An effective 6DoF motion model for 3D-6DoF Monte Carlo Localization

A. L. Ballardini[1] A. Fúrlan[1] A. Galbiati[1] M. Matteucci[2] F. Sacchi[1] D. G. Sorrenti[1]

***Abstract*— This paper deals with the probabilistic 6DoF motion model of a wheeled road vehicle. It allows to correctly model the error introduced by dead reckoning. Furthermore, to stress the importance of an appropriate motion model, i.e., that different models are not equally good, we show that another model, which was previously developed, does not allow a correct representation of the uncertainty, therefore misguiding 3D-6DoF Monte Carlo Localization. We also present some field experiments to demonstrate that our model allow a consistent determination of the 6DoF vehicle pose.**

I. INTRODUCTION

In urban settings, autonomous driving is more similar to mobile robotics, because of the need to have a global localization of the vehicle. Localization cannot be managed using purely dead reckoning, e.g., wheel based odometry [1] [2] [3]. Wheel sliding, e.g., due to contact with the ground surface, weather conditions, unexpected values of the wheels diameters, etc., require the use of external sensors and the corresponding algorithms, to determine the vehicle position [2]. It must be noticed that in urban environments the GPS system, apparently an immediately available solution, has an absolutely not adequate reliability, with respect to the localization and navigation requirements, due to the frequent lack of signal [4] [5].

While the state of the art provides different solutions for the 2D - 3DoF localization problem, these solutions are primarily designed for indoor robotic environments, where the analysis of the motion in a 3D space can be simplified, favoring an estimation of the robot pose limited to a 3DoF pose in the 2D plane. 3D approaches known in the literature, e.g., [4], [5], [6] base on adapting 2D movements to the 3D space. These approaches adopt a 3DoF probabilistic motion model in 2D that do not allow accurate modeling of the uncertainty of a 6DoF movement in a 3D space.

One might argue that a motion model might be not be necessary at all. This might be true when the localization algorithm could be executed at such an high frequency that the displacement involved between two subsequent activations of it, is so short that it is reasonable to model the pose uncertainty as normal, and affecting independently the single components of the pose. The larger the displacement between two activations, the larger the uncertainty of the odometric estimate. Such large uncertainty requires, on one hand a very large number of samples (in current state of the art sample-based approaches), on the other - most important - side, it is unrealistically shaped. Conversely, a proper motion model allows to focus samples where it is realistically possible to have the true pose.

In [5] the robot poses are modeled only in 2D, i.e., the state includes only the components x, y, and ϑ (yaw). The other 3 components, i.e., z, the roll angle φ, and the pitch angle ψ, are calculated from the 2D pose estimate and from the structure of the ground surface. Furthermore, the motion does not consider the interactions between the errors acting on the components, and introduce uncertainty on the single components of the movement according to a velocity model. It is to be observed that the independency between the single components of the pose is also assumed in other works [6].

In [4] a representation called multi-level surface maps is used. This technique is proposed as an extension of the elevation maps used in [7] and introduced in [8]. It allows modeling vertical structures within a grid map used for localization with laser range finders. However, these structures do not allow the representation of some typical urban outdoor situations, like bridges or multilevel parkings. Furthermore, in [4] the motion model, more sophisticated then to the one in [5], bases on an evolution of the model introduced in [9], and is similar to that illustrated in [2], again a purely 2D-3DoF motion model.

The inadequacy of these simplifications in urban outdoor situations has driven us to develop a different probabilistic motion model, based on the modeling of a spatial generic movement considering all the components of the 6DoF state. The model, adaptable to different vehicle kinematics, accommodates 6DoF movements even when sensing of some component is missing, e.g., in case of a wheeled vehicle without an IMU.

The next section introduces the proposed motion model, then in section III we compare the our proposal and model that we developed previously, in order to clarify the relevance of an appropriate model. We then conclude presenting experimental data.

II. PROPOSED MOTION MODEL

The model we propose bases on the 2D-3DoF formulation presented in [2, Sect. 5.4]. In that work a displacement is divided in a sequence of 3 steps: an onsite rotation δ_{rot1}, a translation δ_{transl}, and another onsite rotation δ_{rot2}, see Figure 1. This decomposition allows the introduction of the uncertainty on each step in the form of a normal error. These

The authors thank reviewer n. 1 for the comment about the injection of uncertainty in just 3DoF, in the naive motion model; the paper has been changed in this respect.

This work has been supported by the Italian Ministry of University and Research (MIUR) through the PRIN 2009 project *ROAMFREE*

[1]Dept. Informatica, Sist. e Com., Università di Milano - Bicocca, Build. U14 v.le Sarca 336, 20126, Milano, Italy; {galbiati, sacchi}@ira.disco.unimib.it, {ballardini, furlan, sorrenti}@disco.unimib.it

[2]Dept. Elettronica e Informazione, Politecnico di Milano, P.zza Leonardo da Vinci 32, 20132, Milano, Italy; matteucci@elet.polimi.it

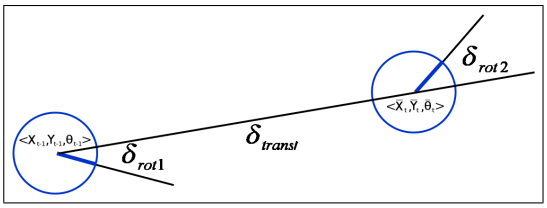

Fig. 1. 2D-3DoF motion model from [2, Sect. 5.4].

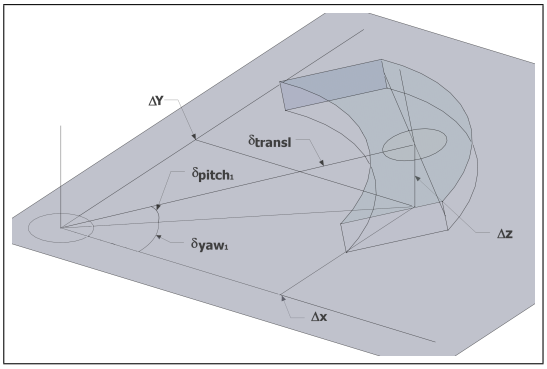

Fig. 2. Uncertainty area of the 3D Model.

errors are zero-mean and are parameterized by a standard deviation that is dimensioned according to the disturbances acting on each step. The composition of the uncertainties of the steps gives a realistic uncertainty affecting the pose, after the application of the motion.

Similarly, the 6DoF motion model has to include motion in all the 6 DoFs of the vehicle and, at the same time, it should divide the whole displacement in a sequence of steps, again zero-mean, so that a parameterized uncertainty on each of the steps turns, after their composition, into a realistic uncertainty on the final pose.

An extended odometer (dead reckoning) can include estimates of displacements in all 6 DoFs, by integrating odometry, i.e., displacements estimated from the rotation of the wheels, with IMU data. In particular, Δx, Δy, and $\Delta\vartheta$ (yaw) can be output by a wheel odometer, while the $\Delta\varphi$ (roll), $\Delta\psi$ (pitch), and Δz, can be output by an IMU. Our proposal includes a set of additional parameters, used when the odometric readings lack some components; e.g., in the case of an IMU-less vehicle.

The state vector has six components, to represent the pose of a rigid body in a 3D world $\mathbf{x}_t = |x, y, z, \varphi, \psi, \vartheta|'$. Let us group the first 3 components in $position_t$ and the last 3 in $orientation_t$, so that $\mathbf{x}_t = |position_t, orientation_t|'$.

Our proposed decomposition of a displacement bases on six steps, which can be grouped in 2 sets: 3 steps to define the new position $\overline{position}_t$ [1], and 3 steps to define $\overline{orientation}_t$. On each step some uncertainty will be added. We now review the first 3 steps, with reference to Figure 2, which give $\overline{position}_t$.

1) Rotation δ_{yaw_1}, which represents a rotation around the Z axis, is necessary to align $orientation_{t-1}$ toward $\overline{position}_t$ in the XY plane; this step corresponds to 2D-3DoF rotation δ_{rot1}.
2) Rotation δ_{pitch_1}, which represents a rotation around the Y axis, and is also necessary to align $orientation_{t-1}$ toward $\overline{position}_t$, but in the XZ plane; this step introduces the possibility of a change in the value of the elevation.
3) Translation δ_{transl}, which represents a translation along the X axis; this translation moves the reference system, after the rotation by δ_{yaw_1} and δ_{pitch_1}, to $\overline{position}_t$; this step corresponds to 2D-3DoF translation δ_{transl}.

The three parameters δ_{yaw_1}, δ_{pitch_1}, and δ_{transl} can be seen as the coordinates, in a spherical coordinate system, of the origin of the new pose $\mathbf{x}_t$. To compute the motion parameters from the extended odometer readings, equations 1 to 3 can be used.

$$\delta_{yaw_1} = arctan(\frac{\Delta y}{\Delta x}) \tag{1}$$

$$\delta_{pitch_1} = arctan(\frac{\Delta z}{\sqrt{\Delta x^2 + \Delta y^2}}) \tag{2}$$

$$\delta_{transl} = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2} \tag{3}$$

For the computation of $\overline{orientation}_t$, our proposal is to compose $orientation_{(t-1)}$ with a generic rotation, which is in turn the composition of 3 last rotation steps. The parameters of these steps, i.e., δ_{roll}, δ_{pitch_2}, δ_{yaw_2}, are sensed directly by the extended odometer.

$$\delta_{roll} = \Delta\varphi; \delta_{pitch_2} = \Delta\psi; \delta_{yaw_2} = \Delta\vartheta \tag{4}$$

In order for the motion model to generate realistic motion uncertainty, it is necessary to add randomness to the components of the state vector $\overline{x}_t$, by acting on the parameters of the motion model. This randomness will be normally distributed, with zero mean. The standard deviation of the components can be calculated according to the following considerations, which are specific to each single step.

1) Rotation δ_{yaw_1}, as in [2], is influenced by:
   - how much the vehicle has rotated, as measured by the wheel odometer;
   - how much space the vehicle has traveled, as measured by the wheel odometer.

   For both factors, the larger the factor, i.e., the change of orientation and/or the traveled distance, the larger the potential mismatch between the odometric measure and real pose.
2) Rotation δ_{pitch_1}, is influenced by:
   - how much the z coordinate has changed, i.e., by Δz, as measured by the extended odometer, from the IMU.
3) Translation δ_{transl} is influenced by:
   - how much space the vehicle has traveled, as measured by the extended odometer; the longer the

[1] With the notation ($\overline{abc}$) we refer to the prediction of the state abc, obtained by the application of the motion model.

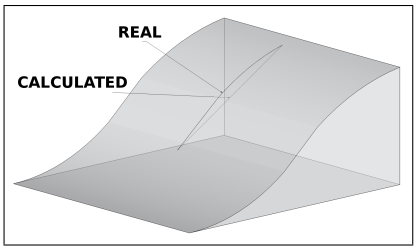

Fig. 3. REAL and CALCULATED (basing on odometry) trajectories, which impact on the uncertainty on δ_{transl}, as due to a change in pitch ($\Delta\psi$).

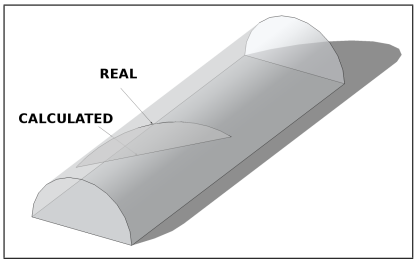

Fig. 4. REAL and CALCULATED (basing on odometry) trajectories, which impact on the uncertainty on δ_{transl}, as due to a change in roll ($\Delta\varphi$).

traveled distance, the larger the potential mismatch between the odometric measure and real pose;

- how much the vehicle has rotated about the Y axis, i.e., the variations $\Delta\psi$, as measured by the extended odometer. A change of pitch while performing a translation, represents a situation where the motion is taking place over a non planar surface. Therefore the traveled distance is larger and the uncertainty is also larger. Figure 3 illustrates the translation resulting from integration of odometry, and the real translation.
- how much the vehicle has rotated about the X axis, i.e., the variation in roll $\Delta\varphi$, as measured by the extended odometer. Figure 4 illustrates the translation resulting from integration of odometry, and the real translation.
- how much the vehicle has rotated about the Z axis, i.e., the variation $\Delta\vartheta$, as measured by the extended odometer. Figure 5 illustrates the translation resulting from integration of odometry, and the real translation.

4) Rotation δ_{roll} is influenced by:
   - how much the vehicle has rotated around its X axis, i.e., variation $\Delta\varphi$, as measured by the extended odometer, from the IMU.
5) Rotation δ_{pitch_2} is influenced by:
   - how much the vehicle has rotated around the Y axis, i.e., the variation $\Delta\psi$, as measured by the extended odometer, from the IMU.
6) Rotation δ_{yaw_2} is influenced by:
   - how much the vehicle has rotated around the Z axis, i.e., the variation $\Delta\vartheta$, as measured by the extended odometer, from the wheel odometer.

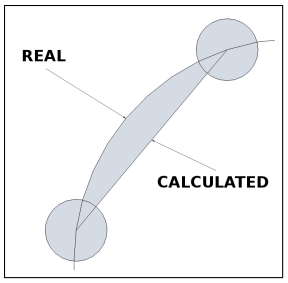

Fig. 5. REAL and CALCULATED (basing on odometry) trajectories, which impacts on the uncertainty on δ_{transl}, as due to a change in yaw ($\Delta\vartheta$).

- how much space the vehicle has traveled: the longer the traveled distance, the larger the potential mismatch between the odometric measure and reality, as measured by the wheel odometer.

Basing on the above mentioned influences, we can define the standard deviations of the noise representing the uncertainty affecting the 6 steps. Finally, in order to gain a better control on the model behavior and similarly to what has been done in [2], we introduce a weight α, for each step.

$$\sigma_{yaw_1} = \alpha_1 \cdot \delta_{yaw_1} + \alpha_2 \cdot \delta_{transl} \tag{5}$$

$$\sigma_{pitch_1} = \alpha_3 \cdot \Delta z \tag{6}$$

$$\sigma_{transl} = \alpha_4 \cdot \delta_{transl} + \alpha_5 \cdot \delta_{yaw_2} + \alpha_6 \cdot (\delta_{roll} + \delta_{pitch_2}) \tag{7}$$

$$\sigma_{roll} = \alpha_7 \cdot \delta_{roll} \tag{8}$$

$$\sigma_{pitch_2} = \alpha_8 \cdot \delta_{pitch_2} \tag{9}$$

$$\sigma_{yaw_2} = \alpha_9 \cdot \delta_{yaw_2} + \alpha_{10} \cdot \delta_{transl} \tag{10}$$

The IMU uncertainty is assumed not correlated with the wheel odometer uncertainty. Moreover, notice that σ_{roll}, σ_{pitch_1}, and σ_{pitch_2} are influenced only by the IMU part of the extended odometer, while σ_{transl} is influenced both by the wheel odometer and the IMU, see Figure 4, 3, and 5.

The sampling motion model will be the following:

$$\hat{\delta}_{yaw_1} = \delta_{yaw_1} + SAMPLE\underbrace{\{\alpha_1 \cdot \delta_{yaw_1} + \alpha_2 \cdot \delta_{transl}\}}_{\sigma_{yaw_1}} \tag{11}$$

$$\hat{\delta}_{pitch_1} = \delta_{pitch_1} + SAMPLE\underbrace{\{\alpha_3 \cdot \Delta z\}}_{\sigma_{pitch_1}} \tag{12}$$

$$\hat{\delta}_{transl} = \delta_{transl} + SAMPLE\left(\underbrace{\begin{array}{l}\alpha_4 \cdot \delta_{transl} + \alpha_5 \cdot \delta_{yaw_2} + \\ \alpha_6 \cdot (\delta_{roll} + \delta_{pitch_2})\end{array}}_{\sigma_{transl}}\right) \tag{13}$$

$$\hat{\delta}_{roll} = \delta_{roll} + SAMPLE\underbrace{(\alpha_7 \cdot \delta_{roll})}_{\sigma_{roll}} \tag{14}$$

$$\hat{\delta}_{pitch_2} = \delta_{pitch_2} + SAMPLE\underbrace{(\alpha_8 \cdot \delta_{pitch_2})}_{\sigma_{pitch_2}} \quad (15)$$

$$\hat{\delta}_{yaw_2} = \delta_{yaw_2} + SAMPLE\underbrace{(\alpha_9 \cdot \delta_{yaw_2} + \alpha_{10} \cdot \delta_{transl})}_{\sigma_{yaw_2}} \quad (16)$$

In case an extended odometer is not available, the expected value will of course be null, and we can use an a priori standard deviation value for each parameter, determined on the basis of the expectations on the change that the terrain can induce in each degree of freedom. Of course this option implies a larger uncertainty, which in turn requires a larger computational effort.

A. Model thresholds

The model exploits a few parameters, i.e., thresholds, in order to handle some situations.

1) Minimum thresholds: As it can be noticed in the relationships above, and similarly to what is done in [2, Sect 5.4], the standard deviations of the uncertainties are proportional to the amount of motion involved into each step. Whenever the motion is too small, the standard deviation gets underestimated. These thresholds are used in such cases; they guarantee a minimum dispersion of the sampled data, which is necessary to correctly represent the real uncertainty.

2) Maximum thresholds: These thresholds have been introduced in order to handle situations where the extended odometer does not give out values in 6DoF, i.e., when there is no IMU on the vehicle. Maximum thresholds represent the maximum a priori uncertainty; on the other hand we expect a better, i.e., more concentrated estimate of robot movements when using a sensor. The σ_{max} value that is associated to every model parameter needs to be suitably large so to ensure that samples can be generated with enough dispersion about the mean value, in order to represent all possible changes on the given degree of freedom. We have chosen the values of these thresholds considering a maximum vehicle speed of 25 Km/h and a 20Hz sampling frequency for the odometer.

III. Comparison with another motion model

In order to clarify the relevance of a careful design of the motion model, we present here also a different model that we developed before the one proposed in this paper. We came first to this model because it was, in our eyes, closely resembling the 2D-3DoF model presented in [2, Sect 5.4]. This model is based on dividing the displacement into 3 steps, see Figure 6.

1) Rotation δ_{rot_1}, a rotation about an axis N_1. To obtain N_1, let us call D the vector $(\overline{position}_t - position_{(t-1)})$. N_1 is the vector product of the X axis of frame $pose_{(t-1)}$ and D.

$$N_1 = (\overline{position}_t - position_{(t-1)}) \times X_{pose_{(t-1)}} \quad (17)$$

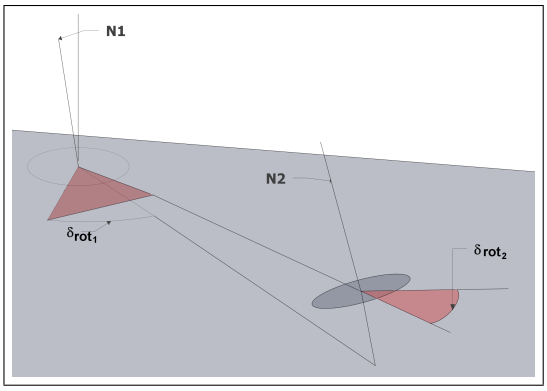

Fig. 6. The first (naive) motion model that was developed.

2) Translation δ_{transl}, a translation along the X axis of the frame obtained at the previous step, i.e., after the rotation of $pose_{t-1}$ by δ_{rot_1}. At the end the origin will reach $\overline{position}_t$.
3) Rotation δ_{rot_2}, a rotation about an axis N_2. N_2 is the vector product of the X axis of frame $pose_t$ and D.

$$N_2 = (\overline{position}_t - position_{(t-1)}) \times X_{pose_t} \quad (18)$$

This rotation aligns the reference frame, which has been obtained rotating $pose_{(t-1)}$ by δ_{rot_1} and then translating by δ_{transl}, to $\overline{orientation}_t$.

The uncertainty on the components of the motion model is sampled from normal distributions, for each of the 3 parameters δ_{rot_1}, δ_{transl}, and δ_{rot_2}. Such distributions have zero-mean and standard deviations computed similarly to what has been done for the motion model in [2, Sect 5.4]. It is just a similarity because of the need to introduce other degrees of freedom to the uncertainty affecting $\overline{position}_t$, which would be just 2 (δ_{rot_1}, and δ_{transl}), for a 3D point. We therefore add noise to the vector N_1 as, if we added noise to the vector D, we would obtain the model proposed above. Notice that the 2 parameters of N_1 are not independent w.r.t. the uncertainty of rotating about N_1, so the DoF count for $\overline{position}_t$ is correct.

This naive model, which turned out not being well performing, demonstrates how heavily the decomposition of the overall displacement, i.e., the motion model, affects the capability to produce realistic poses. Actually, the poses generated by this model are not realistically distributed about the real pose, see Figure 7 and Figure 8, where it can be observed that the uncertainty is rotated along the X axis; the larger Δz, the more rotated the particle cloud. Figure 9 shows the corresponding uncertainty for the motion model proposed above.

IV. Experimental Results

We first tested the software implementation of the proposed motion model in simulation; because of space limits we report hereafter only some real tests performed on our research vehicle. Testing have been performed is the parking area of the U5 building of Università di Milano - Bicocca, see Figures 10, 12, 13.

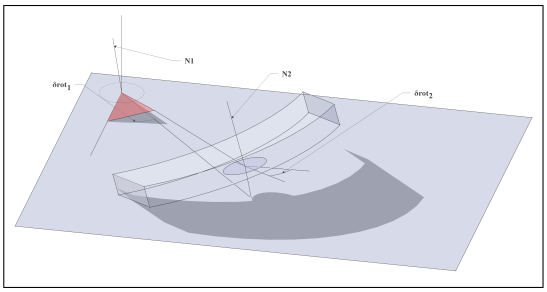

Fig. 7. Uncertainty area for the naive motion model.

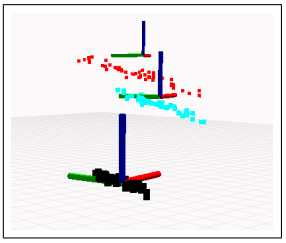

Fig. 8. 3D view of the particle set for the naive motion model. Notice that the larger the overall Δ_{yaw} and Δz, the larger the distortion of the particle set.

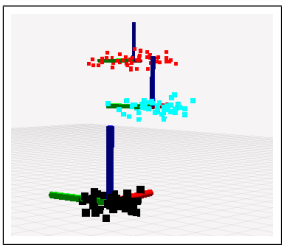

Fig. 9. 3D view of the particle set for the proposed motion model. Notice the absence of the distortion that is affecting the particle set presented in Figure 8

The motion model have been plugged into a state of the art Monte Carlo Localization software [10]. We performed the tests in this order: first we verified that the localization was correct when moving on an almost planar surface, i.e., the model was performing at least as the state of the art 2D-3DoF model. This has been done in the underground garage of the building, where the floor appears to be reasonably planar. We did obtain results comparable to the ones obtained with the 2D-3DoF state of the art software [10]. Secondly, we drove along a path including a ramp, from the garage to the outdoor parking area, see Figure 11. Also in these last tests the localization was successful, one experiment is depicted in Figure 14. As we have no ground truth for the pose, we checked that at the end of the path, at about pose n. 8 in Figure 10, the estimated pose was matching the real one, and we always obtained this result.

On the other hand, the naive motion model fails in high curvature curves, as it might be expected from observing, in Figure 7, the unrealistic uncertainty implied by this model; an example of failure is presented in Figure 15.

Despite roll and pitch data were available, thanks to an MTi X-sens IMU sensor, we verified that using only the available LIDAR sensors, altogether with appropriate

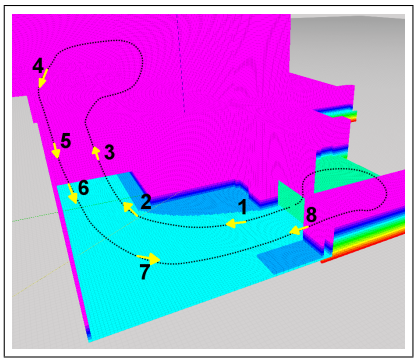

Fig. 10. Voxel representation of the U5 building underground garage. Here it is shown the part that includes the ramp leading to the outdoor parking: pose n. 8 is at the gate of the underground garage, pose n. 4 is in the outdoor parking nearby where the cart is depicted in Figure 13, poses n. 3 and 5 are on the ramp, poses n. 1, 2, 6, 7 are in the road leading to the ramp.

minimum and maximum thresholds, sufficed for a correct localization. We also noticed that using together both types of LIDAR models we had available (Sick LMS111 and LDMRS4001) is extremely useful, since they measure on different scanning planes.

CONCLUSIONS

We presented a motion model for 3D-6DoF localization, and showed that a careful design is required to obtain a realistic representation of the involved uncertainties. The presented model demonstrated its suitability in different experiments and is currently in use for our research in urban autonomous driving.

REFERENCES

[1] S. Thrun, "Probabilistic algorithms in robotics," *AI Magazine*, vol. 21, no. 4, pp. 93–109, 2000.

[2] S. Thrun, W. Burgard, and D. Fox, *Probabilistic robotics*. MIT Press, 2005.

[3] C. Olson, "Probabilistic self-localization for mobile robots," *Robotics and Automation, IEEE Trans. on*, Feb 2000.

[4] R. Kümmerle, R. Triebel, P. Pfaff, and W. Burgard, "Monte carlo localization in outdoor terrains using multi-level surface maps," in *In Proc. of the Int. Conf. on Field and Service Robotics*, 2007.

[5] T. Suzuki, M. Kitamura, Y. Amano, and T. Hashizume, "6-dof localization for a mobile robot using outdoor 3d voxel maps," in *IROS*, 2010.

[6] I. Baldwin and P. Newman, "Road vehicle localization with 2d push-broom lidar and 3d priors," in *ICRA*, Minnesota, USA, May 2012.

[7] S. Lacroix, A. Mallet, D. Bonnafous, G. Bauzil, S. Fleury, M. Herrb, and R. Chatila, "Autonomous rover navigation on unknown terrains: Functions and integration," *Int. Jour. Robotics Research*, vol. 21, no. 10-11, 2002.

[8] A. I. Eliazar and R. Parr, "Learning probabilistic motion models for mobile robots," in *21st Int. Conf. Machine Learning*, 2004.

[9] N. Roy and S. Thrun, "Online self-calibration for mobile robots," in *ICRA*, 1999.

[10] B. Gerkey, "Amcl." [Online]. Available: http://ros.org/wiki/amcl

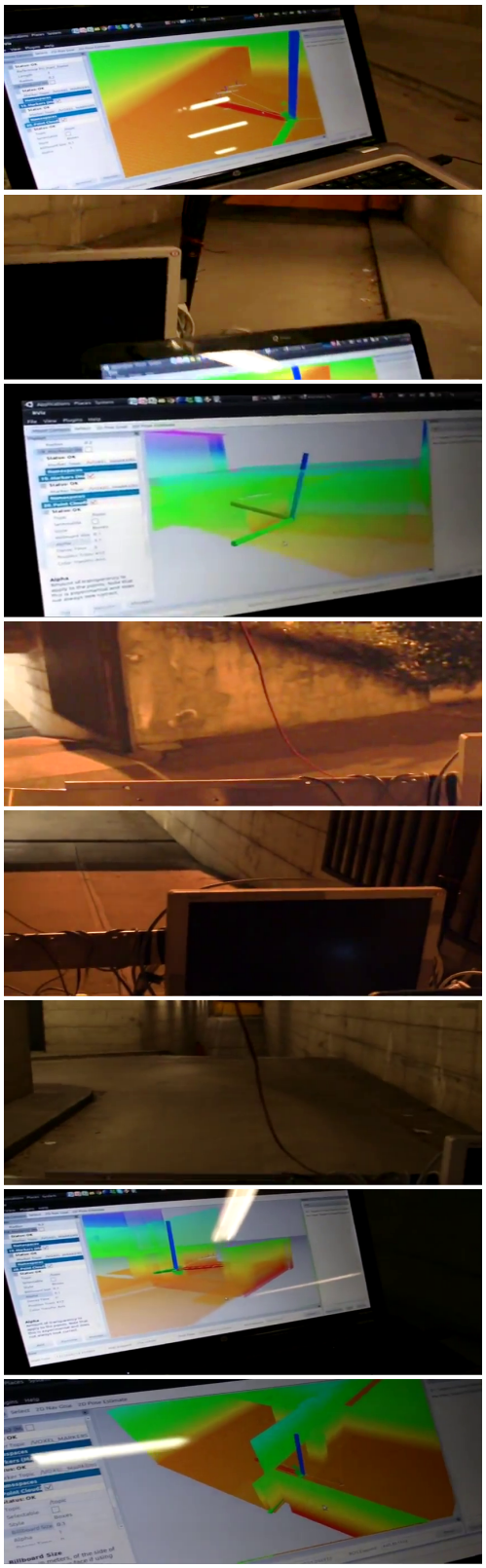

Fig. 11. Snapshots along the path shown in Figure 10: from the underground garage, through the ramp, to the outdoor parking lot and back into the garage; 1) top to 8) bottom. Notice in 1), 3), 7) and 8) the cart reference frame, shown in the same map used by the software. Notice in 3) the frame pitching up along the ramp.

Fig. 12. U5 building - Aerial view, the ramp from the underground garage can be noticed on the left of the largest tree.

Fig. 13. The ramp from the underground garage to the outdoor park, picture taken from the outdoor park.

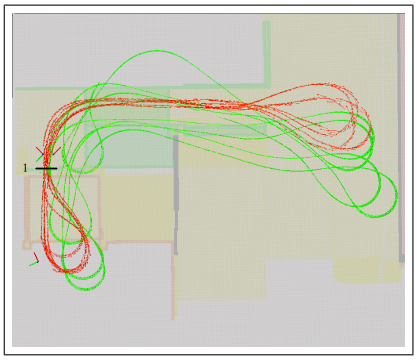

Fig. 14. The green path represents the odometric path; the red path represents the localization obtained using the proposed motion model. Notice at 1) i.e., nearby pose n. 8, the correctness of the localization.

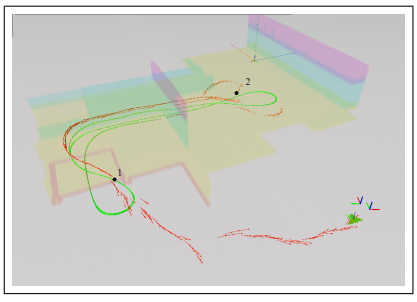

Fig. 15. A typical experiment with the naive motion model. In the path, just before 2), the localization system failed (whether this error is tolerable is not analyzed here), but in 2) it could luckily recover; in 1) it fails completely, without recovery.