

Fusing Vision and Odometry for Accurate Indoor Robot Localization

Bastian Bischoff, Duy Nguyen-Tuong, Felix Streichert, Marlon Ewert
Robert Bosch GmbH, Corporate Research
Robert-Bosch-Str. 2, 71701 Schwieberdingen, Germany

Alois Knoll
TU Munich, Robotics and Embedded Systems
Boltzmannstr. 3, 85748 Munich, Germany

Abstract—For service robotics, localization is an essential component required in many applications, e.g. indoor robot navigation. Today, accurate localization relies mostly on high-end devices, such as A.R.T. DTrack, VICON systems or laser scanners. These systems are often expensive and, thus, require substantial investments. In this paper, our focus is on the development of a localization method using low-priced devices, such as cameras, while being sufficiently accurate in tracking performance. Vision data contains much information and potentially yields high tracking accuracy. However, due to high computational requirements vision-based localization can only be performed at a low frequency. In order to speed up the visual localization and increase accuracy, we combine vision information with robots odometry using a Kalman-Filter. The resulting approach enables sufficiently accurate tracking performance (errors in the range of few cm) at a frequency of about 35Hz. To evaluate the proposed method, we compare our tracking performance with the high precision A.R.T. DTrack localization as ground truth. The evaluations on real robot show that our low-priced localization approach is competitive for indoor robot localization tasks.

I. INTRODUCTION

Localization is employed to estimate the position and orientation of a mobile robot given a map of the surroundings. There has been a lot of research activities in this field, as localization is a crucial component for many robot applications, such as navigation and autonomous driving [1], [2], [3]. Typically, localization methods require accurate sensor information, for example, from laser scanners for outdoor applications [3] or using infrared-based systems such as DTrack and VICON for indoor navigation [4]. Localization needs to be performed accurately at a high frequency. DTrack, for example, is able to track the mobile robot at a rate of 60Hz with an accuracy of 1mm. However, the major drawback of such infrared-based systems is the limited tracking area (e.g. $3 \times 3\text{m}^2$ for DTrack) and the high investment cost (e.g. several ten thousand € for our improved DTrack system). In this paper, we investigate possibilities for mobile robot localization using off-the-shelf vision cameras instead of expensive high-end devices.

In recent years, vision cameras have become more important in robot applications [5], as they provide rich information while requiring little investments (starting from 100 USD). However, a disadvantage of vision are high computational requirements resulting in a low tracking frequency, e.g. 3 – 7Hz, which is not sufficient for many robot applications, such as indoor navigation. In this work, the key idea is to combine the vision data with other robot's sensor signals, e.g. odometry, in order to obtain information which enables accurate localization at

higher tracking frequencies. Odometry is a natural choice as it is available in most mobile robot systems, where sensors are used for monitoring the rotation of the robot's wheels. Odometry can be employed for pose estimation at a high frequency. However, in opposite to localization methods based on vision or laser data, odometry allows only a relative localization, i.e. it computes the movements relative to the last pose. The consequence is that the estimation errors are accumulated over time and, thus, the odometry localization becomes inaccurate in the long run.

Our approach is to merge vision and odometry information using a Kalman-Filter while combining the strengths of both methods, i.e. the high tracking frequency of odometry and the accuracy of vision. Intuitively, the vision information is used to correct the odometry estimations making the resulting localization more accurate and yet fast. To evaluate our approach, we use a version of the Festo Robotino with a Guppy PRO F-125 camera which is mounted on top of the robot (see Figure 1). We compare the localization results of our approach, the Kalman-filtered Vision and Odometry localization (KVO), with the highly accurate DTrack system as ground truth.

The remainder of this paper is organized as follows. Section II gives an overview about localization with odometry and vision, respectively. In Section III-B, we explain our KVO localization method in details. We extensively evaluate KVO on the robot platform Robotino in Section IV, followed by a conclusion and an outlook in the last Section. V.

II. LOCALIZATION METHODS

In this section, we describe the vision and odometry localization methods for our mobile robotics platform Robotino. The pose of the robot will be given as (x, y, θ) , where x, y describe the position and θ the orientation. The goal of localization is to estimate this pose in a global coordinate frame.

A. Odometry Localization

Odometry estimates the robot's position and orientation by monitoring the robot's drive. For most drives, this information comes out-of-the-box and is published at a high frequency. Hence, odometry can be seen as a localization method that comes for "free". Odometry computes changes in pose, which



Fig. 1. Service robot platform Robotino with omni-directional drive used for evaluation. On top there are camera and markers required for the reference positioning system Dtrack.

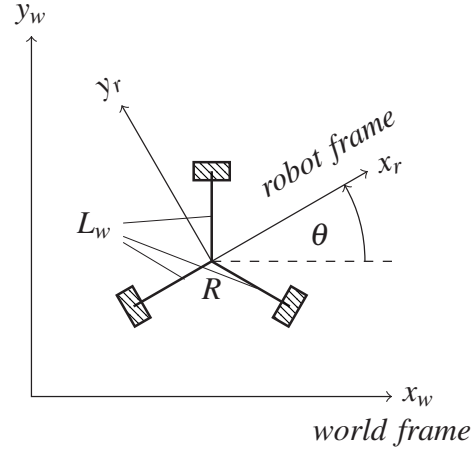


Fig. 2. The figure shows a three wheeled omni-directional drive in a world coordinate frame. The shaded rectangles symbolize the wheels, the point R is the center of mass.

need to be integrated in order to obtain the robot's pose estimation. However, one disadvantage of such relative localization methods is the increasing pose error over time.

The Robotino, as used in this work, has an omni-directional drive consisting of three mecanum wheels. Figure 2 shows the setup of the Robotino drive. The wheels are aligned symmetrically with a displacement of 120° , the engines allow movement speed of about 2.8 m/s. Incremental encoders are attached to each engine and measure the rotations of the electric engine. We denote the rotation speeds of the three wheels by $\dot{q} = [\dot{q}_1, \dot{q}_2, \dot{q}_3]^T$, L_w is the distance between each wheel and the center of mass R . With the rotation speeds \dot{q} , the pose change $[\dot{x}, \dot{y}, \dot{\theta}]^T$ can be calculated as

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \frac{2}{3} \cdot \cos(\theta + \delta) & -\frac{2}{3} \cdot \cos(\theta - \delta) & \frac{2}{3} \sin \theta \\ \frac{2}{3} \cdot \sin(\theta + \delta) & -\frac{2}{3} \cdot \sin(\theta - \delta) & -\frac{2}{3} \cos \theta \\ \frac{1}{3L_w} & \frac{1}{3L_w} & \frac{1}{3L_w} \end{bmatrix} \cdot \dot{q}, \quad (1)$$

see [6]. The parameter δ is the wheel orientation in the robot's coordinate system (30° in our case). To estimate a global pose, all pose changes $[\dot{x}, \dot{y}, \dot{\theta}]^T$ need to be integrated from a given starting point (x_0, y_0, θ_0) .

B. Visual Localization

There are many methods for visual localization [7], [8], [9], [10]. Vision allows an accurate global localization with low cost hardware. However, high computational efforts for vision generally lead to a low tracking frequency. Another drawback of vision is the time-delay problem, i.e. since it takes time to compute a pose estimation from a camera picture, the resulting estimation corresponds to a robot's pose in the past. In practice, the resulting time-delay can deteriorate the localization performance. In this work, we employ a standard

off-the-shelf monovision camera. For visual localization, artificial black-and-white markers are placed on the ceiling to achieve high robustness and tracking accuracy. The markers are squares with 10cm edge length. The $10 \times 10\text{cm}^2$ square is divided into a 4×4 black-and-white pattern, resulting in 16456 possible markers. The positions and orientations of the markers in the world-frame are measured beforehand. Subsequently, transformations between robot's pose relative to the markers and pose in the global world-frame can be made.

We use the ARToolKit library [11] to track the visual markers. Here, 30 markers are employed to cover an area of about 15 m^2 . The tool searches for the black borders of markers in the camera picture. A detected marker is reduced to a 16×16 pixel image and can then be compared to the set of known patterns. Since the marker dimensions are fixed, the pose of the marker in the camera frame can be calculated. For localization, we invert this transformation to get the pose of the robot in the marker frame. This pose is subsequently transformed to a global pose estimation of the robot in the world-frame. Accordingly, every marker in the camera field of view gives an estimation of the robot pose. We take the mean of all these pose estimations to average out the noise.

III. KALMAN-FILTERED VISION AND ODOMETRY

In this section, we describe a method to fuse the information of vision and odometry for localization using a Kalman-Filter. The approach unifies the strengths of both methods, i.e. the high tracking frequency of odometry and the accuracy of vision.

A. Kalman-Filter

Kalman-Filters are a widely used method to estimate a state s_t given noisy measurements from different sensors, for more

details see [12]. Here, the belief state $\hat{s}_t \sim \mathcal{N}(\mu_t, \Sigma_t)$ at time t is represented as a normal distribution. The system is given by the equations

$$\begin{aligned} s_{t+1} &= A_t \cdot s_t + B_t \cdot u_t + \varepsilon_t \\ z_t &= C_t s_t + \gamma_t, \end{aligned}$$

where A_t, B_t, C_t describe the system dynamics, u_t is the control vector and z_t a state observation. The noise is modelled as $\varepsilon_t \sim \mathcal{N}(0, R_t)$ and $\gamma_t \sim \mathcal{N}(0, Q_t)$, the matrices R_t and Q_t describe the noise property of the system and of state observations, respectively. The setting of these uncertainties strongly influences the behavior of the Kalman-Filter. The Kalman-Filter has two operations, i.e. predict and update:

$$\begin{array}{l} \text{Predict:} \\ \mu_{t+1} = A_t \cdot \mu_t + B_t \cdot u_t \\ \Sigma_{t+1} = A_t \cdot \Sigma_t \cdot A_t' + R_t \end{array} \quad (2)$$

$$\begin{array}{l} \text{Update:} \\ \mu_{t+1} = \mu_t + K \cdot (z_t - C \cdot \mu_t) \\ \Sigma_{t+1} = (I - K \cdot C) \cdot \Sigma_t \end{array} \quad (3)$$

The matrix $K = \Sigma_t \cdot C' \cdot (C \cdot \Sigma_t \cdot C' + Q_t)^{-1}$ is called *Kalman gain* and I is the identity matrix. The prediction approximates the next belief state $\hat{s}_{t+1} \sim (\mu_{t+1}, \Sigma_{t+1})$ if an action u_t is applied in the belief state $\hat{s}_t \sim \mathcal{N}(\mu_t, \Sigma_t)$. This operation increases the uncertainty in the belief state. The update on the other hand is a correction, which decreases the variance of the new belief state. Here, the new belief state \hat{s}_{t+1} is inferred with help of a state observation z_t and the old belief state \hat{s}_t .

B. KVO – Filtering Vision and Odometry

Kalman-filtering vision and odometry (KVO) is an approach to combine the benefits of both sensors. Here, the vision pose estimation can be seen as a correction for the odometry estimation. On the other hand, the odometry information can be employed to overcome time-delays appeared in vision, as well as for regions where no visual markers are available. Furthermore, faulty visual marker detection often leads to corrupt pose estimations. In our KVO approach, the faulty marker recognition can be identified using the uncertainties returned by the Kalman-Filter.

For the KVO localization, the state $s_t = (x_t, y_t, \theta_t)$ is three dimensional representing the robot position x, y and orientation θ . The belief state $\hat{s}_t \sim \mathcal{N}(\mu_t, \Sigma_t)$ of the Kalman-Filter returns the estimation of the robot's pose with a mean μ_t and variance Σ_t . Using the Kalman-Filter, the operations *predict* and *update* need to be defined. Here, the matrices A_t, B_t in equation (2) are set to identity. With the odometry information, pose changes $(\Delta x, \Delta y, \Delta \theta)$ can be calculated using the equation (1). This pose change $u_t = (\Delta x_t, \Delta y_t, \Delta \theta_t)$ – computed from odometry information – is subsequently employed for the Kalman-Filter prediction. For the update operation in equation (3) on the other hand, the vision-based localization from Section II-B gives global pose estimates z_t , the matrix C_t is set to identity.

Using the KVO localization, the predict operation of the Kalman-Filter is executed when a new odometry pose estimation is available. Thus, the estimation of the robot's state can

Algorithm 1 The KVO algorithm

```

 $\mu, \Sigma$                                  $\triangleright$  belief state mean and variance
 $R, Q$                                      $\triangleright$  noise of odometry and vision
buffer                                     $\triangleright$  buffer of last predictions
function ON_NEW_LOCALIZATION_INFO
  if is_odometry  $u$  then                                 $\triangleright$  predict
     $u = (\Delta x, \Delta y, \Delta \theta)$ 
     $\mu \leftarrow \mu + u$ 
     $\Sigma \leftarrow \Sigma + R$ 
    PUT_U_IN_BUFFER()
  else if is_vision  $z$  then                                 $\triangleright$  update
    ALIGN_Z_WITH_BUFFER_DATA()
    Compute  $\sigma_\theta$  and  $\sigma_{xy}$ 
    if  $\sigma_\theta$  or  $\sigma_{xy}$  is smaller than 2 then
       $K \leftarrow \Sigma + (\Sigma + Q)^{-1}$ 
       $\mu \leftarrow \mu + K \cdot (z - \mu)$ 
       $\Sigma \leftarrow (I - K) \cdot \Sigma$ 
    end if
  end if
end function

```

be performed at a high frequency, as the odometry estimation is fast. The update step of the Kalman-Filter is executed when a new pose estimation is obtained from the visual localization. The update step is employed to correct the estimations done by the odometry. As shown by the results in Section IV, the update steps of KVO can be performed at much lower frequency than the prediction steps.

However, visual localization is prone to outliers. In order to detect these outliers, we define the following criteria

$$\sigma_\theta = \frac{|d_\theta|}{\sqrt{\Sigma_\theta}} \quad \text{and} \quad \sigma_{xy} = \sqrt{\left(\frac{d_x}{\sqrt{\Sigma_x}}\right)^2 + \left(\frac{d_y}{\sqrt{\Sigma_y}}\right)^2}, \quad (4)$$

where $d = \hat{s} - z = (d_x, d_y, d_\theta)^T$ is the difference between the estimations (\hat{s} is the belief state and z is the pose estimation from the visual localization). The variance in the belief state is denoted Σ_x and Σ_y for the position, and Σ_θ for the orientation. This variance is naturally given by the Kalman uncertainty, i.e. $\Sigma = (\Sigma_x, \Sigma_y, \Sigma_\theta)^T$. The estimate z is considered to be an outlier, if σ_θ or σ_{xy} is larger than 2. Intuitively, z is considered outlier, if the estimation is outside the ellipsoid defined by the double standard deviations Σ_x and Σ_y .

Time-delay of the visual localization is also an issue which needs to be tackled. In our robot system, for example, the vision time-delay is about 300ms which is measured beforehand. In order to overcome this problem, the past odometry information is saved in a buffer. When obtaining a vision estimation, this information is aligned to the robot's pose in the past. Thus, the Kalman belief state now corresponds to the (past) robot's pose and the update can be performed. The complete KVO algorithm is show in Algorithm 1.

IV. EVALUATION

In this section, we evaluate the KVO localization approach and compare it with the A.R.T DTrack system as ground truth. For the evaluation, different movement patterns are performed on the Robotino (see Figure 1) which is localized using odometry, vision, KVO and DTrack.

For the vision localization, the markers are placed on a grid with 70cm edge length and all markers have the same orientation. This results in 2-3 markers in the camera frame for most of the time. For the localization task, we have hard environment conditions. The lamps on the ceiling are turned on, which increases the difficulty of correct marker detection. Our robot has been used for several years, hence mechanical wear and tear becomes noticeable. The robot also has a high, unsymmetrical setup which displaces the center of mass unfavorably. This leads to inaccurate odometry information. In some cases, vision pose estimates are faulty because a marker has been confused with another marker, or a structure on the ceiling has been interpreted as visual marker by mistake. Such faulty vision pose estimation are detected using the state uncertainty of the Kalman-Filter. If the position or orientation of the vision estimation is outside of the 2σ region, as described in Section III-B, the estimation is assumed to be corrupted and ignored. For the Kalman-Filter, the noise matrices R_t and Q_t for pose estimation are set beforehand. We assume x and y errors to be independent with standard deviations of 10cm in each direction for vision estimation, and a standard deviation of 5cm for odometry estimation. The standard deviation of the angle is set to 1.5° for vision and 0.75° for odometry.

The first experiment is shown in Figure 3. The left figure shows the resulting x,y -trajectory when the robot drives a square with 150cm edge length, starting from the left bottom corner. As seen from the figures, the actual trajectory (black line, measured by DTrack) is not a perfect square due to mechanical uncertainties described previously. These uncertainties deteriorate the odometry localization resulting in a suboptimal tracking performance (magenta dashed line). While the odometry localization largely deviates from the actual positions, visual (green crosses) and KVO (blue line) localization can track the robot sufficiently well. However, the tracking frequency of vision is much lower as given in Table I. The right plot in Figure 3 shows the corresponding robot's orientation in degree over time. As shown by the results, the KVO localization is close to the ground truth in position, as well as in orientation. The KVO estimation returns in some trajectory parts a "zig-zag" pattern. The reason for this behaviour is the inaccurate odometry which is periodically corrected by vision pose estimations.

The second experiment shown in Figure 4 is again a square with 150cm edge length, where the robot additionally makes a rotation of 90° in the corners. Starting point is again the bottom left corner. The few green crosses aside the true trajectory are faulty vision pose estimations. As KVO employs a 2σ confidence range, these outliers can be detected and, thus, the corresponding vision pose estimations are ignored.

Experiment 3 in Figure 5 shows a circle trajectory in counterclockwise direction. This trajectory includes simultaneous translation and rotation of the robot. KVO position and orientation (blue solid line) follow the real trajectory very closely. Only at the coordinates (110,80) the position differs significantly from ground truth due to some inaccurate vision pose estimates. The few green crosses beside the trajectory in the center of the circle are faulty vision pose estimates.

As final experiment, we used a complex trajectory shown in Figure 6. Here, we also included an area without visual markers. Hence, visual localization is not possible in this area and KVO employs the odometry information. For this scenario, odometry is unable to provide a reasonable position estimation and over-estimates the rotation in a range of 10° . The vision localization returns accurate position and orientation with few outliers. As expected, the accuracy of KVO decreases in the area without visual markers (as it relies on the odometry only). However, when the robot leaves the area without markers the tracking accuracy of KVO quickly increases.

For further evaluations, we examine the Euclidean distance of position estimations to the ground truth position. We also consider the absolute value of the difference between the angle estimation and the ground truth angle. The results of all experiments are shown in Table I. Here, the mean values of the errors represent the accuracy, while the 95% percentile and the maximal error describe the robustness of the localization methods. As expected, the vision pose error is in most cases significantly smaller than the odometry estimations. On the other hand, the vision maximal error can be extremely high due to faulty vision pose estimations. In all experiments, the KVO localization has not only a much higher localization frequency than vision, it is also significantly more accurate and robust (shown by the 95% percentile and the maximal errors). Higher accuracy of KVO compared to vision localization can be obtained due to the outlier detection and lack of time-delay, as described in Section III-B.

V. CONCLUSION AND OUTLOOK

In this paper, we present a simple but effective method to fuse vision and odometry for accurate robot indoor localization using low-cost devices, e.g. using off-the-shelf cameras. For fusing sensor's information, a Kalman-Filter was employed. The proposed method is able to combine the strengths of both, vision and odometry localization, while being fast and accurate.

The presented approach, the KVO localization, was evaluated on different driving movements using the omni-directional Robotino. At a tracking frequency of about 35Hz, the mean euclidean distance error of KVO to the ground truth was about 3-4cm. The mean absolute angle error was in the range of 1-3 degree. Further possibilities to increase this accuracy is, for example, the integration of gyroscope data with odometry. In future work, we investigate possibilities to employ natural landmarks instead of artificial markers for localization.

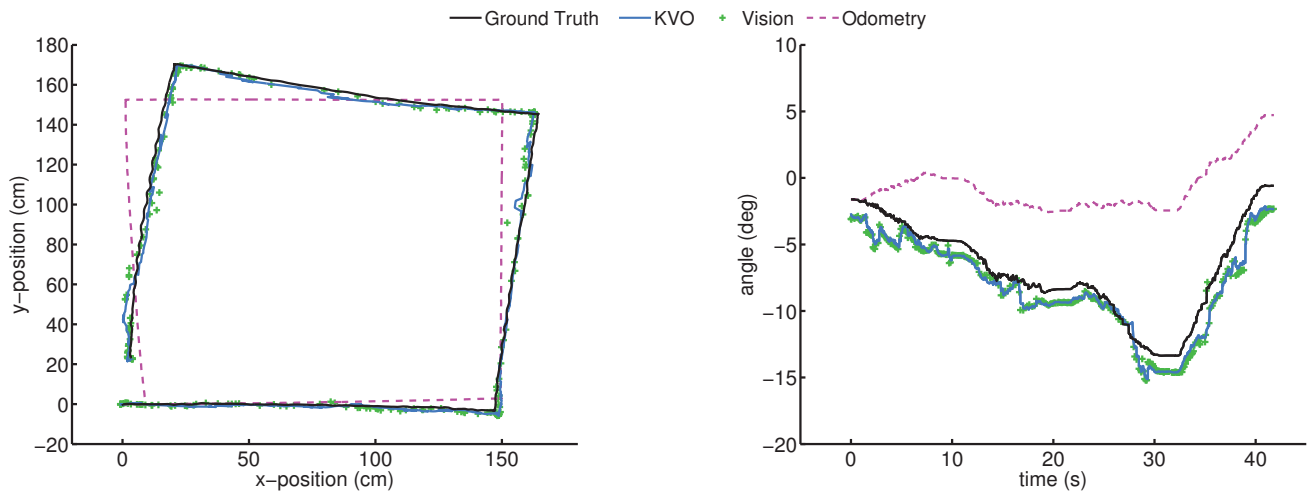


Fig. 3. For experiment 1, the goal was to drive a square of 150cm edge length with translation only. The left plot shows the robots x,y position, the right plot shows the angle θ over time. Due to mechanical uncertainties of the robot, the square (black solid line) is somewhat rotated. The KVO localization is represented by the blue solid line, the green crosses show the vision pose estimates, odometry is the dashed (magenta) line.

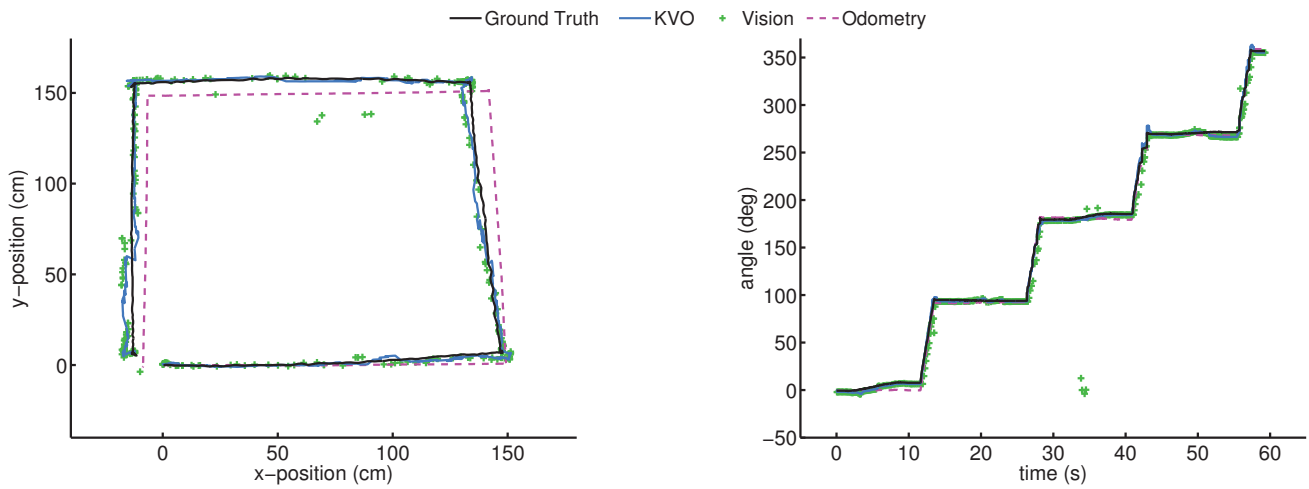


Fig. 4. For experiment 2, the goal was to follow a square-trajectory with 150cm edge length with additional rotation in the corners. The left plot shows the ground truth position (black solid line), as well as the position estimates of the localization methods KVO, vision and odometry. The right plot displays the angle over time.

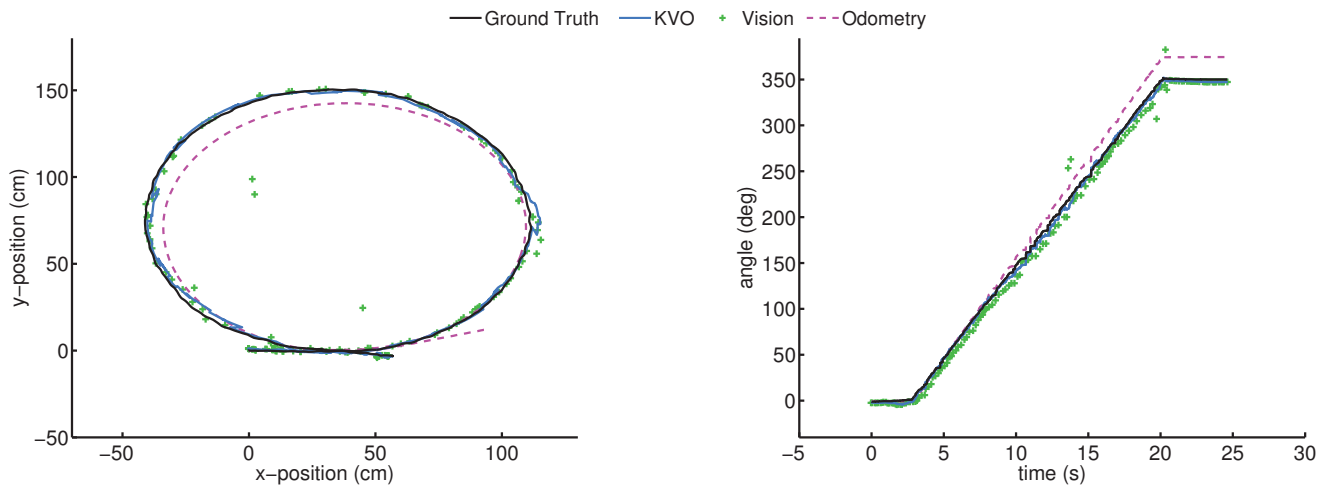


Fig. 5. Experiment 3 shows a circle trajectory with simultaneous translation and rotation. The left figure shows the robot position, while the right figure displays the robot angle over time.

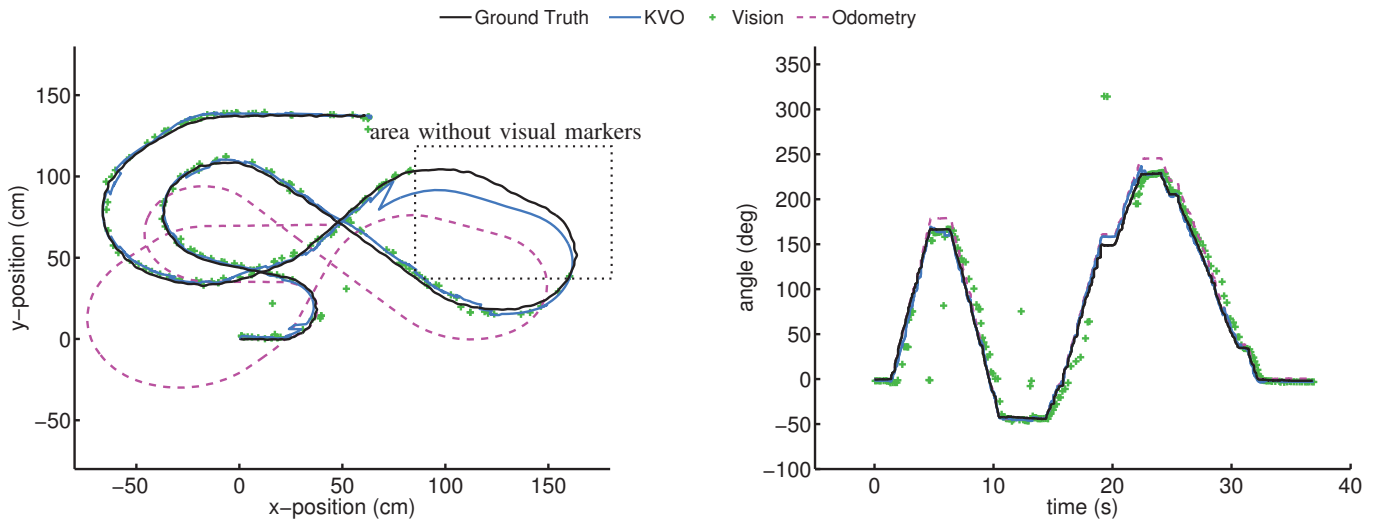


Fig. 6. Experiment 4 is a complex trajectory with simultaneous translation and rotation. The position of the robot is shown on the left, the angle over time on the right. In the left figure, the dotted square shows the area without visual markers. In this area, KVO localization relies on the odometry only.

TABLE I

THE TABLE SHOWS THE ERRORS OF ALL THREE LOCALIZATION METHODS COMPARED TO THE GROUND TRUTH ON FOUR DIFFERENT EXPERIMENTS.

THE POSITION ERROR IS THE EUCLIDEAN DISTANCE TO THE TRUE POSITION, WHILE THE ANGLE ERROR IS THE ABSOLUTE VALUE OF THE DIFFERENCE BETWEEN THE ANGLE ESTIMATE AND THE TRUE ANGLE. IN EXPERIMENT 4, KVO(*) SHOWS THE LOCALIZATION RESULTS FOR THE AREAS WHEN VISUAL MARKERS ARE AVAILABLE.

		position error in cm			angle error in degree		
		mean	95% perc.	max	mean	95% perc.	max
Exp. 1	KVO [~ 37Hz]	3.21	8.62	22.39	1.24	2.46	3.72
	Vision [~ 7Hz]	5.89	15.69	38.62	1.38	2.53	4.16
	Odometry [~ 30Hz]	14.01	24.74	29.38	4.05	8.48	8.65
Exp. 2	KVO [~ 37Hz]	2.63	6.26	14.40	1.99	5.02	22.63
	Vision [~ 7Hz]	4.33	12.90	25.55	4.87	16.96	178.48
	Odometry [~ 30Hz]	8.05	11.23	16.47	3.49	8.55	13.87
Exp. 3	KVO [~ 35Hz]	3.29	7.90	15.44	2.67	6.67	11.82
	Vision [~ 5Hz]	8.73	19.58	46.17	7.28	15.47	40.50
	Odometry [~ 30Hz]	19.01	39.21	42.77	12.11	24.45	24.64
Exp. 4	KVO [~ 35Hz]	4.11	12.02	22.28	3.36	10.32	18.01
	KVO(*) [~ 35Hz]	3.37	7.42	17.61	2.68	8.09	15.60
	Vision [~ 5Hz]	9.98	22.66	73.49	13.47	36.41	166.89
	Odometry [~ 30Hz]	35.68	71.08	74.67	6.48	16.43	19.91

[6] X. Li, M. Wang, and A. Zell, "Dribbling control of omnidirectional soccer robots," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 2623–2628.

[7] S. Se, D. Lowe, and J. Little, "Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks," *International Journal of Robotics Research*, vol. 21, pp. 735–758, 2002.

[8] A. Ramisa, A. Tapus, D. Aldavert, R. Toledo, and R. Lopez De Mantaras, "Robust vision-based robot localization using combinations of local feature region detectors," *Auton. Robots*, vol. 27, no. 4, pp. 373–385, Nov. 2009.

[9] J. M. Porta, J. J. Verbeek, and B. J. A. Kröse, "Active appearance-based robot localization using stereo vision," *Auton. Robots*, vol. 18, no. 1, pp. 59–80, Jan. 2005.

[10] P. Fernandez, S. Oh, G. Mariottini, and F. Dellaert, "Learning visibility of landmarks for vision-based localization," in *icra*, Ankara, AL, 2010, accepted.

[11] H. Kato and M. Billinghurst, "Marker tracking and hmd calibration for a video-based augmented reality conferencing system," in *Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality*, 1999, pp. 85–94.

[12] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, ser. Intelligent robotics and autonomous agents. The MIT Press, Aug. 2005.

REFERENCES

[1] I. J. Cox and G. T. Wilfong, Eds., *Autonomous robot vehicles*. New York, NY, USA: Springer-Verlag New York, Inc., 1990.

[2] Y. Qing-xiao, Y. Can, F. Zhuang, and Z. Yan-zheng, "Research of the localization of restaurant service robot," *International Journal of Advanced Robotic Systems*, vol. 3, 2010.

[3] J. Levinson and S. Thrun, "Robust vehicle localization in urban environments using probabilistic maps," in *ICRA*, 2010, pp. 4372–4378.

[4] G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin, "Precision flight control for a multi-vehicle quadrotor helicopter testbed," Sept 2011.

[5] A. Pronobis and B. Caputo, "COLD: COsy Localization Database," *The International Journal of Robotics Research (IJRR)*, vol. 28, no. 5, pp. 588–594, 2009.