Unsupervised Contextual Task Learning and Recognition for Sharing Autonomy to Assist Mobile Robot Teleoperation

Ming Gao, Ralf Kohlhaas and J. Marius Zöllner
FZI Research Center for Information Technology, 76131 Karlsruhe, Germany

Keywords: Shared Autonomy, Assisted Teleoperation, Mobile Robot, Unsupervised Learning from Demonstration.

Abstract: We focus on the problem of learning and recognizing contextual tasks from human demonstrations, aiming to efficiently assist mobile robot teleoperation through sharing autonomy. We present in this study a novel unsupervised contextual task learning and recognition approach, consisting of two phases. Firstly, we use Dirichlet Process Gaussian Mixture Model (DPGMM) to cluster the human motion patterns of task executions from unannotated demonstrations, where the number of possible motion components is inferred from the data itself instead of being manually specified *a priori* or determined through model selection. Post clustering, we employ Sparse Online Gaussian Process (SOGP) to classify the query point with the learned motion patterns, due to its superior introspective capability and scalability to large datasets. The effectiveness of the proposed approach is confirmed with the extensive evaluations on real data.

1 INTRODUCTION

We focus on mobile robot teleoperation, which has been widely applied to situations where human excursion is impractical or infeasible, such as search and rescue in hazardous environments, and telepresense for social needs in domestic scenarios. Due to time delay and lack of situational awareness (SA), it is troublesome and stressful for the human operator to simply teleoperate the robot without assistance. On the other hand, robot cannot yet carry out tasks alone based on the current achievements in cognitions and controls. Hence the human and the robot have to cooperate with each other in an appropriate way, in order to efficiently perform tasks in remote. The major challenge is how to best coordinate the two intelligent sources from the human and the robot, to guarantee an optimal task execution, which has been the research focus of *shared autonomy* (Sheridan, 1992).

To address the challenge, we argue that, the robot is supposed to recognize the on-going tasks the human operator performs with the contextual information, aiming to provide proper motion assistance in a task-aware manner, in order to optimally cope with the human operator to improve the remote task performance. Such strategy is proved to promote task performance in the presence of time delays (Hauser, 2013), because the robot is able to predict the desired task in the midst of a partially-issued command. In our work, a *task* refers to an intermediary between *user intention* and *robot action*, which is a metric representation of the user intention for a robot to complete an *action primitive*.

Since the way the user executes a task is implicit, most state-of-art approaches on this topic employ supervised learning approaches to encode and recognize the human motion patterns executing multiple task types from labelled demonstrations (Stefanov et al., 2010)(Hauser, 2013)(Gao et al., 2014). However, it is difficult for the human expert to manually segment a demonstration into meaningful action primitives for the robot to learn, and in the long run, the manual annotation will be error-prone to limit the applicability of the system, when demonstration data for more and more task types need to be labelled. To scale with such situation, we report in this study a novel unsupervised contextual task learning and recognition approach, consisting of two phases. In the first place, we use Dirichlet Process Gaussian Mixture Model (DPGMM) to cluster the motion patterns of task executions from demonstrations without labels. The major advantage of applying DPGMM for clustering is that the number of possible motion modes (*i.e.* motion clusters) is inferred from the data itself instead of being manually specified *a priori* or determined through model selection, which is required when using *e.g.* Gaussian Mixture Model (GMM) and K-Means on this topic. Moreover, we are able to dis-

Gao, M., Kohlhaas, R. and Zöllner, J.
Unsupervised Contextual Task Learning and Recognition for Sharing Autonomy to Assist Mobile Robot Teleoperation.
DOI: 10.5220/0005972002380245
In *Proceedings of the 13th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2016) - Volume 2*, pages 238-245
ISBN: 978-989-758-198-4

cover both overlaps and distinctions of the task execution patterns through clustering, which can be used as the knowledge base for interpreting the query patterns later on. Post clustering, we employ a non-parametric Bayesian method, *i.e.* Sparse Online Gaussian Process (SOGP) classifier to classify the query point with the learned motion clusters during operation, due to its superior introspective capability over other state-of-art classifiers, such as Support Vector Machine (SVM), which is probably most widely used algorithm for classification. Thanks to this property, the SOGP classifier favours an *active learning* strategy, where the robot can actively ask for demonstrations to add to training datasets when facing *un-modelled* motion patterns. Meanwhile, the SOGP classifier is able to maintain scalability to large datasets, which is significant for our application.

The remainder of this paper is organized as follows. Section 2 introduces the related work. Section 3 describes the proposed approach in detail. The experimental results and discussions are presented in section 4. Finally, we summarize the conclusions in section 5.

2 RELATED WORK

Prior work in the field of assisted teleoperation has been mainly conducted in the context of telemanipulation. Dragan *et al.* (Dragan and Srinivasa, 2012) formalize the assisted teleoperation to consist of two parts: the user intention prediction and the arbitration of the user input and the robot's prediction. In their work, the user intention is assumed to track *optimal* trajectories towards the grasp objects. Haus (Hauser, 2013) proposes a task inference and motion planning system to assist the teleoperation of a 6D robot manipulator using a 2D mouse. However, it concentrates on the *freeform* tasks compared with our problem of recognizing the contextual tasks, which depends on the information of related objects in the environment.

In the context of assisting mobile robot teleoperation, Fong *et al.* (Fong et al., 2001) propose a dialogue based approach, where the user can communicate explicitly with the robot to obtain a better situational awareness. Though being intuitive for the human operator, the dialogue based framework can increase the user workload, especially when robot raises a huge amount of questions through the dialogues. Sa *et al.* report a shared autonomy control scheme for a quadcopter in (Sa et al., 2015), aiming to assist the inspection of vertical infrastructure, where an unskilled user is able to safely operate the quadcopter in close proximity to target structures due to on-board sensing and partial autonomy of the robot. Okada *et al.* (Okada et al., 2011) introduce a shared autonomy system for tracked vehicles. Based on the continuous three-dimensional terrain scanning, the system actively assists the control of the robot's flippers to reduce the workload of the operator, when the robot is teleoperated to traverse rough terrains. Both works consider only single task type for assistance, *i.e.* inspection of vertical infrastructure or rough terrain traversal. In contrast, our approach is able to recognize and support multiple contextual task types through learning motion patterns from demonstrations.

3 METHODOLOGY

3.1 Overview

To employ machine learning approach to learn the human motion patterns performing various contextual task types from demonstrations, the motion patterns are described with a set of *task features*, which will be firstly introduced in this section. The technique details of employing DPGMM and SOGP on clustering and recognizing the motion patterns of task executions from unlabelled demonstrations will be presented respectively in the following parts of the section.

3.2 Task Feature

A task feature $\mathbf{q}$ embodies an instantiation of the motion pattern executing certain contextual task type, which is built upon environmental information and user input $\mathbf{u}$.

The user input $\mathbf{u}$ is issued from a normal joystick, which consists of translational velocities along x and y axes, and rotational velocity around z axis in the robot's local coordinate frame: $\mathbf{u} = (v_x, v_y, v_\omega)$. Each input channel is normalized to the range of $(-1, 1)$, where the positive sign indicates that, for the translational velocities, the input is along the positive direction of the corresponding axis, and for the rotational velocity, it is in the counter-clockwise direction around z axis.

The environmental information is encoded with the *intentional target point* $\mathbf{s}$, which is extracted from the semantic components of indoor scenarios, *i.e.* doorway, object and wall segments, and transferred to a two-dimensional coordinate in the local frame fixed on the robot center: $\mathbf{s} = (x_\eta, y_\eta)$, since we employ a 2D Laser Range Finder (LRF) to perceive the environment, but it is straightforward to extend the definition to 3D configuration. More specifically, for doorway, we select its center point as the intentional target point for the robot to reach or cross. For object and wall segments, we choose their nearest surface points to the robot center during operation to be the intentional target point for the robot to follow, since the surface points of an object or wall segment implicitly characterize its shape.

In addition to $\mathbf{s}$ and $\mathbf{u}$, we compute the angle θ between the user input vector $\mathbf{u}_{xy} = (v_x, v_y)$ and the vector $\mathbf{r}_s$ from the robot center to $\mathbf{s}$, to be part of a task feature. θ represents the user input direction, hence the movement direction of the robot, relative to $\mathbf{s}$, which bridges two sources of contextual information: environmental perception and user input, and vaguely indicates the user intention for operating the robot regarding the corresponding semantic components. Based on the above introductions, a task feature can be expressed as $\mathbf{q} = (\mathbf{s}, \theta, \mathbf{u})$.

Although we consider just doorway, object and wall to construct task feature here, it is intuitive to obtain task feature from more types of semantic components of the environment, such as docking place, where the intentional target point is the center point of the docking area, and the human target, *e.g.* when the task is to follow a human during telepresence, the intentional target point of which can be the position of the detected human.

The following part will report how we apply DPGMM to cluster motion patterns with task features.

3.3 Motion Clustering with DPGMM

During demonstration, we obtain the task features $\mathbf{q}$ computed from the target semantic components of the scenario, which are described with no specific task type, *i.e.* unlabelled.

To discover possible clusters of motion patterns (*i.e.* modes) from the unlabelled demonstration data, we use the *Dirichlet Process* (DP) prior on the dataset, which allows an infinite collection of modes, and an appropriate number of modes is inferred directly from the data in a fully Bayesian way, without the need for manual specification or model selection (Blei et al., 2006). Mathematically, the DP is described with the *stick-breaking* process:

$$\begin{aligned} &G \sim DP(\alpha_0 H), \\ &G \triangleq \sum_{k=1}^{\infty} \lambda_k \delta_{\phi_k}, \\ &v_k \sim \mathrm{Beta}(1, \alpha_0), \\ &\lambda_k = v_k \prod_{l=1}^{k-1} (1 - v_l). \end{aligned}$$

Where G is an instantiation of the DP consisting of an infinite set of clusters/mixture components, and λ_k denotes the mixture weight of the component k. Each data item n chooses an assignment according to $\mathrm{w}_n \sim \mathrm{Cat}(\lambda)$, and then samples observations $\mathbf{q}_n \sim F(\phi_{\mathrm{w}_n})$. Since $\mathbf{q}_n$ is multi-dimensional real-valued data in our application, we take F to be Gaussian. ϕ_k is the data-generating parameter for the component k, which is drawn from the *normal-Wishart* distribution H with natural parameters ρ_0, facilitating the full-mean, full-covariance analysis.

At the heart of DPGMM is the inference technique, whose goal is to recover stick-breaking proportion v_k and data-generating parameters ϕ_k for each mixture component k, as well as discrete cluster assignment $\mathrm{w} = \{\mathrm{w}_n\}_{n=1}^{N}$ for each observation from the demonstration dataset, which maximizes the joint distribution:

$$\begin{aligned} p(\mathbf{Q}, \mathbf{w}, \phi, v) = &\prod_{n=1}^{N} F(\mathbf{q}_n | \phi_{\mathrm{w}_n}) \mathrm{Cat}(\mathrm{w}_n | \lambda(v)) \\ &\prod_{k=1}^{\infty} \mathrm{Beta}(v_k | 1, \alpha_0) H(\phi_k | \rho_0). \qquad (1) \end{aligned}$$

We employ a variational Bayesian variant inference algorithm, named *Memoized Online Variational Inference* (Hughes and Sudderth, 2013), to infer the posterior (Eq.1), which scales to large yet finite datasets while avoiding noisy gradient steps and learning rates together, and allows non-local optimization by developing principled birth and merge moves in the online setting. For more details regarding the algorithm, please refer to (Hughes and Sudderth, 2013).

Each learned motion cluster is considered as an *action primitive*, which can be used with the estimated semantic target to interpret the motion patterns of the human operator performing certain contextual tasks associated with the target in the form of the trajectory the human operator intends to execute (will be shown in section 4). Hence the robot can efficiently help with the task execution by assisting the human operator to safely track the intentional trajectory in remote. From this perspective, it is supposed to classify the query task features obtained from multiple candidate semantic components to the learned motion clusters, in order to find the most probable cluster and the associated semantic component during operation, where we employ SOGP classifier to achieve this, which will be covered in the following part.

3.4 Motion Classification with SOGP

To recognize which motion patterns (including the associated semantic targets) the human operator executes, we employ the SOGP classifier (Gao et al., 2016) to classify the query task features to the learned motion clusters. We will briefly introduce how we adapt it to our application in this subsection. For more technical details regarding the SOGP classifier, please refer to (Gao et al., 2016).

Specifically, by following the one-vs-all formulation, we attempt to infer:

$$p(t_*^{(c)}|\mathbf{q}_*,\mathbf{Q}_L,\mathbf{t}_L^{(c)}), \tag{2}$$

where $t_*^{(c)}$ and $\mathbf{t}_L^{(c)}$ indicate the predictive label of a query task feature $\mathbf{q}_*$ and the labels of the motion cluster data $\mathbf{Q}_L$ respectively, with $t^{(c)} \in \{-1,1\}^n$ representing the observation vector of binary labels for cluster $c \in \{1,...,M\}$, where M is the number of the clusters. We assume a zero mean function and that observed values $t_L^{(c)}$ of the latent function are corrupted with independent Gaussian noise with variance $\sigma^2_{(c)}$, resulting into a closed-form solution for the posterior:

$$p(t_*^{(c)}|\mathbf{q}_*,\mathbf{Q}_L,\mathbf{t}_L^{(c)}) = \mathcal{N}(t_*^{(c)}|\mu_*^{(c)},\sigma^2_{*(c)}). \tag{3}$$

Hence the final score of the multi-class classifier is achieved by taking the maximal predictive posterior mean across all clusters:

$$\begin{aligned}\mu_*^{\mathrm{mc}} &= \max_{c=1...M} \mu_*^{(c)} \\ &= \max_{c=1...M} \mathbf{k}_*^T(\mathbf{K}+\sigma^2_{(c)}\mathbf{I})^{-1}\mathbf{t}_L^{(c)},\end{aligned} \tag{4}$$

and returning the corresponding label[1] c, where $\mathbf{k}_* = \kappa(\mathbf{Q}_L,\mathbf{q}_*)$ and $\mathbf{K} = \kappa(\mathbf{Q}_L,\mathbf{Q}_L)$ denote the kernel values of the training set and the query point, which are computed with the squared exponential (SE) kernel in this paper. Meanwhile, we also get the posterior variance $\sigma^2_{*(\mathrm{mc})}$ from the classifier prediction, which facilitates excellent uncertainty estimation together with the predictive posterior mean. This property can be utilized by the robot to discover *unmodelled* data, and ask for an update of the demonstration dataset, *i.e.* favours an *active learning* scenario, which is significant for our application, since we envision a lifelong adaptive assistive robot. Moreover, we would like to maintain the model sparse to limit the amount of storage and computation required, where we use the sparse approximation presented in (Csató and Opper, 2002) to achieve this, which minimises the KL-divergence between the full GP model, and one based on a smaller dataset[2].

4 EXPERIMENTAL RESULTS

4.1 Overview

We employed a holonomic mobile robot to evaluate our approach, which carries a 2D LRF to perceive the environment. The robot was manually driven by the human operator with a normal wireless joystick. The average speed of the robot during the evaluations was approximately $0.3m/s$.

For the convenience of providing demonstrations and analysing the test data, without the loss of generality of our method, in this study, we built the maps of the involved scenarios with the state-of-art SLAM implementation in ROS, and process them to extract the required semantic components beforehand, *e.g.* the center points of the candidate doorways, and the surface points of the candidate objects and walls, respectively.

The following evaluations were made in a *post-experimental stage*. Since we are concerned with whether the most probable semantic component estimated by the SOGP classifier corresponds to the groundtruth target, we computed the rate of the correct correspondence (*i.e.* the *correspondence rate*, or CR for short) per test trajectory, and obtained the average of the correspondence rates (ACR for short) over all test trajectories, to characterize the recognition performance of the SOGP classifier in the tests. Additionally, to evaluate whether the most probable motion cluster found by the classifier is able to appropriately interpret the motion patterns of a test trajectory, we computed the average dissimilarity of the most probable task feature to all points in the assigned motion cluster with each way point along each test trajectory (*i.e.* the intra-cluster average dissimilarity, or ICAD for short), which measures the “tightness” of the most probable query task feature to the assigned motion cluster. This metric is meaningful, since the tightness of the classifications measures how well the proposed approach interprets the motion patterns of a test trajectory: a good interpretation of motion patterns requires a low dissimilarity of the most probable query point to the points in the assigned motion clusters. We used L_2-norm as the dissimilarity metric, and obtained the mean of ICAD (MICAD for short) over all way points along all test trajectories, which was utilized together with ACR to evaluate the performance of the proposed approach. Meanwhile, over the following evaluations post clustering, we employed the discriminative SVM classifier[3] with the SE

[1] If there exist multiple semantic components, the SOGP classifier returns both the cluster label and the associated semantic component with the highest score for each query point.

[2] The size of this smaller dataset refers to the *capacity* of the SOGP model.

[3] Throughout this work we use LIBSVM (Chang and Lin, 2011) for SVM training and testing.

kernel, to compare with the generative SOGP classifier, and we chose the capacity[4] of the SOGP classifier to be 500.

In the following evaluations, three statements will be verified to show that the proposed approach serves as a generic framework for representing and exploiting the knowledge of the contextual task executions from unlabelled demonstrations, in the context of assisting mobile robot teleoperation by inferring the tasks the human operator performs. First, the proposed approach gives very good recognition results on the test data sampled from the task types used for training in an indoor scenario with multiple candidates, which satisfies the basic requirement for the approach. Second, the proposed approach is generalizable to appropriately interpret the motion patterns of new task types not used for training. Finally and most importantly, the proposed approach is able to detect unknown motion patterns distinctive from those used in the training set, due to the superior introspective capability of the SOGP classifier, which is a key property to make the proposed approach appealing for a life-long adaptive assistive robotic system.

4.2 Performance Evaluation with Known Task Types

This subsection aims to evaluate the performance of the proposed approach on recognition of the task types used for training in an indoor scenario with multiple candidates, which is the basic criterion for the approach.

We collected the demonstration data from performing the four contextual task types: Doorway Crossing (DC), Object Inspection (OI), Wall Following (WF) and Object Bypass (OB), in an indoor scenario with random starting poses and semantic targets respectively. The map of the scenario is shown together with the annotated candidate semantic components in figure 1. Totally, we obtained 12 trajectories for each task type, and there were 2254 way points for Doorway Crossing, 6286 points for Object Inspection, 4104 points for Wall Following and 1827 points for Object Bypass, respectively. Firstly, to show the motion clustering result qualitatively, we employed all the collected trajectories as the training data[5] to be clustered by DPGMM, where we ran the inference algorithm iterating the initial number of clusters from 1 to 100, although the clustering results were generally consistent, we selected one with the highest log likelihood (*i.e.* the evidence), to ensure good results. Figure 2(a) displays the discovered motion clusters and the feature data assigned to them in the form of a stacked histogram, colored by the original task types. Moreover, we plot the feature data with their first two components (*i.e.* $\mathbf{s} = (x_\eta, y_\eta)$) on the joint space and color them according to the original task types and the discovered clusters in figure 2(b) and figure 2(c) respectively. As can be viewed, a majority part of Wall Following and Object Bypass feature data are grouped into two sides, representing the motion patterns which are demonstrated in either left or right side regarding the semantic targets for the two task types respectively. Likewise, a majority part of Object Inspection feature data are assigned to two separate clusters, although not evidently illustrated in in figure 2(b) and figure 2(c), corresponding physically to the situations when the robot is demonstrated to inspect the target objects in either clockwise or counter-clockwise direction. Upon consideration, they are reasonable distinctions, and initially not thought of by the demonstrator. This property is key, since it allows the DPGMM to determine action primitives unknown even to the demonstrator. Meanwhile, most motion clusters consist of a blend of feature data from multiple task types, which represents the overlaps of the motion patterns of them, potentially resulting from that the robot was always operated to firstly align with the target, then approach it during demonstration. On the other hand, the split of the overlap feature data into a series of clusters suggests that the DPGMM is finding too many distinctions, rather than not learning to distinguish.

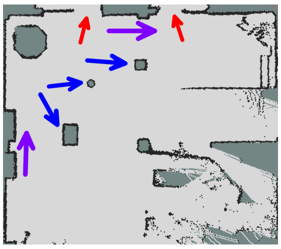
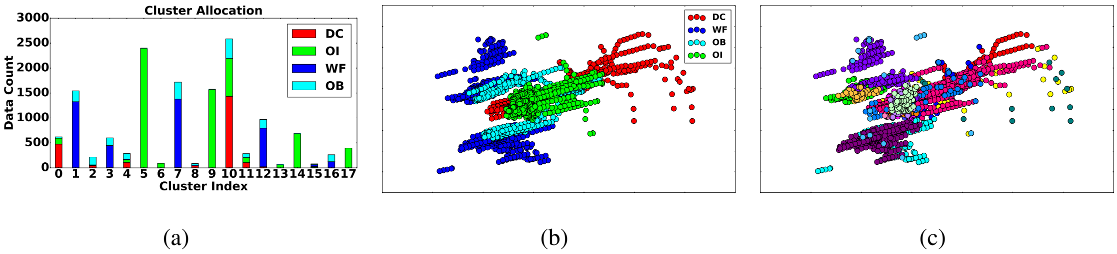


Figure 1: The map of the scenario used for evaluations with different test fractions, and the extracted semantic components, being denoted by arrows with different colors: doorways (red), objects (blue), and wall segments (violet).

Then, to quantitatively evaluate the performance of the proposed approach, the collected dataset was randomly split into test and training with test fractions varying as 0.25, 0.5 and 0.75 based on the tra-

[4] This choice is to make the SOGP classifier sparser than the SVM whose sparsity is denoted with the number of support vectors after training, which will be shown in the following evaluations.

[5] The trajectories used as the training data were transformed to the sequences of task features computed from the groundtruth semantic components, while possessing no labels for the task types.



(a) (b) (c)

Figure 2: The qualitative result of motion clustering: (a) The stacked histogram shows the discovered motion clusters and the feature data assigned to them, colored by the original task types, please note the scale on y-axis; (b) The feature data, mapped into 2D with their first two components and colored by the original task types; (c) The feature data, mapped into 2D with their first two components and colored by the discovered motion clusters.

jectory number. In each test phase, the training data were firstly clustered by DPGMM in the same way as above. Post clustering, the SOGP classifier and the SVM classified each way point along each test trajectory with the learned motion clusters. To determine the model parameters, for SVM, we did a grid-search on its parameters over reasonable sets, while for SOGP classifier, we obtained the (locally) optimised hyper-parameters by maximizing the evidence of full GP via gradient-descent. The test results within different fractions are displayed in table 1, including the numbers of the training samples, the learned clusters and the support vectors used by SVM, the ACR and the MICAD of the two classifiers, respectively. As can be viewed, the ACR of the SOGP classifier decreases obviously compared between the test fraction 0.25 and 0.5, but maintains stable between the fraction 0.5 and 0.75. The tightness measurements of the SOGP classifier keep stable across the test fractions. In general, the SOGP classifier yielded very good recognition results even being trained with much more clusters than the number of task types used for demonstration. In comparison with SVM, being confirmed by the paired T test for the measurements of CR and ICAD, the SOGP classifier performs considerably better than SVM in all test fractions, even with a sparser representation denoted by the capacity size of the SOGP classifier and the number of the support vectors used by SVM. This verified that our approach is able to correctly recognize the motion patterns it learns from the demonstrations.

4.3 Performance Evaluation with New Task Types

Apart from recognizing the learned motion patterns, we are also interested in whether the proposed approach is generalizable to correctly capture new task types not used for training. Hence we collected the test data from performing three new contextual task types: Wall Inspection, Robot Docking[6] and Gap Crossing for two times each with random initial poses respectively, in another cluttered indoor scenario whose map and the corresponding semantic components are illustrated in figure 3. The whole dataset sampled in the subsection 4.2, were provided for clustering (see figure 2) then training the SOGP classifier and the SE SVM in the same manner, and the datasets collected in this subsection were presented for inference. We computed the ACR and the MICAD values of the two classifiers over all test trajectories to compare their recognition performance, which are listed in table 2, together with the numbers of the training samples, the discovered motion clusters and the support vectors used by SVM. As confirmed by the paired T test for the measurements of CR and ICAD, our approach is able to recognize the new task types with considerably better performance on the evaluation metrics using the sparser SOGP classifier than using the SVM.

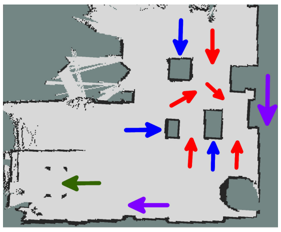


Figure 3: The map of the scenario used for evaluations with three new task types, and the extracted semantic components, being denoted by arrows with different colors: gaps (red), the docking target under the table (green), objects (blue), and wall segments (violet).

4.4 Introspection Evaluation with Distinctively Unknown Task Types

In real and long term applications, it is hardly possible to train the robot with all the needed task types

[6] The robot was to be docked into a table.

Table 1: The ACR and the MICAD comparison for the SOGP classifier and the SE SVM post clustering with varying test data fractions, along with the numbers of the training samples and the discovered clusters. The capacity size of the SOGP classifier and the number of support vectors used by SVM are also listed in each test fraction for comparison of the sparsity of the two classifiers.

Experiment	Test Fraction 0.25		Test Fraction 0.5		Test Fraction 0.75	
No. of Training Samples	10284		7718		3540	
No. of Motion Clusters	14		16		12	
Classifier **Sparsity**	**SOGP** c: 500	**SVM** sv: 1894	**SOGP** c: 500	**SVM** sv: 896	**SOGP** c: 500	**SVM** sv: 513
ACR	0.93	0.80	0.88	0.74	0.88	0.79
MICAD	0.88	1.11	0.87	1.16	0.87	1.32

Table 2: The ACR and the MICAD comparison for the SOGP and the SVM classifiers post clustering on test data collected from performing three new task types, along with the numbers of the training samples and the discovered clusters. The capacity size of the SOGP classifier and the number of support vectors used by SVM are also listed for comparison of the sparsity of the two classifiers.

No. of Training Samples	14471	
No. of Motion Clusters	18	
Classifier **Sparsity**	SOGP c: 500	SVM sv: 2723
ACR	0.82	0.74
MICAD	0.90	1.11

before deployment due to time and economic considerations on one hand. On the other hand, as shown in the previous subsection, the robot is supposed to utilize the motion clusters to generate autonomous motion commands, which assists the teleoperation by fusing them with the user inputs based on the probability/confidence of the task recognition (Gao et al., 2014), hence we would not expect that the system can provide appropriate motion assistance to a previously unseen motion pattern distinctive from the one used for training, for example when it is initially trained with Doorway Crossing and later applied to assist Object Inspection[7], even if the system correctly recognizes the corresponding semantic targets. Therefore, for robotic applications involving mission-critical decision making, such as mobile robot teleoperation where this report focuses, it is imperative to investigate a classifier's capability of uncertainty estimation when classifying the motion clusters for a query task feature, *i.e.* the *introspective* capability (Grimmett et al., 2013) of the classifier. To characterize the introspective capability of a classifier, we compute the normalized entropy value (Grimmett et al., 2013) for each query point based on its discrete probability distribution over the discovered motion clusters. For each way point along each test trajectory used in the following evaluations of this subsection, we computed the task feature of a way point from the groundtruth semantic target, and we queried the SOGP classifier and SVM with this task feature to obtain its discrete probability distribution over the learned motion clusters, to facilitate the computation of the normalized entropy value of this point, for ease of the introspection comparison of the two classifiers.

In this subsection, we used the dataset collected in the subsection 4.2, where we arbitrarily selected two task types for clustering then training the two classifiers: the SOGP classifier and the SE SVM in the same way as the subsection 4.2, and the datasets from the other two task types were used for inference, attempting to do the introspection evaluation with distinctive motion patterns. In order to mitigate any influences of the specific training and test data selected, we repeated such evaluation procedure across all possible task type combinations for training, resulting into six groups of the normalized entropy values. The mean and standard deviation normalized entropies of each of the six test groups are listed in table 3 respectively, together with the MICAD measurements of the two classifiers. As confirmed with the paired T test, the mean normalized entropies for the SOGP classifier are considerably higher than those of the SVM classifier, signifying that the former exhibited greater uncertainty in the judgement, indicating strongly the presence of potentially *un-modelled* motion patterns, which is also suggested by the high MICAD values (compared with those in the subsection 4.2) across all test iterations, while the latter was extremely confident in its classifications with lower values of the normalized entropy, even though the high MICAD values imply a potential inappropriate interpretation of the motion patterns. In practice, the robot can utilize this outstanding introspective capability of the SOGP classifier to *actively* query for an update of the demonstration data without manual labels to increase its knowledge regarding the uncertain motion patterns, which are potentially distinctive to those already absorbed in its knowledge base. This property is key to fulfill our vision of a life-long adaptive assistive robot. How to exploit such uncertainty estimation to interact with human (*e.g.* for further demonstration via dialogue) remains our future work.

[7] Performing Doorway Crossing means to drive the robot to simply approach the target doorway, while Object Inspection aims to not only approach the target object, but also move around it within certain distance while facing it.

Table 3: Mean and standard deviation normalized entropies from six iterations of training and testing, where the datasets from two task types were used for training, and the rest data were presented for inference. The total datasets are collected from performing four task types. The MICAD measurements of the two classifiers in each test iteration are also listed.

Test Task Types	**Classifier**	**Normalized Entropy** $\mu \pm$ std.err.	**MICAD**
OI and WF	SOGP	**0.703 ± 0.402**	1.30
	SVM	0.498 ± 0.351	1.52
OB and WF	SOGP	**0.900 ± 0.232**	1.44
	SVM	0.151 ± 0.290	2.08
OB and OI	SOGP	**0.796 ± 0.369**	1.52
	SVM	0.354 ± 0.215	1.56
DC and WF	SOGP	**0.550 ± 0.331**	1.00
	SVM	0.218 ± 0.286	1.05
DC and OB	SOGP	**0.535 ± 0.367**	1.01
	SVM	0.304 ± 0.332	1.12
DC and OI	SOGP	**0.857 ± 0.271**	1.41
	SVM	0.641 ± 0.263	1.56

5 CONCLUSION

This paper reported an unsupervised approach for learning and recognizing human motion patterns performing various contextual task types from unlabelled demonstrations, attempting to facilitate autonomy sharing to assist mobile robot teleoperation. The motion patterns were described with a set of intuitive, compact and salient task features. The DPGMM was employed to cluster the motion patterns based on the task feature data, where the number of potential motion components was inferred from the data itself instead of being manually specified *a priori* or estimated through model selection. Moreover, both overlaps and distinctions of the task execution patterns can be discovered through clustering, which is used as a knowledge base for interpreting the query patterns later on. Post clustering, the SOGP classifier was used to recognize which motion pattern the human operator executes during operation, taking advantage of its outstanding confidence estimation when making predictions and scalability to large datasets. Extensive evaluations were carried out in indoor scenarios with a holonomic mobile robot. The experimental results from the real data verified that, the proposed approach serves as a generic framework for representing and exploiting the knowledge of the human motion patterns performing various contextual task types without manual annotations, which is not only able to recognize the task types seen during training, but also generalizable to appropriately interpret the motion patters of task types not used for training, and more importantly, the proposed approach is capable of detecting unknown motion patterns distinctive from those used in the training set, due to the superior introspective capability of the SOGP classifier, hence provides a significant step towards a life-long adaptive assistive robot.

REFERENCES

Blei, D. M., Jordan, M. I., et al. (2006). Variational inference for dirichlet process mixtures. *Bayesian analysis*, 1(1):121–143.

Chang, C.-C. and Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27.

Csató, L. and Opper, M. (2002). Sparse on-line gaussian processes. *Neural computation*, 14(3):641–668.

Dragan, A. and Srinivasa, S. (2012). Formalizing assistive teleoperation. *R: SS*.

Fong, T., Thorpe, C., and Baur, C. (2001). Advanced interfaces for vehicle teleoperation: Collaborative control, sensor fusion displays, and remote driving tools. *Autonomous Robots*, pages 77–85.

Gao, M., Oberländer, J., Schamm, T., and Zöllner, J. M. (2014). Contextual Task-Aware Shared Autonomy for Assistive Mobile Robot Teleoperation. In *Intelligent Robots and Systems (IROS), 2014 IEEE/RSJ International Conference on*. IEEE.

Gao, M., Schamm, T., and Zöllner, J. M. (2016). Contextual Task Recognition to Assist Mobile Robot Teleoperation with Introspective Estimation using Gaussian Process. In *Autonomous Robot Systems and Competitions (ARSC), 2016 IEEE International Conference on*. IEEE.

Grimmett, H., Paul, R., Triebel, R., and Posner, I. (2013). Knowing when we don't know: Introspective classification for mission-critical decision making. In *Robotics and Automation (ICRA), 2013 IEEE International Conference on*, pages 4531–4538. IEEE.

Hauser, K. (2013). Recognition, prediction, and planning for assisted teleoperation of freeform tasks. *Autonomous Robots*, 35(4):241–254.

Hughes, M. C. and Sudderth, E. (2013). Memoized online variational inference for dirichlet process mixture models. In *Advances in Neural Information Processing Systems*, pages 1133–1141.

Okada, Y., Nagatani, K., Yoshida, K., Tadokoro, S., Yoshida, T., and Koyanagi, E. (2011). Shared autonomy system for tracked vehicles on rough terrain based on continuous three-dimensional terrain scanning. *Journal of Field Robotics*, 28(6):875–893.

Sa, I., Hrabar, S., and Corke, P. (2015). Inspection of pole-like structures using a visual-inertial aided vtol platform with shared autonomy. *Sensors*, 15(9):22003–22048.

Sheridan, T. B. (1992). *Telerobotics, automation and human supervisory control*. The MIT press.

Stefanov, N., Peer, A., and Buss, M. (2010). Online intention recognition in computer-assisted teleoperation systems. In *Haptics: Generating and Perceiving Tangible Sensations*, pages 233–239. Springer.