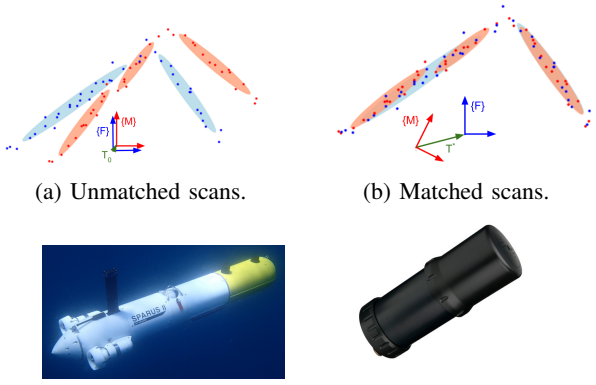
2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

GMM Registration: a Probabilistic scan matching approach for sonar-based AUV navigation

Pau Vial, Miguel Malagón, Ricard Segura, Narcís Palomeras and Marc Carreras[a]

***Abstract*— Acoustic perception in underwater environments is challenging due to the low frequency of the acquisition system and multiple and huge sources of noise. Therefore, point clouds built by profiling sonars mounted on Autonomous Underwater Vehicles (AUV) are sparse and noisy. To solve the mapping task, AUVs need a registration algorithm to prevent maps from inconsistencies. Many scan matching algorithms are available, however, a few of them are specialized in acoustic data. In this paper, a probabilistic scan matching methodology based on Gaussian Mixtures Models (GMM) is presented and, for the first time, the Bayesian-GMM algorithm is applied in this context to model acoustic data. The scan matching problem is properly formulated using Lie groups to define pose. In addition, this methodology can return an uncertainty measure for the matching result, which is fundamental in Pose SLAM applications. This tool is implemented in a public C++ library[1] that can process in real-time 2D and 3D scans acquired by a profiling sonar. Theoretical justification and results with real data are provided to benchmark our method against the state-of-the-art Normal Distributions Transforms (NDT) technique.**



(a) Unmatched scans. (b) Matched scans.

(c) Sparus II AUV [3]. (d) Profiling Sonar [4].

Fig. 1: Scan matching example with corner scans (a) (b). Reference scan in blue and matched scan in red. Ellipsoids plot GMM form and dots, point cloud form. Acoustic scans gathered by a profiling sonar (d) mounted on an AUV (c).

I. Introduction

Water attenuation of electromagnetic waves makes the underwater environment a very challenging domain where the most common sensors in robotics - such as optical cameras and LIDARs - get their capabilities hugely reduced. Thus, underwater perception typically relies on acoustic devices, sensors based on mechanical waves that are slower and noisier than its electromagnetic counterpart. When mapping using acoustic range sensors mounted on an Autonomous Underwater Vehicles (AUV), registration algorithms are needed to establish the displacement between two overlapping scans, as it is exemplified in Fig. 1. A registration algorithm not only lets achieve consistent maps, but also improves AUV dead reckoning navigation. In this paper a scan matching methodology specially suited for noisy and sparse point clouds, the key features of acoustic data, is presented. This methodology uses Gaussian Mixtures Models (GMM) to model sensor noise and the structure of the perceived scene, building a probabilistic model of the robot surroundings. To fit a GMM, the Bayesian-GMM algorithm [1] is applied to the registration problem for the first time. This algorithm learns the optimal number of GMM components needed to model a point cloud without any previous assumption on the model structure. Moreover, to solve the registration, an optimization problem properly defined on Lie groups [2] is solved and an uncertainty measure for each particular result is provided based on the scans structure. This last feature is not common in many scan matching algorithms. The methodology is implemented in a public C++ library, called `GMM Registration`[1], that can register both 2D and 3D scans. The effectiveness of our proposal is shown on real data gathered with our AUVs (Fig. 1c). The contributions of this paper are: (i) the application of the Bayesian-GMM algorithm (Section III-A); (ii) the problem formulation using Lie groups (Section III-B); (iii) an uncertainty measure for each registration (Section III-C); (iv) a public implementation of the proposed methodology (Section IV); and (v) tests on real data to benchmark our method against the state-of-the art probabilistic techniques (Section V).

II. State of the art

Two broad categories of rigid registration algorithms are available. On the one hand, point-to-point techniques are based on a least squares formulation that establishes point correspondences and applies the Expectation-Maximization (EM) algorithm [5] to solve the registration. Iterative Closest Point (ICP) [6] is the most popular point-to-point algorithm, for which a closed form solution for the M-step is available. On the other hand, probabilistic techniques use GMMs to reach a continuous representation of a point cloud that models the shape of the scanned surface affected by sensor noise. This probabilistic representation avoids to establish hard correspondences on data to solve the registration. This way an EM procedure is not need and the problem can be solved in a single step. However, due to the analytical

[a] Computer Vision and Robotics Research Institute (VICOROB), Universitat de Girona, 17003 Girona, Catalonia (e-mail: pau.vial@udg.edu).

[1] The library repository can be found in https://bitbucket.org/gmmregistration/gmm_registration.

complexity that implies the use of GMMs, no closed form solution for the registration is available and the problem must be solved by optimization on the Lie group defining pose [2].

The most popular probabilistic methodology is the Normal Distributions Transforms (NDT) for which two different algorithms have been proposed: [7] and [8]. This methodology makes use of a Cartesian grid to fit a GMM into a point cloud. Recent papers make use of the K-means algorithm [9] or the EM algorithm for the Gaussian [10], to fit a GMM to the point cloud, and to solve the registration applying similar policies to those used by NDT methodologies. Nevertheless, to the best of the authors knowledge, the Bayesian-GMM algorithm [1] has never been applied to model a scan and solve the registration problem. This algorithm, without imposing any previous assumption on the shape or the number of model components, allows learning the optimal number of components K needed to model a point cloud depending on its intrinsic structure. Therefore, it can be used as an adaptive tool for different types of scenes.

III. Methodology

The scan matching methodology is divided in three blocks. Firstly, the Front-End provides different algorithms to fit a GMM into a point cloud. Secondly, the Method defines two different policies that allows to formulate the registration problem as an optimization problem. Finally, several gradient based solvers specialized for pose optimization are provided.

A. GMM Front-End

A GMM is a linear superposition of Gaussians in the form

$$p(x|\Phi) = \sum^{K} \pi_k \mathcal{N}(x|\mu_k, \Sigma_k) \qquad \text{with} \qquad \sum^{K} \pi_k = 1, \quad (1)$$

where model parameters $\Phi = (\pi_k, \mu_k, \Sigma_k)$ are respectively the weight, mean and covariance of each Gaussian distribution k. In this section, GMM Front-Ends are presented in order of increasing complexity.

1) NDT Front-End : NDT [7] projects the point cloud into a Cartesian grid. For each cell with a minimum number of points, a component is set using the Maximum Likelihood (ML) estimator for the Gaussian

$$\pi_k = \frac{N_k}{N}, \quad \mu_k = \frac{1}{N_k}\sum^{N_k} x_i, \quad \Sigma_k = \frac{1}{N_k}\sum^{N_k} x_i x_i^T - \mu_k \mu_k^T \quad (2)$$

where N_k is the number of points x_i falling in cell k and N is the total amount of points falling in active cells. In the definition of the NDT algorithm [7], multiple shifted grids are weighted to ensure model continuity. However, in the presented approach this trick is not needed as all correspondences are considered during the registration.

2) K-means Front-End: K-means [11] uses a fixed number of components K in which the point cloud must be classified. To do so, an iterative procedure based on the EM algorithm [5] is used. In the E-step, each data point is assigned to the closest component in terms of the euclidean distance. In the M-step, the Gaussian parameters are estimated following the assignations done in the E-step and applying equation 2. The EM algorithm iterates until the inertia of the solution stops improving.

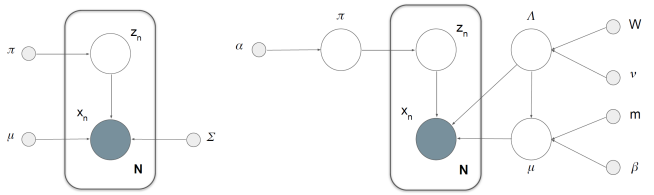
3) EM Front-End: This Front-End applies the EM algorithm for GMMs (see Chapter 9 from [12]). In contrast to K-means, this algorithm does not do hard assignments of points to components, assuming the graphical model of Fig. 2a where latent variables z are modeled as a multinomial distribution. In the E-step a responsibility γ_{ik} for each data point to each cluster is computed

$$\gamma_{ik} = \frac{\pi_k \mathcal{N}(x_i|\mu_k, \Sigma_k)}{\sum_j^K \pi_j \mathcal{N}(x_i|\mu_j, \Sigma_j)}. \quad (3)$$

In the M-step Gaussian components are estimated following the computed responsibilities

$$\pi_k = \frac{N_k}{N}, \quad \mu_k = \frac{1}{N_k}\sum^{N} \gamma_{ik} x_i, \quad \Sigma_k = \frac{1}{N_k}\sum^{N} \gamma_{ik} x_i x_i^T - \mu_k \mu_k^T \quad (4)$$

where $N_k = \sum^{N} \gamma_{ik}$. The EM algorithm iterates until the likelihood of the solution stops improving.



(a) (b)

Fig. 2: Graphical models: (a) GMM and (b) Bayesian-GMM.

4) Bayesian-GMM Front-End: Bayesian-GMM [1] sets conjugate priors to model parameters (see Fig. 2b) and solves the Maximum a Posteriori (MAP) estimator of the M-step by means of Variational Inference (VI). Therefore, the new model parameters are $\Phi^* = (\alpha_k, \beta_k, m_k, W_k, \nu_k)$ which correspond to priors on the original GMM parameters Φ. In the E-step responsibilities are computed as $\gamma_{ik} = \frac{\rho_{ik}}{\sum_j^K \rho_{ij}}$ where

$$\begin{aligned}
\rho_{ik} &= \exp\left(\ln\widetilde{\pi}_k + \frac{1}{2}\ln\widetilde{\Lambda}_k - \frac{D}{2\beta_k} - \frac{\nu_k}{2}(x_i - m_k)^T W_k (x_i - m_k)\right), \\
\ln\widetilde{\pi}_k &= \psi(\alpha_k) - \psi(\textstyle\sum^K \alpha_j), \\
\ln\widetilde{\Lambda}_k &= \sum_j^D \psi(\frac{\nu_k + 1 - j}{2}) + D\ln 2 + \ln|W_k|,
\end{aligned}$$

$\psi(x) \sim \ln x - \frac{1}{2x}$ is the digamma function and D is the points dimension. In the M-step model parameters Φ^* are estimated following the computed responsibilities

$$\begin{aligned}
\alpha_k &= \alpha^0 + \overline{N}_k, \\
\beta_k &= \beta^0 + \overline{N}_k, \\
m_k &= \frac{1}{\beta_k}(\beta^0 m^0 + \overline{N}_k \overline{\mu}_k), \\
W_k^{-1} &= W^{0^{-1}} + \overline{N}_k \overline{\Sigma}_k + \frac{\beta^0 \overline{N}_k}{\beta_k}(\overline{\mu}_k - m^0)(\overline{\mu}_k - m^0)^T, \\
\nu_k &= \nu^0 + \overline{N}_k
\end{aligned} \quad (5)$$

where $\overline{N}_k$, $\overline{\mu}_k$ and $\overline{\Sigma}_k$ are given by equation 4 and α^0, β^0, m^0, W^0 and ν^0 are preset parameters. The EM algorithm iterates until the likelihood of the solution stops improving. Finally, the expectation for the GMM parameters Φ is

$$\mathbb{E}[\pi_k] = \frac{\alpha_k}{\sum_j^K \alpha_j}, \quad \mathbb{E}[\mu_k] = m_k, \quad \mathbb{E}[\Sigma_k] = (\nu_k W_k)^{-1}. \tag{6}$$

A final remark is that K-means and EM algorithms suffer convergence problems when only one point is assigned to a component. However, for the Bayesian-GMM algorithm the convergence is guaranteed as unnecessary components are automatically discarded imposing $\mathbb{E}[\pi_k] \approx 0$ without generating any singularity to the algorithm.

B. Registration Method

GMM provide a continuous representation that lets solve the registration problem by optimization. The optimization variable Ω is the displacement between two overlapping scans defined by a rotation matrix R and a translation vector t. The problem restrictions are keeping Ω in the Lie group defining pose [2]. Two optimization policies are possible:

1) Point to Distribution (P2D): Given a fixed scan represented by a GMM $p(x|\Phi^{\mathscr{F}})$ and a moving scan given in point cloud form $\mathscr{D}^{\mathscr{M}} = \{q_1, \ldots, q_n\}$ whose relative displacement is defined by Ω; the P2D method [7] finds the ML solution of $\mathscr{D}^{\mathscr{M}}$ transformed by Ω into the GMM

$$\Omega^* = \arg\max_{\Omega} p(\mathscr{D}^{\mathscr{M}}|\Phi^{\mathscr{F}}, \Omega) = \arg\max_{\Omega} \prod^{\mathscr{D}^{\mathscr{M}}} \sum^{K} \pi_k \mathscr{N}(x = Rq_d + t|\mu_k, \Sigma_k).$$

In order to solve the ML problem applying gradient-based methods the log-likelihood is minimized and, in order to favour the analytic treatment of derivatives, the logarithm of the Gaussian Mixtures is approximated by another GMM [13]. Therefore, the cost function is $F = -\sum_{i=1}^{\mathscr{D}^{\mathscr{M}}} \sum_{k=1}^{K} f_{ik}$ where

$$f_{ik} = \frac{\pi_k}{(2\pi)^{\frac{D}{2}} \sqrt{|\Sigma_k|}} exp\left(-\frac{1}{2} x_{ik}^T \Sigma_k^{-1} x_{ik}\right) \tag{7}$$

with $x_{ik} = \mu_k - Rq_i - t$, $t \in \mathbb{R}^n$ and $R \in SO(n)$. The rotation matrix R is parameterized using exponential coordinates $\theta \in \mathbb{R}^n$ through the exponential map $R = \exp(\theta)$ of the SO(n) group [14]. Taking the derivatives of the cost function in the composite manifold $\langle \mathbb{R}^n, SO(n) \rangle$ [2], instead of in the SE(n) group, translation is considered independent from rotation and consequently the jacobian vector is

$$\begin{aligned} \frac{\partial F}{\partial t_a} &= -\sum_{i=1}^{\mathscr{D}^{\mathscr{M}}} \sum_{k=1}^{K} f_{ik} x_{ik}^T \Sigma_k^{-1} \frac{\partial t}{\partial t_a}, \\ \frac{\partial F}{\partial \theta_a} &= -\sum_{i=1}^{\mathscr{D}^{\mathscr{M}}} \sum_{k=1}^{K} f_{ik} x_{ik}^T \Sigma_k^{-1} \frac{\partial R}{\partial \theta_a} q_i \end{aligned} \tag{8}$$

for $a = 1, \ldots, D$ where $\frac{\partial t}{\partial t_a}$ corresponds to the a-th vector of the standard basis in $\mathbb{R}^n$ and for the 2D case $\frac{\partial R}{\partial \theta} = \begin{bmatrix} -sin(\theta) & -cos(\theta) \\ cos(\theta) & -sin(\theta) \end{bmatrix}$ and for the 3D case $\frac{\partial R}{\partial \theta_a}$ is given in Result 2 from [15]. The hessian matrix is symmetric and is computed as

$$\begin{aligned} \frac{\partial^2 f_{ik}}{\partial t_a \partial t_b} &= f_{ik}\left(x_{ik}^T \Sigma_k^{-1} \frac{\partial t}{\partial t_b} x_{ik}^T \Sigma_k^{-1} \frac{\partial t}{\partial t_a} - \frac{\partial t}{\partial t_a}^T \Sigma_k \frac{\partial t}{\partial t_b}\right), \\ \frac{\partial^2 f_{ik}}{\partial t_a \partial \theta_b} &= f_{ik}\left(x_{ik}^T \Sigma_k^{-1} \frac{\partial R}{\partial \theta_b} q_i x_{ik}^T \Sigma_k^{-1} \frac{\partial t}{\partial t_a} - \frac{\partial t}{\partial t_a}^T \Sigma_k^{-1} \frac{\partial R}{\partial \theta_b} q_i\right), \\ \frac{\partial^2 g_{ik}}{\partial \theta_a \partial \theta_b} &= f_{ik}\left(x_{ik}^T \Sigma_k^{-1} \frac{\partial R}{\partial \theta_b} q_i x_{ik}^T \Sigma_k^{-1} \frac{\partial R}{\partial \theta_a} q_i \right. \\ &\quad \left. - q_i^T \frac{\partial R}{\partial \theta_a}^T \Sigma_k^{-1} \frac{\partial R}{\partial \theta_b} q_i + x_{ik}^T \Sigma_k^T \frac{\partial^2 R}{\partial \theta_a \partial \theta_b} q_i\right) \end{aligned} \tag{9}$$

for $b = 1, \ldots, D$ where for the 2D case $\frac{\partial^2 R}{\partial \theta^2} = -R$ and for the 3D case $\frac{\partial^2 R}{\partial \theta_a \partial \theta_b}$ is given in Appendix A from [10].

In the NDT algorithm [7] only correspondences with the closest component to each point are established. However, in the proposed algorithm, in order to favour continuity, all possible correspondences are considered, only filtered by a Chi squared test with a 95% of confidence.

2) Distribution to Distribution (D2D): Given two GMM, one modeling a fixed scan $p(x|\Phi^{\mathscr{F}})$ and the other modelling a moving scan $p(x|\Phi^{\mathscr{M}}, \Omega) = p(x|\pi^{\mathscr{M}}, R\mu^{\mathscr{M}} + t, R\Sigma^{\mathscr{M}} R^T)$ whose relative displacement is defined by a pose Ω; the $\mathscr{L}^2$ distance between both distributions is defined as $\mathscr{L}^2 = \int \left(p(x|\Phi^{\mathscr{F}}) - p(x|\Phi^{\mathscr{M}}, \Omega)\right)^2 dx$. Thus, the D2D method [16] finds the displacement that maximizes the $\mathscr{L}^2$ distance between both distributions. Developing the square product of the $\mathscr{L}^2$ definition, only the product between distributions varies through Ω. Using the closed form of this term for the Gaussian distribution given in [16], the cost function has the following analytical form

$$\Omega^* = \arg\max_{\Omega} \sum_{i=1}^{K^{\mathscr{F}}} \sum_{j=1}^{K^{\mathscr{M}}} \pi_i^{\mathscr{F}} \pi_j^{\mathscr{M}} \mathscr{N}(0|\mu_i^{\mathscr{F}} - R\mu_i^{\mathscr{M}} - t, \Sigma_i^{\mathscr{F}} + R\Sigma_i^{\mathscr{M}} R^T).$$

The normalization factor for this distribution depends on the optimization variables Ω, as its covariance matrix is dependent on the rotation matrix. This dependency increases hugely the analytical complexity of the derivatives. In order to reach simplicity, our results show that the approximation of the normalization factor by a constant d_1 proposed in [8] is enough to get good matching results when using acoustic data. Therefore, the cost function is $G = -\sum_{i=1}^{K^{\mathscr{F}}} \sum_{k=1}^{K^{\mathscr{M}}} g_{ik}$ where

$$g_{ik} = \pi_i^{\mathscr{F}} \pi_k^{\mathscr{M}} d_1 exp\left(-\frac{1}{2} y_{ik}^T \eta_{ik} y_{ik}\right) \tag{10}$$

with $y_{ik} = \mu_i^{\mathscr{F}} - R\mu_k^{\mathscr{M}} - t$, $\eta_{ik} = (\Sigma_i^{\mathscr{F}} + R\Sigma_k^{\mathscr{M}} R^T)^{-1}$, $t \in \mathbb{R}^n$ and $R \in SO(n)$. The rotation matrix R is parameterized using exponential coordinates $\theta \in \mathbb{R}^n$. Taking the derivatives of the cost function again in the $\langle \mathbb{R}^n, SO(n) \rangle$, the jacobian vector is

$$\frac{\partial G}{\partial t_a} = -\sum_{i=1}^{K^{\mathscr{F}}} \sum_{k=1}^{K^{\mathscr{M}}} g_{ik} y_{ik}^T \eta_{ik} \frac{\partial t}{\partial t_a}, \tag{11}$$

$$\frac{\partial G}{\partial \theta_a} = -\sum_{i=1}^{K^{\mathscr{F}}} \sum_{k=1}^{K^{\mathscr{M}}} g_{ik} \left(y_{ik}^T \eta_{ik} \frac{\partial R}{\partial \theta_a} \varepsilon_k + y_{ik}^T \eta_{ik} \frac{\partial R}{\partial \theta_a} \Gamma_k R^T \eta_{ik} y_{ik} \right)$$

for $a = 1, \ldots, n$ where $\varepsilon_k = \mu_k^{\mathscr{M}}$, $\Gamma_k = \Sigma_k^{\mathscr{M}}$. The hessian matrix is symmetric and is defined by

$$\begin{aligned}
\frac{\partial^2 g_{ik}}{\partial t_a \partial t_b} &= g_{ik} \left(y_{ik}^T \eta_{ik} \frac{\partial t}{\partial t_b} y_{ik}^T \eta_{ik} \frac{\partial t}{\partial t_a} - \frac{\partial t}{\partial t_a}^T \eta_{ik} \frac{\partial t}{\partial t_b} \right), \\
\frac{\partial^2 g_{ik}}{\partial t_a \partial \theta_b} &= g_{ik} \left(y_{ik}^T \eta_{ik} \left(\frac{\partial R}{\partial \theta_b} \varepsilon_k y_{ik}^T + \frac{\partial R}{\partial \theta_b} \Gamma_k R^T \eta_{ik} y_{ik} y_{ik}^T - \frac{\partial M}{\partial \theta_b} \right) \eta_{ik} \frac{\partial t}{\partial t_a} \right. \\
&\quad \left. - \frac{\partial t}{\partial t_a}^T \eta_{ik} \frac{\partial R}{\partial \theta_b} \varepsilon_k \right), \\
\frac{\partial^2 g_{ik}}{\partial \theta_a \partial \theta_b} &= g_{ik} \left(y_{ik}^T \eta_{ik} \left(\frac{\partial R}{\partial \theta_b} \varepsilon_k + \frac{\partial R}{\partial \theta_b} \Gamma_k R^T \eta_{ik} y_{ik} \right) \right. \\
&\quad y_{ik}^T \eta_{ik} \left(\frac{\partial R}{\partial \theta_a} \varepsilon_k + \frac{\partial R}{\partial \theta_a} \Gamma_k R^T \eta_{ik} y_{ik} \right) - \varepsilon_k^T \frac{\partial R}{\partial \theta_a}^T \eta_{ik} \frac{\partial R}{\partial \theta_b} \varepsilon_k \\
&\quad - y_{ik}^T \eta_{ik} \left(\frac{\partial M}{\partial \theta_b} \eta_{ik} \frac{\partial R}{\partial \theta_a} + \frac{\partial M}{\partial \theta_a} \eta_{ik} \frac{\partial R}{\partial \theta_b} - \frac{\partial^2 R}{\partial \theta_a \partial \theta_b} \right) \varepsilon_k \\
&\quad \left. - \frac{1}{2} y_{ik}^T \eta_{ik} \left(\frac{\partial M}{\partial \theta_b} \eta_{ik} \frac{\partial M}{\partial \theta_a} - \frac{\partial M}{\partial \theta_a \partial \theta_b} + \frac{\partial M}{\partial \theta_a} \eta_{ik} \frac{\partial M}{\partial \theta_b} \right) \eta_{ik} y_{ik} \right)
\end{aligned} \tag{12}$$

for $b = 1, \ldots, n$ and

$$\begin{aligned}
\frac{\partial M}{\partial \theta_a} &= \frac{\partial R}{\partial \theta_a} \Gamma_k R^T + \left(\frac{\partial R}{\partial \theta_a} \Gamma_k R^T \right)^T, \\
\frac{\partial M}{\partial \theta_a \partial \theta_b} &= \frac{\partial^2 R}{\partial \theta_a \partial \theta_b} \Gamma_k R^T + 2 \frac{\partial R}{\partial \theta_a} \Gamma_k \frac{\partial R}{\partial \theta_b}^T + \left(\frac{\partial^2 R}{\partial \theta_a \partial \theta_b} \Gamma_k R^T \right)^T
\end{aligned}$$

where $M = R \Gamma_k R^T$.

In the NDT algorithm [8] only correspondences with the closest component of $\mathscr{M}$ to each component of $\mathscr{F}$ are set. However, in our proposal, in order to favour continuity, all possible correspondences are considered as acoustic data is sparse and Front-Ends act as data compressors.

C. Solver

Gradient methods specialized for the $\langle \mathbb{R}^n, SO(n) \rangle$ are proposed to solve the optimization based on the iterative procedure $\Omega_{j+1} = \Omega_j \oplus \alpha_j d_j$ where $\alpha_j \in \mathbb{R}$ is called step size, $d_j \in \mathbb{R}^n$ is called method direction and $\oplus$ is the sum operator for the $\langle \mathbb{R}^n, SO(n) \rangle$. Given a pose Ω and a perturbation $\eta = (\eta_t, \eta_\theta)$ expressed in the tangent space of $\langle \mathbb{R}^n, SO(n) \rangle$, the $\oplus$ operator is defined as

$$\Omega \oplus \eta = \begin{bmatrix} t + \eta_t \\ R Exp(\eta_\theta) \end{bmatrix} \tag{13}$$

where $Exp(\eta_\theta)$ is the exp map of the SO(n) group [2].

Two gradient methods are implemented. Steep Descent method (Algorithm 11.6 from [17]) sets $d_j = -J_j$ and Newton Method (Algorithm 10.1 from [17]) applies $d_j = -H_j^{-1} J_j$, where J_j and H_j are respectively the first and second derivative of the policy evaluated at Ω_j. Two tools are provided to ensure convergence. A line search algorithm based on the Wolfe conditions (Algorithm 11.5 form [17]) can be used to set an optimal α_j at each iteration. The Gill, Murray and Wright modified Cholesky factorization algorithm (Algorithm MC form [18]) can be applied to invert the hessian matrix H_j. If H_j is an indefinite matrix, this algorithm perturbs it to get a positive definite matrix to guarantee a minimization direction and ensure that the resulting matrix is reasonably well conditioned.

Finally, as it is suggested in [19], H_j gives information about the curvature of the policy at Ω_j. This means that computing H_j at the optimum, it is a measure of the uncertainty of the registration result. In [20] it is suggested that the uncertainty of a match is also related to the noise of the sensor used to build the scan. Nevertheless, in [8] it is empirically proved that both approaches for the NDT-D2D method give the same uncertainty shape, only with different size. Therefore, we conclude that the covariance matrix of a match is proportional to H_j^{-1} computed at the optimum. This constitutes an automatic tool to recover the uncertainty for each particular match according to the optimization process, automatically classifying good matches from those that are not so good and, also, detecting in which direction of Ω_j the solution is better in comparison to the others. This result is not common in many scan matching techniques, as solvers are only based on first derivatives and H_j evaluations are not available. A final remark is that this uncertainty is given in the $\langle \mathbb{R}^n, SO(n) \rangle$, not in the SE(n) group, as derivatives are computed in the composite manifold. However, tools are provided to map them from one space to the other.

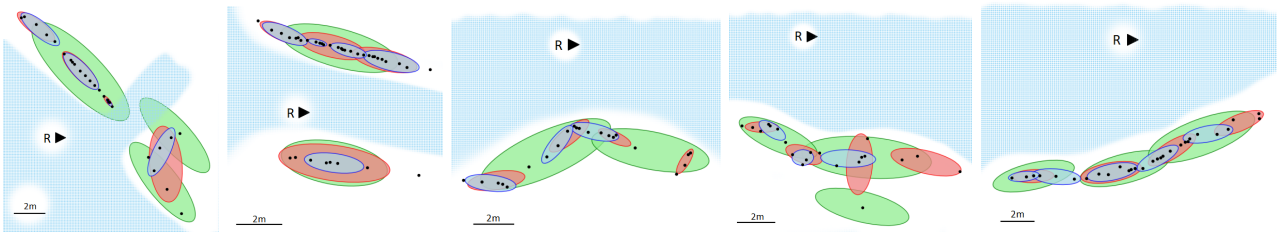
IV. Software architecture

All the proposed algorithms have been implemented as a C++ library called `GMM Registration`[1]. The main class of this library is `GaussianMixturesModel` which stores and plots a GMM. The GMM Font-End is implemented as a library of functions. All registration methods and solvers are implemented as inheritances of two interfaces: `ScanMatchingMethods` and `Solver`. This way, the library can grow modularly adding new methods or solvers as needed. Finally, all classes and functions are templarized to process 2D and 3D point clouds. In the library repository, the documentation and several examples for its use are provided.

V. Results

The proposed techniques have been tested using real data from sea experiments. 2D and 3D tests have been performed.

1) 2D tests: 2D data was gathered using the Sparus II AUV [3] (Fig. 1c) equipped with a Mechanical Scanning Profiling Sonar [4] (Fig. 1d) that builds the scan in combination with the inertial navigator while the AUV is in motion. For the 2D tests, two small data sets were used. In the first one, formed by 8 scans, the AUV followed a trajectory through corridors in Sant Feliu de Guixols harbour (Girona). In the second one, formed by 11 scans, the AUV circumnavigated a natural rocky island called La Galera located at the Cap de Creus Natural Park (Girona). All these scans are formed by an average of 78 points, from 49 to 105 points. In order to reach a similar performance between the four Front-Ends: NDT was tuned with a cell size of $3m$ and a minimum of 3 points to set a component; K-means and EM used 4 components; and EM was initialized with the K-means algorithm to favour convergence. Finally, Bayesian-GMM used an upper bound of 10 components. In order to favour uniformity, all algorithms applied a covariance correction of 0.1, meaning that for each covariance matrix its minimum

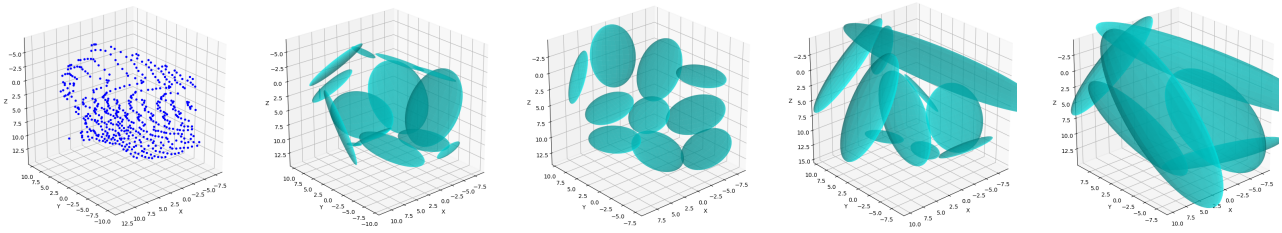


(a) (b) (c) (d) (e)

Fig. 3: Components plots from GMMs obtained by proposed Front-Ends applied on $2D$ individual scans (black dots). Water is represented by light blue dots and $\{R\}$ represents the robot frame. Blue: NDT with 3m cell size and 3 points threshold. Red: EM with 4 components. Green: Bayesian-GMM with an upper bound of 10 components. Real scans acquired in: (a) harbour corner, (b) harbour corridor and (c,d,e) rocky island wall.

TABLE I: Two-dimensional registration comparison on acoustic data.

Method	Translation RMSE [m]	Rotation RMSE [rad]	Convergence rate [%]	Time mean [ms]	Time std [ms]	Mean K [u]	K std [u]
NDT+P2D	0.989	0.145	98	0.46	2.33	4.0	1.0
K-means+P2D	1.130	0.198	96	0.57	2.60	4	-
EM+P2D	1.117	0.221	96	2.07	5.62	4	-
Bayes-GMM+P2D	0.388	0.079	99	0.78	3.06	2.7	0.7
NDT+D2D	0.821	0.117	100	0.53	2.48	3.9	1.0
K-means+D2D	0.838	0.129	100	0.68	3.24	4	-
EM+D2D	0.641	0.118	100	2.10	6.34	4	-
Bayes-GMM+D2D	0.440	0.084	100	1.08	3.61	2.7	0.6



(a) (b) (c) (d) (e)

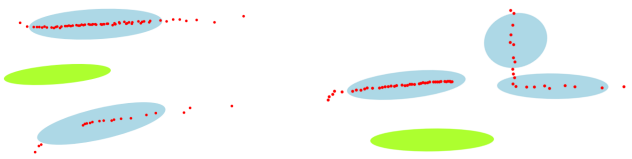
Fig. 4: Components plots from GMMs obtained by proposed Front-Ends applied on a scan perceiving the base of a harbour dock corner. (a) point cloud, (b) NDT with 8m cell size, (c) K-means with 10 components, (d) EM with 10 components and (e) Bayesian-GMM with a upper bound of 30 components.

TABLE II: Three-dimensional registration comparison on acoustic data.

Method	Translation RMSE [m]	Rotation RMSE [rad]	Convergence rate [%]	Time mean [ms]	Time std [ms]	Mean K [u]	K std [u]
NDT+P2D	1.094	0.329	99	152.06	49.29	17.5	2.6
K-means+P2D	1.323	0.479	97	127.76	82.70	10	-
EM+P2D	1.128	0.255	99	581.50	94.50	10	-
Bayes-GMM+P2D	0.852	0.276	99	292.63	97.79	11.3	1.6
NDT+D2D	1.696	0.390	100	57.71	27.83	19.0	1.9
K-means+D2D	1.729	0.405	100	39.71	41.72	10	-
EM+D2D	1.575	0.298	100	506.39	82.69	10	-
Bayes-GMM+D2D	1.533	0.282	100	161.42	62.47	11.4	1.4
Generalized ICP [21]	5.819	0.895	74	185.37	88.95	-	-

eigenvalue is forced to be at least 10% of its maximum eigenvalue. Fig. 3 shows the results of the four Front-Ends over some representative scans where K-means is not shown for clarity, as its results are close to the EM. As it can be seen, Bayesian-GMM is the most flexible algorithm, perceiving the scan morphology and fitting a model with the optimal number of components. Moreover, in Fig. 3d it is shown how this algorithm can reject outliers, setting a low weighted component to a single point to isolate it from the rest of the scan avoiding its involvement to the match.

Using these scans, a comparison between the available scan matching algorithms was performed. As in sea applications ground truth measures are normally not possible, the comparison was performed using the same scan



(a) (b)

Fig. 5: Bayesian-GMM+P2D match of 2D scans. Blue: reference scan in GMM form. Red: matching scan in point cloud form. Green ellipsoid: translation covariance.

applying a random displacement to the entire point cloud, moving it as a whole to generate two misaligned scans. This way, the ground truth was known. Then, the scan matching problem was initialized with zero displacement and the result was compared to the ground truth, computing the Root Mean Squared Error (RMSE) for the translation and the rotation. Every scan of the data set was matched 100 times and all optimizations were solved using the `CholeskyLineSearchNewtonMethod` solver, as it is the most versatile available. Random translations in range of $[-1.0, 1.0]m$ and random rotations in range of $[-0.25, 0.25]rad$ were applied, which cover a range wider than the standard drift that an underwater dead reckoning system can accumulate between two successive scans. Tests were solved in a computer equipped with an Intel i7-9700 CPU with 8 cores running at 3GHz. The scan matching results are given in Table I where it is seen that P2D outperforms D2D method. This is expected as in the P2D only one scan is compressed in a GMM and more data is available to improve accuracy. Looking at the results, the Bayesian-GMM Front-End provides the best performance, reaching the best results for both methods. As it was appointed, in contrast to the other Front-Ends, Bayesian-GMM does not need any previous assumption on the structure of the model, reaching an adaptive GMM that favours scan matching. Different scan morphologies are considered verifying that using a unique K does not allow matching optimally different kind of scenes.

Finally, Fig. 5 shows the uncertainty measure of the register provided by the algorithm on two Bayesian-GMM+P2D matches. The covariance of the registered translation is plotted as a green ellipsoid, showing as expected that the most certain direction of the match is normal to the most informative direction of the point cloud. This is exemplified in the corridor of Fig. 5a where the algorithm returns a match uncertain in the longitudinal direction as there is not any information of the corridor in that direction.

2) 3D tests: 3D data was gathered using the Girona 500 AUV [22] equipped with a Multibeam Profiling Sonar [23] mounted on a Pan & Tilt platform rotating around the robot y direction while keeping the AUV static to avoid distortion. For the 3D tests 5 scans were used, perceiving different views on the basis of the harbour docks. These scans are formed by an average of 5211 points, from 4893 to 5437 points. To reach similar performance between the four Front-Ends: NDT was tuned with a cell size of $8m$ and a minimum of 6 points to set a component; K-means and EM used 10 components; and EM applied the K-means initialization. Bayesian-GMM used an upper bound of 30 components. All algorithms applied a covariance correction of 10%. In Fig. 4 the results of the Front-Ends over a particular scan are shown. Again, Bayesian-GMM fits the most compressed model with 6 components. EM fits a model with a similar structure to the Bayesian-GMM, K-means uniformly distributes all components through the point cloud and NDT returns a GMM of 14 components.

Using this data, a comparison between the available scan matching algorithms was performed following the same scheme used on the 2D comparison but cutting the reference and moving scans in different directions to reduce the scans overlap up to 72%. Random translations in range of $[-2.0, 2.0]m$ and random rotations in range of $[-0.25, 0.25]rad$ were applied. In Table II the scan matching results are given. Again P2D outperform D2D method and the Bayesian-GMM Front-End provides the best performance due to its adaptability capability to different scan morphologies. Moreover, we extended the comparison to the Generalized ICP (GICP) algorithm [21] implemented in [24] as it is a popular probabilistic registration method, a complementary algorithm to the NDT taken as reference in this paper. Results show how our proposal clearly outperforms GICP.

3) Final remarks: P2D method feed with Bayesian-GMM Front-End provides the best accuracy, although, D2D provides the lowest execution time in 3D point clouds, when the matching time is comparable with the fitting time. These results show the effectiveness of the Bayesian-GMM to adapt to different scan morphologies found during AUV operation in real environments. Time measures show that the provided implementation is fast enough to process acoustic data online with the sonar as the AUV spent an average of $7,30s$ and $12.84s$ to build respectively the 2D and 3D scans.

VI. Conclusions

In this paper, a rigid registration methodology specially suited for sonar data has been presented. This methodology provides an uncertainty measure for the match, which is fundamental in Pose SLAM applications, and the Bayesian-GMM is first introduced to automatically learn the optimal GMM structure needed to model a noisy scan. Using this algorithm, scan matching can adapt to different types of noisy scenes. All GMM methodologies have been implemented in C++ and are available as the open source `GMM Registration` library. Results using real data show clearly the effectiveness of the proposal. This work constitutes a more comprehensive contribution in underwater scan matching and provides a tool suitable for a Pose SLAM application for AUV navigation, for instance, in combination with the well known GTSAM library [25].

ACKNOWLEDGMENT

This work was supported by the PLOME Project under the Grant agreement PLEC2021-007525 and by the Spanish Government FPU19/03638 PhD grant (to P. Vial).

REFERENCES

[1] H. Attias, "A variational bayesian framework for graphical models," *Advances in Neural Information Processing Systems*, vol. 12, 2000.

[2] J. Solà, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," *ArXiv*, vol. abs/1812.01537, 2020.

[3] M. Carreras, J. Hernández, E. Vidal, N. Palomeras, D. Ribas, and P. Ridao, "Sparus ii auv - a hovering vehicle for seabed inspection," *IEEE Journal of Oceanic Engineering*, vol. 43, pp. 344–355, 2018.

[4] "Tritech international ltd, super seaking profiler," https://www.tritech.co.uk/media/products/dual-frequency-profiler-super-seaking-dfp.pdf?id=9e4bad8a.

[5] A. P. Dempster, N. M. Laaird, and D. B. Rubin, "Maximum likelihood for incomplete data via the em algorithm," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 39, pp. 1–38, 1977.

[6] P. J. Besl and H. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, pp. 239–256, 1992.

[7] P. Biber and W. Strasser, "The normal distributions transform: a new approach to laser scan matching," *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003)*, vol. 3, pp. 2743–2748, 2003.

[8] T. Stoyanov, M. Magnusson, H. Andreasson, and A. J. Lilienthal, "Fast and accurate scan registration through minimization of the distance between compact 3d ndt representations," *The International Journal of Robotics Research*, vol. 31, pp. 1377–1393, 2012.

[9] M. Jiang, S. Song, F. Tang, Y. Li, J. Liu, and X. Feng, "Scan registration for underwater mechanical scanning imaging sonar using symmetrical kullback-leibler divergence," *Journal of Electronic Imaging*, vol. 28, 2019.

[10] W. Tabib, C. O'Meadhra, and N. Michael, "On-manifold gmm registration," *IEEE Robotics and Automation Letters*, vol. 3, pp. 3805–3812, 2018.

[11] S. Lloyd, "Least squares quantization in pcm," *IEEE Transactions on Information Theory*, vol. 28, pp. 129–137, 1982.

[12] C. Bishop, *Pattern Recognition and Machine Learning*. Springer, 2006.

[13] M. Magnusson, "The three-dimensional normal-distributions transform, an efficient representation for registration, surface analysis and loop detection," Ph.D. dissertation, Örebro University, 2009.

[14] G. Terzakis, M. Lourakis, and D. Ait-Boudaoud, "Modified rodrigues parameters: An efficient representation of orientation in 3d vision and graphics," *Journal of Mathematical Imaging and Vision*, vol. 60, pp. 422–442, 2018.

[15] G. Gallego and A. Yezzi, "A compact formula for the derivative of a 3-d rotation in exponential coordinates," *Math Imaging Vis*, vol. 51, pp. 378–384, 2015.

[16] B. Jiang and B. C. Vemuri, "Robust point set registration using gaussian mixture models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, pp. 1633–1645, 2011.

[17] M. Bierlaire, *Optimization: Principles and Algorithms*. EPFL Press, 2015.

[18] P. Gill, W. Murray, and M. Wright, *Practical Optimization*. Academic Press, 1981.

[19] O. Bengtsson and A. J. Baerveldt, "Robot localization based on scan-matching – estimating the covariance matrix for the idc algorithm," *Robotics ans Autonomous Systems*, vol. 44, pp. 29–40, 2003.

[20] A. Censi, "An accurate closed-form estimate of icp's covariance," *IEEE International Conference on Robotics and Automation*, pp. 3167–3172, 2007.

[21] A. Segal, D. Haehnel, and S. Thrun, "Generalized icp," *Robotics: Science and Systems*, vol. 2, p. 435, 2009.

[22] D. Ribas, N. Palomeras, P. Ridao, M. Carreras, and A. Mallios, "Girona 500 auv: From survey to intervention," *IEEE/ASME Transactions on mechatronics*, vol. 17, pp. 46–53, 2011.

[23] "Imagenex technology corp., 837bxi delta t," https://imagenex.com/assets/images/downloads/837BXi_Delta_T_-Profiling_1000_m_Specs.pdf.

[24] R. Bogdan Rusu and S. Cousins, "Point cloud library," https://pointclouds.org/.

[25] F. Dellaert, "Gtsam 4.0: Factor graphs for sensor fusion in robotics," https://gtsam.org/.