Underwater Robot-Object Contact Perception using Machine Learning on Force/Torque Sensor Feedback

Nawid Jamali[1,2], Petar Kormushev[1], Arnau C. Viñas[3], Marc Carreras[3], and Darwin G. Caldwell[1]

***Abstract*— Autonomous manipulation of objects requires reliable information on robot-object contact state. Underwater environments can adversely affect sensing modalities such as vision, making them unreliable. In this paper we investigate underwater robot-object contact perception between an autonomous underwater vehicle and a T-bar valve using a force/torque sensor and the robot's proprioceptive information. We present an approach in which machine learning is used to learn a classifier for different contact states, namely, a contact aligned with the central axis of the valve, an edge contact and no contact. To distinguish between different contact states, the robot performs an exploratory behavior that produces distinct patterns in the force/torque sensor. The sensor output forms a multidimensional time-series. A probabilistic clustering algorithm is used to analyze the time-series. The algorithm dissects the multidimensional time-series into clusters, producing a one-dimensional sequence of symbols. The symbols are used to train a hidden Markov model, which is subsequently used to predict novel contact conditions. We show that the learned classifier can successfully distinguish the three contact states with an accuracy of 72% ± 12 %.**

I. INTRODUCTION

Autonomous manipulation of objects in underwater environments is a challenging task. The turbidity in water makes visual detection of objects unreliable. Thus, sensing a direct contact with an object is an important modality that can help in successfully inspecting and/or manipulating the object. Applications of such a system is not limited to underwater manipulation. For example, in disaster zones a burst pipe may reduce the quality of vision. It can also be applied to a number of manipulations tasks including domestic robots, search and rescue, and manipulation in hazardous environments such as decommissioning of a nuclear plant.

In this paper we investigate the problem of underwater contact perception that relies only on direct contact. Within the framework of the PANDORA project [1], [2] we consider an object inspection problem, in which, an autonomous underwater vehicle (AUV) inspects and manipulates a T-bar valve attached to a control panel. The AUV is free floating in the water, that is, it is not allowed to dock during inspection and manipulation. We propose a machine learning approach that predicts the location of a contact between a robotic gripper, attached to the AUV, and the T-bar valve. Such a system is necessary to allow the AUV to safely manipulate the valve.

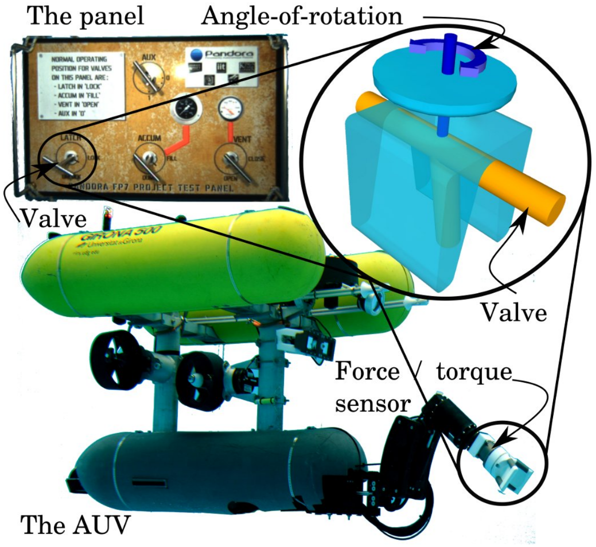


Fig. 1. Experimental setup for underwater experiments.

We propose an exploratory behavior that can be used to determine the contact location. At each contact location of interest, the exploratory behavior excites the force/torque sensor differently, producing a unique pattern. The sensor data forms a multidimensional time-series. Contact states such as no-contact/contact with the valve can be detected by analyzing the temporal patterns that emerge in the time-series data. Our contribution is a robust underwater contact state estimation that relies only on non-vision data, namely, the contact forces and torques.

We learn a classifier for predicting a contact state using our previously developed method to analyze temporal patterns in a multidimensional time-series data [3]. In this paper we apply the learning method to a new problem, i.e., underwater robot-object contact state estimation. We modified the method by adding the control signal in the feature space of the algorithm to correlate patterns that arise due to control commands. In Section III we provide the details of the method and describe how it is applied to the current problem.

Fig. 1 shows the experimental setup for underwater experiments used to test the proposed method. A passive gripper is attached to a robotic arm (ECA robotics ARM 5E Micro). A waterproofed ATI Mini-45 force/torque sensor is sandwiched

This research has received funding from the European Unions Seventh Framework Programme under grant agreement no. ICT-288273.

[1] Department of Advanced Robotics, Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova, Italy.

[2] iCub Facility, Istituto Italiano di Tecnologia, via Morego, 30, 16163 Genova, Italy.

[3] Computer Vision and Robotics Group (VICOROB), University of Girona, 17071, Girona, Spain.

Email:{nawid.jamali, petar.kormushev, darwin.caldwell}@iit.it, {arnau.carrera, marc.carreras}@udg.edu

between the gripper and the arm's end-effector. The arm is connected to an autonomous underwater vehicle (AUV), the Girona 500 [4]. A T-bar valve is attached to a panel that resembles a typical control panel found in underwater structures. Under a teleoperator's control, the robot approaches the panel and performs the proposed exploratory behavior, which as illustrated by the blue arrow in Fig. 1, is a rotary movement with a given angle-of-rotation. This action excites the force/torque sensor differently at each contact location. We show that using the proposed exploratory behavior the robot is capable of learning a classifier to successfully predict the contact state: no-contact, edge-contact, and center-contact.

II. RELATED WORK

Earlier research in valve detection and manipulation assumes a structured environment, where the robot stores detailed information about the environment [5], [6]. Non-vision sensors such as force/torque sensors, binary touch sensors and inductive proximity sensors have been used to confirm contact and monitor applied forces [5], and detect orientation of a valve handle and manipulate the valve without over-tightening/loosening [6]. However, these approaches have been limited to in-air applications. To facilitate underwater manipulation, grippers instrumented with tactile sensors that can operate underwater have been developed [7], [8].

Anisi et al. [6] propose use of an inductive proximity sensor and a torque sensor to detect the orientation of a metallic T-bar valve handle and subsequently manipulate the valve without over-tightening/over-loosening the valve. However, use of an inductive proximity sensor limits the application of the system to metallic objects.

Marani et al. [9] used vision to locate an underwater object and hook a cable to it so that it can be lifted to surface. Recently, Prats et al. [10] used a laser scanner to build a 3D point cloud of an underwater object, which is then used to plan and execute a grasp. However, in both cases an operator has to indicate the region of interest to the robot. Moreover, vision and laser are adversely affected by turbidity in water.

III. METHODOLOGY

The input to the algorithm consists of six-dimensional force/torque sensor data and robot's proprioceptive information, namely, the AUV's pitch, and the end-effector's depth. The rational for the selected features will be discussed in Section IV-C, where we describe the exploratory behavior. We also include the control command, that is, the angle-of-rotation in our analysis, resulting in a nine-dimensional time-series data. To learn concepts from a multidimensional time-series we divide the problem into two stages: dimensionality reduction using clustering and temporal pattern extraction. In the first stage, probabilistic clustering [11] is applied to discover the intrinsic structure of the data. Clustering transforms the multidimensional time-series into a sequence of symbols, each of which is an identifier for a cluster. Thus, an abstract contact condition such as the location of a contact can be represented by a sequence of cluster identifiers. In the second stage, using a supervised hidden Markov model (HMM), the algorithm analyzes the sequence and builds a probabilistic model of temporal patterns that correspond to different abstract contact conditions.

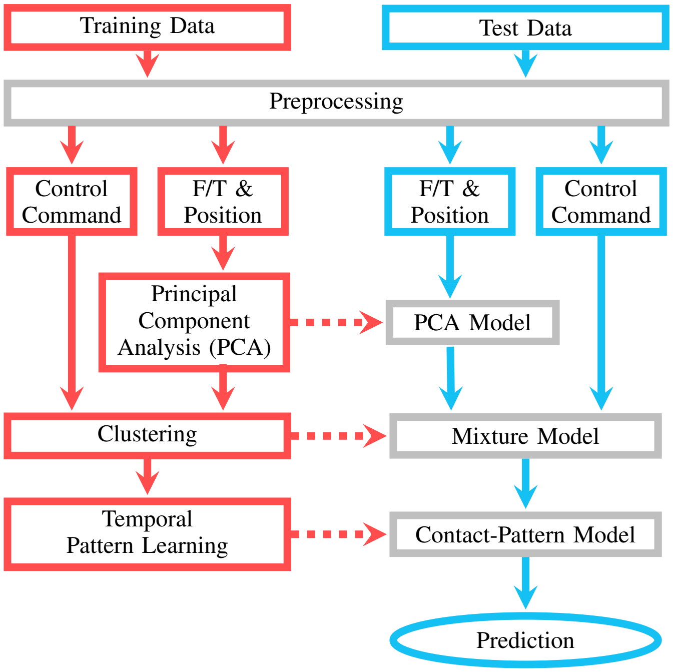


Fig. 2. Overview of the learning method.

An overview of the learning algorithm is shown in Fig. 2. The first step of the analysis involves preprocessing the data. The clustering method assumes that all of the variables are independent [11]. Hence, in the first stage of training, the sensor data are projected onto a new basis using principal component analysis (PCA). The principal component coefficients from the training data are saved. Later, these coefficients are used to transform the test data to the new coordinate system. The control command along with the output of PCA are then used as an input to the clustering algorithm. Clustering plays two roles. Firstly, clustering is used to discover the intrinsic structure within the data. Secondly, it reduces the high dimensional time-series data into a one-dimensional sequence of clusters. The mixture model output by the clustering algorithm is also saved, which, is subsequently used to transform the test data into a sequence of clusters. Each cluster is denoted by a letter. $S = \{\text{BED}\underline{\text{BCCA}}\text{CDDADEC}\underline{\text{BCCA}}\text{EBDA}\ldots\}$ is an example of a sequence of cluster memberships.

In the second stage, the output of the clustering algorithm, that is, the sequence of cluster memberships is analyzed to discover and learn different patterns that represent different contact locations. We want to discover unique patterns that emerge during each contact. For example, in the sequence S, BCCA is a recurring pattern that can be learned as a pattern that represents, say, a center-contact. The model for these patterns is also saved.

The algorithm is tested using an unseen test set. In the testing phase, the models saved during training are used to transform the data into a temporal sequence of cluster memberships. The models for the patterns discovered during training is used to predict the state of a contact. For example, encountering the pattern BCCA will signify a center-contact. We use HMMs to discover the patterns for each contact condition. The following sections provide a detailed description of each step.

A. Preprocessing

All signals are preprocessed before any further analysis is performed. The preprocessing procedures are implemented in MATLAB.

1) Drift compensation: Force/toque sensors use strain gauges to determine applied forces and torques. The output signal of strain gauges drift over time even if there is no applied stimulus. Drift compensation is commonly used in industrial and academic applications of strain gauge based sensors. We use the force/torque data just before a contact to calculate a bias point. The bias point is subtracted from the force/torque output.

2) Filtering: The force/torque data are sampled at 250 Hz. The signals are oversampled to avoid aliasing. The force/torque data is filtered using a digital filter with a 3 dB point of 2 Hz. After application of the digital filters, the data is downsampled by a factor of 25.

3) Scaling: The torque data is an order of magnitude smaller than the forces. We scale the torques by a factor of 10 to make the torques comparable to the forces. Similarly, the AUV's pitch and the angle-of-rotation, which are measured in radians, are scaled by a factor of 10.

B. Principal Component Analysis

Principal component analysis (PCA) is a mathematical transformation that converts a set of variables to a new basis, called the principal components. Each principal component is a linear combination of the original variables. In this space, all variables are orthogonal to each other, i.e, they are uncorrelated. Moreover, in the principal component space, the first component explains the greatest variance in the data, every subsequent component captures lower variance as compared to the preceding component. A consequence of this property is that PCA can also be used as a dimensionality reduction tool.

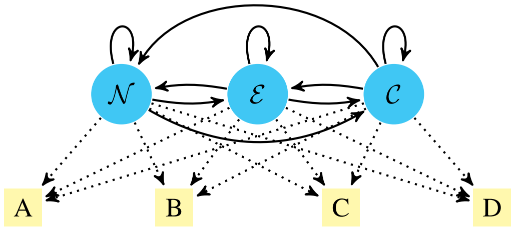
Dimensionality reduction is achieved by only keeping the lower components. A rule of thumb is to drop components once the ratio of cumulative-variance[1] to total-variance has exceeded a predetermined threshold, we used a ratio of 0.9. Hence, in the subsequent analysis we only consider the components that have not exceeded the cumulative-variance threshold. In our experiments, this threshold was exceeded at the 5^{th} component. In the subsequent analysis, we only consider 5 components.

C. Control Command

The control command plays an important role in inducing the sensors. The control signal for our experiments is the commanded angle-of-rotation. We feed the control command directly to the clustering algorithm to allow the algorithm to correlate the control with the sensor data.

$$S_{original} = \{\text{BCCDDCAAAABAABB}\ldots\}$$
$$S_{labelled} = \Big\{(\text{B},\mathcal{N})(\text{C},\mathcal{N})(\text{C},\mathcal{N})(\text{D},\mathcal{N})(\text{D},\mathcal{N})(\text{C},\mathcal{E})$$
$$(\text{A},\mathcal{E})(\text{A},\mathcal{E})(\text{A},\mathcal{E})(\text{A},\mathcal{E})(\text{B},\mathcal{C})$$
$$(\text{A},\mathcal{C})(\text{A},\mathcal{C})(\text{B},\mathcal{C})(\text{B},\mathcal{C})\ldots\Big\}$$

(a) Example of a feature vector.



(b) A three state hidden Markov model.

Fig. 3. A feature vector and a hidden Markov model. The letters A, B, C and D indicate membership to the corresponding cluster. The letters $\mathcal{N}$, $\mathcal{E}$, and $\mathcal{C}$ represent contact states: no-contact, an edge-contact, and a center-contact, respectively.

D. Clustering

We apply probabilistic clustering [11], which uses the minimum message length (MML) principle as the optimization criterion, to build a Gaussian mixture model of the data. We use MML as an optimization criterion to stop the clustering algorithm. Any other method such as the minimum description length [12] will also work.

E. Learning

Once the multidimensional signals from the robot are transformed into a temporal sequence of clusters, we use HMMs to discover the patterns for each contact condition. The training examples are generated by allowing the robot to perform an action. In this case, the action is to perform an exploratory behavior at a contact point. The training sequence is labeled with the contact location, which is recorded during data collection. This allows the algorithm to learn a mapping from the temporal sequence of clusters to a classification of the contact state: no-contact, an edge-contact, and a center-contact. The accuracy of the classifier is tested by applying it to a novel sequence, where the contact state is unknown to the robot.

Fig. 3(a) shows an example of a sequence generated after the application of the clustering algorithm. The corresponding feature vector is also shown and consists of temporal sequence of couples of the form (cluster-membership, contact-state). Fig. 3(b) shows the topology of the HMM used to learn a representation for the emerging temporal patterns. The topology of the HMM is determined by the number of contact states. It has three states, one for each contact location, which will be explained in Section IV-B. The HMM is trained using the sequence of clusters as the observation. When the robot is presented with a novel pattern, the robot uses the model to make a prediction.

[1] Cumulative-variance is calculated by summing the variance of all components up to the component of interest.

IV. Experimental Setup

The experimental setup illustrated in Fig. 1, described in Section I, was used to test the method underwater in a pool. In this section we will describe the waterproof force/torque sensor that was developed to facilitate underwater experiments. We will also define the contact locations, which is followed by a description of the exploratory behavior and a detailed description of the experiments.

A. *Waterproof force/torque sensor*

In order to perform underwater experiments, we have designed and developed an integrated gripper (Fig. 4(a)) that provides a waterproof housing for a camera and a force/torque sensor. The designed gripper is modular, for example, the camera compartment can be removed if it is not needed. The griper can be changed to meet the needs of the task at hand. In our experiments we used a V-shaped passive gripper. The developed system uses the position of the end-effector and the AUV's depth sensor to compensate the forces induced by the water pressure.

B. *Definition of contact locations*

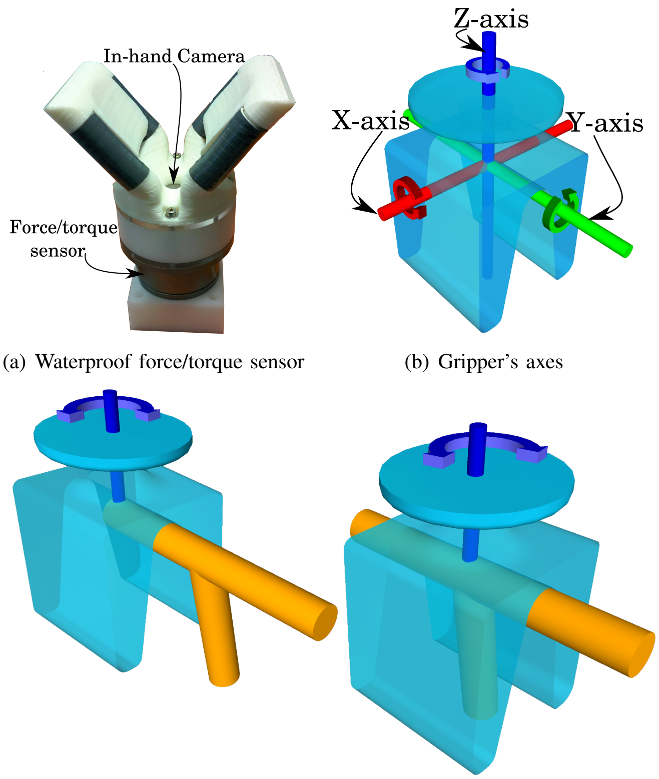
A contact between the gripper and the valve is categorized as no-contact, an edge-contact, and a center-contact. In a center-contact (Fig.4(c) – right), the center of the gripper is aligned with the central axis of the T-bar valve within a tolerance, ϵ, which is the error margin that will allow the robot to safely rotate the valve. We define the value of ϵ as $\frac{1}{4}$ of the length of the valve handle. The valve used in our experiments had a length of 10 cm.

An edge-contact (Fig.4(c) – left) is defined as a contact in which at least $\frac{1}{3}$ of the width of the gripper is in contact with the valve and the central axis of the gripper is at least an ϵ away from the central axis of the T-bar valve. No contact is produced by performing the exploratory behavior in the water without making a contact with any object. We used a 9 cm wide gripper.

C. *The exploratory behavior*

The exploratory behavior is achieved by rotating the gripper around one of its principal axes at a pivot point. Fig. 4(b) shows the principal axes of the gripper. The pivot point is at the gripper-valve boundary. The chosen exploratory behavior is a rotation around the z-axis of the gripper. A behavior is achieved by rotating the gripper around the z-axis by a given angle-of-rotation. The angle-of-rotation can take an arbitrary value. This action induces the sensor to produce different temporal force and torque patterns.

The exploratory behavior also affects the position of the end-effector and the pitch of the AUV. For example, in an edge-contact the end-effector will follow a circular trajectory around the rotational axis of the valve. While, in a center-contact the end-effector will be stationary. Similarly, the pitch of the AUV is affected by such misalignments. Hence, we include them in the feature space of the proposed method.



(a) Waterproof force/torque sensor

(b) Gripper's axes

(c) The exploratory behavior (edge-contact, center-contact)

Fig. 4. The waterproof force/torque sensor and an illustration of the exploratory behavior: a) shows the gripper with integrated waterproof force/torque sensor, b) the gripper's principal axis, and c) illustrates the exploratory behavior for an edge-contact and a center-contact, respectively, which is a rotation around the z-axis of the gripper.

D. *The learning task*

The experimental setup shown in Fig. 1 is used to collect data to train and test the learning algorithm. In this setup, the gripper is attached to an AEC 5E Micro arm. An ATI Mini-45 force/torque sensor is placed between the gripper and the robot's end-effector. The arm is attached to the Girona 500 AUV. The AUV and a panel that has T-bar valves are placed underwater in a pool. The AUV approaches the a valve in the panel and performs the exploratory behavior, which we refer to a trial. In our experiments the robot was controlled, in Cartesian coordinate frame, by a human operator under teleoperation. The controller allows simultaneous control of the AUV and the arm through a joystick. The rotation of the gripper is achieved by controlling the last joint of the robot, which is a rotary joint dedicated for end-effector tool attachments.

A trial consists of a teleoperator positioning the AVU in front of the panel such that the AUV's gripper is positioned a short distance away from the location of interest on the selected valve. In our experiments we use the same valve for all trials. At this point, the surge thrusters of the AUV are switched on with a forward thrust of 10 Newtons. Once the gripper makes contact with the valve, the gripper is rotated twice around the z-axis clockwise/counterclockwise approximately 180°, at least 90°, then rotated in the reverse direction. The direction – clockwise or counterclockwise –

depended on the direction that would produce a 180° rotation without exceeding the software limits of the joint. Note, since we use the control command in our learning method, the direction of the rotation has no adverse effect.

To expose the algorithm to variations in the gripper-valve contact, each location of interest is sampled three times. We sampled both edges, namely, left edge and right edge of the valve for the edge-contact, hence the dataset has twice as many samples for the edge-contact as the center-contact. The attached video visualizes a trial, described in this section, at different locations of interest.

Three separate datasets were collected resulting in a large database consisting of nine samples for each location. For the no-contact scenario we collected one sample per set.

V. Results

The feature vector for the learning algorithm is a nine-dimensional time-series that consists of the six-dimensional force/torque sensor data, the AUV's pitch, the end-effector's position and the commanded angle-of-rotation. As described in Section III-E, a three state HMM was trained for the exploratory behavior.

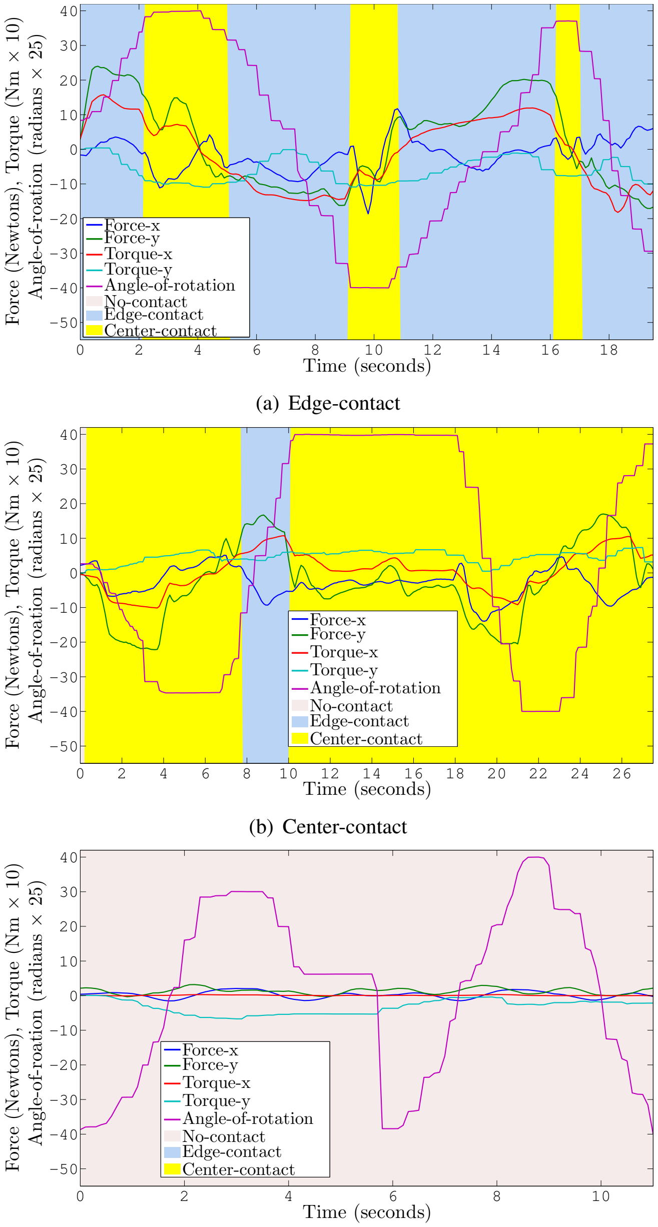
A. Evaluation method

1) Threefold cross-validation: To evaluate the proposed approach an HMM model was learned using two of the three dataset while the third dataset was used to evaluate the accuracy of the learned model. We repeat this process to get a threefold cross-validation of the performance of the proposed method.

2) Classification point: As described earlier in Section IV-D, in each sample the gripper was rotated twice, that is, from the starting position to a final position of at least 90° away from the starting position, then back, twice. This results into four starting points sample of a contact location. Since we are interested in assessing the state of the contact for safe manipulation, we take the prediction of the classifier at these points as the prediction of the contact state to evaluate the accuracy of the proposed method. We wait n samples after the gripper starts rotating to allow the HMMs to converge. The value of n is arbitrary. Smaller values will increase the safety margin, but the prediction may not be reliable. In our experiments, we took a sample halfway between the starting point and the stopping point, at this point the HMM should have enough evidence to converge.

B. Classifier performance

The performance of the method was evaluated as described in Section V-A. Fig. 5 shows an example of the classifier's output for each contact state. The colored regions in the figures represent the classifier output. Note, that the classifier makes a prediction at each sample point, hence the classification changes over time. Fig. 5(a) shows the classifier's output for an edge-contact. We notice that the classifier's output changes from a edge-contact to a center-contact at approximately 2 s mark, which then changes back to edge-contact at approximately 5 s mark. By examining the data we notice that the angle-of-rotation is stationary throughout this period, which explains the misclassification. When the gripper is stationary, the forces and torques resemble the center-contact condition. As the gripper starts moving again at 5 seconds mark, the classifier correctly outputs an edge-contact prediction.



(a) Edge-contact

(b) Center-contact

(c) No-contact

Fig. 5. Examples of classifier output for each contact location. The colored regions represent the classifier output, for example, in Fig. 5(a) for the first 2 seconds it is correctly classified as an edge-contact. At this point the contact is classified as a center-contact, which is followed by an edge-contact at 5 s mark.

Fig. 5(b) shows the output of the classifier for the center-contact. We notice that the center-contact is misclassified as an edge-contact between 8 seconds and 10 seconds. One reason for such misclassification during the transitionary period of the angle-of-rotation could be due to temporary movements of the gripper away from the center of the valve. Fig. 5(c) shows the output of the classifier for the no-contact

TABLE I
CONFUSION MATRIX, ACCURACY 72% ± 12%, USING 95% CONFIDENCE INTERVAL.

No-contact	Edge-contact	Center-contact	Class
10	0	2	No-contact
0	55	17	Edge-contact
0	14	22	Center-contact
83 %	76 %	61 %	Accuracy

case where the classifier correctly predicts a no-contact.

Table I shows the confusion matrix for the learned policies. The confusion matrix is generated by superimposing the results of all three sets. Note that for the no-contact state we collected one sample per set instead of three. We are able to classify with an accuracy of 72% ± 12%. Studying the confusion matrix, we notice that the center-contact has larger misclassifications compared to an edge-contact. One reason for this is that, as the gripper rotates, it can shift from the center to the edge of the valve. Similarly, it is possible for an edge-contact to shift towards the center of the valve. We label a trial based on the initial contact location. Hence, some of these misclassifications may arise due to a mismatch between the ground truth and the label.

VI. DISCUSSION

A. Use of hidden Markov models

Any temporal machine learning method, for example, recurrent neural networks, can be used to solve the problem. However, learning directly from a nine-dimensional time-series requires a large dataset. Gathering a large dataset in underwater experiments is not practical. Therefore, we analyze the time-series using the proposed method, which transforms the data into a one-dimensional sequence of clusters, resulting in a symbolic sequence. Hidden Markov models (HMMs) are good at discovering such patterns in symbolic time-series. Hence, we chose to use HMMs for the learning task.

B. Effect of teleoperation

The experimental setup is designed to allow application and testing of the method underwater. In our experiments, the AUV and the arm are teleoperated, which is justified by the nature of the underwater robotics. Firstly, autonomous operation of the robot is still in progress. Secondly, the perturbations experienced due to teleoperation makes sure that the data collected can handle such cases, which it will experience in natural waters where the robot will be moved by currents.

VII. FUTURE WORK

In future we will consider a larger repertoire of behaviors that provide more detailed information such as the orientation of the valve or whether the valve is stuck. We will also look into improving the method to autonomously choose an exploratory behavior that, based on its current belief, will be more informative in increasing the probability of the contact state estimate.

Future research will also consider relaxing the approximate knowledge of the location of the valve, where a blind search of the robot's workspace will be conducted for detecting the valve. We will also look into extending the method to handle other types of valve found in industrial/underwater environments.

VIII. CONCLUSION

We have presented a method that uses machine learning to successfully perceive a contact state between the gripper of a robot and a T-bar valve in an underwater environment. The main contribution of this paper is a method that can robustly perceive a contact state using non-vision information such as force/torque sensor data. We show that the method is capable of perceiving underwater by distinguishing three contact states, namely, a no-contact state, an edge-contact, and a center-contact with an accuracy of 72% ± 12%.

REFERENCES

[1] D. M. Lane, F. Maurelli, P. Kormushev, M. Carreras, M. Fox, and K. Kyriakopoulos, "Persistent autonomy: the challenges of the PANDORA project," *Proc. IFAC MCMC*, 2012.

[2] PANDORA, "Persistent Autonomy through learNing, aDaptation, Observation, and Re-plAnning," 2012.

[3] N. Jamali and C. Sammut, "Slip prediction using hidden Markov models: Multidimensional sensor data to symbolic temporal pattern learning," in *IEEE Int. Conf. Robot. Autom.*, 2012, pp. 215–222.

[4] D. Ribas, N. Palomeras, P. Ridao, M. Carreras, and A. Mallios, "Girona 500 AUV: From Survey to Intervention," pp. 46–53, 2012.

[5] M. A. Abidi, R. O. Eason, and R. C. Gonzalez, "Autonomous robotic inspection and manipulation using multisensor feedback," *Computer (Long. Beach. Calif).*, vol. 24, no. 4, pp. 17–31, 1991.

[6] D. A. Anisi, E. Persson, and C. Heyer, "Real-world demonstration of sensor-based robotic automation in oil & gas facilities," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, Sep. 2011, pp. 235–240.

[7] D. M. Lane, J. B. C. Davies, G. Casalino, G. Bartolini, G. Cannata, G. Veruggio, M. Canals, C. Smith, D. J. O'Brien, M. Pickett, G. Robinson, D. Jones, E. Scott, A. Ferrara, D. Angelleti, M. Coccoli, R. Bono, P. Virgili, R. Pallas, and E. Gracia, "AMADEUS: advanced manipulation for deep underwater sampling," *Robot. Autom. Mag. IEEE*, vol. 4, no. 4, pp. 34–45, 1997.

[8] J. Lemburg, P. Kampmann, and F. Kirchner, "A small-scale actuator with passive-compliance for a fine-manipulation deep-sea manipulator," in *Ocean. 2011*, 2011, pp. 1–4.

[9] G. Marani, S. K. Choi, and J. Yuh, "Underwater autonomous manipulation for intervention missions AUVs," *Ocean Eng.*, vol. 36, no. 1, pp. 15–23, Jan. 2009.

[10] M. Prats, J. J. Fernandez, and P. J. Sanz, "Combining template tracking and laser peak detection for 3D reconstruction and grasping in underwater environments," in *IEEE/RSJ Int. Conf. Intell. Robot. Syst.*, 2012, pp. 106–112.

[11] C. S. Wallace and D. L. Dowe, "MML clustering of multi-state, Poisson, von Mises circular and Gaussian distributions," *Stat. Comput.*, vol. 10, no. 1, pp. 73–83, 2000.

[12] J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, no. 5, pp. 465–471, 1978.