Proprioceptive Sensing in Soft-Robotic Arm

Shubhom Bhattacharya
Department of Electrical and Computer Engineering

Cornell University
Ithaca, United States
sb2287@cornell.edu

Patricia Xu
Department of Mechanical Engineering

Cornell University
Ithaca, United States
pax2@cornell.edu

***Abstract*—Soft robots utilize deformable materials for actuation. As a result, rigid body modeling fails to characterize soft robot motion, especially with greater degrees of freedom. Inspired by the biological faculty of proprioception, we propose treating this problem as a machine learning problem for sensing the robot's own movement. Specifically, we test this approach on a robotic arm that is composed of a mesh of optical lightguides. After trialing several different supervised regression algorithms, we find that using artifical neural networks (ANNs) empirically yields the lowest metric regression error and is able to successfully predit the arm's motion. Further work is planned to assess the learned model's performance on robots with additional degrees of freedom.**

***Index Terms*—soft robotics, machine learning, deformation sensing**

I. Introduction

Proprioception is the ability of many living organisms to sense body position and movement, enabling coordination independent of visual feedback [1]. For example, humans are able to reason about the location of their extremities without looking at them. Robots traditionally make use of rigid-body dynamics equations to approximate a similar sense, but relying on these methods restricts the complexity of robot motion. Soft robots are robots built with nonrigid materials that can add degrees of freedom to the robots motion, such as increased flexibility, expansion and contraction, and so on.

II. Prior research

While some work has been done in the application of machine learning to soft robotics, most control algorithms for soft robots rely on mechanical models that rely on constraining assumptions about the robot [2]. Meanwhile, autonomous sensing has made considerable progress as a result of advances in machine learning, leading to levels of autonomy previously not possible [3]. The emergence of machine learning therefore has great promise because data-driven approaches can be applied to high-dimensional, noisy sensing problems [2], making the extension of these methods to soft robotics intuitive and promising. However, one of the greatest challenges in using machine learning approaches is the need for vast amounts of unbiased data, which is difficult to achieve in experiments especially with high-variance signals with large dimensionalities. Carefully designed data collection and validated learning algorithms are therefore critical to determining the effectiveness of any machine-learning based sensing scheme.

In this project we used machine learning for proprioception sensory in soft robots. Tactile sensing and proprioception are important because they characterize an autonomous agent's interactions with surrounding objects, facilitating increasingly complex tasks [2]. While computer vision offers some semantic knowledge of the world, tactile sensing allows soft robots that can change shape and size added inference capability that more closely resembles a living organism's [1]. Studies such as [4] led us to believe that proprioception sensory with machine learning is a feasible undertaking [5]. For example, inorganic sensors are used to classify muscle fatigue in [6] and ridge regression modeling predicts robotic arm deformation angle in [7] . Furthermore, previous work in our group [8] found that machine learning could be used to analyze light intensity signals in sensory foams, and that waveguide position modeling without machine learning was not robust to the complexity of motion [9]. We thus believed that modeling similar signals in optical lightguide deformations would be possible,

III. Data Collection and Preprocessing

Data collection for machine learning algorithms is ideally unbiased and abundant. However, sampling from all combinations of continuous degrees of freedom of an arm is impossible, so discrete joint angles of and block positions were sampled instead. The angles were linearly spaced in the range -1.06 radians to 1.06 radians with intervals of 0.11 radians and the block positions were linearly spaced in the range 0 to 95 mm with intervals of 8.5 mm. These intervals were chosen to approximately mimic the range of motion of a real arm. To ensure consistency of measurement and minimize human error, a testing apparatus was built to move the arm to the desired position and rotation (Figure 1) . 10 trials were taken with each angle to reduce variance in the training set. The feature space of the data was the intensity measurement from 11 sensors and the goal of a candidate multiple regression algorithm was to map these 11 sensors to the corresponding joint angle and block position.

The machine learning problem was a multi-output regression, with two predicted variables (joint angle and block position) given 11 light intensity sensor readings for a given data sample. The output variables cannot be assumed to be independent because the light intensity for a given joint angle may vary with block position and vice versa. One approach is to chain models [7] such that a base regressor is trained to predict one label element and then predict the next label element using the previous label estimate as a feature. This introduces a bias for the ordering of training but allows for univariate regressors to be applied in series to multi-output problems. To maximize performance, principal component analysis (PCA) was applied to the input data prior to assess whether any models benefitted from dimensionality reduction primarily as a low-pass filter in the feature space but also as a dimensionality reduction, which is a common practice in machine learning. The transformation did not yield significant improvements and thus the original 11-dimensional feature space was retained and scaled by median and interquartile range to eliminate outlier effects [10].

IV. Machine Learning Algorithms [1]

A. Decision Tree Regressors

The decision tree is a model that divides the training samples based on feature values to minimize a metric of homogeneity. Common metrics include information gain and the Gini impurity. In the regression problem, inference is performed by taking the average of a given leaf. Because decision trees can account for combinations of labels, they avoid the independence assumptions made by regressor chains. In this experiment, decision trees and their variants were thus promising.

A single decision tree is likely to overfit to the entire training set. A random forest regressor corrects this issue by forming a set of decision trees is formed with each individual tree responsible for learning a partition of the training set. Overfitting is therefore reduced by subsampling from the original training data and using majority voting of several decision trees instead of relying on one decision tree.

Gradient-boosting algorithm assign a differentiable loss function to iteratively optimize a set of weak learners' hypotheses.

B. Regressor Chains

Two independent single-output models must be trained, one per predicted variable, if single-output regression algorithm are to be considered. Evaluating this approach showed that this independence effect was non-negligible. Regressor chains allow correlated multioutput prediction by using the prediction of the first independent model as a feature that is fed to the second model, such that the feature space of the first model is the 11 sensor readings and the feature space of the latter model is 11 sensor readings plus the first model's output. Regression chaining provided a framework for structurally independent single-output regressors to perform correlated predictions. The method was attempted with polynomials (degrees 1 to 5) and Elastic Net models based on [12], as well as with individual random forest models to compare mean-squared error with respect to the multiple output model.

[1]algorithm descriptions are supported by [11] unless otherwise mentioned

C. Artificial Neural Networks

Another approach to the multioutput regression problem is using a neural network (also known as a multilayer perceptron). These methods approximate a gradient with respect to a mean-squared error loss function between a 2-vector y, y* representing (actual angle, actual block position) and (predicted angle, predicted block position) respectively. Neural networks have the advantage of being capable of modeling complex, nonlinear, and high-dimensional functions [13]. Note that normalization of the block position and joint angle was necessary prior to training to avoid biasing optimization towards minimizing block position error only.

D. Evaluation

Each model was assessed according to the root-mean squared error metric:

$$RMSE(y, y*) = \frac{1}{n}\sum_{i=1}^{n}(y_i - y_i^*)^2$$

where the data set consists of n samples and y_i and y_i^* are labels and model predictions respectively.

TABLE I
Selected Model Joint Angle RMSE (rad)

Model	Training	Test
Decision Tree	0.12	0.46
Random Forest	0.11	0.34
Gradient Boost	0.28	0.32
Cubic + ElasticNet Chain	0.10	0.41
Neural Net	0.17	0.31

TABLE II
Selected Model Block Position RMSE (mm)

Model	Training	Test
Decision Tree	8.8	24.3
Random Forest	8.9	21.6
Gradient Boost	15.8	21.9
Cubic + ElasticNet Chain	9.8	25.1
Neural Net	11.8	20.7

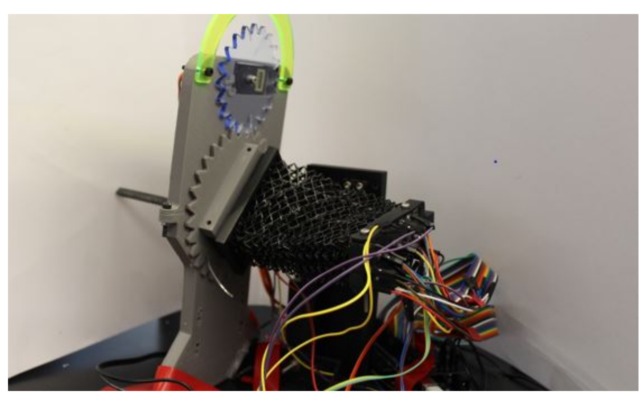

Fig. 1. The arm apparatus

Conclusions

The experiments conducted showed that the degrees of freedom could be accurately estimated by a machine learning model. We showed that real-time inference can detect changes with no discernible latency. In comparison of multioutput regressors, the artificial neural networks generalized best to unseen data by demonstrating the lowest test accuracy.

Future Work

We hypothesize that as further degrees of freedom are explored, the properties of the ANN as a universal function approximator [14] will allow it to continue outperforming other models in terms of metric error. Our next steps are therefore to explore such problems. Bayesian parameteric estimation may be another approach to the problem that may be less data intensive. Finally, studying balance of locomotive robot using these findings would be a useful extension and allow another comparison between human and robot proprioceptive sensing.

Acknowledgment

Thank you to everyone who helped make this experiment possible, including the Hunter R. Rawlings III Cornell Presidential Research Scholarship program and the Organic Robotics Lab at Cornell.

References

[1] Tuthill and E. Azim, "Proprioception", Current Biology, vol. 28, no. 5, pp. R194-R203, 2018. Available: 10.1016/j.cub.2018.01.064.

[2] D. Rus and M. Tolley, "Design, fabrication and control of soft robots", Nature, vol. 521, no. 7553, pp. 467-475, 2015. Available: 10.1038/nature14543.

[3] Q. Zhang, L. Yang, Z. Chen, and P. Li, "A survey on deep learning for big data", Information Fusion, vol. 42, no.1, pp. 146-157, 2018. Available: 10.1016/j.inffus.2017.10.006.

[4] C. Shang, F. Yang, D. Huang and W. Lyu, "Data-driven soft sensor development based on deep learning technique", Journal of Process Control, vol. 24, no. 3, pp. 223-233, 2014. Available: 10.1016/j.jprocont.2014.01.012.

[5] R. Pfeifer, M. Lungarella and F. Iida, "The challenges ahead for bio-inspired 'soft' robotics", Communications of the ACM, vol. 55, no. 11, p. 76, 2012. Available: 10.1145/2366316.2366335.

[6] J. Zhang, T. Lockhart and R. Soangra, "Classifying Lower Extremity Muscle Fatigue During Walking Using Machine Learning and Inertial Sensors", Annals of Biomedical Engineering, vol. 42, no. 3, pp. 600-612, 2013. Available: 10.1007/s10439-013-0917-0.

[7] Nakajima, K., Hauser, H., Li, T., Pfeifer, R. (2018). Exploiting the Dynamics of Soft Materials for Machine Learning. Soft Robotics, 5(3), 339–347. doi: 10.1089/soro.2017.0075

[8] I. Van Meerbeek, C. De Sa and R. Shepherd, "Soft optoelectronic sensory foams with proprioception", Science Robotics, vol. 3, no. 24, p. eaau2489, 2018. Available: 10.1126/scirobotics.aau2489.

[9] P. Xu, A. Mishra, H. Bai, C. Aubin, L. Zullo and R. Shepherd, "Optical lace for synthetic afferent neural networks", Science Robotics, vol. 4, no. 34, p. eaaw6304, 2019. Available: 10.1126/scirobotics.aaw6304.

[10] "sklearn.preprocessing.RobustScaler — scikit-learn 0.22 documentation", Scikit-learn.org, 2019. [Online]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.RobustScaler.html. [Accessed: 09- Dec- 2019].

[11] T. Hastie, R. Tibshirani and J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd ed. Springer, 2009.

[12] Read, B. Pfahringer, G. Holmes and E. Frank, "Classifier chains for multi-label classification", Machine Learning, vol. 85, no. 3, pp. 333-359, 2011. Available: 10.1007/s10994-011-5256-5.

[13] Y. LeCun, Y. Bengio and G. Hinton, "Deep learning", Nature, vol. 521, no. 7553, pp. 436-444, 2015. Available: 10.1038/nature14539.

[14] I. Goodfellow, Y. Bengio and A. Courville, Deep learning, 1st ed. MIT Press, 2016.