Comparing Single Touch to Dynamic Exploratory Procedures for Robotic Tactile Object Recognition

Elliot Kirby[1], Rodrigo Zenha[1] and Lorenzo Jamone[1]

***Abstract*—Recognizing objects by touch is a very useful skill for robots to be employed in both structured and unstructured environments. While in some applications it is useful to recognize an object from a single touch, in other scenarios specific robot movements can be used to obtain more information about the object, making recognition easier. In this paper, we show how this can be obtained through the combination of: (i) a recently developed tactile sensor that measures both normal and shear forces on multiple contact points, and (ii) an exploratory procedure that involves dynamic shaking of the gripped object. We compare the recognition accuracy in three conditions: static (i.e. single touch), short dynamic (i.e. using a small fraction of the exploratory procedure), and dynamic (i.e. using the entire exploratory procedure). We report experiments with six different machine learning techniques, and several combinations of tactile features, to recognize ten objects. Overall, our results demonstrate that: (i) the sensor we use is well suited for recognizing grasped objects with high accuracy, and (ii) the dynamic exploratory procedure provides a 38% improvement over single touch recognition. We make our data and code publicly available, to encourage reproduction of our results.**

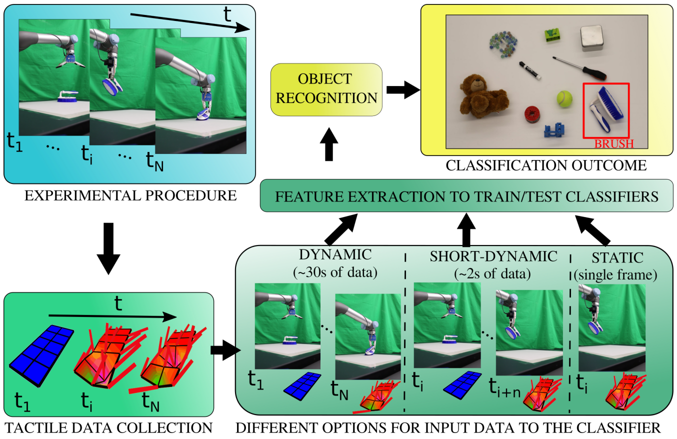
t
t_1 ... t_i ... t_N
EXPERIMENTAL PROCEDURE
OBJECT RECOGNITION
BRUSH
CLASSIFICATION OUTCOME
FEATURE EXTRACTION TO TRAIN/TEST CLASSIFIERS
DYNAMIC (~30s of data)
SHORT-DYNAMIC (~2s of data)
STATIC (single frame)
t_1 t_N t_i t_{i+n} t_i
t
t_1 t_i t_N
TACTILE DATA COLLECTION
DIFFERENT OPTIONS FOR INPUT DATA TO THE CLASSIFIER

Fig. 1. High-level overview of the collection of tactile data being used to inform a classification model, which predicts the grasped object as an output. In this example the Brush is the object used.

I. Introduction

Tactile perception is crucial for robots to interact safely and effectively with both structured and unstructured environments [1], [2]. In particular, robotic manipulators can leverage tactile sensors to estimate physical properties of the objects being manipulated [3], and even to achieve tactile object recognition [4], [5]. Although visual sensing can be used to recognize objects [6], [7], common issues such as low light conditions or occlusions can deteriorate the accuracy of the recognition. Tactile perception is a valuable alternative (or complementary measure) since it provides direct physical feedback about the objects being manipulated.

For tactile perception to happen, the robot must generate the physical interaction with the objects, i.e. active [8] or interactive perception [9]. This typically involves a dedicated Exploratory Procedure (EP) that requires the robot to probe a surface [10] or an object [11], or to hold the object within a grasp [12], [13], either keeping it static or performing a set of controlled actions (e.g. squeezing). However, objects are often strategically placed so as to maximize the regions of contact with the sensor. It is unclear if these strategies are fit for robots that operate autonomously in semi-structured environments, performing different types of movements (e.g. grasping, lifting, transporting, at different speeds and with different accelerations), and often dealing with unstable grasps, unexpected slips, and high variability in the gripper-object physical interaction. In fact, if this variability was reflected in the sensor measurements (i.e. by leveraging a tactile sensor that can reliably detect rich contact information), and if the data coming from such interactions was present in the training data, object recognition models could ideally perform much better in such real-world scenarios.

Therefore, to overcome the limitations of previous work, we propose to train object recognition models using data: (i) extracted from a recently developed tactile sensor; (ii) during an exploratory procedure that involves dynamic movements of a robotic gripper. One key objective of this paper is to show that if data is collected from such a dynamic procedure, then we have access to more information that is helpful to better recognize objects. However, to take full advantage of this we need: (i) a sensor that can collect a large set of contact information; (ii) appropriate features that represent such dynamic information. Therefore: (i) we use the recently developed uSkin tactile sensor [14], which is able to detect both normal and shear forces on multiple contact points; (ii) we use an exploratory procedure that involves grasping, lifting, shaking and releasing each object, and we compare the effectiveness of a large set of tactile features.

This paper reports the following:

- an analysis of different tactile features that can be extracted from the measured shear and normal forces through a robotic exploratory procedure in a realistic

Manuscript received: September, 9, 2021; Revised November, 19, 2021; Accepted January, 31, 2022.

This paper was recommended for publication by Editor Markus Vincze upon evaluation of the Associate Editor and Reviewers' comments.

Work partially supported by the EPSRC UK through projects NCNR (EP/R02572X/1) and MAN^3 (EP/S00453X/1).

[1] E. Kirby, R. Zenha and L. Jamone are with ARQ (Advanced Robotics at Queen Mary), School of Electronic Engineering and Computer Science, Queen Mary University of London, UK e.kirby@se20.qmul.ac.uk, {r.neveszenha, l.jamone}@qmul.ac.uk

Digital Object Identifier (DOI): see top of this page.

unstructured environment (results in Table I);

- an evaluation of tactile object recognition capability when data representative of the full exploratory procedure is provided, compared to alternative scenarios in which only a short part of the exploration (or, a single tactile frame) is considered (results in Fig. 5);
- evidence of how a trained classifier can be used in real-world settings, by evaluating the generalization to grasp poses not seen during training (results in Fig. 6).

II. Related Work

Exploratory Procedures (EPs) have been used in previous works to enable robotic tactile perception, either in the form of probing or grasping/manipulation.

The works in [10] and [15] use probing EPs to scan the surface of objects and identify their materials, using either piezoelectric [10] or capacitive [15] tactile sensors. Both works use hand-crafted features to identify characteristics of the material, such as the frictional coefficients or cracks and bumps in the surface, and they achieve accuracies of 89% and 99% respectively. An alternative probing EP is a simple touch of a tactile sensor as shown in [12], where an array of piezoresistive tactile sensors with high spatial resolution is used. A recently popular approach to tactile object recognition is to treat pressure images obtained from tactile sensors as traditional RGB images and to train a convolutional neural network (CNN) to identify the object through deep learning [12], [16], typically by using a single high resolution tactile image; this strategy is even more natural when using camera-based tactile sensors [17].

Another type of EP is to use an actuated gripper or hand to physically grasp an object and hold it in position or conduct a fixed routine whilst the object is gripped (e.g. squeezing [18]). If the sensor mounted to the gripper has sufficiently high spatial resolution, then computer vision techniques and deep learning can again be used, as shown in [18]. An alternative approach is explored in [11], where a multi-fingered robot hand equipped with a distributed version of the uSkin tactile sensors [19] gathers tactile information from all sides of an object; several grasps are used in combination to identify the object with deep learning; when only a single grasp is used, an accuracy of 49% is achieved, however using a combination of grasps increases the accuracy up to 88%. In a later work [13] the same authors show an even increased accuracy of 95%; they also show how a dynamic hand exploration provides more (and useful) data than a single grasp; however, the data used for classification is not only tactile, but it includes finger joints positions and force sensors in the fingertips; overall, this is a complex and specialized robotic setup that might not be easily available to researchers and companies, and it might not fit standard applications in industry.

Unlike previous works which use highly controlled or specialized EPs and robotic setups, our EP could be easily included within routine robot operations (i.e. pick and place of objects) in unstructured environments (i.e. objects are placed on the table and autonomously picked by the robot, using depth sensing). The sensor we use (uSkin) has lower spatial resolution than other pressure sensitive (e.g. resistive [18] or camera-based [17]) tactile sensors; however, we believe that the ability of uSkin to measure not just the normal forces but also the shear forces on each contact point can prove valuable when using a dynamic exploratory procedure (i.e. lifting and shaking). For recognition, we favour classifiers with hand-crafted features over deep learning approaches as it is expected that higher accuracies can be achieved with the few training samples available.

III. Methodology

An overview of our method is shown in Fig. 1. Our objective is to recognize an object after it has been picked by a robot gripper, during normal robot operations, using tactile data. To this end, we can either consider tactile data from the entire EP (dynamic classification), a short fraction of the EP (short dynamic classification) or a single tactile frame (static classification). A training strategy is proposed to evaluate the model generalization capabilities to previously unseen object poses. Next we describe the tactile sensor, experimental set up and data collection process.

A. Tactile Sensor

The uSkin sensor [14] (Fig. 2 left) allows to measure signals that proportionally relate to the shear and normal forces individually applied to each of its 18 taxels, distributed in a 3x6 layout. The forces observed by each taxel are measured through changes of a magnetic field caused by the displacement of small magnets held within a malleable silicone rubber dome; more details on the working principle, at the level of the single taxel, are provided in previous works [20], [21]. A piece of fabric placed across the surface of the sensor creates an artificial skin which is sensitive to stretching and friction. At each instance, $1 \leq t \leq N$ (where N corresponds to the last available sample instance), each taxel, $p \in \{1, \cdots, 18\}$, measures 3 signals, $[x_{p,t},\, y_{p,t},\, z_{p,t}]$, at a frequency of 180Hz; signals $x_{p,t}$ and $y_{p,t}$ relate to the shear forces applied to it; $z_{p,t}$ relates to the normal forces. Fig. 2 right, shows a graphical representation of a complete (18 taxels) tactile imprint; the signals $z_{p,t}$ measured by each taxel are represented by the vertical motion of the corresponding vertices; the red vectors represent the direction of the contact with each taxel (extracted from $x_{p,t}$ and $y_{p,t}$).

B. Experimental Setup

We consider the robotic setup shown in Fig. 3, composed of a 6 DOFs arm (UR5) with a 1 DOF 2-jaws parallel gripper (EZGripper). One uSkin tactile sensor is mounted on one jaw of the gripper. Although two sensors (i.e. one on each jaw) could provide more tactile information, that could be useful for better object recognition, using one sensor only has the advantage of reducing costs (of both purchase and maintenance) and complexity (e.g. cabling), and might therefore be a favourable choice in practical applications. A Kinect2 depth camera is fixed perpendicularly above the robot workspace.

C. Data Collection

To collect the tactile data, a robotic Exploratory Procedure (EP) is performed on 10 objects with distinct physical and geometric properties (Fig. 4). The EP includes grasping, lifting, shaking and finally placing each object back in its original pose (total duration of 30 seconds); the whole procedure has been programmed and executed using the GRIP software framework [22] and it is described in detail in [23]. In our data collection, each objects is placed in 5 different poses, within a 50cm x 50cm workspace, and a total of 15 grasps are executed for each pose, leading to a total of 75 EPs for each object. The grasps are generated automatically from depth sensing, using GGCNN2 [24].

Because during the EP objects can slip from the gripper, and we collect data only when the object is within the gripper, the overall amount of data that we collect for each object is different, since some objects slip more frequently than others. For example, we have a total of about 33K tactile frames for the Brush, about 130K for the Screwdriver and the Spoolsolder, and about 160K-240K for each of the other objects. We discuss in the results (Sec. V-B) how this data imbalance may affect the recognition results.

The dataset is freely available here:
https://github.com/ARQ-CRISP/tactile_object_recognition.

D. Feature Extraction

This section describes the tactile features which are extracted from the data collected during the EP which are used during the classification stage. As shown in V-A, each are extracted either form the detected shear forces, normal forces, or both.

Data statistical descriptions: For each component x, y, z, the data is grouped in 3 groups: the raw data across all taxels ($\mathbf{w}_1$); the sum of readings of each taxel, for each frame ($\mathbf{w}_2$); or averaged by taxel ($\mathbf{w}_3$). Once the data is grouped (e.g. considering both normal and shear forces), their minimum, f_1, maximum, f_2, and standard deviation values, f_3, are extracted as features:

$$\mathbf{w}_1 = [x,y,z]_{\substack{p=1:18,\\ t=1:N}}; \quad \mathbf{w}_2 = [\sum_{p=1}^{18} x_p, \sum_{p=1}^{18} y_p, \sum_{p=1}^{18} z_p]_{t=1:N}$$

$$\mathbf{w}_3 = [\sum_{t=1}^{N} x_t/N, \sum_{t=1}^{N} y_t/N, \sum_{t=1}^{N} z_t/N]_{p=1:18};$$

$$[std(\mathbf{w}_{1:3}), max(\mathbf{w}_{1:3}), min(\mathbf{w}_{1:3})] = [f_1, f_2, f_3](\mathbf{w}_{1:3}).$$

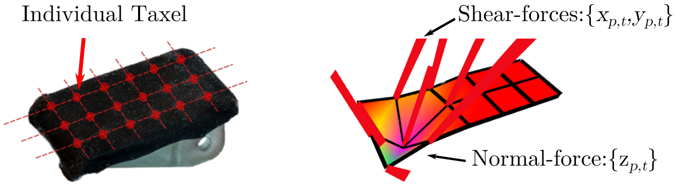
Individual Taxel
Shear-forces:$\{x_{p,t}, y_{p,t}\}$
Normal-force:$\{z_{p,t}\}$

Fig. 2. The uSkin sensor (left) and a visualization of 3D tactile readings (right) when the sensor is subject to a contact. For each taxel, the deformation of the surface reflects the normal force, and the direction of red vector represents the shear forces.

Ratio of normal and shear forces: Two approaches are considered to characterise frictional behaviour; the first (f_4) looks at the ratio between the maximum averaged normal force experience by a single taxel and magnitude of the maximum of the averaged shear forces (by taxel); the second (f_5) looks at the ratio of the forces at the moment when the sum of the normal forces across all taxels is maximal, t_{max}.

$$f_4(\mathbf{w}_3) = \frac{max(\mathbf{w}_3(z))}{\sqrt{max(\mathbf{w}_3(x))^2 + max(\mathbf{w}_3(y))^2}};$$

$$f_5(\mathbf{w}_2) = \frac{\mathbf{w}_2(z, t_{max})}{\sqrt{\mathbf{w}_2(x, t_{max})^2 + \mathbf{w}_2(y, t_{max})^2}};$$

$$t_{max} = \underset{t}{\operatorname{argmax}}\, \mathbf{w}_2(z, t).$$

Shear forces correlation: For $t = t_{max}$, the shear forces at each taxel, $\mathbf{w}_1(x, y)$ are transformed into polar co-ordinates, with angle ϕ, and magnitude r. ϕ is then used to determine the quadrant where each signal rests. We consider two signals to be strongly correlated if they live in the same quadrant, weakly correlated if they live in adjacent quadrants, or non-correlated if they live in opposite quadrants. We attempt to characterise objects with different shapes and different frictional characteristics following two approaches:

- **Proximal shear forces correlation** ($f_6(\mathbf{w}_1, \mathbf{w}_2)$): The number of taxels, adjacent to compressed taxels, that are strongly, weakly or non-correlated to the compressed taxels.
- **Overall shear forces correlation** ($f_7(\mathbf{w}_1, \mathbf{w}_2)$)**:** Extracting two binary flags that indicate if the majority of shear forces detected by all taxels rest in the same or adjacent quadrants (evidence of parallel forces), opposite quadrants (evidence of symmetric forces), or a mixture of both.

Squared Force Components in Phases: The EP is divided into windows of 0.5 seconds each (n tactile frames). For each window the individual force components are squared and

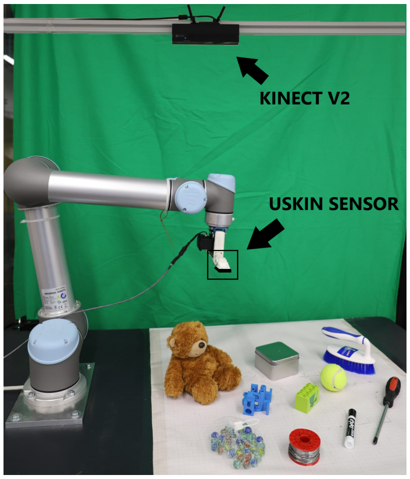

Fig. 3. Experimental setup for data collection: the Kinect v2 depth sensor is used to automatically locate each object, which is then picked by the UR5 robot arm with the EZGripper (equipped with the uSkin sensor).

averaged ($\mathbf{w}_4$):

$$\mathbf{w}_4 = [\sum_{t=1}^{n} x_t^2/n, \sum_{t=1}^{n} y_t^2/n, \sum_{t=1}^{n} z_t^2/n, \cdots ,$$
$$\cdots , \sum_{t=N-n}^{N} x_t^2/n, \sum_{t=N-n}^{N} y_t^2/n, \sum_{t=N-n}^{N} z_t^2/n]_{p=1:18}.$$

These windows are then grouped into three key phases of the EP; “Pick & Lift” ($\leq$3 seconds); “Raise” (3-11 seconds) and “Shake & Place” ($\geq$11 seconds). For each phase the corresponding average and standard deviation are extracted as features ($f_8(\mathbf{w}_4)$ and $f_9(\mathbf{w}_4)$, respectively).

Stability description: We start by determining if substantive changes occur between each consecutive frames. To do this, measurements from consecutive frames are subtracted, $\mathbf{w}_1(t) - \mathbf{w}_1(t-1)$, and compared to a predetermined tolerance value. We proceed to extract the number of consecutive frames where changes in tactile data is observed, $f_{10}(\mathbf{w}_1)$.

EP data length: In an attempt to capture objects which were regularly dropped at a similar stages, the length of each EP is considered: $f_{11} = N$.

Number of taxels compressed: When the normal force, $\mathbf{w}_1(z)$, measured at each taxel is greater than a predetermined threshold (determined experimentally), that taxel is considered to be compressed. Two approaches were taken for extracting features based on compressed taxels: (1) extracting the total number of taxels compressed at $t = t_{max}$ ($f_{12}(\mathbf{w}_1, \mathbf{w}_2)$); (2) the EP is broken into windows of equal length in which the average number of compressed taxels is recorded. The maximum ($f_{13}(\mathbf{w}_1, \mathbf{w}_2)$), mode ($f_{14}(\mathbf{w}_1, \mathbf{w}_2)$) and standard deviation ($f_{15}(\mathbf{w}_1, \mathbf{w}_2)$) across all windows are extracted as features.

Initial Contact: The first 0.5 seconds of the EP is broken into two equally sized windows. For each window, $f_{10}(\mathbf{w}_1)$ and $max(\mathbf{w}_3(z))$ (taxel that experiences the largest normal force, in average) are recorded. The computed difference of the averaged forces for each window is obtained. The same operation is performed for $f_{10}(\mathbf{w}_1)$ values. Finally, a binary flag is also extracted indicating whether $max(\mathbf{w}_3(z))$ changes between windows. A feature, $f_{16}(\mathbf{w}_1, \mathbf{w}_3)$, is obtained from the combination all of the previous.

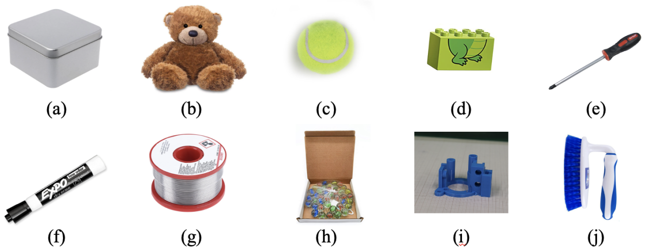

Fig. 4. Images of the 10 objects used for classification. From left to right: (a) metal box, (b) teddy bear, (c) tennis ball, (d) Lego Duplo block, (e) screwdriver, (f) marker pen, (g) spool solder, (h) marble net (cardboard box not included), (i) 3D printed adversarial, (j) brush.

E. Classifiers Used

We compare six classifiers (available from the Scikit-learn python packages [25]) in terms of overall recognition accuracy:

- **K-Nearest Neighbour Classifiers (with either K=1, K=3 or K=5)**: Classifiers which assign new samples to the same class as the closest neighbour(s) available from training samples in the attribute space [26].
- **Random Forest Classifier**: An ensemble learning algorithm which creates many binary decision trees based on random sampling of the training data using subsets of features. The classification of new samples is based on the most popular class observed across all trees [27]. For this paper the maximum number of trees is fixed at 100 and the maximum depth is 7.
- **Gaussian Naïve-Bayes Classifier**: A classifier which relies on probability theory and Bayes’ rule. The Gaussian Naïve Bayes (GNB) classifier assumes each feature is independently generated from a Gaussian distribution which is dependent on each object [28].
- **Linear SVC Classifier**: A support vector classifier (SVC) which maximises the distance between the linear boundary and the closest samples from each class [29].

IV. Experiments

In this section we describe a set of experiments that aim at: identifying the best classifier, features and exploration strategy (Sec. IV-A), and testing the generalization capabilities of the best performing model (Sec. IV-B).

A. Training and Validation

We perform different training and validation strategies with 3 objectives: identify the best classifier; select the best combination of tactile features; identify the best exploration approach. For all classifiers, the dataset was divided into a training dataset and a validation dataset with a 70:30 split respectively. This split is stratified based on objects to ensure a consistent ratio regardless of the total number of samples available for an object.

To evaluate the performance of each classifier, the accuracy metric is considered:

$$Accuracy = \frac{1}{n_{objects}} \sum_{object=1}^{n_{objects}} \frac{TP_{object} + TN_{object}}{P_{object}} \times 100;$$

where, $n_{objects}$, are the total number of objects, $TP_{object} + TN_{object}$ are the number of correctly classified (true positive and true negative) samples per object, and P_{object} are the total number of samples fed to the model, per object. For each classifier the random split of training and validation data was repeated 10 times with the overall accuracy of the classifier being the average achieved across all repetitions. The samples used to train and validate the classifiers combine the features described in Sec. III-D, extracted in 3 different scenarios: **Dynamic**, with features generated from the whole EP; **Short Dynamic**, with features generated from a period of two seconds selected randomly from the EP; **Static**, with features generated from a single frame selected randomly from the EP.

For each of these 3 scenarios, the validation phase was split into two key stages. The initial stage validates individual features in isolation, for each classifier, whilst also assessing the impact of the different force components (shear vs normal). In the second stage, the most promising features were combined iteratively to create a subset of features with which to train our classifiers; we then identify the classifiers that produce the best overall accuracies across objects.

B. Generalization Testing

The objective of generalization testing is to evaluate how the classifier might behave in a real-world scenario where objects are not placed in a previously observed poses, i.e. an object being grasped in a different way from what was recorded during training. For all objects, one of the five poses is withheld from the training dataset to act as a test dataset. Only the best classifier and the best combination of tactile features obtained during the validation phase are considered IV-A.

V. Results and Discussion

Next we discuss the performance of the different classifiers and feature combinations. Each is evaluated with regards to the accuracy in recognizing objects from validation and test datasets described in IV.

A. Training and Validation Results

First, each feature was tested in isolation. The accuracy achieved by each individual feature, with the best classifier (among the six classifiers compared) and using the full dynamic approach is shown in Table I. Then, to select the best set of features, the following criteria was used: for each type of feature, if the combination of shear and normal forces did not increase the overall accuracy more than a predefined value (i.e. 10% in our case), then only the best performing individual component was selected (i.e. either shear or normal). This was done to minimise the number of features and to keep the complexity of the model bounded: ideally, this helps to avoid overfitting and to obtain better generalisation, and it reduces the computation time needed to train the model and to generate predictions. The selected features (that will be used in the final classifier) are highlighted in Table I. In general, while the combination of shear and normal consistently leads to better accuracy, shear-only typically outperforms normal-only, and in some cases shear+normal does not bring much advantage with respect to shear-only (i.e. less than 10%): in those cases, we select the shear-only features, to reduce the overall computational complexity of the model.
For the short-dynamic approach, the identified best performing features are the same as in the dynamic approach (although in this case f_8 and f_9 are extracted over the entire short-dynamic window, and not during a specific phase of the EP); for the static approach, features $f_1(\mathbf{w}_{2:3})$, $f_3(\mathbf{w}_{2:3})$, f_4 and f_{12} have been selected as the best ones.

Fig. 5 shows boxplots for the peak accuracy achieved in classification of all ten objects using each type of classifier with full dynamic approach, short dynamic approach and static approach, respectively. In all approaches, the RF classifier stands above the other ones, with an average accuracy of either 72%, 62% or 52% (depending on the approach) and very low standard deviation (less than $\pm 2\%$), while the GNB shows to be the lowest performing. Regardless the classifier used, the full dynamic approach always shows the highest classification accuracy, followed by the short dynamic approach, and then by the static approach.

Since the best results were achieved using the RF, this model will be considered for the generalization test. For this classifier, the average computation time (i.e. the time between the instant in which the raw input data is measured by the sensor and the instant in which the classification output is generated) is 189ms, composed of 187ms for computing the features (for the Dynamic and Short-Dynamic case) and 2ms for generating the prediction, running on a MacBook machine with a 2GHz Quad-Core Intel Core i5 CPU and 16GB of memory. For the Static case, the average time for computing the features is shorter, only 60ms, since there are less features.

B. Generalization Testing Results

The generalization results are presented in Fig. 6. The confusion matrices show the performance of the RF classifier when one of the object poses (from Pose 1 to Pose 5) is not used for training, but only for testing. Only the best performing selected features highlighted in Table I are used. The last confusion matrix shows the results when all the poses are used for both training and testing: this is the same of the RF result in Fig. 5(a), but shown "per-object" rather than averaged. The confusion matrices indicate that, for some geometrically uniform objects, such as the Tennis Ball, the

TABLE I
A SUMMARY OF THE MAXIMUM ACCURACY ACHIEVED ACROSS ALL CLASSIFIERS USING ISOLATED FEATURES AND FULL DYNAMIC INFORMATION FOR ALL 10 OBJECTS

Features	**Peak accuracy across all classifiers (%)**		
	Shear (S)	***Normal (N)***	***(S) & (N)***
Raw Taxels ($\mathbf{w}_1$)			
f_1 / f_2 / f_3	32/ 31/ 32	22/ 26/ 21	44[a]/44[a]/44[a]
f_{10}			27[a]
Sum of Forces ($\mathbf{w}_2$)			
f_1 / f_2 / f_3	32[a]/29/ 30[a]	20/ 22/ 18	37/ 43[a]/38
f_5			23
Averaged Taxels ($\mathbf{w}_3$)			
f_1 / f_2 / f_3	33/ 27/ 34	26/ 20/ 20	47[a]/42[a]/44
f_4			23[a]
Squared Forces ($\mathbf{w}_4$)			
Pick & Lift: f_8 / f_9	32[a] / 29	20 / 20	42 / 38
Raise: f_8 / f_9	38[a] / 38	29 / 24	45 / 42
Shake&Place: f_8 / f_9	36[a] / 32	30 / 24	45 / 42
EP Data Length: f_{11}	20		
SF Correlation: f_6 / f_7			14 / 19
Taxel Compression			
f_{12}		33	
f_{13} / f_{14} / f_{15}		33[a]/31/ 22	
Initial Contact: f_{16}	44	32	52[a]

[a] Features selected for final classifier.

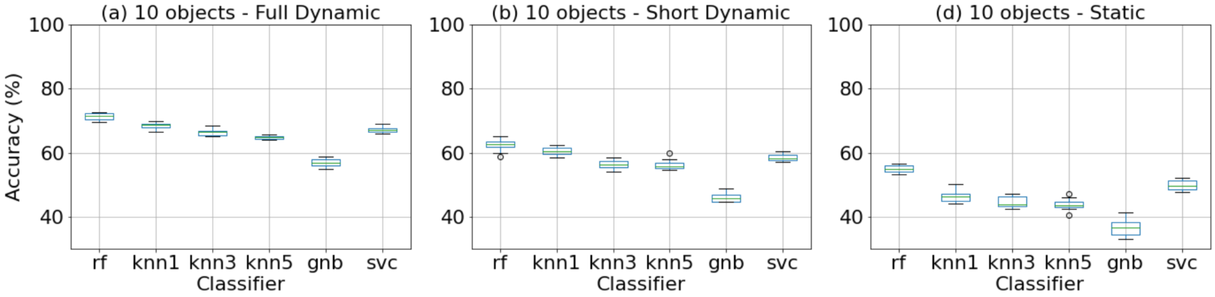

Fig. 5. Boxplots of the highest accuracy achieved for each classifier. (a), (b) and (c) show the results for classifying 10 objects using each validation approaches: (from left to right) dynamic, short dynamic, and static.

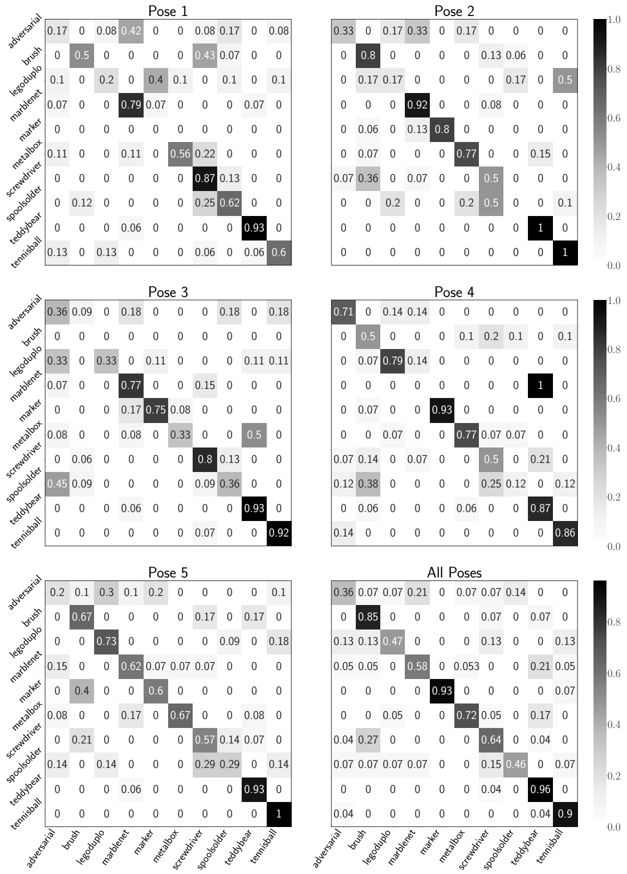

Fig. 6. Confusion matrices presenting the predicted label (x-axis) against the true label (y-axis). The RF classifier is used, with the selected features from Table I. The pose used for testing (and therefore omitted during training) is highlighted above each matrix.

Teddy Bear or the Marker, classification rates are high and consistent in all cases, with an accuracy of more than 90% when all poses are used, and always more than 60% when one pose is missing. In contrast, for other objects, it is very difficult for the classifier to generalize to a certain pose that was not observed during training: an example of this can be seen in Pose 4 for the Marble Net, that is consistently wrongly classified as the Teddy Bear. This is not surprising, since objects which are uniform or symmetrical in terms of shape and stiffness create similar tactile imprints even when grasped in different poses, whilst objects with a more complex and deformable shape, such as the Marble Net, may appear completely differently in testing (i.e. when a novel grasp is performed) compared to the data used for training. In fact, for these complex objects, recognition is difficult even when all the poses are considered, exactly because of this higher variability in the gripper-object contacts from one pose to another (see for example the Adversarial object, which in fact is a very complex shape). In addition, as explained in Sec. III-C, for objects that tend to slip more during the EP (e.g. Brush, Screwdriver, Spoolsolder) there is less data available, and this might affect the recognition accuracy. In fact, this could be considered a general limitation of the EP approach: in order to collect valuable tactile data, the grasp should be robust enough for the object not to slip during the EP. Ideally, this data imbalance could be avoided by using the same number of samples for each object, i.e. the number of samples of the object with less samples. However, we tested this solution with our dataset (detailed results not shown for space constraints) and we verified that using such a balanced dataset (with little data for each object) increases the recognition accuracy for the ”worst” objects only slightly (i.e. between 2% and 5%) and instead reduces the recognition accuracy for the ”best” objects of a consistent amount (i.e. more than 10%).

An additional analysis we performed was to test the RF classifier trained with all features described in Sec. III-D, instead of the best performing selected features only. Results are displayed in Table II, and they show that there is no clear increase in recognition accuracy when using all the features. Therefore, we can conclude that the selected features we identified are indeed a good subset that generalizes well to unseen poses, while keeping the computational complexity of

TABLE II
COMPARISON BETWEEN USING SELECTED FEATURES AND ALL FEATURES IN THE FULL DYNAMIC CASE: THE DIFFERENCE IN ACCURACY IS NEGLIGIBLE

Pose	Selected Features	All Features	Difference
1	62.5%	62.5%	-
2	66.7%	68.3%	+1.6%
3	64.0%	60.4%	- 3.6%
4	60.5%	62.8%	+2.3%.
5	67.6%	69.4%	+1.8%

the classifier (and therefore also the time needed to obtain predictions) bounded.

VI. Conclusions and Future Directions

We show experimentally that using a dynamic Exploratory Procedure (i.e. pick, lift, shake, place an object) improves tactile object recognition, as compared to a single touch on the object; this is possible by using a tactile sensor (uSkin, in our case) that can measure both normal and shear forces on multiple contact points, and it is particularly interesting for applications in which the object can be recognized after it has been grasped and manipulated for at least a few seconds, for example in pick and place operations in logistics or manufacturing. We also show that the combination of both shear and normal forces improves the recognition performance, with respect to using only normal or only shear, making a case for the use of tactile sensors that can collect this type of information, even if the spatial resolution is lower than e.g. camera-based tactile sensors; when a smaller subset of tactile features is selected, to reduce the computational complexity of the classifier, we show that shear forces seem to be even more useful than normal force, especially with data collected during the shaking of the object. Our results demonstrate that when the model has full visibility of data gathered throughout the entire EP an average recognition accuracy of 72% can be achieved (which is a 38% improvement with respect to single touch, that showed a 52% accuracy); the specific numerical values are not very relevant, and it would be hard to compare to recent results in the literature, that are obtained with different objects and robotic setups; however, what 72% tells is that despite some success in object recognition, there is still margin for improvement. Notably, our EP is realized with a standard and relatively simple robotic setup, in semi-structured settings, i.e. the objects are autonomously picked by the robot using vision. This represents a crucial difference with respect to most works in the literature, and it expands the applicability of these systems to real-world scenarios. However, the lack of a fully structured procedure also comes at a cost. While in general the model shows very good recognition capabilities for geometrically uniform objects (more than 90% accuracy), the accuracy drops significantly for more complex objects; this is expected, since complex objects can generate a wide variety of gripper-object interactions when they are grasped in semi-structured settings (i.e. autonomous grasping from vision), and therefore the tactile readings could be very different the next time the same object is grasped. This problem could be mitigated by making sure that each object is always grasped with the same gripper-object configuration: this might be possible in some applications (i.e. very structured industrial settings, in which the target objects are always in the same fixed position, possibly tightly held in place before they are grasped and picked by the robot), but challenging in more unstructured settings.

References

[1] R. Dahiya and M. Valle, ”Robotic tactile sensing: technologies and system”, Springer Science & Business Media, 2012.

[2] D. Silvera-Tawil, D. Rye and M. Velonaki, ”Artificial skin and tactile sensing for socially interactive robots: A review.”, Robotics and Autonomous Systems 63, 230-243, 2015.

[3] S. Luo, J. Bimbo, R. Dahiya and H. Liu, ”Robotic tactile perception of object properties: A review”, Mechatronics, 48:54-67, 2017.

[4] H. Liu, Y. Wu, F. Sun and D. Guo, ”Recent progress on tactile object recognition”, Int. Journal of Advanced Robotic Systems, 2017.

[5] T. Corradi, P. Hall and P. Iravani, ”Bayesian Tactile Object Recognition: learning and recognising objects using a new inexpensive tactile sensor, IEEE ICRA, 2015.

[6] L. E. Carvalho and A. Von Wangenheim, ”3D object recognition and classification: a systematic literature review”, Pattern Analysis and Applications, 22(4):1243-92, 2019.

[7] P. Loncomilla, J. Ruiz-del-Solar and L. Martínez, ”Object recognition using local invariant features for robotic applications: A survey”. Pattern Recognition, 60:499-514, 2016.

[8] R. Bajcsy, Y. Aloimonos and J. K. Tsotsos, ”Revisiting active perception”. Autonomous Robots, 42(2): 177-196, 2018.

[9] J. Bohg, K. Hausman, B. Sankaran, O. Brock, D. Kragic, S. Schaal and G. S. Sukhatme, ”Interactive perception: Leveraging action in perception and perception in action”. IEEE Transactions on Robotics, 33(6), 1273-1291, 2017.

[10] H. Liu, X. Song, J. Bimbo, L. Seneviratne and K. Althoefer, ”Surface Material Recognition through Haptic Exploration using an Intelligent Contact Sensing Finger”, IEEE/RSJ IROS, (pp. 52-57), 2012.

[11] A. Schmitz, Y. Bansho, K. Noda, H. Iwata, T. Ogata and S. Sugano, ”Tactile Object Recognition using Deep Learning and Dropout”, IEEE-RAS Humanoids, 2014.

[12] J. M. Gandarias, A. J. García-Cerezo and J. M. Gómez-de-Gabriel, ”CNN-Based Methods for Object Recognition With High-Resolution Tactile Sensors”, IEEE Sensors Journal, 19(16):6872-6882, 2019.

[13] S. Funabashi et al., ”Object Recognition Through Active Sensing Using a Multi-Fingered Robot Hand with 3D Tactile Sensors,” IEEE/RSJ IROS, pp. 2589-2595, 2018.

[14] T. P. Tomo et al., ”A New Silicone Structure for uSkin - A Soft Disrtibuted, Digital 3-Axis Skin Sensor and its Integration on the Humanoid Robot iCub”, IEEE Robotics and Automation Letters, 3(3):2584-2591, 2018.

[15] P. Giguere and G. Dudek, ”A simple tactile probe for surface identification by mobile robots”, IEEE Transactions on Robotics, 27(3):534-544, 2011.

[16] Z. Pezzementi, E. Plaku, C. Reyda and G. D. Hager, ”Tactile-object recognition from appearance information”, IEEE Transactions on Robotics, 27(3):473-487, 2011.

[17] R. Li and E. H. Adelson, ”Sensing and recognizing surface textures using a gelsight sensor”. IEEE CCVPR, pp. 1241-1247, 2013.

[18] F. Pastor et al., ”Bayesian and Neural Inference on LSTM-Based Object Recognition From Tactile and Kinesthetic Information”, IEEE Robotics and Automation Letters, 6(1):231-238, 2021.

[19] T. P. Tomo et al., ”Covering a Robot Fingertip With uSkin: A Soft Electronic Skin With Distributed 3-Axis Force Sensitive Elements for Robot Hands”, IEEE Robotics and Automation Letters, 3(1):124-131, 2018.

[20] T. Paulino et al., ”Low-cost 3-axis soft tactile sensors for the human-friendly robot Vizzy”. IEEE ICRA, 2017.

[21] L. Jamone, L. Natale, G. Metta and G. Sandini, ”Highly sensitive soft tactile sensors for an anthropomorphic robotic hand”. IEEE Sensors Journal 15(8):4226-4233, 2015.

[22] B. Denoun, B. Leon, M. Hansard and L. Jamone, ”Grasping Robot Integration and Prototyping: the GRIP Software Framework”. IEEE Robotics and Automation Magazine 28(2), 101-111, 2021.

[23] R. Zenha, B. Denoun, C. Coppola and L. Jamone, ”Tactile Slip Detection in the Wild Leveraging Distributed Sensing of both Normal and Shear Forces”, IEEE/RSJ IROS, 2021.

[24] D. Morrison, P. Corke and J. Leitner, ”Learning robust, real-time, reactive robotic grasping”, The International Journal of Robotics Research, pp. 183-201, 2020.

[25] F. Pedregosa et al., ”Scikit-learn: Machine Learning in Python”, JMLR 12, pp. 2825-2830, 2011.

[26] S. Zhang, X. Li, M. Zong, X. Zhu and R. Wang, ”Efficient kNN Classification With Different Numbers of Nearest Neighbors”, IEEE Transactons on Neural Networks and Learning Systems, 29(5), 2018.

[27] L. Breiman, ”Random Forests”, Machine Learning 45.1, 5-32, 2001.

[28] J. Hoelscher, J. Peters and T. Hermans, ”Evaluation of Tactile Feature Extraction for Interactive Object Recognition”, IEEE Humanoids, 2015.

[29] C. Cortes and V. Vapnik, ”Support Vector Networks”, Machine Learning 2-.3, 273-297, 1995.