2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

Mobile Robot Localization under Non-Gaussian noise using Correntropy Similarity Metric

Elerson R. S. Santos[1] Marcos A. M. Vieira[1] Gaurav S. Sukhatme[2]

Abstract— **In this paper, we study the localization problem under non-Gaussian noise. In particular, we consider systems that can be represented by a state transition and a measurement component. The state transition indicates how the system evolves given a control variable. The measurement component compares, for a given state, the received and predicted measurements. Here we consider a radio based range sensor which is the primary source of non-Gaussian noise in the system. We solve the problem using a MHE (Maximum Horizon Estimator) with a correntropy similarity metric. Given a time window, the MHE seeks the best set of states that explains the system for the received measurements. Moreover, the main advantage of a MHE is that it allows the re-estimation of past states. Additionally, the correntropy is a similarity metric that, given the amount of error in the estimation, behaves as L2, L1 or L0 norms and has been successfully used in many applications under non-Gaussian noise. We evaluate our proposed method using both simulated and real data. The results show that correntropy is able to work well in comparison with other methods in presence of impulsive noise.**

I. Introduction

Recent advances in robotics are leading to applications that are becoming part of daily life. These include hospitality automated services [1], search and rescue operations [2], and many others. State estimation has an important role in this progress as we can expect noise and unreliable data from sensors, which requires robust methods for state estimation. Here, we focus on the robot localization problem under non-Gaussian noise. We study the problem where a robot needs to localize itself using a periodic range measurement. Furthermore, we consider that the range measurements contain outliers, thus creating an unexpected non-Gaussian distribution.

We study the localization problem under a Maximum Horizon Estimator (MHE) method using the correntropy similarity metric. Given a time window, the MHE is a batch optimization formulation that aims to estimate the best set of states that explain the system dynamics and measurements. Compared to other state estimators, the main advantage of MHEs are their capability of re-estimating past states and dealing with under and over-constrained systems[3]. The correntropy is a robust similarity metric that has been successfully applied in realistic applications [4], [5], [6]. Generally, the correntropy is defined by a kernel and behaves as an L2 norm when the estimated error is small, and L1 and L0 norms as the error increases. Here, we propose to combine the MHE and the correntropy similarity metric, thus creating a state estimator robust to non-Gaussian distributed errors.

State estimation is well studied in the literature. In this sense, the Kalman filter [7] is optimal under the assumption of linear systems and Gaussian distributed noise. However, this assumption does not hold for many real applications. To deal with the non-linearity, we find formulations such as the extended Kalman filter (EKF) [8] and the unscented Kalman filter (UKF) [9]), which use different alternatives to propagate mean and covariance. Meanwhile, we also find KF methods developed considering noise robust methods. In general, we find methods that use different error distributions, such as a t-student [10], or robust statistical methods, such M-estimators methods[11].

From another point of view, we find state estimators based on batch optimization. In [12], the authors have proposed Confusion, a framework for batch estimation implemented over the ceres solver [13]. The proposed system intends to be easy to implement new models, while the framework deals with complex tasks such as state marginalization and the state optimization algorithms. While in [14], the authors have proposed the libRSF, which also gives a framework for state estimation under an MHE implemented over the ceres solver. The libRSF also provides the implementation of some known robust methods from literature. Going towards the direction of a batch estimation, we also find two-state implicit filter (TSIF) [3], which can be seen as MHE of two states while preserving some characteristics of Kalman filters and allowing a previous state to be re-estimated.

The main contributions of this work are the following. We present a state estimation algorithm based on a MHE and correntropy similarity metric. The objective of this algorithm is to provide a robust estimation for the localization problem under non-Gaussian noise. We develop a variable kernel strategy targeting an adaptive and parameter-free algorithm. We compare our proposed strategy with the state of the art of robust state estimation methods. Last, we provide the code and the simulated and real datasets created in the evaluation of our proposed algorithm.

The remainder of this paper is organized as follows. First, we present a related work II. Next, we present an overview of the correntropy metric III, followed by our proposed state estimation methodology IV and we evaluate our algorithm with simulation and real experiment datasets in Section V. Finally, we finish this work with a conclusion in Section VI.

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001. The work was also been supported by grants from CNPq No. 205154/2018-2 and FAPEMIG.

[1] Department of Computer Science – Universidade Federal de Minas Gerais, Belo Horizonte, Brazil. E-mails: {elerson, mmvieira}@dcc.ufmg.br

[2] Department of Computer Science – University of Southern California, Los Angeles, USA. Email: gaurav@usc.edu

II. Related Work

Robust state estimation is still a topic of interest in many areas. In robotics specifically, we find a collection of approaches targeting these topics, which include the Kalman Filters, particle filters and optimization methods. Here, we briefly review some of these works.

The Kalman filter [7] is one of the most known algorithms for state estimation in robotics. Given some conditions, as linearity and Gaussian distributed noise, this filter is optimal. To deal with non linearity, we find a handful of Kalman filter variants, such as the EKF [8], UKF [9]. In general, these filters propagate the mean and covariance of the states estimates employing different techniques, such linearization and sigma point sampling.

Meanwhile, considering Kalman filters under non-Gaussian noise, we find approaches that model these filters with heavy tail distributions, such the t-student[10], and the ones that use robust statistics methods, such the M-estimators [11]. In overview, the M-estimators are robust regression methods that are used as a substitute for least-square formulation. The robustness from these methods come from considering different weights depending on the size of the error. The most known M-estimators is the rubber-estimator, which creates a weight that behaves as a quadratic function with small errors and as a linear one for big errors. Other formulations, such as the dynamic covariance scaling (DCS) [15] and the Dynamic covariance estimation (DCE) [16] fall in this class of M-estimators. The disadvantage of these estimators ranges from difficult parametrization to robustness only to small numbers of outliers. Here, we use the correntropy similarity metric to achieve robustness.

In a recent approach, [17] has proposed a robust Kalman filter based on a reject-detect formulation. In this work, each measurement is associated with an indicator variable modeled with a beta-Bernoulli distribution. In the solution, at each step of the Kalman filter algorithm, the measurements are automatically classified as inliers and outliers using a variational Bayesian inference estimator. This same strategy was also successfully applied in [18], where the authors propose an Error-State Kalman filter.

Meanwhile, the particle filters [19] represent some of the most robust estimators. In this sense, using multiple particles, each representing a state, these filters are able to handle noise with complex error distributions. The main disadvantage of the particle filters are their high computational cost. While from the perspective of batch optimization methods, these filters are not able to re-estimate past states, thus outliers are fully considered under the filter distribution. In [20] the authors use a particle filter to estimate the localization of a robot under line-of-sight and non-line-of-sight between receiving and transmitting devices. In this work, the authors estimate the distribution of the error using Expectation Maximization, modeling the expected error in an online manner.

From another point of view, we find the state estimator methods that are built upon optimization methods. We find these methods under different names, such as graph optimization [21], [14], maximum horizon estimation [12] and batch optimization (MHE) [3]. In this work we use the term MHE considering a batch optimization that occurs in a time window of a given size.

In [3], the authors proposed the two-state implicit filter (TSIF). This filter was introduced as having the properties of being able to be applied in over and under constrained systems. The over-constrained systems occur when we have more than one variable explaining the evolution of a system, one example of such a situation is the angular velocities that can come from an IMU and from the odometer. The under-constrained occurs when the system is not sufficient to explain a given state evolution. An example of a situation is the direction of a planar robot that does not measure its angular velocity, but its orientation can be estimated using pairs of consecutive states. In Kalman filters and particle filters, these can not be treated directly, while in batch optimization methods under and over constrained systems can be associated with residuals and optimized to their best local states.

Under a pose graph estimation, [21] considers optimization where a subset of measurements outliers, thus requiring a robust method for solving the problem. The authors propose a solution of convex relaxation formulation that is solved by semidefinite programming method. However, different from a MHE, which considers a window of time, the solution proposed aims to solve all the poses of the graph, leading to a formulation that does not scale for big problems. Furthermore, the solution proposed degrades with graphs low connectivity, a situation that is expected in the localization problem studied in this work.

III. Correntropy

Correntropy is a measure of similarity between two random variables X and Y, which is defined as [5]:

$$V(X,Y)_\sigma = E[\kappa_\sigma(X-Y)] = \int \kappa_\sigma(x,y)F_{XY}(x,y) \quad (1)$$

Where $\kappa_\sigma(.,.)$ is a kernel function with width parameter $\sigma > 0$ which follows the Mercer's theorem and $F_{XY}(x_i, y_i)$ is the joint distribution function of the variables (X, Y). In this work, we use the Gaussian kernel, which is a common choice for the correntropy and is defined as:

$$\kappa_\sigma(x,y) = \exp\left(\frac{-(x-y)^2}{\sigma^2}\right) \quad (2)$$

Considering a set of random measurements $\{x_i, y_i\}_i^N$ drawn from a distribution $F(X, Y)$. A discrete correntropy metric $V(X, Y)$ is defined as:

$$V(X,Y)_\sigma = \frac{1}{N}\sum_i^N \exp\left(\frac{-(x_i-y_i)^2}{\sigma^2}\right) \quad (3)$$

The robustness associated with the correntropy can be associated with its property of behaving as an L2 norm when the residual between the random variables (X, Y) is close to zero, going to L1 while the residual increases, and eventually approximating an L0 norm. The correntropy metric also has other properties such as, symmetry $V(X,Y) = V(Y,X)$, being bound $0 \leq V(X,Y) \leq V(0,0)$.

IV. Methodology

Here we consider a discrete time system which states are represented by $\boldsymbol{x}_t$. Each estate $\boldsymbol{x}_t$ represents a robot pose (i,e. (x, y, θ) for a 2D pose). Given a previous state $\boldsymbol{x}_{t-1}$ and a control variable u_t, the state transition is represented by a function $f(\boldsymbol{x_{t-1}}, u_t)$ and expects a Gaussian noise $\boldsymbol{v} \sim N(0, \Sigma_v)$. For each state $\boldsymbol{x}_t$, we expect a range measurement y_t, which we represent by a function $h(\boldsymbol{x}_t)$ with Gaussian noise $\boldsymbol{w} \sim N(0, \sigma_w^2)$. This system is represented in the following equations.

$$\begin{aligned} \boldsymbol{x}_t &= f(\boldsymbol{x_{t-1}}, u_t) + \boldsymbol{v} \\ \boldsymbol{y}_t &= h(\boldsymbol{x_t}) + \boldsymbol{w} \end{aligned} \quad . \tag{4}$$

Here, we tackle the problem of robust estimation of the system states. We solve this problem using a Maximum Horizon Estimator (MHE), which targets to estimate the best set of states for a given time window. Here we assume that the primary source of non-Gaussian noise comes from the measurement function. Thus, we model our measurement function with a correntropy similarity metric, which is robust to outliers. Finally, we derive a variable kernel width estimation, targeting improved noise robustness.

A. The localization problem

The system in equation 4 can be represented as residual functions in the form $r_f(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, u_t) = \Sigma_v^{-\frac{1}{2}}(\boldsymbol{x}_t - f(\boldsymbol{x}_{t-1}, u_t))$ and $r_g(\boldsymbol{x}_t, \boldsymbol{y}_t) = \sigma_w^{-1}(\boldsymbol{y}_t - h(\boldsymbol{x}_t))$. From this, we are able to derive a probabilistic optimization problem with the objective of finding, for a given time window n, the set of states $(\boldsymbol{x}_t, .., \boldsymbol{x}_{t-n})$ that maximizes the given likelihood:

$$\underset{\boldsymbol{x}_i, \ldots, \boldsymbol{x}_n}{\arg\max} \; p(\boldsymbol{x}_{prior}) \prod_{t=i+1}^{n} p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}) p(y_t | \boldsymbol{x}_t) \tag{5}$$

Here, we consider that $p(\boldsymbol{x}_t | \boldsymbol{x}_{t-1}; u_t) \sim \exp(-\frac{1}{2} r_f(.)^2)$, $p(y_t | \boldsymbol{x}_t) \sim \exp(-\frac{1}{2} r_g(.)^2)$. The $p(\boldsymbol{x}_{prior}) \sim N(\boldsymbol{x}_{prior}, \boldsymbol{P})$ is responsible for representing the states that are outside the time window. If it is the first state, it would represent the knowledge corresponding to the initial pose.

We can represent Equation 5 in a log-likelihood form, which gives us a least-squares formulation such as:

$$\begin{aligned} \underset{\boldsymbol{x}_i, \ldots, \boldsymbol{x}_n}{\arg\min} \; \frac{1}{2} ||\boldsymbol{x}_i - \boldsymbol{x}_{prior}||_P^2 + \sum_{t=i+1}^{n} \frac{1}{2} ||r_f(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, u_t)||^2 \\ + \frac{1}{2} ||r_g(\boldsymbol{x}_t, \boldsymbol{y}_t)||^2 \end{aligned} \tag{6}$$

B. The localization problem under the correntropy similarity metric

Next, we substitute the least-square of the measurement by a correntropy similarity metric. The correntropy metric $V(X,Y)$ is maximized when $X = Y$, which is the inverse of the least-square formulation. Thus, we use a correntropy induced metric $CIM(X,Y) = \sqrt{V(0,0) - V(X,Y)}$. We also introduce a variable center to our kernel formulation [22], which takes advantage of the MHE formulation to find the best bias parameter c that introduced in the formulation can reduce the overall error. Thus we have the following batch optimization formulated with a correntropy similarity metric:

$$\begin{aligned} \underset{c, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_n}{\arg\min} \; \frac{1}{2} ||\boldsymbol{x}_i - \boldsymbol{x}_{prior}||_P^2 + \sum_{t=i+1}^{n} \frac{1}{2} ||r_f(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, u_t)||^2 \\ + [1 - \exp(-\frac{r_c(\boldsymbol{x}_t, \boldsymbol{y}_t, c)^2}{2\sigma_t^2})] \end{aligned} \tag{7}$$

In Equation 7, we define as $r_c(\boldsymbol{x}_t, \boldsymbol{y}_t, c) = (\boldsymbol{y}_t - h(\boldsymbol{x}_t)) - c$, thus we remove the normalization factor and we add a kernel bias c to the kernel formulation. The normalization will be integrated in the kernel width parameter σ_t^2. We expect that $\frac{1}{2}||r_g(\boldsymbol{x}_t, \boldsymbol{y}_t) - c||^2 \approx [1 - exp(-\frac{r_c(\boldsymbol{x}_t, \boldsymbol{y}_t, c)^2}{2\sigma_t^2})]$ when the residual $r_c(\boldsymbol{x}_t, \boldsymbol{y}_t, c)$ is close to zero. Thus, considering a kernel width $\sigma_t^2 = \sigma_{w_t}^2$, and given that $\exp(\delta) \approx 1 + \delta$, for a small δ, Equation 7 becomes as same as Equation 6, which was directly derived by the likelihood of the proposed state estimation problem.

The problem of Equation 7 is solved using a non-linear least square optimization. In this fashion, given all the states parameters stacked in $\overset{s}{\boldsymbol{x}}$, with its associated Jacobian $J(\overset{s}{\boldsymbol{x}})$ and error $\boldsymbol{e}$ derived from Equation 7. We solve the non-linear least square minimization $\min_{\delta \overset{s}{\boldsymbol{x}}} ||J(\overset{s}{\boldsymbol{x}}) \delta \overset{s}{\boldsymbol{x}} - \boldsymbol{e}||^2$. Where, given a set of iterations, our objective is to find the best increments $\delta \boldsymbol{x}_t$ which will minimize the Equation 7. Considering a Quasi-Newton optimization method, and defining a Hessian matrix $\boldsymbol{H} = J(\overset{s}{\boldsymbol{x}})^T J(\overset{s}{\boldsymbol{x}})$ and $\boldsymbol{b} = J(\overset{s}{\boldsymbol{x}}) \boldsymbol{e}$, the optimal update is defined as $\delta \overset{s}{\boldsymbol{x}} = \boldsymbol{H}^{-1} \boldsymbol{b}$. We define the prior $\boldsymbol{x}_{prior}$ by a marginalization, which can be done using shur complement. This process can be seen with more details in [12].

C. Variable Kernel Width Estimation

To find the best kernel width parameter for our correntropy optimization, here we follow a strategy similar to the one proposed in [4]. In this work, we consider looking for the parameter σ that minimizes the variance between the current and optimal gradients of the minimization problem.

In this fashion, for a given state $\boldsymbol{x}_t$ at a time t we have the minimization gradient represented as:

$$\nabla \boldsymbol{J} = \boldsymbol{A} + \boldsymbol{B} \exp\left(\frac{-r_{c_t}^2(.)}{2\sigma_t^2}\right) \frac{1}{\sigma_t^2} \tag{8}$$

Where, $\boldsymbol{A} = \boldsymbol{r}_{f_t}(.) \frac{\partial r_{f_t}(\boldsymbol{x}_t, \boldsymbol{x}_{t-1}, u_t)}{\partial \boldsymbol{x}_t} + \boldsymbol{r}_{f_{t+1}}(.) \frac{\partial r_{f_{t+1}}(\boldsymbol{x}_{t+1}, \boldsymbol{x}_t, u_{t+1})}{\partial \boldsymbol{x}_t}$ and $\boldsymbol{B} = \frac{\partial r_{c_t}(\boldsymbol{x}_t, \boldsymbol{y}_t, c)}{\partial \boldsymbol{x}_t} \boldsymbol{r}_{c_t(.)}$

Now, considering an optimal state $\mathring{\boldsymbol{x}}_t$, and $\tilde{\boldsymbol{X}}_t = \boldsymbol{x}_t - \mathring{\boldsymbol{x}}_t$. We are looking for the parameter σ_t^2 that minimizes $(\tilde{\boldsymbol{X}}_t - \nabla \boldsymbol{J})^2$, thus we have the following problem:

$$\underset{\sigma_t^2}{\arg\min}\ \tilde{\boldsymbol{X}}_t^2 - 2u\nabla J \tilde{\boldsymbol{X}}_t + (u\nabla \boldsymbol{J})^2 \tag{9}$$

Considering just the part of the formulation that contains σ_t^2, we need to maximize:

$$F(\sigma_t^2) = 2u\nabla \boldsymbol{J}\tilde{X}_{k-1} - (\nabla \boldsymbol{J})^2 \tag{10}$$

From the derivative of 10, we get:

$$\begin{aligned}\frac{\partial F(\sigma_t^2)}{\partial \sigma_t^2} &= \\ \frac{(\boldsymbol{r}_{c_t}(.)^2 - 2\sigma_t^2)}{\sigma_t^5} &\exp\left(\frac{-\boldsymbol{r}_{c_t}^2}{2\sigma_t^2}\right)\left(-\tilde{\boldsymbol{X}}^T\boldsymbol{B} + \nabla \boldsymbol{J}^T\boldsymbol{B}\right)\end{aligned} \tag{11}$$

From Equation 11, when $(\boldsymbol{r}_{c_t}(.)^2 - 2\sigma_t^2) = 0$ and $-\tilde{\boldsymbol{X}}^T\boldsymbol{B} + \nabla \boldsymbol{J}^T\boldsymbol{B} = 0$, the equation is equal to zero. The first solution is a trivial case. Thus we want to find the parameter that solves $-\tilde{\boldsymbol{X}}^T\boldsymbol{B} + \nabla \boldsymbol{J}^T\boldsymbol{B} = 0$.

From Taylor expansion approximation of the function $r_{c_t}(.)$, we know that $\frac{\partial r_{c_t}(.)}{\partial \boldsymbol{x}_t}\tilde{\boldsymbol{X}}$ is the optimal range measurement error $\mathring{r}_{c_t}$. Thus we substitute $\tilde{\boldsymbol{X}}^T\boldsymbol{B}$ by $\mathring{r}_{c_t} r_{c_t}(.)$. Last, $r_{c_t}(.) = \mathring{r}_{c_t} + \eta_t$, with η being the noise, which gives us that $\mathring{r}_{c_t} r_{c_t}(.) = r_{c_t}(.)^2 - \eta_t^2 - \mathring{r}_{c_t}(.)\eta_t$.

Finally, given Equation 10, we solve equation 11 for the parameter σ^2 as:

$$\frac{[r_{c_t}(.)^2 - \eta_t^2 - \mathring{r}_{c_t}\eta_t] - \boldsymbol{A}^T\boldsymbol{B}}{\boldsymbol{B}^T\boldsymbol{B}} = \exp\left(\frac{-r_{c_t}(.)^2}{2\sigma_t^2}\right)\frac{1}{\sigma_t^2} \tag{12}$$

Which gives us:

$$\sigma_t^2 = \frac{-r_{c_t}(.)^2}{2\mathcal{W}\left(\frac{-r_{c_t}(.)^2\left([r_{c_t}(.)^2 - \eta_t^2 - \mathring{r}_{c_t}\eta_t] - \boldsymbol{A}^T\boldsymbol{B}\right)}{2\boldsymbol{B}^T\boldsymbol{B}}\right)} \tag{13}$$

Here, $\mathcal{W}$ represents a Lambert function. To find the solution of equation 13, we approximate the η_t^2 by the σ_w^2 and we set $\mathring{r}_{c_t}\eta_t$ to zero. The approximation $\mathring{r}_{c_t}\eta_t$ follows from the assumption that the noise free measurement is independent from the noise [4].

The proposed solution is an approximation for the optimal parameter of σ_t^2, which means that it is susceptible to error. Furthermore, given $\chi_t = (\boldsymbol{r}_{c_t}(.)^2 - \eta_t^2 - \boldsymbol{A}^T\boldsymbol{B})$, we want Equation 12 to be positive, solution that occurs when $\chi_t > 0$. Thus, considering $\sigma_{new}^2 = \min(\sigma_{est}^2, \sigma_t^2)$, the minimal value between the estimated and current parameters, we use a moving average as follows:

$$\sigma_t^2 = \begin{cases} \alpha\sigma_t^2 + (1-\alpha) * \sigma_{new}^2 & \chi_t > \delta_{err} \\ \sigma_t^2 & otherwise \end{cases} \tag{14}$$

Here, we consider δ_{err} a small constant value, while $\alpha \in [0, 1]$ and is the parameter from the moving average.

V. Evaluation

The proposed methodology was implemented using the Ceres solver[13] over the framework for state estimation provided by the libRSF [14]. We evaluate the proposed method using both simulated and real experiments. Furthermore, we compare the proposed algorithm with robust state estimation algorithms provided by the libRSF. Here we compare the algorithms, Sum of Gaussian Mixtures (SM), DCS, DCE, Gaussian. The code and the data used to evaluate the algorithms are made avalaible [1]. We consider all experiments a window size of 30 states. Except for the labyrinth dataset, which, to compare with results from the literature, we use 200 states. All the results are shown with 90% of confidence.

A. Simulated Experiments

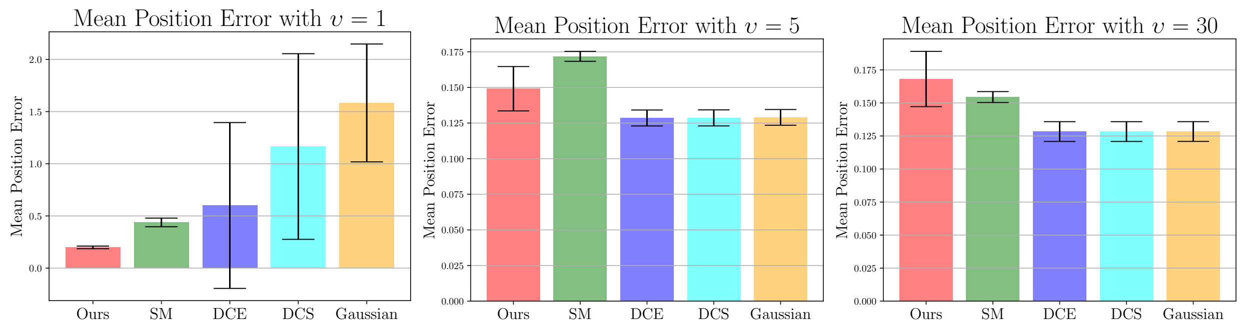
The data for the simulated experiments were generated using ROS[2] and the stage simulator. The environment size was configured with 30x30 meters. Stage does not provide radio range measurement sensors. Thus we have simulated sensors that return the sensor distance with a given noise. The simulated devices provided range measurements at 10 Hz frequency. We have created 20 simulations for each error distribution parameter. In each simulation, the robot executes a square path. Each experiment has 60 seconds. The error in the sensor follow the distributions:

1) A Gaussian mixture distribution $(1-\alpha_e)\mathcal{N}(0, \sigma_w^2) + (\alpha_e)\mathcal{N}(0.2, 100 * \sigma_w^2)$. Here, α_e, represents the amount of each Gaussian distribution in the mixture.
2) A t-Student distribution in the form $\left(\frac{\Gamma(\frac{\nu+1}{2})}{\sqrt{\nu\pi}\Gamma\left(\frac{\nu}{2}\right)}\left(1 + \frac{x^2}{\nu}\right)^{-\frac{\nu+1}{2}}\right) \times \sigma_w$. Here, when $\nu \to \infty$, the proposed distribution approximates to $\mathcal{N}(0, \sigma_w^2)$.

Figures 1 show the result of the experiments considering three values for the t-Student distribution ($\nu = 1$, $\nu = 5$ $\nu = 30$). The results show that the proposed correntropy method was the most robust of the tested algorithms when $\nu = 1$, followed by a mixture of Gaussians. In this situation, we have the t-Student probability distribution the most distant from a Gaussian distribution, presenting impulsive noise. For $\nu = 5$, we see that the proposed method has a higher mean error compared with DCS, DCDE, and Gaussian. When $\nu = 30$, the distribution approaches a Gaussian distribution, the Gaussian method presents the best estimate. In overview, we see that the proposed method is robust to impulsive outliers.

Figure 2 shows the results for the experiment with a Gaussian Mixture Model. Here we expect that the Gaussian mixture method to have the best estimate compared with the other algorithm. The methods compared are not designed to deal with multi-modal distributions. Thus we see that their performance degrades as the parameter α of the mixture increases. Overall, the methods DCS, DCE, and ours, had similar behavior, increasing the error with the parameter α.

[1] https://github.com/verlab/2020-iros-elerson-localization-code
[2] https://www.ros.org/



(a) $\nu = 1$ (b) $\nu = 5$ (c) $\nu = 30$

Fig. 1: Figure showing the results for the t-Student range sensor noise. Here we use $\nu = 1$, $\nu = 5$, $\nu = 30$

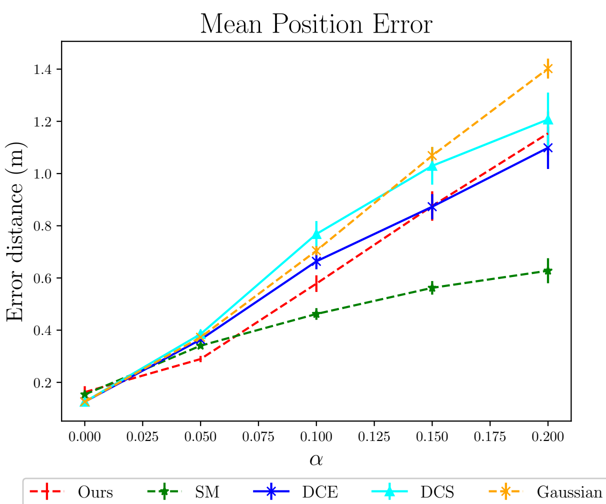


Fig. 2: Simulated experiment with a Gaussian mixture.

B. Real Experiments

For real experiments, we evaluate our algorithm using two datasets, one provided by the authors of [14] and the other created in this work. Next, we show the description and the evaluation of each dataset.

1) Labyrinth Dataset: The labyrinth dataset was developed to have a high level of noise caused by the non-line of sight between the receiver and transmitter devices. Furthermore, the walls of the labyrinth reflect the signal, thus increasing the amount of uncertainty in the data. The data are captured by a robot that moves throughout the environment.

Table I shows the results for the algorithms in the labyrinth dataset. This table shows the average error, the maximum error, and the execution time for each algorithm evaluated. In this table, we can see that our algorithm can achieve the same level of accuracy as the Gaussian mixture. On the other hand, the computational cost for our algorithm is more significant.

2) Square Path Dataset: Here we created a UWB dataset with a non-holonomic robot moving in a square pattern in a closed environment. In this environment, noise in the received signal appears when the robot moves close to the walls. We have used a Kobuki robot mounted with two Decawaves DW 1000 Ultra-Wideband receiving devices, and each device was positioned under known distances in the x-axis of the robot (+-20 cm). We notice that we consider this displacement in the measurement model. Furthermore, the environment was configured with 6 anchors. Figure 3, shows the environment and the robot. Considering that each sensor is able to receive four measurements at the same time, with a 10 Hz frequency, we split the data into eight datasets. For a given time t, all the datasets have the same odometry with different measurements. The objective here was to isolate the process noise from the measurement noise. Thus we can properly evaluate the proposed methodology. The ground truth for the dataset was captured using an Optitrack localization system [23].

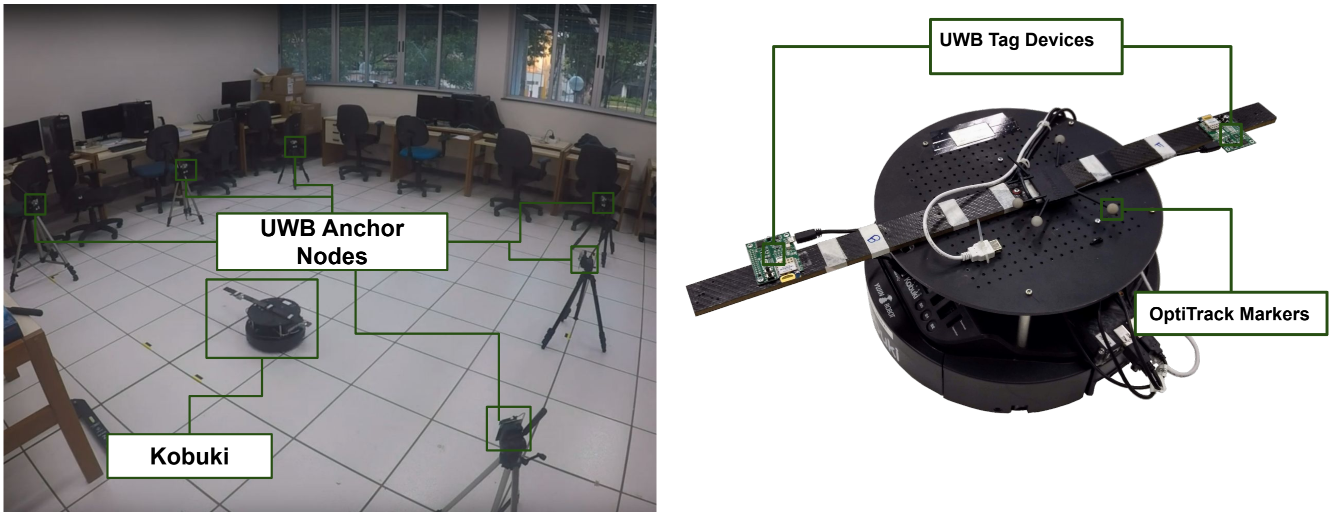
Algorithm	Avg (Meters)	Max (Meters)	Time
DCS	0.1443	1.1605	1m36.033s
DCE	0.1197	0.4519	1m29.912s
Gaussian	0.1211	0.41658	1m04.074s
SM	0.0656	0.369	2m00.522s
Proposed	0.0647	0.2208	3m10.520s

TABLE I: Labyrinth Results

Figure 4 shows the results for the evaluated algorithms with a confidence interval of 90%. The proposed algorithm presented the mean error distance of 0.09 meters, and with a statistical difference to the other methods. It is important to note that we could expect the mixture of Gaussian methods to improve as the number of states were increased. But our dataset is small. Thus we evaluate it with a small number of states.

VI. Conclusion

In this work, we have studied the integration of a maximum horizon estimator (MHE) with the correntropy similarity metric. In the proposed solution, we take advantage of the MHE structure to add a variable center for the correntropy. Furthermore, we were able to develop a variable kernel width, which can tackle the problem of kernel width estimation. We have evaluated the proposed solution in both simulation and real experiments. The results show that correntropy can be integrated into state estimation algorithms, providing increased robustness to real applications.



(a) (b)

Fig. 3: (a) Environment and (b) Robot, used to create a UWB dataset.

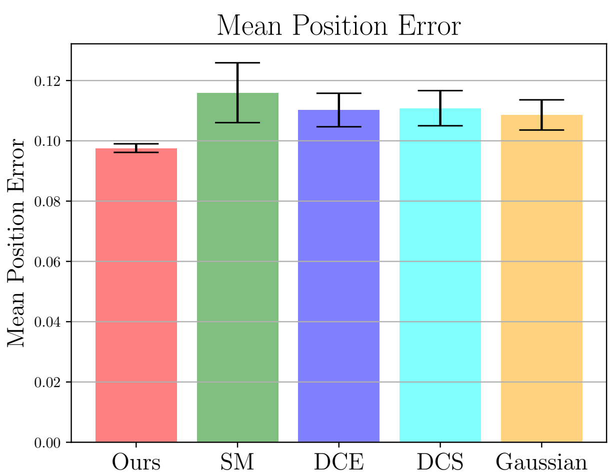


Fig. 4: Results from the real dataset square.

REFERENCES

[1] S. Ivanov, C. Webster, and K. Berezina, "Adoption of robots and service automation by tourism and hospitality companies," 06 2017.

[2] R. R. Murphy, *Disaster Robotics*. The MIT Press, 2014.

[3] M. Bloesch, M. Burri, H. Sommer, R. Siegwart, and M. Hutter, "The two-state implicit filter recursive estimation for mobile robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 573–580, Jan 2018.

[4] L. Shi, H. Zhao, and Y. Zakharov, "An improved variable kernel width for maximum correntropy criterion algorithm," *IEEE Transactions on Circuits and Systems II: Express Briefs*, 11 2018.

[5] W. Liu, P. P. Pokharel, and J. C. Principe, "Correntropy: Properties and applications in non-gaussian signal processing," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5286–5298, Nov 2007.

[6] L. Chen, H. Qu, J. Zhao, B. Chen, and J. C. Principe, "Efficient and robust deep learning with correntropy-induced loss function," *Neural Comput. Appl.*, vol. 27, no. 4, p. 1019–1031, May 2016.

[7] R. E. Kalman, "A new approach to linear filtering and prediction problems," *ASME Journal of Basic Engineering*, 1960.

[8] G. Welch and G. Bishop, "An introduction to the kalman filter," Chapel Hill, NC, USA, Tech. Rep., 1995.

[9] E. A. Wan and R. V. D. Merwe, "The unscented kalman filter," in *Kalman Filtering and Neural Networks*. Wiley, 2001, pp. 221–280.

[10] Y. Huang, Y. Zhang, N. Li, Z. Wu, and J. A. Chambers, "A novel robust student'st-based kalman filter," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 53, no. 3, pp. 1545–1554, June 2017.

[11] P. Huber, J. Wiley, and W. InterScience, *Robust statistics*. Wiley New York, 1981.

[12] T. Sandy, L. Stadelmann, S. Kerscher, and J. Buchli, "Confusion: Sensor fusion for complex robotic systems using nonlinear optimization," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1093–1100, April 2019.

[13] S. Agarwal, K. Mierle, and Others, "Ceres solver," http://ceres-solver.org.

[14] T. Pfeifer and P. Protzel, "Expectation-maximization for adaptive mixture models in graph optimization," in *Proc. of Intl. Conf. on Robotics and Automation (ICRA)*, 2019.

[15] P. Agarwal, G. D. Tipaldi, L. Spinello, C. Stachniss, and W. Burgard, "Robust map optimization using dynamic covariance scaling," in *2013 IEEE International Conference on Robotics and Automation*, May 2013, pp. 62–69.

[16] T. Pfeifer, S. Lange, and P. Protzel, "Dynamic covariance estimation — a parameter free approach to robust sensor fusion," *2017 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, pp. 359–365, 2017.

[17] H. Wang, H. Li, J. Fang, and H. Wang, "Robust gaussian kalman filter with outlier detection," *IEEE Signal Processing Letters*, vol. 25, no. 8, pp. 1236–1240, Aug 2018.

[18] S. Piperakis, D. Kanoulas, N. G. Tsagarakis, and P. Trahanias, "Outlier-robust state estimation for humanoid robots*," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nov 2019, pp. 706–713.

[19] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, 2005.

[20] A. Prorok and A. Martinoli, "Accurate indoor localization with ultra-wideband using spatial models and collaboration," *The International Journal of Robotics Research*, vol. 33, no. 4Prorok, pp. 547–568, 2014.

[21] L. Carlone and G. C. Calafiore, "Convex relaxations for pose graph optimization with outliers," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1160–1167, 2018.

[22] B. Chen, X. Wang, Y. Li, and J. C. Principe, "Maximum correntropy criterion with variable center," *IEEE Signal Processing Letters*, vol. 26, no. 8, pp. 1212–1216, Aug 2019.

[23] NaturalPoint, "Motion capture systems - optitrack," http://optitrack.com/, feb 2019.