

“© 2021 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.”

# Gaussian Process Gradient Maps for Loop-Closure Detection in Unstructured Planetary Environments

Cedric Le Gentil<sup>1†</sup>, Mallikarjuna Vayugundla<sup>2†</sup>, Riccardo Giubilato<sup>2</sup>, Wolfgang Stürzl<sup>2</sup>,  
Teresa Vidal-Calleja<sup>1</sup>, Rudolph Triebel<sup>2,3</sup>

**Abstract**—The ability to recognize previously mapped locations is an essential feature for autonomous systems. Unstructured planetary-like environments pose a major challenge to these systems due to the similarity of the terrain. As a result, the ambiguity of the visual appearance makes state-of-the-art visual place recognition approaches less effective than in urban or man-made environments. This paper presents a method to solve the loop closure problem using only spatial information. The key idea is to use a novel continuous and probabilistic representations of terrain elevation maps. Given 3D point clouds of the environment, the proposed approach exploits Gaussian Process (GP) regression with linear operators to generate continuous gradient maps of the terrain elevation information. Traditional image registration techniques are then used to search for potential matches. Loop closures are verified by leveraging both the spatial characteristic of the elevation maps ( $SE(2)$  registration) and the probabilistic nature of the GP representation. A submap-based localization and mapping framework is used to demonstrate the validity of the proposed approach. The performance of this pipeline is evaluated and benchmarked using real data from a rover that is equipped with a stereo camera and navigates in challenging, unstructured planetary-like environments in Morocco and on Mt. Etna.

**Index Terms**—Localization; Space Robotics and Automation; Multi-Modal Perception; Visual-Based Navigation

## I. INTRODUCTION

Autonomous robots operating in unstructured and unknown environments require simultaneous localization and mapping (SLAM). Given the information provided by one or multiple sensors, SLAM algorithms estimate the system's pose as well as a representation of the environment. A key feature of these systems is the ability to recognize previously mapped places, allowing for the correction of the drift inherently occurring in open-loop estimation frameworks. Place recognition or, more specifically, loop closure detection is a well-studied topic in robotics. The development of appearance-based 2D local feature detectors and descriptors

<sup>†</sup> \*The authors assert equal contribution and joint first authorship.

<sup>1</sup>Centre for Autonomous Systems at the Faculty of Engineering and IT, University of Technology Sydney {cedric.legentil@student.uts.edu.au, teresa.vidalcaljeja@uts.edu.au}

<sup>2</sup>German Aerospace Center (DLR), Institute of Robotics and Mechatronics {firstname.lastname@dlr.de}

<sup>3</sup>Technical University of Munich (TUM), Department of Computer Science {rudolph.triebel@in.tum.de}

©2020 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works

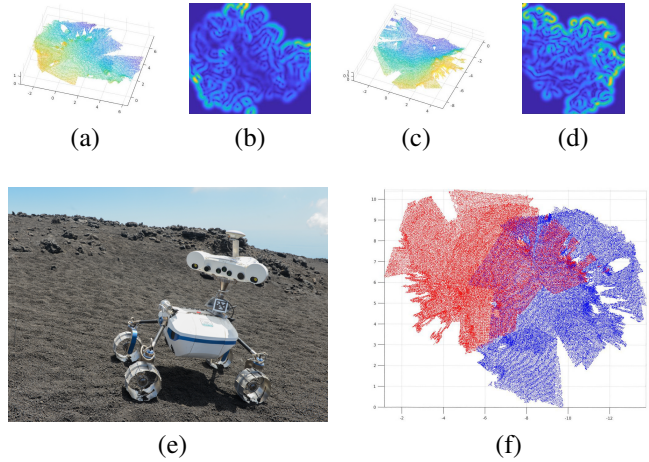


Fig. 1: The proposed method performs loop-closure detection in challenging unstructured scenarios based on geometric representations of the environment. The input maps (a) and (c) are used to generate Gaussian gradient maps of the terrain elevation (b) and (d), respectively. These gradient maps are matched via a feature-based RANSAC algorithm. The platform used in our experiments on Mt. Etna (Italy) is shown in (e), photo courtesy of Esther Horvath. (f) shows the successful loop-closure detection from (b) and (d) verified from aligning input pointclouds using their poses.

has enabled many reliable and large-scale place recognition systems using vision [1].

In this paper, we are interested in tackling the place recognition problem encountered in planetary exploration. Unstructured planetary alike environments make the place recognition task extremely challenging using vision-based systems, especially when viewpoints of the same location differ due to the exploratory nature of the robot's path. In the literature, besides appearance, the 3D structure of the environment has also been exploited for loop closure detection. Features extracted from registered 3D point clouds coming from stereo-systems, depth cameras or LiDARs are commonly used in the recognition tasks. This 3D representation, however, is usually sparse, noisy and incomplete. Moreover, the existing 3D descriptors are in general not as reliable as the visual features, but more importantly for our application, are also challenged by the lack of structure of the terrain.

The hereby proposed method aims to establish correspondences for loop closure detection by leveraging the information contained in the 3D representation of the environment combined with 2D robust visual descriptors. Based

on gravity-aligned 3D point clouds of the robot’s vicinity, our approach introduces a novel continuous and probabilistic representation of the terrain elevation and its variations. This new type of map can be sampled into an image-like data structure, and therefore, be used to perform visual feature extraction and matching.

The proposed approach makes use of Gaussian Process (GP) regression [2] in association with linear operators [3] to build continuous representations of the gradient of the terrain elevation. These high entropy representations, denoted by GP gradient maps, deal with both data uncertainty and incompleteness effectively thanks to the continuous and probabilistic nature of GPs. Our approach takes advantage of this high entropy gradient maps to extract visual-like features and descriptors to find potential correspondences with previously mapped areas. Given these correspondences, a RANSAC-based  $SE(2)$  registration approach on the GP gradient maps and the associated uncertainties are used to verify the loop closure detection. A stereo-visual-inertial SLAM system is used to demonstrate the validity and performance of the proposed approach in an arid-desert and a volcano area, emulating the challenges encountered in planetary exploration scenarios.

In summary, the paper contributions are:

- A novel loop closure detection pipeline for unstructured planetary environments using GP and linear operators for point-cloud-based mapping systems.
- A representation based on continuous and probabilistic gradient maps that enables both, 2D feature matching and  $SE(2)$  registration in a RANSAC-based approach that makes use of the uncertainty information to robustly detect loop closures.
- An experimental validation of the proposed approach in a submap-based SLAM system with real datasets collected on Mt. Etna and in Morocco.

Fig. 1 shows an example of the input and output data from our proposed loop-closure detection approach using data from Mt. Etna.

The remainder of the paper is organized as follows: in Section II, a short overview of the existing work on place recognition with a focus on point-cloud-based methods as well as on the usage of GPs for modelling of the environment is presented. In Section III, the overview of the proposed method is described, followed by the sections IV and V where the two main components of the pipeline, i.e. the GP gradient map representation and the loop closure validation are explained. Section VI describes briefly the submap based SLAM system, which we used to validate the proposed method along with the implementation details regarding the Gaussian gradient maps. Finally, the experiments and results are presented in Section VII with some concluding remarks in Section VIII.

## II. RELATED WORK

Establishing loop closures in visual SLAM is traditionally performed by detecting the similarity of the current image to

a database. Visual similarity is usually expressed as the co-occurrence of image features such as SIFT [4], SURF [5] or ORB [6], often detected by means of aggregation techniques such as bag-of-words [7], VLAD [8] or the more recent HBST [9] targeted at binary descriptors. Pure image-based approaches suffer to relocalize while revisiting the same places from different viewpoints and lighting conditions. For this reason, many approaches to loop closure detection leverage the 3D structure of the environment in the form of point clouds, obtained from LiDAR sensors, stereo cameras or dense multi-view stereo from monocular observations. The authors of [10] evaluate a localization system where LiDAR point clouds captured by a ground vehicle are matched to point clouds from dense visual structure-from-motion through a variety of 3D descriptors. In [11] global localization of a camera in LiDAR maps is obtained by computing and matching structural descriptors, such as the 3D Gestalt [12], on both the map from visual SLAM and from the aggregated LiDAR point clouds. Similarly, monocular LiDAR point clouds are matched to a global LiDAR map in [13] for the purpose of online localization although requiring initial pose priors.

Loop closure detection from homogeneous data (point clouds to point clouds) is addressed in several ways. For LiDAR scans, relocalization can be achieved using global descriptors such as ScanContext [14] where the spatial distribution of points is encoded in an image, which is matched to a database. The approach presented in [15] relies on matching local 3D features, derived from SHOT [16], followed by a probabilistic voting step to reject false transformation hypotheses. In [17]–[19], LiDAR clouds are discretized in segments, representing clusters of points spatially separated between each other. In the latest iteration of the author’s pipeline, segments are encoded into low dimensional descriptors from an auto-encoder and matched to a database to either close loops or relocalize on a prior map. Recently, deep learning techniques have been leveraged for point cloud registration to either obtain more accurate local descriptors [20], [21] or robust keypoints [22]. Although demonstrating higher performances than handcrafted methods, the computational complexity makes their implementation on resource-constrained vehicles challenging.

Although not for loop closure detection, application of Gaussian Processes to produce continuous and probabilistic elevation maps from 3D point clouds has been previously studied in the literature. In [23], GPs have been used to model the spatial correlation of observed terrain data to infer topography from the new observations. GP elevation maps in [24] are used as a prior for Bayesian fusion. Other GP-based mapping properties such as occupancy [25], thickness [26], implicit surface [27], [28], among others, have been extensively studied in recent years. Most of these works exploit the probabilistic nature and the inference capabilities to fill-up areas of missing data, to produce maps of a desirable resolution, to filter noise or for data fusion. Our work, in contrast, exploits the derivative (through linear operators) of the continuous elevation terrain to extract features of the

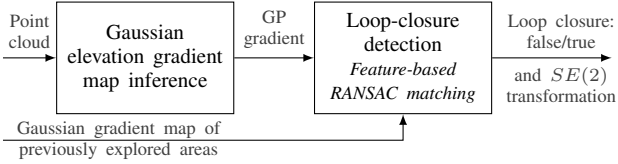


Fig. 2: Overview of the proposed pipeline of place recognition for loop-closure detection based on geometric representations of the environment.

gradient maps and the probabilistic aspect to limit the area of interest and validate loop closures.

Moreover, the use of GP regression and linear operators has already been leveraged for robotics state estimation in [29] to create continuous and accurate pre-integrated measurements from noisy inertial readings. In [30], GP regression is used to recover implicit surfaces with normal estimates for 3D surface reconstruction. In a similar way, here, it is used to generate gradient maps. Operating on the gradient instead of elevation directly makes the proposed pipeline elevation-invariant, a key aspect that is leveraged for registration in the loop closure validation process.

### III. METHOD OVERVIEW

The proposed method performs loop-closure detection given gravity-aligned 3D point clouds of the environment. As shown in Fig. 2, our method is divided into two distinct parts.

The first one is the application of GP regression [2] and linear operators on kernels [3] to convert incoming point clouds into Gaussian gradient maps of the terrain elevation. It relies on the assumption that the true underlying geometry can be modelled with terrain elevation maps. This assumption is accurate as per the rarity of overhanging structures in the considered scenario. The key feature of this continuous probabilistic representation is the possibility to sample the elevation’s variance and gradient at any point in the  $x$ - $y$  plane.

The second part matches newly inferred gradient maps to previously mapped areas. A RANSAC matching procedure is conducted upon traditional visual features extracted from the gradient maps while leveraging the geometric ( $SE(2)$  transformations) and probabilistic (availability of variance) properties of the novel representation.

Note that this method is highly versatile as it is agnostic to the mapping technique used to generate the 3D-geometry of the environment. Therefore, one can directly apply this pipeline to the output of any mapping system that generates 3D data regardless of the modality used (LiDAR, stereo-vision, RGBD cameras, etc.). The loop closure detection in the context of this paper aims at establishing the correspondence between the current and previously mapped areas in the form of submaps. Online loop closure triggering is out of the scope of this work. In the following, for convenience, we consider a collection of submaps  $\mathcal{S}^m$ , with  $m$  the index of the submap, to represent both the previously explored area and the newly discovered one.

## IV. GP GRADIENT MAP REPRESENTATION

This section introduces the derivations associated with the generation of the *gradient maps* based on Gaussian Process regression [2] and the use of linear operators on the kernel covariance function [3].

### A. Gaussian Process regression

Let us consider a terrain elevation submap  $\mathcal{S}^m$  constituted of  $N^m$  points  $\mathbf{p}_i^m = [x_i^m \ y_i^m \ z_i^m]^\top$  ( $i = 1, \dots, N^m$ ) with  $m$  being the index of the submap. The  $z$ -coordinate of  $\mathbf{p}_i^m$  is represented as a function  $z_i^m = f(\mathbf{x}_i^m)$  at the  $x$  and  $y$ -coordinates of the same point, with  $\mathbf{x}_i^m = [x_i^m \ y_i^m]$ .

Considering the elevation function  $z^m$  modelled with a zero-mean GP [2] as

$$f(\mathbf{x}^m) \sim \mathcal{GP}(0, k(\mathbf{x}_i^m, \mathbf{x}_{i'}^m)),$$

$$z_i^m = f(\mathbf{x}_i^m) + \eta_i, \quad \eta_i \sim \mathcal{N}(0, \sigma_z^2), \quad (1)$$

where  $k(\mathbf{x}_i^m, \mathbf{x}_{i'}^m)$  is the kernel covariance function, the elevation can be inferred for any  $x$ - $y$  coordinates  $\mathbf{x}^m$  as

$$z_*^m(\mathbf{x}^m) = \mathbf{K}(\mathbf{x}^m, \mathbf{X}) [\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_z^2 \mathbf{I}]^{-1} \mathbf{z}, \quad (2)$$

$$\text{var}(z_*^m) = \mathbf{K}(\mathbf{x}^m, \mathbf{x}^m) - \mathbf{K}(\mathbf{x}^m, \mathbf{X}) [\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_z^2 \mathbf{I}]^{-1} \mathbf{K}(\mathbf{x}^m, \mathbf{X})^\top, \quad (3)$$

with  $\mathbf{X}$  the matrix built by stacking the  $x$ - $y$  coordinates  $\mathbf{x}^m$  of the training points,  $\mathbf{z}$  the vector of training values  $z_i^m$  at  $\mathbf{X}$ , and  $\mathbf{K}(\cdot, \cdot)$  the matrix of covariances evaluated with the kernel covariance function  $k(\cdot, \cdot)$  between each pair of arguments.

### B. Gaussian gradient maps

In order to propose a method that is invariant with respect to the elevation of the submaps’ origin, the proposed framework generates a representation of the gradient of the submaps’ elevation. Overall this helps the feature matching robustness, and allows for the use of difference-based metrics to register gradient maps together. The  $x$  and  $y$  gradients of  $z^m$  are directly inferred from the GP model (1) by applying the differentiation linear operator to the covariance kernel as introduced in [3]:

$$\frac{\partial z^m}{\partial x^m} = \frac{\partial \mathbf{K}(\mathbf{x}^m, \mathbf{X})}{\partial x^m} [\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_z^2 \mathbf{I}]^{-1} \mathbf{z}, \quad (4)$$

$$\frac{\partial z^m}{\partial y^m} = \frac{\partial \mathbf{K}(\mathbf{x}^m, \mathbf{X})}{\partial y^m} [\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma_z^2 \mathbf{I}]^{-1} \mathbf{z}. \quad (5)$$

From these continuous representations, the proposed method infers the elevation variance and the gradient submaps as mono-channel image-like data-structures. For a submap  $\mathcal{S}^m$ ,  $I_v^m$  denotes the variance image-like data-structure inferred from (3), and  $I_g^m$  the elevation’s gradient computed from (4) and (5) using  $g = \sqrt{\frac{\partial z^m}{\partial x^m}^2 + \frac{\partial z^m}{\partial y^m}^2}$ . Fig. 3 shows an example of a gradient map and the associated terrain variance. As the figure shows, gradient maps produce a higher entropy representation compared to simple elevation maps. Thus, visual features are more present in these maps.

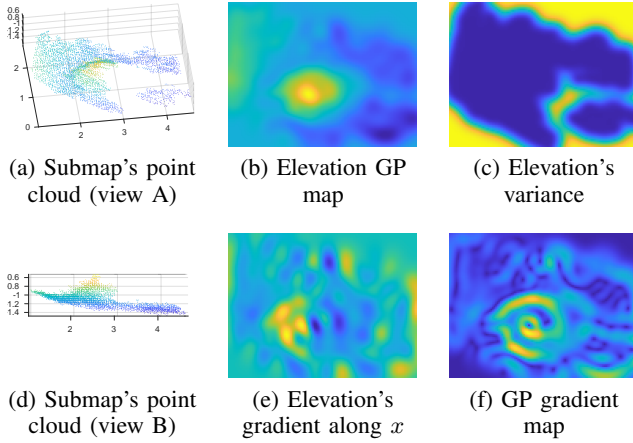


Fig. 3: Example of gradient map generation from a 3D point cloud. (a) and (d) are two different viewpoints of the input 3D point cloud. (b) and (c) are the GP continuous representations of the terrain’s elevation and the associated variance (from (2) and (3)). (e) represents an intermediate step (from (4)) needed to compute the GP gradient map shown in (f).

## V. LOOP-CLOSURE DETECTION

Given two instances of Gaussian gradient maps,  $I_g^d$  and  $I_g^q$ , the proposed method classifies a pair of maps as loop-closure or not by attempting a feature-based registration between  $I_g^d$  and  $I_g^q$ . Note that the superscript  $d$  corresponds to the index of a submap in the database of previously explored areas, and the superscript  $q$  corresponds to the index of the query submap. First, 2D keypoints are extracted individually in  $I_g^d$  and  $I_g^q$ . The keypoint detection leverages the gradient maps’ variances,  $I_v^d$  and  $I_v^q$ , by applying masks to reject keypoint candidates that are far from the GP training data (high variance areas). Each keypoint is associated with a 2D descriptor computed on the gradient map instances  $I_g^d$  and  $I_g^q$ . As illustrated in Fig. 4, the descriptors are matched across both gradient maps based on a brute-force strategy and  $L_2$  proximity metric.

This last step results in a set  $\mathcal{A}$  of keypoint associations  $\mathbf{a}_i$  across both maps. Given this set  $\mathcal{A}$ , the proposed method attempts the  $SE(2)$  registration of  $I_g^q$  to  $I_g^d$  in a RANSAC algorithm shown in Alg. 1. The different elements employed are:

- *SelectRandomAsso*: the random selection of a pair of associations  $\mathbf{a}_a$  and  $\mathbf{a}_b$  from  $\mathcal{A}$ .
- *GetSE2Transform*: the estimation of a  $SE(2)$  transformation  $\mathbf{T}_i$ , given a subset of  $\mathcal{A}$ , based on centroid alignment and SVD.
- *GetInliers*: the identification of the inlier set of association  $\mathcal{A}_i$ , given  $\mathcal{A}$  and a  $SE(2)$  transformation  $\mathbf{T}_i$ , based on 2D distance threshold.
- *ComputeSSDMetric*: the computation of the similarity metric  $h_i$  between two registered gradient maps as the Sum of Squared gradient Differences (SSD) weighted by the inverse of the product of the corresponding variances  $h_i = \sum_{\mathbf{x}^{d,q}} \frac{(I_g^q(\mathbf{T}_i \mathbf{x}^{d,q}) - I_g^d(\mathbf{x}^{d,q}))^2}{I_v^q(\mathbf{T}_i \mathbf{x}^{d,q}) I_v^d(\mathbf{x}^{d,q})}$ .

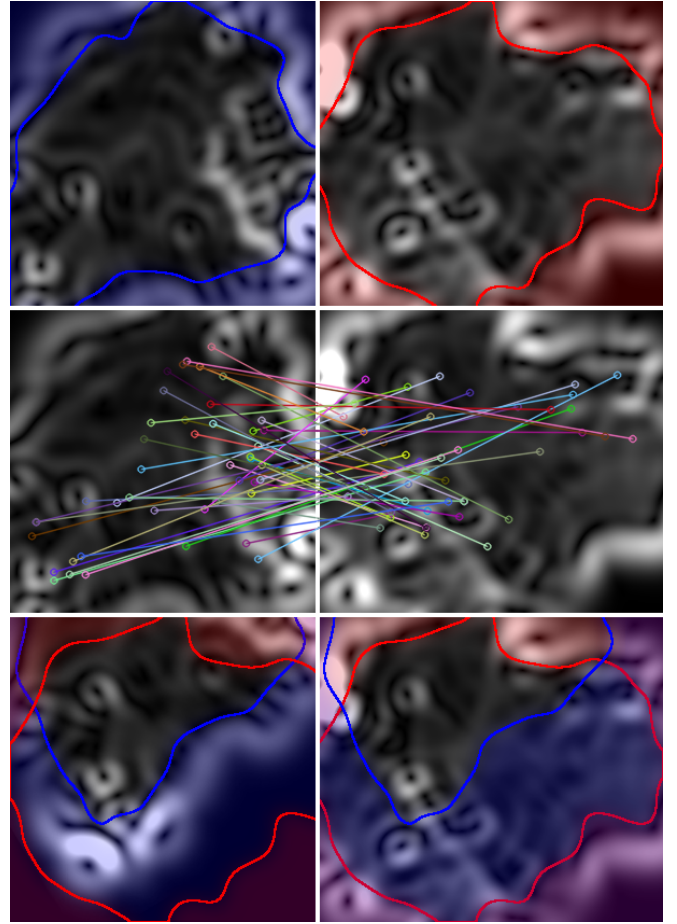


Fig. 4: Visualization of the loop-closure check via gradient map matching. Top row: Input gradient submaps from Etna3 dataset (overlaid with corresponding variance in red and blue for intuitive visualization). Middle row: 2D descriptor (SURF) correspondences in low variance areas between the two gradient submaps. Bottom row: Gradient maps aligned after finding the loop closure (overlaid with both submaps’ variance for intuitive visualization).

- *AcceptTransformation*: the decision *true/false* to accept the estimated transformation  $\mathbf{T}_i$  based on the SSD metric  $h_i$  and the number of inliers found  $|\mathcal{A}_i|$ .

The output of this procedure is the maximum number of inliers found  $n$  and the associated SSD similarity metric  $h$ . Empirically we found that setting a threshold on the number of inliers appeared to be the most effective metric to classify two maps as loop closure. Extensive tests have been conducted in our experiments to best define this threshold. Note that this pipeline is agnostic to the choice of feature descriptor. In our implementation, we use Speeded Up Robust Features (SURF) as their blob detection ability matches the smooth nature of the Gaussian gradient maps.

## VI. IMPLEMENTATION

This section contains a brief introduction of the mapping system used in our experiments, as well as implementation details regarding the Gaussian gradient maps and the overall computation time.

---

**Algorithm 1:** Loop-closure classification based on  $SE(2)$  registration between two Gaussian gradient maps.

---

**Input :**

- $I_g^d, I_g^q, I_v^d, I_v^q$ : Gradient maps and associated variance
- $\mathcal{A}$ : Set of keypoint associations between  $I_g^d$  and  $I_g^q$

**Output:**

- $n$ : maximum number of inliers found
- $h$ : associated SSD metric
- $\mathbf{T}$ : associated  $SE(2)$  transformation estimate

**foreach** until Max Nb Iterations **do**

```

    { $\mathbf{a}_a, \mathbf{a}_b$ } = SelectRandomAsso( $\mathcal{A}$ );
     $\mathbf{T}_i$  = GetSE2Transform({ $\mathbf{a}_a, \mathbf{a}_b$ });
     $\mathcal{A}_i$  = GetInliers( $\mathcal{A}, \mathbf{T}_i$ );
     $\mathbf{T}_i$  = GetSE2Transform( $\mathcal{A}_i$ );
     $h_i$  = ComputeSSDMetric( $\mathbf{T}_i, I_g^d, I_g^q, I_v^d, I_v^q$ );
    if AcceptTransformation( $|\mathcal{A}_i|, h_i$ ) and  $|\mathcal{A}_i| > n$  then
         $n = |\mathcal{A}_i|$ ;
         $h = h_i$ ;
         $\mathbf{T} = \mathbf{T}_i$ ;

```

**return**  $\mathbf{T}, n, h$ ;

---

### A. Submap-based SLAM

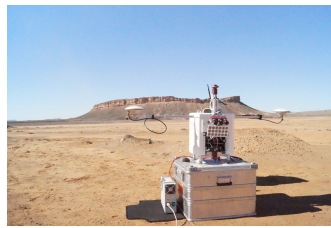
In order to evaluate the performance of our pipeline, we employ our submap-based SLAM framework [31], where gravity-aligned 3D submaps are incrementally built from aggregated stereo clouds and positioned according to visual-inertial pose estimates. The size of submaps is bounded either by limits on the trajectory length from their respective origins or on the pose uncertainty. This ensures that the resulting point clouds are locally accurate. Submap origins are joined consecutively in a graph by relative visual-inertial constraints and, in the original implementation, inter-submap matches [32] establish loop closure constraints. In this work, we evaluate the feasibility of establishing the loop closure between submaps using a GP representation instead of matching 3D keypoints using CSHOT descriptors as in [32].

### B. Gaussian gradient maps and Computation

Our implementation is built upon the use of the square-exponential kernel

$$k(\mathbf{x}_i^m, \mathbf{x}_{i'}^m) = \sigma_k \exp\left(-\frac{(\mathbf{x}_i^m - \mathbf{x}_{i'}^m)^\top (\mathbf{x}_i^m - \mathbf{x}_{i'}^m)}{2l_k}\right) \quad (6)$$

to generate the GP gradient submaps. The hyperparameters  $\sigma_k$ ,  $l_k$ , and  $\sigma_z$  (data noise from (1)) are respectively initialized with the elevation's empirical variance for the submap, a manually-set length-scale (around 0.1 m), and the typical elevation accuracy of the submapping system, before being fine-tuned as described in [2]. The submaps are arbitrarily downsampled to 5000 points in order to reduce the computation time. The query point inferences are performed with a spatial resolution of 0.03 m/pixel over the minimum  $x$ - $y$  bounding box of each point cloud. Our current implementation computes each GP gradient map in approximately five minutes, and the RANSAC matching takes around seven minutes for a database of 14 submaps. The computations were performed on a laptop with Intel



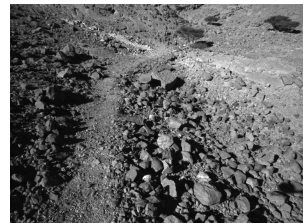
(a) DLR Sensoric Unit of Planetary Exploration Rovers (SUPER)



(b) Morocco Experiment Site



(c) Etna3 Image frame



(d) Morocco Image frame

Fig. 5: DLR Sensoric Unit of Planetary Exploration Rovers (SUPER) at the experiment site in Morocco (a) and the location at which the particular dataset from Morocco was collected (b). Impressions of the Etna (c) and Morocco (d) landscapes as seen from the navigation camera. Notice the lack of unique visual features in (c) and the harsh lighting conditions in (d).

Core i7-6700HQ CPU. The overall computation time can be greatly reduced, as mentioned in the future work discussed in Section VIII.

## VII. EXPERIMENTS AND DISCUSSION

The pipeline was tested with five datasets. Four were collected with the Lightweight Rover Unit (LRU) (Fig 1) during the 2017 Etna demo space mission under the scope of the ROBEX project [33]. The LRU is a lightweight space rover prototype equipped with an IMU, wheel odometry and a stereo camera as the primary sensor for mapping. A detailed description of the LRU with its hardware and software architecture can be found in [34].

The fifth dataset used has been collected in a planetary analogue environment in Morocco in 2018. The hand-held sensor suite employed (same as the one integrated into the rover) is shown in Fig. 5a. This dataset has been specifically collected to validate mapping and navigation algorithms in particularly challenging unstructured environments. Different datasets collected in Morocco are in the process of being made publicly available for the robotics community.

From hereon we refer to the five datasets as Etna1, Etna2, Etna3, Etna4 and Morocco. To give an impression of the trajectories from these datasets, ground-truth position tracks as computed from DGPS are shown in Fig. 6. To give an impression of views from the navigation camera instead, see Fig. 5c, and Fig. 5d.

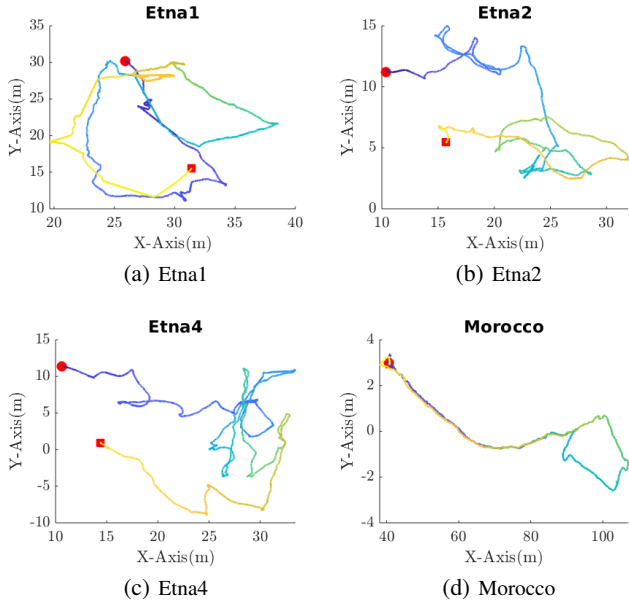


Fig. 6: Ground-truth trajectories of the datasets computed using DGPS data (not shown for Etna3 due to missing data). Trajectories are colored according to the timestamps. The start positions are shown with a red circle and the end positions with a red square.

#### A. Evaluation metrics

We evaluate the performances of our loop closure detector using the *precision-recall* metric:

$$P = \frac{T_p}{T_p + F_p} \quad R = \frac{T_p}{T_p + F_n}, \quad (7)$$

where  $T_p$  is the number of correct submap matches validated by the proposed method,  $F_p$  is the number of wrong matches, and  $F_n$  is the number of true submap matches which were not detected. We choose to determine whether a match is true or false from the spatial overlap of submaps, which are positioned in world coordinates using the estimated poses from our SLAM system [31]. We consider the output of SLAM accurate enough to validate the approach. Specifically, we compute a bounding box for each submap in the  $x$  and  $y$  global coordinates. Let  $A(S^m)$  and  $A(S^n)$  be areas of the bounding boxes for submaps  $S^m$  and  $S^n$ . The overlap  $o(S^m, S^n)$  is defined as the Intersection over Union (IoU):

$$o(S^m, S^n) = \frac{A(S^m) \cap A(S^n)}{A(S^m) \cup A(S^n)} \in [0, 1]. \quad (8)$$

Resulting from empirical considerations about our data, for the Etna datasets we decide that true matches are those for which  $o(S^m, S^n) > 0.3$ , and for Morocco where we choose a minimum overlap of 0.1. Although it is guaranteed that submaps which have little to none overlap are false matches, overlapping submaps which have no unique 3D structure will not be matched regardless of the parameter choices for our algorithm. Therefore, performances will be penalized in terms of *recall* in most datasets. Precision-recall curves are generated by varying the minimum number of inlier SURF matches  $n$  between GP gradient maps from 1 to 20.

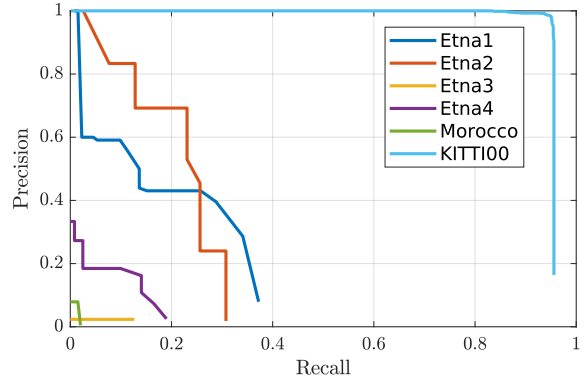


Fig. 7: Results for place recognition with the state-of-the-art, visual-only iBoW-LCD approach [35]. We compare precision-recall curves obtained from the KITTI autonomous driving dataset with those obtained from our data. We conclude that traditional visual place recognition is perfectly adequate in the context of autonomous driving, but the performance drops significantly on images captured by a planetary exploration rover in challenging unstructured environments.

#### B. The inadequacy of visual-only approaches

Traditional loop closure detection based on image features and bag-of-words demonstrates excellent performances in datasets oriented towards autonomous driving. Nonetheless, the extreme visual aliasing and strong viewpoint differences encountered in planetary exploration scenarios pose great challenges for visual-only place recognition algorithms. To demonstrate this, we evaluated the state-of-the-art incremental loop-closure detector iBoW-LCD [35] on images from our Etna and Morocco datasets as well as on the first sequence from the KITTI dataset [36]. We extracted 1500 ORB features per frame and adapted the parameters recommended in the paper to achieve the best performances in each dataset. True matches were labelled based on the spatial and viewpoint proximity of the camera pose related to each frame. Fig 7 shows the precision-recall curves obtained with iBoW-LCD applied to our datasets and the KITTI100 sequence. The appearance-based place recognition method performs well on the KITTI sequence thanks to the specific motion of the car and the distinctive features present in man-built environments. However, iBoW-LCD performed poorly on the Etna and Morocco datasets due to the ambiguous visual appearance and the proximity of the viewpoints with the ground. Furthermore, on the Morocco sequence, visual-only loop-closure detection through standard appearance approaches is not possible as the camera trajectory repeats the same path from an almost opposite direction. This experiment demonstrates the necessity for alternative solutions. In the following subsection, we will show how our proposed approach based on GP gradient maps helps to overcome the limitations of traditional place recognition methods.

#### C. GP gradient maps vs. 3D feature matching

Here we evaluate the proposed pipeline for loop closure detection against a traditional approach based on matching

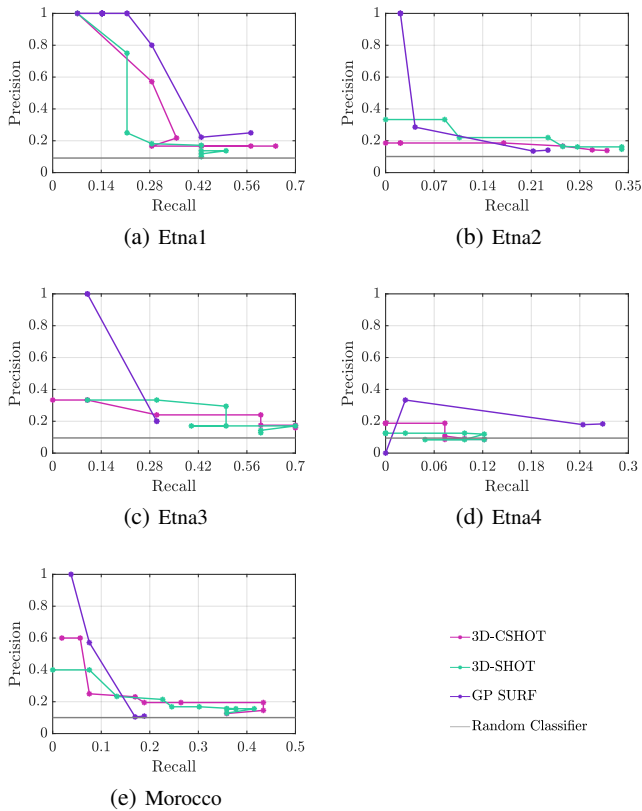


Fig. 8: Precision-recall curves comparing the performance of the proposed GP gradient method with 3D descriptor matching as well as random classifier.

3D features extracted from submap point clouds. To this end, keypoints are sampled from high curvature regions as in [37] and matched across submap pairs using SHOT and CSHOT features. The resulting keypoint matches are then filtered from outliers using a RANSAC approach as implemented in the PCL library [38], setting the probability of selecting an outlier-free model to 0.99.

Fig. 8 shows the precision-recall curves for the Etna and Morocco sequences. As can be observed for all tests, except for Etna4 in Fig. 8d, all the curves for our method start at 100% precision. The curves show that our approach is more accurate and conservative than the baseline: for a set of input parameters, the gradient maps pairs that pass the RANSAC test for SURF matching will likely be all true, while many validated matches from CSHOT+RANSAC and SHOT+RANSAC are wrong even for the more conservative descriptor matching thresholds. Fig. 9 reports the precision obtained in all datasets as a function of the minimum number of inlier SURF matches after RANSAC  $n_{inl}$ . The plot shows that for  $n_{inl} \geq 4$ , submap matches can be selected with 100% precision on all datasets except for Etna4, which the precision-recall curves in Fig. 9 prove to be a very challenging scenario for loop closure detection. Indeed on Etna4 performances are very limited also for appearance-based methods, see Fig. 7. For  $n_{inl} \geq 5$  all the selected

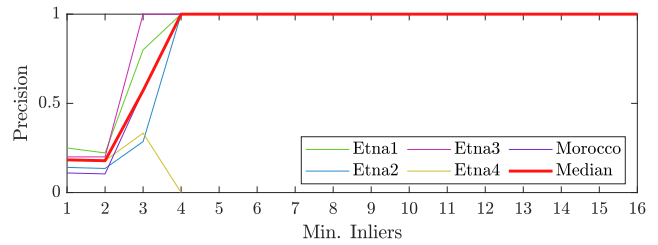


Fig. 9: Average precision across all the datasets as a function of the threshold on the number of inliers in the proposed RANSAC-based matching.

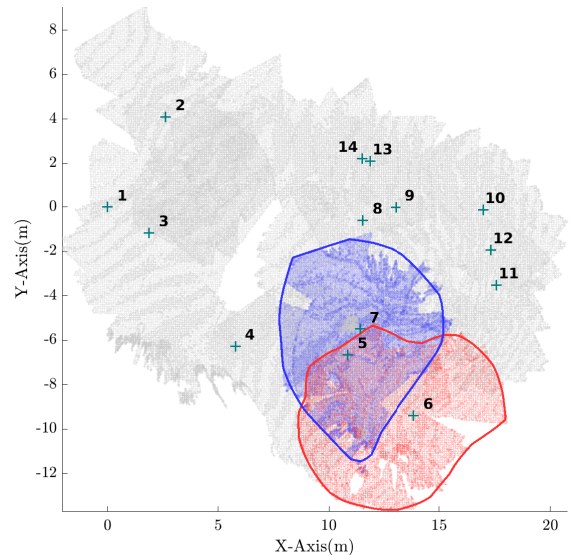


Fig. 10: The two identified loop closure candidates from Etna3 i.e. submaps 5 and 6 in red and blue, rest of the submaps from the session in grey aligned using pose estimation results. The numbers represent the submap index and the position of each submap local reference frame estimated by the visual-inertial odometry is marked with a '+' symbol.

matches are instead correct, achieving the highest precision among the tested baseline approaches.

The two matching submaps that were correctly identified from Etna3 with this method are shown in Fig. 10 as an example. Notice how for Etna3, loop closures can not be established with the tested appearance-based method [35], see Fig. 7. The benefits provided by our method against both 3D feature matching and visual methods especially emerge when the camera traverses a path from opposite directions, such as for the traverse in Morocco (see Fig.6). In this case, overlapping submaps do not share significant structures that can be matched, such as rocks, as they are reconstructed on opposing viewpoints. In addition, images captured from close positions observe different portions of the environment. Our approach is instead able to deal with missing observations and produce correct submap matches.

## VIII. CONCLUSIONS

In this paper, we presented a method to detect loop closures in unstructured planetary environments based on



Gaussian gradient maps. The novel Gaussian gradient representation is a continuous probabilistic representation of the environment's terrain elevation based on GP regression and the application of linear operators on the kernel. This method allows image-like inferences of the terrain's elevation alongside its associated variance. The gradient maps are then matched through a  $SE(2)$ -constrained feature-based RANSAC algorithm that leverages the uncertainty knowledge of the terrain. The method and its robustness to noisy input data have been validated in five challenging datasets outperforming traditional 3D feature matching.

Future works include the integration of this pipeline into a real-time multi-modal and multi-session mapping framework leveraging both visual and structural similarities to better generalize to a wider range of scenarios, as well as extending it to heterogeneous sensor setups (LiDAR, stereo and RGB-D cameras). To this end, work will also be conducted to reduce the computational cost of the overall pipeline. At the method level, one can easily consider inducing points as in [39] to breakdown the cubic complexity of the GP gradient map generation. The use of bag-of-words techniques like [35] can greatly reduce the number of RANSAC matching attempts performed. In terms of implementation, many operations are highly parallelizable (GP inferences, RANSAC runs, matching score, etc.) and would benefit from optimized computation on GPU.

#### ACKNOWLEDGMENT

This work was supported by the UA-DAAD 2018 Australia-Germany Joint Research Cooperation Scheme, project COSMA (contract number 57446007) and the project ARCHES (contract number ZT-0033).

#### REFERENCES

- [1] S. Lowry, *et al.*, "Visual place recognition: A survey," *IEEE Transactions on Robotics*, vol. 32, no. 1, pp. 1–19, 2016. [Online]. Available: <https://doi.org/10.1109/TRO.2015.2496823>
- [2] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.
- [3] S. Särkkä, "Linear operators and stochastic partial differential equations in Gaussian process regression," in *International Conference on Artificial Neural Networks (ICANN)*, 2011, pp. 151–158.
- [4] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.
- [5] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *European Conference on Computer Vision (ECCV)*, 2006, pp. 404–417.
- [6] E. Rublee, *et al.*, "ORB: An efficient alternative to SIFT or SURF," in *International Conference on Computer Vision (ICCV)*, 2011, pp. 2564–2571.
- [7] D. Galvez-Lopez and J. D. Tardos, "Bags of binary words for fast place recognition in image sequences," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1188–1197, 2012.
- [8] R. Arandjelovic and A. Zisserman, "All about VLAD," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2013, pp. 1578–1585.
- [9] D. Schlegel and G. Grisetti, "HBST: A Hamming Distance Embedding Binary Search Tree for Feature-Based Visual Place Recognition," *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3741–3748, 2018.
- [10] A. Gawel, *et al.*, "3d registration of aerial and ground robots for disaster response: An evaluation of features, descriptors, and transformation estimation," in *IEEE International Symposium on Safety, Security and Rescue Robotics (SSRR)*, 2017, pp. 27–34.
- [11] —, "Structure-based vision-laser matching," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 182–188.
- [12] M. Bosse and R. Zlot, "Place recognition using keypoint voting in large 3D lidar datasets," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2013, pp. 2677–2684.
- [13] T. Caselitz, *et al.*, "Monocular camera localization in 3D LiDAR maps," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 1926–1931.
- [14] G. Kim and A. Kim, "Scan context: Egocentric spatial descriptor for place recognition within 3D point cloud map," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4802–4809.
- [15] J. Guo, *et al.*, "Local descriptor for robust place recognition using lidar intensity," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1470–1477, 2019.
- [16] F. Tombari, S. Salti, and L. Di Stefano, "A combined texture-shape descriptor for enhanced 3D feature matching," in *IEEE International Conference on Image Processing (ICIP)*, 2011, pp. 809–812.
- [17] R. Dubé, *et al.*, "SegMap: 3D segment mapping using data-driven descriptors," in *Robotics: Science and Systems (RSS)*, 2018.
- [18] —, "SegMap: Segment-based mapping and localization using data-driven descriptors," *The International Journal of Robotics Research*, vol. 39, no. 2–3, pp. 339–355, 2020.
- [19] —, "Incremental-segment-based localization in 3-D point clouds," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1832–1839, 2018.
- [20] Z. J. Yew and G. H. Lee, "3DFeat-Net: Weakly supervised local 3D features for point cloud registration," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 630–646.
- [21] Z. Gojcic, *et al.*, "The perfect match: 3D point cloud matching with smoothed densities," *arXiv preprint arXiv:1811.06879*, 2018.
- [22] H. Wang, *et al.*, "Learning 3D keypoint descriptors for non-rigid shape matching," in *European Conference on Computer Vision (ECCV)*, 2018, pp. 3–19.
- [23] S. Vasudevan, *et al.*, "Gaussian process modeling of large-scale terrain," *Journal of Field Robotics*, vol. 26, no. 10, pp. 812–840, 2009.
- [24] L. Sun, T. Vidal-Calleja, and J. Valls Miro, "Bayesian fusion using conditionally independent submaps for high resolution 2.5D mapping," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2015, pp. 3394–3400.
- [25] S. O'Callaghan and F. Ramos, "Gaussian process occupancy maps," *International Journal of Robotics Research*, vol. 31, no. 1, pp. 42–62, 2012.
- [26] T. Vidal-Calleja, *et al.*, "Learning spatial correlations for Bayesian fusion in pipe thickness mapping," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 683–690.
- [27] O. Williams and A. Fitzgibbon, "Gaussian process implicit surfaces," in *Gaussian Processes in Practice*, 2007.
- [28] L. Wu, *et al.*, "Skeleton-based conditionally independent gaussian process implicit surfaces for fusion in sparse to dense 3D reconstruction," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1532–1539, 2020.
- [29] C. Le Gentil, T. Vidal-Calleja, and S. Huang, "Gaussian Process Preintegration for Inertial-Aided State Estimation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2108–2114, 2020.
- [30] W. Martens, *et al.*, "Geometric priors for gaussian process implicit surfaces," *IEEE Robotics and Automation Letters*, vol. PP, pp. 1–1, 11 2016.
- [31] M. J. Schuster, *et al.*, "Distributed stereo vision-based 6d localization and mapping for multi-robot teams," *Journal of Field Robotics*, vol. 36, no. 2, pp. 305–332, 2019. [Online]. Available: <https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.21812>
- [32] C. Brand, *et al.*, "Submap matching for stereo-vision based indoor/outdoor slam," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5670–5677.
- [33] A. Wedler, *et al.*, "First results of the ROBEX analogue mission campaign: Robotic deployment of seismic networks for future lunar missions," in *68th International Astronautical Congress (IAC)*, 2017.
- [34] M. J. Schuster, *et al.*, "Towards autonomous planetary exploration the lightweight rover unit (LRU), its success in the spacebotcamp challenge, and beyond," *Journal of Intelligent & Robotic Systems*, 2017.
- [35] E. Garcia-Fidalgo and A. Ortiz, "IBoW-LCD: An Appearance-Based Loop-Closure Detection Approach Using Incremental Bags of Binary

- Words,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3051–3057, 2018.
- [36] A. Geiger, P. Lenz, and R. Urtasun, “Are we ready for autonomous driving? the KITTI vision benchmark suite,” *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, 2012.
- [37] R. Giubilato, *et al.*, “Relocalization with submaps: Multi-session mapping for planetary rovers equipped with stereo cameras,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 580–587, 2020.
- [38] R. B. Rusu and S. Cousins, “3D is here: Point Cloud Library (PCL),” *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–4, 2011.
- [39] J. Quiñero-Candela and C. E. Rasmussen, “A unifying view of sparse approximate Gaussian process regression,” *Journal of Machine Learning Research*, vol. 6, pp. 1939–1959, 2005.