
Soft Robotics for Tactile Object Classification

Lingyun Xu

Carnegie Mellon University
Pittsburgh, PA 15213
lingyun3@andrew.cmu.edu

Donggu Lee

Mechanical Engineering Department
Carnegie Mellon University
Pittsburgh, PA 15213
donggul@andrew.cmu.edu

Louis Plottel

The Robotics Institute
Carnegie Mellon University
Pittsburgh, PA 15213
lplotte@cs.cmu.edu

1 Overview and Context

1.1 Abstract

Tactile object classification enables robots and automated systems to identify objects through touch-based sensing, crucial in environments where visual perception is limited or impossible. This project aims to develop a deep learning approach for classifying objects based on their tactile signatures, addressing real-world scenarios such as low-light conditions and vision-restricted settings. By analyzing contact profiles during object interaction, our system will perform categorical object classification. We use a soft robotic finger with a single position sensor for this project. Soft robots offer advantages in safety and allow for rich contact with objects without risk of damage, and utilizing a single sensor reduces cost and system complexity. Those who stand to benefit include package sorting and logistics companies, humanoid robot developers working on advanced manipulation, and the scientific community more broadly.

1.2 Motivation

Vision-based object recognition, while widely adopted, faces significant limitations in many real-world scenarios. Low-light environments, occluded spaces, and situations requiring object identification through sealed containers all demonstrate the need for alternative sensing methods. Tactile sensing fills this crucial gap by mimicking human ability to recognize objects through touch. This capability is particularly valuable in applications such as warehouse automation, hazardous material handling, and assistive technologies for the visually impaired. Moreover, tactile sensing complements visual data in robotic manipulation tasks, potentially improving grasp stability and object handling precision. Our specific aim is to achieve a high-level of success using a single position sensor, whereas previous work relies on multiple of the same type or a combination of different sensors (see Table 1). Using a single sensor offers meaningful advantages by reducing cost and system complexity. By developing robust tactile classification systems, we can enhance the reliability and versatility of automated systems across diverse operating conditions.

1.3 Objective

We propose a tactile object classification system integrated with a custom-designed soft robot. Our approach utilizes a single finger to collect curvature measurements during standardized grasping motions, establishing a foundation that can be readily scaled to multi-finger sensing. The experimental setup involves:

Data Collection Systematic gathering of contact profiles from 7 distinct objects. Fixed robot position with varied object poses to ensure diverse tactile signatures. Temporal sequences of position measurements form contact histories. A total of 30 trials are conducted for 7 distinct objects, generating a novel dataset of 10-15 hours of real-world physical interaction data.

Classification Framework We develop a deep learning model that processes temporal sensory data for object classification based on contact history features. This controlled experimental design ensures consistent data collection while maintaining sufficient variability to test the system’s robustness and generalization capabilities.

Goal Demonstrate ability to learn object classification from the given data by having a much-better-than-random classification accuracy.

2 Related Work and Background

2.1 Related Work

Tactile Data Enhances Perception [2] propose a machine learning approach that achieves a 97.6% average success rate in classifying a diverse set of 49 objects using handcrafted features extracted solely from tactile data. However, they acknowledge that while these features are effective in recognizing material properties, they struggle to distinguish objects with different shapes.

[6] introduce a visuotactile system that predicts object shape and pose from a single-view RGB-D image combined with vision-based touch sensors. Their findings indicate that incorporating tactile data significantly improves shape reconstruction and enhances pose estimation performance by 94% compared to a vision-only approach, especially under heavy occlusion.

[1] propose an end-to-end approach using convolutional neural networks (CNNs) to process RGB images and tactile images captured by GelSight sensors, predicting the success probability of a grasp. Their results demonstrate that the visuotactile multi-modal model outperforms the vision-only counterpart by an additional 14% in success rate.

These work prove that tactile data encodes rich physical information about the objects, showing our project has deep potential and feasibility.

Sensory Hardware for Tactile Profiling Tactile-based tasks heavily depend on the specific sensory hardware used for data collection, leading to different forms of tactile profiles. GelSight sensors capture high-resolution, camera-like tactile images that reflect the intensity of surface deformation [1], whereas traditional heat and force sensors generate vectorized signals over time [2]. These varying data modalities necessitate different preprocessing and feature extraction techniques.

Image-based tactile data is well-suited for convolutional operations, enabling the use of state-of-the-art (SOTA) deep learning architectures such as ResNet. Conversely, traditional tactile sensors provide rich physical information, facilitating precise modeling of hand-object interactions while requiring minimal preprocessing (e.g., statistical feature extraction). Our work focuses on identifying the most informative features based on data collected from our custom-designed robotic fingers. More details on data collection are provided in section 3.2.

Design of Tactile Features [2] apply traditional machine learning techniques to process handcrafted tactile features. They perform principal component analysis (PCA) on statistical features derived from multi-channel sensory inputs, followed by classification using either a multilayer perceptron (MLP) or a support vector machine (SVM).

[1] employ CNNs to extract neural features from image-like tactile data, capturing spatial and contact information. However, their approach requires extensive training on diverse datasets. Given our experimental constraints—limited data and fewer sensory channels—we initially adopt handcrafted features as baselines. We then aim to enhance these features by leveraging temporal neural networks to encode the complete tactile trajectories.

Sensors	Learning Architecture	Reference
19 pressure, temperature, and vibration	SVM, Gaussian	[2]
Position and force	KNN	[3]
Position and camera	CNN	[7]
Position, visuotactile, and camera	Transformer	[6]
6 Pressure sensors	biLSTM	[4]
19 Pressure sensors	CNN	[5]
1 Position	CNN	This work

Table 1: Previous work relies on multiple sensors and a variety of machine learning approaches.

2.2 Summary

A brief summary of relevant prior approaches is presented in Table 1. Our approach is additionally shown at the end, with our best performing architecture shown.

3 Methodology

3.1 Model Description

3.1.1 CNN

CNNs are effective for local pattern recognition in data, and we hypothesized they could work well for detecting local patterns in the time-series data, for example, object stiffness or surface characteristics. Our best performing CNN architecture consists of three 1D CNN layers, an average pooling layer, a flattening layers, two fully connected layers, and a softmax function. Our 9D time-series data is fed into this model, and the softmax function returns one of seven class labels. We use ReLU() activation functions and batchnorm and dropout to improve performance. The total parameter size is approximately 160,000 and the test accuracy is 51.3%. A visualization of this architecture is shown in Figure 5.

3.1.2 CNN + LSTM

In order to capture longer-term patterns in the data, we added a biLSTM layer to our CNN model. We use the CNN as an encoder, followed by the bLSTM and then the MLP layer. Currently, our best iterations of this architecture under perform the pure CNN, achieving a classification accuracy of 35.3%. Figure 6 contains a visual depiction of this architecture.

3.1.3 CNN + transformer

CNNs are known to extract local spatial patterns but fail to capture long-range dependencies due to limited perception scope. Since our trajectory length is especially long (2500), we introduce transformer encoder layers after CNN to extract global context. Our current best model has four CNN layers and two transformer encoder layers with [CLS] token aggregation for feature representation, achieving 42.1% val accuracy. Due to the $O(N^2)$ computational complexity of the transformer, it's expensive to tune model structures or hyperparameters. We've also tried using pooling layers with stride>1 in CNN layers to reduce sequence length, but it turns out to significantly hinder model performance (acc drops to around 30%).

3.1.4 CNN + MLP ensemble

To further enhance model performance while maintaining low complexity, we implemented an ensemble architecture combining CNN-based temporal feature extraction with statistical descriptors. The CNN component consists of two convolutional blocks with batch normalization and max pooling, significantly reducing the time-series dimensionality while preserving key spatial patterns. In parallel, the model extracts statistical features including mean, standard deviation, maximum, and minimum across the input sequence. These two feature sets are concatenated and passed through a dimensionality reduction layer followed by a lightweight MLP classifier. This architecture outperformed all previous models, achieving the highest test accuracy of 60.4%, and demonstrated

improved robustness across object categories. Its balance between capacity and regularization makes it well-suited for resource-constrained tactile sensing applications.

3.2 Dataset

We collect our own dataset for this project. We probe a set of 7 objects (Fig. 1) using a finger and record the time series data from an embedded bend sensor. The finger follows a pre-programmed trajectory defined by 14 waypoints. We intentionally selected items with various shapes and material properties. In addition, we included three items (pink ball, blue ball, and tennis ball) that have near identical shapes but significantly different material properties. The pink ball is quite soft, the blue ball is firm, and the tennis ball is hard and fuzzy.



Figure 1: Items in the dataset include a stuffed animal, a pencil case, a bottle, a box, a soft ball, a firm ball, and a tennis ball.



Figure 2: Probing the tennis ball during a data collection trial

We use a pre-programmed probing method, which is the same for all objects touched. We probe a set pattern, probing different points on an object and brushing a finger against an object as a proxy for friction measurement. (See Fig. 2)

We slightly adjust the position of objects (e.g., rotating them or changing the distance from the finger) between runs.

The data we collect is timeseries data, with curvature about x- and y-axes of the finger at each timestep. We also record the finger's target location (from the probing pattern) and the PWM control commands sent to each of the finger's four actuators. A visualization of the data from a single trial is shown in Figure 3.

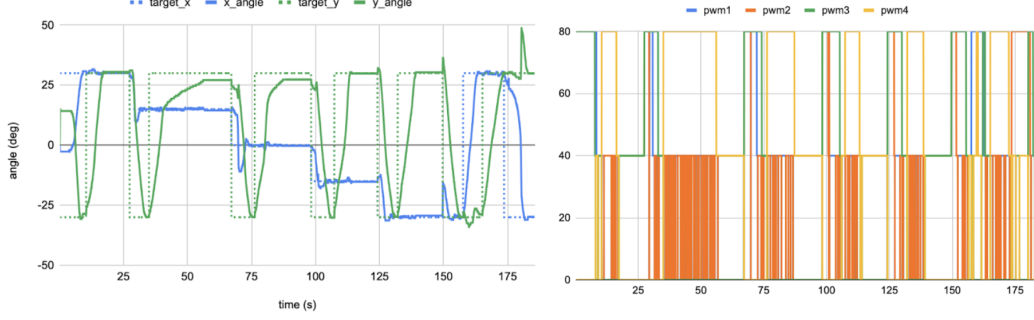


Figure 3: Data Visualization for a single trial run. Left: x- and y- curvature position and targets positions. Right: PWM inputs to each SMA (max 255).

3.3 Evaluation Metric

Our model evaluation is classification accuracy, measured both overall and per-class. We track the ratio of correctly classified samples to total samples across the test set, while also monitoring individual class performance to identify recognition disparities between different object types. This approach provides insights into how our model handles various material properties and interaction patterns. Additionally, we record prediction probabilities to analyze model confidence levels across different object categories.

3.4 Loss Function

We implement Cross-Entropy Loss as our objective function, which effectively handles multi-class classification by penalizing incorrect predictions proportionally to their confidence. This is paired with the AdamW optimizer that incorporates weight decay regularization to reduce overfitting. Our training process uses a ReduceLROnPlateau learning rate scheduler that adaptively decreases the learning rate when validation performance plateaus, helping navigate the complex loss landscape of temporal contact data while avoiding poor local minima.

4 Baseline and Extensions

4.1 Baseline Selection and Evaluation

Our baseline approach employs a feature-based methodology for analyzing contact-based interactions. We use statistics as handcraft features. Since waypoints separate the data trajectory, we also include per-waypoint statistics to provide more distinguishable information. In conclusion, we extract statistical features from sliding windows of sensor data, including:

- Per-column per-waypoint min, max, mean, std., range, time
- Per-column overall min, max, mean, std, range, time
- Per-waypoint total time
- Overall total time

These features are fed into a three-layer MLP as the classification head. We use Python 'multiprocessing' library to precalculate the features and store as local cache to speed up training. We achieve 47.3% accuracy on test set, which performs significantly than random guessing (14.3%).

4.2 Implemented Extensions / Experiments

Our analysis identified two key limitations in the current implementation:

- **Dataset Redundancy:** The data collection setup resulted in highly similar sequences across certain time periods, creating potential redundancy. This temporal correlation may artificially inflate validation performance and limit the model's ability to generalize to new interactions.

- **Network Capacity:** The current model contains approximately 8 million parameters, which may be insufficient to capture the full complexity of contact-based interactions across diverse object categories.

To enhance recognition performance, we propose several extensions targeting the identified limitations: data preprocessing and augmentation, dual-path temporal processing architecture, and material-object co-learning framework.

We will address dataset redundancy through intelligent filtering to reduce temporal correlation between examples. Our preprocessing pipeline will normalize sensor readings and create synthetic variations using noise injection and time warping. By applying variable stride windows, we'll increase sample diversity and reduce the model's dependence on specific temporal patterns in our current dataset.

Our model will process tactile information through two complementary paths: a Static Multi-Frame Integration Model using CNN encoders with temporal attention mechanisms for capturing stable material properties, and a Dynamic Sequential Processing Model using Transformer/LSTM architecture for processing varying-length contact sequences. This dual-path approach allows simultaneous capture of both stable material characteristics and evolving temporal dynamics.

These extensions aim to address the fundamental limitations in our current approach while capturing more nuanced patterns in contact-based interactions and improving the model's ability to generalize across object categories.

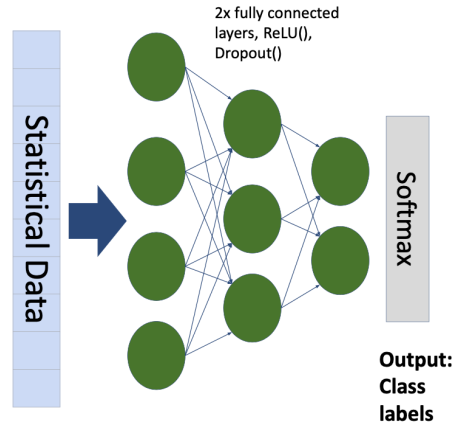


Figure 4: Statistical data was fed into an MLP to create a baseline classifier.

5 Results and Analysis

5.1 Results

Our baseline implementation achieved a test accuracy of 47.3% across seven object categories. The CNN + Statistics-based MLP model achieves the highest classification accuracy of 60.4%, outperforming all other architectures. This improvement demonstrates the benefit of combining learned representations with handcrafted statistical features. The pure CNN model achieves 51.3% accuracy, while the LSTM and Transformer models remain less effective at 35.3% and 42.1%, respectively. Confusion matrices for baseline, CNN, LSTM, RNN (LSTM), Transformer, and ensemble models are presented in Figures 7, 8, 9, 10, and 11, respectively. A summary of training accuracies is shown in Table 2.

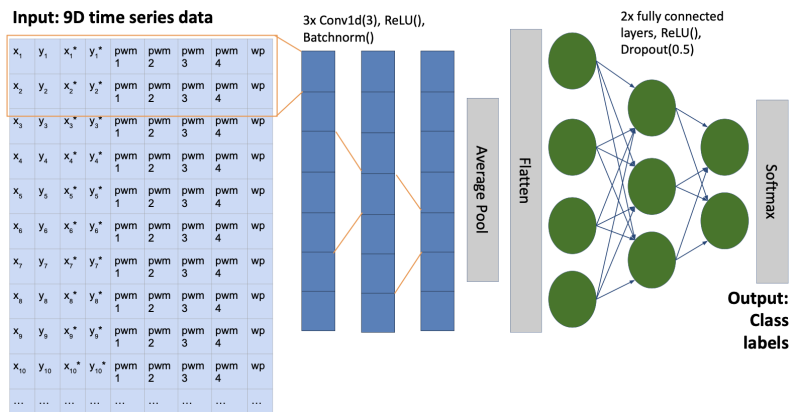


Figure 5: CNN architecture resulting on the best classification accuracy.

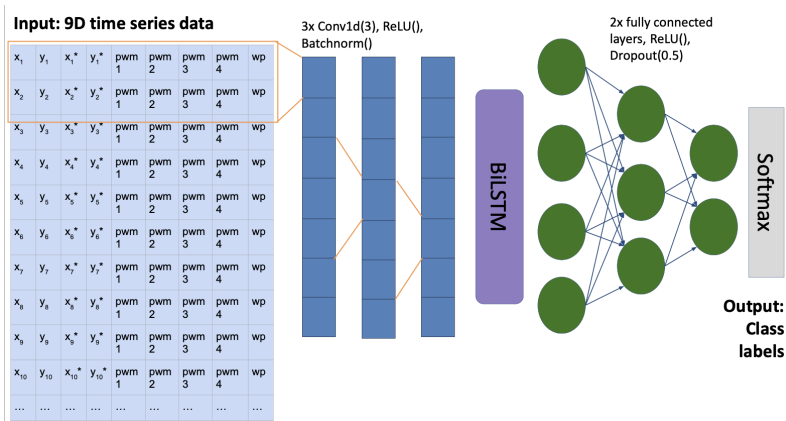


Figure 6: Architecture of the RNN classifier.

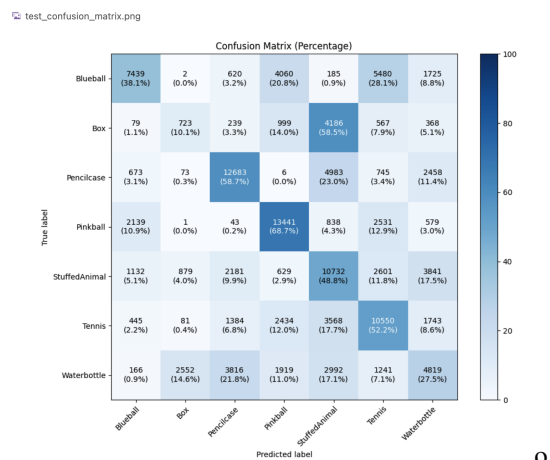


Figure 7: Baseline confusion matrix

Model	Test Classification Accuracy
Naive Baseline (random guess)	14.3%
Smart Baseline (Statistics-based MLP)	47.3%
CNN	51.3%
RNN (biLSTM)	35.3%
Transformer	42.1%
CNN + Statistics-based MLP	60.4%

Table 2: Comparison of different classification models.

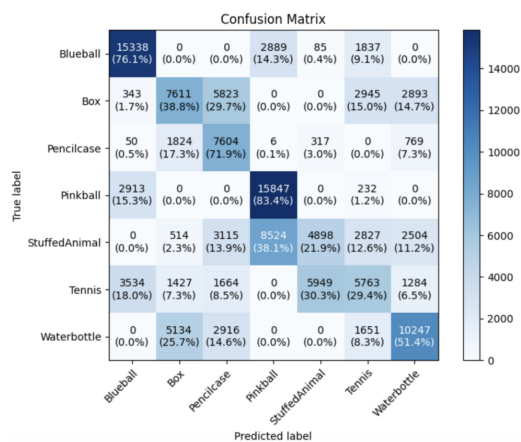


Figure 8: CNN confusion matrix

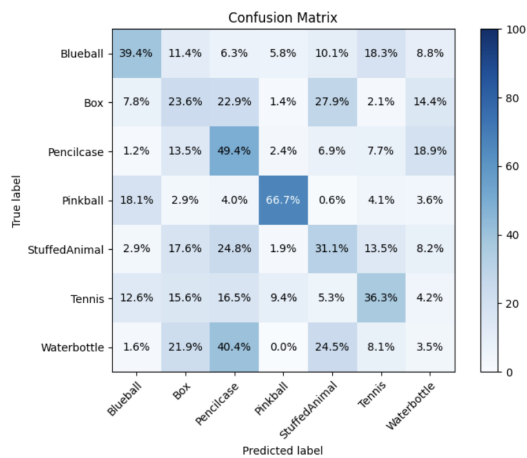


Figure 9: RNN confusion matrix

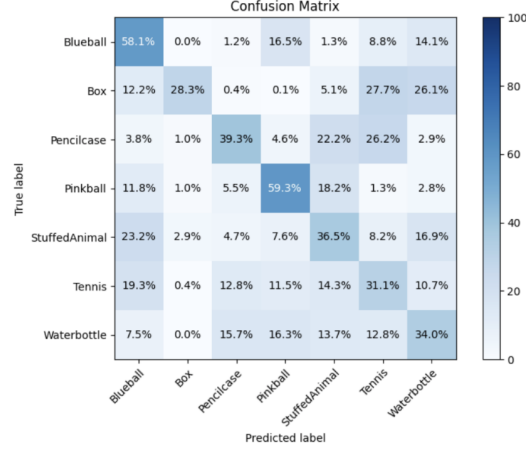


Figure 10: Transformer confusion matrix

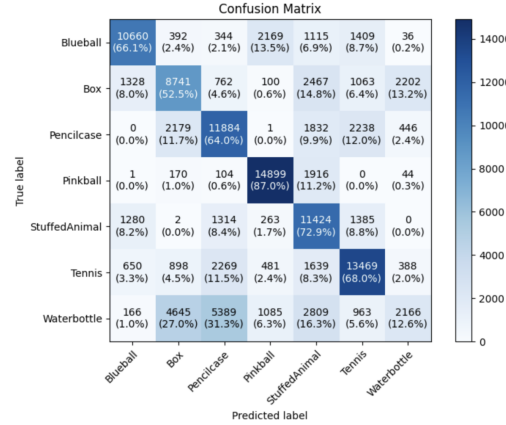


Figure 11: CNN + MLP confusion matrix

5.2 Discussion

Key challenges include the gap between training and validation performance, suggesting limited generalization beyond specific interaction patterns. This aligns with our identified dataset redundancy issue. The improved performance of the CNN + Statistics-based MLP model suggests that temporal and material-specific features captured by statistical descriptors provide valuable complementary information to CNN-learned spatial features. This hybrid approach better distinguishes objects with similar shapes but different stiffness, reducing confusion between categories like the pink and blue balls. Furthermore, the updated confusion matrix shows improved balance across all classes, indicating enhanced generalization.

6 Future Directions

1. Continue development and refinement of CNN, RNN, Transformer and ensemble models.
2. Expand the object set, for example by including objects which differ in only one aspect. This could include multiple objects of different shapes or sizes made from the same material, and multiple objects of the same size and shape made from different materials.
3. Vary the data collection script to randomly or adaptively probe points.

7 Conclusion

In this work, we collect a novel dataset containing real-world interaction data between a soft robotic finger and seven distinct objects. We implement five learning models, (1) a statistics-based MLP classifier, (2) a CNN, (3) a biLSTM, (4) a transformer model and (5) an ensemble model. All models significantly outperform random guessing, with the CNN + Statistics-based MLP ensemble model achieving the highest classification accuracy of 60.4%, showing over $> 4.3\times$ improvement over random guessing. This demonstrates the effectiveness of combining deep-learned features with handcrafted statistical insights, especially when working with limited sensor modalities. Our results validate the potential of tactile object classification using a single soft robotic finger and highlight promising directions for future sensor fusion and multi-modal learning.

8 Team Contributions

- Lingyun Xu:
 - Transformer architecture lead
 - Baseline architecture lead
 - Report writer
- Donggu Lee:
 - Data augmentation lead
 - Soft robot fabrication
 - Data collection
 - Report writer
- Louis Plottel:
 - CNN architecture lead
 - RNN architecture lead
 - Data collection script
 - Data collection
 - Report writer

9 Github Link

https://github.com/Plottt/IDL_Soft_Robot

10 Dataset Link

https://drive.google.com/drive/folders/1_nPwEAPmrw6t-rxVRmkfM8vOCUeZgRWw?usp=sharing

References

- [1] Roberto Calandra, Andrew Owens, Manu Upadhyaya, Wenzhen Yuan, Justin Lin, Edward H Adelson, and Sergey Levine. The feeling of success: Does touch sensing help predict grasp outcomes? *arXiv preprint arXiv:1710.05512*, 2017.
- [2] Janine Hoelscher, Jan Peters, and Tucker Hermans. Evaluation of tactile feature extraction for interactive object recognition. In *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pages 310–317, 2015.
- [3] Bianca S Homberg, Robert K Katzschmann, Mehmet R Dogar, and Daniela Rus. Robust proprioceptive grasping with a soft robot hand. *Autonomous robots*, 43:681–696, 2019.
- [4] Tetsushi Nonaka, Arsen Abdulali, Chapa Sirithunge, Kieran Gilday, and Fumiya Iida. Soft robotic tactile perception of softer objects based on learning of spatiotemporal pressure patterns. In *2023 IEEE International Conference on Soft Robotics (RoboSoft)*, pages 1–7. IEEE, 2023.
- [5] Oliver Shorthose, Luca Scimeca, Alessandro Albini, and Perla Maiolino. Bioinspired tactile object identification leveraging deep learning and soft body compliance. *Advanced Intelligent Systems*, page 2400802, 2025.
- [6] Sudharshan Suresh, Haozhi Qi, Tingfan Wu, Taosha Fan, Luis Pineda, Mike Lambeta, Jitendra Malik, Mrinal Kalakrishnan, Roberto Calandra, Michael Kaess, et al. Neuralfeels with neural fields: Visuotactile perception for in-hand manipulation. *Science Robotics*, 9(96):eadl0628, 2024.
- [7] Thomas George Thuruthel and Fumiya Iida. Multi-modal sensor fusion for learning rich models for interacting soft robots. In *2023 IEEE International Conference on Soft Robotics (RoboSoft)*, pages 1–6. IEEE, 2023.