Learning Terrain Types with the Pitman-Yor Process Mixtures of Gaussians for a Legged Robot

Patrick Dallaire[1], Krzysztof Walas[2], Philippe Giguère[1] and Brahim Chaib-draa[1]

Abstract— **One of the major goals for mobile robots is to be able to traverse any kind of terrains. A possible way to achieve this goal is by the use of legged robots, as they have increased mobility. However, this would require them to be able to modify their gaits, based on the identification of the terrain that they are currently traversing. In this paper, we introduce a number of novel methods to address this issue of autonomous terrain classification and clustering, based on tactile data collected with a walking robot. The proposed learning methods are based on the Pitman-Yor process mixture of Gaussians, a Bayesian nonparametric prior, well-suited for density estimation. This model is initially used to learn the non-Gaussian distribution of the features produced from proprioceptive (force/torque) signals from the legs, registered during the interaction of one robot foot with a terrain. Then, we exploit its capacity on clustering and discovering structures in the data to identify terrains in the feature space. Experiments were conducted on a six-legged robot, thus demonstrating the applicability of the Pitman-Yor process mixture of Gaussians for terrain identification. In particular, we obtained a classification success rate of 82% and 51% accuracy, with our supervised learning and unsupervised learning approach respectively.**

I. INTRODUCTION

The general problem of estimating the physical properties of terrains, called terramechanics, is an important one from a point of view of mobile robotics, as it has a strong impact on locomotion performance and strategies. Indeed, knowledge about the terrain type and its properties such as stiffness, roughness or friction coefficient, would allow a mobile robot to adapt its control algorithm to changing traction conditions [1]. Thus, being able to identify the terrain type located in front of a robot would be a way to increase the effectiveness of navigation, so that a mobile platform can go efficiently through a specified area.

Of all possible terrain properties, this paper focuses on identifying the material type of terrains. This key problem of terrain identification in mobile robotics has been studied for at least 30 years. For instance, a theoretical background for revealing material properties through interactions with objects was described in [2] in the late 80's. Subsequent experimental work was presented in [3], where the authors used a robotic leg in their experiments.

Most approaches found in the literature use the following steps to identify terrains: i) a probe (wheel or leg of the robot) physically interacts with the terrain; ii) acceleration, force or torque signals are recorded; iii) signal dimensionality reduction is performed through feature extraction; iv) supervised learning is applied, often in the form of discriminative probabilistic models. The last step is more demanding, as it requires a human to label the training data set. To alleviate this problem of hand-labelling, more and more attention is nowadays paid in the machine learning community to unsupervised learning algorithms, which allows to perform a task with limited human intervention. Unsupervised learning is slowly proliferating in the robotics community [4], [5], [6], but its influence is still limited. Additionally, generative models are increasingly considered given their powerfulness, as they are capable of predicting data from their joint probability distribution [7], [8], [9], [10], [11]. This is in opposition to conditional probability from discriminative models.

In this paper, we present solutions to both the terrain classification (supervised) and terrain clustering (unsupervised) tasks. To solve these two learning problems, we adopt a Bayesian nonparametric approach based on the Pitman-Yor process mixture of Gaussians (PYPMoG), which is a prior distribution on infinite mixture of Gaussians. To this end, we first discuss in Section II previous work on terrain identification and learning methods. Section III provides the details related to the robot and the data acquisition. Thereafter, the PYPMoG is introduced in Section IV with the aim of conveying the general intuition behind infinite mixture modeling. Experiments with *Messor*, our walking robot, are presented in Section V. We finally discuss the results and conclude in Section VI and VII respectively.

II. RELATED WORK

A. Terrain identification

Tactile perception is a broad field within robotic research. In this paper, we focus on terrain identification for mobile robots, which we consider as a sub-field of tactile sensing. For wheeled robots, terrain sensing is often performed by measuring the vibrations of the chassis of the vehicle with an Inertial Measurements Unit (IMU), and then performing classification on the acceleration signals [12], [13]. Similarly using an IMU, but mounted on a simple tactile probe, the authors of [14] were able to perform terrain classification, with the advantage of by-passing the low-pass filtering effect of the weight of the chassis and the suspension system. An example of combining vibrations measurements and visual information to terrain classification was described in [15].

For legged robots, several studies of terrain classification algorithms using measurements from a single robotic leg

[1]The authors are with the department of Computer Science and Software Engineering Universit Laval, Canada {firstname.lastname}@ift.ulaval.ca

[2]Krzysztof Walas is with the Faculty of Electrical Engineering, Poznan University of Technology, 60-965 Poznan, Poland krzysztof.walas@put.poznan.pl; * His work was partially supported by MSHE grant 04/45/DSMK/0141.

have been done. For instance, a vibration-based approach applied to natural terrains is presented in [16]. In recent work, pressure-sensitive skin mounted on a robot foot was used as a sensing device for terrain classification. The use of a pressure image provides, according to authors, results which are independent of the dynamics [17]. Terrain identification for a *RHex*-like robot named *Aqua* is described in [18], where they employed a combination of actuator cues (motor currents) and accelerometer readings to perform terrain identification. This research was further extended to other terrain types [19], alongside a walking-speed adaptation algorithm. Other researchers have looked at pushing the number of sensed signals to the extreme (18) [20]. For this work, results were obtained for 4 terrain types with 5 distinct gaits. There is also a previous work on this topic done by some of our authors [1], although relying solely on supervised learning techniques. Finally, some recent work presented analytical model describing the difficult problem of foot-ground interaction [21].

B. Probabilistic models

The aforementioned works in the previous subsection II-A were all based on applying standard discriminative machine learning techniques to perform the classification (voting linear classifiers, neural networks, SVM, etc). In this work, we advocate the use of fully probabilistic models, as they have multiple advantages over discriminative models, as related in subsection V-A and V-B. In [7], a generative model allowed a system operating in an off-road autonomous navigation scenario to infer the supporting ground surface, even when it was hidden under dense vegetation. To this end, they predict the missing data. Gaussian Processes, another generative approach, were used for large scale mapping [8] and planetary rover navigation [10]. It could also learn human behaviours and its relation to objects in the environment. This way, given an object instance, it would "hallucinates" humans employing it for better object categorisation [9]. The availability of these full models can also help us in establishing a better confidence level of the decision made by the robot. In some sense, it gives the autonomous system the capability of knowing when it does not know [11].

From a machine learning perspective, Gaussian Mixture Models (GMMs) are well-suited for density estimation tasks. Under some assumptions, they can also be used to perform clustering efficiently, enabling the unsupervised discovery of terrain types. In the context of terrain identification, a robot may interact with an unknown and potentially unbounded set of surfaces. However, standard GMM learning methods such as the Expectation-Maximization (EM) algorithm [22] require the exact number of clusters to be provide beforehand, and are therefore unsuitable for our problem. To eliminate this constraint, the EM algorithm can be combined with Minimum Description Length (MDL) [23], resulting in an agglomerative clustering strategy estimating the number of components which best fit the data. While the EM algorithm is good at learning a single maximum a posteriori (MAP) model efficiently, it lacks estimating the posterior uncertainty, due to its optimization nature.

Fig. 1. The *Messor* robot, used to collect the data sets in the experiments.

Another way to learn GMMs is to use a Bayesian framework and rely on Markov chain Monte Carlo (MCMC) to simulate the joint posterior distribution. When using an appropriate prior distribution, the joint posterior distribution provides information not only on the GMM's parameters, but also on the dimensionality of the GMM. For instance, in [24], Zhang *et al.* used a uniform prior distribution on the bounded number of components, along with a Dirichlet prior on the mixture distribution. Then, they learned the dimensionality of the model with a reversible-jump MCMC algorithm. In [25], Rasmussen derived a prior as the limiting case of a finite prior on GMMs. The resulting Bayesian nonparametric prior is in fact a Dirichlet process on infinite dimensional GMMs capable of learning a finite number of *active* components in infinite mixture models.

The extension of GMMs with *nonparametric* learning techniques has recently gained popularity for robotic applications. One of such approach, where the model of the manipulator is learned from raw RGB data, is presented in [5]. An application, which is closer to our work, is described in [4]. The foothold selection using clustering methods and clustering validation techniques for estimating the number of classes is investigated. In our earlier work the unsupervised learning of surface types in the highly controlled environment was performed [6].

C. Contributions

The main contribution of our work is on the application of the Pitman-Yor process mixture of Gaussians to autonomously learn terrain types in the form of clusters. The advantage of such unsupervised learning is allowing the robot to freely navigate its environment and self-improve over time.

Our second contribution is a supervised learning method based on learning the non-Gaussian features distribution associated with individual terrain types. This methods achieves state-of-the-art performances and has the advantage of being a probabilistic generative model.

III. MATERIALS

A. Walking Robot

The *Messor* robot, shown in Fig. 1, is a six-legged machine which weighs approximately 4.3 kg, depending on the sensor payload mounted on its back. The stretched leg of the

robot has a length of 0.445 m. Starting from the trunk, the segments have the following dimensions: 0.055 m, 0.16 m and 0.23 m. The distance between mounting points of hind and fore leg is equal to 0.36 m, and between middle-right and middle-left leg is equal to 0.26 m. The legs of the robot are powered using Hitec HSR-5990TG robotic servo-motors. A full description of the robot can be found in [26].

The robot is equipped with a rich variety of sensors: video camera, depth camera, IMU and a Force/Torque (F/T) sensor. For the research described in this paper, the latter is the most important measuring device. The F/T sensor mounted at the tip of the robotic leg is the *ATI Mini-45* (calibrated by the manufacturer). In the experiments the robot was walking straight with the average speed V = 2 cm/s, using wave gait. The measurements were acquired for static and dynamic states with sensor sampling frequency set to 200 Hz. The data consists of 3 force vector components ($F_x,\ F_y,\ F_z$) and 3 torque vector components ($T_x,\ T_y,\ T_z$).

Fig. 2 shows, for two sample terrains, the force and torque signals registered during the initial 5 seconds of walking. In a) and b), we have a concrete floor and c) and d) is a sand surface. As it can be remarked, when the robot walks on the concrete floor, the force signal F_z is regular with constant amplitudes for each step. The sand terrain signals exhibit different features. We can immediately see that torque amplitudes are increasing over time in d), and that some drift affects c) as the signals slowly move towards negative values. We attribute this to the fact that the robot had problems pulling its feet out of the sandy ground. Additionally, we observe a difference in the force signal amplitude where c) is smaller than a), which is especially true for F_z. We ascribe this disparity to the soft nature of sand as opposed to the firmness of concrete. We also consider that the torque signals of concrete and sand are notably different as well. The previous observation that concrete and sand can be distinguished based on their respective signals seems reasonable. For this reason, we assume that it should extend to multiple type of terrains.

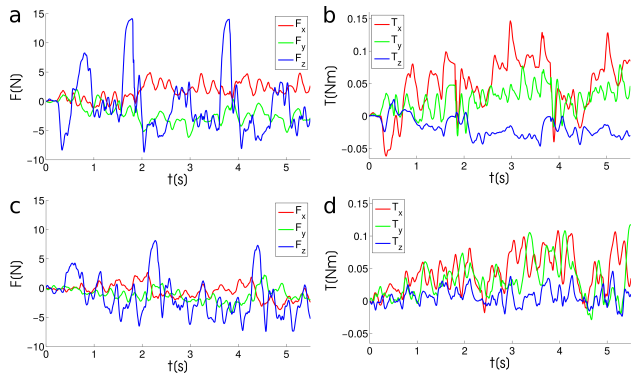

Fig. 2. Force/Torque signals registered with the foot-mounted sensor. Subfigures are: a) forces from concrete floor; b) torques from concrete floor; c) forces from sand; d) torques from sand. Each color denotes a signal axis: x-axis (red), y-axis (green) and z-axis (blue).

We sampled data from a set of 12 terrains (shown in Fig. 3), each having different mechanical properties. There are four types of powdery materials of different granularity: sand, grit, pebbles and rocks. Our data set contains two types of rubber, with different bounciness. Additionally, the set comprises PVC tiles, ceramic tiles, carpet tiles, artificial grass, wooden boards and concrete floor.

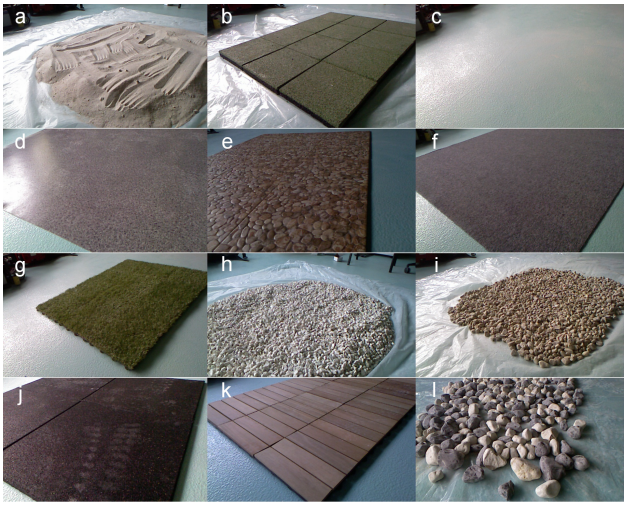

Fig. 3. Terrain samples used in the experiments: a) sand; b) green rubber; c) concrete floor; d) PVC tiles; e) ceramic tiles; f) carpet tiles; g) artificial grass, h) grit; i) pebbles; j) black rubber; k) wooden boards and l) rocks.

IV. LEARNING INFINITE GAUSSIAN MIXTURES

In this section, we present a Bayesian nonparametric prior on Gaussian mixtures based on the Pitman-Yor process, a two-parameter generalization of the Dirichlet process [27]. The Pitman-Yor process has an extra parameter governing its tail behavior, making it useful when modeling data with either exponential or power-law tails [28].

A. Pitman-Yor Process Mixture Models

The Pitman-Yor process $\mathrm{PY}(d, \alpha, G_0)$ is a probability distribution over infinite dimensional discrete probability distributions. It is parametrized by a discount parameter $0 \leq d < 1$, a concentration parameter $\alpha > -d$ and a base probability distribution G_0 on the space where the learning takes place. As the discount parameter increases, the Pitman-Yor process produces discrete distributions with heavier tails. The stick-breaking construction of Ishwaran and James [29] is particularly useful to understand the support of this process. First, a draw G from a Pitman-Yor process has the following form:

$$G(\cdot) = \sum_{k=1}^{\infty} \pi_k \delta_{\theta_k^*}(\cdot) \tag{1}$$

where $\theta_k^* \sim G_0$ and $\delta_\theta(\cdot)$ denotes a unit mass located at parameter θ. When the base distribution G_0 is continuous, equation (1) assigns positive probability to a countably infinite subset of parameters, which results in a discrete G. To make sure G is also a probability distribution, the

sequence $(\pi_k)_{k=1}^{\infty}$ must sum to 1. This constraint is enforced by recursively breaking a unit-length stick as follow:

$$\begin{aligned} v_k &\sim \text{Beta}(1-d, \alpha+kd) \\ \pi_k &= v_k \prod_{j=1}^{k-1}(1-v_j) \quad \text{for } k = 1, \ldots, \infty \end{aligned} \tag{2}$$

where v_k is the portion to cut from the remainder of the stick and π_k is the length of this portion. With (2), one can have a better grasp of parameter d's impact on distribution G.

At this point, it probably becomes much clearer how Pitman-Yor processes can be applied to mixture modeling. The actual definition of the Pitman-Yor process mixture model (PYPMM) is the following:

$$\begin{aligned} G \mid d, \alpha, G_0 &\sim \text{PY}(d, \alpha, G_0) \\ \theta_i \mid G &\sim G \\ \mathbf{x}_i \mid \theta_i &\sim F(\theta_i) \end{aligned} \tag{3}$$

where F is the continuous probability density function of your choice. When learning Gaussian mixtures, F is the multivariate Gaussian density, θ contains both the mean vector and covariance matrix and component weights $\boldsymbol{\pi}$ are included within distribution G as shown in equation (1). Notice that θ_i is the label of $\mathbf{x}_i$ that all data points generated by component k will share the same parameters θ_k^*.

The explicit representation of random distribution G requires truncation due to its infinite nature. On the other hand, marginalizing out G allows us to avoid the truncation step and makes more obvious both the clustering and discreteness properties of Pitman-Yor processes. In that case, one can directly draw a random sequence of labels from the following conditional probability:

$$\theta_i \mid \theta_1, \ldots, \theta_{i-1} \sim \frac{\alpha + Kd}{\alpha + i - 1} G_0 + \sum_{k=1}^{K} \frac{n_k - d}{\alpha + i - 1} \delta_{\theta_k^*} \tag{4}$$

where n_k is the number of times θ_k^* is observed among $\theta_1, \ldots, \theta_{i-1}$ and K is the total number unique labels so far. Equation (4) comprises a discrete measure (right term) over the K previously observed components, meaning there is a positive probability for an existing θ_k^* to be resampled. As more observations become available, the discrete measure acquires more mass and the novelty probability (left term) decreases accordingly. Thus, it results in a suitable rich-gets-richer clustering behavior.

There exists multiple methods to perform posterior inference with PYPMMs. For instance, equation (4) gives rise to a Gibbs sampler on cluster membership while equation (1) can serve the same purpose. However, for the sake of brevity, we refer interested readers to three key references for details on inference methods [29], [30], [31].

B. Pitman-Yor Process Mixture of Gaussians

In this section, we complete the PYPMM with the necessary specifications leading to the Pitman-Yor process mixture of Gaussians (PYPMoG) which is used to learn infinite GMMs in our experiments. To get to a fully probabilistic model we still have to specify hyperpriors on the hyperparameters, define density function F and determine base distribution G_0.

For the probability density function F, we use the following multivariate Gaussian representation:

$$\mathbf{x}_i \mid \theta_i \sim \mathcal{N}(\boldsymbol{\mu}_i, S_i^{-1}) \tag{5}$$

where $\boldsymbol{\mu}$ is the mean and S is the precision matrix. Again, if $\mathbf{x}_i$ belongs to the kth cluster, then $\theta_i = \theta_k^*$.

The base distribution G_0 is a prior indicating the kind of Gaussians we are likely to get as components. Inspired from the conditionally conjugate model proposed in [32], we define distribution G_0 on θ as:

$$\boldsymbol{\mu}_k \sim \mathcal{N}(\mathbf{0}, I) \tag{6}$$

$$S_k \mid \beta, \Sigma \sim \mathcal{W}(\beta, (\beta\Sigma)^{-1}) \tag{7}$$

where $\mathcal{W}$ denotes the Wishart distribution with β degrees of freedom and scale matrix $(\beta\Sigma)^{-1}$. When assuming properly scaled training data with zero mean and unit covariance, (6) is a reasonable prior as it reflects the dispersion of the components center in the $\mathbf{x}$-space. However, we have no clue regarding the shape of the components. Equation (7) is a distribution over the precision matrices where Σ is the expected covariance matrix and β acts on its variance.

For robustness, we adopt a hierarchical structure and specify vague hyperpriors on hyperparameters β and Σ, meaning we are learning G_0. The following prior distribution reflects our uncertainty concerning the unknown distribution (7) over cluster covariances:

$$\Sigma \sim \mathcal{W}(D, \frac{I}{D}), \qquad (\frac{1}{\beta - D + 1}) \sim \mathcal{G}(1, \frac{1}{D}) \tag{8}$$

where D is the number of dimensions of $\mathbf{x}$. Notice that this prior makes it increasingly harder to expect smaller and smaller Gaussians during the inference, thus favoring larger covariance for precision matrices S.

Not forgetting, we also specify the following prior distributions on α and d.

$$\alpha^{-1} \sim \mathcal{G}(1/2, 1/2), \qquad d \sim \mathcal{U}(0, 1) \tag{9}$$

corresponding to an inverse-gamma distribution for α and a uniform distribution for d. That finally completes the hyperprior on $\text{PY}(d, \alpha, G_0)$ that we used to perform density estimation and clustering in the experiments.

V. EXPERIMENTS

The aim of our experiments is to learn generative models of the terrain features and use them to make various predictions. We consider both the classification problem and clustering problem, and evaluate the PYPMoG performances on these tasks.

In section III, we described our walking robot, its sensing capabilities and provided a list of the 12 terrains to learn. We gathered a number of foot touch-down samples using the F/T sensor (3 force and 3 torque signals) on each terrain, ranging from 36 to 85 traversals per terrain, for a total of

652 6-dimensional time series. We consequently have a non-uniform representation of the terrains in our data, making it more challenging than if we had a balanced number of samples per terrain. Next, we extract features from time series. The selected features, taken from [14] (and themselves adapted from [33]) are common statistics including *i)* variance, *ii)* skewness, *iii)* kurtosis, *iv)* fifth moment, *v)* sum of variation over time, *vi)* number of times 20 uniform separations are crossed, which is equivalent to discretizing the signal and counting the number of state transitions, and *vii)* sum of higher half of the amplitude spectrum, which is the area under the FFT curve from 200 Hz to 400 Hz. To further reduce the dimensionality, we thereafter performed a principal component analysis (PCA) on the 42-dimensional feature data, lowering the number of dimensions to 6 (95% of the variance). This last transformation helps estimating Gaussian components with smaller amounts of data, which in turns should increase the accuracy of the mixture model.

The PYPMoG learning is done by simulating its posterior distribution over infinite mixture of Gaussians. To do so, we rely on Markov chain Monte-Carlo (MCMC) methods combining Gibbs sampling and Metropolis-Hasting steps. When a unique model is required, we simply use the maximum a posteriori (MAP) estimate of a Markov chain without involving any form of optimization.

A. Terrain classification with the PYPMoG

In a context where the terrain labels are available, we can adopt a supervised learning approach to tackle our problem. Thus, we used the PYPMoG for training multiple models in a per-terrain fashion. Such approach exploits the density estimation capacity of the infinite GMM by actually learning $p(\mathbf{x}|y)$, which is the conditional joint probability of the features $\mathbf{x}$ for a specific terrain y. We assume a uniform distribution on $p(y)$ to obtain a fully probabilistic model $p(\mathbf{x}, y)$, but this one could have been learned from the terrain frequencies with a Dirichlet prior.

For this experiment, we adopted a 7-fold cross-validation technique for performance assessment. Each fold was divided into a test set containing 5 samples per terrain and a training set having all the remaining data, resulting in 592 training examples and 60 testing examples per fold. Using the training data along with their labels, we trained 12 independent PYPMoG models, i.e. one per terrain. The learning procedure was carried out by simulating the posterior for 50k iterations and the MAP model was kept for terrain prediction purpose.

Once we have an estimated probability density function (PDF) associated with all surfaces, we proceed to the testing phase. To classify a datum, we individually compute the probability that this datum was generated from each terrain PDF. This is an easy step, since all PDFs are simple Gaussian mixtures. Finally, we select the most likely terrain as our prediction. When a Dirichlet distribution is available on the probability to observe a terrain $p(y)$, then the maximum likelihood would become a maximum a posteriori.

Overall, the cross-validation procedure required 420 predictions (7-fold $\times$ 60 test data). The confusion matrix of the learned model is presented in Fig. 4 and Table I reports the mean accuracy over the 7 folds. For comparison purpose, we evaluated the k-nearest neighbors algorithm and Expectation-Maximization + Minimum Description Length on the same problem. For the k-nearest neighbors, the highest average success rate was achieved with $k = 5$.

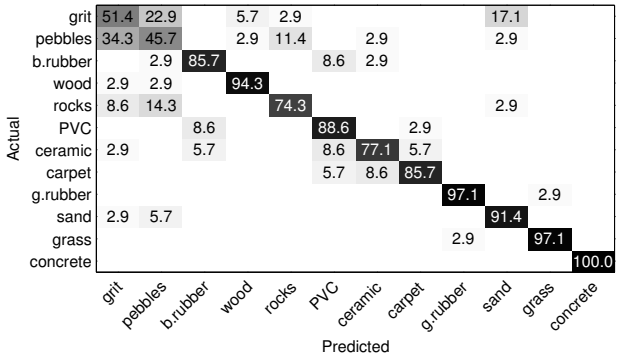

Fig. 4. Confusion matrix of the maximum a posteriori PYPMoG.

TABLE I
AVERAGE TERRAIN CLASSIFICATION ACCURACY

PYPMoG	**EM+MDL**	**k-NN**
82.38%	82.86%	78.39%

The average number of Gaussians per terrain indicates the amount of structure learned in the data. In the case of PYPMoG, the model used 2.08 Gaussians per terrain on average while EM+MDL used 1.40 Gaussians.

In Table I, we can see that EM+MDL and PYPMoG performed equally well, even though PYPMoG found slightly more structure than EM+MDL. This result might be explained by the small number of data per terrain, making it harder to find good structure in the data. However, the PYPMoG has the advantage to learn infinite GMMs via posterior inference, providing a set of candidate models instead of a unique optimized GMM. This set of posterior GMMs can be used to estimate the uncertainty on predicted labels and decisions can be taken accordingly.

B. Terrain clustering with the PYPMoG

As stated earlier, we are looking for an algorithm capable of distinguishing different terrains with little human intervention. In that context, labels are not available, but we still have to find the proper groups related to terrains among the data. Here, we use all 652 examples for training and ask the PYPMoG to find clusters under the hypothesis that terrain features are Gaussianly distributed.

For this first clustering experiment, we ran 20 independent Markov chains of 50k iterations to simulate the posterior on the PYPMoG (Fig. 5). After a burn-in period of 25k iterations, we kept the remaining 25k models as posterior samples for estimating the distribution over the number of clusters and computing what we call the *pairwise correct classification* metric. The later uses hidden labels to calculate

a dissimilarity measure with the true clustering as follow: an error is made when two data points are assigned to the same cluster and should not be, or when they are in different clusters and should be in the same one. The resulting clustering accuracy is then 1 minus the error ratio.

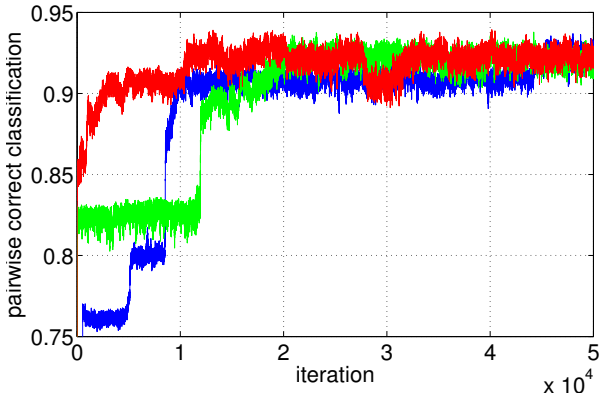

Fig. 5. Three non-overlapping Markov Chain Monte Carlo simulations on clustering accuracy selected for visual assessment.

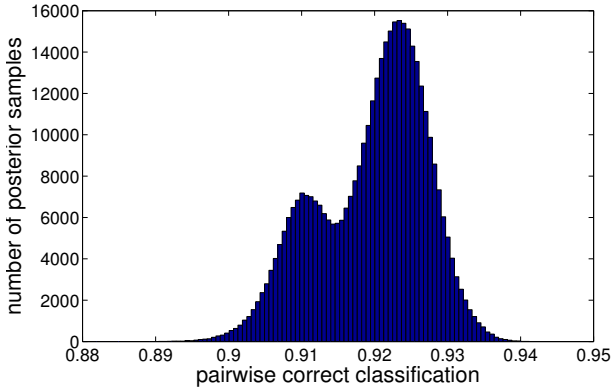

Fig. 6. Posterior distribution on clustering accuracy.

Fig. 6 depicts the posterior distribution on pairwise correct classification, with an empirical mean of 92.01% and standard-deviation of 0.72%. Regarding the number of terrain types found by the PYPMoG, we obtained an empirical mean of 34.79 and a standard-deviation of 4.41 for the posterior distribution shown in Fig. 7.

The fact that our pairwise correct classification measures 2 types of errors makes it difficult to interpret the results. We therefore designed a second clustering experiment to measure the predictive performances of the PYPMoG. To do so, we evaluate the scenario of a human manually labeling the training data once the clustering is completed. Then, predictions are made by computing the frequency of every label in a cluster to obtain a probability vector over potential terrains. This approach is similar to a majority vote. However, we do not know what cluster the datum to label belongs to. To solve this, we again measure the likelihood of each cluster given the datum and normalize the likelihoods, giving us a probability vector over clusters. Combining these quantities, we obtain a prediction matrix of size (number of

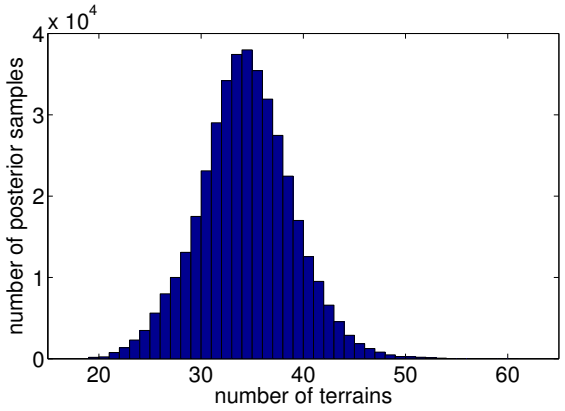

Fig. 7. Posterior distribution on identified terrains.

clusters)×(number of terrains), which we then sum along one dimension to obtain the probability over terrain labels.

For this experiment, we used a leave-one-out cross-validation technique. The PYPMoG learning is done by running a 50k iterations Markov chain on the 651 remaining unlabeled examples. After a burn-in of 25k iterations, we predicted the label of the left-out datum at each iteration and estimated the expected posterior accuracy by the empirical mean. The results, presented in Table II, are based on 40 independent experiments using a single randomly selected data (without replacement) for cross-validation, where the expected posterior accuracy is averaged over the 40 runs.

TABLE II
LEAVE-ONE-OUT AVERAGED PREDICTION ACCURACY

PYPMoG	**EM+MDL**	**EM(12)**	k**-means(12)**
51.22%	27.05%	37.66%	43.95%

According to Table II, the PYPMoG outperforms other approaches with an appreciable margin. The EM learning algorithm performs poorly when it relies on the MDL to determine the number of cluster. The 2 other methods required the true number of cluster to be supplied beforehand, and even with the additional information, they were unable to match the PYPMoG accuracy.

VI. DISCUSSION

The main focus of our work was the application of the Pitman-Yor process mixture of Gaussians, which is a probability distribution over infinite Gaussian Mixture Models. The word infinite is the most important here, as it points out the fact the model really comprises infinitely many components, but only a finite number of them are active.

We argue that in the real world, there really exists infinitely many terrains, but not enough time to observe them all. Bayesian nonparametric models share this common property that the prior probability distribution changes as more data becomes available. In other words, the prior depends on the number of data, as opposed to the fixed classic Bayesian prior. The impact on learning is that for small number of observations, the prior is less inclined to add clusters. On

the other hand, with large number of observations, the prior will force the introduction of new clusters.

Such behavior is desirable in the context of terrain clustering. As an example, if a robot repeatedly walks on sand, the prior will eventually encourage the splitting of the sand cluster. This behavior may lead to the discovery of different type of sand. As a results, Bayesian nonparametric priors are well-suited for the task of terrain clustering, if such behavior is desirable. In the opposite case, one could simply replace the PYPMoG prior by a Bayesian prior on GMMs and use the reversible-jump MCMC algorithm of [24].

VII. CONCLUSIONS

In this work, we presented two generic and novel learning methods addressing the problem of autonomous terrain type learning. Both methods are based on the Pitman-Yor process mixture of Gaussians (PYPMoG), a Bayesian nonparametric prior on infinite Gaussian mixtures.

The first method produced a classifier by training multiple PYPMoGs, one for each terrain type. The approach aimed to learn the generative process of the different terrain type and use the learned probabilistic model to perform predictions. One of the advantages of this method is its capacity of modeling non-Gaussian distribution on the terrain features. We then presented a second method using the PYPMoG for its clustering capability. The main advantage of this approach is the great flexibility provided by the infinite dimensional model introduced through the use of a PYPMoG prior.

We demonstrated the applicability of our method through experiments on real world data collected by a six-legged robot, on 12 different terrain types. Our results showed that our methods achieved state-of-the-art performances on terrain classification, both in the supervised and unsupervised cases. The latter, combined with the flexibility of employing infinite mixture models, paves the way to true long-term learning of terrains, over the lifetime of a robot.

REFERENCES

[1] K. Walas, "Terrain Classification and Negotiation with a Walking Robot," *J. of Intelligent & Robotic Systems*, vol. 78, no. 3-4, pp. 401–423, 2015.

[2] A. Pentland and J. Williams, "Perception of non-rigid motion: Inference of shape, material and force," in *Proceedings of the 11th IJCAI-Volume 2*. Morgan Kaufmann Publishers Inc., 1989, pp. 1565–1570.

[3] R. Hoffman and E. Krotkov, "Perception of rugged terrain for a walking robot: true confessions and new directions," in *'Intelligence for Mechanical Systems, Proceedings IROS'91.*, 1991, pp. 1505–1510.

[4] M. Hoepflinger, M. Hutter, C. Gehring, M. Bloesch, R. Siegwart, *et al.*, "Unsupervised identification and prediction of foothold robustness," in *Robotics and Automation (ICRA), 2013*. IEEE, 2013, pp. 3293–3298.

[5] B. Boots, A. Byravan, and D. Fox, "Learning predictive models of a depth camera & manipulator from raw execution traces," in *Robotics and Automation (ICRA), 2014*, May 2014, pp. 4021–4028.

[6] P. Dallaire, P. Giguère, D. Émond, and B. Chaib-Draa, "Autonomous tactile perception: a combined improved sensing and bayesian nonparametric approach," *Robotics and Autonomous Systems*, vol. 62, no. 4, pp. 422–435, 2014.

[7] C. Wellington, A. Courville, and A. T. Stentz, "A generative model of terrain for autonomous navigation in vegetation," *The Int. J. of Robotics Research*, vol. 25, no. 12, pp. 1287–1304, 2006.

[8] S. Vasudevan, F. Ramos, E. Nettleton, and H. Durrant-Whyte, "Gaussian process modeling of large-scale terrain," *J. of Field Robotics*, vol. 26, no. 10, p. 812840, 2009.

[9] Y. Jiang and A. Saxena, "Infinite latent conditional random fields for modeling environments through humans," in *Proceedings of Robotics: Science and Systems*, Berlin, Germany, June 2013.

[10] K. Ho, T. Peynot, and S. Sukkarieh, "Traversability estimation for a planetary rover via experimental kernel learning in a gaussian process framework," in *(ICRA)*, May 2013, pp. 3475–3482.

[11] H. Grimmett, R. Paul, R. Triebel, and I. Posner, "Knowing when we don't know: Introspective classification for mission-critical decision making," in *(ICRA)*, May 2013, pp. 4531–4538.

[12] C. Brooks and K. Iagnemma, "Vibration-based terrain classification for planetary exploration rovers," *Robotics, IEEE Trans. on*, vol. 21, no. 6, pp. 1185–1191, dec. 2005.

[13] L. Ojeda, J. Borenstein, G. Witus, and R. Karlsen, "Terrain characterization and classification with a mobile robot," *J. of Field Robotics*, vol. 23, no. 2, pp. 103–122, 2006.

[14] P. Giguere and G. Dudek, "A Simple Tactile Probe for Surface Identification by Mobile Robots," *Robotics, IEEE Trans. on*, vol. 27, no. 3, pp. 534–544, june 2011.

[15] C. Weiss, H. Tamimi, and A. Zell, "A combination of vision- and vibration-based terrain classification," in *Intel. Robots and Systems (IROS), IEEE/RSJ Int. Conf. on*, Sept 2008, pp. 2204–2209.

[16] M. Hoepflinger, C. Remy, M. Hutter, S. Haag, and R. Siegwart, "Haptic Terrain Classification on Natural Terrains for Legged Robots," in *(CLAWAR)*. World Scientific, September 2010, pp. 785 – 792.

[17] J. J. Shill, E. G. Collins, E. Coyle, and J. Clark, "Terrain identification on a one-legged hopping robot using high-resolution pressure images," in *(ICRA)*. IEEE, 2014, pp. 4723–4728.

[18] P. Giguère, G. Dudek, S. Saunderson, and C. Prahacs, "Environment Identification for a Running Robot Using Inertial and Actuator Cues." in *Robotics: Science and Systems*. The MIT Press, 2006.

[19] S. Manjanna, G. Dudek, and P. Giguere, "Using Gait Change for Terrain Sensing by Robots," in *Computer and Robot Vision (CRV), 2013 Int. Conf. on*, May 2013, pp. 16–22.

[20] M. Hoffmann, K. Stepanova, and M. Reinstein, "The effect of motor action and different sensory modalities on terrain classification in a quadruped robot running with multiple gaits," *Robotics and Autonomous Systems*, vol. 62, no. 12, p. 17901798, 2014.

[21] L. Ding, H. Gao, Z. Deng, J. Song, Y. Liu, G. Liu, and K. Iagnemma, "Footterrain interaction mechanics for legged robots: Modeling and experimental validation," *The Int. J. of Robotics Research*, vol. 32, no. 13, pp. 1585–1606, 2013.

[22] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *J. OF THE ROYAL STATISTICAL SOCIETY, SERIES B*, vol. 39, no. 1, pp. 1–38, 1977.

[23] J. Rissanen, "A universal prior for integers and estimation by minimum description length," *The Annals of statistics*, pp. 416–431, 1983.

[24] Z. Zhang, K. L. Chan, Y. Wu, and C. Chen, "Learning a multivariate gaussian mixture model with the reversible jump mcmc algorithm," *Statistics and Computing*, vol. 14, no. 4, pp. 343–355, 2004.

[25] C. E. Rasmussen, "The infinite gaussian mixture model." in *NIPS*, vol. 12, 1999, pp. 554–560.

[26] K. Walas and D. Belter, "Messor – Versatile Walking Robot for Search and Rescue Missions," *J. of Automation, Mobile Robotics & Intelligent Systems*, vol. 5, no. 2, pp. 28 – 34, 2011.

[27] H. Ishwaran and L. F. James, "Generalized weighted chinese restaurant processes for species sampling mixture models," *Statistica Sinica*, vol. 13, no. 4, pp. 1211–1236, 2003.

[28] J. Pitman and M. Yor, "The two-parameter poisson-dirichlet distribution derived from a stable subordinator," *The Annals of Probability*, pp. 855–900, 1997.

[29] H. Ishwaran and L. James, "Gibbs sampling methods for stick-breaking priors," *J. of the American Statistical Association*, vol. 96, no. 453, pp. 161–173, 2001.

[30] A. Dubey, S. A. Williamson, and E. P. Xing, "Parallel markov chain monte carlo for pitman-yor mixture models," in *(UAI)*, 2014.

[31] M. D. Fall and É. Barat, "Gibbs sampling methods for pitman-yor mixture models," 2012.

[32] D. Görür, "Nonparametric Bayesian Discrete Latent Variable Models for Unsupervised Learning," Ph.D. dissertation, Max Planck Inst. for Biological Cybernetics, 2007.

[33] C. Weiss, H. Frohlich, and A. Zell, "Vibration-based Terrain Classification Using Support Vector Machines," in *Intel. Robots and Systems, 2006 IEEE/RSJ Int. Conf.*, oct. 2006, pp. 4429–4434.