

# Doppler-only Single-scan 3D Vehicle Odometry

Andres Galeote-Luque, Vladimír Kubelka, Martin Magnusson,  
 Jose-Raul Ruiz-Sarmiento and Javier Gonzalez-Jimenez

**Abstract**—We present a novel 3D odometry method that recovers the full motion of a vehicle only from a Doppler-capable range sensor. It leverages the radial velocities measured from the scene, estimating the sensor’s velocity from a single scan. The vehicle’s 3D motion, defined by its linear and angular velocities, is calculated taking into consideration its kinematic model which provides a constraint between the velocity measured at the sensor frame and the vehicle frame.

Experiments carried out prove the viability of our single-sensor method compared to mounting an additional IMU. Our method provides a more reliable translation of the sensor, compared to the errors linked to IMUs due to noise and biases. Its short-term accuracy and fast operation ( $\sim 5\text{ms}$ ) make it a proper candidate to supply the initialization to more complex localization algorithms or mapping pipelines. Not only does it reduce the error of the mapper, but it does so at a comparable level of accuracy as an IMU would. All without the need to mount and calibrate an extra sensor on the vehicle.

**Index Terms**—Localization, Range Sensing, Autonomous Vehicle Navigation, Range Odometry, Radar, Doppler.

## I. INTRODUCTION

Self-localization is one of the fundamental components of autonomous mobile robots and vehicles. This problem can be tackled by employing different sensors, but cameras and lidars are among the most widely used in contemporary methods. These sensors provide enough advantages to justify their relevance, though their performance can be severely hindered under challenging circumstances, like extreme lighting or weather conditions, or the presence of dust or mist [1].

Radar sensors, on the other hand, are a promising but underexplored alternative able to provide geometric information of its surroundings even in challenging low-visibility scenarios [2]–[4]. This makes them highly suitable for underground operations, construction sites, and other harsh environments. The relevance of radars in industrial and on-road applications has driven sensor technology forward, making them more affordable, accurate, and fast. Among the various advances, it is worth highlighting the introduction of radar sensors

\*This work has been supported by the grant program PRE2018-085026 and the research project ARPEGGIO (PID2020-117057GB-I00), all funded by the Spanish Government. The authors would also like to express their gratitude to Maximilian Hilger for his assistance with the experimental part of this work.

Andres Galeote-Luque (corresponding author), Jose-Raul Ruiz-Sarmiento and Javier Gonzalez-Jimenez (andresgalu; jotaraul; javiergonzalez@uma.es) are with Machine Perception and Intelligent Robotics (MAPIR) Group, System Engineering and Automation Department, University of Malaga, Spain.

Vladimír Kubelka and Martin Magnusson (vladimir.kubelka; martin.magnusson@oru.se) are with the Mobile Robotics & Olfaction (MRO), AASS research centre, Örebro University, Sweden.

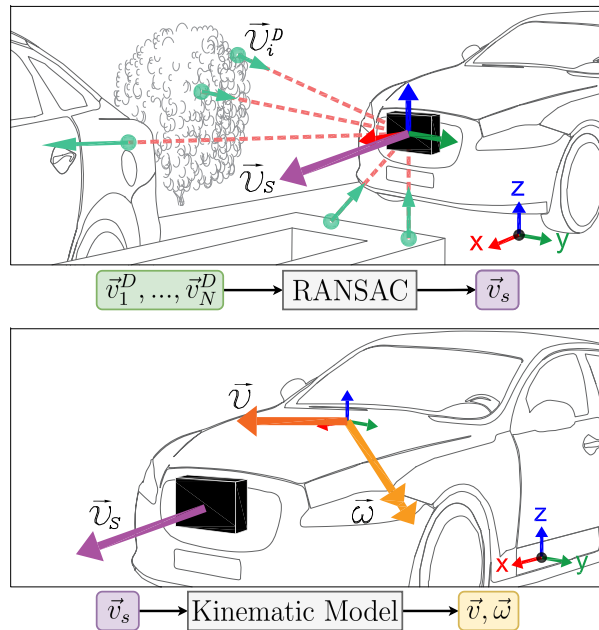


Fig. 1: Representation of the working principle of the proposed method. Top, the radial velocity  $\vec{v}_i^D$  of the observed points in the scene is leveraged to estimate the sensor’s linear velocity  $\vec{v}_s$ . Bottom, the kinematic model provides the relation between  $\vec{v}_s$  and both the linear  $\vec{v}$  and angular  $\vec{\omega}$  velocities of the vehicle.

measuring in 3D (azimuth, elevation and range), as well as their ability to estimate the radial velocity of each sensed 3D point. The latter is possible by leveraging precise phase measurements of the returning signal, which allows the sensor to measure the velocity of a point along the radar-point direction. Further on, we denote it as Doppler velocity. Note that this technology has also been implemented in certain lidars, so from now on we will refer to all of them as Doppler-capable range sensors.

The radial velocity provided by these sensors has proven to be advantageous for odometry methods, aiding in the segmentation of dynamic and static objects [5], as well as introducing more constraints to the movement estimation [6]. Given the right conditions, it can be as powerful as to recover the 2D ego-motion of the sensor from a single scan of its surroundings [7], [8]. Note that, to estimate the localization, these works rely upon either knowing the kinematic model of the vehicle or mounting multiple sensors, since not all three degrees of freedom (DOFs) of a 2D movement are observable with just the radial velocity of points of the scene. Estimating

the movement of the sensor this way removes the need to perform data association, bringing it closer to employing an Inertial Measurement Unit (IMU) or wheel odometry. These methods provide the relative motion of the sensor, and tend to accumulate error over time, also known as *drift*. However, their short-term accuracy obtained in a fast and simple way makes them widely adopted in different odometry solutions.

In this paper, we propose a 3D odometry method that relies only on the Doppler velocity to estimate the vehicle motion from a single scan, hence avoiding data association. Doppler-capable range sensors can only capture the radial velocity of a point. Assuming most of the scene is static, the linear velocity of the sensor can be recovered via RANSAC [9]. However, there is a lack of information to obtain the 6 DOFs of a 3D movement. The observability of this problem was studied by Yoon et al. [10], where a gyroscope was added to completely determine the vehicle 3D motion. However, in this work we propose a simpler solution closer to the one presented by Kellner [7] but in 3D: by taking into consideration the kinematic model of the vehicle, we can recover its 3D movement using only the Doppler velocities. In the following sections, we describe the proposed odometry algorithm, as well as the experiments performed to validate it. An overview of its working principles can be seen in Figure 1. The results obtained prove that the presented method yields a short-term accurate motion estimation. When it is employed as initialization of more advanced localization algorithms like mapper pipelines, it improves the accuracy without needing a different sensor mounted on the vehicle. The code is publicly available at [https://github.com/andresgalu/doppler\\_odometry](https://github.com/andresgalu/doppler_odometry).

## II. RELATED WORK

The first publication to recover the ego-motion of a vehicle employing Doppler velocity without data association is due to Yokoo et al. [11], who employ 1D radar sensors mounted on a vehicle with Ackermann steering performing 2D planar motion. They propose two sensor configurations to recover the forward and angular velocities, one with two radars mounted in front of the vehicle, and another with a single radar and a gyroscopic sensor. The method requires the observed objects to be static in order to provide a reliable localization estimation. Kellner et al. expanded this concept for a single 2D radar [7]. Since the sensor can only measure the radial velocity of the objects, the angular velocity cannot be directly observed. To circumvent this issue, again the vehicle has to comply with the Ackermann kinematic model, reducing the DOFs from 3 (complete 2D motion) to 2 (forward and angular velocities). The increase in data points due to leveraging a 2D sensor removes the need to assume a static scene. By applying RANSAC [9], dynamic points can be identified and discarded as outliers. Kellner et al. later developed a method to completely recover the 3 DOFs of the 2D motion by employing multiple radar sensors [8], avoiding the Ackermann model requirement.

Stepping into 3D movement, works of both Kramer et al. [12] and Doer and Trommer [13] combine the information from a 3D radar and an IMU to recover the 6 DOFs

of the motion, employing batch optimization by the former and an Extended Kalman Filter (EKF) by the latter. The use of an IMU as a way to make the movement observable is reminiscent of the second configuration proposed by Yokoo et al. [11]. Similarly, Yoon et al. [10] employ a Doppler-capable 3D lidar and a gyroscope to decouple the 3D motion estimation, with the lidar and gyroscope yielding the translation and rotation respectively. Said work includes a study of the observability of the motion, explaining the need for either a gyroscope or multiple sensors to completely recover the 3D movement. Even without a 3D radar, Park et al. [14] estimate the 3D motion of a vehicle by compositing two orthogonal 2D radar sensors, along with an IMU.

As well as providing a motion estimation, the Doppler velocity has proven to be helpful in more traditional odometry algorithms. They usually register consecutive point clouds, and thus are out of the scope of this review, only those that employ Doppler velocities are included. The most common approach is to add a cost function to the optimization problem, where the measured radial velocity is compared to the expected velocity of a point given an estimation of the motion. This can be applied to registration problems commonly found in the literature such as Iterative Closest Point (ICP) [15] variations [6], [16], [17], Normal Distribution Transform [18] (NDT) [19], and similar approaches [20]. The work of Monaco and Brennan [21] stands out for decoupling the motion estimation: the translation is obtained from the Doppler velocity, while the collected spatial information provides the rotation. The recent rise of Neural Networks (NN) has also reached radar odometry as can be seen in the work of Rennie et al. [22], where Doppler velocity is used to obtain an estimation of the motion which can later be fused with the scan registration result.

In contrast to the literature that is reviewed above, the method proposed in this article provides the 3D motion of a Doppler-capable range sensor by only leveraging the radial velocities from the scene. By taking into consideration the kinematic model of the vehicle the movement can be recovered without needing an extra IMU. The efficiency and short-term accuracy of this algorithm make it more than helpful to provide an estimation of the motion, which is also valuable as initialization in more resource-hungry algorithms.

## III. METHOD OVERVIEW

In this section, we explain how the proposed method estimates the 3D movement of the vehicle from the Doppler velocity measurements of points in the scene. It first estimates the sensor velocity, then finds the vehicle movement that fulfills a certain kinematic model and explains said velocity. In Section III-A we will describe how the velocity of the sensor itself can be calculated from the captured radial velocities. As explained before, from the sensor velocity we cannot obtain the 6 DOFs of the vehicle's 3D motion. By introducing the kinematic model of the vehicle in Section III-B, the DOFs of the movement get reduced and hence it can be solved. Finally, Section III-C will analyze the variance of the estimated vehicle movement variables.

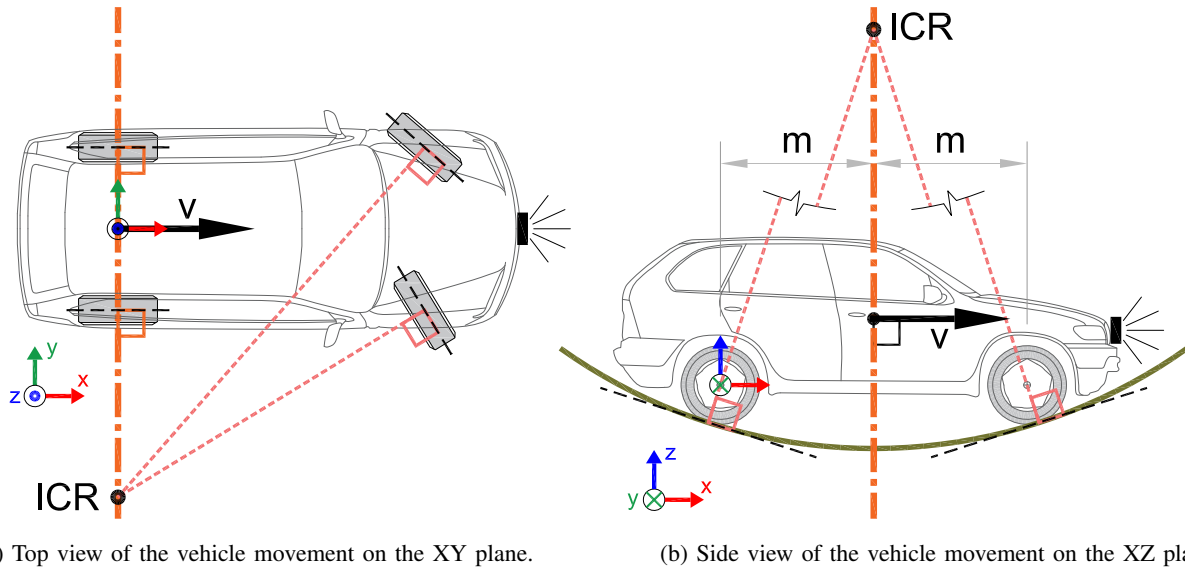


Fig. 2: Simple cases of movement of the vehicle showing the location of the line along which the ICR can be found (orange). The velocity of points located on the ICR axis is perpendicular to it. Black box represents an example of sensor location.

### A. SENSOR VELOCITY

In this section, we will delve into the procedure that yields the sensor velocity from the Doppler velocities of the observed points. First, we will derive the equations assuming the scene is completely static, and then we will tackle the issue of removing dynamic objects.

Assuming a static scene, the velocity of an observed object is the opposite of the sensor velocity  $v_s$ . Since the Doppler velocity from each point  $v_i^D$  is only measured along the radial direction, this value is the result of projecting the opposite of the sensor velocity on the radial direction.

$$-v_i^D = \begin{bmatrix} \cos \phi_i \cos \theta_i & \sin \phi_i \cos \theta_i & \sin \theta_i \end{bmatrix} \begin{bmatrix} v_{sx} \\ v_{sy} \\ v_{sz} \end{bmatrix} \quad (1)$$

The azimuth  $\phi$  and elevation angles  $\theta$  of each point  $i$  are obtained by converting their cartesian coordinates to spherical. Applying Equation (1) to all sensed points results in a set of linear equations in the form of  $B = Ax$  that can be solved with least squares regression. To reduce the influence of noise in the final result, we weight each point by its signal power. The sensor velocity is thus estimated as follows, with  $W$  being the diagonal matrix of the weights.

$$v_s = (A^T W A)^{-1} (A^T W B) \quad (2)$$

In the case there are dynamic objects in the scene, not all the points will follow the same model. In our implementation, we chose RANSAC [9] to get rid of these outlier points that do not comply with the estimated sensor velocity. They are labeled as dynamic, which can be helpful for later processing of the data.

### B. VEHICLE KINEMATIC MODEL

Now that the sensor velocity has been calculated from the Doppler velocity measurements, we can estimate the

vehicle movement. There exists an infinite number of 3D motions that explain the calculated sensor velocity, and we lack the information to determine all the 6 DOFs. However, by constraining the vehicle motion to a certain kinematic model, it is possible to simplify the movement enough to obtain an estimation.

First of all, we need to define the frame of reference used throughout this section. Its position in the vehicle is not particularly relevant, but its orientation is. As shown in Figure 2 we establish the Y axis to be aligned with the rear wheel axis, pointing to the left of the vehicle. The X axis is perpendicular to it and points toward the front wheel axis. Finally, the Z axis points upwards, perpendicular to both X and Y. Without loss of generality, we will locate the frame of reference at the center of the rear wheel axis.

For the sake of clarity, we will start analyzing the movement of an Ackermann steering vehicle moving along a flat surface, which is simply a 2D motion. In that case, the Instant Center of Rotation (ICR) can be found somewhere along the line of contact between the rear wheels and the ground, but it can be well approximated as the line that goes through the rear wheel axis. From now on we will refer to this line as the Z-ICR axis, since it relates to the rotation around Z. The velocity along the Y direction of any point located in the Z-ICR axis is null by definition. See Figure 2a for a graphic representation. Note that the key concept here is to have the ICR restricted to a given line, and the Ackermann kinematic model is just an example. As long as the kinematic model fulfills this requirement, like in differential drive and skid-steer vehicles, the viability of the proposed method holds.

For a 3D motion, the movement along Z has to be analyzed too. Similar to the previous paragraph, we can first study the simplified movement where the vehicle only moves in the XZ plane, along a curved surface. The ICR in this case is located at the intersection of the lines perpendicular to the contact

point between the wheels and the ground. Assuming the curvature of the ground is locally constant in the relatively short distance between both wheel axes, the ICR will be located in a line parallel to the Z axis that goes through the midpoint between both wheel axes. Every point located along this Y-ICR axis will consequently have zero velocity in the Z direction. Figure 2b shows this concept for better clarity. This holds true even when driving on a flat plane, in which case every point in the vehicle has zero vertical velocity.

Until now, we have studied the rotation around the Z axis (yaw) and Y axis (pitch) through two basic movement examples (Figures 2a and 2b respectively). The roll, or rotation around the X axis, can be assumed negligible for a vehicle moving on a road. The complete 3D motion can then be seen as a combination of these two basic movements, and the introduced restrictions will still be present in the resulting 3D motion model. Namely, the velocity of points located in the Z-ICR axis will have no Y component, and similarly points in the Y-ICR (the middle of the vehicle along the X axis) will have zero Z velocity.

We can now recover the motion of the vehicle by applying the mentioned restrictions, knowing that the velocities of two points  $\vec{p}, \vec{s}$  of the same rigid moving object are related through their relative position and angular velocity  $\vec{\omega}$ :

$$\vec{v}_p = \vec{v}_s + \vec{\omega} \times (\vec{p} - \vec{s}) \quad (3)$$

Using the previously estimated sensor velocity  $\vec{v}_s$  (2), along with the restrictions, we can recover the angular velocity vector. To that end, point  $\vec{s}$  is now the sensor position, and point  $\vec{p}$  belongs to the Z-ICR axis, fulfilling the restriction of having zero Y velocity. Since all points on said axis comply with the requirement, we choose a point  $\vec{p}_a$  located at the same Y coordinate as the sensor, for simplicity.

$$\begin{cases} \vec{p}_a = [0, s_y, 0]^\top \\ \vec{v}_a = [v_{ax}, 0, v_{az}]^\top \end{cases} \rightarrow 0 = v_{sy} - \omega_z s_x + \omega_x s_z \quad (4)$$

In a similar manner, we can apply the second restriction, which indicates that a point  $\vec{p}_b$  located on the Y-ICR (in the middle of the vehicle) has zero Z velocity.

$$\begin{cases} \vec{p}_b = [m, 0, s_z]^\top \\ \vec{v}_b = [v_{bx}, v_{by}, 0]^\top \end{cases} \rightarrow 0 = v_{sz} - \omega_x s_y - \omega_y (m - s_x) \quad (5)$$

Where  $m$  is half the distance between the two wheel axes of the vehicle, see Figure 2b. Finally, we apply the restriction  $\omega_x = 0$  since the rotation around the X axis is negligible, and thus the angular velocity vector can be recovered.

$$\vec{\omega} = [0 \quad v_{sz}/(m - s_x) \quad v_{sy}/s_x]^\top \quad (6)$$

The global pose of the sensor with respect to the reference frame of the world can be updated from the previous instance after a period of time  $\Delta t$  by employing the velocity  $\vec{v}_s$  for the position  $\vec{p}_s$ , and the angular velocity  $\vec{\omega}$  for the orientation  $R$ . Note that the velocity of any point of the vehicle can be calculated by applying (3), and thus its pose can be updated:

$$\begin{cases} R(t + \Delta t) &= R(t)e^{\vec{\omega}\Delta t} \\ \vec{p}_s(t + \Delta t) &= \vec{p}_s(t) + R(t)\vec{v}_s\Delta t \end{cases} \quad (7)$$

### C. COVARIANCE ANALYSIS

In this section, we analyze the variance of the estimated sensor and angular velocities, in order to better understand how their accuracy can be improved. First, we start with the sensor velocity  $\vec{v}_s$  estimated in Section III-A. Since we employed least squares (2) to find the velocity, its covariance matrix  $C_v$  can be estimated from the residuals  $\rho = A\vec{v}_s - B$  [23], with  $N$  being the rows of  $A$ :

$$C_v = \frac{\rho^\top W \rho}{N - 3} (A^\top W A)^{-1} \quad (8)$$

Other than the noise of the measurements, the distribution of the points in the scene is highly significant. The more spread they are, the higher the accuracy of the sensor velocity will be. The covariance matrix  $C_\omega$  of the angular velocity vector can then be calculated via error propagation:

$$C_\omega = J C_v J^\top = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \sigma_{vz}^2 / (m - s_x)^2 & \sigma_{vyz}^2 / (s_x m - s_x^2) \\ 0 & \sigma_{vyz}^2 / (s_x m - s_x^2) & \sigma_{vy}^2 / s_x^2 \end{bmatrix} \quad (9)$$

Note the impact the X-location of the sensor on the vehicle has on the covariance. If the sensor's X coordinate coincides with that of either ICR axis, then the related angular velocity cannot be observed. Thus, for better accuracy it is advisable to place the sensor far from these axes along the X direction.

## IV. EXPERIMENTS AND RESULTS

In this section, we describe the experiments performed in order to validate the proposed method, and subsequently evaluate it based on the obtained results. The test platform used is a standard PC with 8GB of RAM, running Ubuntu 20.04 with an Intel Core i7-7700HQ CPU with four cores and eight execution threads at 2.8 GHz. The employed setup consists of a Hugin 4D imaging radar, an Ouster OS1-128 3D lidar, and an Xsens MTi-30 IMU all mounted on a Clearpath Husky robotic base. The proposed method makes use only of the radar, while the lidar and IMU are used here for comparison in the evaluation. A dataset was collected while the Husky was driven around sloped terrain. Six different sequences were recorded: 04 and 05 have the most hills and thus the most vertical movement; 06 represents driving on a flat ground; 07 and 08 are similar multifaceted trajectories, with the second having a higher velocity throughout; and 11 is a longer route. Both the environment and the sensor setup can be seen in Figure 3, and the dataset is publicly available at <https://zenodo.org/record/8346769>. Note how the sensor is mounted on a lever arm on the vehicle. This is because increasing the distance along X from the sensor to both ICR axes results in a more accurate angular velocity, as established in Section III-C. On a larger vehicle, like a car, the sensor can be mounted far from the ICR without needing a lever arm, as can be seen in Figure 2.

In contrast to the Ackermann vehicle used as an example in Section III-B, the Husky is a skid-steer vehicle. As explained before, the proposed method works with every vehicle whose Z-ICR is in a consistent location. We are aware of the implications of working with this type of vehicle

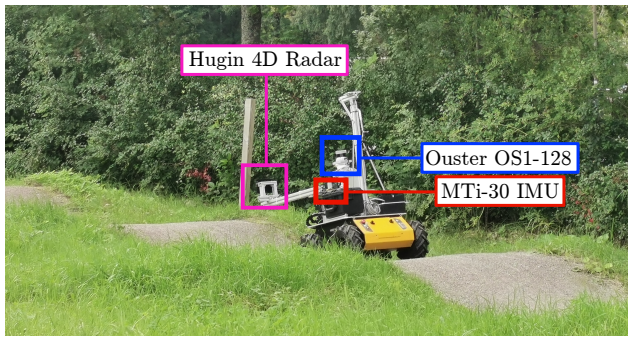


Fig. 3: Sensor setup and test environment from sequence 04.

[24], but the results show that the kinematic model is accurate enough to recover the angular velocity.

#### A. SENSOR CALIBRATION

For the proposed method to properly work, the sensor orientation and position with respect to the vehicle must be known. To that end, we propose a two-step calibration algorithm. First, the vehicle moves in a straight line, forward and backward. We then find the rotation that minimizes both Y and Z velocities calculated by the method, since only the X component should be non-zero. The second step is similar, but in this case the rotation required is the one around the X axis, which is not determined by the previous step. Therefore, the vehicle moves freely around a flat surface, to then find the rotation that minimizes the estimated Z velocity. By combining both rotations, the calculated sensor velocity can be transformed into the reference frame of the vehicle, and thus be used to obtain the angular velocities.

Just as important as the orientation of the sensor is its position, in particular, the distance along the X axis from the sensor to the location of the Z-ICR. This distance can be measured, but to avoid the possible error introduced by manually measuring it, we decided to calibrate  $s_x$  employing the IMU data. The procedure is simple: the vehicle moves in a flat circle at a constant speed, and then  $s_x$  is chosen to minimize the difference between the calculated angular velocity and the one measured by the IMU. Note that this is not the only possible way to calibrate the sensor position, but information about the movement is required either way.

#### B. ACCURACY RESULTS

In this section we will evaluate the accuracy of the proposed method for providing single-scan and single-sensor 3D motion estimation. We compare the results to an IMU, since both provide a short-term accurate estimation of the motion, but show drift over time. As the ground truth for the comparison, we use the trajectory obtained by running an ICP-based mapper on the 3D lidar data.

An IMU provides information on the angular velocity and linear acceleration of the sensor. Its trajectory can be estimated by integrating these values over time. However, these measurements suffer from having noise and a non-constant bias. Combined with the high frequency at which these sensors operate, said trajectory shows significant drift

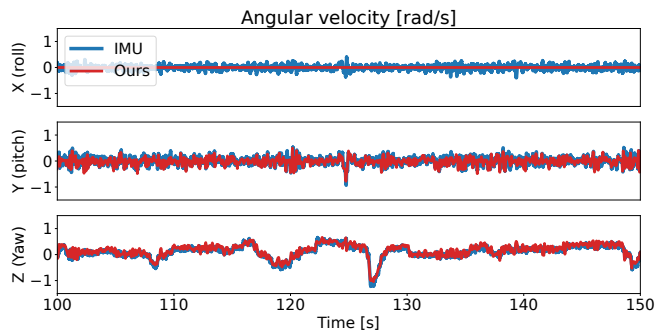


Fig. 4: Angular velocity estimated from the Doppler velocities compared to IMU measurements, from sequence 11.

over time. The position is impacted even more since two integration steps are performed from the measured linear acceleration [25]. Therefore, the translation estimated from the IMU is not reliable enough to be included in the results.

Our method, in contrast to the IMU integration, obtains the translation of the sensor directly from its estimated velocity. The rotation, however, depends also on the calibration previously mentioned, as well as the fulfillment of the kinematic hypothesis. Our simple approach allows for fast execution, taking on average  $4.50 \pm 1.94$  ms per scan.

Figure 4 shows the angular velocities estimated by the proposed method compared to those measured by the IMU. Despite the mentioned limitations, our method provides an estimation similar to the IMU. Notice also how the roll velocity measured by the IMU is not as significant as the yaw and pitch. It is normal to undergo some roll rotation in the rough terrain of the experiments, but it is small enough to consider it inconsequential. On smoother surfaces, like roads, it will be even smaller. This supports the restriction imposed by the kinematic model on the roll.

It makes sense to combine the advantages of both approaches into one, namely the translation from Doppler velocities with the rotation from the IMU, as done by Doer and Trommer [13]. However, this combined method needs information from two different sensors to recover the motion.

To compare these three approaches (IMU, Doppler, IMU + Doppler), we include in Figure 5 the resulting Relative Pose Error (RPE) per frame of the test sequences. This evaluates the short-term accuracy of the different methods, without taking into account their drift over time. Regarding the translation, IMU-only has been omitted because of its large error as discussed above, with the trajectory being hundreds of meters away from the ground truth. Both Doppler-only and IMU + Doppler recover the translation in a similar manner, and thus it makes sense for them to provide comparable accuracy. The difference is more apparent in the rotation. The accuracy of our method depends not only on the noise of the input data, unlike IMU. The fulfillment of the kinematic model and the calibration play an important role, hindering the rotation estimation. Sequences 04, 08 and 11 stand out as the most challenging because of their uneven terrain, causing more vibrations and defying the no-roll assumption.

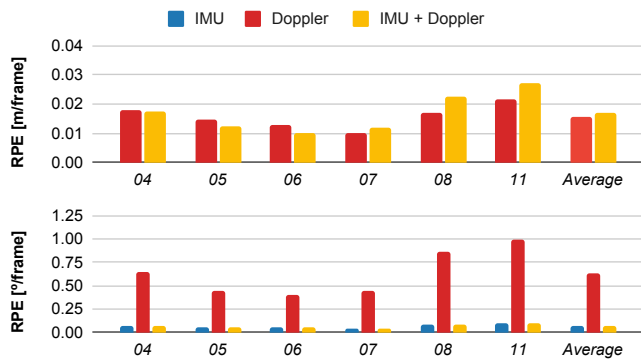


Fig. 5: Translational and rotational RPE per frame.

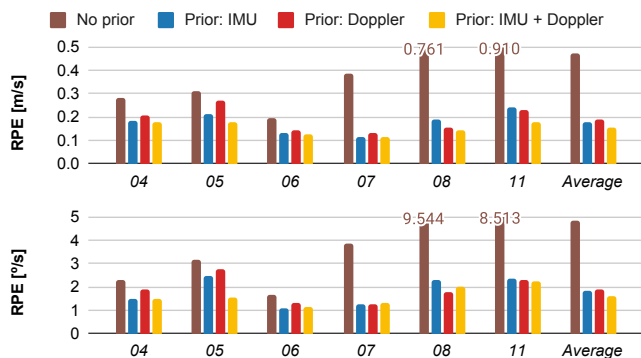


Fig. 6: RPE per second of the mapper employing different prior data. Values out of the graph are labeled.

### C. INITIALIZATION OF SLAM

The proposed method’s short term accuracy and fast execution make it an interesting candidate to provide a motion prior to a more complex and accurate localization method, e.g. SLAM. ICP [15] is a widespread point cloud registration algorithm used among SLAM methods but is prone to falling into local minima. By providing a motion prior this limitation can be greatly reduced. It is already fairly common in the literature to employ an IMU to initialize the motion.

To evaluate the impact of using the proposed method as initialization for SLAM, we feed the radar data and a motion prior to an ICP-based mapper<sup>1</sup> [26] and analyze its accuracy based on how the motion is initialized. We test 3 different prior estimators: IMU only, Doppler only (our method), and IMU combined with radar. The trajectory of the mapper without any prior is also included as the base case in the comparison. Again, the ground truth used for the comparison is the one obtained from running the mapper on the 3D lidar data. The evaluation metric in this case is the RPE per second, which focuses more on long-term accuracy, a desirable trait for SLAM algorithms.

The resulting error values of the experiments can be seen in Figure 6, and the trajectories in Figure 7. Despite the variation along different sequences, all methods perform at a similar level of accuracy. What is obvious is the need

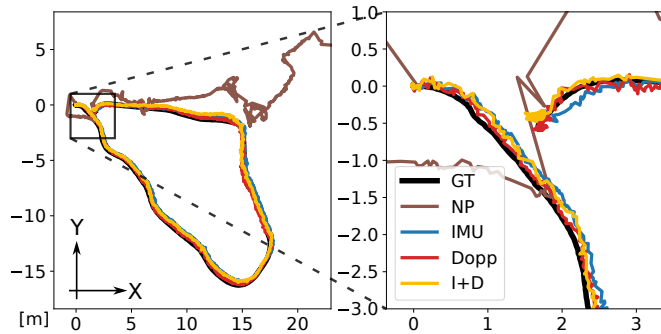


Fig. 7: Trajectory in meters generated by the mapper with different prior data on sequence 07 (left), and a closer view of the start and finish points (right). Prior used: none (NP), IMU, our method (Dopp), and IMU and Doppler (I+D).

for a prior input to the mapper, without which it fails. Our method provides a reliable motion prior, resulting in a similar accuracy to IMU, without the need to mount and calibrate an extra sensor. Furthermore, the proposal can identify and remove dynamic points from the scene, which can help at later stages of the mapper pipeline.

### V. CONCLUSIONS

In this paper, we have introduced a fast 3D odometry that estimates the motion of a vehicle from a single scan leveraging the Doppler velocity. The method first obtains the sensor velocity based on the measured radial velocities of points from the scene, while rejecting dynamic objects. Based on the kinematic model of the vehicle, the relation between its velocity and the sensor’s is established, and thus the 3D motion of the vehicle can be recovered.

A series of experiments have been carried out to validate our method, evaluating the accuracy of the odometry both when used as-is for pose tracking and when used as input to an ICP-based mapping pipeline. We compared its performance with an IMU, given their similar ability to efficiently provide short-term accurate odometry. These qualities make the proposal especially suitable as initialization for radar-only SLAM algorithms, as it boosts the accuracy and identifies dynamic objects at a very low cost. The results prove how our method provides a similar accuracy to an IMU without needing an external sensor mounted on the vehicle. Both the code and the dataset are publicly available.

At the core of the method lies the kinematic model of the vehicle. Granted that it simplifies the movement enough to be observable with a single sensor, it also sets the limitations needed for the proposal to accurately estimate the motion. Challenging scenarios include the Z-ICR being unstable, for example in slippery terrain, bumpy roads with abrupt and uneven vertical variation (potholes), and roads with considerable cant that generate roll rotation. These issues cannot be solved solely through software. On the other hand, the impact of having large amounts of outliers or noisy measurements on the accuracy can be reduced by correctly filtering the data, which should be the focus for future work.

<sup>1</sup>[https://github.com/norlab-ulaval/norlab\\_icp\\_mapper\\_ros](https://github.com/norlab-ulaval/norlab_icp_mapper_ros)

## REFERENCES

- [1] A. Carballo, J. Lambert, A. Monrroy, D. Wong, P. Narksri, Y. Kit-sukawa, E. Takeuchi, S. Kato, and K. Takeda, "LIBRE: The multiple 3d lidar dataset," in *2020 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2020, pp. 1094–1101.
- [2] M. Mielle, M. Magnusson, and A. J. Lilienthal, "A comparative analysis of radar and lidar sensing for localization and mapping," in *2019 European Conference on Mobile Robots (ECMR)*. IEEE, 2019, pp. 1–6.
- [3] A. Venon, Y. Dupuis, P. Vasseur, and P. Merriaux, "Millimeter wave fmcw radars for perception, recognition and localization in automotive applications: A survey," *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 3, pp. 533–555, 2022.
- [4] Z. Hong, Y. Petillot, A. Wallace, and S. Wang, "RadarSLAM: A robust simultaneous localization and mapping system for all weather conditions," *The International Journal of Robotics Research*, vol. 41, no. 5, pp. 519–542, 2022.
- [5] A. Kingery and D. Song, "Improving ego-velocity estimation of low-cost Doppler radars for vehicles," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9445–9452, 2022.
- [6] B. Hexsel, H. Vhavle, and Y. Chen, "DICP: Doppler iterative closest point algorithm," *arXiv preprint arXiv:2201.11944*, 2022.
- [7] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using Doppler radar," in *16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013)*. IEEE, 2013, pp. 869–874.
- [8] D. Kellner, M. Barjenbruch, J. Klappstein, J. Dickmann, and K. Dietmayer, "Instantaneous ego-motion estimation using multiple Doppler radars," in *2014 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2014, pp. 1592–1597.
- [9] M. A. Fischler and R. C. Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Communications of the ACM*, vol. 24, no. 6, pp. 381–395, 1981.
- [10] D. J. Yoon, K. Burnett, J. Laconte, Y. Chen, H. Vhavle, S. Kammel, J. Reuther, and T. D. Barfoot, "Need for speed: Fast correspondence-free lidar odometry using Doppler velocity," *arXiv preprint arXiv:2303.06511*, 2023.
- [11] K. Yokoo, S. Beauregard, and M. Schneider, "Indoor relative localization with mobile short-range radar," in *VTC Spring 2009-IEEE 69th Vehicular Technology Conference*. IEEE, 2009, pp. 1–5.
- [12] A. Kramer, C. Stahoviak, A. Santamaria-Navarro, A.-A. Agha-Mohammadi, and C. Heckman, "Radar-inertial ego-velocity estimation for visually degraded environments," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 5739–5746.
- [13] C. Doer and G. F. Trommer, "An EKF based approach to radar inertial odometry," in *2020 IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2020, pp. 152–159.
- [14] Y. S. Park, Y.-S. Shin, J. Kim, and A. Kim, "3d ego-motion estimation using low-cost mmwave radars via radar velocity factor for pose-graph slam," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7691–7698, 2021.
- [15] P. J. Besl and N. D. McKay, "Method for registration of 3-d shapes," in *Sensor fusion IV: control paradigms and data structures*, vol. 1611. International Society for Optics and Photonics, 1992, pp. 586–606.
- [16] K. Retan, F. Loshaj, and M. Heizmann, "Radar odometry on  $SE(3)$  with constant velocity motion prior," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 6386–6393, 2021.
- [17] Y. Wu, D. J. Yoon, K. Burnett, S. Kammel, Y. Chen, H. Vhavle, and T. D. Barfoot, "Picking up speed: Continuous-time lidar-only odometry using Doppler velocity measurements," *IEEE Robotics and Automation Letters*, vol. 8, no. 1, pp. 264–271, 2022.
- [18] P. Biber and W. Strasser, "The normal distributions transform: a new approach to laser scan matching," in *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2003) (Cat. No.03CH37453)*, vol. 3, 2003, pp. 2743–2748 vol.3.
- [19] M. Rapp, M. Barjenbruch, M. Hahn, J. Dickmann, and K. Dietmayer, "Probabilistic ego-motion estimation using multiple automotive radar sensors," *Robotics and Autonomous Systems*, vol. 89, pp. 136–146, 2017.
- [20] M. Barjenbruch, D. Kellner, J. Klappstein, J. Dickmann, and K. Dietmayer, "Joint spatial- and Doppler-based ego-motion estimation for automotive radars," in *2015 IEEE Intelligent Vehicles Symposium (IV)*. IEEE, 2015, pp. 839–844.
- [21] C. D. Monaco and S. N. Brennan, "Radarodo: Ego-motion estimation from Doppler and spatial data in radar images," *IEEE Transactions on Intelligent Vehicles*, vol. 5, no. 3, pp. 475–484, 2020.
- [22] F. Rennie, D. Williams, P. Newman, and D. De Martini, "Doppler-aware odometry from FMCW scanning radar," *arXiv preprint arXiv:2308.10597*, 2023.
- [23] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, 2009, vol. 2.
- [24] J. L. Martínez, J. Morales, J. M. García, and A. García-Cerezo, "Analysis of tread ICRs for wheeled skid-steer vehicles on inclined terrain," *IEEE Access*, vol. 11, pp. 547–555, 2022.
- [25] O. J. Woodman, "An introduction to inertial navigation," University of Cambridge, Computer Laboratory, Tech. Rep., 2007.
- [26] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat, "Comparing ICP Variants on Real-World Data Sets," *Autonomous Robots*, vol. 34, no. 3, pp. 133–148, Feb. 2013.