scientific reports

Check for updates

OPEN

Enhancing real-time heading estimation for pedestrian navigation via deep learning and smartphone embedded sensors

Junhua Ye[1], Ahmed Mansour[2,3] & Fenghua Huang[4]✉

The accurate smartphone-based pedestrian navigation significantly depends on the precise heading estimation. However, heading estimation is still a challenging problem in most pedestrian navigation applications because of the bias of low-cost smartphone sensors, thermal drift with long-term operation, and the unexpected changes in the carrying mode of handheld devices. To address these challenges, many existing methods based on pervasive resources encounter severe errors. Conversely, auxiliary resources-based approaches may hinder ubiquitous and seamless indoor-outdoor navigation experiences. This research aims to enhance heading estimation by leveraging pervasive measurements such as LVGOs and straight-line features self-recognized from camera images. The proposed method mitigates the accumulated gyro drift using the absolute heading angle estimated by LVGOs. However, these absolute angles are highly prone to false estimation while navigating near areas with high electric and magnetic activities due to stable geomagnetism anomalies. Encouraged by the pervasiveness of straight-line features in indoor and outdoor environments, we developed a deep learning method-based visual tracking of these features to enhance the gyroscope and magnetic field fusion-based heading estimation. A convolutional neural network was developed using a U-Net network to accurately and quickly recognize these features, then leverage them as heading constraint to overcome long-term gyro drift and short-term compass heading bias. The proposed method superiorly ensured the balance between recognition time delay and precision, which enabled smooth real-time performance. The achieved results improved the heading estimation and could provide significant help, especially for visually impaired people, as they mostly track tactile paving. This encourages future tests and assessments using visually impaired people to reliably include the proposed method in their applications.

Currently, smartphones have become integral to our daily routines because they offer convenience and versatility for a wide range of functions and services. Among these services, navigation and positioning services stand out in high demand due to the constant need for location-based information and wayfinding. However, locating a mobile user anytime and anywhere is still a demanding task because GNSS signals are easily corrupted or unavailable in urban canyons and indoor environments[1]. It is known that outside, especially in an open-sky environment, professional-grade GNSS receivers can obtain high-precision positioning locations by RTK; however, consumer-grade receivers, such as smartphones, have difficulty obtaining centimeter-level positioning stability[2]. In addition, in some complex scenarios, such as closed buildings, we still lack mass technical means to achieve accurate, stable indoor positioning[3,4]. To date, some studies or products have been developed in smart-scale indoor positioning, such as wireless signal positioning (UWB, BLE), inertial sensor positioning, pedestrian dead reckoning, matching positioning (map matching or fingerprint matching), vision positioning(VSLAM etc.), and visible light positioning[5–8]. Pedestrian navigation also needs to solve seamless positioning issues, especially in harsh environments[9,10]. In the pedestrian navigation procedure, heading is a core issue that determines the precision and stability of the whole system. Therefore, ensuring reliable heading is critical to positioning[11–14].

To date, several studies have been performed to improve heading precision, and smartphone or other mobile device-embedded IMU measurements are most commonly used for heading estimation. Some strategies focus on

[1]Zhejiang Agriculture and Forestry University, Hangzhou 310000, China. [2]Public Works Department, Faculty of Engineering, Cairo University, Giza, Egypt. [3]The Hong Kong Polytechnic University, Hong Kong, China. [4]Fujian Key Laboratory of Spatial Information Perception and Intelligent Processing, Yango University, Fuzhou 350015, China. ✉email: fhhuang@ygu.edu.cn

using different processing algorithms to improve heading precision. Manos et al. employed a deep network-based framework to estimate walking direction using smartphone sensors[15], Wu et al. proposed a heading estimation method based on an adaptive Kalman filter[16], Wei et al. fused IMU measurements to compute heading with EKF[1]. Guo et al. employed complementary principal component analysis (PCA) and RA methods to estimate the smartphone state, avoiding the 180° directional ambiguity and reducing the impacts of cumulative errors to improve heading precision[17]. One common absolute heading estimation sensor is the magnetometer, which senses the Earth's magnetic field; hence, the true heading angle can be calculated.However, magnetometers are susceptible to magnetic distortions, especially in indoor environments. Ilyas et al. proposed an algorithm called MAD, which was designed by incorporating only healthy magnetometer data in the EKF updating step to reduce drift in zero-velocity updated INS[18]. In real-life scenarios, people usually hold smartphones in their hands, but sometimes, smartphones are placed in a pocket or IMUs are mounted on their feet. Therefore, to address these challenges, Deng et al. proposed a novel heading estimation approach based on a rotation matrix and PCA to evaluate pedestrian heading[19]. Usually, the pedestrian holds the terminal; however, sometimes, the terminal is also fixed to other body parts, such as the foot. Abdulrahim et al. proposed a shoe-mounted IMU approach integrated with ZUPT and building heading information in a Kalman filter environment to reduce heading drift[20–22]. Manos et al. estimated gravity direction to compute the heading angle[23]. Asraf et al. proposed PDRNet, a deep learning PDR framework, for user positioning, which includes a smartphone location recognition classification network followed by a change in heading and the distance regression network[24] (Table 1).

However, the above mentioned methods still face some challenges. IMU measurements have large errors, especially for consumption-level devices, pedestrians switch smartphone postures randomly, wireless signals or geomagnetic signals are easily disturbed, and wireless signal-based heading correction algorithms are susceptible to multipath effects resulting in low stability[25,26]. In addition, magnetometer measurements, which are used for heading estimation, can be easily affected by magnetic interference in indoor environments. Other data sources are used for heading estimation. Liew et al. combined the received RSSI to correct pedestrian heading[27]. Some previous studies also tried to use camera images to assist pedestrian navigation, Ruotsalainen et al., demonstrated that the performance of the visual-aiding algorithm in calculating heading change is much better than that of the built-in digital compass, heading change can be detected with the proposed algorithm at about 1 Hz frequency under the PC environment[28]. Elloumi et al., had used smartphone camera images and vanishing points to assist pedestrian navigation[29]. The developed algorithm enables indoor pedestrian localization in two steps: an off-line learning step defines a reference path by selecting key frames along the way using saliency extraction method and computing the camera orientation in these frames. Then, in localization step, an approximate but realistic position of the walker is estimated in real time by comparing the orientation of the camera in the current image and that of reference to assist the pedestrian with navigation guidance[29]. Wang et al., proposed a method which uses a LED carrying a modulation code as a beacon for absolute position and the camera of smartphone to capture/decode the LED image to obtain the absolute posiqtion and periodically correct the position of the PDR positioning, thereby avoiding the accumulation of PDR positioning technology errors[30]. Traditional SLAM algorithms such as ORB-SLAM2 and ORB-SLAM3 are highly optimized. Yet, they still involve heavy feature processing (e.g., detecting and matching hundreds of keypoints per frame) and nonlinear optimization, which tax computational resources. For instance, even ORB-SLAM2, often cited as efficient, cannot sustain real-time rates (greater than 15 Hz) on a modern smartphone CPU without external assistance. In one study, ORB-SLAM2 running on a Galaxy S10 smartphone failed to reach 15 FPS due to limited CPU resources[31].

In conclusion, most algorithms are however often too massive for a smartphone with limited computing power and storage space[28]. Besides, the smartphone heading is not the same as the pedestrian heading in navigation procedure, due to the swinging caused by movement of pedestrian, it causes deviation between them. So based on smartphone measurements, we consider collecting smartphone images to recognize the pedestrian heading in real-time. Therefore, in this paper, we combine smartphone internal embedded sensors, such as gyroscopes, accelerometers, magnetometers, and camera measurements, to develop a heading fusion algorithm to improve heading estimation precision. Specifically, the main contribution of this paper is the development

Method	Principle	Advantages	Disadvantages
Magnetometer	Measures Earth's magnetic field to determine heading relative to magnetic north.	Provides absolute heading. Works indoors/outdoors without interference.	Highly sensitive to magnetic disturbances. Requires frequent calibration. Poor accuracy in urban/indoor areas.
Gyroscope	Integrates angular velocity to estimate relative heading changes.	High short-term accuracy. Immune to magnetic interference.	Accumulates drift over time. Requires initial heading reference. Unsuitable for long-term use.
Sensor Fusion	Combines multiple sensors using algorithms (Kalman Filter, Madgwick).	Balances gyro precision with magnetometer reference. Reduces drift and interference. Works well in dynamic motion.	Complex implementation. Still affected by strong fields. Depends on sensor quality.
GNSS	Estimates heading from direction of movement using position changes.	Absolute heading without interference. Reliable in open areas.	Requires movement. Poor performance indoors. High power consumption.
Computer Vision	Uses camera or AR frameworks (ARCore/ARKit) to estimate orientation.	High accuracy in good conditions. Unaffected by magnetic fields.	High computational cost. Needs good lighting/texture. Fails in motion blur.
Wi-Fi/Bluetooth	Uses signal strength or Angle-of-Arrival from transmitters.	Works indoors where GNSS fails. Can provide absolute heading.	Requires infrastructure. Low accuracy (5°-15° error). Multipath interference.
Hybrid	Combines multiple methods for robustness.	Resilient to sensor failures. Adapts to diverse environments.	Increased complexity. May struggle in extreme cases.

Table 1. Comparison of smartphone heading estimation methods

of a novel method that leverages deep learning algorithms, particularly a U-Net-based convolutional neural network, to accurately estimate absolute heading angles using LVGOs. This approach effectively mitigates long-term gyro drift and short-term compass heading bias, thereby improving the accuracy and reliability of pedestrian navigation systems. This study innovatively designed an image-based real-time heading correction algorithm combined with mobile deep learning technology to achieve rapid pedestrian heading correction based on smartphone, thus improving pedestrian navigation and positioning accuracy. This paper is organized as follows: Sect. 1 mainly introduces the relevant domestic and international research progress of pedestrian navigation and cell phone navigation heading, and analyzes it. Section 2 mainly describes the heading correction method used in this paper, including the design of the U-Net convolutional neural network and how it is used to recognize features and provide heading constraints to mitigate gyro drift and compass bias. Section 3 presents the experimental setup, results, and analysis, demonstrating the effectiveness of the proposed approach. Finally, Sect. 4 discusses the implications of our findings, potential applications, and directions for future research. The last section summarizes the paper.

Materials and methods

This study proposes a deep learning-aided heading estimation framework that fuses inertial sensor data with visual structural cues extracted from smartphone camera images. The core idea is to enhance gyroscope-based heading estimates by incorporating visual constraints obtained from ground reference lines, which are ubiquitous in structured pedestrian environments. The fusion framework leverages a lightweight U-Net-based convolutional neural network (CNN) to detect and segment straight-line features from low-resolution RGB images in real time (see Fig. 1). The U-Net model is trained on a curated dataset of labeled ground reference line images and optimized to balance recognition accuracy with minimal latency. To ensure feasibility on resource-constrained mobile devices, all input images and corresponding labels are downscaled to 64×64 resolution prior to training. The network outputs a binary segmentation mask that highlights dominant ground-aligned linear features, which are then geometrically interpreted to infer visual heading cues. These visual heading estimates are subsequently fused with inertial measurements-namely, gyroscope, accelerometer, and magnetometer readings-via a Kalman filter framework. The fusion process incorporates high-frequency IMU updates and selectively integrates visual estimates when reliable reference lines are detected. By treating the U-Net output as a virtual heading observation, the method effectively suppresses long-term gyroscopic drift and reduces compass-related biases, enabling more robust and precise pedestrian navigation.

Heading estimation using Acc and Mag measurements

Most smartphones integrate MEMS sensors. Smartphones have their own coordinate system that we call the body coordinate system (b). We often use the navigation coordinate system (n) to perform navigation, so we need to use the rotation matrix C_n^b to transform measurements from (b) to (n).

$$C_n^b = C_n^b(Y)C_n^b(X)C_n^b(Z) \tag{1}$$

rotation matrix:

$$C_n^b = \begin{bmatrix} \cos\gamma\cos\psi + \sin\gamma\sin\theta\sin\psi & -\cos\gamma\sin\psi + \sin\gamma\sin\theta\cos\psi & -\sin\gamma\cos\theta \\ \cos\theta\sin\psi & \cos\theta\cos\psi & \sin\theta \\ \sin\gamma\cos\psi - \cos\gamma\sin\theta\sin\psi & -\sin\gamma\sin\psi - \cos\gamma\sin\theta\cos\psi & \cos\gamma\cos\theta \end{bmatrix} \tag{2}$$

Traditionally, we can use the following equations to calculate smartphone heading via accelerometer and magnetometer measurements. Roll γ can be obtained by X- and Z-axis accelerometer measurements ($AccX$ and $AccZ$):

$$\gamma = \arctan 2(-AccX, AccZ) \tag{3}$$

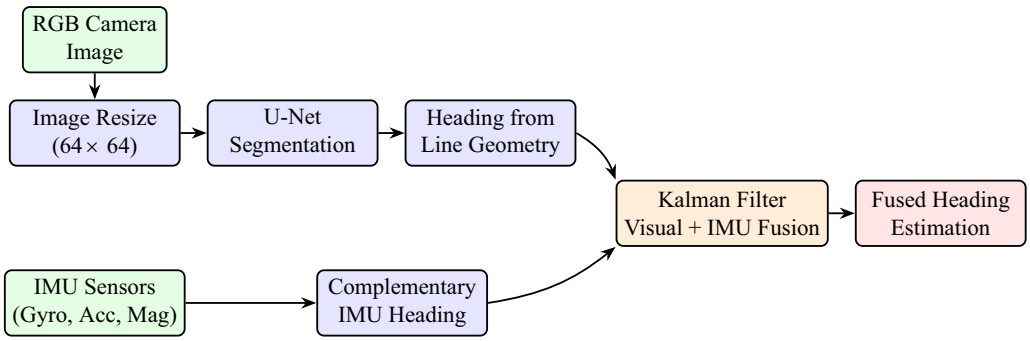

Fig. 1. Proposed visual-inertial heading estimation framework. The method integrates U-Net-based visual heading cues derived from ground reference lines with inertial sensor data. Kalman filtering yields robust fused heading estimates.

Pitch θ can be computed using $X-$, $Y-$ and $Z-$axis accelerometer measurements:

$$\theta = \arctan 2(AccY, \sqrt{AccX^2 + AccZ^2}) \tag{4}$$

Magnetic north ψ_m and true north ψ_{north} can be obtained by combining γ, θ and magnetometer measurements:

$$\psi_m = \arctan(-\frac{M_y^b cos(\gamma) - M_z^b sin(\gamma)}{M_x^b cos(\theta) + M_y^b sin(\theta) sin(\gamma) + M_z^b sin(\theta) cos(\gamma)}) \tag{5}$$

where M_X^b, M_Y^b and M_Z^b denote the *X*-, *Y*- and *Z*-axis values in the body coordinate system of the magnetometer; $\Delta\psi$ represents 'magnetic declination', which can be obtained from the International Geomagnetic Reference Field (IGRF) model. Formulas (1)--(4) describe how the smartphone internal MEMS sensor measurements can be processed to obtain the heading. The heading has a large bias compared with the true pedestrian direction due to sensor error, environment interference and orientation bias of pedestrians and smartphones[32,33].

$$\psi_{TrueNorth} = \psi_m + \Delta\psi - \gamma \tag{6}$$

ψ_m is the true north local magnetic declination, and γ is the true north grid convergence.

Heading fusion combining Acc, Mag, and Gyro measurements

From the section A, we know that smartphone heading can be computed by Acc and Mag measurements, and it can also be estimated by combining the gyroscope integration angle and initial orientation.

$$Heading_{Now} = Heading_{Initial} + \int_{T_k}^{T_{k+1}} \omega_i dt \tag{7}$$

In the above formula, ω_i is the angle update rate of the gyroscope output, and T_k and T_{k+1} are the start epoch and end epoch, respectively. $Heading_{Initial}$ is the initial heading, which can be obtained from the initial estimation result, and it is vital to heading accuracy.The unit of $Heading_{Now}$ and $Heading_{Initial}$ is radian. Actually, heading precision degrades when integration lasts due to sensor errors. Many heading fusion algorithms combining Acc, Mag and Gyro measurements have been proposed in previous studies; in this study, we employ a complementary filter to compute smartphone heading. Formula 8 presents the fusion algorithm

$$Heading_{CF}^i = \beta * (Heading_{CF}^{i-1} + \omega_i * \Delta_T) + Heading_{AccMag}^i (1 - \alpha) \tag{8}$$

$Heading_{AccMag}^i$ is the heading calculated by the Acc and Mag measurements, $Heading_{CF}^{i-1}$ is the fusion heading in the last epoch, β is the smoothing factor, ω_i is the gyroscope output angle change rate, and Δ_T is the sampling period, obtained from start epoch T_{k-1} and end epoch T_k.The unit of $Heading_{AccMag}$ and $Heading_{CF}$ is radian.

$$\beta = \frac{\Delta_T}{RC + \Delta_T} \tag{9}$$

$$RC = \frac{1}{2\pi f_c} \tag{10}$$

in the above formula, the expression *RC* denotes the equivalent time constant, and f_c is the cutoff frequency.

Heading recognition combining camera image and deep learning

Pedestrian heading computation with ground reference line

Figure 2a is the smartphone coordinate system, and we find that the Y-axis of the smartphone coordinate system is parallel to the smartphone screen, in fact, for the image captured by camera, its image coordinate axes (*cX*,*cY*,*cZ* in Fig. 2b) point in the same direction as smartphone coordinate system, the difference is the origin of the coordinates. If we project the smartphone screen plane to local ground, combining the reference line *AB* and intersection point *A*, we can create a local ground coordinate system. Figure 2c indicates this coordinate system; actually, it is a special case of the world geodetic coordinate system when $Zw = 0$. From Fig. 2c , we know α is the direction bias of the smartphone Y-axis and reference line in the local ground plane. The azimuth of reference line AB can be denoted as:

$$\alpha = arctan2((\Delta X_w^{AB})/(\Delta Y_w^{AB})) \tag{11}$$

$$\alpha = \begin{cases} \alpha & \Delta Y_w^{AB} > 0; \Delta X_w^{AB} > 0 \\ \alpha + 180 & \Delta Y_w^{AB} < 0; \Delta X_w^{AB} > 0 \\ \alpha + 180 & \Delta Y_w^{AB} < 0; \Delta X_w^{AB} < 0 \\ \alpha + 360 & \Delta Y_w^{AB} > 0; \Delta X_w^{AB} < 0 \end{cases} \tag{12}$$

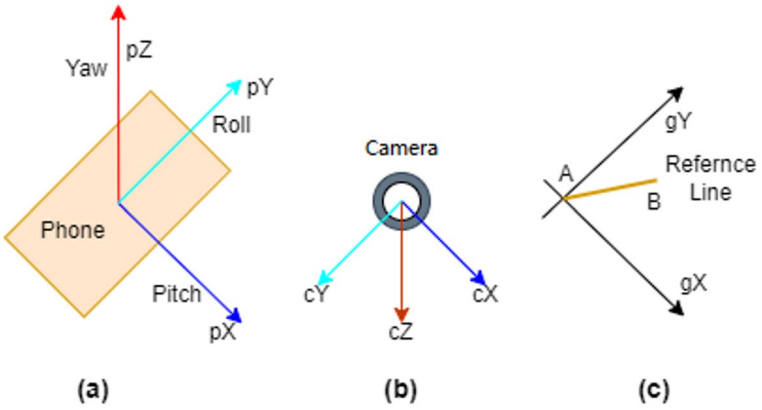

Fig. 2. Graphic (**a**) denotes the smartphone coordinate system, (**b**) is the camera coordinate system, and (**c**) is the ground coordinate system.

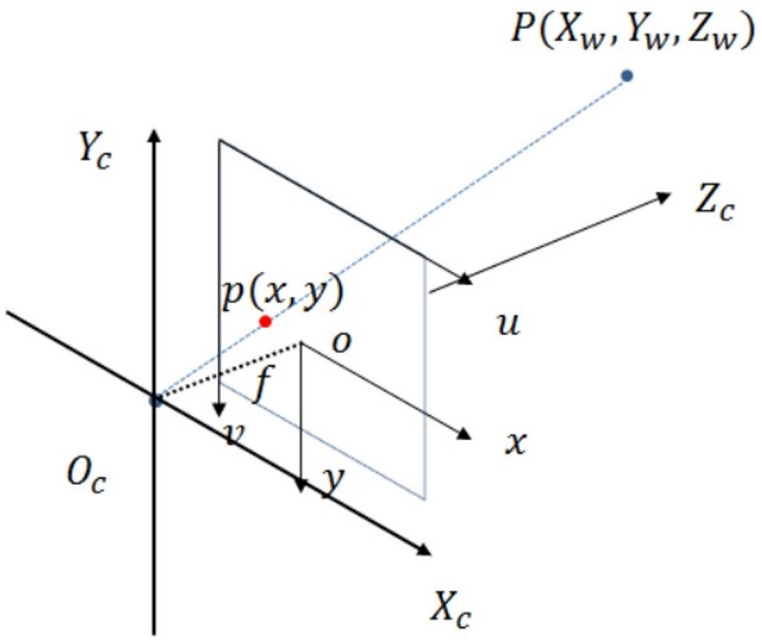

Fig. 3. Projection of point P in the world coordinate system and the image plane coordinate system.

When the pedestrian is in motion, the smartphone camera can obtain the reference line in real time. Therefore, how can this angle deviation be calculated in real time? What is the change in this angle deviation in the phone pixel coordinate system?

From the definition, we know that the original points of the smartphone body coordinate system and camera coordinate system are in different locations: one is the smartphone body center, the other is in the camera lighting center, pX and cX have the same directions, $cY(+)$, $cZ(+)$ and $pY(+)$, $pZ(+)$ are in opposite directions, which means rotating the smartphone coordinate system 180deg along pX and transforming the original point bias$(\delta x, \delta y, \delta z,)$. Then, we can obtain the camera coordinate system. Actually, if there are only rotation and coordinate origin translations, the angular bias cannot be changed, rolling and pitching during the phone's attitude transformation does not affect this relationship either, and is also independent of external translations. In Fig. 3, $X_cO_cY_c$ denotes the camera coordinate system, $u - v$ is the pixel coordinate system, and xoy is the image coordinate system. $P(X_w, Y_w, Z_w)$ is the coordinate of point P in the world geodetic coordinate system, and $p(x, y)$ is the corresponding point on the image. What is the change in the angle deviation in the pixel coordinate system? $O_w - X_wY_wZ_w$ is the world geodetic coordinate system in m, $O_c - X_cY_cZ_c$ is the camera coordinate system in m, $O - XY$ is the graphic coordinate system in mm, $u - v$ is the pixel coordinate system in pixels, P is the true point in the world geodetic coordinate system, p is the point in the graphic coordinate system in (x, y), and the pixel coordinate is (u, v).

Figure 4 shows the image and pixel coordinate systems. We know that the image coordinate system and pixel coordinate system are both plane coordinate systems, and the coordinate axes point in the same direction.

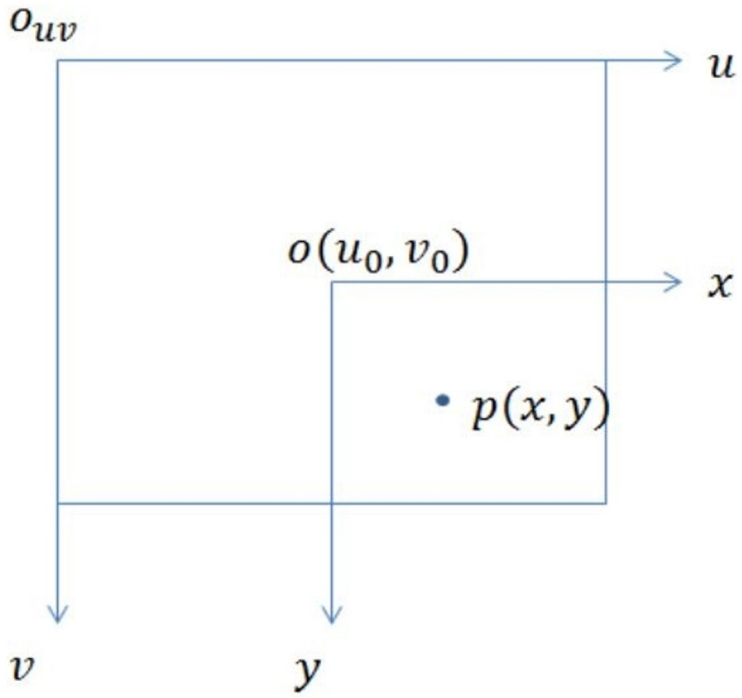

Fig. 4. The pixel coordinate system and the image plane coordinate system.

Therefore, transformation will not produce any distortion. Both the pixel and image coordinate systems in Fig. 4 are planar coordinate systems with the same scale, the same axes and different points of origin. Therefore, we can obtain the transformation formula between the two coordinate systems: This is the example 2 of equation:

$$Z_c \begin{bmatrix} u_p \\ v_p \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ \overrightarrow{0} & 1 \end{bmatrix} \begin{bmatrix} X_w^P \\ Y_w^P \\ Z_w^P \\ 1 \end{bmatrix} \tag{13}$$

R and T are the rotation matrix and origin translation matrix, respectively, between the world geodetic coordinate system and camera coordinate system; dx and dy are the row and column distances of one pixel, respectively, and their units are *mm*; f is the camera focal length; and Z_c is a coefficient. The end points A and B of the reference line in Fig. 2 can be denoted as:

$$Z_c \begin{bmatrix} u_{A/B} \\ v_{A/B} \\ 1 \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 & u_0 \\ 0 & \frac{1}{dy} & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} R & T \\ \overrightarrow{0} & 1 \end{bmatrix} \begin{bmatrix} X_w^{A/B} \\ Y_w^{A/B} \\ 0 \\ 1 \end{bmatrix} \tag{14}$$

We know that rotation and origin translation cannot change the reference line orientation. Combining equation (14), without considering the rotation and translation of the coordinate system, we calculate the difference in the coordinate components of the two points of AB, and then we can obtain equation (15)

$$Z_c \begin{bmatrix} \Delta u_{A/B} \\ \Delta v_{A/B} \end{bmatrix} = \begin{bmatrix} \frac{1}{dx} & 0 \\ 0 & \frac{1}{dy} \end{bmatrix} \begin{bmatrix} f & 0 \\ 0 & f \end{bmatrix} \begin{bmatrix} \Delta X_w^{A/B} \\ \Delta Y_w^{A/B} \end{bmatrix} \tag{15}$$

The azimuth of reference line AB can be denoted as:

$$\begin{aligned} \alpha &= arctan2((\Delta X_w^{AB})/(\Delta Y_w^{AB})) * 180/(\pi) \\ &= arctan2((\Delta u^{AB})/(\Delta v^{AB})) * 180/(\pi) \end{aligned} \tag{16}$$

If smartphones can capture the reference line in real time, the angular bias α can be computed in theory, which will help correct pedestrian heading errors.

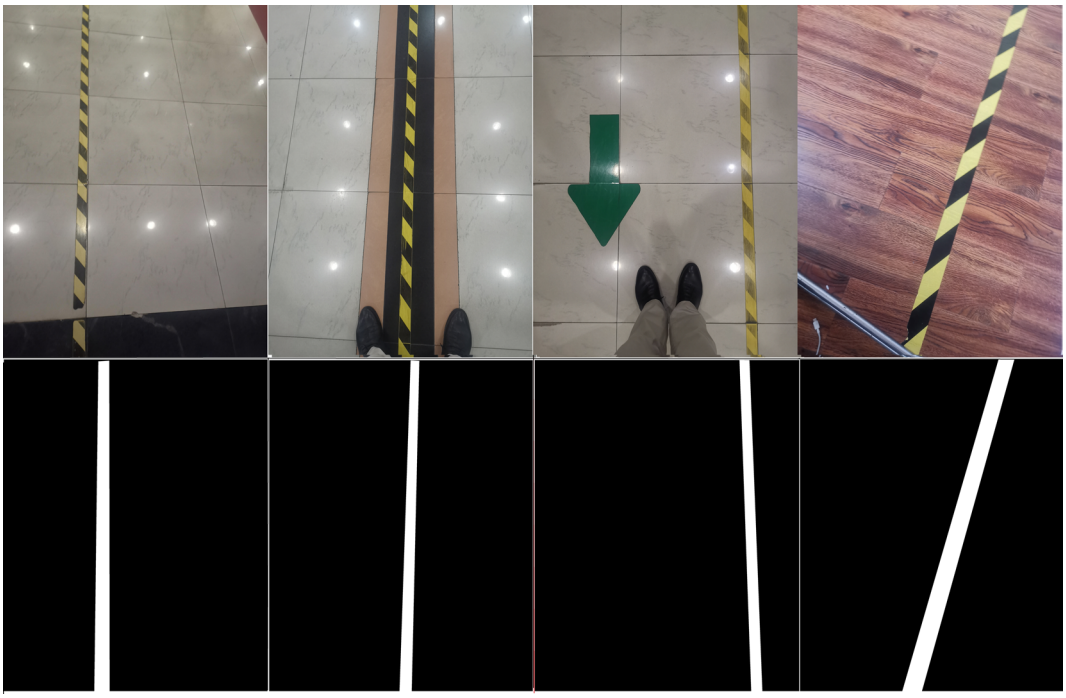

Fig. 5. Captured raw images and related labels.

Size	Speed	Precision
512*512	Slow	Higher
256*256	Slow	Higher
128*128	Litte slow	High
64*64	Real time	Normal
32*32	Rapid	Low

Table 2. Recognition speed and accuracy of different sizes of input pictures

Reference line real-time recognition based on smartphone camera images and deep learning

In this study, to recognize the ground reference lines in real time, we employ smartphone-embedded cameras to capture images. Based on smartphone-captured real-time images, computer vision technology and deep learning methods were used to recognize ground reference lines in real time. In the first step, we installed special textured tape on the floor as reference lines at the experimental site (Fig. 5), which is very common, inexpensive and can be widely used.

Over 1,000 ground reference line photographys were collected and verified through testing to find that these test data were adequate. In our validation experiments, the reference line recognition precision exceeds 92%. We can also use augmentation algorithms, such as AutoAug and RandAug, to expand the dataset. The training dataset contains two parts, raw pictures and label files. Before training, in the first step, we need to create labels for these training pictures. We use LabelMe to label reference lines and convert labels to JSON files. Finally, these labels are transformed to single-channel PNGs. The size of the raw RGB picture captured by a smartphone camera, such as a Huawei Mate20 (the camera is an HMA-AL00), has a width of up to 2976 pixels, a height of 3, 968 pixels, and horizontal and vertical resolutions of 96 dpi. Obviously, these pictures are too large. Considering that pedestrian navigation needs to balance real-time and accuracy, the processing performance of smartphone chips is very limited, and the amount of pixels of inputted image directly affects the processing speed, which ultimately leads to smartphone can't achieve real-time or near-real-time output recognition results. Therefore, in this study, different sizes of compression processing strategies were carried out for camera raw images and labelled images. Finally, real-time experiments were carried out for these models. Table2 shows processing speed and recognition accuracy results for different sizes of inputted images. From Table 2, it can be seen that the image of 64 pixels*64 pixels is able to balance the speed and accuracy, so this size of image was finally adopted for model training and real-time recognition experiments.

In this study, considering the balance of recognition time delay and precision, all training JPG pictures and label images are resized to $64 * 64$. We divide the training, testing and validation datasets at a ratio of 8:1:1. Once the preparations are complete, we start to train the model. In this study, the only goal is to detect the

ground reference line of the image. Other parts of the image can be set as background, which is a typical binary classification problem. Therefore, UNet (Fig. 6) was used to train the recognition model.

UNet was originally used for segmenting medical images, and UNet's design combines classical codec structures, convolution, pooling, and ReLU activation functions. In addition, padding is not used to prevent position shifts; upsampling using +conv interpolation instead of upsampling algorithms, such as transpose convolution, allows the feature map to maintain uniform chromatic aberrations and skip connections to maintain better information fusion[34].

In this study, the training dataset contains more than 800 pictures, and we set the number of training epochs to 200.

The validation results (Fig. 7) show that the designed network can extract the target well. Therefore, how can we combine the recognition model with mobile phone images for real-time ground reference line recognition? The primary issue is to make the trained model lightweight so it can run on mobile phones in real time. The following process (Fig. 8) is used in this study.

In this study, we convert the trained *.H5* model to the *.tflite* model, which can run on the Huawei Mate20 smartphone. The size of the real-time image captured by the smartphone camera is $1080 * 1920$, while the size of the input tensor is $64 * 64$, so it is necessary to resize the image. The output image contains the predicted label, that is, the reference line, and the following screenshots are real-time prediction results obtained from our application "PedestrianNav".

In this study, experiments were carried out for this recognition model in various scenes with different backgrounds, different light and dark levels and different smartphone poses. Figure 9a shows the phone parallel to the ground reference line while the pedestrian is walking. Figure 9b and c show the real-time recognition results with the phone in a tilted pose against different backgrounds. Figure 9d and e show the recognition results with the ground reference line at night when the light is poor. Overall, the recognition accuracy meets the requirements, and the time delay is short and nearly real-time.

Visual heading fusion with IMU-based estimation

Following the initial computation of heading using accelerometer and magnetometer measurements and the refinement of short-term heading updates via gyroscope integration, the proposed system further enhances heading accuracy by fusing inertial and visual cues using a state-space estimation framework. Specifically, we adopt a Kalman filter formulation to correct heading drift and incorporate visual measurements when available. The fusion process maintains a recursive estimate of the pedestrian's heading state by continuously predicting and correcting the heading based on sensor observations.

We define the system state vector at time step *k* as:

$$\mathbf{x}_k = \begin{bmatrix} h_k \\ \bar{\omega}_k \end{bmatrix} \tag{17}$$

where h_k represents the estimated heading at time *k*, and $\bar{\omega}_k$ denotes the mean angular rate of heading rotation. The system state evolves according to a linear process model governed by the following discrete-time update:

$$\begin{aligned} h_{k+1} &= h_k + \bar{\omega}_k \Delta t + \delta_h \\ \bar{\omega}_{k+1} &= \bar{\omega}_k + \delta_\omega \end{aligned} \tag{18}$$

In this formulation, Δt is the sampling interval between two consecutive measurements, and δ_h, δ_ω are zero-mean Gaussian noise terms modeling uncertainty in the heading and angular rate updates, respectively. These noise terms capture stochastic effects due to pedestrian dynamics, sensor quantization, and real-world physical

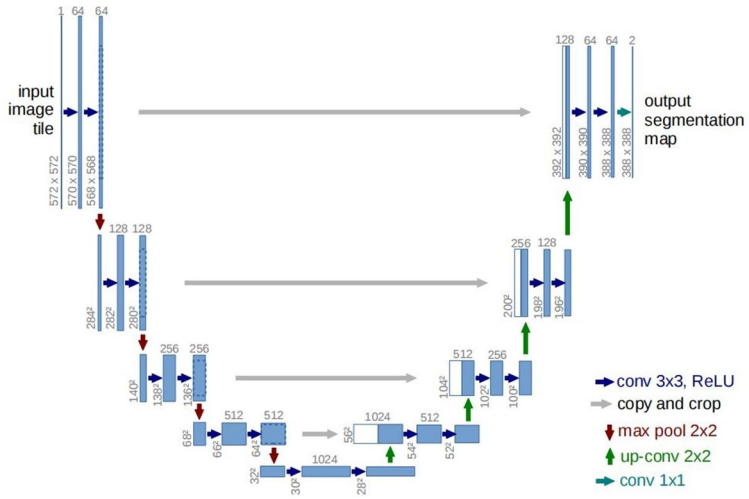

Fig. 6. UNet network architecture.

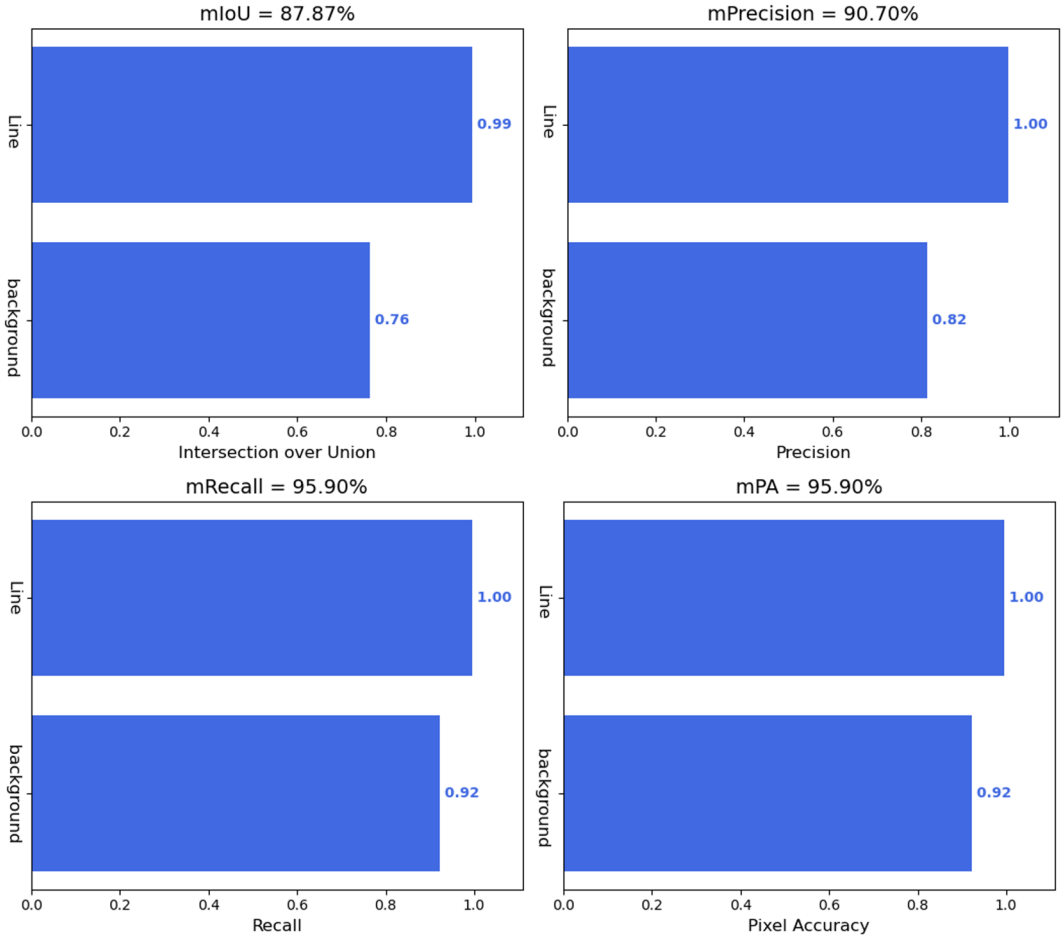

Fig. 7. Statistical results of validation.

variability. They are assumed to follow white noise distributions consistent with typical inertial measurement errors.

At each step, the system receives external measurements to update the state. Two types of heading observations are considered:

- High-frequency heading updates derived from gyroscope-integrated inertial data, denoted h_{imu}.
- Low-frequency but absolute heading estimates derived from visual reference line detection, denoted h_{visual}.

The gyroscope-based heading estimate h_{imu} is computed via a complementary filter as described previously. Although this estimate accumulates drift over time, it provides smooth and frequent updates, making it suitable for continuous state propagation. When this estimate is used to update the Kalman filter, the observation vector is defined as:

$$\mathbf{z}_k = \begin{bmatrix} h_{\text{imu}} \\ \frac{1}{2}(\omega_k + \omega_{k+1}) \end{bmatrix} \tag{19}$$

Here, the second term represents the average angular velocity during the interval, approximating the continuous angular rate over the step period. These measurements allow the Kalman filter to track and correct the system state using high-frequency inertial cues.

In contrast, when the smartphone camera reliably detects a ground reference line, a visual heading measurement h_{visual} becomes available. This estimate, while updated less frequently, is not subject to cumulative drift and provides an accurate correction for the heading. When such a measurement is obtained, the observation vector becomes:

$$\mathbf{z}_k = \begin{bmatrix} h_{\text{visual}} \\ \frac{1}{2}(\omega_k + \omega_{k+1}) \end{bmatrix} \tag{20}$$

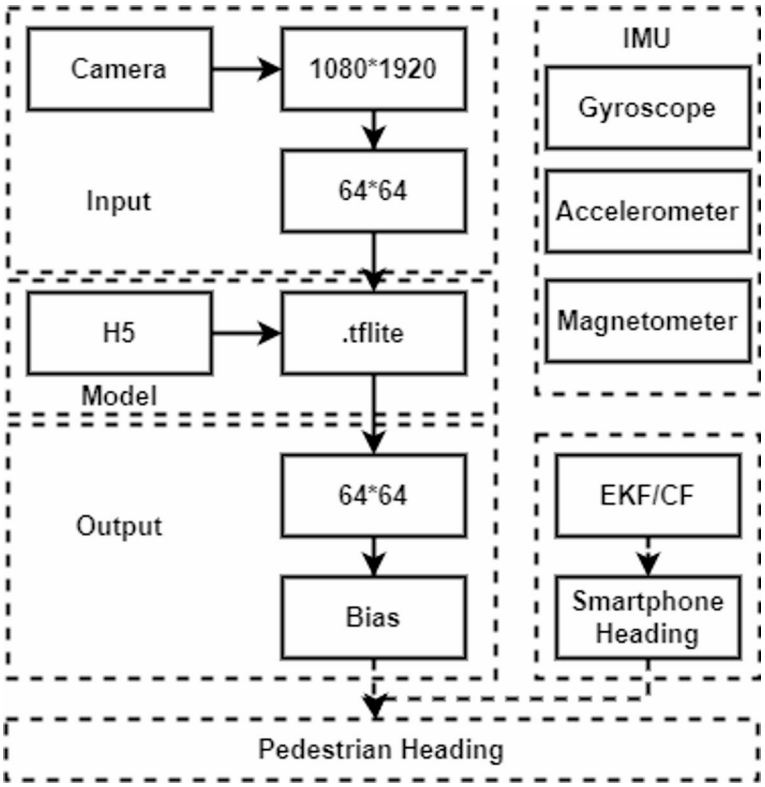

Fig. 8. Pedestrian heading detection workflow using a camera image.

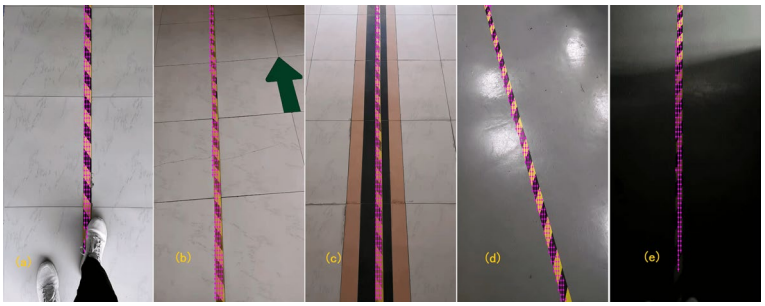

Fig. 9. Real-time recognized ground reference lines.

The inclusion of visual heading observations in the Kalman update step allows the filter to correct for accumulated errors in the inertial estimate. Over time, this fusion process balances the short-term responsiveness of gyroscope-based updates with the long-term stability of camera-derived absolute heading cues.

To initialize the Kalman filter, an initial heading h_0 must be provided. This is obtained by computing the orientation from accelerometer and magnetometer measurements, as described in earlier sections. The local magnetic declination is applied to convert the magnetometer-based heading to true north, resulting in an absolute initial heading h_{accMag} used to seed the filter:

$$h_0 = h_{\text{accMag}} \tag{21}$$

By fusing inertial and visual information within this recursive estimation framework, the proposed method ensures both responsiveness and robustness in dynamic pedestrian navigation scenarios. Visual corrections compensate for drift, while inertial updates provide continuity and temporal resolution, resulting in an accurate and stable heading estimate suitable for real-time mobile deployment.

Evaluation and experiment

In the previous section, we introduced the algorithms we developed and used in this study. We use these algorithms in a real environment to improve pedestrian navigation capability and stability. Most smartphones

currently integrate MEMS sensors; in particular, Google provides APIs for developers to obtain multiple sensor measurements in real time. We develop a related real-time navigation application, named "PedestrianNav", based on our algorithms and real conditions that can run on most mainstream Android smartphones. In this section, we perform field experiments to validate the proposed algorithms and evaluate the fused heading precision. PedestrianNav integrates the algorithms proposed in this study, which can be run on most Android smartphones. In this study, we chose the Huawei Mate20 as the handheld test. The experimental site was chosen at a school athletic field where the white lines on the ground were straight lines and used as ground reference lines for the experiments. It should be noted that the deep learning model in this study trains a large number of different types of ground reference lines, such as lane lines and blind alleys, so that the algorithm in this study can achieve real-time recognition and computation of a wide range of ground reference lines. In this paper, we list two mode experiments: one is a kinematic experiment, in which pedestrians hold smartphones in their hands and walk along the ground reference line, and the other is a static test.

Kinematic heading experiment

In this section, we execute a kinematic heading experiment with the application Fig. 10 is the screenshot of the real-time test result. In the center of the image (Fig. 10), the cany points are the recognized center points of the ground reference line. We performed kinematic experiments along two ground reference lines. First, its azimuth is 0.014 deg, and the second is 180.014 deg. For each ground reference line, we repeated the test three times. The following graphics and tables show the statistical results.

Figure 11 shows the experimental results: the pedestrian walked along the first ground reference line (azimuth is 0.014 deg). Three graphics show different heading strategies: the upper graphic is the $Acc + Mag$ computed heading, the central graphic is the $Acc + Mag + Gyro$ fusion heading, and the lower graphic is the fused vision fusion result combining IMU measurements.

Figure 12 shows the experimental results: the pedestrian walked along the first ground reference line (azimuth is 180.014 deg). It contains three graphics and shows different heading strategies: the upper graphic is the $Acc + Mag$ computed heading, the central graphic is the $Acc + Mag + Gyro$ fusion heading, and the lower graphic is the vision fusion result combining IMU measurements.

Table 3 summarizes the heading average bias of the kinematic experiments. We repeated kinematic experiments three times in two ground reference lines, so in Table 3, $Test1 - Test3$ are the mean absolute error(MAE) values of the first reference line (azimuth is 180.014 deg), and $Test4 - Test6$ are the MAEs of the second reference line (azimuth is 0.014 deg).

Static heading experiment

Figure 13 shows the equipment installation scenario of the static heading experiment. Smartphones can capture the ground reference line (white line). The smartphone is mounted on the device, stationary, with the longitudinal axis of the smartphone inclined at an angle to the ground reference line. Smartphones can capture and recognize ground reference lines in real time.

In Table 4, *Test*1, *Test*2 and *Test*3 denote the MAEs of these repeated experiments on the first ground reference line (180.014 deg), and others denote the repeated experiment results on the second ground reference line (0.014 deg).

Discussion

Static and kinematic mode heading experiments

Based on the above heading experiments, from the statistical results, we found that in the pedestrian dynamic mode, both reference line 1 and reference line 2, the $Acc + Mag + Gyro$ fusion heading accuracy improved by 5% and 15%, respectively, compared to the $Acc + Mag$ heading strategy, and the proposed visual fusion IMU algorithm heading accuracy improved by an average of 5% (reference line 1) and 6%(reference line 2). In static mode, again, the average deviation is reduced to a certain extent, and the average improvement of the

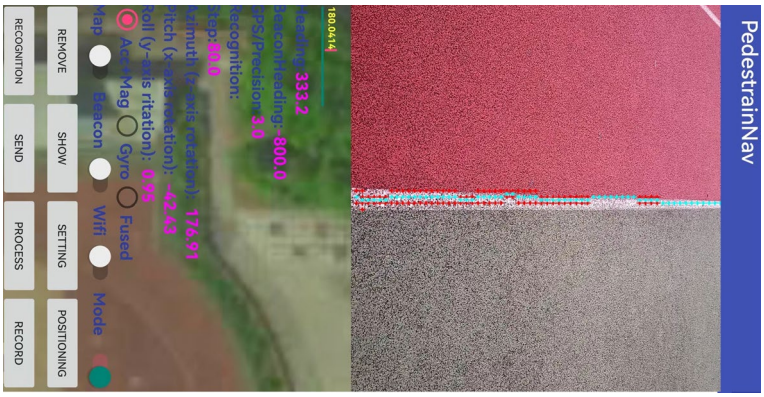

Fig. 10. The screenshot shows real-time recognized ground reference lines using "PedestrianNav", it is the self-developed Android application, the map calls the Gaode online map API (https://mobile.amap.com/).

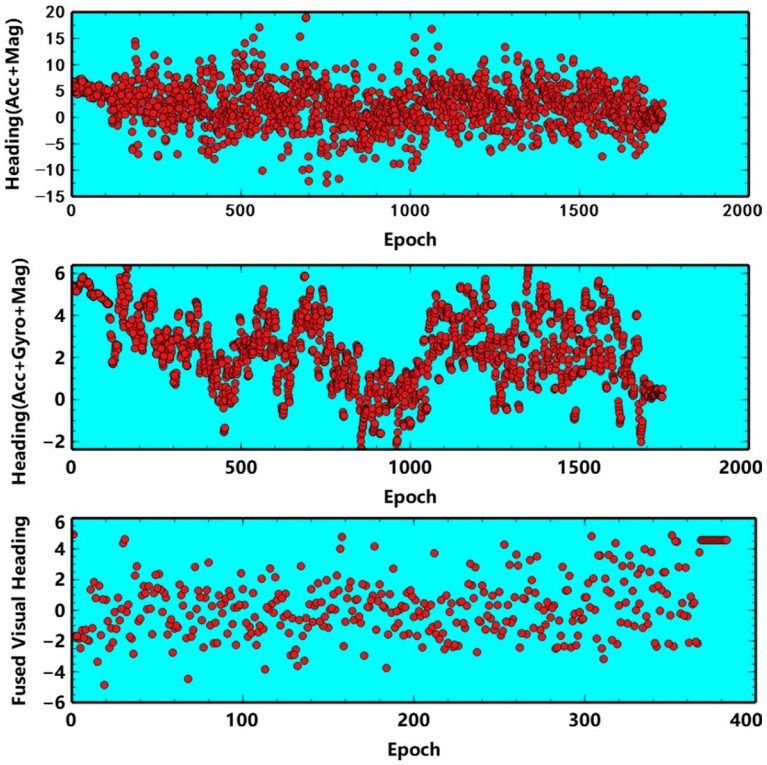

Fig. 11. Time series plot of pedestrian heading for real-time kinematic experiment (the azimuth of ground reference line is 0.014 deg).

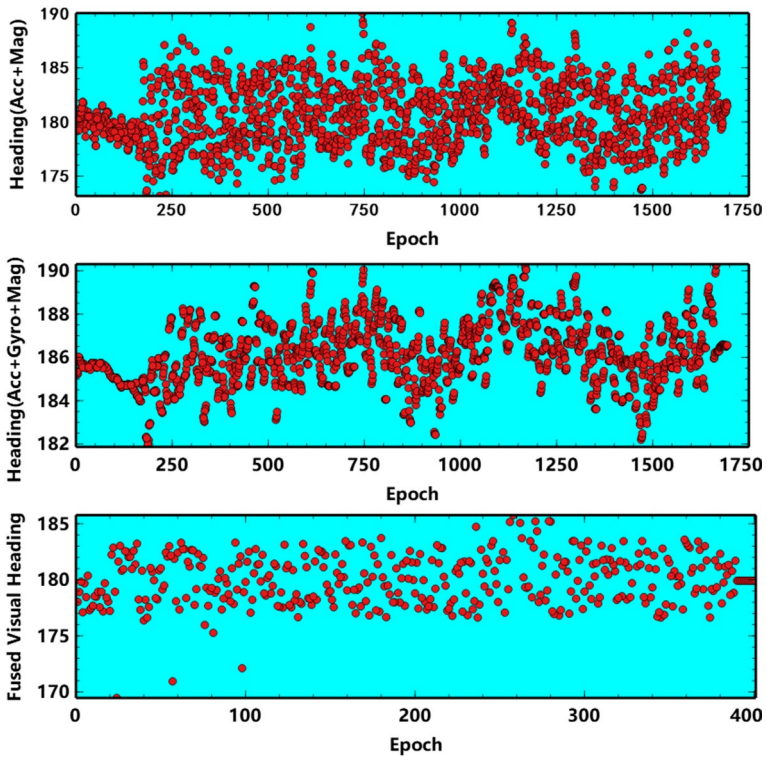

Fig. 12. Time series plot of pedestrian heading for real-time kinematic experiment (the azimuth of ground reference line is 180.014 deg).

Mode	Mean absolute error of heading(deg)					
	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
Acc+Mag heading	7.1	7.9	8.5	9.3	8.6	7.9
Acc+Mag+Gyro heading	6.9	7.8	7.5	6.7	7.5	7.5
Fused visual heading	6.8	6.7	7.4	6.6	6.8	6.9

Table 3. Statistical results of kinematic heading experiment.

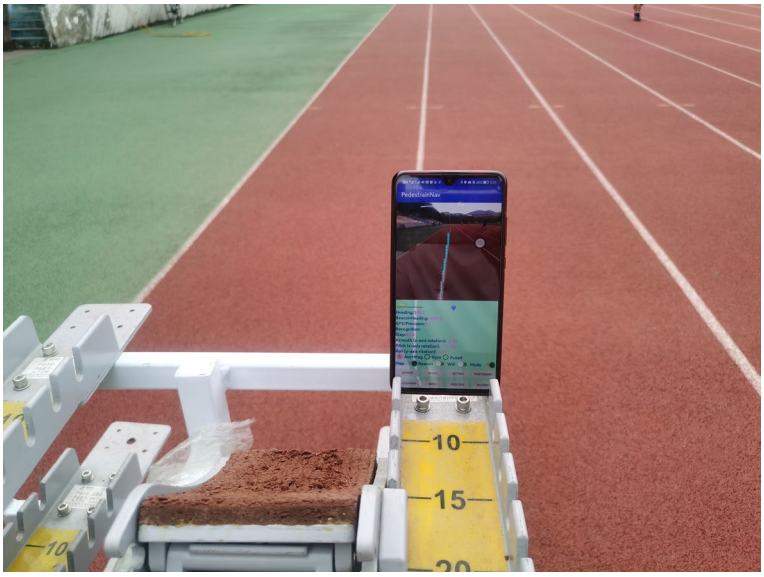

Fig. 13. Static heading validation experiment.

Mode	Mean absolute error of heading(deg)					
	Test 1	Test 2	Test 3	Test 4	Test 5	Test 6
Acc+Mag heading	5.3	6.9	6.5	6.3	5.8	5.8
Acc+Mag+Gyro heading	4.6	5.5	5.9	5.4	5.5	5.8
Fused visual heading	5.5	4.8	5.1	5.2	4.8	5.7

Table 4. Statistical results of static heading experiment

$Acc + Mag + Gyro$ fusion heading accuracy is 14% and 6% compared to the $Acc + Mag$ heading strategy. In summary, the $Acc + Mag + Gyro$ fusion heading algorithm is superior to the $Acc + Mag$ heading strategy in both dynamic and static modes, and the visual fusion IMU heading algorithm proposed in this paper further reduces the heading bias. This innovative approach is effective, which, therefore, has implications for pedestrian navigation and positioning accuracy improvement.

Experiments were carried out in this study under various environmental conditions and for different ground reference lines. In addition to the above mentioned tape on the ground and the lines on the sports field, we can also identify blind alleys. (Fig. 14 shows the real-time recognition results of a blind path on the pavement. The red recognition is the blind path floor tiles and the cyan is the calculated blind path centerline. Immediately adjacent to the pavement, many tall trees cause serious satellite signal blocking. If only using GNSS positioning, consumer-grade intelligent terminal positioning accuracy is relatively poor, especially in some places. In addition to trees, there are tall buildings, resulting in serious multipath interference. Therefore, real-time recognition of trees to improve pedestrian heading accuracy can help improve pedestrian navigation positioning accuracy. In the future, computer vision, deep learning and speech technologies can be further combined to enhance or develop functions or products suitable for assisting navigation for people with visual impairments.

In addition, in our test field, we also set up continuous ground reference lines along the floor of the building corridors, and in fact, our algorithm also divides the continuous reference lines to multiple straight segments, which are then recognized; the straight segments have a clear direction and can be used as heading correction, while the curved segments are not easy to be used.

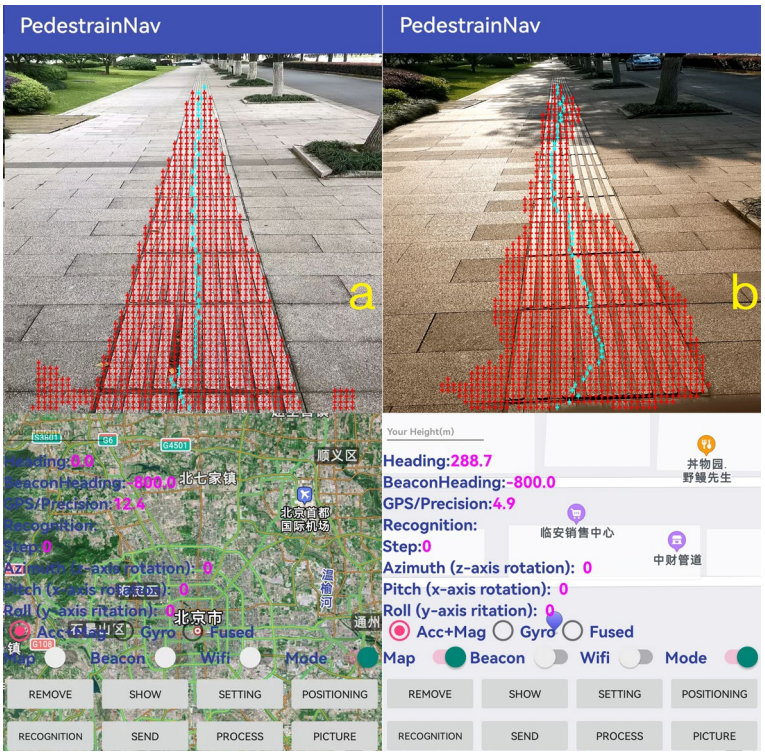

Fig. 14. The screenshot shows real-time recognized ground blind walkways using "PedestrianNav", it is the self-developed Android application, the map calls the Gaode online map API.

Pedestrian heading experiment in a dark light environment

The pedestrian heading algorithm proposed in this study is based on real-time image recognition, where the clarity of the image is directly correlated with the brightness of the light. The above experiments were mostly conducted during the daytime with high light intensity, and we also conducted experiments under low light conditions to verify that the algorithm can identify ground reference lines in real time under dim light at night, cf. Fig. 15. Of course, as the light gradually fades, the heading calculation algorithm that relies on image recognition will fail, and at this time, it needs to be fused with the heading calculated from IMU observations and GNSS observations to ensure adaptation to various scenarios.

In addition, we try to improve pedestrian heading accuracy by constructing a ground reference line in a simple way or using an existing ground reference as a ground reference line to assist in improving the navigation and positioning accuracy of a consumer-grade smartphone in a satellite signal occlusion environment, and to promote mass adoption, and then many times the reference line disappears or there is the problem of multiple turns, etc., which needs to be adapted to more situations, and I think that in the future, we can combine visual odometry and deep learning methods to improve the research on heading error prediction during missing reference lines.

Comparison with classical VIO and SLAM methods

Classical VIO and SLAM methods typically rely on point features. However, point-based odometry offers only local orientation constraints (derived from short-baseline feature geometries) and often struggles under pure rotational motion or in low-feature environments. In contrast, incorporating line features and vanishing points (VPs) introduces global orientation constraints; for example, in indoor Manhattan environments, orthogonal wall directions provide strong cues about the camera's yaw angle. In addition, our method avoids the full pipeline of computationally intensive tasks (global map optimization, loop closure, bundle adjustment, etc.) that classical SLAM systems undertake. Please refer to Table 5 for more details.

By contrast, our approach uses a small neural network to estimate heading from a single camera frame, combined with lightweight IMU fusion, drastically reducing per-frame computation. The network operates on a single image without iterative feature matching or bundle adjustment and has a modest number of parameters, making inference fast. In our implementation, inference takes on the order of only tens of milliseconds per frame. This is on par with, or faster than, the per-frame processing time of optimized VIO algorithms running on mobile-class hardware. Our method avoids these bottlenecks entirely by focusing solely on heading estimation; we eliminate the need for map management and long-term optimization. This focus, combined with an efficient neural network (which can be accelerated via modern mobile NPUs or GPUs), allows real-time performance on the edge with a small computational footprint. In summary, we stand by the claim that our approach is lightweight relative to full SLAM: it requires fewer CPU cycles and less memory, making it well-suited for deployment on resource-constrained platforms.

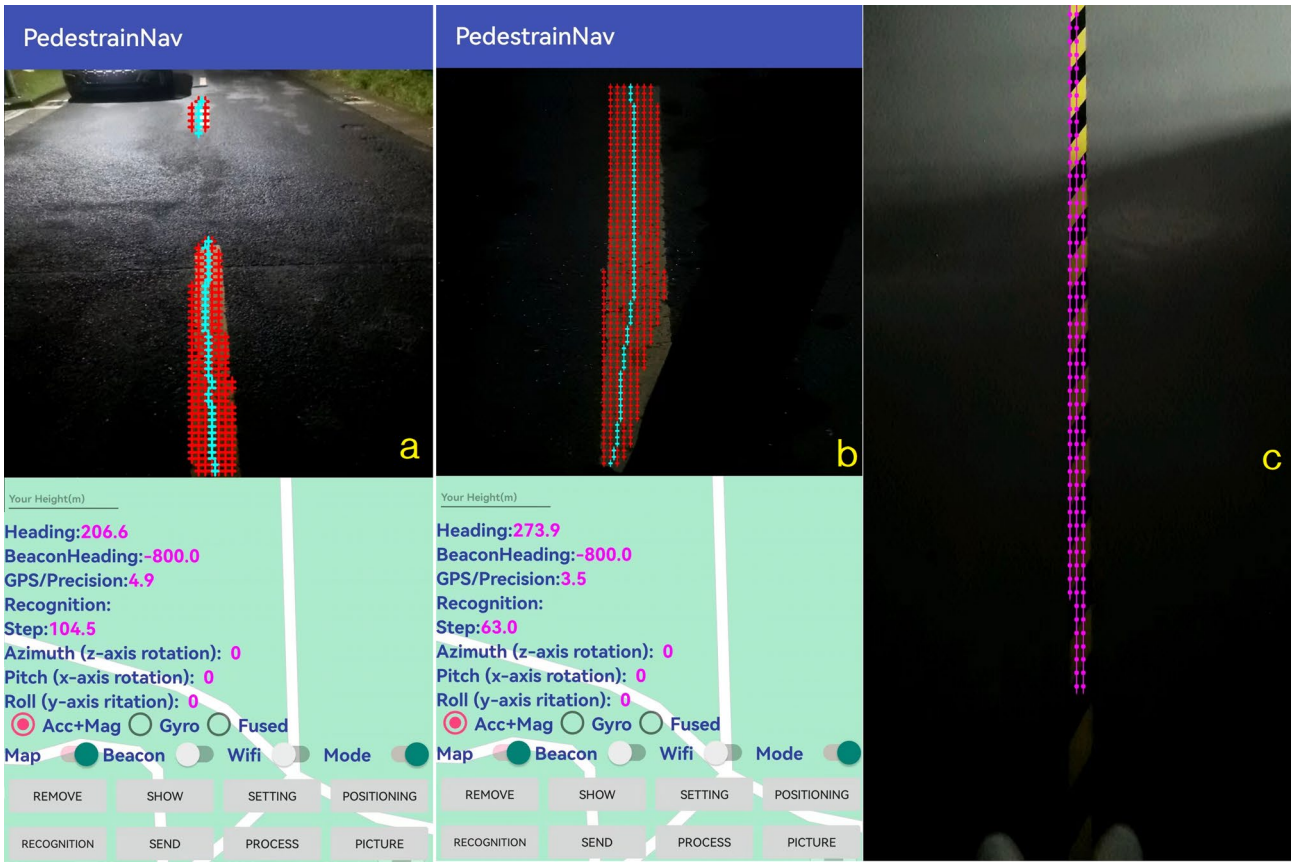

Fig. 15. The screenshot shows real time recognized ground lane lines at night using "PedestrianNav", it is the self-developed Android application, the map calls the Gaode online map API .

Method	Runtime (ms)	FPS	Max Heading Error ($^\circ$)	Hardware	Notes, Trade-offs and Implementation Remarks
ORB-SLAM2 [31]	80–120	∼10–12	$> 10^\circ$	Desktop CPU (i7)	Full SLAM pipeline including loop closure and global BA. No IMU fusion; efficient keypoint tracking but high yaw drift without loop closure; suboptimal in low-texture or dynamic scenes.
VINS-Mono [35]	25–40	25–40	$5–10^\circ$	ARM Cortex-A72 (Raspberry Pi 4)	Filter-based monocular VIO; real-time on embedded CPUs. Sliding-window optimization yields good accuracy but degrades with poor texture or motion blur.
ORB-SLAM3 [36]	30–50	∼20	$\sim 5–8^\circ$	Desktop CPU	Visual-inertial SLAM with multi-map and loop closure support; higher accuracy in large-scale settings, but slower on resource-constrained devices.
StructVIO [37]	∼30	∼30–35	$\sim 4.7^\circ$	Desktop CPU	Adds Manhattan-world structural constraints for improved yaw correction. Still relies on geometric line extraction and post-processing; moderate computational overhead.
SG-VIO [38]	∼26	∼38	$\sim 3.6^\circ$	Desktop CPU	Tightly couples structural lines and gravity direction. Improves yaw estimation in indoor settings; optimized for scenes with orthogonal structures.
POL-VIO [39]	22	45	$\sim 3.2^\circ$	NVIDIA TX2 (edge GPU)	Combines point-line features with optimized feature processing. Enables real-time edge inference but requires careful geometric threshold tuning.
BIPS [40]	18	55	$5–8^\circ$	Snapdragon 855	Hybrid indoor positioning using BLE + monocular VINS + BIM priors. Fast, mobile-ready vision front-end; heading accuracy depends on semantic alignment and BLE signal integrity.
IMU-Only [41]	$\ll 1$	>500	(1°/min drift) Unlimited with long paths without calibration and thermal effect	Smartphone IMU	Pure inertial integration; extremely lightweight but suffers from rapid yaw drift. Requires external heading correction to remain usable.
Proposed (Ours)	∼15	60–66	$\mathbf{2-6^\circ}$	Mobile ARM CPU	CNN-based heading estimation from single RGB frame fused with IMU. Robust to texture loss and blur. No map, loop closure, or keypoint tracking required. Real-time, mobile-friendly orientation aid with minimal overhead.

Table 5. Comparison of SLAM/VIO and learning-based odometry methods in terms of runtime, orientation accuracy, hardware suitability, and practical trade-offs

Conclusions

Heading accuracy directly affects positioning accuracy, so this study focuses on the key element of heading in pedestrian navigation and positioning. Previous studies have mostly used the built-in accelerometer and magnetometer observations of smart terminals to derive the magnetic north azimuth of the terminal, combined with the local magnetic declination to obtain the true north azimuth, and finally used the smart terminal azimuth to replace the pedestrian heading. Of course, there is a certain swing during pedestrian movement, and the real heading of the pedestrian is not the same as the heading of the intelligent terminal. Gyroscope observations are also often used for heading integration. However, they also lead to the accumulation of heading errors due to device errors, so scholars also often use Kalman filters, EKFs, particle filters, compensation filters and other algorithms to achieve IMU observation fusion to deduce terminal heading. Based on the IMU observations of intelligent terminals and traditional algorithms, this study investigates the use of image data acquired by terminal vision sensors to identify ground reference lines in real time and combines computer vision and deep learning algorithms to improve pedestrian heading. Experimental results show that the method significantly improves the pedestrian heading accuracy in dynamic and, in most cases, in static mode. In addition, this study also conducted experiments on the identification of blind lanes in urban streets. The results show that the algorithm can also acquire blind reference lines in real time in boulevards with weak satellite signals, which can improve the heading accuracy. In addition, this study also verifies adaptability in dim light conditions. The results show that in night mode, ground reference lines under light, such as lane lines, can still be recognized accurately, although the recognition accuracy decreases as the light intensity becomes weaker until it is not recognized. In general, the algorithm proposed in this study has practical implications for assisting pedestrian heading accuracy. However, there are also shortcomings, such as the complexity of the application scenario and the fact that although we choose to install inexpensive tape as a reference line, it still needs to be laid out in advance, so subsequent research will hopefully be able to adapt to more types of ground reference lines that do not need to be laid out, such as road edges and fences, to improve the practical application capability.

Data availability

The datasets or code generated and/or analysed during the current study are available in Google Drive, everyone can download the site: https://drive.google.com/drive/folders/1_ilgxOcdIAlkQuS8SVLS4MYIMmZ3C5Ma?usp=sharing The folder ”Pedestrian Navigation Heading Code” contains three subfolders ”App code” ”Preprocess” ”Training”. 1, Preprocess, it provides raw data and preprocess program (python). 2, Training, it provides U-net network (python), traning and validation data. 3, Android application code (Java).

Received: 11 December 2024; Accepted: 23 July 2025
Published online: 28 August 2025

References

1. Chen, W., Chen, R., Chen, Y., Kuusniemi, H. & Wang, J. An effective pedestrian dead reckoning algorithm using a unified heading error model. In *IEEE/ION Position, Location and Navigation Symposium*, 340–347 (IEEE, 2010).
2. Huang, C., Liao, Z. & Zhao, L. Synergism of ins and pdr in self-contained pedestrian tracking with a miniature sensor module. *IEEE Sens. J.* **10**, 1349–1359 (2010).
3. Elloumi, W. *et al.* Indoor navigation assistance with a smartphone camera based on vanishing points. In *International Conference on Indoor Positioning and Indoor Navigation*, 1–9 (IEEE, 2013).
4. Renaudin, V. & Combettes, C. Magnetic, acceleration fields and gyroscope quaternion (magyq)-based attitude estimation with smartphone sensors for indoor pedestrian navigation. *Sensors* **14**, 22864–22890 (2014).
5. Farahsari, P. S., Farahzadi, A., Rezazadeh, J. & Bagheri, A. A survey on indoor positioning systems for iot-based applications. *IEEE Internet Things J.* **9**, 7680–7699 (2022).
6. Zhao, C. et al. Data-driven indoor positioning correction for infrastructure-enabled autonomous driving systems: A lifelong framework. *IEEE Trans. Intell. Transp. Syst.* **24**, 3908–3921 (2023).
7. Liu, Z. et al. Machine learning for time-of-arrival estimation with 5g signals in indoor positioning. *IEEE Internet Things J.* **10**, 9782–9795 (2023).
8. Ye, H., Yang, B., Long, Z. & Dai, C. A method of indoor positioning by signal fitting and pdda algorithm using ble aoa device. *IEEE Sens. J.* **22**, 7877–7887 (2022).
9. Kang, W., Nam, S., Han, Y. & Lee, S. Improved heading estimation for smartphone-based indoor positioning systems. In *2012 IEEE 23rd International Symposium on Personal, Indoor and Mobile Radio Communications-(PIMRC)*, 2449–2453 (IEEE, 2012).
10. Afzal, M. H., Renaudin, V. & Lachapelle, G. Magnetic field based heading estimation for pedestrian navigation environments. In *2011 International Conference on Indoor Positioning and Indoor Navigation*, 1–10 (IEEE, 2011).
11. Zhou, Z., Zhang, Z., Mo, S., Wu, J. & Fourati, H. Online calibrated, energy-aware and heading corrected pedestrian navigation with foot-mounted marg sensors. *Measurement* **206**, 112268 (2023).
12. Wang, Y., Kuang, J., Li, Y. & Niu, X. Magnetic field-enhanced learning-based inertial odometry for indoor pedestrian. *IEEE Trans. Instrum. Meas.* **71**, 1–13 (2022).
13. Yao, Y., Liu, Y., Zhou, Z. & Xu, X. A magnetic interference detection-based fusion heading estimation method for pedestrian dead reckoning positioning. *IEEE Sens. J.* **23**, 677–688 (2022).
14. Jiang, C. et al. Smartphone pdr/gnss integration via factor graph optimization for pedestrian navigation. *IEEE Trans. Instrum. Meas.* **71**, 1–12 (2022).
15. Manos, A., Hazan, T. & Klein, I. Walking direction estimation using smartphone sensors: A deep network-based framework. *IEEE Trans. Instrum. Meas.* **71**, 1–12 (2022).
16. Wu, D., Xia, L. & Geng, J. Heading estimation for pedestrian dead reckoning based on robust adaptive kalman filtering. *Sensors* **18**, 1970 (2018).
17. Guo, Y., Liu, H., Ye, J., Wang, S. & Duan, C. Pedestrian heading estimation methods based on multiple phone carrying modes. *Mob. Inf. Syst.* **2021**, 1193268 (2021).
18. Ilyas, M., Cho, K., Baeg, S.-H. & Park, S. Drift reduction in pedestrian navigation system by exploiting motion constraints and magnetic field. *Sensors* **16**, 1455 (2016).
19. Deng, Z.-A., Wang, G., Hu, Y. & Wu, D. Heading estimation for indoor pedestrian navigation using a smartphone in the pocket. *Sensors* **15**, 21518–21536 (2015).

20. Abdulrahim, K., Hide, C., Moore, T. & Hill, C. Aiding mems imu with building heading for indoor pedestrian navigation. In *2010 ubiquitous positioning indoor navigation and location based service*, 1–6 (IEEE, 2010).
21. Foxlin, E. Pedestrian tracking with shoe-mounted inertial sensors. *IEEE Comput. Graphics Appl.* **25**, 38–46 (2005).
22. Stirling, R., Fyfe, K. & Lachapelle, G. Evaluation of a new method of heading estimation for pedestrian dead reckoning using shoe mounted sensors. *J. Navig.* **58**, 31–45 (2005).
23. Manos, A., Klein, I. & Hazan, T. Gravity direction estimation and heading determination for pedestrian navigation. In *2018 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 206–212 (IEEE, 2018).
24. Asraf, O., Shama, F. & Klein, I. Pdrnet: A deep-learning pedestrian dead reckoning framework. *IEEE Sens. J.* **22**, 4932–4939 (2021).
25. Mansour, A. et al. Everywhere: A framework for ubiquitous indoor localization. *IEEE Internet Things J.* **10**, 5095–5113 (2022).
26. Mansour, A. & Chen, W. Suns: A user-friendly scheme for seamless and ubiquitous navigation based on an enhanced indoor-outdoor environmental awareness approach. *Remote Sens.* **14**, 5263 (2022).
27. Shen, L. L. & Hui, W. W. S. Improved pedestrian dead-reckoning-based indoor positioning by rssi-based heading correction. *IEEE Sens. J.* **16**, 7762–7773 (2016).
28. Ruotsalainen, L., Kuusniemi, H. & Chen, R. Heading change detection for indoor navigation with a smartphone camera. In *2011 International Conference on Indoor Positioning and Indoor Navigation*, 1–7 (IEEE, 2011).
29. Elloumi, W. et al. Indoor navigation assistance with a smartphone camera based on vanishing points. In *International Conference on Indoor Positioning and Indoor Navigation*, 1–9 (IEEE, 2013).
30. Wang, Y. & Zhao, H. Improved smartphone-based indoor pedestrian dead reckoning assisted by visible light positioning. *IEEE Sens. J.* **19**, 2902–2908 (2018).
31. Mur-Artal, R. & Tardós, J. D. Orb-slam2: An open-source slam system for monocular, stereo, and rgb-d cameras. *IEEE Trans. Rob.* **33**, 1255–1262 (2017).
32. Lachapelle, G. Pedestrian navigation with high sensitivity gps receivers and mems. *Pers. Ubiquit. Comput.* **11**, 481–488 (2007).
33. Yang, F. et al. A pairwise ssd fingerprinting method of smartphone indoor localization for enhanced usability. *Remote Sens.* **11**, 566 (2019).
34. Ronneberger, O., Fischer, P. & Brox, T. U-net: Convolutional networks for biomedical image segmentation. In *Medical image computing and computer-assisted intervention–MICCAI 2015: 18th international conference, Munich, Germany, October 5-9, 2015, proceedings, part III 18*, 234–241 (Springer, 2015).
35. Qin, T., Li, P. & Shen, S. Vins-mono: A robust and versatile monocular visual-inertial state estimator. *IEEE Trans. Rob.* **34**, 1004–1020 (2018).
36. Campos, C., Elvira, R., Rodríguez, J. J. G., Montiel, J. M. & Tardós, J. D. Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam. *IEEE Trans. Rob.* **37**, 1874–1890 (2021).
37. Zou, D., Wu, Y., Pei, L., Ling, H. & Yu, W. Structvio: Visual-inertial odometry with structural regularity of man-made environments. *IEEE Trans. Rob.* **35**, 999–1013 (2019).
38. Yao, H. et al. Sg-vio: Monocular visual-inertial odometry with tightly coupled structural lines and gravity to avoid degeneracy. *IEEE Internet Things J.* (2024).
39. Si, H., Yu, H., Chen, K. & Yang, W. Point-line visual-inertial odometry with optimized line feature processing. *IEEE Trans. Instrum. Meas.* (2024).
40. Kong, X., Wu, C., You, Y., Lv, Z. & Zhao, Z. Hybrid indoor positioning method of ble and monocular vins-based smartphone. *IEEE Trans. Instrum. Meas.* **72**, 1–13 (2023).
41. Labs, K. navx-sensor imu drift behavior (2023). Available at: https://pdocs.kauailabs.com/navx-mxp/usage/guidance/imu-drift/.

Author contributions

We would like to acknowledge the valuable contributions made by each author to this article. As the lead author, Junhua Ye has taken primary responsibility for conceptualizing the article, developing the structure, and writing the majority of the content. His expertise in the field and dedication to the project have been instrumental in shaping the final version. Ahmed Mansour analyzed the test data and validated algorithm. Fenghua Huang , has provided invaluable guidance and supervision throughout the writing process.

Funding

This work was supported by Zhejiang Provincial Natural Science Foundation Project (LTGG23D040001) and Open Project of Fujian Key Laboratory of Spatial Information Perception and Intelligent Processing(Yango University, Grant No. FKLSIPIP1023).

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to F.H.