Preparation of Papers for IEEE Sponsored Conferences & Symposia*

Ankush Gupta[1], Dylan Hadfield-Menell[2], Robbie Gleichman, and Pieter Abbeel

***Abstract*— Very**

I. INTRODUCTION

A large challenge in applying standard manipulation and planning techniques to deformable object manipulation is that of tractable modelling. Deformable object are often characterized by high-dimensional, continuous state-action spaces. Model-based planning has yet to scale up to the task of efficient planning in this setting.

Recent work have gained traction on this problem through the technique of learning from demonstrations [5], [4]. These results are achieved through *trajectory transfer*, where a demonstration trajectory is generalized to fit a new scenario. Trajectory transfer finds a non-rigid registration is found an example scene and the current scene and that is used to modify the demonstration trajectory so that it better fits the scene at hand. This method of transfer is model-free and obviates the need to plan in complicated and intractable models of deformable objects. Trajectory transfer has demonstrated state-of-the-art performance for knot-tying and suturing.

An important aspect of these strategies is the incorporation of multiple demonstrations into the process. By increasing the amount of instruction, it becomes possible to do more tasks. Additionally, demonstrations can take the form of steps in a task and can be order or recombined to further increase the set of possible successful manipulations.

However, an important problem remains: how should we pick which trajectory to transfer from a library of options given an input scene? Incorrect selection may lead us to fail at a task which would otherwise be possible for the correct selection of trajectories.

Schulman et al. propose to use nearest-neighbors with registration cost (a measure of the goodness of fit for the registration) as a similarity measure to solve this problem. In this paper, we consider the problem of effectively learning to pick the trajectory to generalize from experience.

We frame this problem in terms of manifold learning with respect to the state space of our object. Given a manipulation task, m, and demonstration state-trajectory pair, d, t, there is region of state space, S, such that t will perform m successfully when transferred to any $s \in S$. In this framework, the nearest-neighbor selection rule represents S with the singlton d and chooses the trajectory which minimizes distance to an known example from S. Dylan: This is pretty sloppy, and probably too detailed for the intro but I'm including it so we have it down on paper. A natural way to extend this approach uses successful traces of trajectory transfer to draw new states from S.

The contributions of this paper are as follows: **(i)** we frame the problem of selecting a trajectory to transfer from a library as one of estimating distance to a manifold; **(ii)** we propose a method for building a model-free representation of the manifold associated with a demonstration from successful traces of execution; **(iii)** we show how this approach can be leveraged to improve finding correspondences with new scenes; **(iv)** we demonstrate the effectiveness of this approach by showing an improvement of Dylan: Number here over the nearest-neighbor selection strategy in a simulated knot-tying task.

Dylan: We need a name

Dylan: Figures: illustration of segment tree; Graphs/Table showing performance of NN/No Correspondence/Best C-Forest; Graph showing performance of C-Forest as fn of number of training examples; Illustration of manifold idea

II. RELATED WORK

- Reinforcement Learning
- Deformable Object Manipulation (particularly knot-tying)
- Manifold Learning
- Anything else?

III. TECHNICAL BACKGROUND

A. Trajectory Transfer

Trajectory transfer is an approach to learning from expert demonstrations [5]. The trajectory transfer algorithm is given a current scene, s_{test}, demonstration scene, s_{demo} and a demonstration trajectory, t_{demo}, as input. We assume that the scenes are made up of matched points in $\mathbb{R}^3$. The first step is to find a function, $f^* : \mathbb{R}^3 \to \mathbb{R}^3$, as the solution to the following optimization problem:

$$\min_f \sum_i ||s_{test}^{(i)} - s_{demo}^{(i)}||^2 + C \int dx ||D^2(f)||_{Frob}^2. \quad (1)$$

The minimizing f will be a Thin Plate Spline, and can be expressed as a linear combination of basis functions about the correspondence points [6]. C is a hyper-parameter that trades off between goodness of fit and the curvature of the function. The solution to this optimization can be computed as the solution to a linear system of equations.

Given a warping, f^*, between the demo and test scenes, we take each pose from the demo trajectory and pase it

*This work was not supported by any organization

[1]Albert Author is with Faculty of Electrical Engineering, Mathematics and Computer Science, University of Twente, 7500 AE Enschede, The Netherlands albert.author@papercept.net

[2]Bernard D. Researcheris with the Department of Electrical Engineering, Wright State University, Dayton, OH 45435, USA b.d.researcher@ieee.org

through f^*. Poses are transferred by mapping coordinate frames through the jacobian of f^*. The trajectory that results from this is used to guide a motion planner that finds a similar feasible trajectory. This trajectory is executed in the test scene. In the case where correspondences are not known initially, one can use TPS-RPM, an approach that jointly finds correspondences and a mapping between them by alternating between estimating correspondences and solving for a thin plate spline[2].

Schulman et al. [5] provide some intuition for scenarios where this approach is likely to succeed. They assume a cost function, L, on states and trajectories, a reasonable option might be 0-1 loss, depending on whether the trajectory successfully executes a desired manipulation in a given state. Then we can justify warping the state s and the trajectory t in the case where $L(s,t) = L(f(s), f(t))$. Essentially, the of a manipulation is preserved under a class of transformations, thus, we can successfully transform a state and trajectory and maintain the relation that the manipulation succeeds. The set of functions that have this property define a set of states that a particular demonstration trajectory can transfer to.

A final aspect of this approach is the incorporation of multiple trajectories. Given a library of trajectories, one can increase the number of states that can be generalized to. This allows an expert to demonstrate steps of a complex task which can be sequenced at test time. This can make trajectory transfer more robust and reliable, as an expert can also include demonstrations to recover from common failures. Current approaches use the nearest-neighbor with respect to registration cost (the value of the optimization problem defined in (1)). This corresponds to modeling the set of states a demonstration can generalize to—that is states for which L is invariant to the TPS warping found by TPS-RPM—as a hyper-sphere in a high-dimensional space where this registration cost is a distance function.

B. *Manifold Learning*

One way to characterize our approach is that of characterizing, for a particular demonstration, the manifold in state space such that the TPS warping of that demonstration will correctly executed the desired task. The field of *manifold learning* concerns itself with a similar problem: that of learning a low-dimensional representation of data. This low-dimensional representation is arrived at by assuming that it is drawn from a lower dimensional manifold.

Formally, a d-dimensional manifold, M, is a set of points such that a local neighborhood around any point can be mapped to $\mathbb{R}^d$ with a homomorphism [1]. Informally, this is usually a set of points in a high dimensional euclidean space that can be paramaterized with d values. Manifold learning attempts to learn these manifolds from data.

This has a broad range of applications and approaches. Principal Component Analysis solves this problem for the case where the manifold is a linear subspace [3]. For manifolds that are not linear, many approaches attempt to find an embedding in low dimensional euclidean space that preserves geodesic distances, distance measured along the manifold, between points. One approach, Isomap, approximates this geodesic distance by the euclidean distance to the nearest neighbor. The underlying assumption here is that locally, the geodesic distance on the manifold is well approximated by the euclidean distanceDylan: cite Isomap paper.

IV. Approach

In this section,

A. *Transfer Manifolds*

B. *Trajectory Selection and a Nearest-Manifold Query*

C. *Learning Segment Trees*

V. Experimental Setup

this is experiments

VI. Results

The section describes the experiments we did and the results we got.

VII. Conclusion and Future Work

This section summarizes the contributions and suggests some next steps.

APPENDIX

Appendixes should appear before the acknowledgment.

ACKNOWLEDGMENT

The preferred spelling of the word acknowledgment in America is without an e after the g. Avoid the stilted expression, One of us (R. B. G.) thanks . . . Instead, try R. B. G. thanks. Put sponsor acknowledgments in the unnumbered footnote on the first page.

References are important to the reader; therefore, each citation must be complete and correct. If at all possible, references should be commonly available publications.

References

[1] Lawrence Cayton. Algorithms for manifold learning. 2005.

[2] Haili Chui and Anand Rangarajan. A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding*, 89(2-3):114–141, 2003.

[3] Ian Jolliffe. *Principal component analysis*. Wiley Online Library, 2005.

[4] John Schulman, Ankush Gupta, Sibi Venkatesan, Mallory Tayson-Frederick, and Pieter 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.

[5] John Schulman, Jonathan Ho, Cameron Lee, and Pieter Abbeel. Learning from demonstrations through the use of non-rigid registration. In *Proceedings of the 16th International Symposium on Robotics Research (ISRR)*, 2013.

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