IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received October 26, 2019, accepted November 18, 2019, date of publication November 22, 2019, date of current version December 10, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2955046

An Indoor and Outdoor Navigation System for Visually Impaired People

DANIELE CROCE[1,5], LAURA GIARRÉ[2], (Senior Member, IEEE), FEDERICA PASCUCCI[3], (Member, IEEE), ILENIA TINNIRELLO[1], GIOVANNI ETTORE GALIOTO[1], DOMENICO GARLISI[4], AND ALICE LO VALVO[1]

[1]Engineering Department, Università di Palermo, 90128 Palermo, Italy
[2]Engineering Department "Enzo Ferrari", Università di Modena e Reggio Emilia, 41125 Modena, Italy
[3]Engineering Department, Università Roma Tre, 00146 Roma, Italy
[4]National Inter-University Consortium for Telecommunications, 43124 Parma, Italy
[5]Computer Science Department, Università di Roma La Sapienza, 00198 Rome, Italy

Corresponding author: Laura Giarré (laura.giarre@unimore.it)

This work was supported in part by the European Union (EU) Funded Research Projects INTERREG V-A Italia-Malta 2014–2020 "I-ACCESS Implementing the Accessibility to Urban Historic Center's Use and Knowledge."

ABSTRACT In this paper, we present a system that allows visually impaired people to autonomously navigate in an unknown indoor and outdoor environment. The system, explicitly designed for low vision people, can be generalized to other users in an easy way. We assume that special landmarks are posed for helping the users in the localization of pre-defined paths. Our novel approach exploits the use of both the inertial sensors and the camera integrated into the smartphone as sensors. Such a navigation system can also provide direction estimates to the tracking system to the users. The success of out approach is proved both through experimental tests performed in controlled indoor environments and in real outdoor installations. A comparison with deep learning methods has been presented.

INDEX TERMS Navigation, Kalman filtering, visually impaired, computer vision, dead reckoning.

I. INTRODUCTION

The possibility of exploiting Information and Communication Technologies (ICT) for supporting Visually Impaired People (VIP) and promoting vision substitution has been widely considered over the last five decades, especially with the emergence of electronic Braille displays, synthetic speech and ultrasonic sensors. Most of these solutions focus on a specific vision problem, being in general very complicated the design of a general-purpose solution for vision substitution [1]. For example, some devices have been designed for helping in the perception of spatial layouts, monitoring of the near environment to facilitate mobility, or providing general knowledge about a large scale environment to improve orientation. At the same time, wireless technologies such as GPS, WiFi, iBeacons, etc. have significantly contributed to the proliferation of localization systems, due to their pervasiveness, with the ambition of improving the independence and social inclusion of blind and low vision people [2].

The associate editor coordinating the review of this manuscript and approving it for publication was Xiao-Yu Zhang.

To support autonomous mobility of blind people in unknown environments, in [3] we presented a smartphone-based system different from previous solutions. Here, the localization functionality required for supporting orientation and autonomous mobility is implemented by interacting with GPS (in outdoor) or WiFi networks and iBeacons (in indoor), our solution is based on measurements performed by the user device only. Indeed, avoiding the utilization of GPS and other radio networks makes the system suitable for any environment, including indoor. The only requirement is the deployment of colored tapes on the floor, or any other visual marker, for indicating the possible paths. The main idea is exploiting the smartphone camera for detecting the landmarks with computer vision techniques and providing user feedback via vibration signals [3]–[5]. While the original system was designed only for letting users to follow paths, in this paper we present a new solution which integrates different sensors. We aim also to track the position of the users and to give them some feedback via phone vibration.

Tracking of a user in indoor environments is not an easy task, especially when high accuracy is needed to guide VIP.

Solutions available in literature, integrating the incremental movements estimated by the smartphone sensors, are called Pedestrian Dead Reckoning (PDR) schemes and they suffer from significant error drifts over time. We explored the performance improvements of PDR schemes that can be enabled by the exploitation of suitable Computer Vision (CV) functions, devised to provide additional heading and velocity measurements. Since our system continuously captures floor images for detecting the colored tapes, the orientation of these painted lines within the image can be exploited for measuring the heading of user movements. Moreover, the comparison of the same reference points, such as the tile corners or special visual markers, in consecutive images can be exploited for estimating the user velocity [6].

The main contributions of the paper are the following: i) demonstrating how the camera sensors can be exploited for providing measurements similar to the ones provided by the IMU (Inertial Measurement Unit) systems and characterizing the errors of these measurements; ii) quantifying the complexity of the processing required by the computer vision algorithms and discussing the best trade-offs between energy consumption and measurement availability and comparing it with deep learning solutions; iii) designing a tracking system based on the integration of IMU and camera-based measurements and evaluating its accuracy in experiments with real users.

The rest of the paper is organized as follows: Section II presents the state of the art in VIP navigation. Section III presents the system architecture. In section IV we give details on the computer vision algorithms. Section V describes the IMU based Pedestrian Dead Reckoning system. Section VI illustrates our Tracking System, by performing a data fusion and integration of the CV and PDR systems. In Section VII we show the success of the system in validation experiments and, finally, in Section VIII we draw some conclusions.

II. RELATED WORK

As mentioned before, many approaches have been used to support people orientation and navigation. Some are based on the triangulation of radio signals, such as WiFi, some on direct measure of landmarks (with RFIDs, Ultrasound, Bluetooth, Beacons, etc.), and others on ego-motion sensing with *Inertial Measurement Units* (IMU, i.e., accelerometers, magnetometers, and gyroscopes). This last approach (PDR) is accurate in low-medium range, but it suffers from drift due to noise [7] on long distance. Method to reset the IMU bias can be found in [8], where the knowledge of the pedestrian movement is used; or in [9], [10] where an Extended Kalman Filter (EKF) is exploited to compensate the drift. Other solutions adopt reference points provided by *fingerprinting* map such as in [11], [12]. Despite their limitations, such dead-reckoning solutions have been largely employed in real navigation systems [13], [14] and in the Navatar [15] system. Another approach is the adoption of neural network. In [30] two deep Convolutional Neural Network model are presented in order to assist the mobility of blind and visually impaired people. Unfortunately, as described below, this solution is not very efficient in term of time complexity. Concerning navigation systems for blind users, clearly, stricter requirements need to be fulfilled, in terms of accuracy and reaction times. Moreover, the cosmetic aspect is important to be included as part of the design, because the acceptance from the blind community is crucial. Some approaches have adopted the sensory substitution method, such as the one based on LIDARs [16], or in [17] where vibro-tactile stimulation is applied to the hand palms to code and give directions through a temporal sequence of the stimuli. To achieve safe navigation a vibrating belt together with time-of-flight distance sensors [18] or cameras [19] have been used. Indoor navigation has been tackled also based using the fusion of radio-based markers and visual landmarks [20] or with computer vision systems as in [21].

The importance of training blind people to safely and independently travel is addressed in [22], even if in unknown or unfamiliar complex location such autonomous navigation often fails. A number of studies look into their experiences and the challenges they face in general mobility [23] and [24]. For example, in airport applications the state of the art of navigation system for blind mostly rely on the use of beacons [25]. These systems provide an intermittent reference to the user rather than a continuous reference signal similar to the one currently provided by the tactile paving. On the other side, tactile paving does not provide information about the environment and even about the path destination. We envision then the need of proposing a solution able to provide a reliable and continuous system to blind users while tracking their position within a predefined set of possible paths. Our system has been tested involving blind users. Results have been very encouraging, with a satisfaction grade of 90 deg, [4]. Moreover, we have developed and tested [26] a Game (in.line) to help the blind users learn and familiarize with our system and to act on the spatial skills, in order to possibly acquire an allocentric spatial representation.

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.

III. NAVIGATION SYSTEM DESCRIPTION

In this paper we work with a localization and navigation system called ARIANNA (pAth Recognition for Indoor Assisted NavigatioN with Augmented perception). The system is able to work both in indoor and outdoor scenarios, especially designed for environment where GPS-based solutions are not available. The main idea is exploiting a common smartphone as a mediation instrument between the user and the reality: the smartphone camera oriented towards the floor allows the detection of colored tapes or painted lines deployed in

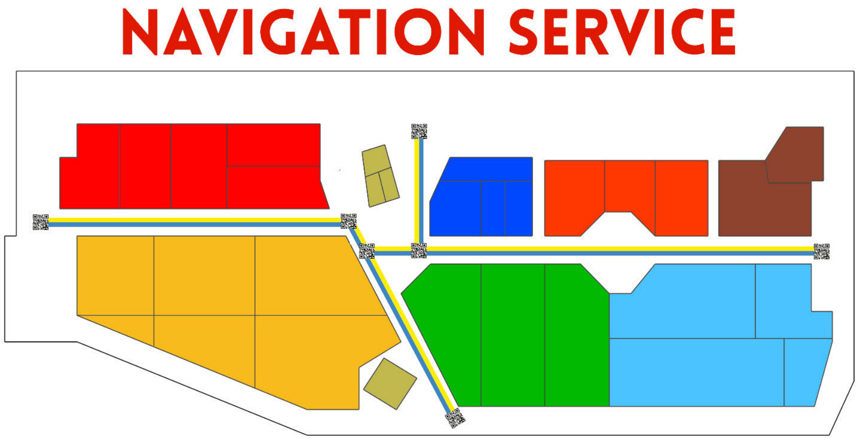


FIGURE 1. Installation of the ARIANNA system: an example.

the environment as special paths, while a vibration signal provides a feedback as long as the user is moving along the path. In other words, our system allows replacing the special tiles deployed along the tactile paving with flexible and easy to deploy painted paths. Moreover, differently from usual tactile paving, ARIANNA can provide additional information about the environment, such as a digital representation of the available paths and destinations, as well as navigation functionalities towards desired points of interest.

A key component of ARIANNA is the computer vision algorithm which recognizes the painted line on the floor, under varying environmental light conditions. There are many different computer vision functions that can be combined for the identification of the painted line, taking into account the constraints of our problem: i) the path identification has to be prompt and reliable, without perceivable latencies for the users, which could correspond to discontinuous signals; ii) the lifetime of the smartphone battery has to be compatible with the timing required, in order to guarantee the practical usage of the system. These constraints correspond to the identification of robust solutions, with limited complexity, able to work in real-time. Along the path, the system also permits to find some points of interests by detecting landmarks (such as QRcodes or iBeacons) and retrieve location-based information. The implementation of our system requires that the computations are all running in real time on-board of the used smartphone. In terms of computation we mean, as detailed below, both the computer vision and the sensor fusion with the inertial dead reckoning algorithms.

Fig. 1 shows a map with a typical installation of ARIANNA, where the colored paths are represented by two colors (blue and yellow), which make the identification of the lines more robust to other similar colors present in the environment, and allow the application to immediately retrieve a direction information. Colored paths can be posed on the floor or carpets, painted on the outdoor pavements or even deployed with aerial tapes, thus enabling flexible and low cost installations. Additional landmarks, such as QRcodes or iBeacons, can be settled close to the points of interest or on line intersections to add audible information to the user upon request or to reset our algorithm in proximity of a change in direction. QRcodes are generally more precise in terms of positioning, although more visible impacting the aesthetic of the installation, while iBeacon can be hidden and be a preferable choice in cultural environment such as museums. A data network connection is optional but if available it can be used to download additional information such as the ambient map (e.g. via a WiFi or cellular network). Finally, The user interface adopts tactile stimuli (vibration of the smartphone) to provide feedback. It has been shown that the current consumption of typical vibration motors has a limited impact on the battery life of commercial smartphones [27] and that the energy savings coming from switching off the screen are higher than the costs introduced by vibrational cues [28]. Unlike other approaches in haptic interfaces, our solution does not need a selective vibration of the touched point (that is also difficult to obtain and requires special piezo-electric materials, etc.).

Although the main sensor of the system is represented by the smartphone camera, in this paper we extend the ARIANNA system with a localization function, able to track the user pose (position and heading) along the colored paths. To this purpose, we propose to exploit the inertial (IMU) sensors available on common smartphones for estimating the incremental movements of the users and correlating such movements with the images acquired by the camera. Thanks to the usage of a common clock for adding time-stamps to the images and the IMU measurements, despite of the different acquisition frequency of cameras and sensors, we demonstrate that we can improve the reliability of direction estimates provided by the IMU sensors, as well as velocity measurements estimated by the computer vision algorithms. The overall tracking scheme is able to localize the users along the painted paths with an accuracy much better than usual dead-reckoning schemes. In Fig. 2, we report the general architecture on how the different sensory system

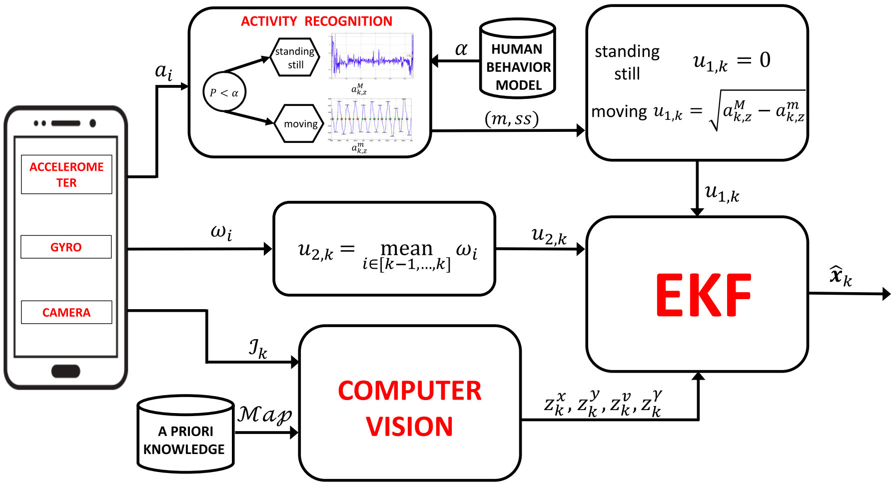


FIGURE 2. **Tracking and Navigation system architecture integrating Computer Vision and Dead-Reckoning techniques.**

are connected and integrated to obtain a reliable system, as described in the following sections.

We can envision several opportunities in adding a localization function to ARIANNA. Users can be assisted towards their destination by also receiving information on the residual distance. Moreover, context information about the surrounding environment, services or points of interests can be dynamically updated. Positioning information can also be exploited in case the users lose the painted path, because for small drifts the positioning accuracy is good enough to bring the user back to the painted line.

IV. COMPUTER VISION ALGORITHMS AS MEASUREMENT SENSORS

The ARIANNA system detects the colored paths on the floor via computer vision algorithms. These algorithms can be extended for providing additional information devised to track the user along the path. Indeed, the slope of the path identified in the camera image can be mapped into a heading measurement, while the comparison between images captured into consecutive frames can provide a velocity measurement, as detailed in the following subsections. In other words, the smartphone camera and the image processing can work as a sensor for providing measurements similar to the ones generated by inertial sensors.

Two main features can be exploited for detecting the paths: the geometry of the tapes (which in the end are given by piecewise lines), and the colors of the tapes (which combine two different colors for representing a direction without ambiguity). We combined two solutions focused on both the geometric factors and the color identification, with different complexity and detection capabilities. Once the path has been identified and mapped into a blob of points within the image, we applied other image processing functions for computing the heading and velocity measurements. Complexity is a very important metric to be considered, because energy consumption is critically affected by it: apart from a fixed amount of energy required for the smartphone operating system, the additional energy consumed by our system is roughly proportional to the algorithm complexity.

We also characterized the errors of these measurements. To this purpose, we recorded the images captured by a smartphone while a reference user was walking in a round path in a room, at the University of Roma Tre, equipped with the OptiTrack system, composed by 10 infrared cameras and able to detect the markers with an accuracy of 10^{-4} m, thus providing a ground truth of user positions over time.

A. HEADING MEASUREMENT

1) GEOMETRY-BASED PATH IDENTIFICATION

An obvious solution for detecting a path is searching lines into the images, by using the well-known Canny algorithm, which is able to identify a set of edge points in an image by delimiting areas with large luminance changes. This set of points can be associated to a path whenever they result aligned. This analysis also provides the slope of the aligned points, which correspond to the user orientation along the path. In case a map of the paths is known and the user can be positioned (even roughly) on this map, the relative heading of the user can be converted into an absolute orientation.

To identify the line seen by the camera, our algorithm is composed by three 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.

a: 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 tile 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.

b: DETECTING EDGES

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

c: 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.

The above scheme can be optimized by considering 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. 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.

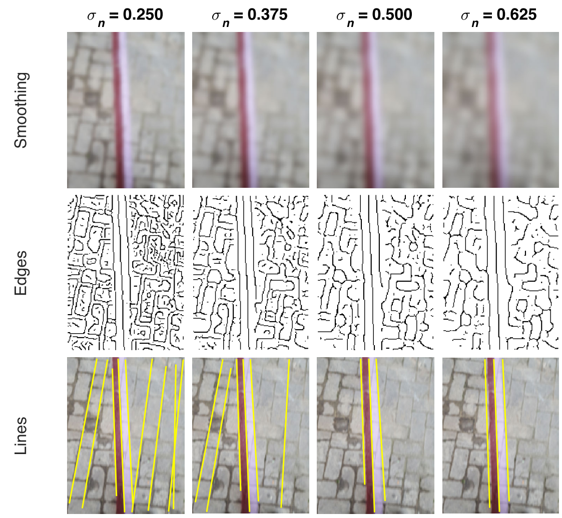


FIGURE 3. Effects of the Gaussian smoothing filter on the capability of detecting the correct path trace.

Regarding complexity, we can roughly estimate the number of operations required for detecting the path slope by considering that we work on colored images scaled to $N = 192 \cdot 144$ pixels. Being m the dimension of the Gaussian filter, the first smoothing operations requires to run m multiplications per-pixel, with a total complexity of $m \cdot N$. The Canny scheme has an additional complexity of $4 \cdot N \cdot \log N + N$ (i.e. four convolutions on the whole image and a per-pixel comparison with a threshold), while the final Hough transform requires other p operations for each edge point detected by Canny, if p is the total number of quantized $(\rho\theta)$. Assuming that the edge points are αN, the total complexity is $\alpha N \cdot p + m \cdot N + 4\, N \log N = o(N \cdot \log N)$. However, the number of quantized values p considered by Hough is usually much higher than N and the predominant term can be $\alpha N \cdot p$.

In order to compare the complexity of the two different approaches for detecting the painted line, we compared the execution times of different functions implemented in MATLAB, where execution times can be easily monitored for each specific sub-routine. Although the implementation within the ARIANNA application is obviously different (being based on the Open CV libraries), we expect that the relative behaviors of the functions can be similar. Table 1 summarizes the execution times of the processing required for the heading measurement according to the geometric-based approach implemented in MATLAB. The times are referred

TABLE 1. Execution times of most relevant computer vision functions on a reference implementation.

Function	Parameter	Execution time [ms]
Smoothing	m=5, N=192 · 144	2.2
Canny	N=192 · 144	5.0
Canny	N= 96 · 72	3.7
Hough	p=1200, N=192 · 144	21.2
Hough	p=1200, N=96 · 72	3.9
Hough	p=120, N=192 · 144	1.8
Moment	N=192 · 144	1.35

to a Windows Operating system and an Intel i7-4510U quad-core processor at 2GHz. From the table we confirm that Hough is the most complex task and its complexity is proportional to N and p. We can conclude that the Hough transform takes most of the energy absorbed by the execution of the geometric-based identification function.

2) COLOR-BASED PATH IDENTIFICATION

Another possible solution for identifying the path is detecting sub-areas of the image characterized by the same colors of the tapes. Image colors can have different representations, among which the HSV (hue, saturation, value) representation is more robust to luminance changes. In principle this approach can be quite simple, because it works by applying conditional statements on each image pixel, in order to verify if it belongs to the desired intervals of hue, saturation and values. The complexity required by this filtering operation is proportional to N, thus resulting much lower than the complexity required by the previous approach. Moreover, the direction information can be easily extracted from the blob of points identified as belonging to the path, by evaluating the second-order moments of the blob, whose complexity is equal to the number of blob points selected into the image size. Assuming that the blob points are usually about 1/8 of the total image, the overall complexity is $1/8 \cdot N + N = o(N)$. Table 1 quantifies the execution time required for computing the second order moments of the blob points in MATLAB for the same reference computer used for the analysis of the geometric-based approach.

Despite of the lower complexity, the approach based on color searches has the drawback of being potentially affected by the light conditions in which the camera works. Indeed, it is not obvious how to define absolute HSV intervals for identifying the path under variable (even on a day-basis) light conditions. Schemes removing shadows on the images or correcting luminance changes can be more complex than edge-based solutions. We therefore implemented a mixed algorithm, in which the two schemes are opportunistically combined: ARIANNA normally works by applying a simple color-based detection, but at regular time intervals of 1 *sec* the geometry-based scheme is executed for delimiting the tapes and re-sampling the color intervals. Since we sample the camera at a rate of 10 frames/sec, the overall consumption is given due to the color-based identification function for 90% and to the geometric-based function for the remaining 10%.

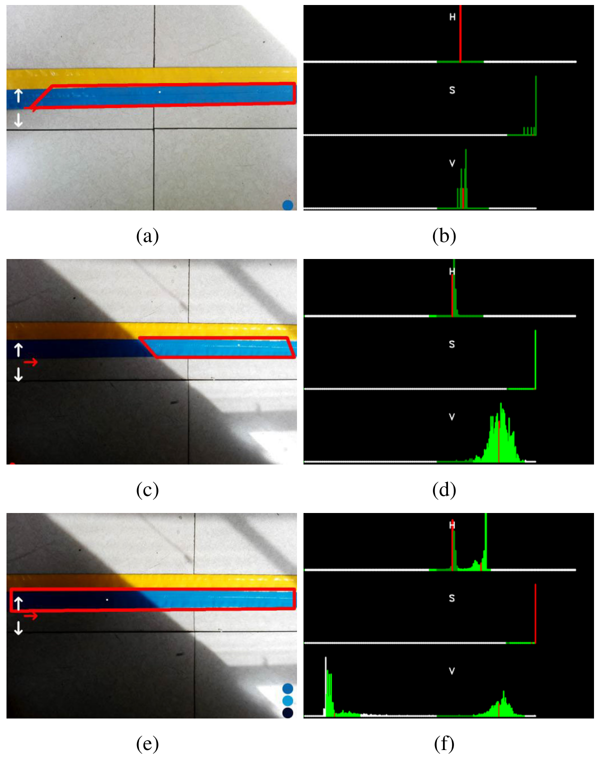
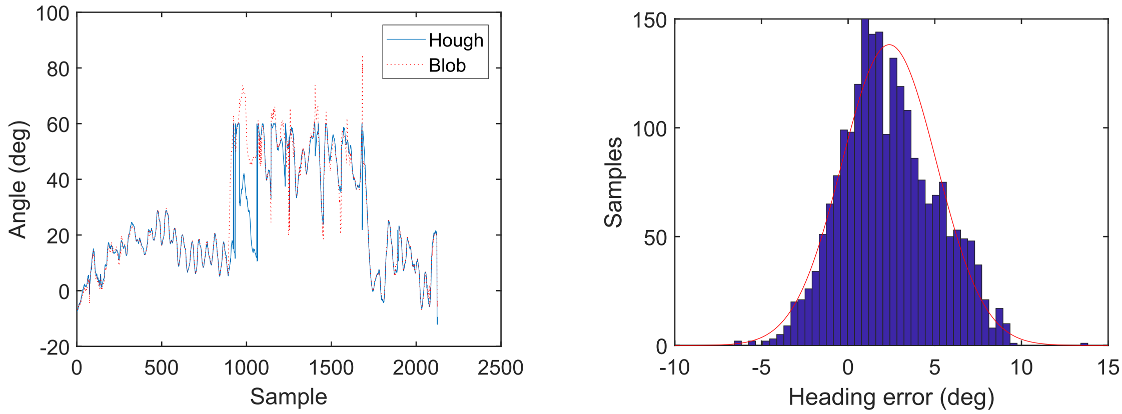


FIGURE 4. Effects of the dynamic resampling of HSV filtering values.

Fig. 4 shows an example of dynamic re-sampling of the HSV values used by the color-based path identification approach. In the left part of the figure, we can see the bi-colored tape under three different light conditions. A red delimiter is used for segmenting the image area recognized as belonging to one color of the path. In the right part of the figure, we also show the HSV distributions of the points belonging to the segmented area. In the top case, when the light conditions are almost uniform, we can see that the HSV distributions are very narrow and color recognition by means of HSV filtering works well. In the middle case, we see that the HSV filtering allows to recognize only one part of the path, not covered by the shadowed area. Only after the execution of the geometric-based path identification function, the whole path captured in the image is correctly identified and the HSV values can be re-sampled, leading to the bi-modal distributions shown in the bottom case of the figure. These distributions can be used for applying the color filters to the subsequent image frames, until the user moves completely to the shadowed area. At this point, the next execution of the geometric-based identification function will update the HSV values to new single-modal distributions typical of the shadowed image.

3) CHARACTERIZATION OF ERRORS

Fig. 5 compares the heading measurements obtained by processing a real trace of images with both the geometric-based and color-based algorithms for path detection. The trace has been recorded in a room equipped with the OptiTrack system



(a) Hough vs Color
(b) Error distribution

FIGURE 5. **Performance of the geometric-based and color-based algorithms and comparison with the OptiTrack ground truth.**

in order to have a ground truth of user positions over time. From the figure, we can see that both the schemes provide similar results in the first portion of the experiment (i.e. for image samples lower than 800), while they differ for a while in the second portion. By comparing the results with the ground truth, we found that the origin of mismatching is due to an abrupt corner along the path. While the Hough scheme suddenly switches from one slope to another (being the result equal to the direction of the longest line detected into the image), the color-based scheme provides the average orientation of the whole blob of points.

By using the heading information acquired by the OptiTrack, we observed that the Hough scheme provides heading measurements with negligible errors, but these measurements are not always available. The overall error on the heading measurements provided by our proposed combination of geometric-based and color-based detection scheme is characterized in figure 5(b). The figure shows that we can approximate the errors with a Gaussian distribution with 2.35 deg mean and 7.39 deg^2 variance.

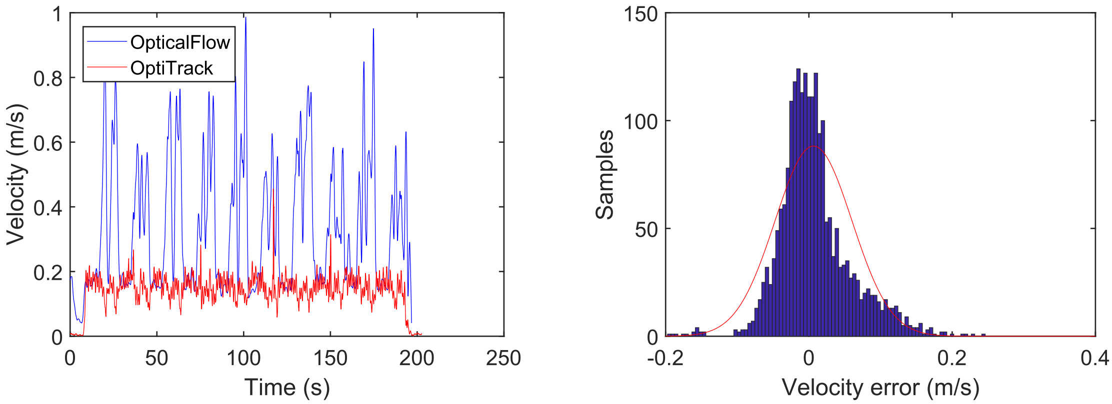
B. VELOCITY MEASUREMENT

We investigated the possibility of exploiting computer vision for providing a measurement of velocity. While the previous schemes identify the heading of the movement, i.e. the orientation of the velocity vector, it is possible to design some extensions devised to also measure the velocity module. The idea is identifying some reference points in two different images acquired by the smartphone camera and evaluating the velocity module by quantifying the translation of the points and the time interval between the two images. For quantifying the translation, it is possible to properly scale the distance between two pixels of the image. For example, the thickness of the tapes used for deploying the paths, whose size is known, can help for estimating the scaling factor between distances on the images and distances in the real word. The above technique is implemented in the so called *Optical Flow* schemes, usually adopted for tracking the movements of objects in images captured by a fixed camera, [6]. We adapted the Optical Flow library functions provided in MATLAB for working with a moving cameras oriented towards fixed path. The scheme returns both the rotation and the translation of the smartphone between two consecutive image samples, provided that the image includes some points, such as tile corners, markers on the path, etc., that can be easily recognized in two different images.

Fig.6 shows the velocity estimates provided by our implementation of the Optical Flow when applied to the image traces acquired in the OptiTrack experiment. From the figure, we can see that the errors alternate intervals where the errors are very small and intervals where the error grow significantly (similarly to the heading measurement results). By synchronizing the results with the OptiTrack ground truth, we observed that the errors grow in proximity of the corners: in other words, the velocity measurement works well only in case of translations with limited rotations. By dropping the measurements relevant to the image samples for which the rotation is higher than a given threshold, we characterized the measurement errors when the user moves without abrupt direction changes (i.e. in the path segments between two consecutive corners). Fig. 6(b) shows the empirical distribution obtained for the filtered measurements.

C. COMPARISON WITH A DEEP LEARNING APPROACH

Since neural networks are currently receiving a lot of attention in the field of computer vision, we also investigated on the comparison between our proposed measurement functionalities and alternative approaches based on deep-learning. In particular, we considered the solution proposed in [30] for guiding a robotic guide-dog along a pre-defined painted path (identical to our path) by means of a Convolutional Neural Network (CNN). The network works on the images acquired by three different fisheye cameras by classifying the dog movements as aligned to the path, drifted on the left or drifted on the right of the path. The classification output is then used for correcting the dog movements in order to keep it on



(a) Sample trace

(b) Error distribution

FIGURE 6. Performance of the Optical Flow velocity estimation against the OptiTrack ground truth.

the path. Despite the simplicity of this classification function, which does not provide a heading measurement but only a quantized direction of the user relative to the path, the number of operations required by the proposed network structures are not negligible. More specifically, the paper considers two deep CNN structures for supporting the path-following mechanism of the dog robot: CaffeNet [31] and TrailNet [32]. Both models consist of different convolutional, max-pooling and fully connected layers, with a dominant contribution on the complexity given by the convolutional layer (the other layers usually contribute for no more that 5-10% [33]). Each convolutional layer has a kernel/filter that consists of a square matrix K. The filter moves to the right (and down when the row is complete) with a certain stride value until the entire image is traversed. With a *stride length* $= 1$ the kernel shifts of one position for each step and every time it performs a matrix multiplication operation between K and the portion P of the image over which the kernel is overlapping. We considered TrialNet for the complexity calculation, considering that it is a small network compared to CaffeNet. For this network design, a generic convolutional layer has $y = 32$ outputs, a kernel size $k = 4$ and a *stride* $= 1$, so the complexity for a color RGB ($c = 3$) squared image of $N = n \cdot n = n^2$ pixels is equal to $O(c \cdot n^2 \cdot k^2 \cdot (n - k + 1)^2 \cdot y)$ that can be simplified in $O(3n^2 \cdot 16 \cdot (n - 3)^2 \cdot 32) = O(n^4) = O(N^2)$. As it is possible to see from the latter result, the complexity of a single convolutional layer is greater than the complexity of our algorithms, even for the geometric-based approaches. For dealing with such a complexity in real-time, the authors in [30] used an NVIDIA Jetson TX1 embedded system with 4 GB of GPU/CPU memory. Indeed, although convolution neural networks are very effective for recognizing objects of different shapes and complexity, our specific problem refers to a linear path, whose geometrical identification is very easy with edge detection and line identification algorithms.

V. IMU BASED DEAD RECKONING

The IMU-based dead reckoning is usually relying only on measurements of the accelerometers and the gyroscopes. It is possible to implement a tracking system that is devoted to understand the activity recognition to estimate both the heading and the position.

A. ACTIVITY RECOGNITION

The *Activity Recognition* is devoted to classifying the human motion. This step is crucial, since according to the output of this subsystem, different models can be applied to track the displacement of the user. This step is also difficult to be performed when the number of motions to be identified is high, however, in this set up only a limited number of activities have been considered.

Most of the applications (indoor and outdoor) of our system are organized in planar environments. The user walks in the environment and stops at certain points of interest to appreciate the artworks (cultural sites) or the shops (commercial sites). We can use landmarks along its path to identify the possible stops and the floor or the area, so the user moves in a two-dimensional space and only *standing still* and *walking* activities need to be recognized. These two activities can be easily detected by exploiting the accelerations recorded by the IMU, as proposed in [34], [37]. Specifically, the accelerometer covariance along the three axes can be analyzed to distinguish among this two activities because when the user is *standing still* these parameters are considerably reduced and mainly characterized by the noise.

B. HEADING ESTIMATION

The *Heading Estimation* measures the direction of the user. Such additional information is used in the sensor fusion algorithm as described in the next section. Such heading changes only when the user is *walking*. As reported in previous works [29], [34], the heading estimate can be evaluated using data from gyroscopes, to provide the heading of the smartphone reference frame (i.e., the Body frame) with respect to the Navigation Frame (NF), i.e. a fixed Cartesian reference frame. Data from gyroscopes are affected by bias that downgrade the accuracy of the estimate. To reduce this effect, the bias is recalculated when the user is not walking and the *standing still* activity is detected.

C. POSITION ESTIMATION

The *Position Estimation* evaluates the user position (x, y) with respect to the NF. According to the hand-held position of the smartphone, the acceleration along the z-axis lies on the user sagittal plane, perpendicular to the floor. This signal is fundamental during the *walking* activity, since it is analyzed to identify the step event. Specifically, local minima α^m maxima α^M of the vertical acceleration signal is retrieved to evaluate both peak detection and zero crossing detection. We also exploit such features to evaluate the cardinality c_i of the set of samples in order to cope with different walking speed. Some learning techniques on the first steps is also implemented to adapt the algorithm to different users. When the user is *walking*, the displacement δ_s is evaluated with the algorithm proposed in [35]: $\Delta_s = \beta\sqrt[4]{\alpha(k)^M - \alpha(k)^m}$ where $\alpha(k)^M$ and $\alpha(k)^m$ are the actual (time k) maximum and minimum vertical accelerations. Notice that while in [35] the IMU is placed near the user Center of Mass and β is the average length of a step, here, such parameter β depends also on the position of the hand with respect to the user body. In our setting we will estimate also such parameter during the sensor fusion implementation. The initial position and heading are supposed known while the user position while *walking* is recursively computed. - Finally, the user displacement is set to zero is the user has been classified as *standing still*.

VI. TRACKING SYSTEM WITH DATA FUSION

To obtain an efficient tracking system specifically designed for VIP which provides localization and navigation features we combine the efficient and lightweight CV and PDR algorithms. The sensor fusion is obtained by implementing an Extended Kalman Filter according to the following model. The general scheme has been anticipated and described in Fig. 2.

A. STATE MODEL

Assuming that the smartphone and user positions coincide, we can consider only movements in a 2D environment (i.e. for simplicity we do not take in consideration changes in user elevation). We consider a five-dimension discrete state model, where the first two components are the 2D coordinates of the user, the third component is the displacement Δ_s of the user and the fourth component is the direction γ of the user velocity. The fifth component is the parameter β which takes in to account the step length and height of the user affecting how the displacement of the user is related to velocity.

Let x and y be the 2D coordinates of the user with respect to the NF, and Δ_s the displacement, and γ the direction of user velocity. We assume that the human correction actions work on the direction of the movement. Under these assumptions, we can consider the following discrete-time state model:

$$\begin{cases} x(k+1) = x(k) + \Delta_s(k)cos(\gamma(k))T + w^x(k) \\ y(k+1) = y(k) + \Delta_s(k)sen(\gamma(k))T + w^y(k) \\ \Delta_s(k+1) = \beta(k)u_2(k) + w^s(k) \\ \gamma(k+1) = \gamma(k) + u_1(k)T + w^\gamma(k) \\ \beta(k+1) = \beta(k) + w^\beta(k) \end{cases} \tag{1}$$

where T represents the time interval for the update, $u_1(k) = \omega_z(k) - b$ (where ω_z is the angular velocity detected by the compass and b is the compass bias) and $u_2(k) = \sqrt[4]{\alpha_k^M - \alpha_k^m}$ are two input signals provided by the IMU as detailed in Section V. Such information provided by the PDR algorithm are used in the state model as external inputs. The additive noise components represent random variations that can be assumed as Gaussian distributed and characterized by a covariance matrix Q.

B. MEASUREMENT MODEL

As measurements we use the ones obtained by the smartphone sensors and the information that can be inferred by the environment, generally provided at regular time intervals T. Some specific components can be available only at specific timing, (e.g. when the smartphone camera is able to read a landmark). More into details, the following measurements have been considered:

- *user heading* (z^γ): user movement direction, measured by means of the computer vision algorithm.
- *user absolute velocity* ($z^{|v|}(k)$: user velocity, measured by means of the computer vision algorithm.
- *user coordinates* (z^x *and* z^y) : user position, read and available only when the landmark is detected or in proximity of a changing direction.

Being ν the generic noise on the measurement components, we can map the measurements to the state vector by using the following equations:

$$\begin{cases} z^x(k) = x(k) + \nu^x(k) \\ z^y(k) = y(k) + \nu^y(k) \\ z^\gamma(k) = \gamma(k) + \nu^\gamma(k) \\ z^{|v|}(k) = \Delta_s(k)/\Delta T(k) + \nu^{\Delta_s}(k) \end{cases} \tag{2}$$

where each relevant measurement is related to its physical meaning trough its apix. Note that the measurement on the angle and the displacement are available at each step of the user, while the other measurements are available only when some landmark is in visibility. The noise in the measurements has been characterized as described in Section IV-A3.

VII. EXPERIMENTAL VALIDATION

The evaluation of our system is tested in indoor laboratory and in outdoor real installations to assess the system accuracy. We have used a Samsung Galaxy S6 (SM-G920F) device, running Android 6.0.1, with embedded sensors as the IMU-MPU6500 by Invensense and the IMX240 camera

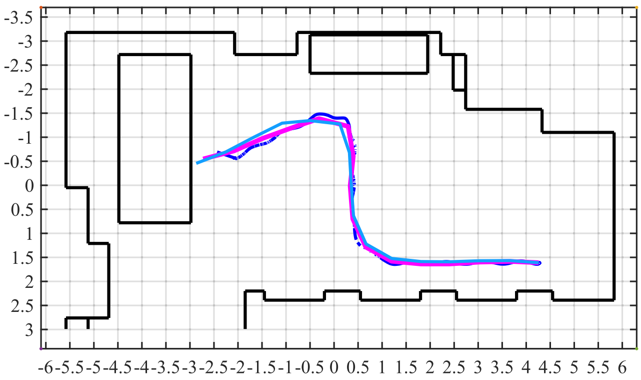


FIGURE 7. S-path: comparison among the PDR (light blue line), GT (blue line) and TS (pink line), the (*x*-*y*) axes represent the NF [*m*].

by Sony. We collect data from the IMU at a sampling frequency 100 *Hz*, while from the camera we acquired images at 20 *Hz*. The indoor scenario is performed in a room equipped with the OptiTrack system. Such capture motion system is composed by 10 infrared cameras and able to detect the markers with an accuracy of 10^{-4} *m* letting us compute the Ground Truth (GT) of user positions over time. To this purpose, the smartphone has been equipped with 4 markers, and the Center of Mass (CoM) of the markers corresponds to the CoM of the smartphone. We evaluate the accuracy of our system using as performance index the estimated error on checkpoints. Such estimated error is the Euclidean distance between the estimated position through the Tracking system or only the PDR algorithm and the corresponding points on the GT.

In outdoor scenario, instead, we use as performance index the error obtained on the end of the line, the accumulated error on the final point of the track. The experimental tests has been carried out in the Farm Cultural Park, Favara (AG), Italy, where the system ARIANNA has been installed to help the VIP visit the open air museum, Fig. 11.

In all the experiments we compare our Tracking System (TS) with a PDR-only system (PDR). The latter is implemented according to [29] and running without the information coming from the CV algorithms.

In Tab. 2, we report all the parameters used to initialize TS. During an initial phase of the experiments, when the user is not moving for 10 *s*, we evaluate and collect the covariances associated to the accelerations.

The first experiment is a short (10 *m*) *S shaped* path, (see Fig. 7) carried out in an office-like environment. The results obtained using our TS and PDR only are shown and compared with the GT retrieved by the capture motion system. AS can be seen by the results, at the end of the experiment, The PDR-only system shows a relevant error that has been accumulated along the path. Considering also that the path is short and the drift is partially compensated by the opposite curves, we notice how our Tracking System outperforms and suitably reduces such error. In the worst case, we the error is limited to 0.54 *cm*.

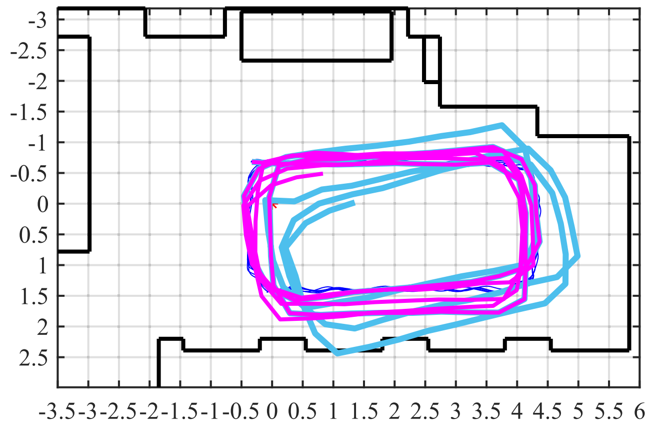


FIGURE 8. Q-path: comparison among the PDR (light blue line), GT (blue line) and TS (pink line), the (*x*-*y*) axes represent the NF [*m*].

TABLE 2. Parameters initialization.

PARAM	INIT VALUE	PARAM	INIT VALUE
α_x	0.02	T	$adaptive$
α_y	0.01	β	0.48
α_z	0.03	R	0.02

TABLE 3. S and Q-Path: performance index.

	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

The second indoor experimental test (*Q-path*) is aimed at evaluating the accuracy in a closed loop setting. The user repeats a squared-path 5times without stops. The total path is now 130 *m* length. In Fig. 8 we report the comparison of the results obtained in such experiment, by the PDR-only system and our TS with respect to the GT evaluated by the capture motion system. The PDR estimate error is growing with no bound, because of the heading drift: the error estimate covariance is much larger than the previous experiment. We notice, on the contrary, the TS error estimate is bounded, because in this case the vision system is constantly correcting the drift in the heading. Now the TS covariance is comparable to the previous trial values. In both experiments, the step events represent the checkpoints and we report in Tab. 3 the obtained corresponding performance index.

In Fig. 9, we report the results obtained during our outdoor experimental test. As previously mentioned, now we accumulate the estimation error till the final point and use this as performance index. Such final point is the end of the painted line on the ground. The overall distance traveled is about 76 *m*. Using the PDR-only system we obtain an accumulated

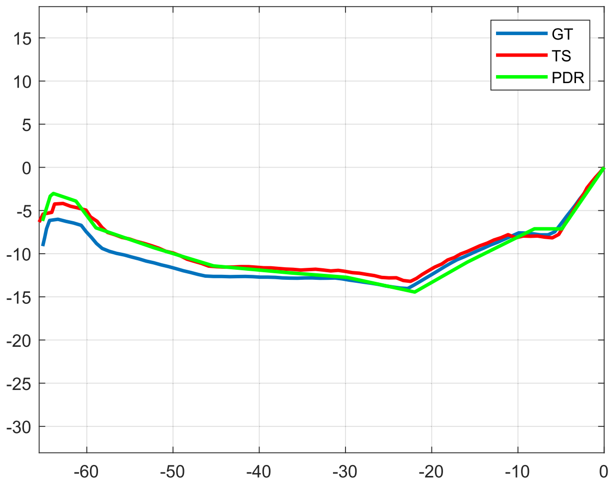


FIGURE 9. Outdoor scenario: (x-y) axes represent the NF [m], GT (blue line), PDR (green line), and TS (red line).

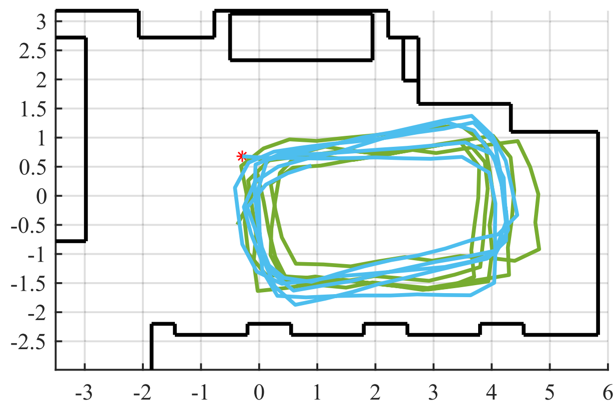


FIGURE 10. Comparison of PDR (light blue line) and NN approach (green line).

error of 3.1 m, corresponding to a 4% of the entire distance. This error is then reduced to 0.41 m, less than the 1% of the traveled distance, when our TS is used.

Finally, Fig. 10 compares the results obtained by using PDR and Deep Learning. Specifically, we used the CNN proposed in [30] in order to extract positioning information with respect to the path through a simple tag ("Left" or "Right", i.e. +1 if we are at the right of the path and −1 if we are to the left of the path). The heading computed by using CNN has been used in the update step. Although the path retrieved by the CNN turns to be more accurate than the PDR, it does not overcome the results obtained using the heading correction of the ARIANNA system. The CNN provides only a quantized absolute heading, that is able to partially compensate the errors accumulated due to the gyro drift but is still less precise than the ARIANNA system. Although more complex algorithms could be implemented to improve the precision of the NN, this comes at a cost of high computational complexity (as discussed in Section IV-C), which becomes unfeasible for mobile hardware such as a normal smartphone.

FIGURE 11. ARIANNA system installation at the Farm Cultural Park in Favara (AG).

VIII. CONCLUSION

An innovative smartphone-centric tracking system is described. This system is designed for indoor and outdoor environments and it is based on both dead-reckoning and computer vision algorithms jointly used. Specifically, we demonstrated that special reference signals deployed into the environment, such as the colored tapes envisioned for replacing the tactile paving, can be exploited for turning the smartphone camera in an additional sensor. Indeed, by processing the images acquired by the camera, it is possible to provide estimates of user heading and velocity, while the detection of special landmarks, such as corners or visual markers, can completely reset the tracking errors in proximity of known coordinates. Measuring the heading by means of computer vision functions is more reliable than gyroscopes, with a maximum error of about 5 deg. Conversely, velocity measurement performed by the computer vision functions work well only when user movements do not include significant rotations. However, rotations can be detected by the smartphone IMU, which also provides an alternative velocity measurements, and therefore the opportunistic combination of all the measurements shows a significant improvement in the tracking system accuracy.

We validated our proposed data fusion scheme in real experiments carried out in a lab equipped with the OptiTrack system, in order to compare the coordinates estimated by the smartphone with a ground-truth reference signal. Our experiments confirmed that using camera-based measurements allow to reduce the PDR error drifts and that the system usability is satisfactory for both blind and low vision users.

ACKNOWLEDGMENT

The authors acknowledge In.Sight s.r.l. (https://in.sight.srl) for developing the localization system (patented [5]).

REFERENCES

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

[2] R. Velázquez, "Wearable assistive devices for the blind," in *Wearable and Autonomous Biomedical Devices and Systems for Smart Environment* (Lecture Notes in Electrical Engineering). Berlin, Germany: Springer, 2010, pp. 331–349, ch. 17.

[3] D. Croce, P. Gallo, D. Garlisi, L. Giarré, S. Mangione, and I. Tinnirello, "ARIANNA: A smartphone-based navigation system with human in the loop," in *Proc. 22nd Medit. Conf. Control Automat. (MED)*, 2014, pp. 8–13.

[4] D. Croce, L. Giarré, F. G. La Rosa, E. Montana, and I. Tinnirello, "Enhancing tracking performance in a smartphone-based navigation system for visually impaired people," in *Proc. 24th Medit. Conf. Control Automat. (MED)*, 2016, pp. 1355–1360.

[5] D. Croce, D. Garlisi, L. Giarre, I. Tinnirello, and P. Gallo, "Sistema di navigazione per non vedenti," Italian Patent 10 2014 902 314 431 (BG2014A000054), Dec. 2014.

[6] J. L. Barron, D. J. Fleet, and S. S. Beauchemin, "Performance of optical flow techniques," *Int. J. Comput. Vis.*, vol. 12, no. 1, pp. 43–77, 1994.

[7] A. R. Jiménez, F. Seco, J. C. Prieto, and J. Guevara, "Indoor pedestrian navigation using an INS/EKF framework for yaw drift reduction and a foot-mounted IMU," in *Proc. Workshop Positioning Navigat. Commun.*, Dresden, Germany, 2010, pp. 135–143.

[8] L. Filardo, F. Inderst, and F. Pascucci, "C-IPS: A smartphone based indoor positioning system," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat. (IPIN)*, Alcala de Henares, Spain, Oct. 2016, pp. 1–7.

[9] A. R. Jiménez, F. Seco, F. Zampella, J. C. Prieto, and J. Guevara, "Improved Heuristic Drift Elimination (iHDE) for pedestrian navigation in complex buildings," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, Guimaraes, Portugal, 2011, pp. 1–8.

[10] F. Montorsi, F. Pancaldi, and G. M. Vitetta, "Design and implementation of an inertial navigation system for pedestrians based on a low-cost MEMS IMU," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Budapest, Hungary, Jun. 2013, pp. 57–61.

[11] K. Kaemarungsi and P. Krishnamurthy, "Properties of indoor received signal strength for WLAN location fingerprinting," in *Proc. 1st Annu. Int. (MobiQuitous)*, 2004, pp. 14–23.

[12] A. M. Ladd, K. E. Bekris, A. P. Rudys, D. S. Wallach, and L. E. Kavraki, "On the feasibility of using wireless Ethernet for indoor localization," *IEEE Trans. Robot. Autom.*, vol. 20, no. 3, pp. 555–559, Jun. 2004.

[13] S. Willis and S. Helal, "RFID information grid and wearable computing solution to the problem of wayfinding for the blind user in a campus environment," in *Proc. IEEE Int. Symp. Wearable Comput.*, Oct. 2005, pp. 1–8.

[14] N. Fallah, I. Apostolopoulos, K. Bekris, and E. Folmer, "Indoor human navigation systems: A survey," *Interacting Comput.*, vol. 25, no. 1, pp. 21–33, Jan. 2013.

[15] N. Fallah, I. Apostolopoulos, K. Bekris, and E. Folmer, "The user as a sensor: Navigating users with visual impairments in indoor spaces using tactile landmarks," in *Proc. ACM Annu. Conf. Human Factors Comput. Syst. (CHI)*, Austin, TX, USA, May 2012, pp. 425–432.

[16] C. Ton, A. Omar, V. Szedenko, V. H. Tran, A. Aftab, F. Perla, M. J. Bernstein, and Y. Yang, "LIDAR assist spatial sensing for the visually impaired and performance analysis," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 9, pp. 1727–1734, Sep. 2018, doi: 10.1109/TNSRE.2018.2859800.

[17] R. Kessler, M. Bach, and S. P. Heinrich, "Two-tactor vibrotactile navigation information for the blind: Directional resolution and intuitive interpretation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 3, pp. 279–286, Mar. 2017, doi: 10.1109/TNSRE.2016.2569258.

[18] R. K. Katzschmann, B. Araki, and D. Rus, "Safe local navigation for visually impaired users with a time-of-flight and haptic feedback device," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 3, pp. 583–593, Mar. 2018, doi: 10.1109/TNSRE.2018.2800665.

[19] H.-C. Wang, R. K. Katzschmann, S. Teng, B. Araki, L. Giarré, and D. Rus, "Enabling independent navigation for visually impaired people through a wearable vision-based feedback system," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2017, pp. 6533–6540.

[20] W. C. S. S. Simões and V. F. de Lucena, "Hybrid Indoor Navigation as sistant for visually impaired people based on fusion of proximity method and pattern recognition algorithm," in *Proc. IEEE 6th Int. Conf. Consum. Electron.-Berlin (ICCE-Berlin)*, Berlin, Germany, Sep. 2016, pp. 108–111, doi: 10.1109/ICCE-Berlin.2016.7684732.

[21] R. Tapu, B. Mocanu, and T. Zaharia, "A computer vision system that ensure the autonomous navigation of blind people," in *Proc. E-Health Bioeng. Conf. (EHB)*, Iasi, Romania, 2013, pp. 1–4, doi: 10.1109/EHB.2013.6707267.

[22] N. A. Giudice, "Navigating without vision: Principles of blind spatial cognition," in *Handbook of Behavioral and Cognitive Geography*. Cheltenham, U.K.: Edward Elgar, 2018, ch. 15, doi: 10.4337/9781784717544.

[23] R. Manduchi and S. Kurniawan, "Mobility-related accidents experienced by people with visual impairment," *AER J., Res. Pract. Vis. Impairment Blindness*, vol. 4, no. 2, pp. 44–54, 2011.

[24] A. Riazi, F. Riazi, R. Yoosfi, and F. Bahmeei, "Outdoor difculties experienced by a group of visually impaired Iranian people," *J. Current Ophthalmol.*, vol. 28, no. 2, pp. 85–90, 2016, doi: 10.1016/j.joco.2016.04.002.

[25] J. Guerreiro, D. Ahmetovic, D. Sato, K. Kitani, and C. Asakawa, "Airport accessibility and navigation assistance for people with visual impairments," in *Proc. CHI*, Glasgow, U.K., 2019.

[26] L. Giarré, I. Tinnirello, and M. L. Jaccheri, "In.Line: A navigation game for visually impaired people," in *Proc. Int. Conf. Entertainment Comput.*, in Lecture Notes in Computer Science, vol. 10507. Cham, Switzerland: Springer, 2017, pp. 147–153.

[27] M. Pielot and R. de Oliveira, "Peripheral vibro-tactile displays," in *Proc. MobileHCI Tactile User Interfaces*, 2013, pp. 1–10, doi: 10.1145/2493190.2493197.

[28] M. Pielot. *How the Phone's Vibration Alarm can help to Save Battery*. Accessed: May 12, 2014. [Online]. Available: http://pielot.org/2012/12/11/how-the-phones-vibration-alarm-can-help-to-save-battery/

[29] L. Faramondi, F. Inderst, F. Pascucci, R. Setola, and U. Delprato, "An enhanced indoor positioning system for first responders," in *Proc. Int. Conf. Indoor Positioning Indoor Navigat.*, 2013, pp. 1–8.

[30] T.-K. Chuang, N.-C. Lin, J.-S. Chen, C.-H. Hung, Y.-W. Huang, C. Tengl, H. Huang, L.-F. Yu, L. Giarre, and H.-C. Wang, "Deep trail-following robotic guide dog in pedestrian environments for people who are blind and visually impaired—Learning from virtual and real worlds," in *Proc. IEEE Int. Conf. Robot. Automat. (ICRA)*, Piscataway, NJ, USA, May 2018, pp. 1–7.

[31] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, "Caffe: Convolutional architecture for fast feature embedding," in *Proc. ACM Int. Conf. Multimedia*, 2014, pp. 675–678.

[32] A. Giusti, J. Guzzi, D. C. Ciresan, F.-L. He, J. P. Rodriguez, F. Fontana, M. Faessler, C. Forster, J. Schmidhuber, and G. Di Caro, "A machine learning approach to visual perception of forest trails for mobile robots," *IEEE Robot. Autom. Lett.*, vol. 1, no. 2, pp. 661–667, Jul. 2016.

[33] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Boston, MA, USA, Jun. 2015, pp. 5353–5360.

[34] F. De Cillis, L. Faramondi, F. Inderst, S. Marsella, M. Marzoli, F. Pascucci, and S. Setola, "Hybrid indoor positioning system for first responders," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published.

[35] *Using the ADXL202 in Pedometer and Personal Navigation Applications*, Analog Devices, Norwood, MA, USA, 2002.

[36] G. Galioto, I. Tinnirello, D. Croce, F. Inderst, F. Pascucci, and L. Giarré, "Sensor fusion localization and navigation for visually impaired people," in *Proc. MobiCom*, 2017, pp. 3191–3196.

[37] G. Galioto, I. Tinnirello, D. Croce, F. Inderst, F. Pascucci, L. Giarré, "Sensor fusion localization and navigation for visually impaired people," in *Proc. ECC*, 2018, pp. 3191–3196.

DANIELE CROCE received the joint Ph.D. degree from Politecnico di Torino, Turin, Italy, and the Università de Nice-Sophia Antipolis, UNSA, Nice, France, and the double M.Sc. degree in networking engineering from the Politecnico di Torino and the EURECOM Institute, Sophia Antipolis, France, in 2006, and the Research Master diploma (ex DEA) degree in networking and distributed systems from UNSA, France. He currently owns a research position at the University of Rome La Sapienza. He has long experience of research collaborations, in several European and national research projects, on Wireless networks, the Internet of Things, high-quality TV streaming, Smart Grid communications, and smart cities. He also worked on assistive technologies for visually impaired people and with the Arianna project he Co-founded the start-up company In.sight s.r.l., spin-off of Palermo University, Italy.

LAURA GIARRÈ (SM'99) received the Laurea degree *(summa cum laude)* in electronic engineering from the University of Florence, in 1986, and the Ph.D. degree in system engineering from the University of Bologna, Italy, in 1992. In 1987, she was a Researcher with the Research Laboratory of the Motorways Public Company. From 1988 to 1993, she was with the Department of Systems and Computer Science, University of Florence, Italy. From 1993 to 1998, she was an Assistant Professor with the Department of Computer and Control Engineering, Turin Polytechnic, Torino, Italy. From 1998 to 2016, she was with the University of Palermo, Italy, as an Associate Professor, and since 2016, she joined at the Department of Engineering (Enzo Ferrari), University of Modena and Reggio Emilia, as a Full professor. She spent several periods (Fall 1996, Fall 1997, Winter/Spring 1999, Winter and Fall 2006) as a Visiting Professor at CCDC, University of California at Santa Barbara and at the Laboratory for Information and Decision Systems, MIT, Boston. In winter/spring 2015, she was a Visiting Scientist with the Computer Science and Artificial Intelligence Laboratory of MIT (Cambridge). In 2018, she was a Visiting Professor at the Johan Kepler University of Linz (Austria). She is currently a Control System Professor. Her research interests include identification, control of networked systems, and assistive technology, focusing her attention on localization, navigation, and obstacle avoidance for assisting blinds and low vision people.

FEDERICA PASCUCCI (M'06) received the M.S. degree in computer science and automation engineering from the University of Roma Tre, Rome, Italy, in 2000, and the Ph.D. degree in system engineering from the University of Rome La Sapienza, Rome, in 2004. Since 2005, she has been an Assistant Professor with the Department of Engineering, University of Roma Tre. She was a Visiting Scholar with the University of Örebro, Örebro, Sweden, in 2003, and the University of Cyprus, Aglantzia, Nicosia, in 2013. She is the Principal Investigator in several EU funded projects. She has published over 80 journal and conference papers. Her current research interests include wireless sensor networks, indoor localization, cyber-physical systems, industrial control systems, and critical infrastructure protection. She was a recipient of the three best conference paper awards.

ILENIA TINNIRELLO received the Ph.D. degree in telecommunications engineering from the University of Palermo, in 2004. She is currently an Associate Professor with the University of Palermo. She has also been a Visiting Researcher with Seoul National University, South Korea, since 2004, and with Nanyang Technological University, Singapore, since 2006. Her research activities have been focused on wireless networks, in particular on the design and prototyping of protocols and architectures for emerging reconfigurable wireless networks. She is also working on the definition of novel services (smart grid, smart metering, and indoor localization) enabled by the pervasive availability of ICT technologies. She has been involved in several European research projects, including the FP7 FLAVIA project in the role of Technical coordinator, and the H2020 WiSHFUL and Flex5Gware projects.

GIOVANNI ETTORE GALIOTO received the M.Sc. degree in electronic engineering from the University of Palermo, Italy, in 2014, where he is currently pursuing the Ph.D. degree in information and communication technologies. His research activities are focused on innovative hardware and software technologies, applied with particular reference to the Underwater Communication field. He also worked in the ARIANNA project dealing with computer vision algorithms and sensors data acquisitions from smart devices.

DOMENICO GARLISI received the Ph.D. degree in electronic and telecommunication engineering from the University of Palermo, in 2014. He was a Visiting Researcher with the Department of Computer Science, University of California at Los Angeles, in 2013, working on wireless protocol optimizations for vehicular networks. He is currently a CNIT Researcher with the Palermo Research Unit. His main research interests are related to wireless networks, software defined radio, sensor networks, vehicular networks, and wireless systems.

ALICE LO VALVO received the Ph.D. degree in information and communication technologies (ICT) from the University of Palermo, in February 2019. She is currently a Postdoctoral Researcher with the University of Palermo, working on augmented reality project. With a background on electronics, she worked on a research project about innovative receiver architectures dealing with spectrum agility and programmable MAC/PHY protocols. In particular, she has extensively worked on solutions for supporting PHY layer flexibility, innovative signaling mechanisms based on tone transmissions, and multiband identification of 802.11 preambles transmitted in unknown sub-channels. She has been a Visiting Researcher at armasuisse, Thun, Switzerland, in 2018, working on physical layer security.