A Non-Rigid Point and Normal Registration Algorithm with Applications to Learning from Demonstrations

Alex X. Lee Max A. Goldstein Shane T. Barratt Pieter Abbeel

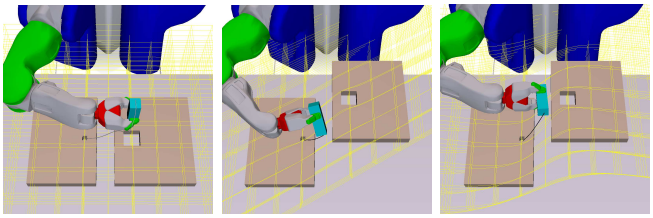
***Abstract*— Recent work [1], [2], [3] has shown promising results in learning from demonstrations for the manipulation of deformable objects. Their approach finds a non-rigid registration between points in the demonstration scene and points in the test scene. This registration is then extrapolated and applied to the gripper motions in the demonstration scene to obtain the gripper motions for the test scene. If more than one demonstration is available, a quality score for the non-rigid registration is used to determine the best matching training scene. For many manipulation tasks, however, the gripper's direction of approach with respect to the objects' surface normals is important in order to succeed at the task. This prior work only registers points across scenes and does not register the surface normals, often leading to warps between scenes that are inappropriate for transfer of manipulation primitives. The main contributions of this paper are (i) An algorithm for non-rigid registration that considers both points and normals, and (ii) An evaluation of this registration approach in the context of learning from demonstrations for robotic manipulation. Our experiments, which consider an insertion task in simulation and also knot-tying and towel-folding executions in a PR2, show that incorporating normals results in improved performance and qualitatively better grasps.**

I. Introduction

Learning from demonstrations has the potential to simplify equipping a robot with new skills by simply having the robot watch a human perform the task. The critical challenge in learning from demonstrations is the ability to generalize the motions that were successful in the demonstration context to motions that will be successful in the new environments the robot is faced with.

Recent work [1], [2], [3] has shown that existing approaches for non-rigid registration can be successful at recovering how demonstration and test environments relate to each other. In particular, their line of work uses thin plate spline robust point-matching (TPS-RPM), proposed by Chui and Rangarajan [4], to register demonstration scene onto test scene, and then extrapolates this registration to transfer the demonstrated gripper or tool trajectory onto the test scene. Their experiments have shown good success rate for tasks such as knot-tying (where generalization is over configurations of the rope), picking up plates (where generalization is over the shape of the plate), and screwing on bottle-caps (where generalization is over bottle and cap shapes). Success in these domains depends heavily on the quality of the registration.

Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, CA, USA. {alexlee_gk@cs.berkeley, mag4@princeton, sbarratt@berkeley, pabbeel@cs.berkeley}.edu



(a) Demonstration scene (b) Test scene using TPS-RPM (c) Test scene using TPSN-RPM

Fig. 1: We evaluate and compare non-rigid registration methods used in trajectory transfer for an insertion task. The yellow grid visualizes the registration warp of the scene with respect to the demonstration scene. (a) The demonstration consists on inserting a needle, which is simulated as a box, into a hole in a pad. A PR2 grasps the needle, which initially lies at the left pad, and then rotates and moves it to the right to insert it through the hole in the right pad. (b) When TPS-RPM is used for non-rigid point registration, the warp of the registration is a shear, which impedes the robot from properly inserting the box. (c) However, when using TPSN-RPM, our non-rigid point and normal registration algorithm, the warp correctly bends at the grasp point and insertion hole, which allows the robot to accurately insert the box.

However, registration with TPS-RPM has its limitations. Consider, for example, the scene shown in Figure 1. Figure 1a shows the demonstration scene. Figure 1b shows a similar scene, but the pads have been shifted. The yellow grid visualizes the warping function found through TPS-RPM. The found warping function is not well aligned with the pads. The reason that such a misaligned warp was found is that TPS-RPM tries to find a function that has low bending energy (i.e., is close to affine) while also having small error on the points that are being registered (in this case the corners of the pad, and the corners of the rectangular hole in the pad). For manipulation purposes, however, alignment is really important. A suture needle, for example, might have to be inserted orthogonally to the pad, and any flattening otherwise straightening out of the pads would typically have to happen aligned with the pad. Intuitively what is missing from TPS-RPM is a term in the objective that ensures the registration matches up the normals. Indeed, including such a term in the registration results in the more desirable warp shown in Figure 1c.

There is a long history in the non-rigid registration literature that considers the incorporation of normals when finding a warping function for a given, fixed registration between scenes. This work includes [5], [6], [7]. However, no

existing work has considered the problem of jointly finding the registration and the warping function, while considering normals (i.e., the extension of TPS-RPM, or a TPS-RPM-like algorithm, to include normals).

The main contributions of this paper are: (i) We describe TPSN-RPM, which extends TPS-RPM to include normals in the objective; (ii) We experimentally study the advantage of TPSN-RPM over TPS-RPM in the context of learning from demonstrations for robotic manipulation.

II. Related Work

The two main areas of related work are non-rigid registration and learning from demonstrations.

Besl et al. [8] uses feature-based matching methods for registering point clouds. Chui and Rangarajan [4] provide a method for registering one point cloud onto another, namely TPS-RPM. There has been some recent work in using surface normals to find a better registration between point clouds. Makadia et al. propose an algorithm [9] which uses the correlations of the two extented Gaussian images (EGI) in the Fourier domain and applies several iterations of Iterative Closest Points (ICP). The algorithm uses normals to generate orientation histograms. These orientation histograms are then used to estimate the initial rotation of the target cloud from the source cloud. Rusinkiewicz et al. [10] design an algorithm that is a modification of ICP. They use normal-space-directed sampling to affect ICP. This turns out to improve convergence for nearly-flat meshes and scenes that involve small features, such as inscribed surfaces.

In the non-rigid registration literature, normals have been incorporated when finding a warping function but they assume a fixed and known registration. Bookstein and Green [7] approximate the normals by adding points along the normal direction very close to the normal site. Mardia et al. [5], [6] use a kriging framework in which they can represent the normals exactly. The form of our thin plate spline with normals follows from this work. The difference between their work and ours is that we consider the case of unknown correspondences.

In the field of learning from demonstrations, also known as programming by demonstrations, expert demonstrations are generalized to new scenarios [11], [12], [13]. In particular, Calinon et al. [14], [15] developed a method for learning to perform manipulation tasks in different starting scenes. Their approach learns a mixture of Gaussians to represent the joint distribution of the robot trajectory and the environment state across multiple demonstrations, and then infer the trajectory for the new environment state conditioned on that state. They assume access to a featurization of the environment, so their approach is not directly applicable in tasks without a fixed environment representation.

In contrast, the approach of Schulman el al. [1] to learning from demonstrations can work directly with point clouds from the environment's scene. They first find a non-rigid registration that maps points of the demonstration scene to points of the new scene, and then extrapolate the registration function to the gripper motions to get new gripper motions for the new scene. Trajectory optimization [16] can then be used to find a feasible trajectory in which the grippers follow these new gripper motions. Schulman et al. [2] also applies this approach for a simplified suturing scenario. They had to artificially add points along the normals of the pads they were suturing in order to get a registration that would produce the correct angle of attack for the robot's end-effector. In this paper, however, we can incorporate this directly by mapping normals between both scenes. Lee et al. [3] extends Schulman et al.'s approach by jointly optimizing the non-rigid registration and the trajectory optimization in a single optimization.

III. Background

In this section we review the TPS-RPM algorithm for non-rigid point registration [17] and Schulman et al.'s approach to learning from demonstrations for robotic manipulation [1]. The idea of this approach is to first find a warping function that maps points in the demonstration scene to points in a new scene and then apply that function to the end-effector motion. This has the effect that the resulting end-effector motion incorporates variations in the new scene, which is important in manipulation tasks.

A. Non-Rigid Point Registration

Consider the case in which the demonstration scene consists of N points $\mathbf{X} = [\mathbf{x}_1 \dots \mathbf{x}_N]^\top$ and the test scene consists of N' points $\mathbf{Y} = [\mathbf{y}_1 \dots \mathbf{y}_{N'}]^\top$. The non-rigid registration problem is to find a function $f : \mathbb{R}^D \to \mathbb{R}^D$ that maps points $\mathbf{X}$ to points $\mathbf{Y}$ for some dimension D. In this paper we consider problems in 2 and 3 dimensions. To find a smooth mapping, we restrict the function f to be a thin plate spline (TPS). We first present the problem for the case in which the correspondences between the points are known and then we present the more general case of unknown correspondences.

1) Non-Rigid Point Registration with Known Correspondences: Assume that $N = N'$ and that there are one-to-one known correspondences between points $\mathbf{x}_i$ and $\mathbf{y}_i$. The registration problem can be formalized as an optimization problem,

$$\min_f \sum_{i=1}^{N} ||\mathbf{y}_i - f(\mathbf{x}_i)||_2^2 + \lambda ||f||_{\text{TPS}}^2, \tag{1}$$

where the regularizer $||f||_{\text{TPS}}^2$ is the TPS energy function,

$$||f||_{\text{TPS}}^2 = \int d\mathbf{x} ||\mathrm{D}^2 f(\mathbf{x})||_{\text{F}}^2, \tag{2}$$

which is a measure of the curvature of the function f. The parameter λ controls the trade-off between matching correspondence points and the smoothness of the spline.

The minimizer f in Equation (1) can be expressed as an affine transformation defined by $\mathbf{B}$ and $\mathbf{c}$, plus a weighted sum of basis functions $\sigma(\cdot)$ centered around the data points [18],

$$f(\mathbf{x}) = \sum_i \mathbf{a}_i \sigma(\mathbf{x} - \mathbf{x}_i) + \mathbf{B}^\top \mathbf{x} + \mathbf{c}, \tag{3}$$

where $\mathbf{a}_i$ is constrained by $\sum_i \mathbf{a}_i^d \mathbf{x}_i^d = 0$ and $\sum_i \mathbf{a}_i^d = 0$ for each dimension $d = 1, ..., D$. As shown in [19], Equation (1) can be efficiently solved analytically.

2) Non-Rigid Point Registration with Unknown Correspondences: When the correspondences between the points are unknown, we need to find a correspondence matrix $\mathbf{M}$, whose elements m_{ij} indicates if point $\mathbf{x}_i$ corresponds to point $\mathbf{y}_j$. If we allow for fuzzy correspondences, $\mathbf{M}$ is a doubly-stochastic matrix where each term m_{ij}, in the interval $[0, 1]$, indicates the degree of correspondence between point $\mathbf{x}_i$ and $\mathbf{y}_j$. An extra $N+1$th row and an extra $N'+1$th column is added to $\mathbf{M}$ to handle outliers. Chui and Rangarajan formulate this problem as a joint optimization [17],

$$\begin{aligned} \min_{f,\mathbf{M}} \quad & E(f, \mathbf{M}; T, \zeta) + \lambda ||f||^2_{\text{TPS}} \\ \text{subject to} \quad & \sum_{i=1}^{N+1} m_{ij} = 1, \sum_{j=1}^{N'+1} m_{ij} = 1, m_{ij} \geq 0, \end{aligned} \tag{4}$$

where

$$\begin{aligned} E(f, \mathbf{M}; T, \zeta) = & \sum_{i=1}^{N} \sum_{j=1}^{N'} m_{ij} ||\mathbf{y}_j - f(\mathbf{x}_i)||_2^2 \\ & + T \sum_{i=1}^{N} \sum_{j=1}^{N'} m_{ij} \log m_{ij} - \zeta \sum_{i=1}^{N} \sum_{j=1}^{N'} m_{ij}. \end{aligned} \tag{5}$$

The parameter T is referred to as a temperature and it controls the fuziness of the correspondences, with a lower temperature specifying correspondences that are less fuzzy and more binary. The parameter ζ prevents all the points from being assigned to outliers.

This optimization problem can be solved via coordinate descent, iterating between optimizing for $\mathbf{M}$ and for f. Since the objective is non-convex, the solution is only guaranteed to be locally optimal. Minimizing with respect to $\mathbf{M}$, gives the update

$$m_{ij} \propto \exp\left(-\frac{1}{T} ||\mathbf{y}_j - f(\mathbf{x}_i)||^2\right), \tag{6}$$

which needs to be followed by iterated row and column normalization. Given these correspondences, f can be minimized by solving for the optimal thin plate spline,

$$f = \arg\min_f \sum_{i=1}^{N} \mathbf{w}_i ||\bar{\mathbf{y}}_i - f(\mathbf{x}_i)||_2^2 + \lambda ||f||^2_{\text{TPS}}, \tag{7}$$

where

$$\mathbf{w}_i = \sum_{j=1}^{N'} m_{ij}, \qquad \bar{\mathbf{y}}_i = \frac{\sum_{j=1}^{N'} m_{ij} \mathbf{y}_j}{\mathbf{w}_i}. \tag{8}$$

The TPS-RPM algorithm consists of embedding the coordinate descent procedure in deterministic annealing. The algorithm alternates between estimating correspondences and fitting the optimal thin plate spline based on these estimated correspondences, while also gradually reducing the temperature T. The TPS-RPM algorithm is summarized in Algorithm 1.

Algorithm 1 TPS-RPM

```
procedure TPS-RPM(T_0, T_f, λ_0, λ_f)
    f ← identity transformation
    T ← T_0, λ ← λ_0
    repeat
        repeat
            Update point correspondences M using (6)
            Update transformation f using (7)
        until convergence
        decrease T and λ
    until T < T_f, λ < λ_f
end procedure
```

B. Learning from Demonstrations with Thin Plate Splines

In Schulman et al.'s work on learning from demonstrations [1], a demonstration consists of a point cloud $\mathbf{X}$ of the demonstration scene and an end-effector trajectory. At test time, a point cloud $\mathbf{Y}$ of the test scene is observed and the TPS-RPM algorithm is used to first find a non-rigid registration f that maps points from the demonstration scene to the new test scene. Then, the registration function is applied to the demonstration trajectory to get a new end-effector trajectory. This trajectory does not incorporate collision avoidance and joint limits, so trajectory optimization [16] is then used to find a feasible joint angle trajectory. The hope is that the resulting trajectory will incorporate variations in the environment and thus succeed in performing the desired manipulation. In practice, this method has been shown to be effective at generalizing expert demonstrations to new, unseen scene configurations.

IV. Non-rigid Point and Normal Registration

As we saw in Section III-A, a thin plate spline can be used to map points $\mathbf{X}$ in the demonstration scene to points $\mathbf{Y}$ in the test scene. In object manipulation tasks, the direction in which the end-effector approaches the object with respect to its surface normals is often important. For this reason, we want the registration function to also register normals from the demonstration scene to the test scene. Assume the demonstration scene has K normals $\mathbf{U} = [\mathbf{u}_1 \cdots \mathbf{u}_K]^\top$ each of them situated at respective sites $\mathbf{S} = [\mathbf{s}_1 \cdots \mathbf{s}_K]^\top$ in space, and the test scene has K' normals $\mathbf{V} = [\mathbf{v}_1 \cdots \mathbf{v}_{K'}]^\top$ at respective sites $\mathbf{T} = [\mathbf{t}_1 \cdots \mathbf{t}_{K'}]^\top$. The normal sites do not need to conincide with any of the points in the scene. In here we address the non-rigid point and normal registration problem which consists of finding a registration function $f : \mathbb{R}^D \to \mathbb{R}^D$ that maps points $\mathbf{X}$ to points $\mathbf{Y}$ and also maps normals $\mathbf{U}$ at sites $\mathbf{S}$ to normals $\mathbf{V}$ at sites $\mathbf{T}$.

A. Non-Rigid Point and Normal Registration with Known Correspondences

Since normals are differential quantities, a transformation f can be applied to a normal $\mathbf{u}$ at position $\mathbf{s}$ by multiplying the normal by $\mathbf{J}_f^{\mathbf{s}}$, the Jacobian of f at $\mathbf{s}$:

$$\mathbf{v} = \mathbf{J}_f^{\mathbf{s}} \mathbf{u}. \tag{9}$$

The point and normal registration can be formalized as an optimization problem,

$$\min_f \sum_{i=1}^{N} ||\mathbf{y}_i - f(\mathbf{x}_i)||_2^2 + \sum_{k=1}^{K} ||\mathbf{v}_k - \tfrac{1}{\beta_k}\mathbf{J}_f^{\mathbf{s}_k}\mathbf{u}_k||_2^2 + \lambda||f||_{\text{TPS}}^2, \tag{10}$$

where β_k is a normalization coefficient that will be explained at the end of this section.

Mardia [5] showed that the minimizer f in Equation (10) can be expressed with the same terms as the thin plate spline of Section III-A, plus a weighted sum of derivative terms of the basis functions,

$$f(\mathbf{x}) = \sum_{i=1}^{N} \mathbf{a}_i \sigma(\mathbf{x} - \mathbf{x}_i) - \sum_{k=1}^{K} \tilde{\mathbf{a}}_k \mathbf{u}_k^\top \nabla\sigma(\mathbf{x} - \mathbf{s}_k) + \mathbf{B}^\top \mathbf{x} + \mathbf{c}, \tag{11}$$

where $\mathbf{a}_i$ and $\tilde{\mathbf{a}}_i$ are now constrained by $\sum_i \mathbf{a}_i^d \mathbf{x}_i^d + \sum_k \tilde{\mathbf{a}}_k^d \mathbf{u}_k^d = 0$ and $\sum_i \mathbf{a}_i^d = 0$ for each dimension d. The basis function is of the form $\sigma(r) = ||r||^{2\alpha}$, with $1 < \alpha < 2$ in order for the function to accomodate the derivative terms. The thin plate spline of Section III-A cannot always be used because it is not differentiable at zero in 3 dimensions. For this work, we use $\alpha = 1.5$, which corresponds to f being a cubic spline. In the literature, the term 'thin plate spline' is used for any spline that minimizes the TPS energy function of Equation (2), so we follow that convention and refer to this cubic spline as a thin plate spline or, to be more specific, a thin plate spline with normals (TPSN). As before, Equation (10) can also be efficiently solved analytically. See the Appendix for an explicit formulation of this optimization as a least-squares problem in matrix form.

It is easy to show that if two normal vectors $\mathbf{u}$ and $\mathbf{v}$ are normalized, then the norm of their difference is related to the angle θ between them,

$$||\mathbf{v} - \mathbf{u}||_2^2 = 2(1 - \cos\theta), \tag{12}$$

with the cost being zero when the normals perfectly match. In the case of the normals cost of Equation (10), $\mathbf{v}_k$ can be normalized but the warped normal $\mathbf{J}_f^{\mathbf{s}_k}\mathbf{u}_k$ cannot since the Jacobian $\mathbf{J}_f^{\mathbf{s}_k}$ depends on the registration function being optimized. However, we can estimate a normalization coefficient β_k by first finding a registration function $\tilde{f}$ that solves a similar problem and then,

$$\beta_k = ||\mathbf{J}_{\tilde{f}}^{\mathbf{s}_k}\mathbf{u}_k||_2. \tag{13}$$

The problem that finds the registration function $\tilde{f}$ could be point registration without the normals information, or the problem from the previous iteration when this optimization is embedded in some iterative procedure as will be the case in Section IV-B.

B. Non-Rigid Point and Normal Registration with Unknown Correspondences

In the case of unknown correspondences, we now also need to find normals correspondences and these are defined by a correspondence matrix $\mathbf{Q}$ in the same way that the point correspondences are defined by $\mathbf{M}$.

Assuming that the points match the normal sites, $\mathbf{X} = \mathbf{S}$, a naïve approach would be to first use TPS-RPM to find point correspondences $\mathbf{M}$ and then use these for normal correspondences. However, besides the required assumption, this approach would find bad registration functions when normals don't have the same correspondences as their respective points. Consider the case in which a point from the demonstration scene corresponds well to a point from the test scene after using TPS-RPM, but the normals that lie at those sites differ by a lot. The fitted spline in this case is forced to match these normals and thus resulting in a poor warping function. In addition, this naïve procedure would not change the registration, which is what would be desired, as the normals in this naïve procedure don't affect the registration. Section V-A will illustrate this with 2-dimensional toy examples.

Similar to TPS-RPM, we find a registration function f and points and normals correspondences $\mathbf{M}$ and $\mathbf{Q}$ by solving the following optimization problem,

$$\begin{aligned} \min_{f,\mathbf{M},\mathbf{Q}} \quad & E(f, \mathbf{M}; T, \zeta) + \nu\tilde{E}(f, \mathbf{Q}; \tilde{T}, \tilde{\zeta}) + \lambda||f||_{\text{TPS}}^2 \\ \text{subject to} \quad & \sum_{i=1}^{N+1} m_{ij} = 1, \sum_{j=1}^{N'+1} m_{ij} = 1, m_{ij} \geq 0 \\ & \sum_{k=1}^{K+1} q_{kl} = 1, \sum_{l=1}^{K'+1} q_{kl} = 1, q_{kl} \geq 0, \end{aligned} \tag{14}$$

where

$$\begin{aligned} \tilde{E}(f, \mathbf{Q}; \tilde{T}, \tilde{\zeta}) = & \sum_{k=1}^{K}\sum_{l=1}^{K'} q_{kl}||\mathbf{v}_l - \tfrac{1}{\beta_k}\mathbf{J}_f^{\mathbf{s}_k}\mathbf{u}_k||_2^2 \\ & + \tilde{T}\sum_{k=1}^{K}\sum_{l=1}^{K'} q_{kl}\log\frac{q_{kl}}{\pi_{kl}} - \tilde{\zeta}\sum_{k=1}^{K}\sum_{l=1}^{K'} q_{kl}. \end{aligned} \tag{15}$$

Here, q_{kl} are the terms of $\mathbf{Q}$, π_{kl} is a prior probability that normals correspond to each other, ν is a parameter that trades off matching the normals, and $\tilde{T}$ and $\tilde{\zeta}$ are analogous to T and ζ but for normals correspondences.

The probability π_{kl} provides a prior that the kth source normal should match to the lth target normal. In our case, the prior takes the form,

$$\pi_{kl} \propto \exp\left(-\frac{1}{T}||\mathbf{t}_l - f(\mathbf{s}_k)||^2\right), \tag{16}$$

where we use the same temperture T that was used for the points. This prior discourages similar normals whose sites are far apart to correspond to each other.

Coordinate descent is used to find a locally optimal registration function and correspondences. Minimizing with respect to $\mathbf{M}$ gives the same update of Equation (6). Minimizing with respect to $\mathbf{Q}$ gives the update

$$q_{kl} \propto \pi_{kl}\exp\left(-\frac{1}{\tilde{T}}||\mathbf{v}_l - \tfrac{1}{\beta_k}\mathbf{J}_f^{\mathbf{s}_k}\mathbf{u}_k||^2\right), \tag{17}$$

which needs to be followed by iterated row and column normalization.

Given these correspondences, f can be minimized by solving for the optimal thin plate spline,

$$f = \arg\min_f \sum_{i=1}^{N} \mathbf{w}_i ||\bar{\mathbf{y}}_i - f(\mathbf{x}_i)||_2^2 + \nu \sum_{k=1}^{K} \tilde{\mathbf{w}}_k ||\bar{\mathbf{v}}_k - \tfrac{1}{\beta_k} \mathbf{J}_f^{\mathbf{s}_k} \mathbf{u}_k)||_2^2 + \lambda ||f||_{\text{TPS}}^2, \quad (18)$$

where

$$\tilde{\mathbf{w}}_k = \sum_{l=1}^{K'} q_{kl}, \qquad \bar{\mathbf{v}}_k = \frac{\sum_{l=1}^{K'} q_{kl} \mathbf{v}_l}{\tilde{\mathbf{w}}_k}. \quad (19)$$

We embed the coordinate descent procedure in deterministic annealing. The resulting algorithm is presented in Algorithm 2, which we name TPSN-RPM: thin plate spline with normals robust point-matching. The algorithm takes in initial and final values for the parameters $T, \lambda, \tilde{T}, \nu$ and returns a registration function f that maps points and normals from the demonstration scene to the test scene.

Algorithm 2 TPSN-RPM

procedure TPSN-RPM($T_0, T_f, \lambda_0, \lambda_f, \tilde{T}_0, \tilde{T}_f, \nu_0, \nu_f$)
  $f \leftarrow$ identity transformation
  $T \leftarrow T_0$, $\lambda \leftarrow \lambda_0$
  $\tilde{T} \leftarrow \tilde{T}_0$, $\nu \leftarrow \nu_0$
  repeat
    repeat
      $\beta_k \leftarrow ||\mathbf{J}_f^{\mathbf{s}_k} \mathbf{u}_k||_2$ (13)
      Update point correspondences $\mathbf{M}$ using (6)
      Update normal correspondences $\mathbf{Q}$ using (17)
      Update transformation f using (18)
    until convergence
    decrease T, λ and $\tilde{T}$, increase ν
  until $T < T_f, \lambda < \lambda_f, \tilde{T} < \tilde{T}_f, \nu > \nu_f$
  return f
end procedure

V. Experiments and Results

We illustrate our algorithm in the registration of 2-dimensional shapes and in learning from demonstrations for manipulation tasks in simulation and executions with a PR2.

A. 2-dimensional Toy Examples

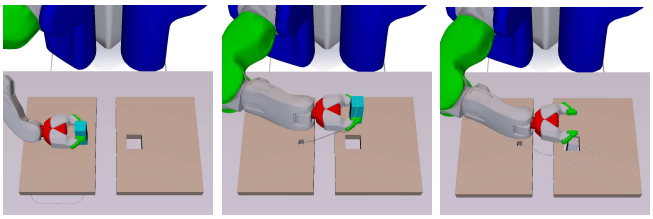
We qualitatively compare TPS-RPM and our algorithm TPSN-RPM for the registration of 2-dimensional shapes. The first example consists of a deformed vase and the second one consists of a square and circle whose position change relative to each other. The setup and the registrations of these examples are shown in Figure 2.

The points for the source and target points were chosen to be on the shape outlines. For the straight segments, the points were chosen such that they are evenly spaced within a distance of 2.5 units in the segment. The normal sites were chosen to be between adjacent points and its directions to be perpendicular to the edge in which they lie.

Notice that in the case of the deformed vase, the number of points and normals differ between the source and target. A reasonable registration should correspond the shape corners to each other. For the vase example, this means that a reasonable correspondence wouldn't be one-to-one.

For a fair comparison, the same final temperature T and bending energy parameter λ were used on both algorithms. The remaining parameters were found using grid search and choosing the ones that achieved the lowest energy on their respective objectives.

We can see that, unlike TPS-RPM, our algorithm TPSN-RPM was able to find a registration in which the corners match each other. Furthermore, the bending energy of the warp found by TPSN-RPM is lower. Notice that in TPSN-RPM, the normal lenghts are also matched to each other, which has the side effect of minimizing warp scaling in the direction of these normals, which might be desirable for manipulation tasks.



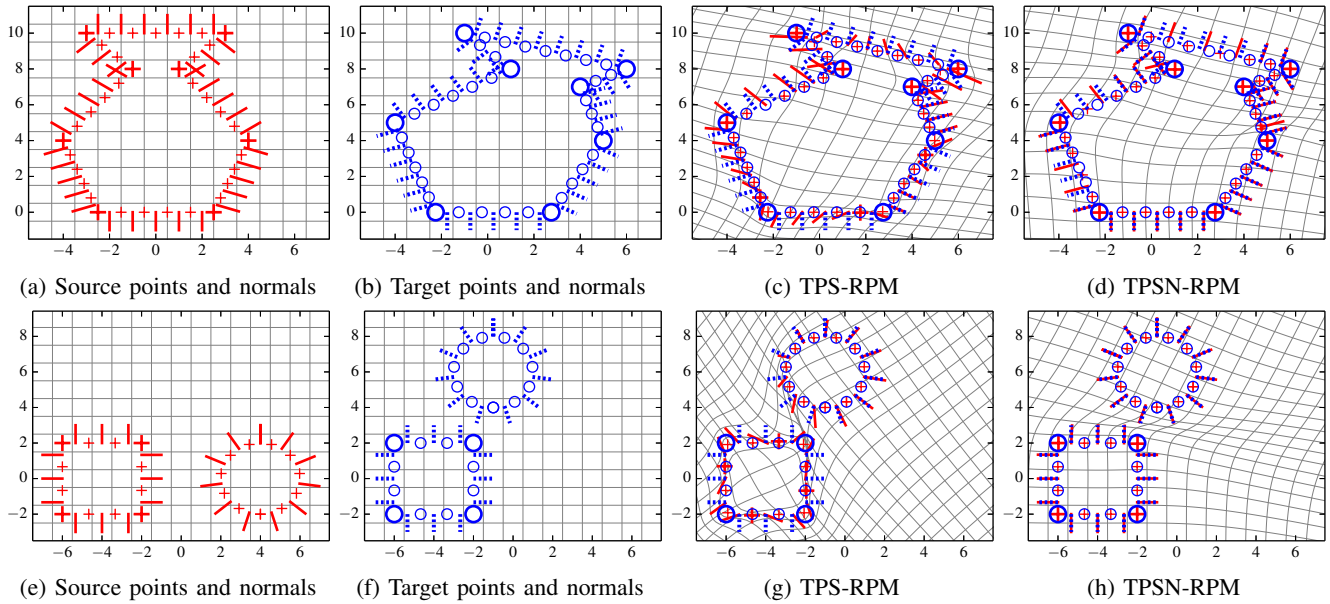
(a) (b) (c)

Fig. 3: Demonstration for a simplified needle insertion task. (a) The robot picks up the needle from the left pad. (b) The robot moves the needle into position to the target hole while dragging a thread. (c) The robot inserts the needle into the hole in the right pad.

B. Insertion Task with a Simulated PR2

We applied our registration algorithm for a learning from demonstration task involving autonomous insertion with a PR2 in a simplified simulation setting. The task involves the robot placing a needle into a hole in a pad. The needle is modeled as a box and the hole is slightly bigger than the box. We gave the robot a *single demonstration* of this task. In the demonstration, the gripper first picks up the box on the left, then rotates the box upside down as it moves from left to right, and then places the box in the hole at the right. The demonstration can be seen in Figure 3. The demonstration trajectory was programatically generated by specifying waypoints for the gripper's pose and interpolating them. We used points and normals around the initial position of the box and the target hole, as seen in Figure 4.

At test time, the robot generalized from the demonstration to execute the task in a new scene. Success was measured based on whether or not the robot was able to successfully place the box in the hole. The test scenes involved displacement of the target pad along an axis towards the robot, from



(a) Source points and normals (b) Target points and normals (c) TPS-RPM (d) TPSN-RPM

(e) Source points and normals (f) Target points and normals (g) TPS-RPM (h) TPSN-RPM

Fig. 2: Non-rigid registration of a deformed vase and rigid shapes using TPS-RPM and TPSN-RPM. The source points are denoted with red plus signs and the normals with red solid lines. The target points are denoted with blue circles and the normals with blue dashed lines. The warping of the space is visualized with the grid in the background. Points at the shape corners are drawn slightly bigger. Notice that in both examples, TPSN-RPM finds a registration in which the corners match each other while TPS-RPM fails to achieve this. In addition, TPSN-RPM aligns the normals for both point sets and finds a warping with a smaller bending energy.

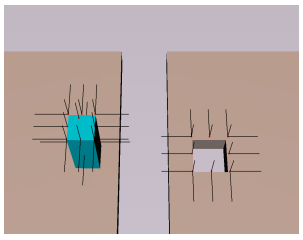


Fig. 4: The points and normals used in the simplified insertion task. The normals are at sites that coincide with the points and the normals are perpendicular to its adjacent faces. There are two or three orthogonal normals at each site.

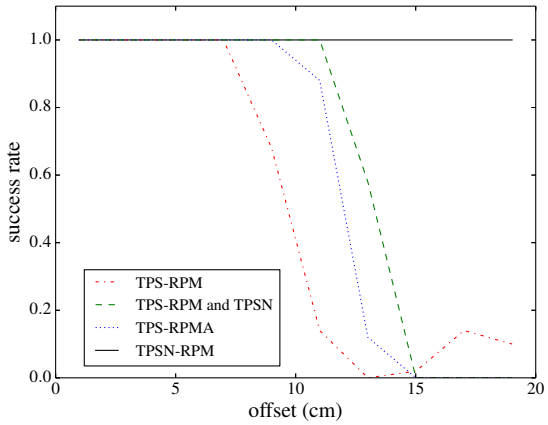


Fig. 5: Insertion success rate for various offset distances using TPS-RPM, TPS-RPM followed by TPSN, TPS-RPMA and our method TPSN-RPM.

0 cm to 20 cm offset. Hyperparameters were determined by performing a grid search for each offset and algorithm.

We compare the success of TPS-RPM, TPSN-RPM, and two variations of these for the insertion task. The first variation consists of first finding point correspondences using TPS-RPM and then using those for normal correspondences to fit a TPSN. The second variation, which we call TPS-RPMA, consists of approximating the normals by adding artificial points along the direction of the normals and then using TPS-RPM with the original and artificial points. This latter method was used by [7] and [2]. The results are summarized in Figure 5 and Table I.

Our algorithm TPSN-RPM outperformed all the other methods achieving 100% success rate for all offsets. At high offsets, the shearing resulting in registration by TPS-RPM caused the robot to insert the box improperly and thus fail at the task. The two variations was able to do better than TPS-RPM for intermediate offsets. Using TPS-RPM for correspondences followed by TPSN failed to find correct correspondences between normals because point correspondences do not neccessarily imply correct normal correspondences. TPS-RPMA wasn't as good as TPSN-RPM because while the latter fits the angle of normals, TPS-RPMA fits exact points in the direction of normals.

There is a tradeoff between success rate and average registration time. TPSN-RPM on average takes the most time to register. This is because the running time of these RPM

(a) Rope-tying using TPS-RPM

(b) Rope-tying using TPSN-RPM

(c) Towel-folding using TPS-RPM

(d) Towel-folding using TPSN-RPM

Fig. 6: Sequential executions of a PR2 tying overhand knots and folding towels on a curved surface. The demonstrations (not shown) were performed on a flat surface and the grippers approached the table in a direction roughly perpendicular to the table. Note that even though both approaches were able to succesfully complete the task, the grippers' approach direction are qualitatively different. When using TPS-RPM, the grippers' approach direction is vertical as in the demonstration, but when using TPSN-RPM, the direction is perpendicular to the new curvature of the table. For videos, see `http://rll.berkeley.edu/icra2015tpsn/`.

Algorithm	Mean Success Rate	Mean and Standard Deviation Registration Time
TPS-RPM (Algorithm 1)	50.8%	0.040 ± 0.004 s
TPS-RPM and TPSN	65.8%	0.046 ± 0.005 s
TPS-RPMA	60%	0.522 ± 0.021 s
TPSN-RPM (Algorithm 2)	100%	0.743 ± 0.035 s

TABLE I: Success rates of a simulated insertion through a pad using a different approaches for non-rigid registration. The TPS-RPM algorithm jointly finds a point registration and a transformation function that map points. The TPS-RPM and TPSN approach first uses TPS-RPM and then uses the point registration in order to fit a transformation function that map points and normals. The TPS-RPMA approach tries to register normals by having additional artificial points in the direction of the normals and then uses TPS-RPM with the original and artificial points to find a registration and transformation function that maps these points. Our algorithm, TPSN-RPM, jointly find a point registration, a normal registration, and a transformation function that map points and normals.

algorithms are cubic in the number of variables and TPSN-RPM has additional weight variables $\tilde{\mathbf{a}}_k$ for the normal basis. However, since the normals provide surface information that would otherwise be provided by dense points, less points could be used in exchange of some normals and thus TPSN-RPM could have a comparable number of variables.

C. Knot-tying and towel-folding with a PR2

We also applied learning from demonstrations for knot-tying and towel-folding executions with a PR2. For the knot-tying task, we used the library of demonstrations containing 148 demonstrations collected by Schulman et al. [1]. For the towel-folding task, we used a library of 3 demonstrations, one for each segment of the task (fold in half, fold one third, and fold the last third). In both of these demonstrations, the demonstrations were recorded using kinesthetic teaching for the trajectory and a RGBD camera for the points clouds. The demonstration scenes consisted of a red rope and towel on a green flat table.

The points were those of the object, which were extracted with a red filter. The normal sites were table points that were within 5 cm of any object point. The table points were extracted with a green filter. The surface normal at each of those sites was computed using principal component analysis (PCA) with a window radius of 5 cm. The rope and towel points were downsampled using a voxel size of 5 cm and 2.5 cm respectively and the table normals were downsampled using a voxel size of 5 cm for both tasks.

The test scenes consisted of rope and towel in a different configuration on a green curved table. The shape of the curved table is the top of a sinusoid with an amplitude of 22.86 cm and a half period of 91.44 cm. The autonomous executions of both tasks using TPS-RPM and TPSN-RPM are shown in Figure 6. Videos of these executions can also be seen at `http://rll.berkeley.edu/icra2015tpsn/`. Both methods were able to succeed at knot-tying and towel folding. Despite of this, we can see that the grippers' approach direction to the table is qualitatively better and perpendicular to the curved table when TPSN-RPM is used. In other tasks in which normals are more important, such as in surgical suturing, TPSN-RPM could improve the insertion of a needle into suturing pads.

VI. Conclusion and Future Work

We presented TPSN-RPM which extends TPS-RPM to include normals in the objective. We introduced a second correspondence matrix to keep track of correspondence between normals, and the transformation update now solves an optimization problem that considers normals.

For various manipulation tasks, the robot gripper's direction of approach with respect to the objects's surface normals is important to succeed at the task. We showed that our method results in improved performance for a simulated

insertion task and in qualitatively better grasps for knot-tying and towel-folding tasks when executed in a PR2.

In the future, we would like to apply our method in real-world tasks in which normal information is critical for the success of the task, such as in surgical suturing. In addition, we would like to investigate effective ways to choose normals since only some normals might be important for a given task.

VII. Acknowledgments

Alex Lee was funded in part by an NSF Graduate Research Fellowship. This research was funded in part by the AFOSR through a Young Investigator Program award and by the NSF NRI program under award #1227536.

References

[1] J. Schulman, J. Ho, C. Lee, and P. Abbeel, "Learning from Demonstrations through the Use of Non-Rigid Registration," in *Proceedings of the 16th International Symposium on Robotics Research (ISRR)*, 2013.

[2] J. Schulman, A. Gupta, S. Venkatesan, M. Tayson-Frederick, and P. Abbeel, "A Case Study of Trajectory Transfer Through Non-Rigid Registration for a Simplified Suturing Scenario," in *Proceedings of the 26th IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[3] A. X. Lee, S. H. Huang, D. Hadfield-Menell, E. Tzeng, and P. Abbeel, "Unifying Scene Registration and Trajectory Optimization for Learning from Demonstrations with Application to Manipulation of Deformable Objects," in *Proceedings of the 27th IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2014.

[4] H. Chui and A. Rangarajan, "A new algorithm for non-rigid point matching," in *CVPR*, 2000, pp. 2044–2051.

[5] C. G. K.V. Mardia, J.T. Kent and J. Little, "Kriging and splines with derivative information," in *Biometrika*, 1996, pp. 207–221.

[6] K. Mardia and J. Little, "Image warping using derivative information," in *SPIE*, 1994.

[7] F. L. Bookstein and W. D. Green, "A feature space for edgels in images with landmarks," in *Journal of Mathematical Imaging and Vision 3*, 1993, pp. 231–261.

[8] P. Besl and N. McKay, "A method for registration of 3-d shapes," in *IEEE Trans. Patt. Anal. Mach. Intell*, 1992, pp. 14(2): 239–256.

[9] P. Makadia, A. and K. A, Daniilidis, "Fully automatic registration of 3d point clouds," in *IEEE*, 2006, pp. 1063–1304.

[10] S. Rusinkiewicz, "Efficient variants of the icp algorithm," in *IEEE*, 2001, pp. 145 – 152.

[11] A. Billard, S. Calinon, R. Dillmann, and S. Schaal, "Robot programming by demonstration," in *Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Secaucus, NJ, USA: Springer, 2008, pp. 1371–1394.

[12] B. D. Argall, S. Chernova, M. Veloso, and B. Browning, "A Survey of Robot Learning from Demonstration," *Robot. Auton. Syst.*, vol. 57, no. 5, pp. 469–483, May 2009.

[13] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in cognitive sciences*, vol. 3, no. 6, pp. 233–242, 1999.

[14] S. Calinon, F. Guenter, and A. Billard, "On Learning, Representing, and Generalizing a Task in a Humanoid Robot," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 37, no. 2, pp. 286–298, April 2007.

[15] S. Calinon, F. D'halluin, D. Caldwell, and A. Billard, "Handling of multiple constraints and motion alternatives in a robot programming by demonstration framework," in *Humanoid Robots, 2009. Humanoids 2009. 9th IEEE-RAS International Conference on*, Dec 2009, pp. 582–588.

[16] J. D. Schulman, J. Ho, A. Lee, I. Awwal, H. Bradlow, and P. Abbeel, "Finding Locally Optimal, Collision-Free Trajectories with Sequential Convex Optimization," in *Proceedings of Robotics: Science and Systems (RSS)*, 2013.

[17] H. Chui and A. Rangarajan, "A new point matching algorithm for non-rigid registration," *Computer Vision and Image Understanding*, vol. 89, no. 2-3, pp. 114–141, 2003.

[18] J. Duchon, "Splines minimizing rotation-invariant semi-norms in sobolev spaces," in *Constructive Theory of Functions of Several Variables*, ser. Lecture Notes in Mathematics, W. Schempp and K. Zeller, Eds. Springer Berlin Heidelberg, 1977, vol. 571, pp. 85–100. [Online]. Available: http://dx.doi.org/10.1007/BFb0086566

[19] G. Wahba, *Spline Models for Observational Data*. Philadelphia: Society for Industrial and Applied Mathematics, 1990.

Appendix

[Thin Plate Spines with Normals]

The thin plate splines with normals, in its general form, solves the optimization problem,

$$\min_f \sum_{i=1}^{N} ||\mathbf{y}_i - f(\mathbf{x}_i)||_2^2 + \nu \sum_{k=1}^{K} ||\mathbf{v}_k - \mathbf{J}_f^{\mathbf{s}_k}\mathbf{u}_k||_2^2 + \lambda ||f||_{\text{TPS}}^2. \tag{20}$$

The constant β_k of Equation (10) can easily be incorporated into $\mathbf{u}_k$. The minimizer f can be expressed as in Equation (11).

Let

$$\mathbf{\Sigma} = \begin{bmatrix} \mathbf{\Sigma}_{00} & \mathbf{\Sigma}_{01} \\ \mathbf{\Sigma}_{01}^\top & \mathbf{\Sigma}_{11} \end{bmatrix} \qquad \mathbf{D} = \begin{bmatrix} \mathbf{1} & \mathbf{X} \\ \mathbf{0} & \mathbf{U} \end{bmatrix}, \tag{21}$$

where

$$\begin{aligned}
(\mathbf{\Sigma}_{00})_{ij} &= \sigma(\mathbf{x}_i - \mathbf{x}_j) \\
&= ||\mathbf{x}_i - \mathbf{x}_j||^3 \\
(\mathbf{\Sigma}_{01})_{ij} &= -(\mathbf{u}_j^\top \nabla)\sigma(\mathbf{x}_i - \mathbf{s}_j) \\
&= 3||\mathbf{s}_j - \mathbf{x}_i||\mathbf{u}_j^\top(\mathbf{s}_j - \mathbf{x}_i) \\
(\mathbf{\Sigma}_{11})_{ij} &= -(\mathbf{u}_i^\top \nabla)(\mathbf{u}_j^\top \nabla)\sigma(\mathbf{s}_i - \mathbf{s}_j) \\
&= -3||\mathbf{s}_i - \mathbf{s}_j||(\mathbf{u}_i^\top \mathbf{u}_j) \\
&\quad - \frac{3}{||\mathbf{s}_i - \mathbf{s}_j||}(\mathbf{u}_i^\top(\mathbf{s}_i - \mathbf{s}_j))(\mathbf{u}_j^\top(\mathbf{s}_i - \mathbf{s}_j)).
\end{aligned} \tag{22}$$

The optimization problem of Equation (20) can be rewritten as a constrained least squares problem,

$$\begin{aligned}
\min_f \quad & ||\mathbf{Y} - [\mathbf{\Sigma}_{00}\ \mathbf{\Sigma}_{01}]\mathbf{A} - \mathbf{X}\mathbf{B} - \mathbf{1}\mathbf{c}^\top||_{\text{F}}^2 \\
& + \nu||\mathbf{V} - [\mathbf{\Sigma}_{01}^\top\ \mathbf{\Sigma}_{11}]\mathbf{A} - \mathbf{U}\mathbf{B}||_{\text{F}}^2 \\
& + \text{Tr}\,(\mathbf{A}^\top \mathbf{K}\mathbf{A}) \\
\text{subject to} \quad & \mathbf{D}^\top\mathbf{A} = \mathbf{0}_{4\times 3},
\end{aligned} \tag{23}$$

where $\mathbf{A}$ are the weights of the basis functions, $\mathbf{A} = [\mathbf{a}_1 \dots \mathbf{a}_N\ \tilde{\mathbf{a}}_1 \dots \tilde{\mathbf{a}}_K]^\top$ and the bending energy matrix $\mathbf{K}$ is given by

$$\mathbf{K} = ((\mathbf{I} - \mathbf{P})\mathbf{\Sigma}(\mathbf{I} - \mathbf{P}))^{-1} \tag{24}$$

$$\mathbf{P} = \mathbf{D}(\mathbf{D}^\top\mathbf{D})^{-1}\mathbf{D}^\top. \tag{25}$$