



HAL
open science

2D SLAM Correction Prediction in Large Scale Urban Environments

Zayed Alsayed, Guillaume Bresson, Anne Verroust-Blondet, Fawzi Nashashibi

► **To cite this version:**

Zayed Alsayed, Guillaume Bresson, Anne Verroust-Blondet, Fawzi Nashashibi. 2D SLAM Correction Prediction in Large Scale Urban Environments. ICRA 2018 - International Conference on Robotics and Automation 2018, May 2018, Brisbane, Australia. hal-01829091

HAL Id: hal-01829091

<https://inria.hal.science/hal-01829091>

Submitted on 3 Jul 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

2D SLAM Correction Prediction in Large Scale Urban Environments

Zayed Alsayed*, Guillaume Bresson†, Anne Verroust-Blondet* and Fawzi Nashashibi*

*Inria Paris
Paris, France
firstname.name@inria.fr

†Institut VEDECOM
Versailles, France
firstname.name@vedecom.fr

Abstract—Simultaneous Localization And Mapping (SLAM) is one of the major bricks needed to build truly autonomous mobile robots. The probabilistic formulation of SLAM is based on two models: the motion model and the observation model. In practice, these models, together with the SLAM map representation, do not model perfectly the robot’s real dynamics, the sensor measurement errors and the environment. Consequently, systematic errors affect SLAM estimations.

In this paper, we propose two approaches to predict corrections to be applied to SLAM estimations. Both are based on the Ensemble Multilayer Perceptron model. The first approach uses successive estimated poses to predict the errors, with no assumptions on the underlying SLAM process or sensor used. The second method is specific to 2D likelihood SLAM approaches, thus, the likelihood distributions are used to predict the corrections, making this second approach independent of the sensor used. We also build a hybrid correction module based on successive estimated poses and the likelihood distributions.

The validity of both approaches is evaluated through two experiments using different evaluation metrics and sensor configurations.

I. INTRODUCTION

SLAM simultaneously estimates a mobile robot’s pose (position and orientation) and incrementally builds a map of its surroundings.

The theoretical formulation of the SLAM concept led to establishing a solution that is subdivided into two recursive steps. The first is the time update step, where the state is propagated over an elapsed time using a *motion model* in order to build a *prediction*. The second step, the measurement update, consists in integrating the sensor observations into the prediction through an *observation model* in order to compute an *estimation*.

Due to the non-linearity of the SLAM solution, SLAM methods suffer from systematic errors related to the linearization of its models. Work on overcoming these errors has been carried out, but has usually focused on assisting KF-like based SLAM approaches. In contrast, we aim to improve the accuracy by estimating a correction to be applied to the SLAM output based on relevant information available from the SLAM algorithm.

In this paper, we propose two approaches. The first approach is designed for 2D SLAM methods, i.e. independently of the underlying SLAM process and sensor used, where we aim to reduce the errors due to the approximations in dynamic

modeling. The second approach is designed to handle errors related to the SLAM map model and the sensor measurement errors based on likelihood distributions, which makes it suitable for 2D Likelihood SLAM methods (i.e. no assumptions on the sensor used). Both approaches are based on the Ensemble Multilayer Perceptron (EMLP) which is a Supervised Artificial Neural Network model. In such a model, an off-line learning phase is required to build an operational model. Once the model has been built, it can be used for correction prediction with negligible execution time.

The rest of this paper is organized as follows: Section II presents methods from the state-of-the-art to overcome systematic SLAM errors. Section III highlights the impact of errors on incremental localization methods and motivates our evaluation metric choice. Section IV presents our methods to predict corrections for 2D SLAM approaches and 2D Maximum Likelihood SLAM approaches. Finally, Section V presents the results of our experiments and we conclude by giving some perspectives in Section VI.

II. STATE OF THE ART

In its probabilistic form [1][2], the SLAM problem is solved via two recursive steps. The first solutions were based on Kalman Filtering (KF) and KF-like processes. The Kalman Filter [3] assumes a linear motion and observation models together with Gaussian error distributions. In spite of the validity of the process to solve SLAM, the performances obtained deteriorate rapidly because of the linearity of the models, which does not reflect the reality.

To overcome these issues, the filter was extended to non-linear models by approximating them using first order Taylor expansion, which results in the well known Extended Kalman Filter (EKF) [4]. Then, the Unscented Kalman Filter (UKF) [5] uses a set of well defined candidates (named sigmas) around the predicted position which allows us to avoid calculating the Jacobian matrices.

The Particle Filter (PF)-based SLAM [6][7][8] uses a large set of candidates (particles) in order to cover all the possibilities in the navigation space.

To improve the accuracy of SLAM algorithms, the authors of [9][10] combine more robust algorithms and apply a bundle adjustment to refine the estimations of an on-line SLAM approach which gives less accurate results. Another possibility

is to use other sensor information, such as in [11][12] where visual data are mainly used to propagate the estimations more frequently; the estimations are then refined using laser data.

Another line of research aims to improve SLAM accuracy by considering the errors due to the model’s linearization. Indeed, [13] includes a measurement noise estimator to assist the SLAM algorithm which is based on an Extended Kalman Filter. M.Choi et al. [14] use a neural network aided Extended Kalman filter in order to perform SLAM. The Kalman Filter is used for both the estimation of the mobile robot state and the learning of the neural network weights. The same approach was extended by using another type of neural network in [15]. Similar work is presented in [16] and [17] where they both propose to enhance filter-based SLAM with neural networks. K.Choi et al. [16] use a Multilayer Perceptron (MLP) neural network with an EKF-SLAM. In contrast, Panah et al. [17] use a Radial Basis Function (RBF) neural network together with a UKF-SLAM approach.

However, the experiments in these papers are validated only on simulation data and are dedicated exclusively to KF-like SLAM approaches.

Similar approaches, which couple neural networks to KF, were proposed to Fuse INS with GPS data [18][19][20][21]. In [20] the neural network was mainly used to predict INS errors when GPS is not available. The KF performs the fusion of INS with GPS or INS with the predicted correction when there is no GPS information available. Although such an approach is interesting, it has a major drawback in that the predicted errors are re-injected as inputs for the next error prediction, which results in a system which amplifies the errors caused by prediction and leads the whole approach to diverge, as can be seen in the experiments.

In this paper, we propose two approaches to predict systematic errors affecting SLAM estimations. The first approach is suitable for 2D SLAM approaches independently of the underlying estimation process and sensor used. The second approach is suitable for 2D Maximum Likelihood SLAM techniques independently of the sensor used. Both approaches are based on Ensemble MLP neural network models and make use of relevant information available from SLAM without additional processing.

III. ERROR PROPAGATION AND SLAM EVALUATION

A review of localization approaches, specifically those based on SLAM, highlights the difficulty of evaluating the reliability of the method in large-scale environments.

Measuring the drift by comparing the estimated localization with the ground truth could be fair when using a global positioning approach. This, however, is not the case for raw SLAM approaches (i.e. based only on exteroceptive sensors data and without a GNSS receiver) which operate in an incremental way. Indeed, the models used in SLAM algorithms (e.g. the non-linearity of motion and observation models in KF-based SLAM approaches, or discretization and map representation for grid-based SLAM approaches), which are largely induced

by angular errors, have a strong impact on the divergence of the global localization. The impact of linear and angular drifts is illustrated in Figure 1.

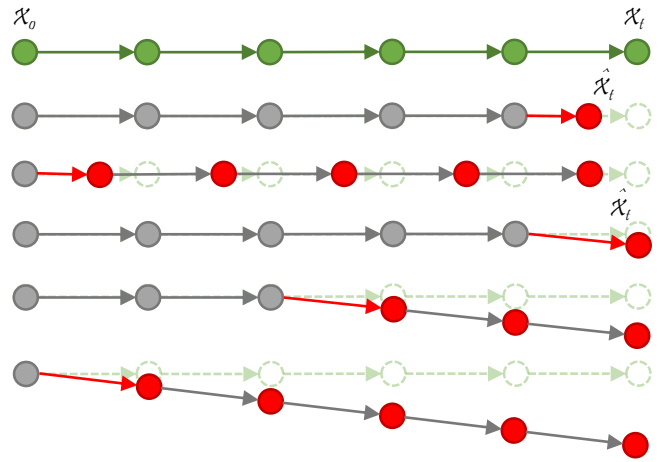


Fig. 1. Impact of linear and angular drift on incremental localization approaches. In green, the ground truth of trajectory. Below, examples of linear and angular drift (red arrow) occurring at different levels on the estimated path.

Burgard et al. [22] and Kummerle et al. [23] compare SLAM approaches and measure their accuracy based on differences in relative poses. For this purpose the data has to be synchronized and must have a high level of accuracy. Making use of the same metrics, Geiger et al. [24] perform the comparison at fixed travel lengths, which reduces the reliance on the synchronization and accuracy of the ground truth. However, measuring errors at the frame level allows us to isolate errors at the frame where they occur. Consequently, this metric is more convenient for building the correction prediction that we present in Section IV.

Navigation in a 2D space implies a state vector \mathbf{x}_t comprising the position (x_t, y_t) and the heading θ_t at the current time t . Thus the drift affecting the localization estimation could be due to errors in position and/or in heading.

In practice, considering errors on each axis independently (i.e. the error ϵ_{x_t} which affects the position following the x axis independently of the error ϵ_{y_t} on the y axis) implicitly involves a high correlation to the heading and its errors in both position components. In contrast, by considering linear-displacement errors ϵ_{d_t} independently of the angular errors ϵ_{θ_t} , reduces the correlation.

We choose to use similar metrics to those in [23], however, we separate translation errors from angular errors (as proposed in [24], but we apply it at the frame level), which results in two elementary values, $(\epsilon_{d_t}, \epsilon_{\theta_t})$ respectively, instead of combining them into a single measure. Let $\delta_{t,t-1} = \mathbf{x}_t \ominus \mathbf{x}_{t-1}$ be the relative transformation which transforms the ground truth pose \mathbf{x}_t into \mathbf{x}_{t-1} by means of the inverse operator of the standard motion composition \ominus , and accordingly for the estimated SLAM path $\hat{\delta}_{t,t-1} = \hat{\mathbf{x}}_t \ominus \hat{\mathbf{x}}_{t-1}$. where \mathbf{x}_t denotes the ground truth pose and $\hat{\mathbf{x}}_t$ the estimated pose.

$$\epsilon_{d_t} = \text{lin}(\delta_{t,t-1} \ominus \hat{\delta}_{t,t-1}) \quad (1)$$

$$\epsilon_{\theta_t} = \text{rot}(\delta_{t,t-1} \ominus \hat{\delta}_{t,t-1}) \quad (2)$$

where $\text{lin}(\cdot)$ and $\text{rot}(\cdot)$ are used to separate the relative translational and rotational components respectively.

IV. SLAM CORRECTION PREDICTION

Predicting systematic errors affecting SLAM estimations requires isolating these errors at the same frame where they occur. Our approach predicts a correction that we systematically apply to the output of SLAM in order to reduce the raw SLAM estimation errors. The general block diagram of SLAM together with the correction estimation is illustrated in Figure 2.

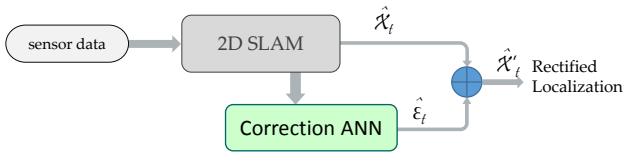


Fig. 2. Block diagram of Correction Prediction algorithm together with SLAM

The corrected estimation $\hat{\mathbf{x}}'_t$ is calculated using Equation 3 by combining the raw SLAM estimation $\hat{\mathbf{x}}_t$ and the predicted correction $\hat{\epsilon}_t$.

$$\hat{\mathbf{x}}'_t = \hat{\mathbf{x}}_t \oplus \hat{\epsilon}_t \quad (3)$$

Our approach relies on an Ensemble MLP model [25][26] to predict the corrections $\hat{\epsilon}_t$. In such a model, an off-line training phase is required to build an operational model. Once the model has been built, it can be used for correction prediction with negligible execution time. The important feature of Artificial Neural Networks (ANN) comes from their ability to learn, memorize and map relations for a set of inputs to expected outputs [27].

We use two kinds of information available from the SLAM process: 1. Successive estimated poses (c.f. Section IV-B). 2. The likelihood distribution (c.f. Section IV-C).

Accordingly, the Correction Prediction Module is denoted:

- **+dS** when using successive SLAM poses as input.
- **+ML** when using the likelihood distribution information.
- **+dSML** when the correction prediction involves both successive SLAM poses and the likelihood distribution.

Consequently, depending on the information exploited, our approach could be suitable for 2D SLAM techniques or for Maximum Likelihood SLAM approaches.

A. The Artificial Neural Network Model

An Ensemble MLP [27][25] network is formed of multiple MLP models organized in parallel with their respective outputs combined in order to produce an improved model [26]. Assuming that each MLP model may make different errors, the idea is to pool together the results from all the MLP models to

find a composite system that outperforms any individual (base) MLP. In contrast to MLP, Ensembles solve local minima and parameter selection for optimal performance.

Moreover, the architecture of our model is of the form:

$$\xi \times [\mathbf{I} \times \lambda \times \mathbf{2}]$$

where the number 2 refers to the number of outputs (i.e. predicted translation and rotation error components), λ is the number of neurons in the hidden layer, ξ the size of the Ensemble (i.e. the number of individual MLPs in the network), and \mathbf{I} denotes the number of inputs which depends on the information used to feed the model. The parameters λ and ξ are both determined experimentally.

B. Correction Prediction for 2D SLAM

Consecutive pose estimations provide information about the velocity profile of the mobile robot, and more specifically about the small oscillations that are mainly due to the linearization and/or discretization of the observation model. Taking the example of a 2D SLAM approach which uses a grid map, such as PML-SLAM [28], the estimated path alternately deviates (oscillating) from the real path. These oscillations are directly related to the measurement errors affecting the sensor observation and the discretization of the map. Consequently, this affects the probability of making an observation given a specific position on the environment map, which represents the measurement model.

Furthermore, the sign and the amplitude of errors on heading and linear displacement may change according to the velocity profile. For example, we observed that the heading error is amplified during a turning maneuver. In addition, the sign of heading errors changes according to the direction in which the turn is made. These errors are induced from the motion model, which describes a probability of a displacement when applying a command input given the previous pose. Therefore, the amplitude and sign of errors are related to the dynamic modeling considered in the motion model.

Our approach to predict linear and angular drifts ($\hat{\epsilon}_{d_t}, \hat{\epsilon}_{\theta_t}$) makes use of a limited number \mathbf{n} of the previous raw SLAM displacements $\{\hat{\delta}_{t,t-1}, \hat{\delta}_{t-1,t-2}, \hat{\delta}_{t-2,t-3}, \dots\}$ where each displacement $\hat{\delta}_{t,t-1}$ is represented by two components $\{\text{lin}(\hat{\delta}_{t,t-1}), \text{rot}(\hat{\delta}_{t,t-1})\}$. Therefore, $\mathbf{I} = 2 \times \mathbf{n}$ with \mathbf{n} determined experimentally.

C. Correction Prediction for 2D Likelihood SLAM

Likelihood SLAM approaches incorporate the sensor observations in pose estimation by finding the candidate which maximizes the likelihood probability of an observation on the map. This is done by exploring the candidate space in which the robot evolves (i.e. the map seen from a given pose).

The likelihood distribution over candidate space reflects how distinctive (Figure 3(b)) and how ambiguous (Figure 3(c)) the situation could be for the SLAM algorithm. Some examples of the likelihood distributions from PML-SLAM [28] implementation are illustrated in Figure 3.

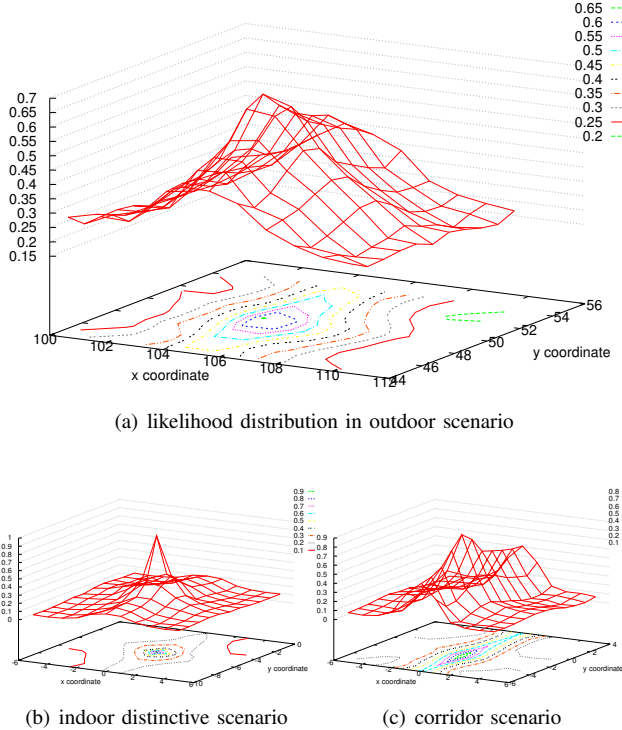


Fig. 3. Likelihood distribution over position candidates for different scenarios. In practice, for a 2D environment, the candidate space is of three dimensions as it includes the orientation as well as the position. For this reason, we represent the best likelihood probability over all orientation values at a given position.

In Figure 3(b) the distribution clearly suggests a unique maximum with a significant difference over the entire candidate area. However, depending on the environment map, the velocity profile and the likelihood measure, the likelihood distribution shape changes (cf. Figure 3(a)). Thus, the simple choice of the pose which maximizes the likelihood over the distribution is subject to errors. These errors are induced by the approximations in the observation model, the map representation, and the observation measurement errors. In contrast, in Figure 3(c) the distribution indicates the presence of the mobile robot in a corridor scenario where multiple solutions could be possible. Hence, the simple choice of the candidate that has the maximum score may not indicate the real position.

The likelihood distributions could contain important information about the estimated pose uncertainty involved in the Maximum Likelihood process. In our work, we aim to exploit this information to predict a correction for Likelihood 2D SLAM approaches. For this purpose, and in order to use the maximum amount of information, we use the full likelihood distribution which is expressed in a 4D space (i.e. composed of the three pose components and the likelihood probability). However, this will include a lot of useless information which, on the one hand, involves extra training time because of the important length of the input vector and, on the other hand, misleads the ANN model from processing the useful information. For these reasons, we transform the full likelihood distribution

to another space by applying Principal Component Analysis (PCA) [29][30] on its probability values. This transformation leads to re-projecting the likelihood probabilities in another space of the same dimension as the initial one, except that the relevant information in the new space (i.e. PCA space) could be reconstructed from a smaller number of axes. Therefore, the information size is reduced to the number of relevant axes in the PCA space.

In practice, taking the example of a 2D likelihood SLAM approach, such as PML-SLAM [28], the candidate space is naturally discretized because of the map representation and the likelihood process. The approach explores an area of 13 candidate positions, each of which is associated to 65 possible heading angles, which results in a total of 845 candidates. Applying PCA leads to transforming the full likelihood distribution from the space of the 845 dimensions to another space of the same size but where the relevant information could be reconstructed through only 24 components. Therefore, the input size of the corresponding correction model $\mathbf{I} = 24$.

D. Correction Prediction for 2D Likelihood SLAM: a Second Method

The information presented in subsection IV-B is generic and so it can be used together with the likelihood distribution to build another correction prediction model suitable for 2D Likelihood SLAM approaches. The input vector will be the sum of the previous displacements together with the reduced full likelihood distribution transformed into PCA space.

V. DATASETS, EXPERIMENTS AND RESULTS

A. Datasets

Our approach is based on supervised machine learning, which means an off-line learning phase is required to build an operational model. Several datasets suited to SLAM benchmarking are available on-line. Bresson et al. [31] give a summary of existing benchmarks. We chose to work mainly with the KITTI dataset [32] to train our models. In addition, we test the models on data acquired using our platforms (see Figure 4).



Fig. 4. VEDECOM Institute experimental platform. The car is equipped with 5 IBEO-Lux laser scanners with a high resolution IMU fused with an RTK-GPS receiver.

The KITTI dataset covers a variety of scenarios mainly collected in urban environments. In addition, the ground truth is synchronized and of high accuracy. It initially contains a total of 54 sequences sorted according to the categories: city, residential and road environments. We eliminate 4 tracks after analyzing the ground truth quality, we choose 4 tracks randomly for testing, and we keep 46 sequences for training. The names of the eliminated tracks are : *city: 2011 09 26 drive 0013 sync; city: 2011 09 26 drive 0093 sync; road: 2011 10 03 drive 0042 sync; road: 2011 10 03 drive 0047 sync.*

The selected Test sequences from the KITTI dataset are named as follows: { **city:** *2011 09 26 drive 0117 sync*, **residential:** *2011 09 26 drive 0064 sync*, **residential:** *2011 09 30 drive 0020 sync*, **road:** *2011 09 29 drive 0004 sync*, } that we tag from Track 1 to 4 respectively.

Then, in order to validate our approach under realistic and more complex conditions where the underlying sensor characteristics are different (i.e. the resolution and the frequency), we acquire data from a circuit around *Bordeaux Lake* (France) which covers different types of scenarios in urban and peri-urban environments with a travel length of about 9km. We use a car equipped with five IBEO-Lux sensors deployed to cover a 360° field of view around the vehicle.

A clear statistical overview of the datasets used in both training and in the experiments is given in Table I.

TABLE I

STATISTICAL OVERVIEW OF DIFFERENT DATASETS AND THE PURPOSE OF EACH IN THE EXPERIMENTS. THE SEQUENCE LENGTHS ARE GIVEN WITH RESPECT TO THE GROUND TRUTH TRAJECTORY.

dataset	KITTI	KITTI	Bordeaux circuit
purpose	training	test	validation
number of frames	17372	2673	23677
length	≈ 13.8km	≈ 2.24km	≈ 9km

For the KITTI dataset we perform a horizontal cut in the 3D laser data in order to simulate a 2D Laser-sensor to be adapted as input to the SLAM implementation used in our experiments.

We use the PML-SLAM [28] implementation which is a 2D Maximum Likelihood SLAM approach based on laser data that uses a grid map representation. Using this approach, we are able to compare the different models as the underlying SLAM implementation is the same.

B. Experiments

The models are trained on the 46 tracks extracted from the KITTI dataset. The validation of our approach is tested through two test scenarios:

1. **The first experiment** is based on KITTI dataset test tracks. We compare the performances obtained using the different *Correction Prediction* modules presented in this article. Moreover, we use this experiment to find the best parameters (i.e. λ and ξ) of the correction prediction model.

2. **The second experiment** is based on the Bordeaux Circuit using the validated parameters from the first experiment. The challenges arising from the test are multiple: the data are collected using our platforms and thus the underlying sensor characteristics are different. In addition, the environment is more complex and the driving conditions are different.

C. Results and Discussion

The results obtained from the first experiment on the KITTI test tracks are illustrated independently for each track. Table II illustrates errors on KITTI test track 1, Table III shows errors on test track 2, Table IV shows errors on track 3 and Table V for the KITTI test track 4.

Each table is divided into two parts, the upper part details linear displacement errors ϵ_d expressed in *cm*, while the lower part gives the angular component of the error ϵ_θ , expressed in degrees.

TABLE II

ERRORS TABLE SHOWING PERFORMANCES OBTAINED USING DIFFERENT MODULES ON **KITTI TEST TRACK 1** (2011 09 26 DRIVE 0117).

<i>lin(cm)</i>	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	1.78	2.38	-0.13			
+dS	1.41	1.94	0.021	79.28%	81.66%	-0.109
+ML	1.63	2.18	0.0004	91.39%	91.74%	-0.1296
+dSML	1.39	1.94	-0.085	77.84%	81.80%	-0.045
<i>rot(°)</i>	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	0.043	0.055	-0.004			
+dS	0.041	0.053	0.041	96.46%	96.67%	+0.037
+ML	0.039	0.051	0.039	92.14%	91.63%	+0.035
+dSML	0.038	0.049	0.038	91.27%	90.36%	+0.034

TABLE III

ERRORS TABLE SHOWING PERFORMANCES OBTAINED USING DIFFERENT MODULES ON **KITTI TEST TRACK 2** (2011 09 26 DRIVE 0064).

<i>lin(cm)</i>	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	1.43	1.87	-0.298			
+dS	1.29	1.85	-0.072	90.47%	98.86%	-0.226
+ML	1.38	1.96	-0.130	93.02%	92.49%	-0.168
+dSML	1.07	1.51	-0.145	74.87%	80.53%	-0.153
<i>rot(°)</i>	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	0.045	0.06	-0.006			
+dS	0.044	0.059	0.044	98.31%	98.64%	+0.038
+ML	0.044	0.07	0.044	92.17%	92.25%	+0.038
+dSML	0.041	0.055	0.041	91.74%	91.97%	+0.035

The tables illustrate different error measures: the Mean Absolute Error (MAE), which expresses the mean value of the disagreement between the estimations and the ground truth. The Mean Signed Error (MSE) which expresses the deviation of the estimator errors from zero (i.e. around which point the estimator errors are centered), the Root Mean Squared Error (RMSE) which expresses the mean standard deviation between the estimated values and the ground truth. The Normalized Mean Absolute Error (NMAE) and the Normalized Root Mean Squared Error (NRMSE) are normalized to the corresponding errors of the raw SLAM on the same track. This gives a percentage of the improvement made by the Correction

TABLE IV
ERRORS TABLE SHOWING PERFORMANCES OBTAINED USING DIFFERENT MODULES ON **KITTI TEST TRACK 3** (2011 09 30 DRIVE 0020).

$lin(cm)$	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	2.01	2.61	-1.166			
+dS	1.87	2.40	-0.383	93.08%	92.11%	-0.783
+ML	1.75	2.27	-0.382	87.12%	86.85%	-0.784
+dSML	1.64	2.15	-0.166	82.02%	82.52%	-1
$rot(^{\circ})$	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	0.056	0.083	-0.008			
+dS	0.055	0.082	0.055	99.65%	99.52%	+0.047
+ML	0.057	0.083	0.057	102.01%	101.97%	+0.049
+dSML	0.056	0.084	0.056	101.07%	101.59%	+0.048

TABLE V
ERRORS TABLE SHOWING PERFORMANCES OBTAINED USING DIFFERENT MODULES ON **KITTI TEST TRACK 4** (2011 09 29 DRIVE 0004).

$lin(cm)$	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	1.30	1.62	0.105			
+dS	0.99	1.25	0.156	76.29%	77.07%	+0.051
+ML	1.26	1.57	0.405	96.95%	96.69%	+0.3
+dSML	0.89	1.13	0.125	69.00%	69.69%	+0.02
$rot(^{\circ})$	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	0.036	0.044	-0.004			
+dS	0.034	0.042	0.034	95.24%	95.83%	+0.03
+ML	0.031	0.038	0.031	86.87%	87.37%	+0.027
+dSML	0.031	0.038	0.031	87.23%	87.52%	+0.027

Module on each error measure. The lower the percentage is, the better the Correction Prediction Module is. Moreover, the difference of the absolute mean signed deviation (dMSD) shows the amplification / reduction of the general estimator bias after applying the corrections. It is calculated as the MSD of the rectified SLAM minus the MSD of raw SLAM. Thus, a negative value sign means that correction prediction reduces the bias, while a positive one means that a more important bias affects the estimation after applying the corrections.

Based on the NRMSE and MAE measures, we clearly see that the correction module enhances the accuracy of SLAM estimations. Furthermore, the SLAM+dSML module generally performs better than the other Correction Predictors.

The NMAE measure shows that error amplitudes generally become smaller, from 7% to a 30% attenuation on the amplitude of linear displacement error and from 4% to 9% attenuation on the amplitude of heading errors using Correction Prediction rather than raw SLAM.

Based on the MSE metric, the center of the linear errors when using the corrections of the proposed approach is slightly closer to zero. This desirable effect means that by using the correction module, linear drift is better centered on zero. Conversely, the correction brings the angular errors center slightly farther from zero. The main reason for this behavior is the unbalanced training dataset in terms of drift scenarios.

Moreover, we can observe from Table IV that corrections based on the likelihood distribution induce a slightly amplified heading error. This could be due to the absence of a similar scenario in the training dataset.

The global path of PML-SLAM, the SLAM+dSML (i.e.

SLAM with correction prediction based on successive SLAM poses and likelihood distribution) path and the ground truth path, are illustrated in Figures 5 and 6.

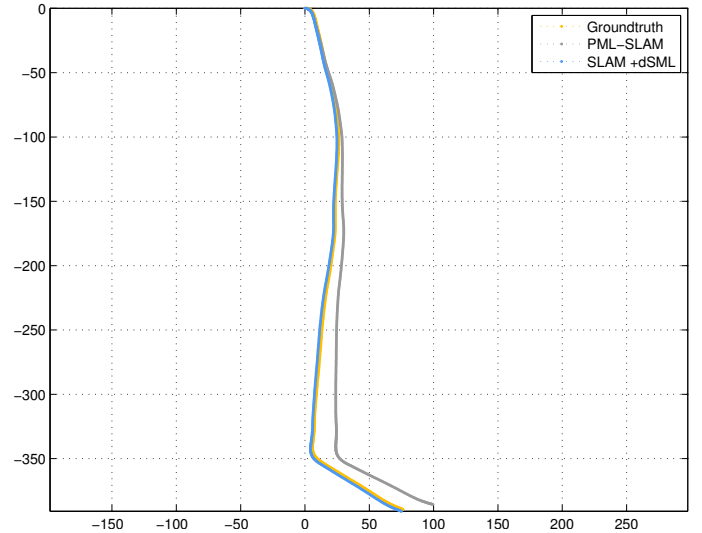


Fig. 5. KITTI test track 2 (residential). A comparison between the PML-SLAM estimated path (illustrated in gray), the ground truth path (yellow) and the SLAM+dSML rectified using the proposed approach (in blue).

The path of rectified SLAM using the correction prediction is almost superimposed on the ground truth path in Figure 5 and Figure 6(c). Furthermore, in Figure 6(b), the rectified path and the endpoint of the rectified path are closer to the ground truth path and endpoint than the path estimated by raw SLAM. In contrast, in Figure 6(a) the raw SLAM final endpoint is closer to the ground truth than the rectified SLAM+dSML. In fact, raw SLAM performs estimation errors that make the path deviate from the ground truth, but later on, other errors reduce the global localization error.

Reducing errors occurring at each time-step along the trajectory should theoretically bring the estimated path closer to the ground truth. However, this is not necessarily always the case, especially in complex and long circuits because of the incremental nature of the underlying approach (c.f. Figure 1). In other words, the accuracy measure cannot be reduced to how far the endpoint of the estimated trajectory is compared to the ground truth.

TABLE VI
PARAMETERS OF CORRECTION PREDICTION MODULES (BASED ON ENSEMBLE MLP) WITH WHICH BEST RESULTS WERE OBTAINED ON THE TEST TRACKS OF KITTI DATASET.

	Inputs number	hidden units (λ)	ensemble size (ξ)
+dS	6	4	5
+ML	24	9	5
+dSML	30	25	5

The validated parameters for each module which were used to obtain the illustrated results are summarized in Table VI. Although these parameters gave the lowest global error results on the KITTI test tracks, the global error does not

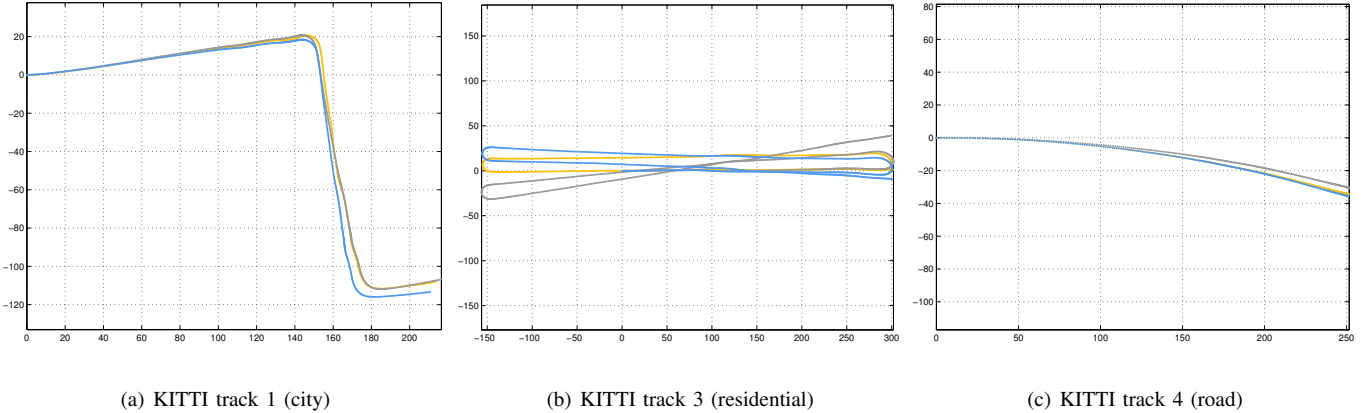


Fig. 6. A comparison between PML-SLAM estimated path (illustrated in gray), the ground truth path (yellow) and the rectified SLAM using the proposed approach (in blue) based on previous displacements and the likelihood distribution

change greatly when using other configurations (i.e. when varying (λ, ξ)). Thus, the results obtained using the approach we propose can be considered quite stable.

The second experiment on the Bordeaux Circuit was performed using the validated models and parameters of the previous experiment. Table VII summarizes the error values based on the same metrics as those used in the first experiment.

We clearly see that the corrections improve the heading accuracy considerably (nearly 20% vs. 10% in the first experiments based on the NRMSE measure). In fact, the Velodyne used to acquire the KITTI dataset has a higher angular resolution 0.09° separating laser beams vs. 0.25° for the IBEO-Lux laser sensors used in the Bordeaux Circuit acquisition. Consequently, the use of the IBEO-Lux laser sensors causes a greater angular error on the bare SLAM than the use of Velodyne. Therefore, the correction module had adapted to handle the errors induced from the underlying sensor to overcome the subsequent additional amount of errors in the heading.

TABLE VII
ERRORS TABLE SHOWING PERFORMANCES OF SLAM WITH AND WITHOUT USING THE CORRECTION PREDICTION MODULE ON THE BORDEAUX CIRCUIT.

$lin(cm)$	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	2.76	3.87	-0.36			
+dSML	2.44	3.61	-0.72	88.26%	93.20%	+0.36
$rot(^{\circ})$	MAE	RMSE	MSD	NMAE	NRMSE	dMSD
SLAM	0.063	0.089	-0.006			
+dSML	0.048	0.071	0.04	76.47%	79.15%	+0.034

The error histograms of PML-SLAM and SLAM+dSML with corrections are illustrated in Figure 7. The histogram emphasizes the number of frames in which a given amount of errors occurs. We see that the rectified estimation histograms peaks are much higher and the bins at the margins are less tall than on the error histograms of the bare PML-SLAM. In other words, applying the corrections in SLAM estimations attenuates the errors.

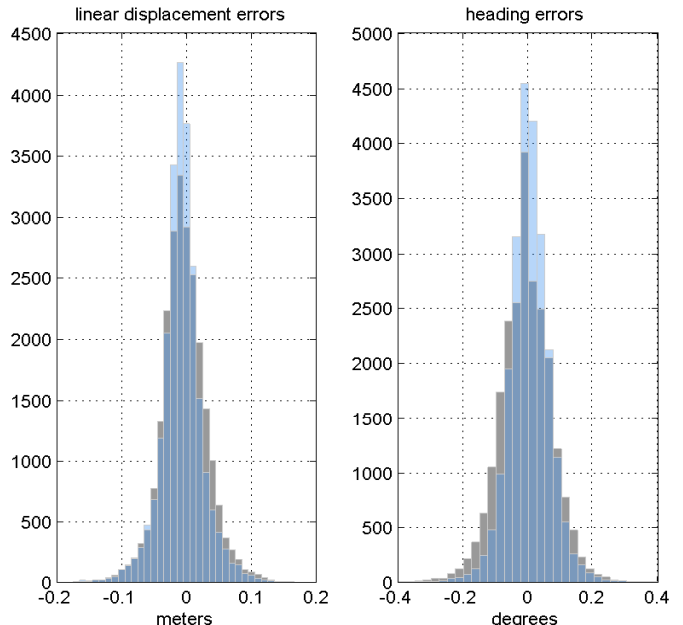


Fig. 7. Error histograms of the SLAM (gray bins) vs. the SLAM+dSML (blue bins) with the correction prediction on the Bordeaux Circuit. The histograms are superposed, thus the dark blue area means a shared area between the SLAM with and without corrections.

VI. CONCLUSION AND PERSPECTIVES

In this paper, we have presented an approach to predict corrections for 2D SLAM methods. Based on two types of relevant information (successive estimated poses and the likelihood distribution), our strategy could be suitable both for 2D SLAM methods (independently of the sensor used and the underlying SLAM approach) and for 2D Maximum Likelihood SLAM approaches (also independently of the underlying sensor). The proposed strategy performs separately from the SLAM module and operates in parallel where the proposed corrections are applied a posteriori to the SLAM estimations. The correction modules are based on MLP Ensembles which can be trained once off-line.

The validity of our approach was proved through two

experiments. We observed an improvement (attenuation of the error) of approximately 20% to 30% in linear displacement errors, and of approximately 10% in heading errors, both based on the NRMSE measure. In addition, we can see that the model's global errors do not change greatly when changing the Ensemble and MLP parameters. Thus, the solution gives a stable behavior to the model parameters. Furthermore, in the second experiment, our approach showed a robustness to additional errors induced from using a lower resolution sensor. Indeed, we managed to reduce the heading errors by 20% when using a laser with 0.25° resolution vs. 10% when using a laser sensor with 0.09° . In contrast, the modules built showed that angular error corrections are less centered to zero than when using SLAM alone. This could be due to a training bias because right turn maneuvers are more frequent in the dataset than left turns.

In future work, we plan to extend our approach and make use of the probabilities of the particles as relevant information in the case of PF-based SLAM approaches, and the sigmas probabilities from the UKF-based SLAM in order to build more suitable correction prediction models for these approaches. Moreover, we aim to evaluate the gain in error attenuation when fusing multiple localization sources. Finally, we believe it is worth studying a homogeneous learning dataset in terms of scenarios and its impact on the quality of the resulting correction modules.

REFERENCES

- [1] T. Bailey and H. Durrant-Whyte, "Simultaneous localization and mapping (SLAM): Part II," *IEEE Robotics and Automation Magazine*, vol. 13, no. 3, pp. 108–117, 2006.
- [2] H. Durrant-Whyte and T. Bailey, "Simultaneous localization and mapping: part I," *IEEE Robotics & Automation Magazine*, vol. 13, no. 2, pp. 99–110, 2006.
- [3] J. Z. Sasiadek and P. Hartana, "Sensor data fusion using Kalman filter," in *Information Fusion, 2000. FUSION 2000. Proceedings of the Third International Conference on*, vol. 2. IEEE, 2000, pp. WED5/19–WED5/25 vol.2.
- [4] A. J. Davison, I. D. Reid, N. D. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE transactions on pattern analysis and machine intelligence*, vol. 29, no. 6, pp. 1052–1067, 2007.
- [5] R. Martinez-Cantin and J. A. Castellanos, "Unscented SLAM for large-scale outdoor environments," in *Intelligent Robots and Systems, 2005.(IROS 2005). 2005 IEEE/RSJ International Conference on*. IEEE, 2005, pp. 3427–3432.
- [6] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "FastSLAM: A factored solution to the simultaneous localization and mapping problem," in *Proc. of 8th National Conference on Artificial Intelligence/14th Conference on Innovative Applications of Artificial Intelligence*, vol. 68, no. 2, 2002, pp. 593–598.
- [7] M. Montemerlo and S. Thrun, "FastSLAM 2.0," *Springer Tracts in Advanced Robotics*, vol. 27, pp. 63–90, 2007.
- [8] S. Grzonka, C. Plagemann, G. Grisetti, and W. Burgard, "Look-ahead Proposals for Robust Grid-based SLAM with Rao-Blackwellized Particle Filters," *The International Journal of Robotics Research*, vol. 28, no. 2, pp. 1–16, 2009.
- [9] E. Mouragnon, M. Lhuillier, M. Dhome, F. Dekeyser, and P. Sayd, "Real time localization and 3D reconstruction," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1. New York, NY, USA: IEEE, 2006, pp. 363–370.
- [10] G. Klein and D. Murray, "Parallel tracking and mapping on a camera phone," in *Science and Technology Proceedings - IEEE 2009 International Symposium on Mixed and Augmented Reality, ISMAR 2009*. IEEE, 2009, pp. 83–86.
- [11] J. Zhang and S. Singh, "LOAM: Lidar odometry and mapping in real-time," in *Robotics: Science and Systems Conference (RSS)*, 2014.
- [12] —, "Visual-lidar Odometry and Mapping : Low-drift , Robust , and Fast," in *International Conference on Robotics and Automation*, 2015, pp. 2174–2181.
- [13] W.-s. Choi, J.-g. Kang, and S.-y. Oh, "Measurement Noise Estimator Assisted Extended Kalman Filter for SLAM Problem," in *International Conference on Intelligent Robots and Systems 2009*. IEEE, 2009, pp. 2077–2082.
- [14] M. Choi, R. Sakthivel, and W. K. Chung, "Neural network-aided extended Kalman filter for SLAM problem," in *Proceedings - IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 1686–1690.
- [15] K. Yu, K. Alexandr, and M. A. Sobolev, "Recurrent neural network and extended Kalman filter in SLAM problem," *IFAC Proceedings Volumes*, vol. 23, no. 3, pp. 4–7, 2013.
- [16] K.-S. Choi and S.-G. Lee, "Enhanced SLAM for a mobile robot using extended Kalman Filter and neural networks," *International Journal of Precision Engineering and Manufacturing*, vol. 11, no. 2, pp. 255–264, 2010.
- [17] O. Panah, A. Panah, A. Panah, and S. Fallahpour, "Enhanced SLAM for Autonomous Mobile Robots using Unscented Kalman Filter and Neural Network," *Indian Journal of Science and Technology*, vol. 8, no. August, 2015.
- [18] R. Sharaf, A. Noureldin, A. Osman, and N. El-Sheimy, "Online INS/GPS integration with a radial basis function neural network," *IEEE Aerospace and Electronic Systems Magazine*, vol. 20, no. 3, pp. 8–14, 2005.
- [19] J. J. Wang, J. Wang, D. Sinclair, and L. Watts, "Neural Network Aided Kalman Filtering for Integrated Gps/Ins Geo-Referencing Platform," in *CiteSeer*, 2007, pp. 1–6.
- [20] Q. Zhang and B. Li, "A low-cost GPS/INS integration based on UKF and BP neural network," in *5th International Conference on Intelligent Control and Information Processing, ICICIP 2014 - Proceedings*. IEEE, 2014, pp. 100–107.
- [21] Z. Jingsen, Z. Wenjie, H. Bo, and W. Yali, "Integrating Extreme Learning Machine with Kalman Filter to Bridge GPS Outages," in *2016 3rd International Conference on Information Science and Control Engineering (ICISCE)*. IEEE, 2016, pp. 420–424.
- [22] W. Burgard, C. Stachniss, G. Grisetti, B. Steder, R. Kümmerle, C. Dornhege, M. Ruhnke, A. Kleiner, and J. D. Tardós, "A comparison of SLAM algorithms based on a graph of relations," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2009*. IEEE, 2009, pp. 2089–2095.
- [23] R. Kümmerle, B. Steder, C. Dornhege, M. Ruhnke, G. Grisetti, C. Stachniss, and A. Kleiner, "On measuring the accuracy of SLAM algorithms," *Autonomous Robots*, vol. 27, no. 4, pp. 387–407, 2009.
- [24] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite," in *Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 3354–3361.
- [25] M. P. Perrone, "Improving Regression Estimation: Averaging Methods for Variance Reduction with Extensions to General Convex Measure Optimization," 1993.
- [26] T. Windeatt, "Ensemble MLP classifier design," *Computational Intelligence Paradigms*, pp. 133–147, 2008.
- [27] S. Hashem and B. Schmeiser, "Approximating a function and its derivative using MSE optimal linear combination of trained feed forward neural networks," in *Proceedings of the 1993 World Congress on Neural Networks, Vol. 2. New Jersey: Lawrence Erlbaum Associates.*, pp. 617–620, 1993.
- [28] Z. Alsayed, G. Bresson, F. Nashashibi, and A. Verroust-blondet, "PML-SLAM : a solution for localization in large-scale urban environments," in *IROS PPNIV 7th Workshop on Planning, Perception and Navigation for Intelligent Vehicles*, Hamburg, Germany, 2015.
- [29] A. Daffertshofer, C. J. Lamoth, O. G. Meijer, and P. J. Beek, "PCA in studying coordination and variability: A tutorial," *Clinical Biomechanics*, vol. 19, no. 4, pp. 415–428, 2004.
- [30] J. Shlens, "A Tutorial on Principal Component Analysis," 2014.
- [31] G. Bresson, Z. Alsayed, L. Yu, and S. Glaser, "Simultaneous Localization And Mapping: A Survey of Current Trends in Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, 2017.
- [32] A. Geiger, P. Lenz, C. Stiller, and R. Urtasun, "Vision meets robotics: The KITTI dataset," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1231–1237, 2013.