Learning-based Object Classification with Optoelectronically Tactile Finger*

Xudong Han[1], Zicong Zhang[1], Zhengtao Liu[1], Ranbao Deng[1], Yichen Song[1], Dapeng Zhang[1], Haowen Wang[1], Yangchenguang Liu[1]

***Abstract*—This paper presents an object classification strategy using an adaptive tactile soft finger with multi-channel optical fibers, which is under the theme of waste classification and sustainable development. We further integrated such fingers in a gripper design with two fingers. Machine learning methods are used to train a model for object identification using the tactile data collected. Detailed experimental results are also included to further validate the proposed method for enhanced classification accuracy. Video: https://youtu.be/JVlRleaBBMI**

***Index Terms*—soft robotics, classification, tactile sensing, optical fiber**

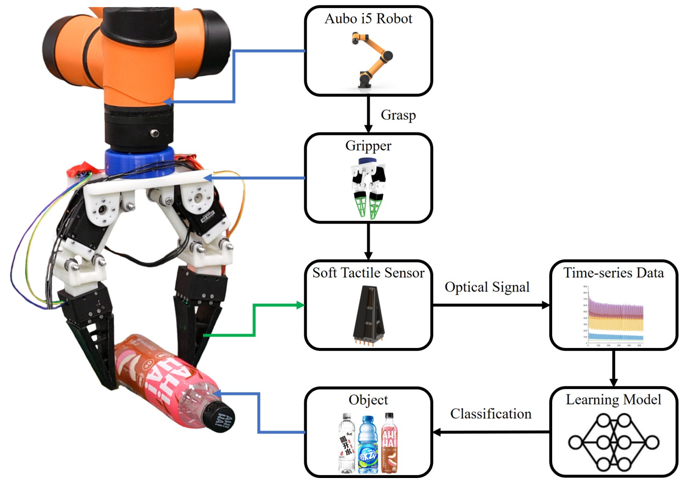


Fig. 1: Overview of the proposed system design with optoelectronically adaptive tactile fingers for object classification. Left shows the system integration of a gripper with the proposed tactile fingers on a Aubo i5 robot. Right shows the workflow of the proposed object classification strategy.

I. Introduction

WASTE classification is an important part of sustainable development [1]. With the promotion of waste sorting, the traditional manual sorting, which is a highly labor-intensive process, needs improvement [2][3]. Automation becomes a perfect solution [4]. In early waste classification, visual approaches are used widely based on previous computer vision work [5][6][7]. However, waste classification focus on the material of the waste [5], which could not be represented well through its appearance [7]. Therefore, tactile object recognition is introduced into this problem [8]. Through tactile sensing, the material and shape information is perceived in a different way than visual method [9][10], providing a more physical view of the surface, which provide more precise classification results.

This paper proposes an object classification strategy, as shown in Fig. 1, using tactile sensing data collected from the optoelectronically tactile soft finger, which establishing the relationship between the material behavior and the pattern change of light flux through machine learning.

A. Tactile Sensing

The importance of tactile sensing was first suggested by researchers in the 1980s. So far there have been several approaches to realize tactile sensing. The state-of-the-art approach through electronic design is based on piezoresistive materials and conductive thread electrodes [11]. The design has simpler complexity and maintains enough sample frequency compared with previous work [12]. The other common approach is optoelectric design. Such designs are based on a light source, a transduction medium, and a photodetector. The tactile information is passed as the contact force affects the transduction medium, causing changes in light intensity. This approach has been widely used in designing soft sensors [9]. The gripper proposed in this paper is also based on the idea.

B. Object Classification through Tactile Information

The object classification through tactile information focuses on the material information. However, the key characteristics of the material are different in previous work. Most results are produced based on learning of the deformation behavior of different materials [13][14]. There are also some works that proposed examples of using texture features [11] or a combination of multiple features [15].

C. Proposed Method and Contributions

This paper is a continuation of our earlier work with the omni-adaptive soft finger for rigid-soft interaction learning [13], rigid-soft tactile grasping [10], and optical fiber-based grasp sensing [9]. In this paper, we continue to use the sensorized design [10] of the omni-adaptive soft finger using multiple optical fibers embedded with friction enhanced soft surface. Tactile data, in the form of light flux signals, can

*This work was supported by Southern University of Science and Technology and AncoraSpring Inc. for the ME336 Collaborative Robot Learning (me336.ancorasir.com) under the supervision of Prof. Song Chaoyang songcy@ieee.org

[1]Xudong Han, Zicong Zhang, Zhengtao Liu, Ranbao Deng, Yichen Song, Dapeng Zhang, Haowen Wang and Yangchenguang Liu are with Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen, Guangdong 518055, China. 11812519, 11711203, 11711018, 11712002, 11710415, 11710203, 11711020, 11812911@mail.sustech.edu.cn

be learned to achieve object classification with enhanced robustness. The contributions of this paper are listed as the following:

1) Proposed a gripper design using the omni-adaptive soft finger with enhanced finger surface and multi-channel optical fiber for proprioceptive, tactile sensing.
2) Achieved sensing of shapes of different objects with proposed design.
3) Realized and compared the object recognition ability of different machine learning methods in this scene.

In the rest of this paper, Section II explains the problem formulation of tactile object classification and proposed method. Section Experimental results and discussion are enclosed in sections III and IV. Conclusion are enclosed in Section V.

II. Method

A. Problem Formulation

In this paper, we would mainly deal with three kinds of object: plastic bottles, aluminum cans, and hard cans. These three categories show a large difference in material behavior and this is the main feature we would be using in the classification. However, the sizes of the object that is made by these materials are usually different. To avoid side effects caused by the difference in target size, we used the same grasp depth in different targets. Also, the grasp speed is also the same as to control variable. For the mechanical part, we are using the optoelectronically innervated tactile finger as the gripper. Each finger is loaded with a circuit board. Also, each finger is driven by one motor. Considering the robot arm and the camera, we are having four different modules, in total 6 components in our system. Therefore, control integration is necessary to lower the complexity in actual practice.

B. Optoelectronically Soft Tactile Finger

The finger used in this paper is a recent work [16]. We utilize the structural space in the early finger design and achieve the tactile perception of finite object classification by introducing optical fiber into the soft structure. With the latest finger, we've vastly improved sensor reliability, tactile capability, and grip robustness. The reasons to use the latest version of the finger are: it increases the contact friction of the finger surface while maintaining its full adaptability; Integrated design to enhance tactile perception; improve manufacturing processes to reduce cost and complexity; maintain compatibility with changes in the unstructured environment.

In this new type of finger design, five optical fibers are evenly arranged on the surface of the finger with increased surface friction and the soft finger structure to form a sensor array, which is used to measure the deformation of the finger during grasping, as shown in Fig. 2 (b) shown. Due to the loss of luminous flux and the degree of bending of the soft finger, we choose the optical fiber to obtain the soft finger deformation. There are 5 LEDs at the transmitting end of each optical fiber as the light source transmitter, and at the same time, there are 5 photoresistors at the receiving end of the optical fiber to obtain the luminous intensity. To reduce the

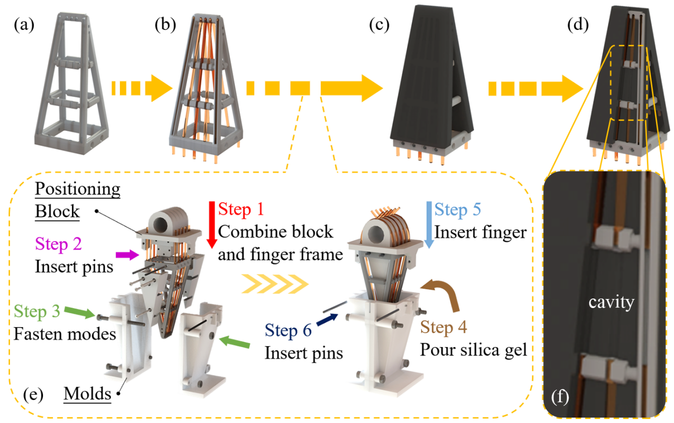


Fig. 2: The design and fabrication process of the optoelectronically soft tactile finger: (a) finger frame; (b) finger frame with the fibers (orange transparent material is used instead to visualize the transparent fibers clearly in the figures); (c) finger frame with silica gel skin; (d) we pull out the optic fiber to leave a cavity in the middle of finger; (e) fabrication process of the black silica gel skin; (f) the cavity in the sensitive area.

influence of light in the surrounding environment and ensure that we can obtain sufficient luminous flux, we chose 520-525NM led and 520-550NM photoresistors. The wavelength bands of these two kinds of light are more concentrated and matched.

In addition, we covered a layer of skin made of black silica gel outside the five rays to isolate it from the environment, which resulted in an increase in the signal-to-noise ratio in Fig. 2 (c). We further leave a cavity in the middle of the contact surface to improve the sensitivity of the fiber [9]. As shown in Fig. 2 (d)(f), each of the five optical fibers is cut into two sections and stopped at the cavity section position shown in Fig. 2 (f).

The fiber we use is a soft PMMA commercial fiber produced by Everheng Fiber. The black skin is made of Smooth-On EcoflexTM 00-30 silicone, its strength meets our requirements. However, this silica gel was originally a milky white translucent liquid and was later mixed with black pigment at a ratio of 20:1, effectively blocking the surrounding light. In addition, we use silica gel to build the finger skin at 3mm on the outer surface of the finger to increase the texture of the finger surface and enhance the grip effect. Fig. 2 (e) shows the details of the manufacturing process of the black silicone skin.

Let I_0 denotes the baseline luminous intensity without any deformation. With the current output luminous intensity I, the luminous flux loss in decibels through the optical fiber is then described as

$$\alpha = 10log_{10}(I_0/I) \tag{1}$$

According to this definition, the output loss α is 0 without deformation, and less than 0 when interacting with the environment.

C. Gripper Design of Two Fingers

To ensure the convenience of collecting our experimental data set and the simplicity of the end effector, we designed a two-finger gripper, as shown in Fig. 3. Each finger has one degree of freedom, that is, one ROBOTIS Dynamixel MX-64 is used to drive the opening and closing of the finger. Dynamixel SDK provides us with a high-level Python API. This API can immediately communicate between the grabbing hardware and the host and can provide information such as current position, target position, torque limit for adjusting the stiffness of the finger, etc.

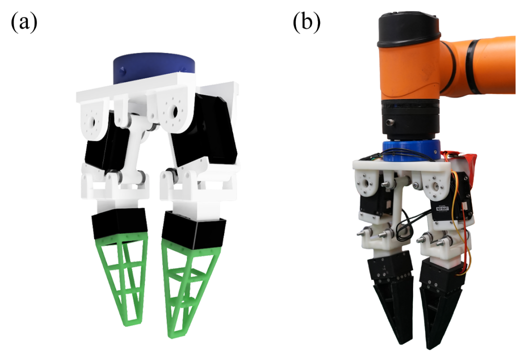


Fig. 3: (a) is the CAD model of the two-finger gripper design and (b) shows the gripper on the Aubo i5 Robot.

As shown in Fig. 3, the holder presents a two-finger configuration: the two fingers are arranged symmetrically, so that the proximal axis of the joint is located in the middle of the other fingers, facilitating more precise grasping. Based on the required distance range of the experiment part, the holder we designed is the simplest device after the experiment can be carried out normally, and the cost is reduced. The simple structure makes it convenient for fast installation and part replacement. At the same time, because the gripper is made of soft material, the holder can adapt to various geometric characteristics of target objects with various typical structures, providing high adaptivity and safety.

D. Object Classification with Gripper

To classify the material of the picked garbage with the provided soft gripper, we are going to implement commonly-used models and by comparison, find the most suitable one and further optimize the accuracy and speed via configuring data prepossessing, feature extraction, hyper parameters in selected models.

The general procedure of the classification via machine learning techniques is shown in Fig. 4.

Where in data preprocessing, certain noise reduction and filtering methods are applied to improve the quality of the data. In feature extraction, normally based on needs and experience, some good features are selected for further discussion. And then, using the selected models to train with part of the processed data. Based on the problem, classification or regression models could be chosen and then using the test set

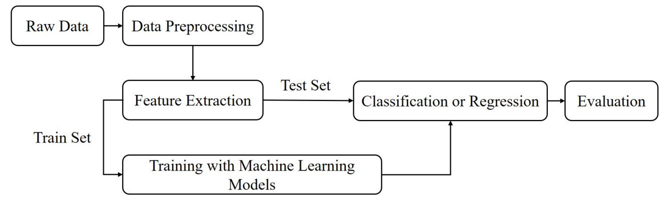


Fig. 4: General procedure of machine learning techniques.

to evaluate the generated model. If the performance reached the expectation, this model could be used in real applications.

To be more specific, the raw data provided would be different channels of time-sequenced values. In data prepossessing, we will use common denoise filters and Kalman filters to get clean data. For feature extraction, we will try to focus on the time-dependent data given by the continuous grasping. For classification algorithms, we are going to implement both supervised learning and deep learning methods to compare and find the most suitable algorithms for our task. The evaluation will be based on the classification accuracy and processing speed of the algorithms. We chose several supervised learning algorithms and discussed the possibility of utilizing LSTM on this problem.

1) Supervised Learning: Supervised learning is the machine learning task of learning a function that maps an input to an output based on example input-output pairs. It infers a function from labeled training data consisting of a set of training examples. In supervised learning, each example is a pair consisting of an input object (typically a vector) and the desired output value (also called the supervisory signal). A supervised learning algorithm analyzes the training data and produces an inferred function, which can be used for mapping new examples. An optimal scenario will allow for the algorithm to correctly determine the class labels for unseen instances. This requires the learning algorithm to generalize from the training data to unseen situations in a "reasonable" way (see inductive bias). This statistical quality of an algorithm is measured through the so-called generalization error.

The learning algorithms we are going to implement include: Support-vector machines; Logistic regression; AdaBoost; Random Forest; Gaussian Naive Bayes; Decision Trees; K-nearest neighbor algorithm; Neural networks (Multi-layer perceptron). We will only introduce the 3 methods below in detail for their significance.

K-nearest neighbor is a non-parametric method proposed by Thomas Cover used for classification and regression. The principle of which is straightforward. Vote by the target's nearest neighbors. To illustrate, take Fig. 5 (a) as an example. The green dot is the target waiting to be classified. The blue squares and red triangles are 2 categories. K-nearest-neighbor simply means the green dot would "ask" its nearest neighbors, for example, those in the solid circle, for advice, if most of them belong to a certain category, the green dot would follow the majority. In this case, the green dot would be a red triangle.

There are several key parameters. First is k, which means how many neighbors should green dot ask. In the aforementioned example, k is 3. When k is increased to 5, the dashed circle would represent the situation. Another one is the distance calculation method. Commonly used are Manhattan, Euclidean, Chebychev, and Minkowski. There is also the weight for the distance which means that the smaller distance would count more. In other words, the green dot would take the advice from his nearest neighbors more seriously than those far away. Another thing to mention that using K-nearest-neighbor needs to normalize the input data to have a good result since it relies on distances.

Support Vector Machine (SVM) was first proposed by Cortes and Vapnik in 1995. It has many unique advantages in solving small sample, nonlinear and high-dimensional pattern recognition, and can be extended to function fitting and other Machine learning problem. The support vector machine method is based on the VC dimension theory of statistical learning theory and the principle of structural risk minimization. According to the limited information, the complexity of the model (that is, the accuracy of the specific training sample, Accuracy) and the learning ability (ie The ability to identify any sample without error) seeks the best compromise to obtain the best generalization ability (or generalization ability). Linear classifier (in a sense, it can also be called perceptron) is the simplest and most effective form of classifier. In a linear classifier, you can see the idea of SVM formation and get in touch with many core concepts of SVM.

Take a classification problem with only two types of samples in a two-dimensional space as a small example, as shown in Fig. 5 (b). C1 and C2 are the two categories to be distinguished. Their samples in the two-dimensional plane are shown in the Figure 4 above. The straight line in the middle is a classification function, which can completely separate the two types of samples. Generally, if a linear function can completely separate the samples correctly, the data is said to be linearly separable, otherwise it is called non-linearly separable.What is a linear function? It is a point in one-dimensional space, a straight line in two-dimensional space, and a plane in three-dimensional space. It can be imagined like this. If you don't pay attention to the dimension of space, this linear function has a unified name—— Hyper Plane.

A linear function is a real-valued function (that is, the value of the function is a continuous real number), and our classification problem (such as the binary classification problem here-answering the question of whether a sample belongs to a category or not) requires Discrete output value. For example, use 1 to indicate that a sample belongs to category C1, and use 0 to indicate that it does not belong (not belonging to C1 means belonging to C2). At this time, you only need to simply add a threshold to the real-valued function. That is, the category attribution is determined by whether the value obtained when the classification function is executed is greater than or less than this threshold. And that is the core idea of SVM.

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean/average prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set. Random forests generally outperform decision trees, but their accuracy is lower than gradient boosted trees. However, data characteristics can affect their performance.

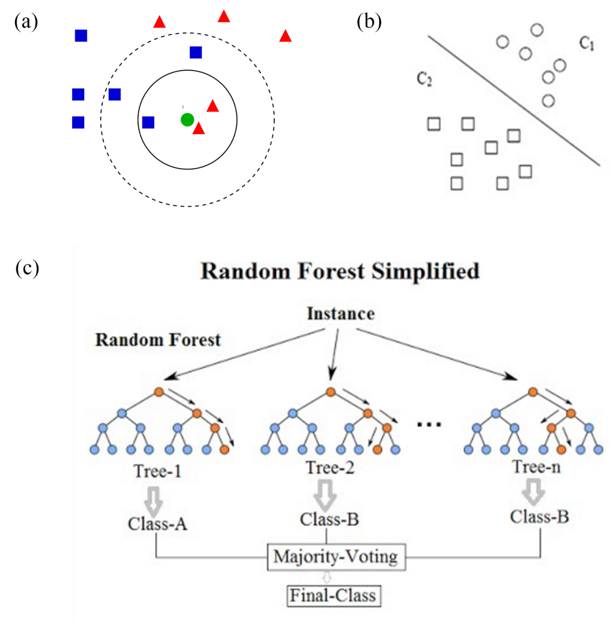


Fig. 5: (a) shows the k-nearest neighbor (KNN), (b) shows the support vector machine (SVM), and (c) shows the random forest.

The training algorithm for random forests applies the general technique of bootstrap aggregating, or bagging, to tree learners. Given a training set X = x1, ..., xn with responses Y = y1, ..., yn, bagging repeatedly (B times) selects a random sample with replacement of the training set and fits trees to these samples: For b = 1, ..., B:

1) Sample, with replacement, n training examples from X, Y; call these Xb, Yb.
2) Train a classification or regression tree fb on Xb, Yb.

After training, predictions for unseen samples x' can be made by averaging the predictions from all the individual regression trees on x':

$$\hat{f} = \frac{1}{B} \sum_{b=1}^{B} f_b\left(x'\right) \quad (2)$$

This bootstrapping procedure leads to better model performance because it decreases the variance of the model, without increasing the bias. This means that while the predictions of a single tree are highly sensitive to noise in its training set, the average of many trees is not, as long as the trees are not correlated. Simply training many trees on a single training set would give strongly correlated trees (or even the same tree many times, if the training algorithm is deterministic);

bootstrap sampling is a way of de-correlating the trees by showing them different training sets.

Additionally, an estimate of the uncertainty of the prediction can be made as the standard deviation of the predictions from all the individual regression trees on x':

$$\sigma = \sqrt{\frac{\sum_{b=1}^{B}\left(f_b\left(x'\right)-\hat{f}\right)^2}{B-1}} \tag{3}$$

The number of samples/trees, B, is a free parameter. Typically, a few hundred to several thousand trees are used, depending on the size and nature of the training set. An optimal number of trees B can be found using cross-validation, or by observing the out-of-bag error: the mean prediction error on each training sample xi, using only the trees that did not have xi in their bootstrap sample. The training and test error tends to level off after some numbers of trees have been fit.

The above procedure describes the original bagging algorithm for trees. Random forests differ in only one way from this general scheme: they use a modified tree learning algorithm that selects, at each candidate split in the learning process, a random subset of the features. This process is sometimes called "feature bagging". The reason for doing this is the correlation of the trees in an ordinary bootstrap sample: if one or a few features are very strong predictors for the response variable (target output), these features will be selected in many of the B trees, causing them to become correlated. An analysis of how bagging and random subspace projection contribute to accuracy gains under different conditions is given by Ho.

Typically, for a classification problem with p features, $\sqrt{p}$ (rounded down) features are used in each split. For regression problems the inventors recommend p/3 (rounded down) with a minimum node size of 5 as the default. In practice the best values for these parameters will depend on the problem, and they should be treated as tuning parameters.

2) Deep Learning: In addition to machine learning models, we also tried to use deep learning models to process a larger amount of data. Due to time series data, we planned to try two models: recurrent neural network (RNN) and Long Short-Term Memory (LSTM).

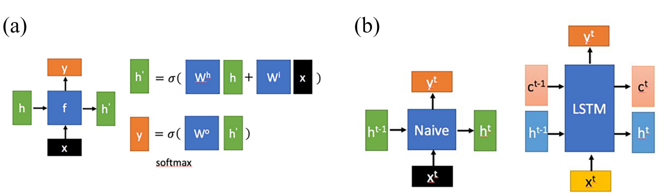


Fig. 6: (a) shows the structure of the Recurrent Neural Network (RNN) and (b) shows that of the Long Short-Term Memory (LSTM).

Recurrent Neural Network (RNN) is a neural network used to process sequence data. Compared with a general neural network, it can process data that changes in sequence and allow the information to persist. The main form of a general RNN is shown in Fig. 6 (a).

In LSTM, x is the input data in the current node state, and h represents the input data received from the previous node. y is the output in the current node state, and h' is the output data passed to the next node.

Besides, in order to solve the problem of gradient disappearance and gradient explosion during long sequence training, and to make the model perform better in longer sequences, we planned to try to use a special RNN: Long Short-Term Memory (LSTM).

Compared with ordinary RNN, the main difference of LSTM is shown in Fig. 6 (b).

Compared with RNN which has only one transfer state, LSTM has two transfer states, ct (cell state) and ht (hidden state).

There are three main stages inside LSTM:

1) "Forget". This stage is mainly to selectively forget the input from the previous node. To put it simply: "forget the unimportant and remember the important".
2) "Select to memory". This stage selectively "memorizes" the input of this stage. The main purpose is to select and memorize the input xt.
3) "Output". This stage will determine which will be regarded as the output of the current state.

Similar to the ordinary RNN, the output yt is often finally obtained through the change of ht.

III. Experiment

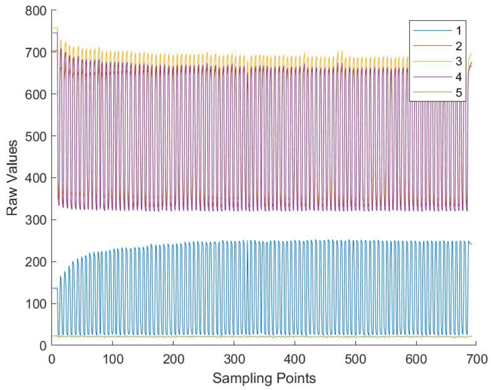
A. Data Collection

We selected four different types of the garbage when collecting data, which are plastic bottles, cans, paper boxes, and hard bottles. Hard bottles are rubbish like glass bottles. We collect data from five kinds of plastic bottles, two kinds of pop cans, one kind of paper boxes and two kinds of hard bottles.

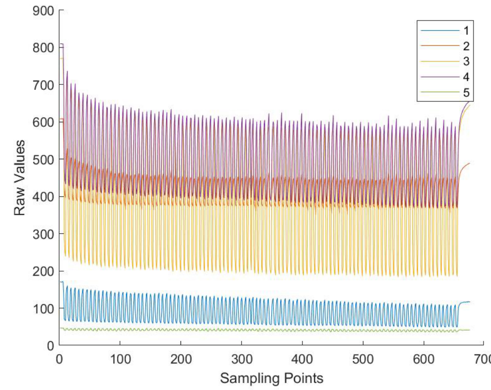
When collecting data, we only use half of the gripper to make measurements. We put the hanging gripper horizontally, put the garbage directly under the paw so that the garbage is aligned with the gripping position of the paw. Let the paws repeatedly press the fixed distance from the fixed height to collect data in the paw fiber feedback. The height of the half gripper relative to the workbench represents the width of the entire gripper when it is open. The paw moves down to the performance of a grab. Since the gripper are symmetrical, the data generated by two fingers is also symmetrical, so we can collect data with only one finger. We collect data in this way because it's much faster.

The movement of the gripper is accomplished by controlling the Dynamixel MX-64 servo. By setting the movement speed and target position of the servo in the MATLAB program, we can make the gripper move according to our expected trajectory. After the finger is deformed, which means it is in contact with the bottle, we then read the data of the five optical fiber channels in the finger through the Python program to realize the data collection.

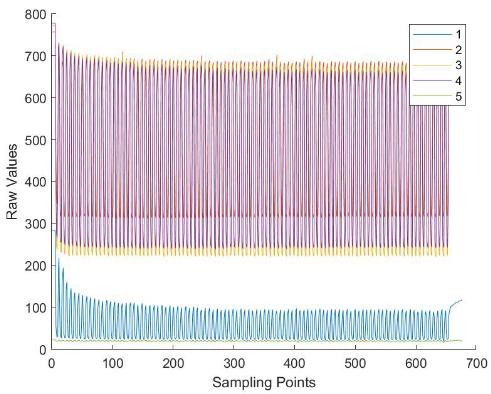
For each of the above bottles, we collected data 100 times. For the axial and radial shapes that have a relatively large



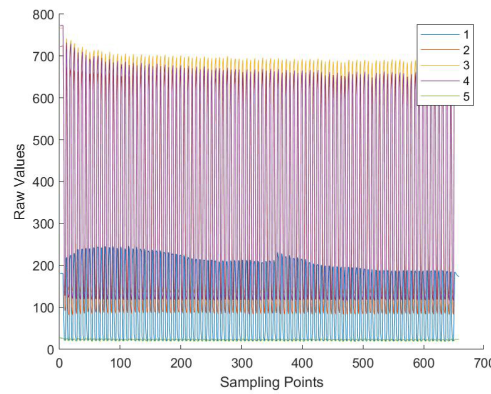
(a) AHHA Plastic Bottle



(b) MaiDong Plastic Bottle



(c) Metal Bottle



(d) JiaDuoBao Can

Fig. 7: Raw data for four kind of objects.

change bottle, we measure multi-group data by changing the grab position. It will eventually get the data when the bottle is in different positions. Selected data are shown in Figure 7. The total Data set include 10 kinds of objects, for each one of them grasps are conducted 100 times.

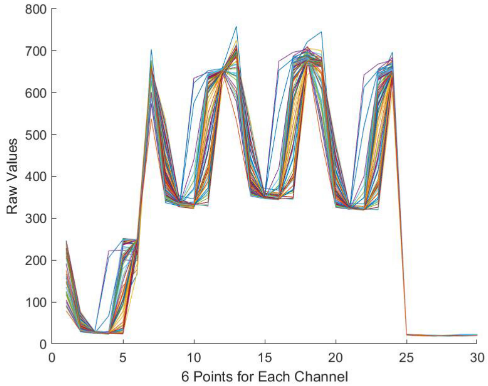
B. *Object Classification*

In this section, we experimented with 7 kinds of different models, as mentioned in the Methods. First, in order to use the raw data collected in the last section, we have to cut the 100 grasps into 100 parts for each data. The cut is done based on the assumption that each grasp takes the same time, which corresponds to 6 sample points. Then, for each grasp, there exist 5 channels. The raw values of the 5 channels are flattened and connected in series. After this procedure, Each grasp of each object contains 5*6=30 numbers. This could be shown in Figure 8. It can be seen that the difference of different kinds of the object could be easily distinguished, implying a great potential to be processed by machine learning methods. However, even though Figure 8a and 8b are both plastic bottles, the traits of these 2, look kind of different. It might mean that the shape of the object may influence more.

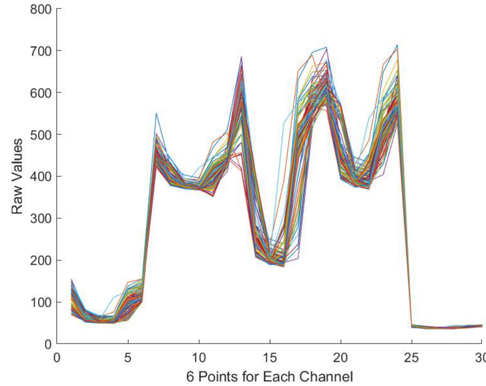
To testify the effectiveness of classification via gripper, we designed 4 kinds of experiments. In experiment 1, we used 5 kinds of plastics bottles as the data set, to testify the performance to classify plastic bottles. Among which, 80 percent are taken as train set and the rest are used as test set. The evaluation of the classification results can be seen in Table I. Almost all models reached accuracy of 1.0.

In experiment 2, we used 10 kinds of objects as the data set, which include 5 kinds of plastic bottles used in the former experiment, 2 kinds of rigid bottles, 2 kinds of aluminum cans and one kind of paper box, to see how good it can classify different objects of shapes and textures. Among which, 80 percent are taken as train set and the rest are used as test set. The evaluation results can be seen in Table I. Almost all models reached accuracy of 1.0. Implying that, the models could handle much different objects.

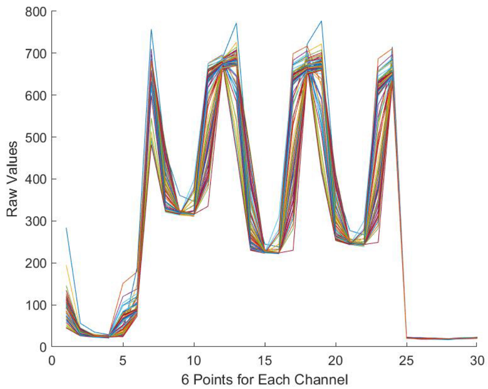
In experiment 3, we used 3 kinds of objects as the data set, which include 1 kind of plastic bottles, 1 kinds of rigid bottles, 1 kinds of aluminum cans. Noted that, all 3 objects have the same diameters. This experiment was conducted to prove whether the model could classify objects in the absence of shape, but only through the softness of the object. Among which, 80 percent are taken as train set and the rest are used as test set. The evaluation results can be seen in Table I. Almost all models reached accuracy of 1.0. This implies the softness conception was as expected.



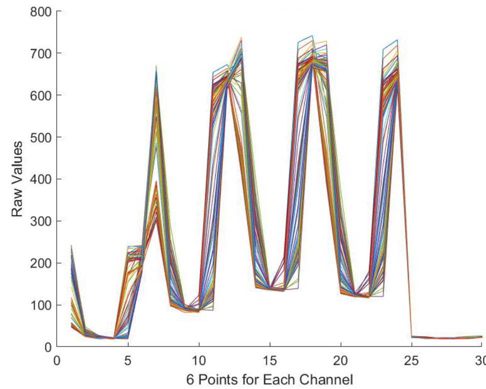
(a) AHHA Plastic Bottle



(b) MaiDong Plastic Bottle



(c) Metal Bottle



(d) JiaDuoBao Can

Fig. 8: Processed data for four kind of objects.

TABLE I: Accuracy and field test results of different models.

Algorithms	Experiment 1	Experiment 2	Experiment 3	Experiment 4	Field Test
Gaussian Naïve Bayesian	1	0.96	1	0.0037	F
Decision Tree:	1	1	0.983	0.32962	T
AdaBoost	1	0.82	1	0.00742	T
K-nearest neighbor	1	1	1	0.09259	F
Random Forest	1	1	1	0.32962	F
Logistic Regression	1	1	1	0.23333	F
NN (MLP)	0.99	0.995	1	0	F

In experiment 4, we used the same data as training data. However, in test data, we introduced 3 brand new kinds of object belonging to plastic, rigid bottles and cans. This experiment was conducted to check the performance of the model when faced with untrained samples, to see how well it could handle new situations. The evaluation results can be seen in Table I. Almost all models have accuracy of 0. This implies that when faced with untrained object, the model behaves poorly.

Finally, to testify the model, we conducted a field test on the platform. The gripper grasps an "AHHA" plastic bottle via 2d location based on YOLOv5. The gripper transfer the data back to the computer to classify the grasped object. The results of the field test is shown in Table I. Decision Trees and AdaBoost classify it perfectly. We believe these 2 models might have the greatest potential. However, more experiments and validation could be future conducted to reveal more details of these models.

All the experiments conducted in this section revealed some interesting traits of the models, implying great potential of classifying obejcts via the soft gripper proposed in this work. More in depth analysis is given in the next section.

IV. Discussion

A. Data Collection

In the literature [10], the data obtained by the flexible gripper is discrete. All the data is collected by the optical fiber sensor after the gripper grasps the target. In this article, our experimental data is time-series. During the downward movement of the gripper, our optical fiber sensor feeds back information at a fixed frequency. Intensity of the transmitted light (a1,a2,a3,a4,a5) was gathered in this process. Save the data collected by the five optical fibers

In terms of results, our gripper was sensitive to the data obtained when grasping objects of different shapes and materials. If we take Maidong Bottle as an example, it was at the middle sampling point when the gripper grasped the deepest. The amount of change in the polyline shown in Fig. 8 would become significantly larger. It means that it represented a larger shape variable. It corresponded to the shape of the bottle. Since the data measured is on the condition that the plane of the side of the bottle is facing upwards, the finger first touches the entire plane when grasping, and then continues to deform and wrap around the side. Therefore, when the gripper contacted with the bottle, the deformation of the second and fourth sampling points was small, and the deformation of the third sampling point was significantly increased.

B. Object Classification

In this paper, we tried seven learning algorithms to realize the object classification, which are Gaussian Naive Bayesian, Decision Tree, AdaBoost, K-nearest Neighbor, Random Forest, Logistic Regression, and Neural Network. When the dimension of the object is kept consistent, the classification accuracy is still very high, which proves that we can successfully recognize the softness information of the object.

Besides, in the experiments, the algorithms all have good performance for the learned samples, but poor performance for the unlearned samples, which shows that the generalization performance of the model is still relatively low. It also should be noted that, in the field test, Decision Tree and AdaBoost behaves better. Therefore, in the future, we will use the gripper to collect more data in different configurations to realize data augmentation while improve the generalization ability of the model. Meanwhile, we will also add visual aids to increase the accuracy of object classification.

V. Conclusion

In conclusion, this paper proposed a learning-based object classification strategy, using tactile sensing data collected by optpelectronically adaptive finger to identify different objects. This strategy has been well validated in the experiment. The results show that it can distinguish different materials to some extent, which provides a new idea for object recognition and garbage classification.

This work is a preliminary practice of the tactile sensing method based on optical fiber in the application of item classification. In future research, we will expand the types of data collection, optimize the methods of data collection, and improve the generalization ability and accuracy of the model.

Acknowledgment

We would like to express our thanks to Professor Chaoyang Song for project theme design and guidance. and we would also like to thank Linhan Yang, Fang Wan, Xiaobo Liu, Yu Jie for their kind help and advice on experiment design and presentation.

References

[1] M. Tian, B. Pu, Y. Chen, and Z. Zhu, "Consumer's Waste Classification Intention in China: An Extended Theory of Planned Behavior Model," *Sustainability (Switzerland)*, 2019.

[2] S. Liang and Y. Gu, "A deep convolutional neural network to simultaneously localize and recognize waste types in images," *Waste Management*, 2021.

[3] F. A. Azis, H. Suhaimi, and E. Abas, "Waste Classification using Convolutional Neural Network," in *ACM International Conference Proceeding Series*, 2020.

[4] S. K. Noh, "Recycled clothing classification system using intelligent IoT and deep learning with ALExNet," *Computational Intelligence and Neuroscience*, 2021.

[5] A. Shaukat, Y. Gao, J. A. Kuo, B. A. Bowen, and P. E. Mort, "Visual classification of waste material for nuclear decommissioning," *Robotics and Autonomous Systems*, 2016.

[6] J. Lindermayr, C. Senst, M. H. Hoang, and M. Hägele, "Visual classification of single waste items in roadside application scenarios for waste separation," in *50th International Symposium on Robotics, ISR 2018*, 2018.

[7] S. Rodrigues, G. Martinho, and A. Pires, "Waste collection systems. Part A: A taxonomy," *Journal of Cleaner Production*, 2016.

[8] L. R. Kambam and R. Aarthi, "Classification of plastic bottles based on visual and physical features for waste management," in *Proceedings of 2019 3rd IEEE International Conference on Electrical, Computer and Communication Technologies, ICECCT 2019*, 2019.

[9] Z. Yang, S. Ge, F. Wan, Y. Liu, and C. Song, "Scalable Tactile Sensing for an Omni-adaptive Soft Robot Finger," in *2020 3rd IEEE International Conference on Soft Robotics, RoboSoft 2020*, 2020.

[10] L. Yang, X. Han, W. Guo, F. Wan, J. Pan, and C. Song, "Learning-based optoelectronically innervated tactile finger for rigid-soft interactive grasping," *IEEE Robotics and Automation Letters*, 2021.

[11] A. Drimus, G. Kootstra, A. Bilberg, and D. Kragic, "Design of a flexible tactile sensor for classification of rigid and deformable objects," *Robotics and Autonomous Systems*, 2014.

[12] Weiss. (2020) Weiss robotics tactile sensor. [Online]. Available: http://www.weiss-robotics.de/en.html

[13] L. Yang, F. Wan, H. Wang, X. Liu, Y. Liu, J. Pan, and C. Song, "Rigid-soft interactive learning for robust grasping," *IEEE Robotics and Automation Letters*, 2020.

[14] L. Chin, J. Lipton, M. C. Yuen, R. Kramer-Bottiglio, and D. Rus, "Automated recycling separation enabled by soft robotic material classification," in *RoboSoft 2019 - 2019 IEEE International Conference on Soft Robotics*, 2019.

[15] G. Li, S. Liu, L. Wang, and R. Zhu, "Skin-inspired quadruple tactile sensors integrated on a robot hand enable object recognition," *Science Robotics*, 2020.

[16] D. Z. Stupar, J. S. Bajic, L. M. Manojlovic, M. P. Slankamenac, A. V. Joza, and M. B. Zivanov, "Wearable low-cost system for human joint movements monitoring based on fiber-optic curvature sensor," *IEEE Sensors Journal*, 2012.