Higher Dimensional Affine Registration and Vision Applications

Yu-Tseh Chi[1], S.M. Nejhum Shahed[1], Jeffrey Ho[1], and Ming-Hsuan Yang[2]

[1] CISE Department, University of Florida, Gainesville, 32607
{ychi,smshahed,jho}@csie.ufl.edu
[2] EECS, University of California, Merced, CA 95344
mhyang@ucmerced.edu

Abstract. Affine registration has a long and venerable history in computer vision literature, and extensive work have been done for affine registrations in $\mathbb{R}^2$ and $\mathbb{R}^3$. In this paper, we study affine registrations in $\mathbb{R}^m$ for $m > 3$, and to justify breaking this dimension barrier, we show two interesting types of matching problems that can be formulated and solved as affine registration problems in dimensions higher than three: stereo correspondence under motion and image set matching. More specifically, for an object undergoing non-rigid motion that can be linearly modelled using a small number of shape basis vectors, the stereo correspondence problem can be solved by affine registering points in $\mathbb{R}^{3n}$. And given two collections of images related by an unknown linear transformation of the image space, the correspondences between images in the two collections can be recovered by solving an affine registration problem in $\mathbb{R}^m$, where m is the dimension of a PCA subspace. The algorithm proposed in this paper estimates the affine transformation between two point sets in $\mathbb{R}^m$. It does not require continuous optimization, and our analysis shows that, in the absence of data noise, the algorithm will recover the exact affine transformation for almost all point sets with the worst-case time complexity of $O(mk^2)$, k the size of the point set. We validate the proposed algorithm on a variety of synthetic point sets in different dimensions with varying degrees of deformation and noise, and we also show experimentally that the two types of matching problems can indeed be solved satisfactorily using the proposed affine registration algorithm.

1 Introduction

Matching points, particularly in low-dimensional settings such as 2D and 3D, has been a classical problem in computer vision. The problem can be formulated in a variety of ways depending on the allowable and desired deformations. For instance, the orthogonal and affine cases have been studied already awhile ago, e.g., [1][2], and recent research activities have been focused on non-rigid deformations, particularly those that can be locally modelled by a family of well-known basis functions such as splines, e.g., [3]. In this paper, we study the more classical problem of matching point sets[1] related by affine transformations. The novel viewpoint taken here is the emphasis on affine registrations

[1] In this paper, the two point sets are assumed to have the same size.

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part IV, LNCS 5305, pp. 256–269, 2008.

in $\mathbb{R}^m$ for $m > 3$, and it differs substantially from the past literature on this subject, which has been overwhelmingly devoted to registration problems in $\mathbb{R}^2$ and $\mathbb{R}^3$.

To justify breaking this dimension barrier, we will demonstrate that two important and interesting types of matching problems can be formulated and solved as affine registration problems in $\mathbb{R}^m$ with $m > 3$: stereo correspondence under motion and image set matching (See Figure 1). In the stereo correspondence problem, two video cameras are observing an object undergoing some motion (rigid or non-rigid), and a set of k points on the object are tracked consistently in each view. The problem is to match the tracking results across two views so that the k feature points can be located and identified correctly. In the image set matching problem, two collections of images are given such that the unknown transformation between corresponding pairs of images can be approximated by some linear transformation $\mathcal{F} : \mathbb{R}^m \rightarrow \mathbb{R}^{m'}$ between two (high-dimensional) image spaces. The task is to compute the correspondences directly from the images. Both problems admit quick solutions. For example, for stereo correspondence under motion, one quick solution would be to select a pair of corresponding frames and compute the correspondences directly between these two frames. This approach is clearly since there is no way to know *a priori* which pair of frames is optimal for computing the correspondences. Furthermore, if the baseline between cameras is large, direct stereo matching using image features does not always produce good results, even when very precise tracking result are available. Therefore, there is a need for a principled algorithm that can compute the correspondences directly using all the tracking results simultaneously instead of just a pair of frames.

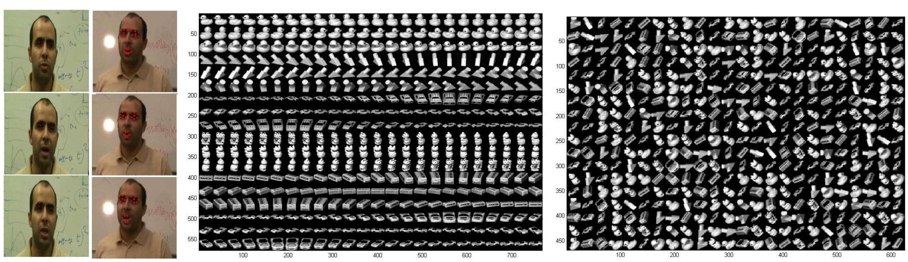

Fig. 1. Left: Stereo Correspondence under Motion. A talking head is observed by two (affine) cameras. Feature points are tracked separately on each camera and the problem is to compute the correspondences between observed feature points across views. **Center and Right:** Image Set Matching. Two collections (432 images each) of images are given. Each image on the right is obtained by rotating and down-sizing an image on the left. The problem is to recover the correspondences. These two problems can be formulated as affine registration problems in $\mathbb{R}^m$ with $m > 3$.

An important point to realize is that in each problem there are two linear subspaces that parameterize the input data. For nonrigid motions that can be modelled using linear shape basis vectors, this follows immediately from the work of [4][5]. For image set matching, each set of images can usually be approximated by a linear subspace with dimension that is considerably smaller than that of the ambient image space. We will show that the correspondences can be computed (or be approximated) by affine registering point sets in these two linear subspaces. Therefore, instead of using quantities derived from image intensities, our solution to these two matching problems is to first formulate them as affine point set matching problems in $\mathbb{R}^m$, with $m > 3$, and solve the resulting affine registration problems.

Let $\mathcal{P} = \{p_1, \cdots, p_k\}$ and $\mathcal{Q} = \{q_1, \cdots, q_k\}$ denote two point sets in $\mathbb{R}^m$ with equal number of points. The affine registration problem is typically formulated as an optimization problem of finding an affine transformation $\mathbf{A}$ and a correspondence map π between points in $\mathcal{P}, \mathcal{Q}$ such that the following registration error function is minimized

$$\mathcal{E}(\mathbf{A}, \pi) = \sum_{i=1}^{k} \mathbf{d}^2(\mathbf{A}p_i, q_{\pi(i)}), \tag{1}$$

where $\mathbf{d}(\mathbf{A}p_i, q_{\pi(i)})$ denotes the usual L^2-distance between $\mathbf{A}p_i$ and $q_{\pi(i)}$. The venerable iterative closest point (ICP) algorithm [6][7] can be easily generalized to handle high-dimensional point sets, and it gives an algorithm that iteratively solves for correspondences and affine transformation. However, the main challenge is to produce good initial correspondences and affine transformation that will guarantee the algorithm's convergence and the quality of the solution. For dimensions two and three, this is already a major problem and the difficulty increases exponentially with dimension. In this paper, we propose an algorithm that can estimate the affine transformation (and hence the correspondences π) directly from the point sets $\mathcal{P}$, $\mathcal{Q}$. The algorithm is algebraic in nature and does not require any optimization, which is its main strength. Furthermore, it allows for a very precise analysis showing that for generic point sets and in the absence of noise, it will recover the exact affine transformation and the correspondences. For noisy data, the algorithm's output can serve as a good initialization for the affine-ICP algorithm. While the algorithm is indeed quite straightforward, it is to the best of our knowledge that there has not been published algorithm which is similar to ours in its entirety. In this paper, we will provide experimental results that validate the proposed affine registration algorithm and show that both the stereo correspondence problem under motion and image set matching problem can be solved quite satisfactorily using the proposed affine registration algorithm.

2 Affine Registrations and Vision Applications

In this section, we provide the details for formulating the stereo correspondence under motion and image set matching problems as affine registration problems.

2.1 Stereo Correspondences under Motion

For clarity of presentation, we will first work out the simpler case of rigid motions. We assume two stationary affine cameras C_1, C_2 observing an object $\mathcal{O}$ undergoing some (rigid or nonrigid) motion. On each camera, we assume that some robust tracking algorithm is running so that a set $\{X_1, \cdots, X_k\}$ of k points on $\mathcal{O}$ are tracked over T frames separately on both cameras. Let (x_{ij}^t, y_{ij}^t) $1 \le i \le 2, 1 \le j \le k, 1 \le t \le T$ denote the image coordinates of $X_j \in \mathcal{O}$ in the $t^{\mathbf{th}}$ frame from camera i. For each camera, the tracker provides the correspondences $(x_{ij}^t, y_{ij}^t) \leftrightarrow (x_{ij}^{t'}, y_{ij}^t)$ across different frames t and t'. Our problem is to compute correspondences across two views so that the corresponding points $(x_{1j}^t, y_{1j}^t) \leftrightarrow (x_{2j}^t, y_{2j}^t)$ are the projections of the scene point X_j in the images. We show next that it is possible to compute the correspondences directly using only the high-dimensional geometry of the point sets $(\, x_{ij}^t,\, y_{ij}^t)$ without referencing to image features such as intensities.

For each view, we can stack the image coordinates of one tracked point over T frames vertically into a $2T$-dimensional vector:

$$\mathbf{p}_j = (\, x_{1j}^1 \; y_{1j}^1 \; \cdots, x_{1j}^T \; y_{1j}^T \,)^t, \;\; \mathbf{q}_j = (\, x_{2j}^1 \; y_{2j}^1 \; \cdots, x_{2j}^T \; y_{2j}^T \,)^t \tag{2}$$

In motion segmentation (e.g., [8]), the main objects of interest are the 4-dimensional subspaces L_p, L_q spanned by these $2T$-dimensional vectors

$$\mathcal{P} = \{\mathbf{p}_1, \cdots, \mathbf{p}_k\}, \qquad \mathcal{Q} = \{\mathbf{q}_1, \cdots, \mathbf{q}_k\},$$

and the goal is to cluster motions by determining the subspaces L_p, L_q given the set of vectors $\mathcal{P} \cup \mathcal{Q}$. Our problem, on the hand, is to determine the correspondences between points in $\mathcal{P}$ and $\mathcal{Q}$. It is straightforward to show that there exists an affine transformation $\mathbf{L} : L_p \rightarrow L_q$ that produces the correct correspondences, i.e., $\mathbf{L}(\mathbf{p}_i) = \mathbf{q}_i$ for all i. To see this, we fix an arbitrary world frame with respect to which we can write down the camera matrices for C_1 and C_2. In addition, we also fix an object coordinates system with orthonormal basis $\{\mathbf{i}, \mathbf{j}, \mathbf{k}\}$ centered at some point $o \in \mathcal{O}$. Since $\mathcal{O}$ is undergoing a rigid motion, we denote by $o_t, \mathbf{i}_t, \mathbf{j}_t, \mathbf{k}_t$, the world coordinates of $o, \mathbf{i}, \mathbf{j}, \mathbf{k}$ at frame t. The point X_j, at frame t, with respect to the fixed world frame is given by

$$X_j^t = o_t + \alpha_j \mathbf{i}_t + \beta_j \mathbf{j}_t + \gamma_j \mathbf{k}_t, \tag{3}$$

for some real coefficients $\alpha_j, \beta_j, \gamma_j$ that are independent of time t. The corresponding image point is then given as

$$(\, x_{ij}^t, y_{ij}^t \,)^t = \tilde{o}_{it} + \alpha_j \tilde{\mathbf{i}}_{it} + \beta_j \tilde{\mathbf{j}}_{it} + \beta_j \tilde{\mathbf{k}}_{it},$$

where $\tilde{o}_{it}, \tilde{\mathbf{i}}_{it}, \tilde{\mathbf{j}}_{it}, \tilde{\mathbf{k}}_{it}$ are the projections of the vectors $o_t, \mathbf{i}_t, \mathbf{j}_t, \mathbf{k}_t$ onto camera i. In particular, if we define the $2T$-dimensional vectors $\mathbf{O}_i, \mathbf{I}_i, \mathbf{J}_i, \mathbf{K}_i$ by stacking the vectors $\tilde{o}_{it}, \tilde{\mathbf{i}}_{it}, \tilde{\mathbf{j}}_{it}, \tilde{\mathbf{k}}_{it}$ vertically as before, we have immediately,

$$\mathbf{p}_j = \mathbf{O}_1 + \alpha_j \mathbf{I}_1 + \beta_j \mathbf{J}_1 + \gamma_j \mathbf{K}_1, \;\; \mathbf{q}_j = \mathbf{O}_2 + \alpha_j \mathbf{I}_2 + \beta_j \mathbf{J}_2 + \gamma_j \mathbf{K}_2. \tag{4}$$

The two linear subspaces L_p, L_q are spanned by the basis vectors $\{\mathbf{O}_1, \mathbf{I}_1, \mathbf{J}_1, \mathbf{K}_1\}$, $\{\mathbf{O}_2, \mathbf{I}_2, \mathbf{J}_2, \mathbf{K}_2\}$, respectively. The linear map that produces the correct correspondences is given by the linear map $\mathbf{L}$ such that $\mathbf{L}(\mathbf{O}_1) = \mathbf{O}_2, \mathbf{L}(\mathbf{I}_1) = \mathbf{I}_2, \mathbf{L}(\mathbf{J}_1) = \mathbf{J}_2$ and $\mathbf{L}(\mathbf{K}_1) = \mathbf{K}_2$. A further reduction is possible by noticing that the vectors $\mathbf{p}_j, \mathbf{q}_j$ belong to two three-dimensional affine linear subspaces L'_p, L'_q in $\mathbb{R}^{2T}$, affine subspaces that pass through the points $\mathbf{O}_1, \mathbf{O}_2$ with bases $\{\mathbf{I}_1, \mathbf{J}_1, \mathbf{K}_1\}$ and $\{\mathbf{I}_2, \mathbf{J}_2, \mathbf{K}_2\}$, respectively. These two subspaces can be obtained by computing the principle components for the collections of vectors $\mathcal{P}, \mathcal{Q}$. By projecting points in $\mathcal{P}, \mathcal{Q}$ onto L'_p, L'_q, respectively, it is clear that the two sets of projected points are now related by an affine map $\mathbf{A} : L'_p \rightarrow L'_q$. In other words, the correspondence problem can now be solved by solving the equivalent affine registration problem for these two sets of projected points (in $\mathbb{R}^3$).

Non-Rigid Motions. The above discussion generalizes immediately to the types of non-rigid motions that can be modelled (or approximated) using linear shape basis [2,5,9]. In this model, for k feature points, a shape basis element $\mathbf{B}_l$ is a $3 \times k$ matrix. For a model that employs m linear shape basis elements, the 3D world coordinates of the k feature points at $t^{\mathbf{th}}$ frame can be written as a linear combination of these shape basis elements:

$$\left[X_1^t \cdots X_k^t\right] = \sum_{l=1}^{m} a_l^t \mathbf{B}_l, \tag{5}$$

for some real numbers a_l^t. Using affine camera model, the imaged points (disregarding the global translation) are given by the following equation [9]

$$\left[\mathbf{x}_1^t \cdots \mathbf{x}_k^t\right] = (\mathbf{a} \otimes P)\mathbf{B}, \tag{6}$$

where $\mathbf{a}^t = (a_1^t, \cdots, a_m^t)$, P is the first 2×3 block of the affine camera matrix and $\mathbf{B}$ is the $3m \times k$ matrix formed by vertically stacking the shape basis matrices $\mathbf{B}_l$. The right factor in the above factorization is independent of the camera (and the images), and we have the following equations similar to Equations 4:

$$\mathbf{p}_j = \mathbf{O}_1 + \sum_{l=1}^{m}(\alpha_{jl}\mathbf{I}_{1l} + \beta_{jl}\mathbf{J}_{1l} + \gamma_{jl}\mathbf{K}_{1l}), \;\; \mathbf{q}_j = \mathbf{O}_2 + \sum_{l=1}^{m}(\alpha_{jl}\mathbf{I}_{2l} + \beta_{jl}\mathbf{J}_{2l} + \gamma_{jl}\mathbf{K}_{2l}), \tag{7}$$

where $\mathbf{I}_{il}, \mathbf{J}_{il}\mathbf{K}_{il}$ are the projections of the three basis vectors in the $l^{\mathbf{th}}$ shape basis element $\mathbf{B}_l$ onto camera i. The numbers α_{jl}, β_{jl} and γ_{jl} are in fact entries in the matrix $\mathbf{B}_l$. These two equations then imply, using the same argument as before, that we can recover the correspondences directly using a $3m$-dimensional affine registration provided that the vectors $\mathbf{O}_i, \mathbf{I}_{il}, \mathbf{J}_{il}, \mathbf{K}_{il}$ are linearly independent for each i, which is typically the case when the number of frames is sufficiently large.

2.2 Image Set Matching

In the image set matching problem, we are given two sets of images $\mathcal{P} = \{I_1, \cdots, I_k\} \subset \mathbb{R}^m$, $\mathcal{Q} = \{I_1', \cdots, I_k'\} \subset \mathbb{R}^{m'}$ and the corresponding pairs of images I_i, I_i' are related by a linear transformation $\mathcal{F} : \mathbb{R}^m \rightarrow \mathbb{R}^{m'}$ between two high-dimensional image spaces:

$$I_k' \approx \mathcal{F}(I_k).$$

Examples of such sets of images are quite easy to come by, and Figure 1 gives an example in which I_i' is obtained by rotating and downsizing I_i. It is easy to see that many standard image processing operations such as image rotation and down-sampling can be modelled as (or approximated by) a linear map $\mathcal{F}$ between two image spaces. The problem here is to recover the correspondences $I_i \leftrightarrow I_i'$ without actually computing the linear transformation $\mathcal{F}$, which will be prohibitively expensive since the dimensions of the image spaces are usually very high.

Many interesting sets of images can in fact be approximated well by low-dimensional linear subspaces in the image space. Typically, such linear subspaces can be computed readily using principal component analysis (PCA). Let L_p, L_q denote two such low-dimensional linear subspaces approximating $\mathcal{P}, \mathcal{Q}$, respectively and we will use the same notations $\mathcal{P}, \mathcal{Q}$ to denote their projections onto the subspace L_p, L_q. A natural question to ask is how are the (projected) point sets $\mathcal{P}, \mathcal{Q}$ related? Suppose that $\mathcal{F}$ is orthogonal and L_p, L_q are the principle subspaces of the same dimension. If the data is "noiseless", i.e., $I'_k = \mathcal{F}(I_k)$, it is easy to show that $\mathcal{P}, \mathcal{Q}$ are then related by an orthogonal transformation. In general, $\mathcal{F}$ may not be orthogonal and data points are noisy, the point sets $\mathcal{P}, \mathcal{Q}$ are related by a transformation $\mathbf{T} = \mathbf{A} + \mathbf{r}$, which is a sum of an affine transformation $\mathbf{A}$ and a nonrigid transformation $\mathbf{r}$. If the nonrigid part is small, we can recover the correspondences by affine registering the two point sets $\mathcal{P}, \mathcal{Q}$. Note that this gives an algorithm for computing the correspondences without explicitly using the image contents, i.e., there is no feature extraction. Instead, it works directly with the geometry of the point sets.

3 Affine Registrations in $\mathbb{R}^m$

The above discussion provides the motivation for studying affine registration in $\mathbb{R}^m$ for $m > 3$. Let $\mathcal{P} = \{p_1, \cdots, p_k\}$ and $\mathcal{Q} = \{q_1, \cdots, q_k\}$ be two point sets in $\mathbb{R}^m$ related by an unknown affine transformation

$$q_{\pi(i)} = \mathbf{A}p_i + \mathbf{t}, \tag{8}$$

where $\mathbf{A} \in \mathbf{GL}(m)$, $\mathbf{t} \in \mathbb{R}^m$ the translational component of the affine transformation and $\pi : \mathcal{P} \to \mathcal{Q}$, the unknown correspondence to be recovered. We assume that the point sets $\mathcal{P}, \mathcal{Q}$ have same number of points and π is a bijective correspondence.

Iterative closest point (ICP) algorithm is a very general point registration algorithm that generalizes easily to higher dimensions. Several papers have been published recently [10,11,12,13,14] on ICP-related point registration algorithms in $\mathbb{R}^2$ and $\mathbb{R}^3$. While these works concern exclusively with rigid transformations, it is straightforward to incorporate affine transformation into ICP algorithm, which iterative solves for correspondences and affine transformation[2]. Given an assignment (correspondences) $\pi : \{1, \cdots, k\} \to \{1, \cdots, k\}$ the optimal affine transformation $\mathbf{A}$ in the least squares sense can be solved by minimizing

$$\mathcal{E}(\mathbf{A}, \mathbf{t}, \pi) = \sum_{i=1}^{k} \mathbf{d}^2(\mathbf{A}p_i + \mathbf{t}, q_{\pi(i)}). \tag{9}$$

Solving $\mathbf{A}, \mathbf{t}$ separately while holding π fixed, the above registration error function gives a quadratic programming problem in the entries of $\mathbf{A}$, and the optimal solution can be computed readily by solving a linear system. With a fixed $\mathbf{A}$, $\mathbf{t}$ can be solved immediately. On the hand, given an affine transformation, a new assignment π can be defined using closest points:

$$\pi(i) = \arg \min_{1 \le j \le k} \mathbf{d}^2(\mathbf{A}p_i + \mathbf{t}, q_j).$$

[2] We will call this algorithm affine-ICP.

Once an initial affine transformation and assignment is given, affine-ICP is easy to implement and very efficient. However, the main difficulty is the initialization, which can significantly affect the algorithm's performance. With a poor initialization, the algorithm almost always converges to an undesirable local minimum and as the group of affine transformations is noncompact, it is also possible that it diverges to infinity, i.e., the linear part of the affine transformation converges to a singular matrix. One way to generate an initial affine transformation (disregarding $\mathbf{t}$) is to randomly pick m pairs of points from $\mathcal{P}$, $\mathcal{Q}$, $\{(x_1, y_1), \cdots, (x_m, y_m)\}$, $x_i \in \mathcal{P}, y_i \in \mathcal{Q}$ and define $\mathbf{A}$ as $y_i = \mathbf{A}(x_i)$. It is easy to see that the probability of picking a good set of pairs that will yield good initialization is roughly in the order of $1/C(k, m)$. For small dimensions $m = 2, 3$ and medium-size point sets (k in the order of hundreds), it is possible to exhaustively sample all these initial affine transformations. However, as $C(k, m)$ depends exponentially on the dimension m, this approach becomes impractical once $m > 3$. Therefore, for affine-ICP approach to work, we need a novel way to generate good initial affine transformation and correspondences.

Our solution starts with a novel affine registration algorithm. The outline of the algorithm is straightforward: we first reduce the problem to orthogonal case and spectral information is then used to narrow down the correct orthogonal transformation. This algorithm does not require continuous optimization (e.g., solving linear systems) and we can show that for generic point sets without noise, it will recover the exact affine transformation. This latter property suggests that for noisy point sets, the affine transformation estimated by the proposed algorithm should not be far from the optimal one. Therefore, the output of our proposed algorithm can be used as the initial affine transformation for the affine-ICP algorithm.

3.1 Affine Registration Algorithm

Let $\mathcal{P}$, $\mathcal{Q}$ be two point sets as above related by an unknown affine transformation as in Equation 8. By centering the point sets with respect to their respective centers of mass $\mathbf{m}_p, \mathbf{m}_q$,

$$\mathbf{m}_p = \frac{1}{k}\sum_{i=1}^{k} p_i, \qquad \mathbf{m}_q = \frac{1}{k}\sum_{i=1}^{k} q_i,$$

the centered point sets $\mathcal{P}^c = \{p_1 - \mathbf{m}_p, \cdots, p_k - \mathbf{m}_p\}$ and $\mathcal{Q}^c = \{q_1 - \mathbf{m}_q, \cdots, q_k - \mathbf{m}_q\}$ are related by the same $\mathbf{A}$: $q_{\pi(i)} - \mathbf{m}_q = \mathbf{A}(p_i - \mathbf{m}_p)$. That is, we can work with centered point sets $\mathcal{P}^c$ and $\mathcal{Q}^c$. Once $\mathbf{A}$ and π have been recovered from the point sets $\mathcal{P}^c$ and $\mathcal{Q}^c$, the translational component $\mathbf{t}$ can be estimated easily. In the absence of noise, determining the matrix $\mathbf{A}$ is in fact a combinatorial search problem. We can select m linearly independent points $\{p_{i_1}, \cdots, p_{i_m}\}$ from $\mathcal{P}$. For every ordered m points $\omega = \{q_{i_1}, \cdots, q_{i_m}\}$ in $\mathcal{Q}$, there is a (nonsingular) matrix $\mathbf{B}_\omega$ sending p_{i_j} to q_{i_j} for $1 \leq j \leq m$. The desired matrix $\mathbf{A}$ is among the set of such matrices, which numbers roughly k^m (k is the number of points). For generic point sets, this exponential dependence on dimension can be avoided if $\mathbf{A}$ is assumed to be orthogonal. Therefor, we will first use the covariance matrices computed from $\mathcal{P}$ and $\mathcal{Q}$ to reduce the problem to the 'orthogonal case'. Once the problem has been so reduced, there are various ways to finish off the problem by exploiting invariants of the orthogonal matrices, namely, distances. Let $\mathbf{S_P}$ and $\mathbf{S_Q}$ denote the covariance matrices for $\mathcal{P}$ and $\mathcal{Q}$, respectively:

$$\mathbf{S_P} = \sum_{i=1}^{k} p_i p_i^t, \qquad \mathbf{S_Q} = \sum_{i=1}^{k} q_i q_i^t.$$

We make simple coordinates changes using their inverse square-roots:

$$p_i \rightarrow \mathbf{S}_{\mathcal{P}}^{-\frac{1}{2}} p_i, \qquad q_i \rightarrow \mathbf{S}_{\mathcal{Q}}^{-\frac{1}{2}} q_i. \tag{10}$$

We will use the same notations to denote the transformed points and point sets. If the original point sets are related by $\mathbf{A}$, the transformed point sets are then related by $\bar{\mathbf{A}} = \mathbf{S}_{\mathcal{Q}}^{-\frac{1}{2}} \mathbf{A} \mathbf{S}_{\mathcal{P}}^{\frac{1}{2}}$. The matrix $\bar{\mathbf{A}}$ can be easily shown to be orthogonal:

Proposition 1. *Let $\mathcal{P}$ and $\mathcal{Q}$ denote two point sets (of size k) in $\mathbb{R}^m$, and they are related by an unknown linear transformation $\mathbf{A}$. Then, the transformed point sets (using Equation 10) are related by a matrix $\bar{\mathbf{A}}$, whose rows are orthonormal vectors in $\mathbb{R}^m$.*

The proof follows easily from the facts that 1) the covariance matrices $\mathbf{S}_{\mathcal{P}}$ and $\mathbf{S_Q}$ are now identity matrices for the transformed point sets, and 2) $\mathbf{S_Q} = \bar{\mathbf{A}} \mathbf{S}_{\mathcal{P}} \bar{\mathbf{A}}^t$. They together imply that the rows of $\bar{\mathbf{A}}$ must be orthonormal.

3.2 Determining the Orthogonal Transformation $\bar{\mathbf{A}}$

Since the point sets $\mathcal{P}$, $\mathcal{Q}$ have unit covariance matrices, the invariant approach in [1] cannot be applied to solve for the orthogonal transformation $\bar{\mathbf{A}}$. Nevertheless, there are other invariants that can be useful. For example, if the magnitudes of points in $\mathcal{P}$ are all different, registration becomes particularly easy: each point p_{i_j} is matched to the point q_{i_j} with the same magnitude. Of course, one does not expect to encounter such nice point sets very often. However, for orthogonal matrices, there is a very general way to produce a large number of useful invariants.

Let p_1, p_2 be any two points in $\mathcal{P}$ and q_1, q_2 their corresponding points in $\mathcal{Q}$. Since $\bar{\mathbf{A}}$ is orthogonal, the distance $d(p_1, p_2)$ between p_1 and p_2 equals the distance $d(q_1, q_2)$ between q_1 and q_2. Although we do not know the correspondences between points in $\mathcal{P}$ and $\mathcal{Q}$, the above observation naturally suggests the idea of canonically constructing two symmetric matrices, $L_{\mathcal{P}}$ and $L_{\mathcal{Q}}$, using pairwise distances between points in $\mathcal{P}$ and $\mathcal{Q}$, respectively. The idea is that the matrices so constructed differ only by an unknown permutation of their columns and rows. Their eigenvalues, however, are not effected by such permutations, and indeed, the two matrices $L_{\mathcal{P}}$ and $L_{\mathcal{Q}}$ have the same eigenvalues. Furthermore, there are also correspondences between respective eigenspaces $E_\lambda^{\mathcal{P}}$ and $E_\lambda^{\mathcal{Q}}$ associated with eigenvalue λ. If λ is a non-repeating eigenvalue, we have two associated (unit) eigenvectors $v_{\mathcal{P}}^\lambda$ and $v_{\mathcal{Q}}^\lambda$ of $L_{\mathcal{P}}$ and $L_{\mathcal{Q}}$, respectively. The vector $v_{\mathcal{P}}^\lambda$ differs from $v_{\mathcal{Q}}^\lambda$ by a permutation of its components and a possible multiplicative factor of -1.

There are many ways to construct the matrices $L_{\mathcal{P}}$ and $L_{\mathcal{Q}}$. Let $f(x)$ be any function. We can construct a $k \times k$ symmetric matrix $L_{\mathcal{P}}(f)$ from pairwise distances using the formula

$$L_{\mathcal{P}}(f) = \mathbf{I}_k - \mu \begin{pmatrix} f(d(p_1,p_1)) & \cdots & f(d(p_1,p_k)) \\ \vdots & \dots & \vdots \\ f(d(p_k,p_1)) & \cdots & f(d(p_k,p_k)) \end{pmatrix}, \tag{11}$$

where $\mathbf{I}_k$ is the identity matrix and μ some real constant. One common choice of f that we will use here is the Gaussian exponential $f(x) = \exp(-x^2/\sigma^2)$, and the resulting symmetric matrix $L_{\mathcal{P}}$ is related to the well-known (unnormalized) discrete Laplacian associated with the point set $\mathcal{P}$ [15]. Denote $U_p D_p U_p^t = L_{\mathcal{P}}, U_q D_q U_q^t = L_{\mathcal{Q}}$ the eigen-decompositions of $L_{\mathcal{P}}$ and $L_{\mathcal{Q}}$. When the eigenvalues are all distinct, up to sign differences, U_p and U_q differ only by some unknown row permutation if we order the columns according to the eigenvalues. This unknown row permutation is exactly the desired correspondence π. In particular, we can determine m correspondences by matching m rows of U_p and U_q, and from these m correspondences, we can recover the orthogonal transformation $\bar{\mathbf{A}}$. The complexity of this operation is $O(mk^2)$ and we have the following result

Proposition 2. *For a generic pair of point sets $\mathcal{P}, \mathcal{Q}$ with equal number of points in $\mathbb{R}^m$ related by some orthogonal transformation $\mathbf{L}$ and correspondences π such that $q_{\pi(i)} = \mathbf{L}p_i$, the above method will recover $\mathbf{L}$ and π exactly for some choice of σ.*

The proof (omitted here) is an application of Sard's theorem and transversality in differential topology [16]. The main idea is to show that for almost all point sets $\mathcal{P}$, the symmetric matrix $L_{\mathcal{P}}$ will not have repeating eigenvalues for some σ. This will guarantee that the row-matching procedure described above will find the m needed correspondences after examining all rows of U_p m times. Since the time complexity for matching one row is $O(k)$, the total time complexity is no worse than $O(mk^2)$.

3.3 Dealing with Noises

The above method breaks down when noise is present. In this case, the sets of eigenvalues for $L_{\mathcal{P}}$, $L_{\mathcal{Q}}$ are in general different, and the matrices U_p, U_q are no longer expected to differ only by a row permutation. Nevertheless, for small amount of noise, one can expect that the matrices $L_{\mathcal{P}}$, $L_{\mathcal{Q}}$ are small perturbations of two corresponding matrices for noiseless data. For example, up to a row permutation, U_q is a small perturbation of U_p. For each eigenvalue λ^p of $L_{\mathcal{P}}$, there should be an eigenvalue λ^q of $L_{\mathcal{Q}}$ such that the difference $|\lambda_p - \lambda_q|$ is small, and this will allow us to establish correspondences between eigenvalues of $L_{\mathcal{P}}$, $L_{\mathcal{Q}}$. The key idea is to define a reliable matching measure $\mathbf{M}$ using eigenvectors of $L_{\mathcal{P}}, L_{\mathcal{Q}}$, e.g., if p, q are two corresponding points, $\mathbf{M}(p, q)$ will tend to be small. Otherwise, it is expected to be large. Once a matching measure $\mathbf{M}$ is defined, it will allow us to establish tentative correspondences $p_i \longleftrightarrow q_j$: $q_j = \arg\min_{i'} \mathbf{M}(p_i, q_{i'})$. Similar to the homography estimation in structure from motion [2], some of the tentative correspondences so established are incorrect while a good portion of them are expected to be correct. This will allow us to apply RANSAC [17] to determine the orthogonal transformation: generate a small number of hypotheses (orthogonal matrices from sets of randomly generated m correspondences) and pick the one that gives the smallest registration error. We remark that in our approach, the tentative correspondences are computed from the geometry of the point sets $\mathcal{P}$, $\mathcal{Q}$ embedded in $\mathbb{R}^m$. In stereo matching and homography estimation [2], they are computed using image features such as image gradients and intensity values.

More precisely, let $\lambda_1^p < \lambda_2^p < \cdots < \lambda_l^p$ ($l \leq k$) be l non-repeating eigenvalues of $L_{\mathcal{P}}$ and likewise, $\lambda_1^q < \lambda_2^q < \cdots < \lambda_l^q$ the l eigenvalues of $L_{\mathcal{Q}}$ such that $|\lambda_i^q - \lambda_i^p| < \epsilon$ for some threshold value ϵ. Let $v_{\lambda_1}^{\mathcal{P}}, v_{\lambda_2}^{\mathcal{P}}, \cdots, v_{\lambda_l}^{\mathcal{P}}$, and $v_{\lambda_1}^{\mathcal{Q}}, v_{\lambda_2}^{\mathcal{Q}}, \cdots, v_{\lambda_l}^{\mathcal{Q}}$ denote the corresponding eigenvectors. We stack these eigenvectors horizontally to form two $k \times l$ matrices $\mathtt{VP}$ and $\mathtt{VQ}$:

$$\mathtt{VP} = [v_{\lambda_1}^{\mathcal{P}}\, v_{\lambda_2}^{\mathcal{P}}\, \cdots v_{\lambda_l}^{\mathcal{P}}], \qquad \mathtt{VQ} = [v_{\lambda_1}^{\mathcal{Q}}\, v_{\lambda_2}^{\mathcal{Q}}\, \cdots v_{\lambda_l}^{\mathcal{Q}}]. \tag{12}$$

Denote the i, j-entry of $\mathtt{VP}$ (and also $\mathtt{VQ}$) by $\mathtt{VP}(i, j)$. We define the matching measure $\mathbf{M}$ as

$$\mathbf{M}(p_i, q_j) = \sum_{h=1}^{l} \min\{\, (\mathtt{VP}(i,h) - \mathtt{VQ}(j,h))^2,\, (\mathtt{VP}(i,h) + \mathtt{VQ}(j,h))^2 \,\}.$$

Note that if $l = k$, $\mathbf{M}$ is comparing the $i^{\mathbf{th}}$ row of $L_{\mathcal{P}}$ with $j^{\mathbf{th}}$ row of $L_{\mathcal{Q}}$. For efficiency, one does not want to compare the entire row; instead, only a small fragment of it. This would require us to use those eigenvectors that are most discriminating for picking the right correspondences. For discrete Laplacian, eigenvectors associated with smaller eigenvalues can be considered as smooth functions on the point sets, while those associated with larger eigenvalues are the non-smooth ones since they usually exhibit greater oscillations. Typically, the latter eigenvectors provide more reliable matching measures than the former ones and in many cases, using one or two such eigenvectors ($l = 2$) is already sufficient to produce good results.

Table 1. Experimental Results I. For each dimension and each noise setting, one hundred trials, each with different point sets and matrix $\mathbf{A}$, were performed. The averaged relative error and percentage of mismatched points as well as standard deviations (in parenthesis) are shown.

Dim → Noise ↓	3 Matrix Error	3 Matching Error	5 Matrix Error	5 Matching Error	10 Matrix Error	10 Matching Error
0%	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)
1%	0.001 (0.0005)	0 (0)	0.002 (0.0006)	0 (0)	0.004 (0.0008)	0 (0)
2%	0.003 (0.001)	0 (0)	0.004 (0.001)	0 (0)	0.008 (0.001)	0 (0)
5%	0.008 (0.003)	0 (0)	0.01 (0.003)	0 (0)	0.02 (0.003)	0 (0)
10%	0.017 (0.01)	0.008 (0.009)	0.05 (0.05)	0.009 (0.04)	0.04 (0.009)	0 (0)

4 Experiments

In this section, we report four sets of experimental results. First, with synthetic point sets, we show that the proposed affine registration algorithm does indeed recover exact affine transformations and correspondences for noiseless data. Second, we show that the proposed algorithm also works well for 2D point sets. Third, we provide two sequences of nonrigid motions and show that the feature point correspondences can be

Table 2. Experimental Results II. Experiments with point sets of different sizes with 5% noise added. All trials match point sets in $\mathbb{R}^{10}$ with settings similar to Table 1. Average errors for one hundred trials are reported with standard deviations in parenthesis.

# of Pts → Errors ↓	100 Points	150 Points	200 Points	250 Points	300 Points	400 Points
Matrix Error	0.02 (0.003)	0.05(0.008)	0.05 (0.009)	0.05 (0.01)	0.05 (0.01)	0.04 (0.009)
Matching Error	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)	0 (0)

satisfactorily solved using affine registration in $\mathbb{R}^9$. And finally, we use images from COIL database to show that the image set matching problem can also be solved using affine registration in $\mathbb{R}^8$. We have implemented the algorithm using MATLAB without any optimization. The sizes of the point sets range from 20 to 432, and on a DELL desktop with single 3.1GHz processor, each experiment does not run longer than one minute.

4.1 Affine Registration in $\mathbb{R}^m$

In this set of experiments, our aim is to give a qualitative as well as quantitative analysis on the accuracy and robustness of the proposed method. We report our experimental results on synthetic data in several different dimensions and using various different noise settings. Tables 1 and 2 summarize the experimental results. In Table 1, the algorithm is tested in three dimensions, 3, 5 and 10, and five different noise settings, $0\%, 1\%, 2\%, 5\%, 10\%$. For each pair of dimension and noise setting, we ran 100 trials, each with a randomly generated non-singular matrix $\mathbf{A}$ and a point set containing 100 points. In trials with $x\%$ noise setting, we add a uniform random noise ($\pm x\%$) to each coordinate of every point independently. Let $\mathbf{A}'$ denote the estimated matrix. A point $p \in \mathcal{P}$ is matched to the point $q \in \mathcal{Q}$ if $q = \min_{q_i \in \mathcal{Q}} \mathbf{dist}(\mathbf{A}'p, q_i)$. For each trial, we report the percentage of mismatched points and the relative error of the estimated matrix $\mathbf{A}'$: $\frac{\|\mathbf{A}'-\mathbf{A}\|}{\|\mathbf{A}\|}$, using the Frobenius norm.

The number of (RANSAC) samples drawn in each trial has been fixed at 800 for the results reported in Table 1. This is the number of samples needed to produce zero mismatch for dimension 10 with 10% noise setting. In general, for lower dimensions, a much smaller number of samples (around 200) would also have produced similar results. In Table 2, we vary the sizes of the point sets and work in $\mathbb{R}^{10}$. The setting is similar to that of Table 1 except with fixed 5% noise setting for all trials. The results clearly show that the proposed algorithm consistently performs well with respect to the sizes of the point sets. Note also that for noiseless point sets the exact affine transformations are always recovered.

4.2 2D Point Sets

In the second set of experiments, we apply the proposed algorithm to 2D image registration. It is known that the effect of a small view change on an image can be approximated by a 2D affine transformation of the image [2]. Using images from COIL database, we manually click feature points on pairs of images with 15° to 30° difference in view

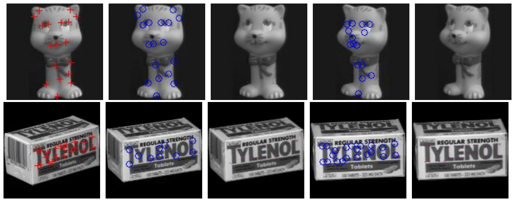

Fig. 2. 2D Image Registration. **1st column**: Source images (taken from COIL database) with feature points marked in red. **2nd and 4th column**: Target images with feature points marked in blue. **3rd and 5th column**: Target images with corresponding feature points marked in blue. The affine transformed points from the source images are marked in red. Images are taken with 15° and 30° differences in viewpoint. The RMS errors for these four experiments (from left to right) $2.6646, 3.0260, 2.0632, 0.7060$, respectively.

Fig. 3. Top: Sample frames from two video sequences of two objects undergoing nonrigid motions. **Bottom:** Sample frames from another camera observing the same motions.

point. The registration results for four pairs of images are shown in Figure 2. Notice the small RMS registration errors for all these results given that the image size is 128×128.

4.3 Stereo Correspondences under Nonrigid Motions

In this experiment, we apply affine registration algorithm to compute correspondences between tracked feature points in two image sequences. We gathered four video sequences from two cameras observing two objects undergoing nonrigid motions (Figure 3). One is a talking head and the other is a patterned tatoo on a man's belly. A simple correlation-based feature point tracker is used to track twenty and sixty points for these two sequences, respectively. Seventy frames were tracked in both sequences and manual intervention was required several times in both sequences to correct and adjust tracking results. We use three shape basis for both sequences [5], and to compute the correspondences, we affine register two point sets in $\mathbb{R}^9$ as discussed before. For the two point sets $\mathcal{P}, \mathcal{Q} \subset \mathbb{R}^9$, we applied the proposed algorithm to obtain initial correspondences and affine transformation. This is followed by running an affine-ICP algorithm with fifty iterations. For comparison, the affine-ICP algorithm initialized using closest points[3] is run for one hundred iterations. For the talking sequence, the

[3] Given two point sets in $\mathbb{R}^9$, the initial correspondence $p_i \leftrightarrow q_j$ is computed by taking q_j to be the point in $\mathcal{Q}$ closest to p_i.

proposed algorithm recovers all the correspondences correctly, while for the tatoo sequence, among the recovered sixty feature point correspondences, nine are incorrect. This can be explained by the fact that in several frames, some of the tracked feature points are occluded and missing and the subsequent factorizations produce relatively noisy point sets in $\mathbb{R}^9$. On the other hand, affine-ICP with closest point initialization fails poorly for both sequences. In particular, more than three quarters of the estimated correspondences are incorrect.

4.4 Image Set Matching

In this experiment, images from the first six objects in the COIL database are used. They define the image set $\mathcal{A}$ with 432 images. Two new sets $\mathcal{B}, \mathcal{C}$ of images are generated from $\mathcal{A}$: the images are 80% down-sampled and followed by $45°$ and $90°$ rotations, respectively. The original images have size 128×128 and the images in the two new sets have size 100×100. An eight-dimensional PCA subspace is used to fit each set of images with relative residue smaller than 1%. Images in each set are projected down to their respective PCA subspaces and the correspondences are automatically computed by affine registering the projected point sets. The two experiments shown in Figure 4 match point sets $\mathcal{A}, \mathcal{B}$ and $\mathcal{A}, \mathcal{C}$. We apply the proposed affine registration algorithm to obtain an initial estimate on correspondences and affine transformation. Since the data is noisy, we follow this with the affine-ICP algorithm running fifty iterations as above. For comparison, we apply the affine-ICP algorithm using closest points as initialization. In both experiments, the affine-ICP algorithm, not surprisingly, performs poorly with substantial L^2-registration errors (Equation 9) and large number of incorrect correspondences. The proposed algorithm recovers all correspondences correctly and it yields small L^2-registration errors.

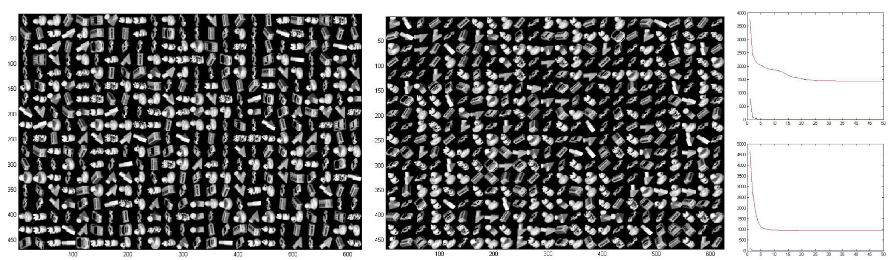

Fig. 4. Image Set Matching. The original image set $\mathcal{A}$ is shown in Figure 1. Image sets $\mathcal{B}, \mathcal{C}$ are shown above. The plots on the right show the L^2-registration error for each of the fifty iterations of running affine-ICP algorithm using different initializations. Using the output of the proposed affine registration as the initial guess, the affine-ICP algorithm converges quickly to the desired transformation (blue curves) and yields correct correspondences. Using closest points for initial correspondences, the affine-ICP algorithm converges (red curves) to incorrect solutions in both experiments.

5 Conclusion and Future Work

In this paper, we have shown that the stereo correspondence problem under motion and image set matching problem can be solved using affine registration in $\mathbb{R}^m$ with $m > 3$. We have also proposed an algorithm for estimating an affine transformation directly from two point sets without using continuous optimization. In the absence of noise, it will recover the exact affine transformation for generic pairs of point sets in $\mathbb{R}^m$. For noisy data, the output of the proposed algorithm often provides good initializations for the affine-ICP algorithm. Together, they provide us with an efficient and effective algorithm for affine registering point sets in $\mathbb{R}^m$ with $m > 3$. We have applied the proposed algorithm to the two aforementioned problems. Preliminary experimental results are encouraging and they show that these two problems can indeed be solved satisfactorily using the proposed affine registration algorithm.

References

1. Scott, G., Lonquiet-Higgins, C.: An algorithm for associating the features of two images. Proc. of Royal Society of London B244, 21–26 (1991)
2. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision. Cambridge University Press, Cambridge (2003)
3. Chui, H., Rangarajan, A.: A new algorithm for non-rigid point matching. In: Proc. IEEE Conf. on Comp. Vision and Patt. Recog, vol. 2, pp. 44–51 (2000)
4. Toamsi, C., Kanade, T.: Shape and motion from image streams under orthography—a factorization method. Int. J. Computer Vision 9(2), 137–154 (1992)
5. Bregler, C., Hertzmann, A., Biermann, H.: Recovering non-rigid 3d shape from image streams. In: Proc. IEEE Conf. on Comp. Vision and Patt. Recog., pp. 2690–2696 (2000)
6. Besel, P.J., Mckay, H.D.: A method for registration of 3-d shapes. PAMI 14, 239–256 (1992)
7. Zhang, Z.: Iterative point matching for registration of free-form curves and surfaces. Int. J. Computer Vision 13, 119–152 (1994)
8. Kanatani, K.: Motion segmentation by subspace separation and model selection. In: Proc. Int. Conf. on Computer Vision, vol. 2, pp. 586–591 (2001)
9. Brand, M.: Morphable 3d models from video. In: Proc. IEEE Conf. on Comp. Vision and Patt. Recog., vol. 2, pp. 456–463 (2001)
10. Fitzgibbon, A.W.: Robust registration of 2d and 3d point sets. Computer Vision and Image Understanding 2, 1145–1153 (2003)
11. Sharp, G.C., Lee, S.W., Wehe, D.K.: Icp registration using invariant features. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 90–102 (2002)
12. Rusinkiewicz, S., Levoy, M.: Efficient variants of the icp algorithm. In: Proc. Third International Conference on 3D Digital Imaging and Modeling (3DIM), pp. 145–152 (2001)
13. Granger, S., Pennec, X.: Multi-scale em-icp: A fast and robust approach for surface registration. In: Proc. European Conf. on Computer Vision, vol. 3, pp. 418–432 (2002)
14. Makadia, A., Patterson, A.I., Daniilidis, K.: Fully automatic registration of 3d point clouds. In: Proc. IEEE Conf. on Comp. Vision and Patt. Recog., vol. 1, pp. 1297–1304 (2006)
15. Chung, F.R.K.: Spectral Graph Theory. American Mathematical Society (1997)
16. Hirsch, M.: Differential Topology. Springer, Heidelberg (1976)
17. Fischler, M., Bolles, R.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. Communications of the ACM 24, 381–395 (1981)