International Journal of Intelligent Robotics and Applications (2025) 9:375–388
https://doi.org/10.1007/s41315-024-00402-6

REVIEW PAPER

A virtual data generator system for shape recognition in haptic robotics

Anna Gutiérrez[1] · Guillem Garrofé[1] · Pau Nonell[1] · Claudia Serrano[1] · Carlota Parés-Morlans[1] · Tomás van den Heijkant[1] · Mireia Vera[1] · Conrado Ruiz Jr.[1,2] · Laia Vidal[1] · Alejandro González[1] · Òscar de Jesús[1] · Raquel Ros[1] · David Miralles[1]

Received: 26 March 2024 / Accepted: 29 October 2024 / Published online: 21 November 2024

Abstract
In robotics, the current state of object recognition in haptic sensory mode falls significantly short of the results obtained in visual mode. One of the main reasons for this is the lack of haptic data sets for training recognition models. A major impediment is the time-consuming and difficult task for a real robot to capture large amounts of haptic information. This paper introduces a virtual haptic dataset generator system that captures haptic features based on the curvatures of an object. The main goal is to show that this capture system is a feasible approach that can eventually be implemented not only in virtual settings but in actual robots. The virtual haptic capture system described speeds up the learning process, where a real robot would learn through virtual simulation. The paper shows three important points that make the system feasible. The capture is independent of the angle of inclination of the end-effector as it approaches the explored object. The system recognition is performed on everyday objects. Since a real system is exposed to noise during data acquisition, the data of the virtual system must also contain noise. High performance is still achieved within the noise ranges of current sensor systems.

Keywords Actuators · Sensors · Haptic · Design · Object recognition · Simulation

1 Introduction

Studying the development of human object recognition abilities, we know that infants learn object features by exploring them with their hands (Gopnik 2012), rather than just passively looking at them. Exploring objects through the sense of touch provides distinctive features such as shape, size, weight, temperature, softness, and texture among others. Together, these features allow us to identify and understand how we can interact with objects in the physical world (Dahiya et al. 2015). Inspired by these human capabilities and with the aim of developing artificial agents that interact efficiently in the real world, the advancement of haptic perception in the field of robotics has become increasingly important over the last decade. For an artificial system, haptic recognition is not only useful in some adverse visual situations, but it also complements vision in the integration of the two modalities (visuo-haptic) and in the interpretation of a common inter-modal reality.

In artificial systems, developments in visual and haptic modalities are not progressing at the same pace. Huge advancements have been made in visual object recognition thanks to the availability of large data sets (Deng et al. 2009) with a significant number of images. On the other hand, data acquisition by touch is a problem that is not yet well solved. Collecting haptic data with real robots is very time-consuming (Levine et al. 2018) and usually requires human assistance. A possible shortcut and the approach of this paper is the generation of virtual data sets where the robot-object interaction takes place in a simulated environment. In addition, this approach can also be used to create new models of haptic interaction and validate them before implementing them in real physical robots, which would significantly reduce costs. It should also be noted that a virtual capture system does not have the sensor noise of a real system. Therefore, it is important to analyze the robustness of the virtual object recognizer when noise is added to the

✉ David Miralles
david.miralles@salle.url.edu

[1] Human-Environment Research (HER), La Salle-Universitat Ramon Llull, Sant Joan de la Salle 42, Barcelona, Spain

[2] De La Salle University, 2401 Taft Ave, Manila, Philippines

synthetic data. Based on this, in this paper: (1) we propose a virtual haptic capture system based on local curvatures, (2) we show its effectiveness in recognizing quotidian objects, and (3) we analyze the feasibility of transitioning from this virtual capture system to a real system by adding noise to the virtual data.

To recognize objects by their shape, we focus on the characterization of them based on local geometry. To extract this information, the virtual haptic data capture system has an end-effector with three equally spaced fingers that contain a contact sensor at the end of each finger for detecting contact with virtual objects (Fig. 1). For each touch of the object's surface, we obtain a three-component vector, each value corresponding to the contact time of each finger. Our system receives neither information about the contact force nor about the deformation of the analyzed object. In other words, the interaction is completely rigid, only the temporal distance between the fingers is captured. Therefore, our approach uses a low-dimensional feature to capture the local curvature in order to recognize objects using a classifier.

The paper is organized as follows. Section 2 positions our work with respect to the advances in object recognition through tactile sensing. Section 3 introduces our approach to local curvature computation based on different finger contact times. Section 4 presents the digital capture system, the ShapeNet stimulus collection (Chang et al. 2015), and the captured data set. Section 5 introduces the classifiers, presents the results on the Shapenet data set, and discusses the main achievements. Finally, Sect. 6 analyzes the behavior of our system when noise is introduced into the data set.

2 Related work

In the current state of the art, we find several proposals for haptic capture systems. These approaches can be categorized based on the input of the recognition system. We can divide them into three categories: (1) those that use the distribution or spatial coordinates of contact points to recognize a 3D shape, (2) those that use a tactile array of sensors that measure pressure patterns, (3) the combination of both.

2.1 Distributed tactile sensors

The works that use spatial coordinates usually have sensors at the end-effector of a robot and ascertain the geometry and global shape of an object based on the location of the contact points (Allen et al. 1988; Zhang et al. 2016). These approaches have been successful for specific scenarios where objects are fixed and stationary. However, it can require some considerable time especially when a huge number of samples is needed to recognize the object's global shape. Moreover, the low precision of the contact points' sensors strongly directly affects the chaining of transformations involved in calculating hand pose configurations that can cause errors (Gorges et al. 2011).

2.2 Tactile sensor arrays

Other approaches have used high-resolution tactile arrays. These arrays of tactile sensors are used to gather pressure patterns based on the measure of contact forces or the deformation of a soft elastomer skin. These tactile patterns can be represented as images, which can be processed using computer vision descriptors. GelSight (Yuan et al. 2017), which uses an optical tactile sensor to obtain a high-resolution view of the contact surface geometry, has successfully been used in numerous challenging robotic tasks (Dong et al. 2017). This sensor has also been used for surface-following tasks (Lu et al. 2019) using deep reinforcement learning.

Several works have tried to mimic the haptic sensing capability of the fingertips of human beings. Early studies (Goodwin et al. 1991) have explored the ability of our fingers to discriminate flat, concave, or convex curvatures.

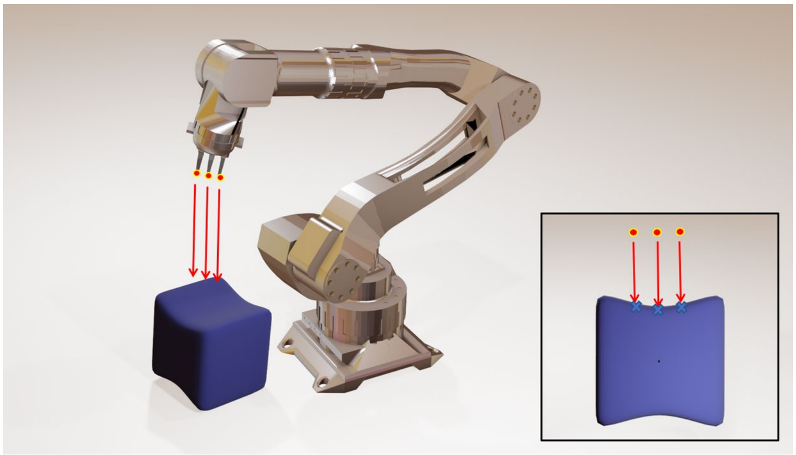


Fig. 1 Our proposed three-finger end-effector added to a conventional robot arm

This has led to a spherical tactile sensor for sensing curvature using the color-interference of a marker array (Lin et al. 2020). Nonetheless, this type of tactile sensor also has several problems, like the expensive computation required to handle the high dimensional raw data (Polic et al. 2019). Mapping this data to a lower-dimensional feature space can also result in a loss of some of the physical information from the input.

Combining these two approaches also show promise in recognizing objects. When the tactile patterns and contact point spatial information are combined, more robust methods can be achieved (Spiers et al. 2016; Luo et al. 2016).

Nonetheless, these approaches usually rely on large data sets of training data in order to utilize modern machine learning algorithms. Capturing haptic data for training by robots is still a very complex and costly task, so the study of generating synthetic data sets can be an interesting alternative (Wu et al. 2019; Gomes et al. 2021) to save time and resources.

2.3 Contribution to the state of the art

The improvements of this work over the state of the art are related to the two approaches previously described. The haptic capture system we present allows us to acquire knowledge about the local geometry of the object being explored without any information regarding its global location, avoiding the errors involved in calculating the hand pose configuration. Furthermore, unlike tactile patterns, the dimensionality of our captured data is extremely low, and despite this, we obtain results similar to the state of the art.

Our proposal is based on Garrofé et al. (2021) where the capture system proposed is just a toy model that only works in ideal conditions with a very limited collection of virtual stimulus shapes. For example, the end effector's fingers had to be perfectly aligned, the direction of approach always had to be perpendicular to the object surface, and as a virtual capture system, there was no noise in the captured data. In our work, our principal objective is to show that this capture system could be implemented in a real robot and we solve all the issues enumerated above. We show this by capturing data from haptic sensors that approach the object with different angles of inclination, using complex everyday objects for testing, and evaluating the robustness of the haptic system by adding synthetic noise to the input digital samples.

3 Local curvature features from virtual haptic sensors

We have designed a virtual manipulator composed of three aligned and equally spaced fingers with a contact sensor on each. The fingers explore a stimulus by moving at a constant speed toward its surface until they detect a collision. The system extracts the relative times between each contact sensor, i.e., it measures the time difference between the first contact with the object and the subsequent contacts. We thus obtain a feature composed of three values (t_1, t_2, t_3), where each t_i corresponds to the time elapsed from the first contact sensor and the contact of sensor i. Hence, the first sensor contact time is always 0 as a reference for the other two, see Fig. 2. The time differences form a discrete function that provides the system with local geometric information regarding the object being explored, i.e., its curvature, computed as follows:

$$(t_1 - t_2) - (t_2 - t_3)$$

Preferably, in order to work in a real environment, the captured parameter should not depend on the inclination of the end-effector with respect to the surface of the explored object. Let us consider an exploration where the manipulator is displayed as in Fig. 3, where the end-effectors' orientation is arbitrary to the objects' surface as shown in Fig. 2.

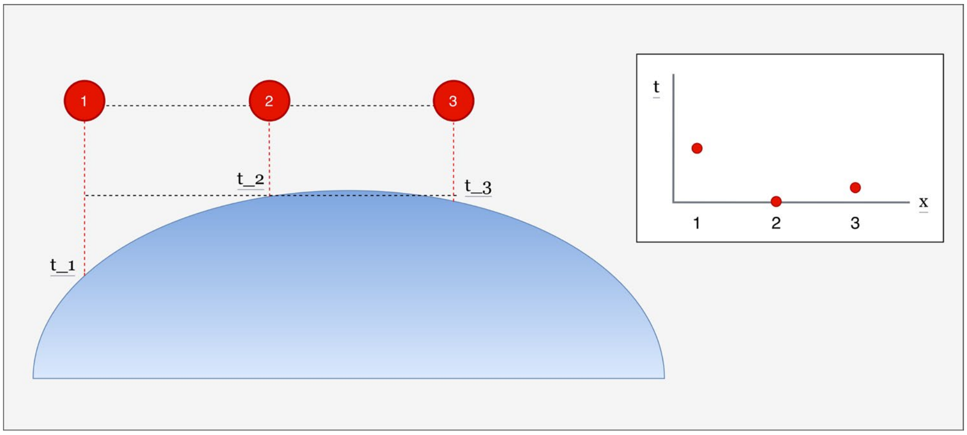


Fig. 2 Representation of an end-effector contact event. The three contact sensors are represented by the red dots and the stimulus by the blue zone. In the box (top right), the graph shows the time versus position relation between the sensors. As an example, the box shows the moment when the first sensor contacts the surface. Once they collide with the stimulus' surface the system generates the temporal feature (t_1, t_2, t_3), where $t_2 = 0$ and $t_1 > t_3$

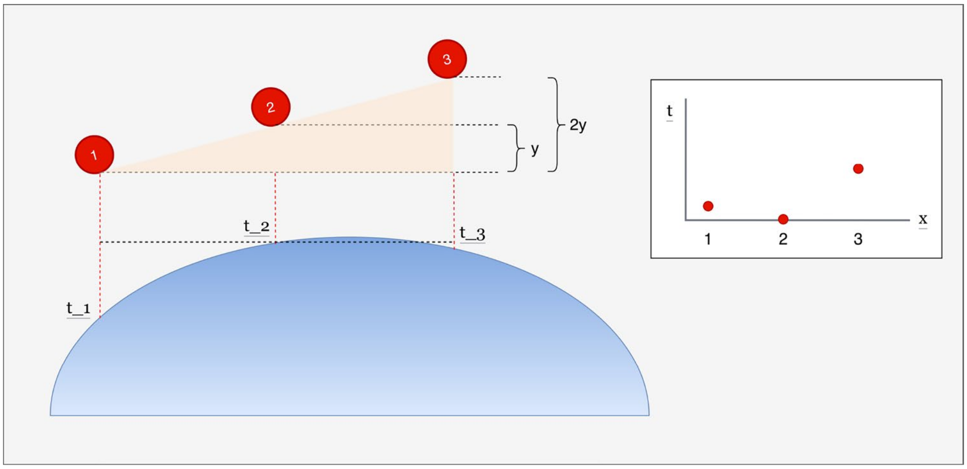


Fig. 3 Representation of an oblique contact. In this image, the contact surface is the same as in Fig. 2. In this case, the end-effector moves remarkably inclined (determined by the distance y) towards the surface. As shown in the box above, finger 1 will make contact with the surface before finger 3 does. However, the resulting curvature computation is invariant as proved in the text

The contact times (t'_1, t'_2, t'_3) will be disrupted by the fingers' initial location as follows:

$$t'_1 = t_1$$
$$t'_2 = t_2 + t_y$$
$$t'_3 = t_3 + 2t_y$$

where t_y is the time required to traverse the space y due to the inclination of the end-effector (see Fig. 3).

In this scenario, the data obtained do not reflect the curvature of the stimulus directly (as in Fig. 2), but it can be inferred from the distance travelled by the inclined end-effector towards the stimulus as follows:

$$(t'_1 - t'_2) - (t'_2 - t'_3) = t_1 - 2t_2 - 2t_y \\ + t_3 + 2t_y = (t_1 - t_2) - (t_2 - t_3)$$

Therefore, the perceived curvature in both scenarios, Fig. 2 and Fig. 3, is the same. Thus, the characteristic information contained in each perceived sample regarding the geometry of the stimulus, i.e. the curvature, can be measured by exploring the stimulus' surface regardless of the inclination angle from which the end-effector approaches the object.

Another important aspect of our haptic capture system is the speed of the end-effector. In this case, two cases must be considered. The first case where the speed differences are within the same data set of the training. A second case in which speed differences occur between training data sets. When creating a dataset to train models, it is important that the speed remains constant, otherwise we may obtain different contact times and therefore different curvatures for the same sample of the same object. Of course, it is not easy for a real robot to always have the same capture speed. For this reason, in Sect. 6 we add noise to our data to simulate differences in speed capture. Another case is when there are two training data sets and these were captured at different speeds. In this case, the same sample would have different contact times in each data set, but this would only be a scaling factor. For example, consider a haptic capture system with velocity measurement v. Given another haptic capture system with the same design but with velocity v', where $v = av'$. Then the same two samples taken from the same object in two different data sets are (t_1, t_2, t_3) and $(t'_1, t'_3, t'_3) = (at_1, at_2, at_3)$. As you can see, the difference between the samples is only a scaling factor, which of course does not affect the predictions of the same model trained with the two different data sets.

4 Capture system, stimuli and data set

The implementation of our haptic capture system has been done in a virtual environment developed in Unity (Juliani et al. 2018) (v2019.4.11f1). In this environment, the three contact sensors are represented as spheres with a scale of 0.3 units, aligned according to what we call an alignment vector (see green line in Fig. 4), and with equal spacing of 1.5 units between them. To extract same-level features for all the objects, the 3D objects are normalized in their size so that we can ensure that the contact sensors can touch them on a similar scale. This is done in this paper in order to be able to compare the curvatures of the different stimuli better and on the same scale. If the proportions of the real stimuli are to be maintained, it is necessary to do the same in the virtual environment. In this case, it is only necessary to ensure that the distance between the fingers of the end effector is always the same and sufficient to explore the smallest object in the set of stimuli.

The capture system process consists of a set of contact iterations per object. The 3D object is placed in the center of the scene and the contact sensors are in a random position (different for each iteration) around it. From this position, a target point inside the objects' bounding box is selected randomly. The contact sensors move towards the object

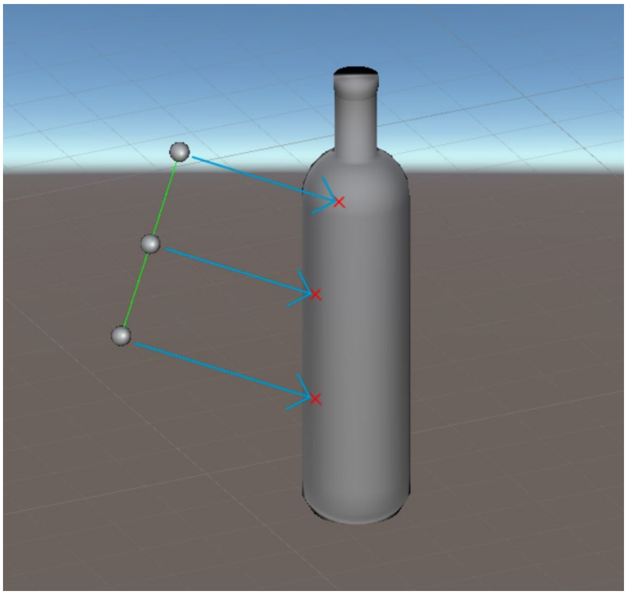


Fig. 4 Unity environment developed to capture haptic data from the stimulus (bottle), three aligned contact sensors (spheres), pointing to a point on the object (red crosses) and moving in the same direction (arrows)

according to the direction of the vector that connects the central contact sensor and the target point (see blue arrows in Fig. 4). This way, the system obtains samples from any part of the object's surface. Instead of aligning the contact sensors orthogonally to the motion direction vector, we have introduced a random tilt [-10º, 10º] to simulate possible misalignment in the robotic manipulator. As described in Sect. 3, this inclination does not compromise the measure of the curvature. Having placed all the elements (see Fig. 4), the three contact sensors move with a constant velocity according to the formerly mentioned direction. The system detects when a contact sensor collides with the object and saves the contact time relative to the first collision. When the three sensors have collided, the system resets the position of these with new locations and captures new information until the number of defined contact iterations per object is accomplished.

Ordinary household objects have been used as stimuli for the experiments described in this paper. The 3D shapes of these objects have been drawn from the online 3D data set, shapenet (Chang et al. 2015). Specifically, we use ten different objects: a bowl, a camera, a paper clip, a fork, a hammer, a computer mouse, a pair of scissors, a spoon, a wristwatch, and a bottle of wine (see Fig. 5). In a more recent work (Hogan et al. 2021), the same objects were used for recognition, which allows us to compare the results.

Having these stimuli, we executed our capture system with 21,000 iterations for each object with a constant velocity of the contact sensors of 10 units/second. Each iteration generated a sample, where a sample is composed of the three float relative time values. The data of each object was stored in a text file, where each sample is written as a line. This leads to a data set composed of ten files with 21,000 lines of tactile data. These samples are available in an open repository.

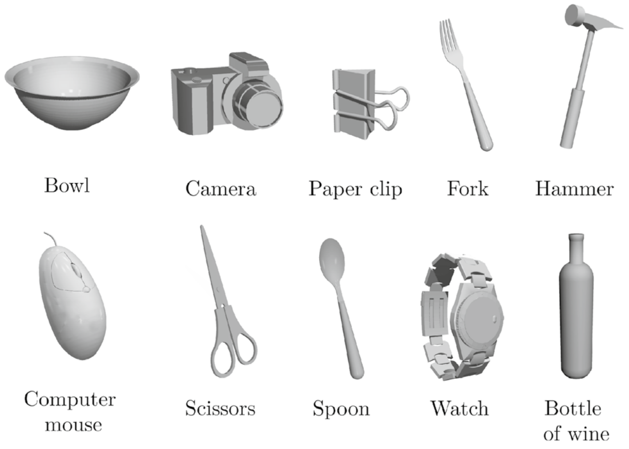


Fig. 5 ShapeNet 3D objects used as stimuli for haptic data capture

As described in Sect. 3, the relative times (t_1, t_2, t_3) captured by the contact sensors provide us with information on the convexity or concavity of the stimulus in the sampled point. By collecting multiple samples we can extract the different curvatures of each object. Using a Kernel Density Estimation, we can compute the probability density function based on second derivatives obtained from each stimulus. Figure 6 shows clear differences between the PDFs of the stimuli, this suggests that curvature can be an appropriate feature for classification.

5 Haptic object recognition

The classifier we use for object recognition is XGBoost. XGBoost is a machine learning system whose implementation is based on a gradient-boosting decision tree algorithm(Chen et al. 2016). It ensembles a sequence of tree models, each one of those being a weak classifier whose training is conditioned by the incorrect predictions of its previous model. The predictions of each model are added, obtaining the final probability predictions of a sample belonging to each class (Fig. 7). The parameters of the model are: 100 decision trees as weak classifiers, with 6 levels of depth each, and trained with a gradient descent algorithm in which the learning rate is set to 0.3, with a gradient tree booster and *multi:softprob* as a learning objective. The training of this classifier is done by feeding the XGBoost model with a set of samples, where each sample consists of the 3 relative time values feature vector described above.

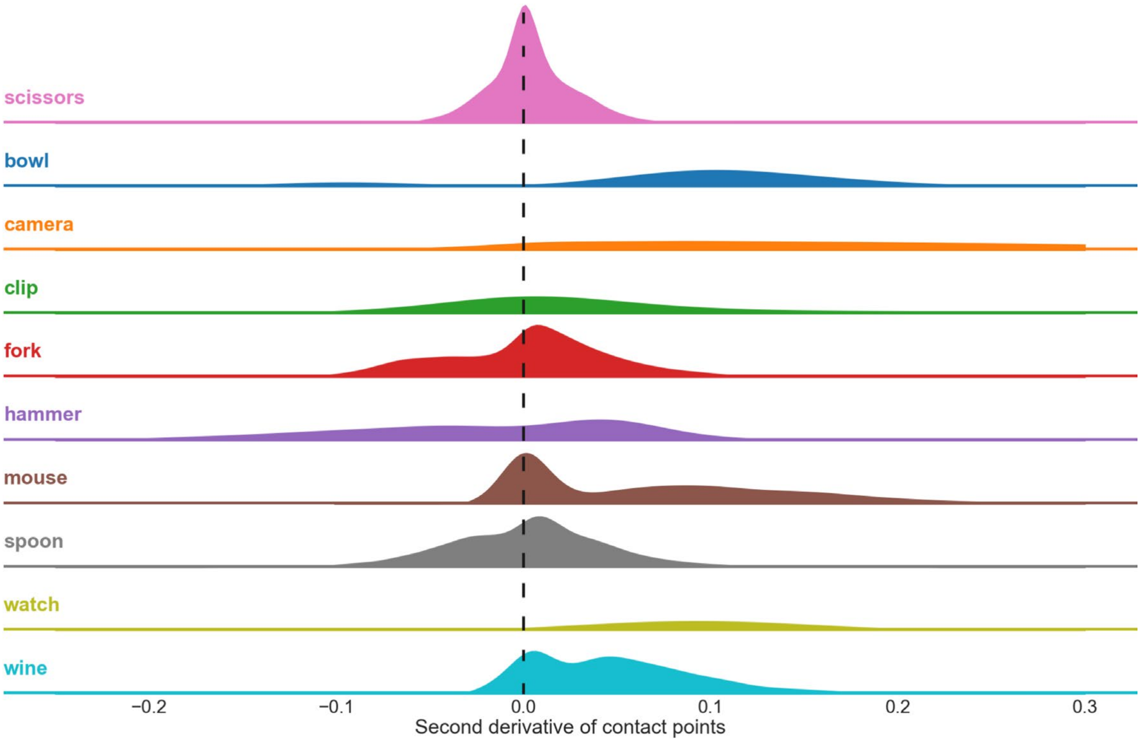


Fig. 6 Haptic descriptors based on PDFs of curvatures. The descriptors shown correspond to each of the 10 selected objects in the ShapeNet data set (Fig. 5). The dashed vertical line marks the zero curvature (flat) of the explored surface, and on its right, the curvature is concave while on its left, the values correspond to convex curvature

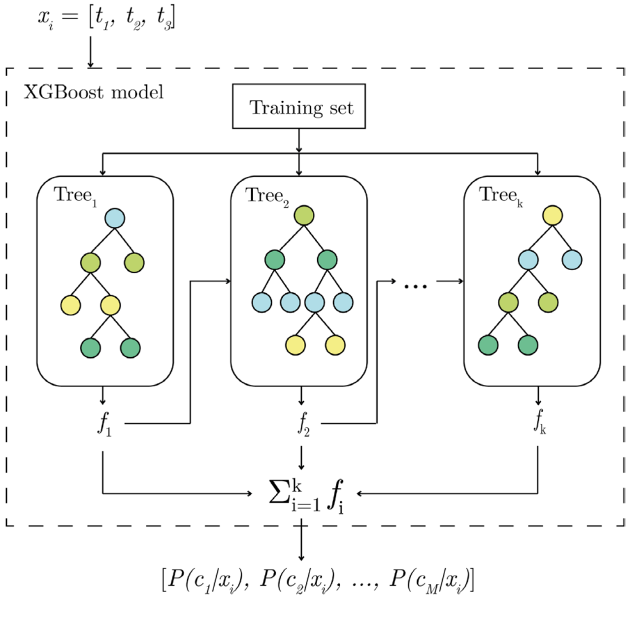


Fig. 7 XGBoost diagram. The test feature vector x_i is processed with the k decision trees and their predictions f_i are added to obtain the probability $P(c_j|x_i)$ for each one of the M classes

Once the model is trained, in the test phase, we take N feature vectors of the stimuli, which defines our set of samples X (Fig. 7). From the predicted probabilities of the XGBoost-trained model, we obtain the likelihood of a sample belonging to each class. We obtain the likelihood that a given sample x_i belongs to a class c_j. The Bayes's theorem allows us to compute the probability for a set of N samples, $(x_1, x_2, \dots, x_N)$. The resulting Bayesian XGBoost classifier determines the class of the given feature vectors by maximizing the combination of the logarithm of their probabilities. It can be formulated as:

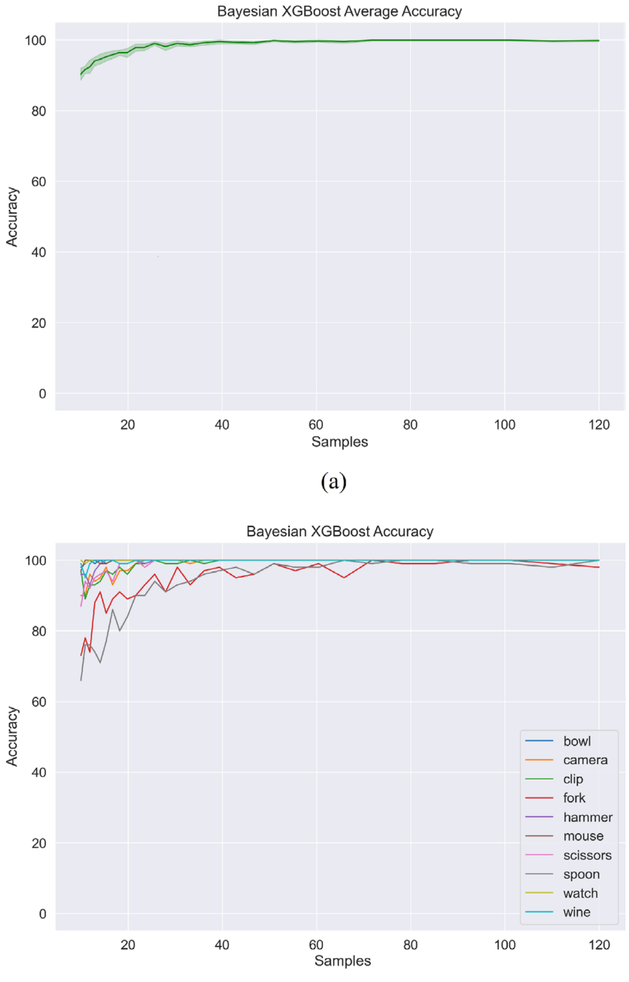
$$y = \underset{j \in [1,M]}{\operatorname{argmax}} \left[\sum_{i=1}^{N} \log \left(P(c_j|x_i) \right) \right] \tag{1}$$

where N is the number of feature vectors in the set of obtained samples X, M the number of classes (10 in this paper), and $P(c_j|x_i)$ is the probability of the feature vector x_i to belong to class c_j, which is obtained from the XGBoost model.

The recognition results have been obtained by means of a 10-fold cross-validation test with the samples collected from our data set based on the ShapeNet objects. To study their accuracy in different amounts of feature vectors or samples (N), we tested the classifiers for 30 ranges of sample sizes distributed in a logarithmic scale from 10 to 120. For each fold and sample size, we have computed the accuracy of 10 groups of feature vectors per object.

In Fig. 8(a), we can observe the average accuracy obtained depending on the taken sample size. This evolution shows that with only 10 samples, the classifier achieves a 90.7% (±2.52) average accuracy. Such performance is desirable when moving from simulation to a real environment (see Sect. 6), where we expect a robot to recognize an object by sensing a few samples. Anyway, we can improve its accuracy by increasing the number of samples. As shown in Fig. 8(a), we achieve around 99% of average accuracy from 30 samples.

In order to better understand the results, individual analysis of the different objects is done by plotting the specific classes' accuracies in Fig. 8(b) and by showing the confusion matrix obtained at a certain number of samples in Fig. 9. As we can observe, there are two objects that fail to be recognized significantly more: the fork and the spoon. The similar shape of these objects causes the classifier to be confused with each other and get a lower accuracy. Even with this confusion, our classifier is able to differentiate them with accuracies higher than 90% from 25 samples, proving that our method can be applied to similar objects. Hogan et al. (2021) propose a tactile recognition method applied to the same Shapenet objects, and their average recall was 90.4%. Our average recall for just 10 samples is 89.5%. Thus, we can affirm that our method reaches similar metrics to theirs. The data set generated for the experiments explained above is available in an open repository. We encourage the machine learning research community to achieve better accuracy with fewer samples through published data.



(a)

(b)

Fig. 8 Bayesian XGBoost classifier: **a** average accuracy, and **b** accuracy for each class given the number of test samples evaluated on the trained PDFs

6 Addition of noise: towards a real environment

The results presented in Sect. 5 have been obtained in a virtual system without considering any possible noise. Obviously, in a real robotic setup, noise is inevitable. In order to assess the robustness of our system, we will subject it to perturbations by adding Gaussian noise. Therefore, the system will classify using virtual data modified by introducing these perturbations to simulate a more realistic scenario.

6.1 Noise insertion to samples

From a normal distribution $(0, \sigma)$ we draw the values of the noise. We test with different values of standard deviation to study accuracies with different noise levels.

As described before, the samples consist of three values (contact times), where one of them is 0, this value indicates the moment when the first finger's end-effector makes contact with the object. The contact time of the two other fingers is computed relative to this value. For each sample, the normally distributed noise is added independently to each contact time value. The noise can be negative, but we prefer to deal with non-negative times and maintain our zero reference. To maintain this zero reference in each set of 3-time samples, the following methodology is applied: once we have applied the noise, We take the lowest value of each set and subtract it from all three values of that set. If that value is negative, a positive offset is applied to the entire set, however, if that value is positive, a negative offset is applied to the entire set. In this way, we can guarantee that despite the noise added to the samples, there is always a contact time with a zero value (reference) and the rest are non-negative.

These noisy samples are a combination of the virtual samples and the commonly occurring sensor noise in actual

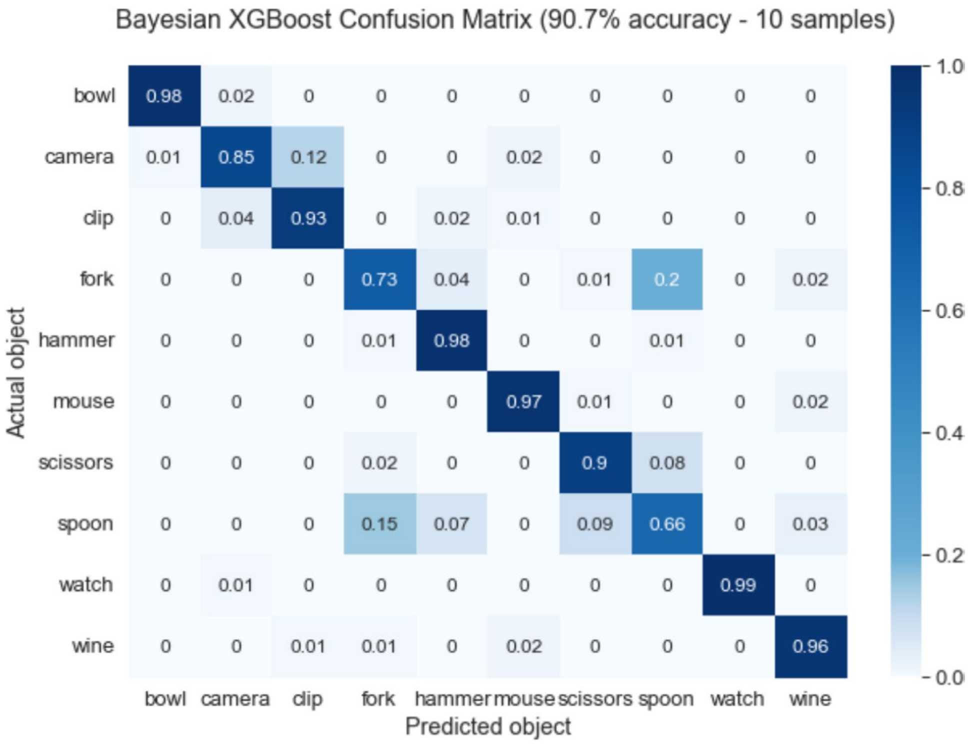


Fig. 9 Bayesian XGBoost classifier's confusion matrix with 10 samples (90.7% accuracy)

robots. As such, by utilizing these samples as test cases, we are able to test the performance of the capture system in a non-ideal scenario that is closer to a real-world environment.

6.2 Noise results

We consider the signal-to-noise ratio (SNR) measure in this study. Figure 10 shows the relationship between the SNR of each object (where the signal is the sum of all the time samples for one stimulus) and the noise level as a function of the standard deviation.

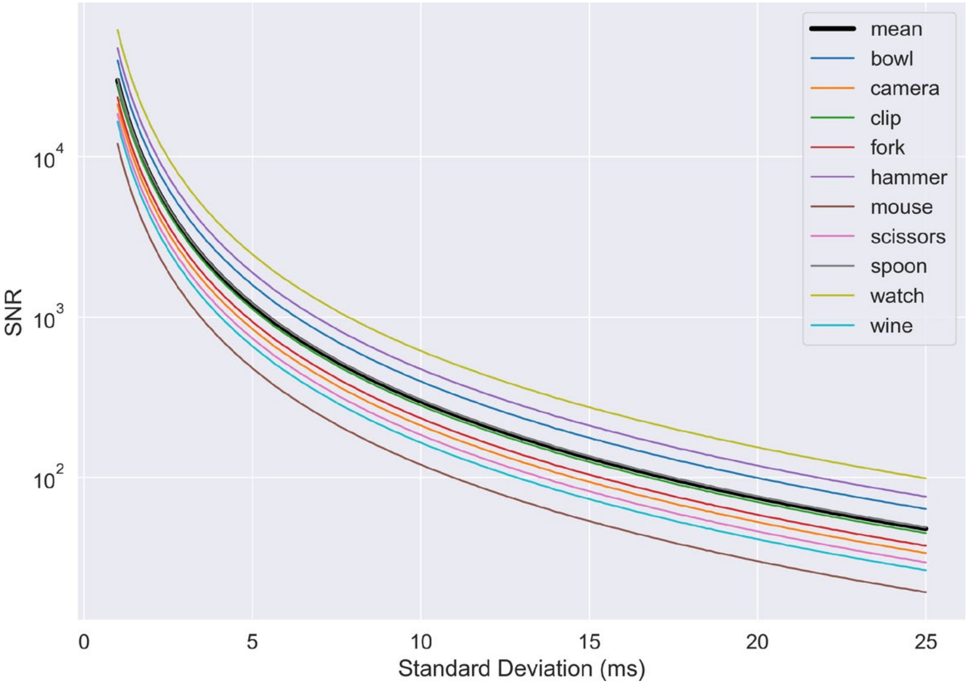


Fig. 10 Signal-to-noise ratio of each stimulus versus Gaussian noise added and quantified with its standard deviation

Noise does not affect all objects equally. We can observe in Fig. 10 that the SNR of the ’watch’ is much higher than the one of the ’mouse’. This is because although both are subjected to the same noise, the mouse samples have smaller values than the watch. This difference in the SNR between objects has a reflection on the accuracy of each object independently, so we will consider the mean SNR to interpret the classifier’s performance results.

To compare the performance of the ideal system presented with the results of testing this system with noise, we evaluated the accuracies obtained by testing with 10 samples.

Figures 11 and 12 exhibit the relationship between the individual and global accuracy of our system classifying with XGBoost and the standard deviation of the noise measured in ms. Actual data (blue) represents the current results obtained as output from the classifier and the filtered data (green) is the result of applying a Savitzky-Golay filter to the actual data to provide a clearer representation of the results.

The accuracy obtained with the virtual system shown in Fig. 9 was 90.7%, whereas Fig. 12 shows an accuracy of higher than 82% with an addition of a Gaussian Noise with a standard deviation of 5 ms. Figure 13 shows the confusion matrix for our model with 5 ms and can be compared with Fig. 9. This corresponds to the 10 ms of disturbance margin in which our system will still be able to recognize the stimuli.

The main cause of disturbances in the real world is the sensitivity and noise produced by the electronic components used to capture samples, specifically the sensors. These components exhibit errors well below 10 ms, which makes our system viable in a real robotic environment.

7 Conclusions

The capture system and the algorithms proposed in this work allow us to recognize everyday objects that we have obtained from the Shapenet data set. With only 10 test samples, the XGBoost algorithm achieves 91% mean accuracy. These results are comparable to other state-of-the-art works and in particular, those that use the same data set (Hogan et al. 2021). These results allow us to consider the proposed model as a viable option to be implemented in a real robot.

A key point in analyzing the robustness of the proposed capture system is noise analysis. Figure 12 shows the relationship between noise and accuracy. It is interesting to see how a real system that captures data with an error below 10ms will keep the accuracy above 80%. This is a more than acceptable error for the current state of sensor electronics.

Local data-based haptic capture systems have the advantage that the captured samples are independent of the robotic reference system used during capture. However, the dimensionality of the data captured by these systems tends to be very large. In this work, a haptic capture system has been presented which extracts very low dimensional data and is associated with the local characteristics of the object (curvature). Unlike other previous implementations (Garrofé et al. 2021; Yuan et al. 2017) where the contact sensors must be oriented orthogonally to the surface of the object, in this work, in a more realistic way, the three fingers of the end-effector can approach the object at any angle and from any direction, this allows us to simply point to the object and head towards it to capture a sample.

Thus, this work has shown that the proposed capture system and recognition algorithms are viable options for future implementation. In addition, capturing training data with a virtual system would save a huge amount of time that would not be possible under other conditions. It is therefore necessary to adapt the data of the virtual model to the real world but this should not be a problem as the curvature is invariant under scale transformations.

In this paper, we have studied one of the possible applications of our capture system which is the recognition of everyday objects. Nonetheless, the proposed system could go further and also be used for the classification or the 3D reconstruction of objects. Currently, the haptic capture system only collects data related to curvature, without other tags. In order to reconstruct the object in 3D, it would be necessary to assign to each of the samples obtained the orientation with which they were taken, i.e. the orientation of the end-effector with respect to the object. Our capture system takes this data, although it was not used in this work.

Beyond 3D reconstruction, the inclusion of the orientation in which the samples were taken could help overcome some of the limitations of our system, especially in cases of disambiguation. For example, in Fig. 6, where the curvature-based haptic descriptors are shown, the first case corresponds to a pair of scissors, but it is clear that many very different geometries could have a similar descriptor profile. Labeling the samples according to orientation would extend the dimension of the haptic descriptor and allow a better geometric understanding of the explored object.

Another possible future research direction is the application of this haptic capture system in a multimodal context (haptic and visual) or even cross-modal. The development of a visual capture system, also based on curvatures, could

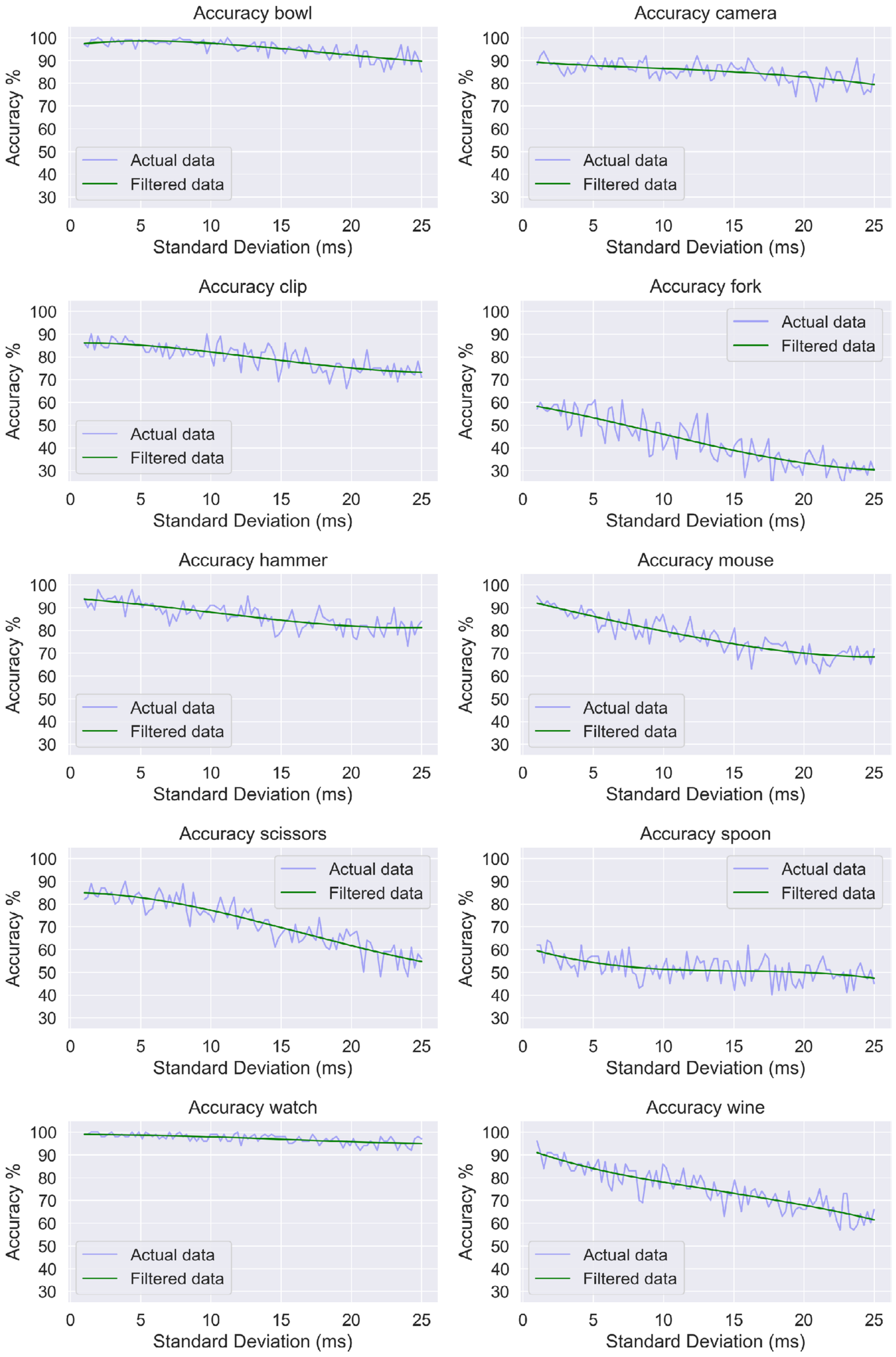


Fig. 11 Individual accuracy obtained from XGBoost classification with 10 test samples and Gaussian noise added

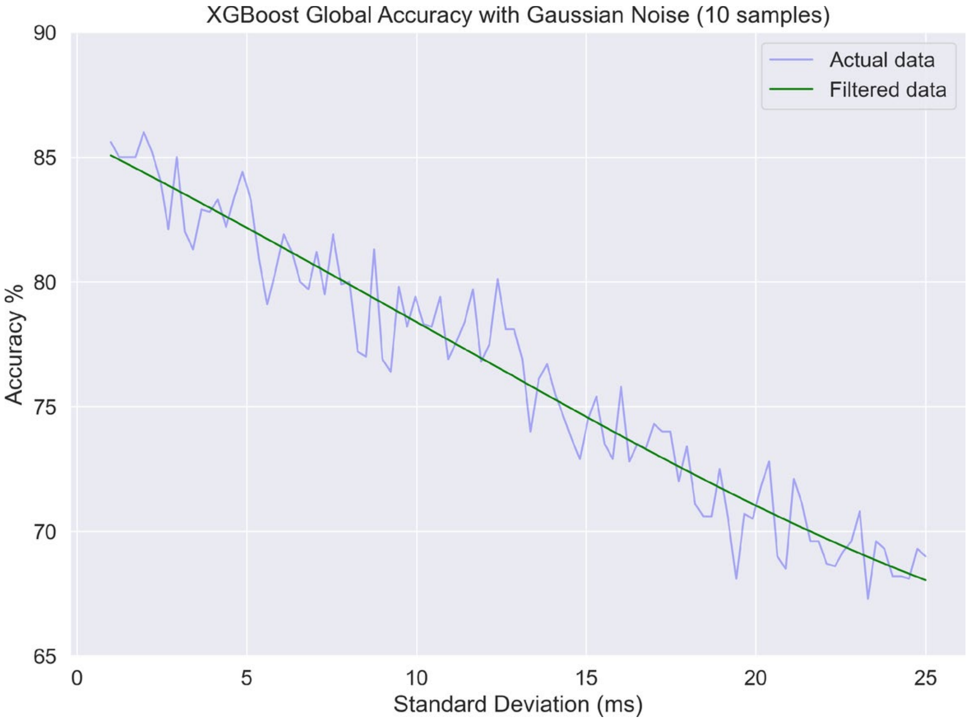


Fig. 12 Global accuracy obtained from XGBoost classification with 10 test samples and Gaussian noise added

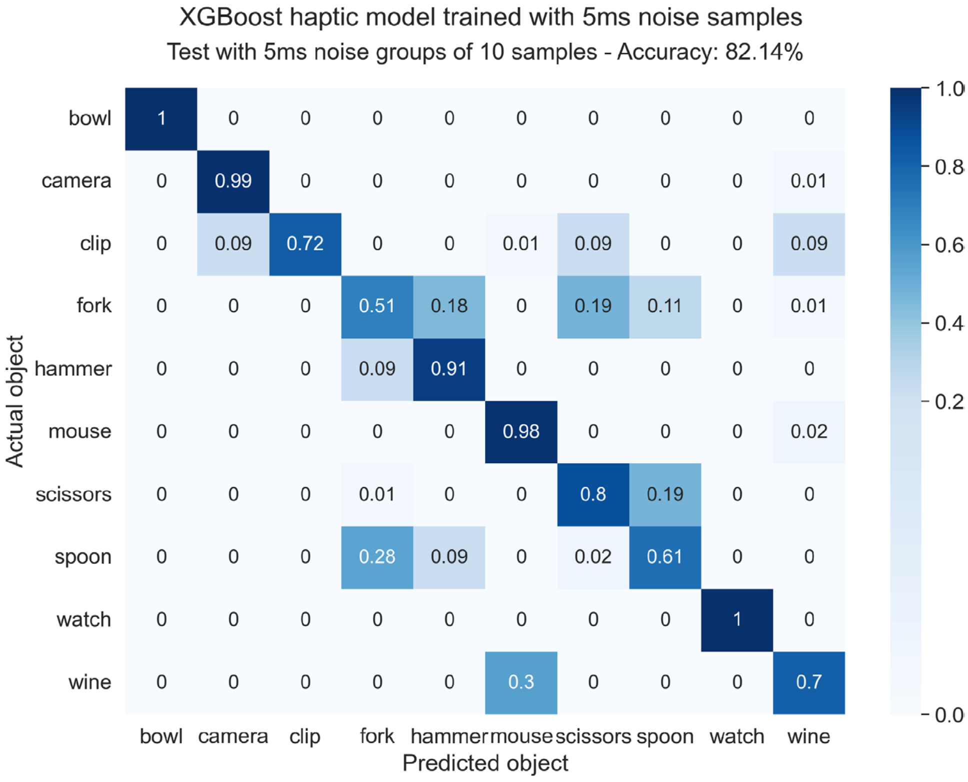


Fig. 13 Bayesian XGBoost classifier's confusion matrix with 10 samples and 5 ms of noise added (82.14% accuracy)

enable the exchange of information between modalities, as is the case with humans (Tabrik et al. 2021).

Based on our findings, the construction of a real three-finger end-effector and the linking of its capture data to the virtual model proposed in this paper becomes the most relevant and immediate challenge. In the context of the captured data, an in-depth study of possible alternatives to the proposed learning algorithm can lead to even better accuracies.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s41315-024-00402-6.

Acknowledgements The 3D objects that were used are from shapenet (Chang et al. 2015), an Information-Rich 3D Model Repository of Stanford University, Princeton University and Toyota Technological Institute at Chicago. The industrial robot arm 3D model is from free3d.com by grabowyr. The work has been partially supported by the CMES project (grant No. PID2020-119725GA-I00) funded by the Spanish Ministry of Science and Innovation. The authors would also like to thank the Departament de Recerca i Universitats (Generalitat de Catalunya) under Grant Ref. 2021 SGR 01396.

Author Contributions G. G., C. P, D. M conceived of the presented idea. A. G. and C. S. developed the theory. G. G., P. N., C. S., C. P., T. H., M. V., C. R., L. V., O. d J. performed the computations. A. G., R. R., D. M. supervised the findings of this work. All authors discussed the results and contributed to the final manuscript.

Funding Open Access funding provided thanks to the CRUE-CSIC agreement with Springer Nature.

Data availability Data is provided within the manuscript or supplementary information files

Declarations

Conflict of interest The authors declare no conflict of interest.

References

Allen, P.K., Roberts, K.S.: Haptic object recognition using a multi-fingered dextrous hand. Technical report, Columbia University (1988)

Chang, A.X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., Xiao, J., Yi, L., Yu, F.: ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago (2015)

Chen, T., Guestrin, C.: XGBoost: a scalable tree boosting system In: Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, NY: ACM; 2016: 785–94 (2016)

Dahiya, R., Oddo, C., Mazzoni, A., Jörntell, H.: Biomimetic tactile sensing. In: Ngo, T.D. (ed.) Biomimetic Technologies, pp. 69–91. Woodhead Publishing, (2015)

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 248–255 (2009). Ieee

Dong, S., Yuan, W., Adelson, E.H.: Improved gelsight tactile sensor for measuring geometry and slip. In: 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 137–144 (2017). IEEE

Garrofé, G., Parés, C., Gutiérrez, A., Ruiz, C., Serra, G., Miralles, D.: Virtual haptic system for shape recognition based on local curvatures. In: Magnenat-Thalmann, N., Interrante, V., Thalmann, D., Papagiannakis, G., Sheng, B., Kim, J., Gavrilova, M. (eds.) Advances in Computer Graphics, pp. 41–53. Springer, Cham (2021)

Gomes, D.F., Paoletti, P., Luo, S.: Generation of gelsight tactile images for sim2real learning. arXiv preprint arXiv:2101.07169 (2021)

Goodwin, A.W., John, K.T., Marceglia, A.H.: Tactile discrimination of curvature by humans using only cutaneous information from the fingerpads. Exp. Brain Res. **86**(3), 663–672 (1991)

Gopnik, A.: Scientific thinking in young children: Theoretical advances, empirical research, and policy implications. Science **337**(6102), 1623–1627 (2012)

Gorges, N., Navarro, S.E., Wörn, H.: Haptic object recognition using statistical point cloud features. In: 2011 15th International Conference on Advanced Robotics (ICAR), pp. 15–20 (2011). IEEE

Hogan, F.R., Jenkin, M., Rezaei-Shoshtari, S., Girdhar, Y., Meger, D., Dudek, G.: Seeing through your skin: Recognizing objects with a novel visuotactile sensor. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision, pp. 1218–1227 (2021)

Juliani, A., Berges, V., Vckay, E., Gao, Y., Henry, H., Mattar, M., Lange, D.: Unity: A general platform for intelligent agents. CoRR (2018) arXiv:1809.02627

Levine, S., Pastor, P., Krizhevsky, A., Ibarz, J., Quillen, D.: Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. Int. J. Robot. Res. **37**(4–5), 421–436 (2018)

Lin, X., Willemet, L., Bailleul, A., Wiertlewski, M.: Curvature sensing with a spherical tactile sensor using the color-interference of a marker array. In: 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 603–609 (2020). https://doi.org/10.1109/ICRA40945.2020.9197050

Lu, C., Wang, J., Luo, S.: Surface following using deep reinforcement learning and a gelsighttactile sensor. CoRR (2019) arXiv:1912.00745

Luo, S., Mou, W., Althoefer, K., Liu, H.: Iterative closest labeled point for tactile object shape recognition. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 3137–3142 (2016). IEEE

Polic, M., Krajacic, I., Lepora, N., Orsag, M.: Convolutional autoencoder for feature extraction in tactile sensing. IEEE Robot. Autom. Lett. **4**(4), 3671–3678 (2019)

Spiers, A.J., Liarokapis, M.V., Calli, B., Dollar, A.M.: Single-grasp object classification and feature extraction with simple robot hands and tactile sensors. IEEE Trans. Haptics **9**(2), 207–220 (2016)

Tabrik, S., Behroozi, M., Schlaffke, L., Heba, S., Lenz, M., Lissek, S., Güntürkün, O., Dinse, H.R., Tegenthoff, M.: Visual and tactile sensory systems share common features in object recognition. eneuro **8**(5):(2021)

Wu, B., Akinola, I., Varley, J., Allen, P.: Mat: Multi-fingered adaptive tactile grasping via deep reinforcement learning. arXiv preprint arXiv:1909.04787 (2019)

Yuan, W., Dong, S., Adelson, E.H.: Gelsight: High-resolution robot tactile sensors for estimating geometry and force. Sensors **17**(12), 2762 (2017)

Zhang, M.M., Kennedy, M.D., Hsieh, M.A., Daniilidis, K.: A triangle histogram for object classification by tactile sensing. In: 2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4931–4938 (2016). IEEE

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Anna Gutiérrez is a telecommunications engineer currently working as a Pre-Sales Engineer at Telefonica, focusing on major accounts in the Catalonia Public Administration. She holds a Master's in Telecommunication Engineering from La Salle - Ramon Llull University and has experience as a software developer and researcher, with a background in digital systems, microprocessors, Machine Learning, and Human-Computer Interaction.

Guillem Garrofé serves as CTO at SPAICE Technology Ltd leading the Research and Development of ML-enhanced perception and control for scalable space logistics. He earned two B.Sc. degrees in Electronics and Computer Engineering from La Salle–Universitat Ramon Llull and a MSc degree in Artificial Intelligence and Machine Learning from Imperial College London. Next, he joined as a PhD candidate the International Max Planck Research School of Intelligent Systems studying whole-body manipulation of objects through tactile sensing.

Pau Nonell earned an MSc in Intelligent Interactive Systems, at Pompeu Fabra University, and worked as an Associate Professor and Researcher in La Salle - Universitat Ramon Llull both being based in Barcelona. His research work and interests include Human-Computer Interaction, Human-Robot Interaction and Computer Vision. He is currently working as a Creative Technologist, exploring the artistic possibilities of using cutting-edge interactive technology.

Claudia Serrano completed a Bachelor's degree in Telecommunications Engineering at LaSalle Barcelona and is currently pursuing a Master's degree in AI at Bath University. With a strong passion for unlocking machine learning's potential in real-world applications, Claudia brings robust technical expertise and collaborative communication skills to any team.

Carlota Parés-Morlans is a Ph.D. student in the Computer Science Department at Stanford University. Her research interests lie at the intersection of robotics, computer vision, and machine learning. She previously earned her M.Sc. in Electrical Engineering from Stanford in 2023, supported by "La Caixa" Fellowship. Prior to Stanford, she completed two bachelor's degrees in Computer Engineering and Networks and Telecommunications Engineering at Ramon Llull University in Barcelona, Spain.

Tomás van den Heijkant earned his B.Sc. in Electrical Engineering from La Salle Ramon Llull University and is currently pursuing a Master's in Embedded Systems at Eindhoven University of Technology. His research interests focus on Embedded Battery Management Systems for multi-cell battery packs.

Mireia Vera holds a B.Sc. in Telecommunication Systems Engineering from La Salle URL, Barcelona, completed in 2024. During her studies, she collaborated with the Seamless research group, contributing to research on artificial cognitive systems' visuohaptic crossmodality. She is currently pursuing an M.Sc. in Communication Technologies and System Design at the Technical University of Denmark.

Conrado Ruiz Jr. earned his master's and doctoral degrees from the School of Computing at the National University of Singapore. He completed his bachelor's degree in computer science at De La Salle University (DLSU) in Manila. Currently, he is an associate professor at DLSU and is a visiting professor and researcher at La Salle - Universitat Ramon Llull in Barcelona, Spain. His research focuses on computer graphics, computer vision, and computational art.

Laia Vidal holds a B.Sc. in Telecommunications Electronic Engineering from La Salle - Ramon Llull University. Currently working as a Cloud Operations Engineer, Laia has a strong interest in robotics and machine learning, with a focus on leveraging these fields to develop innovative solutions.

Alejandro González a Bachelor in Electronics academic coordinator and researcher at La Salle, Ramon Llull University (Barcelona), he received his bachelor degree in electronics at National University of Colombia, then he obtains his M.Sc. and PhD. on Computer Vision and artificial intelligence at Autonomous University of Barcelona. His research topics are concentrated on Computer Vision, Human-Computer Interaction and Robotic vision, participating on several research projects, from 3D reconstruction-based mental diseases diagnosis, or cross-modal artificial agents (visuo-haptic sense) to autismrobot interaction analysis.

Òscar de Jesús earned his B.Sc. degree in Electrical Engineering from La Salle Ramon Llull University, where he is currently pursuing a second degree in Computer Engineering. His research interests encompass the behavior and functioning of the human brain, with a focus on neuroscience and signal processing.

Dr. Raquel Ros is a senior researcher in Social Robotics at PAL Robotics SL, where she leads research and development tasks for social robots in the healthcare field since 2022. She obtained her PhD in Computer Science from the Universitat Autònoma de Barcelona in 2008 working in multi-robot coordination. She then moved to Toulouse as a Marie-Curie fellow to work in the area of HRI at LAAS-CNRS. Next, she joined Imperial College London to continue her research on social robots in educational environments. After three years in industry, working at Cambridge Consultants as user-centre designer, she went back to both academia and Spain, where she enrolled as Lecturer in Robotics at La Salle-Universitat Ramon Llull leading the Robotics research line.

David Miralles is an associate professor at La Salle-Universitat Ramon Llull. He holds a PhD in theoretical physics from the Universitat de Barcelona. He has collaborated with research centres such as Unicamp (Brazil), ICTP (Italy), MIT (USA), the Observatoire de Paris (France) and the Imperial College (UK). He currently leads a research group on human-computer interaction and his interests are centred on the perceptual interpretation of reality that both develop.