A Direct Method for Estimating Planar Projective Transform

Yu-Tseh Chi[1], Jeffrey Ho[1] and Ming-Hsuan Yang[2]

[1]Computer and Information Science and Engineering, University of Florida,
Gainesville, FL 32611, U.S.A.
{ychi,joh}@cise.ufl.edu
[2]Electrical Engineering and Computer Science, University of California at Merced,
Merced, CA 95344, U.S.A.
mhyang@ucmerced.edu

Abstract. Estimating planar projective transform (homography) from a pair of images is a classical problem in computer vision. In this paper, we propose a novel algorithm for direct registering two point sets in $\mathbb{R}^2$ using projective transform without using intensity values. In this very general context, there is no easily established correspondences that can be used to estimate the projective transform, and most of the existing techniques become either inadequate or inappropriate. While the planar projective transforms form an eight-dimensional Lie group, we show that for registering 2D point sets, the search space for the homographies can be effectively reduced to a three-dimensional space. To further improve on the running time without significantly reducing the accuracy of the registration, we propose a matching cost function constructed using local polynomial moments of the point sets and a coarse to fine approach. The resulting registration algorithm has linear time complexity with respect to the number of input points. We have validated the algorithm using *points sets* collected from real images. Preliminary experimental results are encouraging and they show that the proposed method is both efficient and accurate.

1 Introduction

The importance of 2D projective transforms (planar homographies) in computer vision stems from the well-known fact that different images of a planar scene are related by planar projective transforms. For vision applications that deal with planar objects, examples of which include recognizing hand drawings and traffic signs, etc., estimating the projective transform between two images is required for normalizing the image variation due to different cameras. Accordingly, there are now well-established techniques for automatically recovering planar homography between images [1]. For these feature-based methods, the idea is to register two sets of interest points $\mathbb{P} = \{p_1, \cdots, p_k\}$ and $\mathbb{Q} = \{q_1, \cdots, q_k\}$ extracted from the two images using projective transforms. Intensity values (local intensity patterns) provide important clues on establishing correspondences across images, and Direct Linear Transform (DLT) requires only four correspondences in order to compute a homography. Implicitly, the algorithms define a discrete search space $\mathbb{S}$ that contains a large number of projective transforms (from all possible correspondences made across the images), and the image intensity pattern is used

to prune away most part of $\mathbb{S}$ that are irrelevant, and the algorithms exhaustively test each remaining homography according to some matching cost function to obtain the optimal homography.
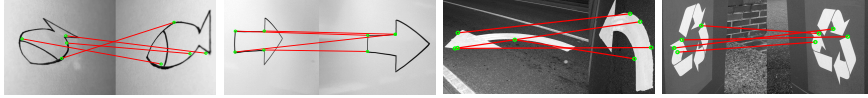

Fig. 1. Four examples for which a direct application of SIFT [2] fails to produce correct correspondences. The first two images are hand drawings on a whiteboard, and the others are images of street signs.

In this paper, we study the 2D projective registration problem in its barest form: using only point sets without intensity values. While DLT requires only four correct correspondences, there are cases where it is difficult to correctly determine these correspondences automatically. Figure 1 shows four pairs of such images that cannot be registered satisfactorily using existing methods. The lack of salient local intensity patterns often confuses algorithms that match local feature descriptors. Furthermore, local geometric structure such as curvatures are not reliable features for computing correspondences since they could vary wildly under projective transforms. Nevertheless, we can still solve the problem by sampling a collection of points from each image and register the point sets using projective transforms[1]. A set of four points (or more) can be chosen, and a homography search space $\mathbb{S}$ can be defined as consisting homographies generated by all correspondences for these four points. For each homography in $\mathbb{S}$, its registration quality can be evaluated by a matching cost function such as

$$\Gamma(\mathbf{H}; \mathbb{P}, \mathbb{Q}) = \sum_{j=1}^{l} \min_{i=1,\cdots,k} |q_j - \mathbf{H}(p_i)|^2. \tag{1}$$

For point sets containing 500 points, $\mathbb{S}$ contains roughly 65 billion homographies. While this is by no mean a small number, it is still within the capability of modern hardware to exhaustively search $\mathbb{S}$ for the optimal homography. However, the quartic time complexity makes this approach unattractive since if 5000 points were sampled instead, the search space would increase enormously to the point that renders the algorithm impotent, even though the two problems do not different in any essential way other than the number of points (they have the same images).

The approach advocated in this paper is to abandon the usual paradigm of defining the search space $\mathbb{S}$ using small number of correspondences, which is often impossible for us anyway. Instead, the search space $\mathbb{S}$ should be defined entirely by the point sets in some global way. The main contribution of this paper is to show that, under a fairly general assumption on the variability of the homographies, we can define an effective three-dimensional (discrete) search space $\mathbb{S}$ for projective registration of two point sets in $\mathbb{R}^2$. The size of $\mathbb{S}$ is independent of the sizes of the input point sets, and in many cases, it is possible exhaustively search for the optimal homography in $\mathbb{S}$. Therefore, the time complexity of the

---

[1] Or we can register the binary image obtained from the segmented image as in [3]

algorithm depends on the complexity of evaluating the matching cost function. For Γ defined in Equation 1, the complexity is quadratic since it requires the computation of pairwise distances. However, by using a less expensive matching cost function constructed from local polynomial moments of the point sets, we show that it is possible to have an accurate projective registration algorithm with linear time-complexity.

The projective registration method proposed in this paper is direct in its truest sense: the algorithm takes in two point sets, which can be readily generated from images using low-level image processing, and outputs a projective transform that matches them. No other extra mid- or high-level processing such as contour detection or local feature extraction are necessary. Since the algorithm exhaustively evaluates every homography in the search space $\mathbb{S}$, it can provide a reasonable guarantee on the quality of the registration. We conclude this introduction with a summary of three main contributions of this paper:

1. We show that the search space $\mathbb{S}$ for planar projective matching can be effectively reduced to a three-dimensional discrete search space, and the resulting algorithm has a quadratic $\mathbf{O}(n^2)$ time-complexity on the size of the input point sets, which is a substantial improvement over the existing quartic complexity.
2. We show that the time-complexity of the matching algorithm can be further reduced to $\mathbf{O}(n)$ by using matching cost function constructed from local polynomial moments.
3. We develop and implement a coarse to fine registration approach. We validate the proposed method using simple point sets and points sets collected from real images, and encouraging preliminary results suggest the method's potential for future applications.

2   Related Works

Estimating homography from pairs of images has been studied quite extensively in the literature. In particular, [1] contains a chapter on feature-based methods using Direct Linear Transform (DLT) and RANSAC [4]. The feature-based algorithms typically extract a large number of interest points from the images, and the tentative correspondences are computed across images by matching descriptors associated to the interest points. A large number of hypothetical homographies are generated, each from a small number of tentative correspondences, and evaluated with some matching cost function. The homography that provides the best registration result is used as the initial guess in a non-linear optimization algorithm that form the second refinement step.

While the feature-based methods have become the standard repertoire of computer vision algorithms, they are clearly inadequate for handling point sets without the associated intensity values since the tentative correspondences cannot be established as easily without local intensity patterns. It seems that not much work has been done on matching points directly using projective transforms. Perhaps the difficulty of the problem is best illustrated by the non-existence result for finite projective invariants shown in [5]. However, in [6], a set of infinite projective invariants has been proposed, and some of its implication

have recently been investigated in the work of [7, 3] on registering binary images. The method proposed in [3] is very general and it works with binary images obtained from segmented images. The algorithm matches the values of a collection of integrals, which can then be turned into a collection of independent equations. The non-linear minimization in [3] is solved using Levenberg-Marquardt algorithm, which in general, unfortunately, does not guarantee convergence to the global minimum. Finally, we remark that the affine registration of point sets have been studied quite thoroughly in the literature e.g., [8–17], and many of these algorithms have found important applications in vision, graphics and other areas. In addition, nonrigid point-set registration has also received fair amount of attention, particularly in the medical imaging community, e.g., [18, 19].

3   2D Projective Registration

Let $\mathbb{P} = \{p_1, \cdots, p_k\}$ and $\mathbb{Q} = \{q_1, \cdots, q_l\}$ denote the two point sets for which we are seeking a 2D planar homography $\mathbf{H}$

$$\mathbf{H} = \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & 1 \end{pmatrix} \tag{2}$$

that will align them. The entry h_{33} is set to one and it corresponds to the assumption that the planar homography does not map the origin to the line at infinity. Assume for the moment that the two point sets have equal number of points $k = l$, and the correspondences under $\mathbf{H}$ is exact: $q_i = \mathbf{H}(p_i)$, for $1 \leq i \leq k$. Given $\mathbf{H}$ as above, the formulas for computing the transformation $q = \mathbf{H}(p)$ (given $p = [p_x, p_y], q = [q_x, q_y]$) are

$$q_x = \frac{h_{11}p_x + h_{12}p_y + h_{13}}{h_{31}p_x + h_{32}p_y + 1}, \qquad q_y = \frac{h_{21}p_x + h_{22}p_y + h_{23}}{h_{31}p_x + h_{32}p_y + 1}. \tag{3}$$

For a fixed pair of (h_{31}, h_{32}), the above formulas become

$$q_i = \frac{1}{\alpha_i}\mathbf{A}\,p_i + \frac{1}{\alpha_i}\mathbf{t}, \tag{4}$$

where

$$\mathbf{A} = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix}, \qquad \mathbf{t} = \begin{pmatrix} h_{13} \\ h_{23} \end{pmatrix},$$

and the point-dependent constant α_i equals the denominator in Equation 3. When $h_{31} = h_{32} = 0$, $a_i = 1$ for all i and Equation 4 reduces to an affine transform. For any other pair h_{31}, h_{32}, Equation 4 defines a transform between the modified point set $\tilde{\mathbb{P}} = \{p_1/\alpha_1, \cdots, p_k/\alpha_k\}$ and $\mathbb{Q}$ that appears superficially similar to an affine transform except for the non-uniform translation $\mathbf{t}/\alpha_i$. The pair h_{31}, h_{32}, the non-affine components of $\mathbf{H}$, will play an important role in the following discussion.

The basic question we strive to answer in this paper is:

  Suppose the ranges of the two non-affine components are known, $\sigma_1 \leq h_{31} \leq \sigma_2$ and $\sigma_3 \leq h_{32} \leq \sigma_4$ for some $\sigma_1, \sigma_2, \sigma_3, \sigma_4$. How should this information be utilized in 2D projective registration?

Our main observation, which motivates our approach, is that for a fixed pair of h_{31}, h_{32}, registration using Equation 4 can be *effectively* reduced to a search problem on a compact and closed one-dimensional subset $\mathbf{O}_{h_{31}, h_{32}}$ of $\mathbf{GL}(2, \mathbb{R})$.

This is a straightforward generalization of a well-known result for affine transforms described below. Now armed with this result, our method is very easy to described: Given the range of the two non-affine components and some specified precision requirement for the homography matrix, we can form a 2D grid $\mathcal{H}$ of values for h_{31}, h_{32}. For each $(h_{31}, h_{32}) \in \mathcal{H}$, we can also discretize its corresponding one-dimensional set $\mathbf{O}_{h_{31}, h_{32}}$, and this provides us with a three-dimensional discrete search space $\mathbb{S} \equiv \{ s_{ijk} \}$ such that each $s_{ijk} \in \mathbb{S}$ corresponds to a 2D projective transform. Our algorithm simply does an exhaustive search on $\mathbb{S}$ to locate the optimal homography.

3.1   Special Case of Affine Transforms

For an affine registration, $q_i = \mathbf{A}p_i + \mathbf{t}$, the translation component $\mathbf{t}$ satisfies the relation $\mathbf{m}_{\mathbb{Q}} = \mathbf{A}\mathbf{m}_{\mathbb{P}} + \mathbf{t}$, where $\mathbf{m}_{\mathbb{Q}}, \mathbf{m}_{\mathbb{P}}$ are the centers of mass of $\mathbb{P}, \mathbb{Q}$, respectively. Once the points are centered:

$$p_i \rightarrow p_i - \mathbf{m}_{\mathbb{P}}, \qquad q_i \rightarrow q_i - \mathbf{m}_{\mathbb{Q}},$$

we have only the linear component $q_i = \mathbf{A}p_i$. The affine transform must match the two covariance matrices

$$S_{\mathbb{P}} = \frac{1}{k} \sum_{i=1}^{k} p_i p_i^t, \qquad S_{\mathbb{Q}} = \frac{1}{k} \sum_{i=1}^{k} q_i q_i^t, \qquad and\ S_{\mathbb{Q}} = \mathbf{A} S_{\mathbb{P}} \mathbf{A}^t.$$

Since both $S_{\mathbb{P}}, S_{\mathbb{Q}}$ are positive-definite, their square roots exist, and the equation above implies that

$$\mathbf{I}_{2 \times 2} = (S_{\mathbb{Q}}^{-\frac{1}{2}} \mathbf{A} S_{\mathbb{P}}^{\frac{1}{2}})(S_{\mathbb{Q}}^{-\frac{1}{2}} \mathbf{A} S_{\mathbb{P}}^{\frac{1}{2}})^t.$$

That is, the matrix $S_{\mathbb{Q}}^{-\frac{1}{2}} \mathbf{A} S_{\mathbb{P}}^{\frac{1}{2}}$ is an orthogonal matrix, i.e., a rotation. In particular, under the coordinates transform $p_i \rightarrow S_{\mathbb{P}}^{-\frac{1}{2}} p_i$, and $q_i \rightarrow S_{\mathbb{Q}}^{-\frac{1}{2}} q_i$, the transformed point sets are now related by an orthogonal transform. Note that the transformed point sets now have the identity matrix as their covariance matrices.

The significance of this result in the context of affine registration is that after the two-steps normalization that turns each point set into a point set with zero mean and unit variance, the affine registration is reduced to a rigid registration: finding an orthogonal transform (rotation) in $\mathbf{O}(2)$, which is a one-dimension subset (subgroup as well) of $\mathbf{GL}(2, \mathbb{R})$. Essentially, this normalization step requires that the affine transform exactly matches the linear and quadratic moments of the two point sets [7], and this constraint reduces the original 4D problem to an 1D problem, which is considerably easier to solve. Once the rotation has been determined, we can "unwrap" it to get the full affine transform.

This result can be generalized directly to the projective case, and the main complication comes from the non-uniform translation in Equation 4. The basic idea is still the same in that the transform specified in Equation 4 should match the linear and quadratic moments of the two point sets. The result of this more general reduction will still be an one-dimensional subset of $\mathbf{GL}(2, \mathbb{R})$, but not a subgroup. Nevertheless, it can be effectively parameterized using trigonometric functions, and hence efficiently discretized as well.

3.2   1D Reduction

The main result of this subsection is the following proposition

Proposition 1. *Suppose* $\mathbb{P}, \mathbb{Q}$ *are two sets of points in* $\mathbb{R}^2$ *related exactly by a projective transform. There exist coordinates transforms*

$$p_i \to \tilde{p}_i, \qquad q_i \to \tilde{q}_i, \tag{5}$$

and a vector $\mathbf{t}$, *a matrix* $\mathbf{E}$, *a diagonal matrix* $\mathbf{D}$ *and constants* $\beta_1, \cdots, \beta_k$ *determined entirely by* $\mathbb{P}, \mathbb{Q}$ *such that the new sets of transformed points* $\tilde{\mathbb{P}}, \tilde{\mathbb{Q}}$ *are related by the following transform*

$$\tilde{q}_i = \mathbf{A}\tilde{p}_i + \beta_i \mathbf{t}, \tag{6}$$

where the matrix $\mathbf{A}$ *satisfies the following equation*

$$\mathbf{A}\mathbf{D}\mathbf{A}^t + \mathbf{A}\mathbf{E} + \mathbf{E}^t\mathbf{A}^t + \mathbf{t}\mathbf{t}^t = \mathbf{I}_{2\times 2}. \tag{7}$$

Proof: We will prove this proposition in a somewhat leisure way by exhibiting all the steps involved in the coordinates transforms, $p_i \to \tilde{p}_i, q_i \to \tilde{p}_i$. Suppose the unknown projective transform is given by Equation 4. We can make the transform $p_i \to \frac{1}{\alpha_i} p_i$ to absorb the constant $\frac{1}{\alpha_i}$:

$$q_i = \mathbf{A}p_i + \frac{1}{\alpha_i}\mathbf{t}. \tag{8}$$

Taking the average of both sides gives $\mathbf{m}_{\mathbb{Q}} = \mathbf{A}\mathbf{m}_{\mathbb{P}} + \alpha\mathbf{t}$, where $\alpha = \frac{1}{k}\sum_{i=1}^{k}\alpha_i^{-1}$. Combining the two equations above yields $q_i = \mathbf{A}p_i + \beta_i(\mathbf{m}_{\mathbb{Q}} - \mathbf{A}\mathbf{m}_{\mathbb{P}})$, where $\beta_i = 1/(\alpha\alpha_i)$.

Make another transform $p_i \to p_i - \beta_i \mathbf{m}_{\mathbb{P}}$, we have $q_i = \mathbf{A}p_i + \beta_i \mathbf{m}_{\mathbb{Q}}$. Let $S_{\mathbb{P}}, S_{\mathbb{Q}}$ denote the covariance matrices, and the transforms $p_i \to S_{\mathbb{P}}^{-\frac{1}{2}} p_i$ and $q_i \to S_{\mathbb{Q}}^{-\frac{1}{2}} q_i$ turn the above equation into

$$q_i = S_{\mathbb{Q}}^{-\frac{1}{2}} \mathbf{A} S_{\mathbb{P}}^{\frac{1}{2}} p_i + \beta_i S_{\mathbb{Q}}^{-\frac{1}{2}} \mathbf{m}_{\mathbb{Q}} \equiv \mathbf{A}p_i + \beta_i \mathbf{t},$$

where we have let $\mathbf{A} = S_{\mathbb{Q}}^{-\frac{1}{2}} \mathbf{A} S_{\mathbb{P}}^{\frac{1}{2}}$ and $\mathbf{t} = S_{\mathbb{Q}}^{-\frac{1}{2}} \mathbf{m}_{\mathbb{Q}}$.

Since now $\sum_{i=1}^{k} q_i q_i^t$ is the identity, we have $\mathbf{I}_{2\times 2} = \mathbf{A}\mathbb{S}\mathbf{A}^t + \mathbf{A}\mathbf{E} + \mathbf{E}^t\mathbf{A}^t + \mathbf{t}\mathbf{t}^t$ where $\mathbb{S} = \sum_{i=1}^{k} p_i p_i^t$, and $\mathbf{E} = \sum_{i=1}^{k} \beta_i p_i \mathbf{t}^t$. Furthermore, since $\mathbb{S}$ is symmetric and positive semi-definite, it has an eigen-decomposition $\mathbb{S} = \mathbf{U}\mathbf{D}\mathbf{U}^t$. Where $\mathbf{D}$ is a diagonal matrix with non-negative entries and $\mathbf{U}$ is an orthogonal matrix. If we define $\mathbf{A} \equiv \mathbf{A}\mathbf{U}$, which is equivalent to the coordinates transform $p_i \to \mathbf{U}^t p_i$, we have finally

$$\mathbf{A}\mathbf{D}\mathbf{A}^t + \mathbf{A}\mathbf{E} + \mathbf{E}^t\mathbf{A}^t + \mathbf{t}\mathbf{t}^t = \mathbf{I}_{2\times 2},$$

where $\mathbf{E} \equiv \mathbf{U}^t\mathbf{E}$.

For each pair of h_{31}, h_{32}, the set of matrices $\mathbf{A}$ satisfying Equation 7 is the set $\mathbf{O}_{h_{31},h_{32}}$ mentioned earlier. The following proposition will show that it is a one-dimensional subset of $\mathbf{GL}(2, \mathbb{R})$. As is clear from the above derivation, the matrices $\mathbf{E}, \mathbf{D}$, the vector $\mathbf{t}$ and the constants β_i, all can be immediately computed from the input points given h_{31}, h_{32}. Once the transformed point sets have been registered using Equation 6 with $\mathbf{A}$ satisfying Equation 7, it is straightforward to unwrap the above steps to recover the homography matrix for the original point sets exactly as in the affine case. What is left now is an effective parameterization for the matrices $\mathbf{A}$ that satisfy Equation 7, which is given by the following Proposition.

Proposition 2. *The components of the matrix* $\mathbf{A} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ *are determined by the matrices* $\mathbf{E}$, $\mathbf{D}$, $\mathbf{T} = \mathbf{t}\mathbf{t}^{\mathbf{t}}$ *and* $\theta, \bar{\theta}$:

$$a = \frac{rP\cos\theta - E}{r}, b = \frac{sP\sin\theta - F}{s}, c = \frac{rP\cos\bar{\theta} - G}{r}, d = \frac{sP\sin\bar{\theta} - H}{s}, \quad (9)$$

where $\mathbf{E} = \begin{bmatrix} E & G \\ F & H \end{bmatrix}$, $\mathbf{T} = \begin{bmatrix} R & T \\ T & S \end{bmatrix}$, $\mathbf{D} = \begin{bmatrix} \sqrt{r} & 0 \\ 0 & \sqrt{s} \end{bmatrix}$

and $P = 1 - R + \frac{E^2}{r} + \frac{F^2}{s}$, $\quad Q = 1 - S + \frac{G^2}{r} + \frac{H^2}{s}$.

The two angles $\theta, \bar{\theta}$ *satisfy* $\cos(\theta - \bar{\theta}) = (\frac{GE}{r} + \frac{FH}{s} - T)/PQ$.

Proof: The proof is straightforward. Expanding the matrix equation (Equation 7), we have

$$1 = ra^2 + sb^2 + 2aE + 2Fb + R, \tag{10}$$

$$1 = rc^2 + sd^2 + 2cG + 2dH + S, \tag{11}$$

$$0 = rac + sbd + cE + dF + aG + bH + T \tag{12}$$

Completing the squares in the first two equations gives

$$(\sqrt{r}a + \frac{E}{\sqrt{r}})^2 + (\sqrt{s}b + \frac{F}{\sqrt{s}})^2 = 1 - R + \frac{E^2}{r} + \frac{F^2}{s} \equiv P$$

$$(\sqrt{r}c + \frac{G}{\sqrt{r}})^2 + (\sqrt{s}d + \frac{H}{\sqrt{s}})^2 = 1 - S + \frac{G^2}{r} + \frac{H^2}{s} \equiv Q \tag{13}$$

This gives the four equations in Equation 9. Equation 12 gives the relation between $\theta, \bar{\theta}$: Substituting the four equations in Equation 9 into Equation 12 and collecting terms, we have

$$PQ\cos\theta\cos\bar{\theta} + PQ\sin\theta\sin\bar{\theta} - \frac{GE}{r^2} - \frac{FH}{s^2} + T = 0,$$

which implies that

$$\cos(\theta - \bar{\theta}) = \cos\theta\cos\bar{\theta} + \sin\theta\sin\bar{\theta} = \frac{\frac{GE}{r^2} + \frac{FH}{s^2} - T}{PQ}. \tag{14}$$

The two propositions together provide us with an efficient way to define the three-dimensional search space $\mathbb{S}$ mentioned earlier. For a given fixed $\mathbb{S}$, the complexity of the algorithm depends on the matching cost function. For example, Equation 1 gives a quadratic time complexity algorithm since it requires the computations of pairwise distances. To further reduce the time complexity in our current setup, a computationally less expensive matching cost function is required.

3.3   Matching Cost Function Using Local Moments

Inspired by the recent works on using moments for registration [7, 20], we proposed a simple and efficient matching cost function constructed directly from local polynomial moments. The (square) region containing the target point set $\mathbb{Q}$ is divided into four subregions R_1, R_2, R_3, R_4. We compute a vector descriptor $\mathbb{D}_{\mathbb{Q}}^n(R_i)$ for each region R_i using the averaged value of monomials of degree n

evaluated at the points of $\mathbb{Q}$ lying in R_i. For example, using the two linear mono-
mials, the descriptor $\mathbb{D}^1_{\mathbb{Q}}(R_i) = \frac{1}{N_i}[\sum_j x_j, \sum_j y_j]$. The sum is over all $q_i \in R_i$ and
N_i is the number of points in R_i. The two components are the linear moments,
and for quadratic moments, we have $\mathbb{D}^2_{\mathbb{Q}}(R_i) = \frac{1}{N_i}[\sum_j x_j^2, \sum_j x_j y_j, \sum_j y_j^2]$. In
general, for degree-n moments, the corresponding descriptor vector has $n+1$
components $\mathbb{D}^n_{\mathbb{Q}}(R_i) = \frac{1}{N_i}[\sum_j x_j^n, \cdots, \sum_j x_j^k y_j^{n-k}, \cdots, \sum_j y_j^n]$.

The descriptors can be defined for any point set, and in particular, we can
defined a matching cost function based on comparing these four local moments,
$\overline{\Gamma}_n(\mathbf{H}; \mathbb{P}, \mathbb{Q}) = \sum_{i=1}^{4} |\mathbb{D}^n_{\mathbb{Q}}(R_i) - \mathbb{D}^n_{\mathbf{H}(\mathbb{P})}(R_i)|^2$. Where $\mathbf{H}(\mathbb{P})$ denotes the image of
$\mathbb{P}$ under $\mathbf{H}$. For example, with linear moments, the matching algorithm will try
to match the center of mass of the points lie in each region.

Unfortunately, the linear moments, while very easy to compute, do not encode
much geometric information, and matching them provides the weakest constraint
for the registration. However, moments of higher-degree, such as quadratic and
cubic moments, offer several possibilities, and matching them in the least-squares
sense as above provide an informative and inexpensive way to compare the point
sets locally. We remark that, because the moments defined above are averaged
values, they are generally insensitive to the change in the number of points.
Finally, it is clear that without computing pairwise distances, the matching cost
function $\overline{\Gamma}_n$ has a linear time complexity.

3.4   Determine the Range for h_{31}, h_{32}

Ideally we would like to determine the range for h_{31}, h_{32} automatically from the
point sets. Unfortunately, we are not aware of any method that can accomplish
this. However, since our method is geared towards vision applications, we propose
to estimate the ranges empirically using two different methods. First, we gather
many pairs of images and manually select correspondences and compute the
homographies. The range for h_{31}, h_{32} can be determined directly from these
empirical data.

For the second method, we use the formula for homographies developed in [1]
(page 327). For two cameras with camera matrices $C_1 = K[\mathbf{I}\,|\,0]$, and $C_2 =
K'[\mathbf{R}\,|\,\mathbf{t}]$ and a plane π with homogeneous coordinates $[n^t\,d]^t$. The homography
matrix is given by the formula $H = K'(R - \mathbf{t}n^t/d)K^{-1}$. Where d is the distance
between the plane π and the first camera. We can sample with respect to different
matrices K, K' using different focal lengths and also different rotations.

Further simplifications can be done by assuming that $d \gg |t|$, and the
homography matrix is approximated as $H \approx \mathbf{K}'\mathbf{R}\mathbf{K}^{-1}$. This gives the infinite
homography between the two cameras [1]. In our simulation with $100,000$ sam-
ples, we randomly sample over a range of focal lengths (from 200 to 1000) and
rotation matrix $\mathbf{R}$, while keeping the cameras' skew to zero and principal points
at the origin. The simulation shows that less than 0.1% of h_{31}, h_{32} have absolute
values greater than one. This contrasts sharply with the results for other com-
ponents such as h_{11}, which has about 50% with magnitude greater than one.
The simulation shows that for applications that require registrations of point
sets using homographies that is close to infinite homographies, the search range
for h_{31}, h_{32} can be reasonable set to be between -5 and 5.

3.5   Refinement

Although the 1D reduction method proposed in sectin 3.2 make the exhaustive homography estimation possible, it consumes a significant amount of time in order to obtain a very precise homography estimation. Therefore, we propose to use looser intervals of h_{31}, h_{32} to have an approximate estimation first. The exhaustive homography estimation in this phase yields a new points set $\mathbb{P}' = \mathbf{H}(\mathbb{P})$ which is reasonably close to the target points set $\mathbb{Q}$. In the refinement phase, we take advantage of this geometry constraint to try to correspond a point $\mathbf{q}$ in $\mathbb{Q}$ only to points from $\mathbb{P}'$ in its local neighborhood.

To build the correspondences, we first compute the normalized orientation histograms of points in $\mathbb{Q}$ and $\mathbb{P}'$. These normalized histograms are used as local shape descriptor and used to compute the similartiy between a point in $\mathbb{Q}$ and its local neighboors in $\mathbb{P}'$. A normalized orientation histogram is computed as follows. For a point $\mathbf{p}$ in a points set $\mathbb{P}$, we apply Principle Component Analysis on it local neighborhood points set $\mathbb{N}$(circles in Figure 2), which is a subset of $\mathbb{P}$. The principle component vector $\boldsymbol{c}$ of the local neighborhood surrounding $\mathbf{p}$ give us the major direction (green lines in Figure 2) in which the points are spreaded. We then build the orientation histogram based on the angle between vector $\boldsymbol{c}$ and vector $\boldsymbol{p_j}$, where $\boldsymbol{p_j}$ is the vector formed by point $\mathbf{p}$ and the j-th point in $\mathbb{N}$. In our experiments, we used six bins for orientation histogram. Building the histogram based on the angle between the vector $\boldsymbol{p_j}$ and the principal component instead of x axis provides a rotationally invariant measurement between points in $\mathbb{Q}$ and $\mathbb{P}'$.

As shown in Figure 2, straight lines do not provide discriminative information for estimating homography. The points with histogram entropy values $\boldsymbol{e}$ less than 0.8 are omitted to speed up the correspondences building process. Entropy is calculated using $\boldsymbol{e} = -\Sigma_{i=1}^{6}\mathbf{h}(i)log(\mathbf{h}(i))$, where $\mathbf{h}$ is the normalized histogram.

After phase 1, points set $\mathbb{Q}$, and $\mathbb{P}' = \mathbf{H}(\mathbb{P})$ should be resonalbly close. Therefore the histogram of a point $\mathbf{q}$ in $\mathbb{Q}$ is only compared to its local neighbooring points $\mathbf{p}_i$ from $\mathbb{P}'$. The similarity of $\mathbf{q}$ and $\mathbf{p}_i$ is measured by using the L2 distance of their histograms. A point $\mathbf{p}_i$ is said to be corresponding to $\mathbf{q}$ if $i = \arg\min_{i}\|\mathbf{h_q} - \mathbf{h_{p_i}}\|$, where $\mathbf{h_q}$ and $\mathbf{h_{p_i}}$ are the histograms of $\mathbf{q}$ and $\mathbf{p}_i$, respectively. After obtaining a collection of correspondences, we utilize RANSAC to estimate the refined homography ($\mathbf{H}_r$). The matching cost function in the RANSAC process is the Hausdorff distance [21] between $\mathbb{Q}$ and $\mathbb{P}'$.

4   Experiments and Results

We applied our proposed method on the *points sets* collected from two different types of images:

Planar Shapes: In the first set of experiments, we drew several shapes on a typical classroom whiteboard. A pair of images were taken for each shape with different camera viewpoints (and focal lengths as well). Adaptive thresholding method [22] is applied to segment the images, and the point sets are then randomly sampled from the foreground pixels of resulting binary images. Three other pairs of images were taken from street scenes and are shown in the first
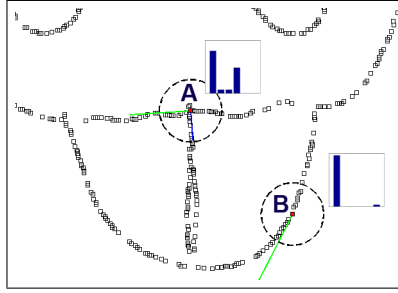
Fig. 2. The normalized orientation histogram of a point is built based on the angles between vectors p_j and its principle component c (green lines). The two bar charts are the normalized orientation histograms for points **A** and **B**, respectively. Points with low normalized histogram entropy value, point **B** for example, are omitted in the process of building correspondences.

three rows in Figure 4. Canny edge detector is applied to compute edges. Point sets were then randomly sampled from the resulting edges.

Complex scene: Instead of gathering points sets from the contours of planar shapes or simple hand-drawing shapes, we also applied SIFT [2] on the graffiti images (Figure 3) to gather points. We adjusted threshold values such that the number of SIFT points from the two images are similar. We then applied our algorithm on the gathered points sets to estimate the homography matrix.

We implemented the proposed algorithm using C++. We first search the optimal h_{31} and h_{32} from -5 to 5 with $\Delta d = 0.1$, $\Delta \theta = 1°$. Once an optimal pair of (h_{31}, h_{32}) has been found, we apply the refinement algorithm on the points sets $\mathbb{Q}$ and $\mathbf{H}(\mathbb{P})$. We define local neighborhood to be a range of radius 25 pixels (circles in Figure 2) around a point for calculating orientation histogram and for building correspondences between $\mathbb{Q}$ and $\mathbf{H}(\mathbb{P})$.

4.1   Results

Our experiments were conducted using a computer with an Intel T7300 2.0 GHz Core 2 Duo processor. The average time taken for estimating the homography in phase 1 are 33.51 seconds. The fourth column in Figure 4 and Figure 3 demonstrates the comparisons of the points sets $\mathbb{Q}$ and $\mathbf{H}(\mathbb{P})$. The first row of Table 1 shows the errors (Hausdorff distance [21]) in pixel between $\mathbb{Q}$ and $\mathbf{H}(\mathbb{P})$ (with images size 800×600). The average time taken in phase 2 - the refined homography estimation is 43.20 seconds. The last column in Figure 4 demonstrates the comparisons of points set $\mathbb{Q}$ and $\mathbf{H}_r(\mathbf{H}(\mathbb{P}))$, where $\mathbf{H}_r$ is the refined homography estimation obtained in phase 2. These images clearly demonstrate that the proposed method in phase 2 is able to obtain a more accuate estimation. The results in the third row of Table 1 also suggest a siginificant imporvement.

Table 1. The errors in pixels after phase 1 and 2 of the points sets listed in Figure 4.

	RCY	TRN	DSB	ARW	FOX	CAR	HUS	PCM	FSH
phase 1	10.49	10.47	13.25	7.85	6.77	5.49	8.92	6.30	9.86
phase 2	4.26	5.25	6.93	1.30	1.72	2.76	3.00	1.60	2.29

5 Conclusion and Future Work

We have proposed a novel registration algorithm for matching 2D point sets related by an unknown planar projective transform. From the two input point sets and the specified range for the two non-affine components of the homography matrix, the proposed algorithm defines a discrete three-dimensional space of homographies that can be searched exhaustively for the optimal homography transform. We also showed that a matching cost function can be constructed using local polynomial moments, and the resulting registration algorithm has linear time-complexity with respect to the size of the point sets.

To increase the efficiency, we chose a slightly coarse search range for h in phase 1. The homography estimated in phase 1 was able to efficiently bring the two points sets reasonably close to each other. Utilizing the orientation histograms, we were able to identify feature points and to build the correspondences between $\mathbb{Q}$ and $\mathbb{P}'$. The homography between these two points sets was then estimated using RANSAC. Preliminary experimental results are encouraging, and they have shown the proposed algorithms are indeed capable of efficiently and accurately estimating the projective transform directly from the point sets without other extra inputs such as intensity values.

We plan to further investigate in the new direction we have proposed in this paper. In particular, we will investigate in more details the sensitivity of the algorithm with respect to the sampling intervals $(\Delta d, \Delta \theta)$, and also the possibility of a GPGPU implementation since the code can be highly parallelizable. However, the most pressing unresolved issue is the question of whether it is possible to determine, in some meaningful way, the range for the two non-affine components directly from the input point sets. Perhaps the infinite projective invariants proposed in [6] can provide some clues on how to solve this difficult problem.
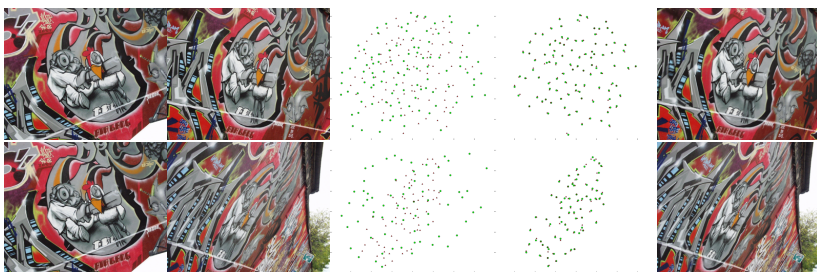

Fig. 3. The original pairs of images are shown in the first two columns. Column 3 displays the comparisons of the point sets of two images. Column 4 shows the comparisons of the transformed points set and the points of the images in second column. Column 5 shows the blending of the transformed image of the first column and the image in the second column.

References

1. Hartley, R., Zisserman, A.: Multiple View Geometry in Computer Vision, second ed. Cambridge University Press (2003)
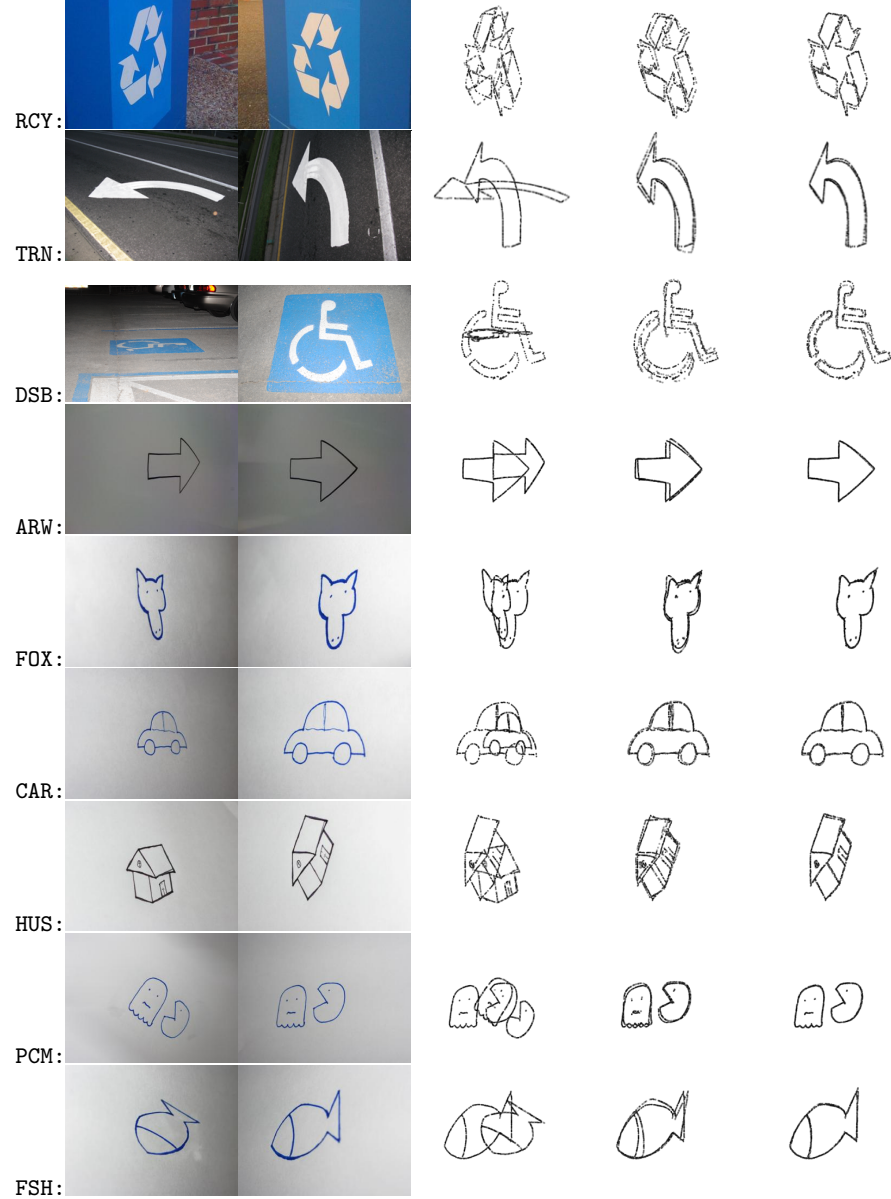
Fig. 4. The first two columns are the images which the points sets are sampled from. Column 3 demonstrates the comparisons of point sets $\mathbb{Q}$ and $\mathbb{P}$. Column 4 demonstrates the results the exhaustive homography estimation ($\mathbb{Q}$ and $\mathbf{H}(\mathbb{P})$). Column 5 demonstrates the results after the refinement step ($\mathbb{Q}$ and $\mathbf{H_r}(\mathbf{H}(\mathbb{P}))$).

2. Lowe, D.: Distinctive image features from scale-invariant keypoints. Int. J. Computer Vision **60** (2004) 91–110
3. Nemeth, J., Domokos, C., Kato, Z.: Recovering planar homographies between 2d shapes. In: Proc. Int. Conf. on Computer Vision. (2009) 889–892
4. 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** (1981) 381–395
5. van Gool, L., Moons, T., Pauwels, E., Oosterlinck, A.: Vision and Lies approach to invariance. Image and Vision Computing **13** (1995) 259–277
6. Flusser, J., Suk, T.: Projective moment invariants. IEEE Transactions on Pattern Analysis and Machine Intelligence **26** (2004) 13641367
7. Domokos, C., Kato, Z., Francos, J.M.: Parametric estimation of affine deformations of binary images. In: In Proc. of Int. Conf. on Acoustics, Speech, and Signal Processing. (2008) 889–892
8. Baird, H.: Model-based image matching using location. MIT Press (1984)
9. Besl, P., Jain, R.: Three dimensional object recognition. ACM Computing Surveys **17** (1985) 75–145
10. Grimson, E.: Object recognition by computer: The role of geometric constraints. MIT Press (1990)
11. Hinton, G., Williams, C., Revow, M.: Adaptive elastic models for hand-printed character recognition. In: Advances in Neural Information Systems. (1992) 512–519
12. Huttenlocher, D., Klanderman, G., Rucklidge, W.: Comparing images using the Hausdorff distance. IEEE Transactions on Pattern Analysis and Machine Intelligence **15** (1993) 850–863
13. Feldmar, J., Ayache, N.: Rigid, affine and locally affine registration of free-form surfaces. Int. J. Computer Vision **18** (1996) 99–119
14. Metaxas, D., Koh, E., Badler, N.: Multi-level shape representation using global deformation and locally adaptive finite elements. Int. J. Computer Vision **25** (1997) 49–61
15. Hofmann, T., Buhmann, J.: Pairwise data clustering by deterministic annealing. IEEE Transactions on Pattern Analysis and Machine Intelligence **19** (1997) 1–14
16. Cross, A., Hancock, E.: Graph matching with a dual-step EM algorithm. IEEE Transactions on Pattern Analysis and Machine Intelligence **20** (1998) 1236–1253
17. Gold, S., Rangarajan, A., Liu, C., Pappu, S., Mjolsness, E.: New algorithms for 2d and 3d point matching: pose estimation and correspondence. Pattern Recognition **31** (1998) 1019–1031
18. Chui, H., Rangarajan, A.: A new point matching algorithm for non-rigid registration. Computer Vision and Image Understanding **89** (2003) 114–141
19. Cao, Y., Miller, M., Winlow, R., Younes, L.: Large deformation diffeomorphic metric mapping of vector fields. IEEE Trans. on Med. Img. **24** (2005) 1216–1230
20. Ho, J., Peter, A., Rangarajan, A., Yang, M.H.: An algebraic approach to affine registration of point sets. In: Proc. Int. Conf. on Computer Vision. (2009) 1335–1340
21. Huttenlocher, D., Klanderman, G., Rucklidge, W.: Comparing images using the Hausdorff distance. IEEE Transactions on pattern analysis and machine intelligence (1993) 850–863
22. Ridler, T., Calvard, S.: Picture thresholding using an iterative selection method. SMC **8** (1978) 629–632