2018 European Control Conference (ECC)
June 12-15, 2018. Limassol, Cyprus

Sensor Fusion Localization and Navigation for Visually Impaired People

G. Galioto[1], I. Tinnirello[1], D. Croce[1], F. Inderst[2], F. Pascucci[2] and L. Giarré[3]

***Abstract*— In this paper, we present an innovative cyber physical system for indoor and outdoor localization and navigation, based on the joint utilization of dead-reckoning and computer vision techniques on a smartphone-centric tracking system. The system is explicitly designed for visually impaired people, but it can be easily generalized to other users, and it is built under the assumption that special reference signals, such as colored tapes, painted lines, or tactile paving, are deployed in the environment for guiding visually impaired users along predefined paths. Differently from previous works on localization, which are focused only on the utilization of inertial sensors integrated into the smartphones, we exploit the smartphone camera as an additional sensor that, on one side, can help the visually impaired user to identify the paths and, on the other side, can provide direction estimates to the tracking system. We demonstrate the effectiveness of our approach, by means of experimental tests performed in a real outdoor installation and in a controlled indoor environment.**

I. INTRODUCTION

The pervasive availability of technologies for localization and user interfaces (such as GPS, WiFi networks, smartphones, etc.) is originating a proliferation of interesting solutions devised to improve the independence and social inclusion of visually impaired people [1]. Some of these solutions consider the design of special-purpose devices that enables specific activities such as pattern identification, perception of spatial layout, control of locomotion with respect to the *near* environment, for facilitating mobility, and *large scale* environment, for facilitating orientation [2].

This paper focuses on navigation, localization, and orientation of visually impaired people using a *non-GPS* approach in both outdoor and indoor environments. Blind users, indeed, cannot rely on GPS outdoor as sighted people, due to the large spatial error of this system in urban canyons. The situation is even more complex in indoor environments, where different solutions have to be considered for supporting user orientation. These solutions are based on the triangulation of RF signals (mainly WiFi), direct sensing of reference points (with RFIDs, ultrasound, bluetooth, etc.), ego-motion estimate using *Inertial Measurement Units* (IMU, i.e., accelerometers, magnetometers, and gyroscopes). This last approach, known as *Pedestrian Dead Reckoning* (PDR), is accurate in low-medium range, but it suffers from drift due to noise [3] on long distance. In [4], the knowledge of the walking patterns is used to reset the IMU bias when the pedestrian is not moving; in [5], [6] the extended Kalman filtering is exploited to compensate the drift; in [7] RFID sensors are used to reset the accumulated errors. Reference points provided *fingerprinting* map are adopted in [8], [9]. Despite their limitations, dead-reckoning techniques are largely employed in real navigation systems [10], [11]: in the Navatar [12] and RF-PATH-ID system, users interact with the application and help to correct possible navigation errors.

Concerning indoor navigation systems for blind users, it is worth noticing that they have stricter requirements than general systems, in terms of accuracy and reaction times. Moreover, visually impaired people are used to a continuous reference signal, which guides the users along the path to the destination. This signal is currently implemented using tactile pavings. However, an interesting solution for providing this signal by means of flexible, economic and easy-to-deploy traces has been proposed with the design of the ARIANNA system (pAth Recognition for Indoor Assisted NavigatioN with Augmented perception) [13], [14], [15]. The system is based on a smartphone and colored tapes deployed on the floors. According to the set up depicted in Fig. 1, the smartphone camera detects the tapes in the environment using computer vision and provides user feedback in terms of vibration signals.

This paper introduces the new system ARIANNA 2.0: it integrates PDR to provide both localization and navigation features explicitly designed for visually impaired users. To this end, the data provided by the IMU system embedded in the smartphone are merged with the additional information (when available) that can be extracted from the images captured by the smartphone camera. The key features of the system stand in the use of low a priori information, pervasive devices already available for most users (i.e., common smartphones) and light infrastructures in the environment. The designed algorithms are efficient and present a low computational burden: a preliminary study on the efficiency of the sensor fusion is proposed to foresee the reliability of the system.

The rest of the paper is organized as follows: Section II presents the addressed problem and the proposed architecture, in Section III the image processing algorithm is detailed. Section IV describes the tracking system and Section V reports the validation tests that have been performed to prove the effectiveness of the proposed method. Finally, in Section VI the conclusions are drawn.

[1] G. Galioto, I. Tinnirello, and D. Croce are with DEIM, Università degli Studi di Palermo, Viale delle Scienze, 9, 90128 Palermo, Italy and with the CNIT consortium, Italy giovannigalioto@unipa.it, ilenia.tinnirello@unipa.it, daniele.croce@unipa.it

[2] F. Inderst and F. Pascucci are with the DING, Università degli Studi Roma Tre, Via della Vasca Navale, 79, 00146 Roma, Italy, federica.inderst@uniroma3.it, federica.pascucci@uniroma3.it

[3] L. Giarré is with the DIEF, Università degli Studi di Modena e Reggio Emilia, Via P. Vivarelli, 10, 41125 Modena, Italy, giarre@ieee.org

978-3-9524-2699-9 ©2018 EUCA

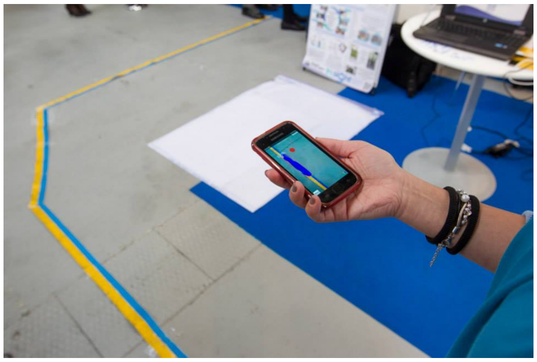


Fig. 1. An exemplary installation of the ARIANNA system.

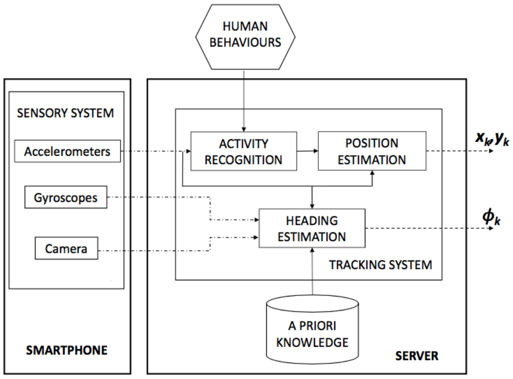


Fig. 2. Navigation system architecture: enabling integration of computer vision and dead-reckoning techniques.

II. PROBLEM SETTING AND NAVIGATION ARCHITECTURE

The key idea behind the proposed system is to track the pose, i.e., the position (x, y) and the heading (ϕ), of visually impaired users with respect to a fixed Cartesian reference frame (i.e., the *Navigation Frame*, NF) to support navigation in unknown planar environments. To this aim, the pose needs to be estimated continuously and its accuracy needs to be able to cope with the hard navigation constraints.

We assume that the environment is augmented with path traces, based on simple colored lines or tapes deployed on the floor. The user carries a smartphone in handheld configuration, as shown in Fig. 1, to detect the lines on the floor using the camera. The camera is then used as an additional sensor, which complements the information provided by the gyroscope and accelerometer available in common smartphones. Specifically, the computer vision algorithm that identifies the paths for providing a continuous feedback to the users is used to quantify the relative direction between the user trajectory and the path. For sake of simplicity, we assume that smartphone and user positions coincide, movements are allowed in a 2D environment and the user starts navigating from a known pose belonging to the path.

Fig. 2 summarizes the main functional components of the navigation system envisioned in ARIANNA 2.0: $i)$ the sensory system, for acquiring inertial data and images of the floor; $ii)$ the tracking system for processing and aggregating the data into a position estimate; $iii)$ the maps of the environment, which represent the a priori knowledge of the available paths and relevant orientation.

The sensory system is implemented by exploiting the IMU and camera embedded in the smartphone: the measurements produced by the inertial sensors and the images acquired by the camera are logged and synchronized using the timestamp. It is worth noticing that IMU measurements are sampled at higher frequency than camera images and therefore, in general, multiple measurements are linked to the same image of the floor.

The tracking system was implemented in an external server: for run-time tracking, in each update interval it is enough to sample a few images (e.g., 2 images/s) and use low resolution (e.g., 144x192 pixel), thus resulting in a limited amount of data to be transmitted (typically, 20 Kbytes for each trace update). The server extracts heading information from the sequence of images and from the map of the paths, as described in Sec. III, and runs the tracking operations presented in Sec. IV. The results are then sent back to the smartphone.

III. EXTRACTING HEADING FROM CAMERA

The basic idea developed in ARIANNA 2.0 is the use of the camera as a heading sensor. It identifies the painted lines in the acquired images, quantifies the slope of the lines seen by the camera, and converts this slope in an absolute orientation of the user, on the basis of a rough positioning of the user on the map. Indeed, the map of the paths are given by a sequence of segments, i.e., a piece-wise line, with different (absolute) orientations: it is enough to know at which segment the user can be located for converting her/his relative orientation to the line in an absolute heading measurement.

To identify the slope of the line seen by the camera, we implemented three different steps: $i)$ filtering the image, for reducing the noise and the details of the image background; $ii)$ applying the Canny algorithm, for detecting the edges of the objects in the image; $iii)$ identifying the sub-set of edges which can be considered as a line using the Hough transform.

Eliminating image details. The first step is performed by using a Gaussian smoothing filter, whose main goal is defocusing the image for avoiding that some regular patterns of the floor (e.g., such as the edges of squared tiles) can be erroneously considered as a path trace. Since the lines deployed on the floor are very thick in comparison with the tiles' edges, such a filtering operation does not affect the identification of the line edges. The filter is characterized by a parameter σ which represents the standard deviation of the Gaussian function used for smoothing the image details. Higher values of σ lead to a more evident loss of image details.

Detecting edges. The second step is given by the application of the well-known Canny scheme. The output is a binary

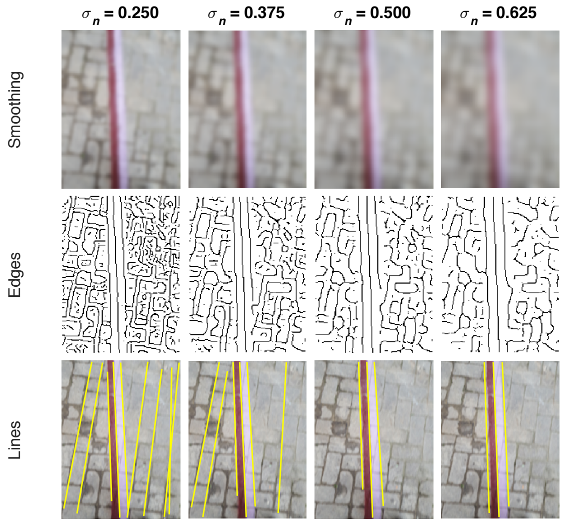


Fig. 3. Effects of the Gaussian smoothing filter on the capability of detecting the correct path trace.

matrix, whose dimension is equal to the original frame and whose values are set to 1 for the pixels corresponding to the detected edges. These pixels are identified by computing the maximum luminosity gradient (in each possible direction) for each pixel, and by selecting the pixels for which the gradient is higher than a threshold T. Higher values of T correspond to a lower number of detected edges.

Detecting lines and slopes. The last step works on the binary image found by the Canny scheme for transforming the line identification problem in a maximum search problem. The Hough transform is used for mapping each edge point in a set of parameters (ρ, θ) representing the bundle of lines passing through that point. When multiple edge points are aligned, there is a common (ρ, θ) value representing the line passing through all the points. Therefore, the scheme simply works by counting the maximum number of occurrences (i.e., votes) of quantized (ρ, θ) values computed for each edge point. The quantization allows compensating noise effects in the identification of the alignments. This step is critically affected by the resolution used for quantization and by the threshold used for assuming that a given (ρ, θ) value corresponds to a line in the image. We chose to only consider the global maximum in the parameter space, i.e., a single (ρ, θ) value receiving the greatest number of votes, because we assume that the greatest number of aligned points always correspond to the thickly painted line. However, since the painted line is actually implemented with a bi-colored stripe, such a choice implies that we are only able to detect one of three parallel linear edges along the stripe. In proximity of a turning point, such as L-like junction points, the number of path segments with a different orientation is at least equal to two. However, by always choosing the line with the highest number of votes, which usually correspond to the longest detected line, we avoid ambiguity in the reference orientation for the user.

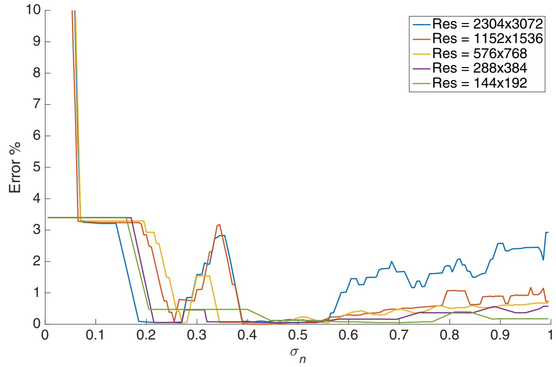


Fig. 4. Normalized error metric as a function of the standard deviation of the smoothing filter.

A. Tuning computer vision parameters

For optimizing the configuration of the above scheme, we have considered the effects of the image resolution, the smoothing factor of the Gaussian filter, and the threshold values of the Canny schemes on the accuracy of line detection.

Fig. 3 visualizes the effects of different smoothing factors on a real image of a path trace. The standard deviation is normalized as a function of the image resolution (for representing the portion of the area involved in the filtering operations). As the standard deviation increases, the image loses details as evident in the reduction of the number of edges identified by the Canny algorithm. This, in turns, corresponds to a lower number of detected lines (from 9 lines for $\sigma = 0.25$ to 3 lines for $\sigma \in [0.5, 0.625]$), which are likely to coincide with the path trace, rather than with the floor regular patterns.

For quantifying these qualitative considerations, we defined an error metric for the line with the highest number of votes, based on two different contributions: $i)$ the error in the orientation of the line (i.e., on θ); $ii)$ the error on the distance ρ between the top left vertex of the image and the line. These two components are then normalized, by considering the maximum possible values of ρ and θ, and equally weighted in the final error metric. Fig. 4 depicts the error metric for a single reference image, as a function of the standard deviation of the Gaussian filter, and for different image resolutions. Although the non-linearity of the scheme operation (quantization of (ρ, θ), on/off detection of lines, etc.) introduce some fluctuations in the error results, we can easily recognize an optimality region, i.e., a set of σ configuration values, which consistently minimize the error. Too high smoothing factors lead to worse results because the detail reduction can prevent the identification of the trace. The results are confirmed by different images.

Image resolutions have a similar effect on line detection accuracy: too high resolutions (apart from complexity issues) include too many details, which can correspond to the detection of wrong lines; too low resolutions imply rough estimates of the line orientation. We also noticed that the threshold of the Canny scheme has a limited impact on the error metric. In the next experiments discussed in the paper, we used the optimal configurations found on the basis of these considerations.

IV. TRACKING SYSTEM

A. Activity Recognition

The *Activity Recognition* identifies the type of motion performed by the user. In the addressed scenario, the user moves in a planar environment, therefore only 2 activities, i.e., *standing still* and *walking*, has been considered. The covariance of the accelerations along the three axes is exploited to classify the activities. During the *standing still* activity, indeed, these parameters are considerably reduced and are characterized only by the measurement noise. Thus, the *standing still* activity can be identified when the following relations are verified $P_j \leq \alpha_j, \quad \forall j \in \{x, y, z\}$ where $P_j = \text{cov}(a_{k-b_k,j}, \cdots, a_{k,j})$ and α_j is a threshold, set by experimental trials.

B. Heading Estimation

According to the proposed scenario, the heading changes only when the *walking* activity is detected, while during *standing still* the measurements collected from gyroscope and accelerometer are used to update the bias, that slowly changes over time. The heading estimate is performed using data from gyroscopes and accelerometer merged with the information provided by the camera. The estimate is computed by a prediction-correction filter. In the prediction step, the Attitude Filter estimates the attitude of the smartphone reference frame (i.e., the Body frame) with respect to the NF. It is based on an Extended Kalman Filter (EKF) that merges data collected from gyroscopes and accelerometers. The vector state is represented by means of quaternions: the details of the filter can be found in [16], [17] and are not reported here for sake of space. The output of the filter is the rotation matrix $\mathbf{R}_b^n$ from the Body frame to the NF, the heading γ_k and its uncertainty Γ_k. The initial condition q_0 is obtained from the acceleration and the data provided by the camera considering the user standing still when the system is activated as in [17].

The heading γ_k and the related uncertainty Γ_k feed the correction step: the output of the image processing is used to update the heading and bound the drift error. Specifically, only the heading component is considered and a simple update is performed

$$\phi_k = \gamma_k + W_k(\gamma_{C,k} - \gamma_k) \tag{1}$$

where W_k is a gain computed according to EKF equation as

$$W_k = \frac{\Gamma_k}{\Gamma_k + R}$$

where R is the uncertainty related to the heading measurement $\gamma_{C,k}$ performed by processing data from the camera and is supposed known and time-invariant. The heading provided by the camera is computed as

$$\gamma_{C,k} = \begin{cases} \theta_k + \delta_j & \text{if } \sin\gamma_k \geq 0 \\ \theta_k + \delta_j + \pi & \text{if } \sin\gamma_k < 0 \end{cases}$$

where δ_j is the slope of the segment j detected by the camera in the estimated position with respect to the NF and it is supposed a priori known.

C. Position Estimation

The *Position Estimation* is devoted to compute the position (x, y) of the user with respect to the NF. This module exploits data from the accelerometer and the rotation matrix $\mathbf{R}_b^n$ provided by the Heading Estimation. The accelerometer data are projected in the NF by using the matrix $\mathbf{R}_b^n$: in this way, the acceleration along the z-axis of the NF corresponds to the acceleration on the vertical axes of the user.

The initial position is supposed known while the position of the user during *walking* is recursively computed, by estimating the length of the stride on step event detection i

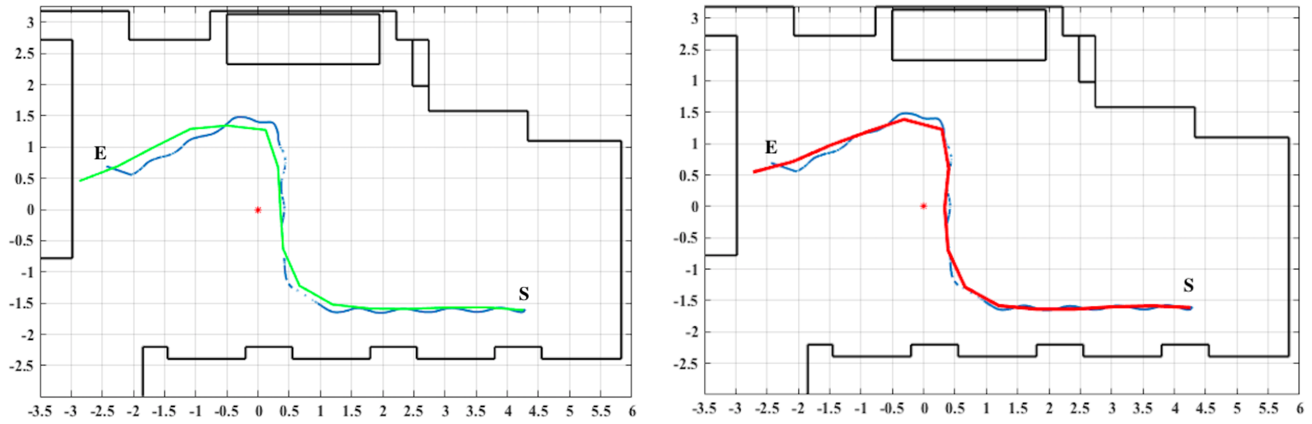
$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} x_{i-1} \\ y_{i-1} \end{bmatrix} + l_i \begin{bmatrix} \sin\bar{\phi}_i \\ \cos\bar{\phi}_i \end{bmatrix} \tag{2}$$

where l_i is the displacement of the user between two step events with respect to the NF and $\bar{\phi}_i$ is the average heading in the same time interval.

The step detection is obtained using the human body dynamic associated with the human gait cycle (i.e., the double limb support period and the single limb support period). These events are detected by analyzing the local minima and the local maxima of the vertical acceleration signal. To this end, both peak detection and zero crossing algorithms are applied to identify the sharp changes to the vertical acceleration associated with the heel strike. These features are also exploited to compute the cardinality c_i of the set of samples to be processed to cope with different walking speed. During *walking* activity, the displacement l_i is estimated as proposed in [18]: $l_i = \beta\sqrt[4]{a_{i,z}^M - a_{i,z}^m}$ where $a_{i,z}^M$ and $a_{i,z}^m$ are the maximum and the minimum vertical accelerations in the time interval $(k - c_i, k]$ and β is a parameter depending on the user and estimated by a calibration procedure. It is worth noticing that in [18] the IMU is placed near the Center of Mass (CoM) of the user and β represents the average length of a step. Here, the parameter β depends also on the position of the hand with respect to the body of the user. Finally, the user displacement is $l_i = \begin{bmatrix} 0 \, 0 \end{bmatrix}^T$ when the output of the classification phase is the *standing still*.

V. EXPERIMENTAL VALIDATION

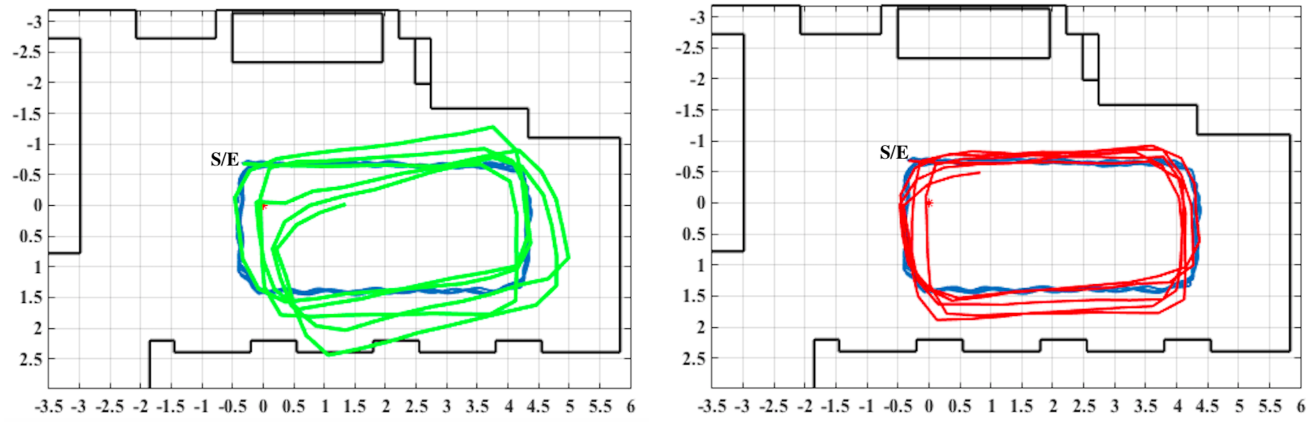
In this section, we present the results of our tracking system (TS) in indoor and outdoor scenarios to evaluate the system accuracy. The device used in these tests is the Samsung Galaxy S6 (SM-G920F), running Android 6.0.1. The sensors embedded in the smartphone are the IMU-MPU6500 by Invensense and the IMX240 camera by Sony. Data from IMU are available at sampling frequency 100 Hz, while the images from the camera are acquired at 20 Hz. In indoor scenario, the optical motion capture system OptiTrack has been used to compute the ground truth (GT). The system is composed of 10 infrared cameras and is able to identify the position of markers in a limited space having accuracy



(a) S-path: comparison among the PDR (green line) and GT (blue line).

(b) S-path: comparison among the TS (red line) and GT (blue line).

Fig. 5. Results of the S-path experiments: the (x-y) axes represent the NF $[m]$, S and E are the starting and the ending points, respectively.



(a) Q-path: comparison among the PDR (green line) and GT (blue line).

(b) Q-path: comparison among the TS (red line) and GT (blue line).

Fig. 6. Results of the Q-path experiments: the (x-y) axes represent the NF $[m]$, S and E are the starting and the ending points, respectively.

TABLE I

PARAMETERS INITIALIZATION.

PARAM	INIT VALUE	PARAM	INIT VALUE
b_k	60	σ	4
α_x	0.02	T	*adaptive*
α_y	0.01	β	0.48
α_z	0.03	R	0.02

10^{-4} m. To set the GT, the smartphone has been equipped with 4 markers: the CoM of the markers corresponds to the CoM of the smartphone. The performance index used to evaluate the accuracy of the proposed system is represented by the estimated error on checkpoints. The estimated error is represented by the Euclidean distance between the estimate (i.e., PDR or TS) and the corresponding points on the GT. In outdoor scenario, the performance index is represented by the error accumulated on the final point of the track (i.e., the end of the line of the ARIANNA 2.0 system). In Tab. I, the parameters used to initialize TS are collected during the experiments. The parameter β has been calibrated on the user adopting the procedure introduced in [16], the covariances associated to the accelerations $\{\alpha_x, \alpha_y, \alpha_z\}$ are computed at the beginning of the experiment when the user is supposed standing still for 10 s.

The first trial is a short (10 m) *S shaped* path, (see Fig. 5) carried out in an office-like environment. In Fig. 5(a), the results obtained using the PDR are shown and compared with the GT retrieved by the capture motion system.At the end of the experiment, the error accumulated along the path using PDR-only is relevant, considering that the path is short and the drift is partially compensated by the opposite curves. The TS suitably reduces the error, that is limited to 0.54 cm in the worst case.

The aim of the *Q-path* test is to estimate the accuracy when a closed loop is considered: the user repeats a squared-path in an indoor environment 5 times without stops. The overall length of the path is 130 m and the results of the trial are reported in Fig. 6. The track obtained using the PDR is shown in Fig. 6(a) and it is compared with the GT retrieved by the capture motion system. The error of the PDR estimate grows unbounded due to the heading drift: the covariance of the error estimate is larger than the one retrieved in the previous experiment. On the contrary, the error of the TS estimate is bounded since the heading is continuously corrected by

TABLE II

S and Q-Path: performance indices.

	S-Path			
	Mean err $[m]$	**Min err** $[m]$	**Max err** $[m]$	**Std dev** $[m]$
PDR	0.21	0.01	1.25	0.09
TS	0.10	0.001	0.54	0.02
	Q-Path			
	Mean err $[m]$	**Min err** $[m]$	**Max err** $[m]$	**Std dev** $[m]$
PDR	0.66	0.15	1.77	0.22
TS	0.34	0.15	0.61	0.02

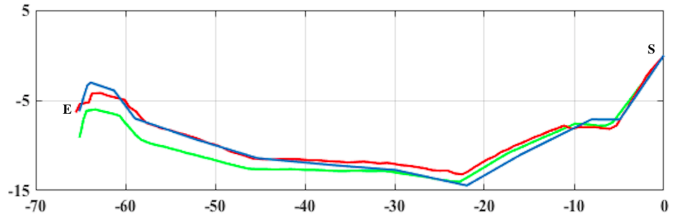


Fig. 7. Estimated long path as real case: the (x-y) axes represent the NF $[m]$, GT (blue line), PDR (green line), and TS (red line), S and E are the starting and the ending points, respectively.

the vision system and the associated covariance does not change with respect to the previous trial. In both experiments, the checkpoints are represented by the step events and the performance indices are reported in Tab. II.

In Fig. 7, an outdoor scenario is considered. The experimental trial has been carried out in Farm Cultural Park, Favara (AG), Italy, where the system ARIANNA has been installed to help the visually impaired people visit the open air museum. In this experiment the performance index is represented by the error accumulated on the final point (i.e., the end of the line of the ARIANNA system). The overall distance traveled is about 76 m and the error accumulated at the end of path using only the PDR is 3.1 m (i.e., about 4% of all the distance traveled). This error is reduced to 0.41 m (i.e., less than the 1% of all the distance traveled) by updating the estimate using the camera in the TS.

VI. Conclusions

In this paper we have presented ARIANNA 2.0, an innovative smartphone-centric tracking system for indoor and outdoor environments, based on the joint utilization of dead-reckoning and computer vision techniques. The system is explicitly designed for visually impaired people and is able to suitably reduce the navigation error with respect to the results obtained using PDR only, as demonstrated by the trial carried out using the precise optical tracking system as GT. In real outdoor tests, the system outperforms the PDR that accumulates position errors due to an inaccurate heading estimate. Future development of the system will include also a model of the hand movement and possibly the design of a vibration feedback to tell the user how to correct its location. Based on the presented tests and experiments, we have implemented and embedded all the algorithms into an app. A demo is also available and has been presented in [19].

References

[1] R. Velázquez, Wearable Assistive Devices for the Blind, Chapter 17 Smart Environment: Issues and Characterization, LNEE 75, Springer, pp 331–349, 2010.

[2] J.M. Loomis, R.L. Klatzky, N.A. Giudice, Sensory substitution of vision: Importance of perceptual and cognitive processing, In R. Manduchi and S. Kurniawan (Eds.) Assistive Technology for Blindness and Low Vision, pp. 162–191, Boca Raton: FL: CRC Press 2012.

[3] A.R. Jimènez, F. Seco, J.C. Prieto, J. Guevara Rosas, Indoor pedestrian navigation using an INS/EKF framework for yaw drift reduction and a foot-mounted IMU, Workshop on Positioning Navigation and Communication, Dresden, pp. 135–143, 2010.

[4] L. Filardo, F. Inderst, F. Pascucci, C-IPS: a Smartphone based Indoor Positioning System, International Conference on Indoor Positioning and Indoor Navigation (IPIN), 4–7 October 2016, Alcala de Henares, Madrid, Spain, 2016.

[5] A.R. Jimènez, F. Seco, F. Zampella, J.C. Prieto, J. Guevara Rosas, Improved Heuristic Drift Elimination (iHDE) for pedestrian navigation in complex buildings, Int. Conf. on Indoor Positioning and Indoor Navigation, Guimaraes, pp.1–8, 2011.

[6] F. Montorsi, F. Pancaldi, G.M. Vitetta, Design and implementation of an inertial navigation system for pedestrians based on a low-cost MEMS IMU, 2013 IEEE International Conference on Communications (ICC 2013), Budapest, Hungary. June, 9–13, 2013.

[7] B. Krach, P. Roberston, Cascaded estimation architecture for integration of foot-mounted inertial sensors, in IEEE/ION Position, Location and Navigation Symp., Monterey, CA, pp. 112–119, 2008.

[8] K. Kaemarungsi, P. Krishnamurthy, Properties of indoor received signal strength for WLAN location fingerprinting, Proc. of 1st Annual International, (MobiQuitous04), 2004.

[9] J.A.M. Ladd, K.E. Bekris, A.P. Rudys, D.S. Wallach, L.E. Kavraki, On the Feasibility of Using Wireless Ethernet for Indoor Localization, IEEE Trans. Wireless Communications, vol. 5, no. 10, October, pp. 555–559, 2006.

[10] S. Willis, S. Helal, RFID information grid and wearable computing solution to the problem of wayfinding for the blind user in a campus environment, IEEE Int. Symp. on Wearable Computers, 2005.

[11] N. Fallah, I. Apostolopoulos, K. Bekris, E. Folmer, Indoor Human Navigation Systems; A survey, Interacting with Computers, Oxford Journals. 25:1 pp. 21–33, February 2013.

[12] N. Fallah, I. Apostolopoulos, K. Bekris, E. Folmer, The User as a Sensor: Navigating Users with Visual Impairments in Indoor Spaces using Tactile Landmarks, Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems (CHI'12), pp. 425–432, Austin, Texas, May 2012.

[13] D. Croce, P. Gallo, D. Garlisi, L. Giarré, S. Mangione, I. Tinnirello, ARIANNA: A smartphone-based navigation system with human in the loop 22nd Mediterranean Conference of Control and Automation (MED), pp. 8–13, 2014.

[14] D. Croce, L. Giarré, F.G. La Rosa, E. Montana, I. Tinnirello, Enhancing tracking performance in a smartphone-based navigation system for visually impaired people, 24th Mediterranean Conference of Control and Automation (MED), 2016.

[15] Italian Patent N. BG2014A000054, Sistema di navigazione per non vedenti, presented 2015, patented 2017.

[16] L. Faramondi, F. Inderst, F. Pascucci, R. Setola, and U. Delprato, An Enhanced Indoor Positioning System for First Responders, International Conference on Indoor Positioning and Indoor Navigation, 2013.

[17] F. De Cillis, L. Faramondi, F. Inderst, S. Marsella, M. Marzoli, F. Pascucci, S. Setola, Hybrid Indoor Positioning System for First Responders, in IEEE Transactions on Systems, Man, and Cybernetics: Systems.

[18] H. Wienberg, Using the ADXL202 in Pedometer and Personal Navigation Applications, *Analog Devices AN-602 application note*, 2002.

[19] G. Galioto, I. Tinnirello, D. Croce, F. Inderst, F. Pascucci, L. Giarré, Demo: Sensor Fusion Localization and Navigation for Visually Impaired People, MobiCom 2017.