

Indoor Navigation Research with the Khepera III Mobile Robot: An Experimental Baseline with a Case-Study on Ultra-Wideband Positioning

Amanda Prorok*, Adrian Arfire*, Alexander Bahr*, John R. Farserotu** and Alcherio Martinoli*

*Ecole Polytechnique Fédérale de Lausanne, Switzerland. Email: firstname.lastname@epfl.ch

**Centre Suisse d'Electronique et Microtechnique SA, Neuchâtel, Switzerland. Email: firstname.lastname@csem.ch

Abstract—Recent substantial progress in the domain of indoor positioning systems and a growing number of indoor location-based applications are creating the need for systematic, efficient, and precise experimental methods able to assess the localization and perhaps also navigation performance of a given device. With hundreds of Khepera III robots in academic use today, this platform has an important potential for single- and multi-robot localization and navigation research. In this work, we develop a necessary set of models for mobile robot navigation with the Khepera III platform, and quantify the robot's localization performance based on extensive experimental studies. Finally, we validate our experimental approach to localization research by considering the evaluation of an ultra-wideband (UWB) positioning system. We successfully show how the robotic platform can provide precise performance analyses, ultimately proposing a powerful approach towards advancements in indoor positioning technology.

I. INTRODUCTION

The last ten years have been marked by substantial progress in the area of indoor positioning systems [16]. In contrast to outdoor navigation, where crude positioning of sub-meter accuracy often satisfies user needs, indoor applications may require accuracies as low as a centimeter. Furthermore, outdoor positioning and navigation has been well explored and standardized, whereas indoor navigation remains a recent research area which is still in the process of generating numerous new systems and algorithms [8, 20].

A growing number of real-life applications that depend on automatic object location detection and navigation capabilities create the need for efficient and accurate testing methods. Also, as the miniaturization of application devices poses challenges to the integration of new technologies, additional testbeds must be considered. In our efforts to help improve indoor navigation capabilities, we propose a mobile robot as a fundamental tool enabling systematic testing under controlled conditions. The first goal of this paper is to rigorously evaluate the mobile robot's on-board navigation capabilities, and thus create a baseline for further navigation research using the selected platform (a Khepera III robot). Secondly, using an absolute positioning system based on UWB technology, the paper provides a proof-of-concept case-study which brings to evidence the potential of our proposed approach.

This paper is organized as follows. In Section II we explain our experimental setup and specify our ground-truth measurement system. Section III is dedicated to the development and validation of our navigation models specific to the Khepera III robot. Finally, in Section IV we describe a navigation scenario augmented with beacon-based, absolute UWB positioning data and discuss the resulting performances.

A. Mobile Robots for Navigation Research

Since the beginning of navigation research with autonomous mobile robots in the mid 80's, a multitude of robotic platforms have been developed to satisfy the needs of this increasingly important domain. Whereas early studies were often performed on robots with simplistic sensing capabilities (for example, limited to the combination of sonar sensors and wheel encoders, as in [2, 14], or to the combination of an optical range finder and odometry as in [7]), the platforms used today are often equipped with powerful sensing units, such as laser range finders and cameras [5, 23]. Moreover, adhoc as well as commercially available platforms are often tailored specifically for the tasks they are meant to solve. When operating in large spaces, i.e. spaces which span several rooms or even entire building floors, robots typically need augmented autonomy and combined sensing and computational capabilities. In [21, 24], the robot Minerva is used as a tour guide in a museum during opening hours. Apart from using sonars and wheel encoders, it uses 2 laser range finders which provide a 360° field of view, and an upwards pointing camera which is used for localization based on the structured ceiling. Similarly, in [21], the RWI B21 robot covers museum grounds, and uses an array of 24 sonar sensors and a laser range finder. Although the above mentioned robots have shown to successfully perform sophisticated real-life tasks, their large size (roughly 1m in height, 0.5m in diameter) makes it hard to integrate them into systematic evaluation systems underlying precise ground truth comparisons. More recently, studies have also been performed on smaller platforms. The limited robot size (below 0.2m in height, 0.15m in diameter) is leveraged to conduct systematic evaluation of specific navigation techniques. In [6] as well as in [1], Khepera III and Khepera II robots are used respectively, for the evaluation of novel calibration techniques.

The advantages of multi-robot versus single-robot naviga-

tion have been pointed out in several publications, where the strategy of multi-robot collaboration is able to compensate for deficiencies in the data owned by a singular robot [11, 18]. Subsequent works have focused on the reduction of computational [19] as well as communication [26] requirements. Thus, future research in collaborative navigation techniques using miniature, multi-robot systems has become tractable.

B. A Flexible, Multi-purpose Mobile Robotic Platform

In order to integrate single- and multi-robot systems into systematic and precise evaluation frameworks, it is practical to reduce the individual robot size to its minimum possible, without stripping it from vital resources such as autonomy, computation, and sensing. Also, a number of further design choices can be considered to maximize the utility of a robotic platform: (i) off-the-shelf components and software/hardware standards, (ii) modularity enabling extensions for different modalities, (iii) non-stop energetic autonomy, and (iv) wireless techniques for programming, data logging, and controlling.

The Khepera III fulfills all the above requirements. Indeed, it leverages the embedded system/PDA/cell phone market, uses wireless software and hardware standards, and runs Linux as its OS (i), for which it is aligned with its device size. The platform is modular (ii), possesses a modern battery technology (hot swap exchangeable) (iii), and includes both WiFi and bluetooth technology (iv).

II. EXPERIMENTAL SETUP

In this work, we present a minimal but necessary set of analyses and models for mobile robot navigation with the Khepera III platform. Our studies are evaluated using a precise ground truth measurement system for real-time tracking. The findings are finally validated in a simple navigation scenario. All experiments were conducted in a lab room with a hard plastic floor in a 3x3m robotic arena. The arena is delimited by 30cm high light gray borders with smooth surfaces. We used a fleet of 3 Khepera III robots, which we will denote as R_1, R_2, R_3 .

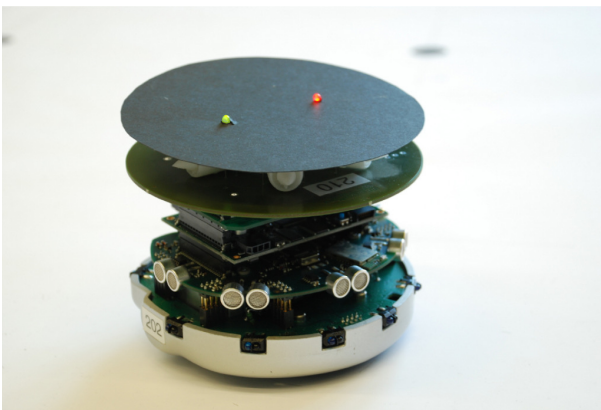


Fig. 1. The Khepera III robot. Here, the upper body is unmounted, and a module is stacked onto the extension bus providing two colored LEDs.

A. The Khepera III Mobile Robot

The Khepera III is a differential drive robot of 12cm diameter, produced by K-Team corporation¹ with development assistance from the Distributed Intelligent Systems and Algorithms Laboratory (DISAL). It is a descendant of the first generation Khepera robot which is smaller in size (5.5cm diameter) and since its first release in 1995 has been in use worldwide—to date, around 2000 Khepera I and Khepera II robots have been sold to over 600 universities. In its latest incarnation, the Khepera III offers a much higher computing power than its predecessors. A KoreBot extension board provides a standard embedded Linux operating system (Ångström distribution) on an Intel XSCALE PXA-255 processor running at 400 MHz. Basic sensing is formed by a ring of 9 infrared sensors (TCRT5000 Vishay Telefunken) and an additional pair facing downwards (for table top navigation and line following), and 5 ultrasound sensors (400ST/R100 Midas Components Ltd.) placed on the front semicircle. For a precise positioning of the sensors, the reader should refer to Figure 2. The infrared sensors have both active and passive functioning modes (for reflected emitted light and ambient light measurements, respectively), and an approximate range of 2cm-25cm. The ultrasound sensors have an approximate range of 20cm-400cm and record a maximum of 3 echoes and their corresponding timestamps. Further, the robot has a stackable expansion bus that enables the addition of custom robot modules. Finally, in order to facilitate communication, an IEEE 802.11 wireless card can be installed in the built-in CompactFlash slot. Since its market release in September 2006, 550 Khepera III robots have been sold to over 150 universities worldwide.

B. Real-time Ground Truth Measurements

There are several possible ways to gather ground truth information about a robot’s position, or even its trajectory. Due to their practicality (cheap and easy to use), cameras are often used for such purposes. For the work in this paper,

¹<http://www.k-team.com/>

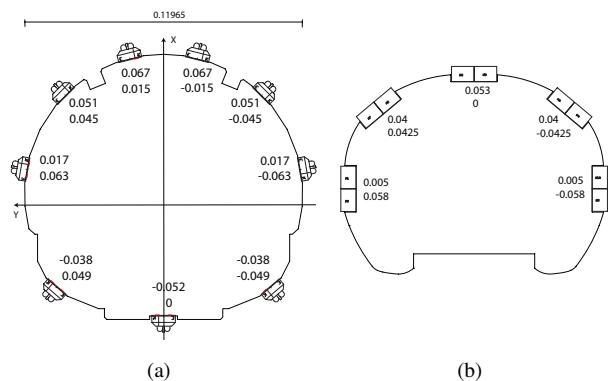


Fig. 2. The figure shows schematics of the Khepera III robot in true proportion. We mark the exact positions of all sensors relative to the robot center, a) for the 9 infrared body sensors (excluding the floor sensors) and b) for the 5 ultrasound sensors. The units are in meters.

	Dot detection	KIII detection
Mean pos. error [cm] $\mu \pm \sigma$	0.71 \pm 0.31	1.2 \pm 0.58
Max. pos. error [cm]	1.45	2.74
Mean orient. error [rad] $\mu \pm \sigma$	–	0.068 \pm 0.055
Max. orient. error [rad]	–	0.253

TABLE I
ACCURACY OF OVERHEAD TRACKING SYSTEM

we installed an overhead camera system in combination with the open source tracking software *SwisTrack* previously developed at the Distributed Intelligent Systems and Algorithms Laboratory². This software package is especially useful due to its modular architecture including numerous algorithmic components [17]. As well as using the available camera calibration components, we have exploited real-time tracking modules enabling us to perform precise analyses of time-dependent robot behavior.

In our setup, we use a GigE color camera (Basler-SCA1000-30GC) which has a standard resolution of 1032x778 pixels, and is mounted 2.5m above the robotic arena. The resulting picture resolution is 5.5 pixels per cm². Detected ground positions are calibrated using the calibration algorithm proposed by Tsai et al. [27], a method which requires information on the position of at least 5 non-collinear points on the image. We performed the calibration procedure on 64 points by installing a uniform grid with dots of 3cm diameter at its intersections. Further, we performed a second calibration for robot detection which additionally integrates a model of the setup, taking into account the robot height (11cm). In order to increase precision of the robot pose estimate, we install one red and one green LED onto the robot top. The image captured by *SwisTrack* is processed in a pipeline which includes the detection of a red-green blob pair and calculates the corresponding robot orientation. Using this described framework, we perform real-time tracking at a frequency of 10Hz. Lastly, in order to synchronize the real-time robot data with the real-time ground truth measurements we build a UDP communication channel. A simple script timestamps the data from the robot as well as from *SwisTrack* at the incoming ports. This architecture proved highly reliable (we considered the 3ms communication delay as negligible).

We evaluate our ground truth measurements system by calculating the mean positioning error on detected grid dots as well as detected KIII robots. We measure the positions of 64 dots, and of 28 placements of the KIII robot, uniformly distributed in the arena. We also measure the error of the orientation of the robot by rotating it 4 times by 90° per position. While the error on the dot detection is absolute, the error on KIII detection is subject to minor misalignments due to manual placement of the robot on the grid. Table I reports the achieved accuracy.

III. NAVIGATION WITH ON-BOARD SENSORS

Navigation systems typically need two components to estimate the current target position: a sensor model and a motion model. In the following section, we first discuss the effect of various calibration techniques on the accuracy of the Khepera III's motion model. Secondly, we elaborate sensor models for both the infrared and ultrasound sensors. Finally, in Section III-C, we validate our models by performing a simple navigation scenario.

A. Odometry

The Khepera III robot possesses high resolution wheel encoders (2764 ticks per revolution, roughly 7 ticks per degree). Yet in order to take full advantage of the high precision, odometry parameters must be calibrated.

1) *Motion Model*: We adopt the same formalism as described in [1], where robot velocity ν and angular velocity ω are related to the angular velocities of the robot wheels ω_R, ω_L as in

$$\begin{pmatrix} \nu \\ \omega \end{pmatrix} = C \begin{pmatrix} \omega_R \\ \omega_L \end{pmatrix}. \quad (1)$$

with matrix C equal to

$$\begin{pmatrix} \frac{r_R}{2} & \frac{r_L}{2} \\ \frac{r_R}{b} & -\frac{r_L}{b} \end{pmatrix} \quad (2)$$

where r_R and r_L are the right and left wheel radii respectively, and b the robot wheelbase. The Khepera III factory specifications are $r_R = r_L = 0.0021\text{m}$ and $b = 0.08841\text{m}$. Using these nominal values we have

$$C = \begin{pmatrix} 0.0105 & 0.0105 \\ 0.23753 & -0.23753 \end{pmatrix}. \quad (3)$$

In order to correctly predict robot displacement, these parameters (r_R, r_L and b) are tuned for an individual robot in a specific setting.

2) *Calibration*: Since the beginning of research on odometry calibration in the late 80's, a number of calibration methods have been proposed [1, 4, 13, 21, 29]. We have here chosen to test the appropriateness of 3 particular (offline) methods

- a** Borenstein et al. [4]
- b** Kelly et al. [13]
- c** Antonelli et al. [1]

using three sets of trajectories, as shown in Figure 3:

- set1** 10 trajectories on shape B (CW/CCW)
- set2** 12 trajectories on shapes A and C (CW/CCW)
- set3** 3 trajectories on shape D

where CW and CCW stand for clockwise and counter-clockwise. Each of the methods **a, b** and **c** is based on a different concept and separate hypotheses. In their work, Borenstein et al. propose a straightforward calibration method which treats systematic errors (as opposed to non-systematic). Their parameter correction method is based on a geometric derivation, and assumes that the average of the actual wheel diameters is equal to the nominal one. In order to apply the method, only the beginning- and end-positions of the robot

²<http://en.wikibooks.org/wiki/SwisTrack>

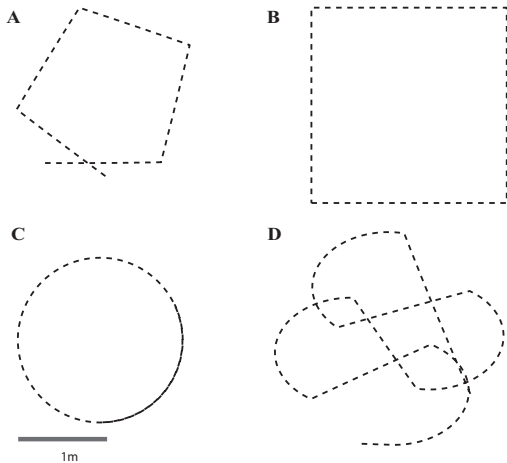


Fig. 3. Trajectories used to calibrate and evaluate the motion model. Trajectories **A**, **B**, and **C** were performed in both CW and CCW directions.

	Borenstein	Kelly	Antonelli
R_1	$\begin{pmatrix} 0.0105 & 0.0105 \\ 0.2345 & -0.2359 \end{pmatrix}$	$\begin{pmatrix} 0.0155 & 0.0106 \\ 0.2357 & -0.2369 \end{pmatrix}$	$\begin{pmatrix} 0.0142 & 0.0074 \\ 0.2345 & -0.2354 \end{pmatrix}$
R_2	$\begin{pmatrix} 0.0105 & 0.0105 \\ 0.2366 & -0.2365 \end{pmatrix}$	$\begin{pmatrix} 0.0106 & 0.0106 \\ 0.2385 & -0.2385 \end{pmatrix}$	$\begin{pmatrix} 0.0105 & 0.0112 \\ 0.2347 & -0.2344 \end{pmatrix}$
R_3	$\begin{pmatrix} 0.0105 & 0.0105 \\ 0.2348 & -0.2353 \end{pmatrix}$	$\begin{pmatrix} 0.0106 & 0.0106 \\ 0.2389 & -0.2393 \end{pmatrix}$	$\begin{pmatrix} 0.0118 & 0.0097 \\ 0.2336 & -0.2341 \end{pmatrix}$

TABLE II
ODOMETRY COEFFICIENTS DERIVED FROM CALIBRATION METHODS

have to be recorded. We performed this calibration on **set1**, for each robot.

Kelly et al. pose odometry as a nonlinear dynamical system. Optimal odometry parameters are found by solving a nonlinear optimization problem which minimizes the error between actual and nominal trajectories. Thus, we have to gather and synchronize robot wheel speed data with tracking information in real-time (as explained in Section II-B). We performed this calibration on **set2**, for each robot.

Antonelli et al. propose a least-squares method to estimate the matrix C directly. In order to apply this method, robot wheel speed data as well as the robot end-positions is necessary. We performed this calibration on **set2**, for each robot.

Finally, we obtain the calibrated parameters shown in Table II.

3) *Experimental Results*: In order to evaluate the performance of the tested methods, we compute the following metric which gives us the normalized error at time T

$$\epsilon_{od}(T) = \frac{2}{T} \int_0^T \frac{\epsilon(t)}{d_R(t) + d_L(t)} dt \quad (4)$$

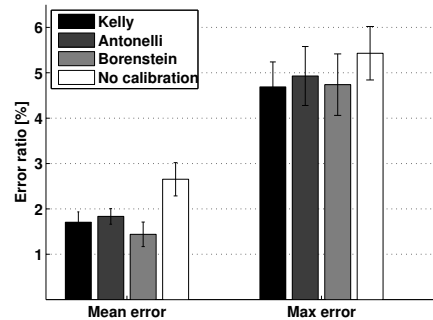


Fig. 4. Normalized (over distance) error ratio obtained for 3 different odometry calibration methods, as well as for non-calibrated robots. The errorbars show a 95% confidence interval.

where $\epsilon(t)$ is the euclidean distance between the actual and estimated robot positions, and $d_L(t)$, $d_R(t)$, are the distances traveled by the left and right wheels at time t . We perform this evaluation on both **set2** and **set3**, for each robot. The performance of the calibrated motion models is reported in Figure 4. The results of running the Wilcoxon rank-sum test (with threshold 0.05) on the mean error performance data show that while there is a statistically significant improvement in performance when using the calibration methods, there is no significant difference in performance between the three methods themselves. In terms of maximum error, there is no statistically significant difference in the results obtained with calibration from the ones obtained without calibration.

B. Sensor Models

Due to their fast response time and low cost, infrared sensors are often employed on mobile robots. Yet, because of their non-linear behavior and dependence on the reflectance of surrounding objects, they are rather used for proximity sensing (in collision avoidance) than for range sensing (in localization and mapping) [3]. Ultrasound sensors are a popular choice among range sensors, despite the many shortcomings that they have, such as low resolution, poor directionality and corruption by reflections and specularities [15]. Still, when compared to other rangefinders such as lasers or stereo vision, they are orders of magnitude cheaper and consume little power, and thus offer interesting characteristics for miniaturized systems. Moreover, ultrasound and infrared sensors can be combined to compensate for each others weaknesses and provide improved accuracies [9].

In this section, we investigate the utility of the Khepera III's sensors as rangefinders. We develop range models and elaborate the general sensor characteristics.

1) *The Gaussian Assumption*: The assumption that sensors behave according to a Gaussian distribution is the baseline for several localization methods [25]. Thus, before developing localization methods for a specific platform, it is useful to analyze the distribution of raw sensor values for a fixed, known position. To test the behavior of the infrared sensors, we per-

	Skewness	Kurtosis	Kolmogorov-Smirnov
Infrared	0.218	2.371	0.0588
Ultrasound	0.256	2.693	0.0441
Normal	0	3	0

TABLE III
NORMALITY OF INFRARED AND ULTRASOUND SENSOR DATA.

formed 3000 measures for each of the distances 20cm, 10cm, 6cm 4cm and 3cm. For the ultrasound sensors, 200 measures were taken for 7 distances equally spaced out within 40cm and 280cm. For each sensor, a Skewness-Kurtosis test as well as a Kolmogorov-Smirnov test were performed. Table III reports the obtained statistics, as well as the reference values of a Normal distribution. Although the sensors exhibit distributions which slightly deviate from a Normal distribution (rejected at a 0.05 significance level), we observe that the value distributions are strongly related to their Gaussian approximations. We will from hereon assume that the Gaussian noise assumption holds.

2) *Range Model*: The methodology we use to develop the range models for the infrared and ultrasound sensors is general and applicable to any experimental setup, even if the actual results obtained here are specific to our setting. In order to obtain a range model, where for each raw sensor value we have a distance (range) value, we gather a large amount of data. For the infrared sensors, we record a total of 5400 sensor measurements per distance in $R_{IR} = \{0, 2, \dots, 26\text{cm}\}$, and for the ultrasound sensors, we record a total of 1500 measurements per distance in $R_{US} = \{20, 40, \dots, 400\text{cm}\}$.

We denote by d the distance values, and by v the raw sensor values. For all $d_e \in R_{IR}, R_{US}$, we estimate the mean μ_e and variance σ_e . Then, for D points within the ranges R_{IR}, R_{US} we interpolate μ_i and σ_i with $i \in [0..D]$. Now we can sample a set of $S \times D$ points to create an ensemble Ω defined as

$$\Omega = \bigcup_{i \in [0..D]} \{(v_j, d_i) | v_j \sim \mathcal{N}(\mu_i, \sigma_i), j \in [0..S]\} \quad (5)$$

where we make the assumption that sensor values v_j are Gaussian distributed. In order to calculate the means and variances along the sensor value axis v , we define a sliding window with width w . Thus, the number of points contained in a window is

$$N = \sum_{(v_j, d_i) \in \Omega} \Pi\left(\frac{v_i - v}{w}\right) \quad (6)$$

where $\Pi(x)$ is the rectangular function. The mean distance $\mu(v)$ is equal to the average value of the points contained in this window, and

$$\mu(v) = \sum_{(v_j, d_i) \in \Omega} \frac{d_i \cdot \Pi\left(\frac{v_i - v}{w}\right)}{N}. \quad (7)$$

The corresponding standard deviation is

$$\sigma(v) = \sqrt{\frac{1}{N} \sum_{(v_j, d_i) \in \Omega} (d_i - \mu(v))^2 \cdot \Pi\left(\frac{v_i - v}{w}\right)}. \quad (8)$$

Figure 5 shows the range models for both infrared and ultrasound sensors, resulting from the measurements made in our experimental setup. We note that in the case of the infrared sensor, raw values below a certain threshold correspond to pure sensor noise when no object is in the sensor range. This lower threshold can be easily defined by performing an initial calibration for each individual sensor.

3) *Ultrasound Beam Opening Angle*: Unlike the infrared sensors, ultrasound sensors have poor directionality and signal echoes are the result of reflection off various objects within the sensor beam. In order to properly model and predict these return echoes, it is useful to have an accurate estimate of the ultrasound sensor's opening angle. Here, we present an experiment devised to capture the opening-angle characteristics of the Khepera III's ultrasound sensors. The robot was placed with the sensor in question facing a wall, with no other possible targets within its range. Then, the robot was slowly rotated on the spot with 1° increments over the interval $5^\circ - 90^\circ$, while range measurements were taken for each increment. The basic idea is that the measurements will return the perpendicular distance to the wall as long as the sensor's angle relative to the wall is not larger than half the beam angle. Past this maximum angle, no echoes are able to return to the receiver. We performed this experiment for the distances $\{0.5, 1.0, \dots, 4.0\}\text{m}$, as the amplitude of the signals decrease with increasing distances to the wall, thus affecting the maximum opening angles. The results of this experiment are shown in Figure 6. The amplitude of the signal is plotted in function of the distance and angle to the wall. Also, the curve describing the relation of half the opening-angle to distance is

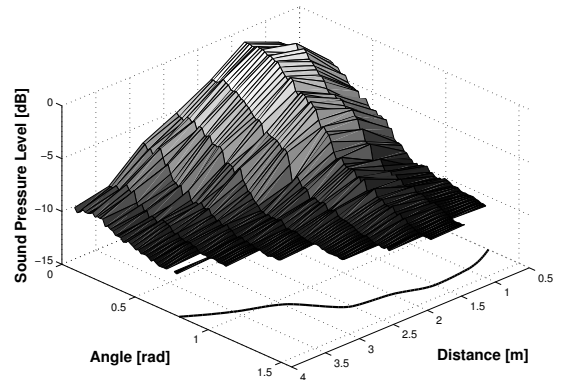


Fig. 6. This graph describes the sound pressure level of ultrasound sensor signal echoes, in function of the distance and angle to an object. Also, the relation of the maximal ultrasound beam opening angle to the distance from an object is described by the curve on the XY plane.

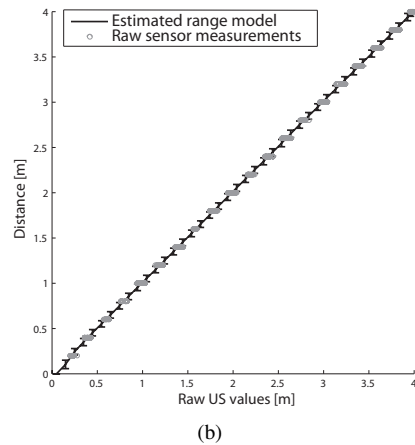
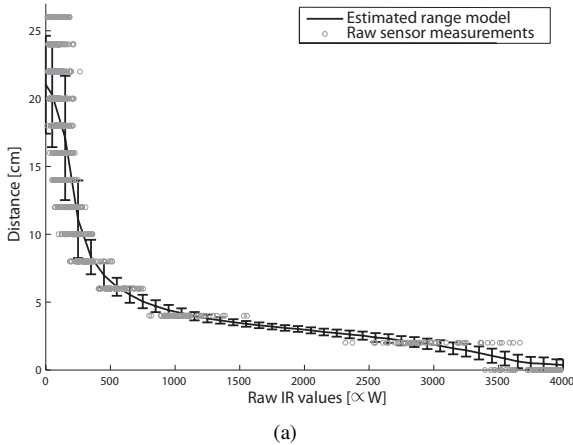


Fig. 5. Estimated range models and the raw sensor measurements on which they are based for (a) infrared sensors, with a window-size of $w = 40$ and (b) ultrasound sensors, with a window-size of $w = 0.04$. The errorbars show the standard deviation.

plotted on the XY plane; the maximal angles range from 70° to 150° .

4) *Maximum Update Rate*: The performance of a localization method depends on the rate at which new observations of the environment are made. In other words, the faster a mobile robot moves, the more often it will need to update its observations in order to preserve the same localization accuracy. Assuming that the time a CPU spends processing the localization algorithm is negligible, this performance becomes inherently dependent on the maximal sensor update rates. The Khepera III tech-report [12] cites an update time of 33ms for the infrared sensors. We perform a simple experiment which validates these numbers, and thus the maximum frequency at which all 11 infrared sensors read new values is 30.3 Hz. As no documentation was found regarding the update rate of the ultrasound sensors, we performed a series of experiments to report these values. We found that the ultrasound sensors can be used in two different schemes: concurrent or sequential readings. Although sequential readings are bound to be slower than concurrent ones, crosstalk among the sensors is limited, which can be beneficial for localization [15]. Table IV reports the maximal frequencies found for the ultrasound sensors. The frequency is defined by the time it takes in between two updates (in the case of multiple sensors, an update occurs when all sensors have refreshed their values). Highest frequencies are obtained when using a single sensor, whereas lowest when using several sensors sequentially. The obtained frequencies could potentially be improved if we were to decrease the sensor timeout time (thus also decreasing the range), but this flexibility is not offered by the current firmware.

C. Validation: A Basic Navigation Scenario

Finally, we design a simple navigation scenario which employs our models described in Section III-B. In our setup, we require the robot to drive a 20m long path along a 1m large square, at an average speed of 5cm/s. The robot localizes itself while moving along an outer wall, where in a first sub-scenario, it does this by utilizing its infrared sensors, and in

	$\mu \pm \sigma$
Single sensor [Hz]	4.23 ± 0.08
5 sensors, concurr. [Hz]	3.8 ± 0.19
5 sensors, sequential. [Hz]	0.74 ± 0

TABLE IV
MAX. ULTRASOUND SENSOR UPDATE FREQUENCIES

a second sub-scenario, it utilizes its ultrasound sensors. The wall is placed in such a way that its visibility with respect to the sensors is preserved throughout the run. For the runs performed with the ultrasound sensors, we tried three different schemes: 5 sensors concurrently, 3 sensors concurrently, 5 sensors sequentially. The infrared sensors were used in a single scheme where all 9 body sensors are activated concurrently. The sensor update frequencies correspond to the ones reported in Section III-B4. We perform two runs per robot (CW and CCW directions), for each sensor configuration. Throughout the experiment, we employ real-time tracking as explained in Section II-B. Odometry measurements are taken at a frequency

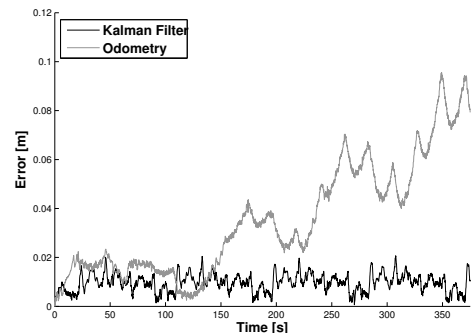


Fig. 7. A qualitative overview of localization performance. Localization was performed (i) with an extended Kalman filter (using 9 concurrently activated infrared sensors), and (ii) with dead-reckoning (odometry).

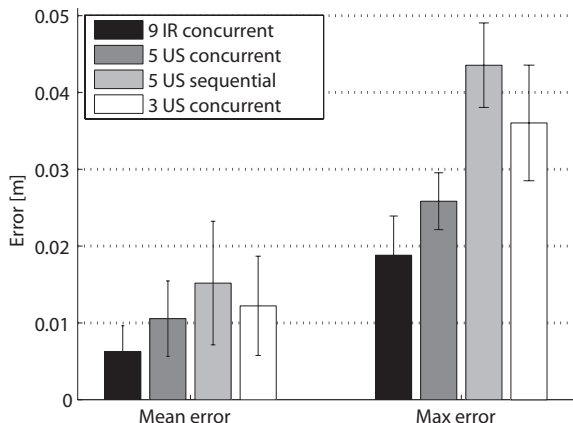


Fig. 8. Performance of an extended Kalman filter localization algorithm. We tested 4 different sensor configurations: 9 concurrent infrared sensors, 5/3 concurrent ultrasound sensors, and 5 sequential ultrasound sensors. The errorbars show a 95% confidence interval.

of 10Hz .

In order to localize, we implement an extended Kalman filter referring to a feature-based map. We complete our ultrasound measurement model by implementing the scan predictor and matching method as proposed by Leonard et al. [15]. A validation gate is used to determine the correspondence between predictions and observations (we set the gate threshold to two standard deviations). Because feature extraction is more difficult to achieve with the Khepera III’s infrared sensors due to poor coverage of surrounding space, we implement a scan matching method. We use our range model to generate a range scan from the 9 infrared sensors, which is then translated and rotated to produce a maximum overlap with our map. The resulting pose update is then integrated into our Kalman filter as proposed in [10].

The qualitative localization performance can be observed in Figure 7: the error of the position estimates remains bounded, whereas the dead-reckoning error accumulates constantly. The quantitative results are reported in Figure 8. They show that in this basic navigation scenario, the Khepera III is able to localize itself with an error that is below 1.5cm on average (which is in the order of the ground truth measurement error), validating our previously developed sensor models.

IV. CASE-STUDY: LOCALIZATION WITH ABSOLUTE UWB POSITIONING

In comparison with known positioning technologies, and especially in indoor environments, in absence of global navigation satellite systems (GNSS), the potential strengths of UWB localization systems become apparent [22]. Due to the high bandwidth, UWB receivers are capable of decomposing multipath signal components, yielding potentially high precision with a high update rate. Furthermore, the technology is highly scalable and can be implemented at low cost and using very little power (as the complexity of the positioning algorithms can be outsourced to an external system). Nevertheless, UWB positioning performance may be heavily affected by various

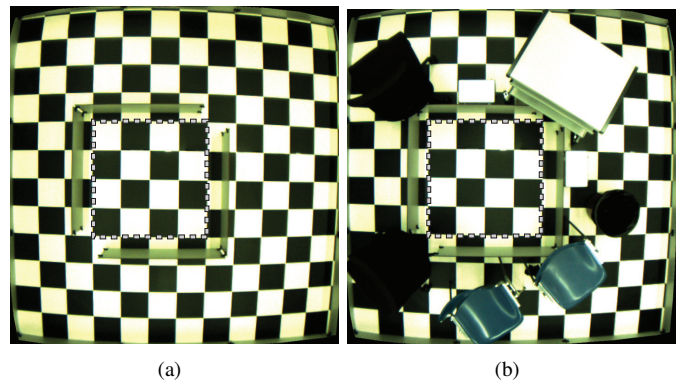


Fig. 9. Overhead images of two variant experimental setups (one checker square is 25cm large). Common to both variants are the 3m large square arena, and a short wall surrounding the robot’s path. The robot trajectory is shown by a dashed line. (a) The arena is empty. (b) The area surrounding the center square path is cluttered with everyday objects and office furniture (i.e. metallic desk, chairs, bin, boxes).

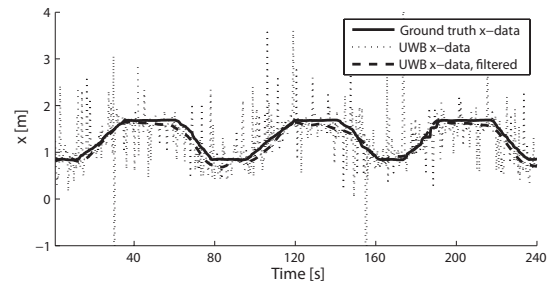


Fig. 10. The graph shows one-dimensional ground-truth and UWB positioning data gathered during an experiment. Raw UWB data are highly noisy, and are thus filtered to produce the shown result.

factors induced by dynamic or highly cluttered operational environments, leading to non-line-of-sight (NLOS), multipath, and shadowing artifacts which are hard to overcome.

The work in this section shows how our previously developed setup and set of tools can be leveraged to perform in-depth analyses of the localization performance of an additional system, here a state-of-the-art UWB positioning system commercially available from Ubisense [28], Version 1.1. In order to perform this analysis, we design an augmented experimental setup including the absolute UWB positioning system.

A. Experiments

The experimental setup is similar to the one introduced in Section III-C. Here, the robot drives along a 4m long path following the perimeter of a 1m large square at an average speed of 5cm/s, and uses its infrared sensors (which detect the inner square wall as depicted in Figure 9) to localize itself along this path. For future reference, we consider the path’s origin as the bottom left corner of the 1m large square, with the robot driving in counter-clockwise direction. The trajectory is contained in a larger, 3m large square arena as shown in Figure 9 (a). Additionally, an UWB emitter tag (1cm × 9cm × 6cm large) which sends localization signals at a frequency of 5Hz is attached to the robot, centered on its top. Four UWB

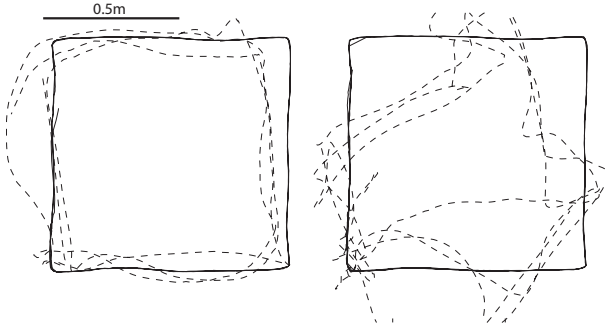


Fig. 11. Robot trajectory (solid line) and trajectory reconstructed from filtered UWB (dashed line) positioning data. *Left*: Data gathered from **exp1**. *Right*: Data gathered from **exp2**. It becomes visible how a highly cluttered environment may affect the localization performance of an UWB positioning system.

receiver stations are mounted at the corners of the arena at the height of the ceiling (2.45m). We measure the localization performance of the UWB system in terms of the euclidean distance to the localized robot position (which deviates less than 2cm from the ground truth position, as shown in Figure 8). In order to test the UWB positioning system, we run the following two experiment variants:

- exp1** The arena is empty (illustrated in Figure 9 (a)).
- exp2** The area surrounding the 1m square robot trajectory is cluttered with everyday objects and office furniture (illustrated in Figure 9 (b)).

For each of the two experiment variants, the robot drove 25 times along the 4m long path. Due to the high noise level of the UWB positioning data, we employ a robust, least-mean-square filter with a 12s time window (which was the time window yielding the lowest error). This filter is applied equally on all obtained UWB data. Figure 10 shows x-coordinate data gathered during an experiment (of variant **exp1**). One observes that original UWB data frequently deviate from the ground-truth values, often by more than 1m, and that the localization performance is strongly improved by our filter.

B. Discussion

Figure 11 shows the robot trajectories and the trajectories reconstructed from filtered UWB data gathered during 3 runs around the square of the variant **exp1** and **exp2** experiments. It becomes clear that the performance of the UWB positioning system is heavily affected by the cluttered environment present in the **exp2** setup. Also, given the repetitive nature of the robot's trajectory, it becomes apparent how certain objects in the cluttered environment consistently deviate the position estimates made by the UWB system.

We exploit the robot's precise localization information to quantitatively evaluate the UWB system's positioning error. In order to do this, we track the error as a function of the robot's position along the 4m long path. Figure 12 shows the position estimation error obtained from the UWB system for the two experiment variants **exp1** and **exp2**. It shows how in a clear environment the UWB system performs relatively well,

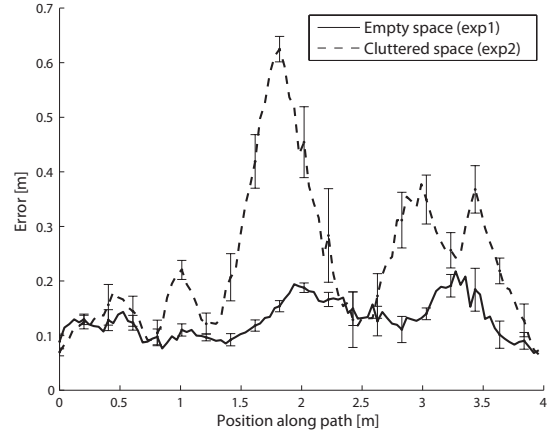


Fig. 12. The graph shows the localization error averaged over 25 runs along the 4m long path. Two experiments were conducted, one for an empty space (**exp1**) and one for a cluttered space (**exp2**). The errorbars show a 95% confidence interval.

and has a near to constant error throughout the run (0.13 ± 0.035 m). Further, it shows how in a cluttered environment the overall performance deteriorates (0.23 ± 0.13 m). In particular, for the experiment in the cluttered environment the graph shows how the error is strongly dependent on the locality of the measurement (which is also confirmed by the small errorbars). In this case, the graph suggests that the position 1.8m along the path (corresponding to the top right corner of the square) is particularly susceptible to positioning errors, as the average value increases to over 0.6m. This is due to the placement of the table (as seen in Fig. 9) composed mainly of metal, which is known to heavily affect UWB signaling.

By analyzing positioning errors (offline or even online) in combination with information on the measurements' localities, one can potentially deduce how the specific configuration of objects in a cluttered environment affect an UWB system's positioning performance. Further, in unknown environments, the actual sensor data collected by the robot could be considered in order to make conclusions about the presence (or absence) of objects, as well as their type. Yet finally, it is important to note that while the robotic platform localizes very well in environments with a dense distribution of features on which it can apply its localization filter, in an empty environment, the robot may have to rely on dead-reckoning localization over long time spans, which may result in high errors (as was seen in Figure 7). The latter situation is contrasted by the performance of the UWB positioning system, which can provide absolute positioning at all times and performs more reliably in an empty space.

V. CONCLUSION

In this work we have provided an analysis and demonstration of the Khepera III robot's localization capabilities based on its on-board sensors. We performed our evaluations with a precise, real-time ground truth measurement system using an overhead camera. The experimental results presented here have shown that our models can be integrated into classical

navigation algorithms to produce good performance, creating a baseline reference for future navigation research using the Khepera III platform. Furthermore, we perform a proof-of-concept case-study employing a commercially available state-of-the-art absolute UWB positioning system. We show how the UWB system's performance can be precisely evaluated. Moreover, given the strong localization capabilities of the robot, the error statistics can be formulated as a function of the measurement localities. Ultimately, our baseline set of tools proposes a powerful approach towards advancements in indoor positioning technology.

ACKNOWLEDGMENT

This work was partially sponsored by the National Center of Competence in Research on Mobile Information and Communication Systems (NCCR-MICS). Thanks goes to Julien Tharin, Frederic Lambercy and Laurence Schneider-Vidi of K-Team Corporation for providing invaluable information on the Khepera III platform. This work made use of the Khepera III Toolbox, available at http://en.wikibooks.org/wiki/Khepera_III_Toolbox.

REFERENCES

- [1] G. Antonelli, S. Chiaverini, and G. Fusco. A calibration method for odometry of mobile robots based on the least-squares technique: theory and experimental validation. *IEEE Transactions on Robotics*, 21(5):994–1004, 2005.
- [2] R. C. Arkin. Motor schema-based mobile robot navigation. *The International Journal of Robotics Research*, 8(4):92–112, 1989.
- [3] G. Benet, F. Blanes, J. E. Simó, and P. Pérez. Using infrared sensors for distance measurement in mobile robots. *Robotics and Autonomous Systems*, 40(4):255–266, 2002.
- [4] J. Borenstein and L. Feng. Measurement and correction of systematic odometry errors in mobile robots. *IEEE Transactions on Robotics and Automation*, 12(6):869–880, 1996.
- [5] R. Brega, N. Tomatis, and K. Arras. The need for autonomy and real-time in mobile robotics: a case study of xo/2 and pygmalion. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 1422–1427, 2000.
- [6] A. Censi, L. Marchionni, and G. Oriolo. Simultaneous maximum-likelihood calibration of odometry and sensor parameters. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 2098–2103, 2008.
- [7] I. Cox. Blanche—an experiment in guidance and navigation of an autonomous robot vehicle. *IEEE Transactions on Robotics and Automation*, 7(2):193–204, 1991.
- [8] Z. Ebrahimiyan and R. Scholtz. Receiver sites for accurate indoor position location systems. In *Proceedings of the IEEE/ACES International Conference on Wireless Communications and Applied Computational Electromagnetics*, pages 23–26, 2005.
- [9] A. M. Flynn. Combining sonar and infrared sensors for mobile robot navigation. *Int. Journal of Robotics Research*, 7(6):54–64, 1988. ISSN 0278-3649.
- [10] J.-S. Gutmann, W. Burgard, D. Fox, and K. Konolige. An experimental comparison of localization methods. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 736–743, 1998.
- [11] A. Howard, M. J. Mataric, and G. S. Sukhatme. Localization for mobile robot teams using maximum likelihood estimation. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 1, pages 432–439, 2002.
- [12] K-Team. Khepera III user manual, version 2.2. Technical report, March 2002.
- [13] A. Kelly. Fast and easy systematic and stochastic odometry calibration. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 4, pages 3188–3194, 2004.
- [14] Y. Koren and J. Borenstein. Potential field methods and their inherent limitations for mobile robot navigation. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, volume 2, pages 1398–1404, 1991.
- [15] J. J. Leonard and H. F. Durrant-Whyte. *Directed Sonar Sensing for Mobile Robot Navigation*. Kluwer Academic Publishers, 1992.
- [16] H. Liu, H. Darabi, P. Banerjee, and J. Liu. Survey of wireless indoor positioning techniques and systems. *IEEE Transactions on Systems, Man and Cybernetics*, 37(6):1067–1080, 2007.
- [17] T. Lochmatter, P. Roduit, C. Cianci, N. Correll, J. Jacot, and A. Martinoli. SwisTrack - A Flexible Open Source Tracking Software for Multi-Agent Systems. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4004–4010, 2008.
- [18] A. I. Mourikis and S. I. Roumeliotis. Performance analysis of multirobot cooperative localization. *IEEE Transactions on Robotics*, 22(4):666–681, 2006.
- [19] E. D. Nerurkar, S. I. Roumeliotis, and A. Martinelli. Distributed maximum a posteriori estimation for multi-robot cooperative localization. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1402–1409, 2009.
- [20] V. Renaudin, B. Merminod, and M. Kasser. Optimal Data Fusion for Pedestrian Navigation based on UWB and MEMS. In *Proceedings of the IEEE/ION Position Location and Navigation System Conference (PLANS)*, pages 341–349, 2008.
- [21] N. Roy, W. Burgard, D. Fox, and S. Thrun. Coastal navigation-mobile robot navigation with uncertainty in dynamic environments. In *Proceedings of the 1999 IEEE International Conference on Robotics and Automation (ICRA)*, volume 1, pages 35–40, 1999.
- [22] Z. Sahinoglu, S. Gezici, and I. Guvenc. *Ultra-wideband Positioning Systems*. Cambridge, 2008.
- [23] S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hahnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz. Minerva: a second-generation museum tour-guide robot. In *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, volume 3, pages 1999–2005, 1999.
- [24] S. Thrun, D. Fox, W. Burgard, and F. Dellaert. Robust monte carlo localization for mobile robots. *Artificial Intelligence*, 128(1-2):99–141, 2000.
- [25] S. Thrun, W. Burgard, and D. Fox. *Probabilistic Robotics*. The MIT Press, 2005.
- [26] N. Trawny, S. I. Roumeliotis, and G. B. Giannakis. Cooperative multi-robot localization under communication constraints. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4394–4400, 2009.
- [27] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323–344, 1987.
- [28] Ubisense. Real-time location systems (RTLs) and geospatial consulting. URL <http://www.ubisense.net>.
- [29] C. Wang. Location estimation and uncertainty analysis for mobile robots. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 1231–1235, 1988.