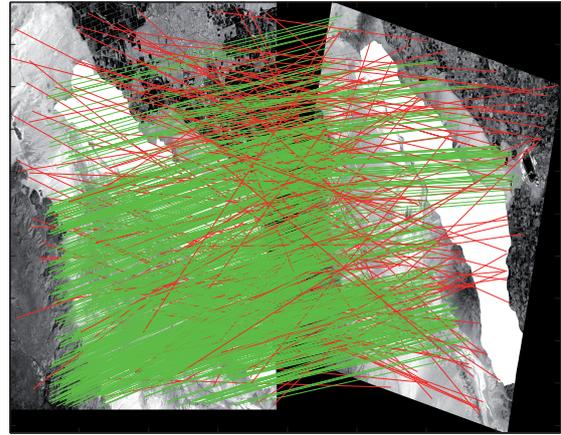


Figure 3. Pair of SPOT 5 satellite images above Mexicali, Mexico. 926 matches, 163 outliers (18%) and 763 inliers (82%). A part of the image, in white, has been masked to avoid tie-points in a specified area.

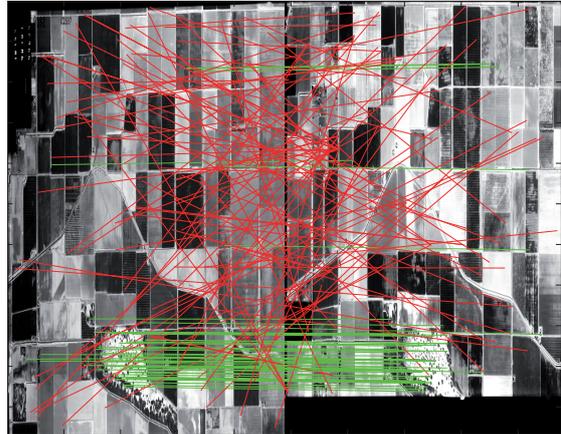


Figure 4. Pair of NAPP USGS aerial images from Imperial Valley, CA, USA. 155 matches, 100 outliers (65%) and 55 inliers (35%)

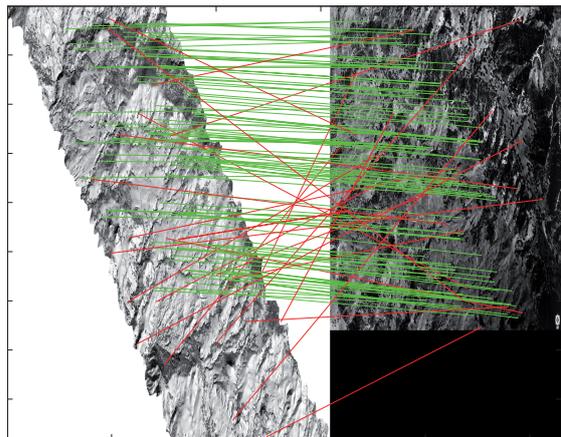


Figure 5. Pair of images from Central California along the San Andreas fault, USA. The master is a Lidar shaded DEM, compared with an aerial acquisition. 130 matches, 24 outliers (18%) and 106 inliers (82%)

5. Conclusion

We developed an algorithm, APERS, that separates inliers and outliers among a set of matches between two images mapped by an affine transform. The best matching affine transform is estimated, along with uncertainty on its parameters. The algorithm is fast for sets of matches with high proportion of inliers, detects inliers even when the proportion of outliers reaches 90%, and systematically identifies all the outliers, even when there are no inliers. Moreover, the algorithm does not need any manual intervention, as all the parameters have default values that make possible to process all kinds of images automatically. Nevertheless, several improvements could be done. The affine model could be upgraded to a projective model, with two additional coefficients, which would require to use another method to estimate the transform, such as the eight-point algorithm [10]. The computation time should stay reasonable, as it depends linearly on the number of coefficients to be estimated because all dimensions are processed separately. One could also optimize the evolution of parameter p_{min} and thus reduce the computation time or increase the proportion of inliers correctly identified.

In multi-temporal and multi-sensor image registration, current point matching algorithms are often subject to wrong pairing and therefore require to be supervised. By allowing for a robust and automatic extraction of correct tie-points, APERS could provide the missing link for complete automation of multi-temporal and multi-sensor image registration.

Acknowledgments We thank the Keck Institute for Space Studies (KISS) foundation for their support, as well as Jean-Philippe Avouac for hosting this research, and Sylvain Barbot for his stimulating discussions.

References

- [1] H. Bay, T. Tuytelaars, and L. Van Gool. Surf: Speeded up robust features. In *ECCV*. Springer Berlin / Heidelberg, 2006. 1
- [2] O. Chum and J. Matas. Optimal randomized ransac. *PAMI*, 30:1472–1482, 2008. 1
- [3] M. A. Fischler and R. C. Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24:381–395, 1981. 1, 2
- [4] C. Harris and M. Stephens. A combined corner and edge detector. In *Pro of the Alvey Vision Conference*, 1988. 1
- [5] W. S. Hoge. Subspace identification extension to the phase correlation method. *TMI*, 22:277–280, 2003. 1
- [6] W. S. Hoge, D. Mitsouras, F. J. Rybicki, R. V. Mulkern, and C. Fredrik Westin. Registration of multi-dimensional image data via sub-pixel resolution phase correlation. In *P IEEE ICIP*, 2003. 1
- [7] H. Jiang and S. Yu. Linear solution to scale and rotation invariant object matching. In *CVPR*, 2009. 1, 2
- [8] S. Leprince, S. Barbot, F. Ayoub, and J.-P. Avouac. Automatic and precise orthorectification, coregistration, and sub-pixel correlation of satellite images, application to ground deformation measurements. *GRSS*, 45:1529–1558, 2007. 1
- [9] H. Li, E. Kim, X. Huang, and L. He. Object matching with a locally affine-invariant constraint. In *CVPR*, 2010. 2
- [10] H. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, 1981. 8
- [11] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60:91–110, 2004. 1, 7
- [12] L. Moisan and B. Stival. A probabilistic criterion to detect rigid point matches between two images and estimate the fundamental matrix. *IJCV*, 57:201–218, 2004. 1, 2
- [13] K. Ni, J. H., and F. Dellaert. Groupsac: Efficient consensus in the presence of groupings. In *ICCV*, 2009. 1, 6
- [14] D. Nister. Preemptive ransac for live structure and motion estimation. *MVA*, 16:321–329, 2005. 1
- [15] J. Rabin, J. Delon, and Y. Gousseau. Circular earth mover’s distance for the comparison of local features. In *ICPR*, 2008. 1
- [16] J. Rabin, J. Delon, and Y. Gousseau. A contrario matching of sift-like descriptors. In *ICPR*, 2008. 1
- [17] J. Rabin, J. Delon, Y. Gousseau, and L. Moisan. Mac-ransac: a robust algorithm for the recognition of multiple objects. In *3D PVT*, 2010. 1
- [18] N. Sabater, J.-M. Morel, and A. Almansa. How accurate can block matches be in stereovision? *SIAM Journal on Imaging Sciences*, 4:472–500, 2011. 1
- [19] H. Sawhney and R. Kumar. True multi-image alignment and its application to mosaicing and lens distortion correction. *PAMI*, 21:235–243, 1999. 1
- [20] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004. 4
- [21] M. Steinbach, L. Ertöz, and V. Kumar. The challenges of clustering high-dimensional data. In *New Vistas in Statistical Physics: Applications in Econophysics, Bioinformatics, and Pattern Recognition*, 2003. 2
- [22] H. S. Stone, B. Tao, and M. McGuire. Analysis of image registration noise due to rotationally dependent aliasing. *JVCIR*, 14:114–135, 2003. 1
- [23] E. Wiggin. mathworks.com/matlabcentral/fileexchange/30809, 2011). 1, 6
- [24] G. Yu and J.-M. Morel. A fully affine invariant image comparison method. In *ICASSP*, 2009. 1
- [25] J. Yu, A. Eriksson, T.-J. Chin, and D. Suter. An adversarial optimization approach to efficient outlier removal. In *ICCV*, 2011. 1
- [26] B. Zitová and J. Flusser. Image registration methods: a survey. *IVC*, 21:977–1000, 2003. 1