A Computer Vision Approach for Pedestrian Walking Direction Estimation with Wearable Inertial Sensors: PatternNet

Hanyuan Fu, Thomas Bonis, Valerie Renaudin, Ni Zhu

A Computer Vision Approach for Pedestrian Walking Direction Estimation with Wearable Inertial Sensors: PatternNet

Hanyuan Fu
AME-GEOLOC
Univ. Gustave Eiffel
F-44344 Bouguenais, France
hanyuan.fu@univ-eiffel.fr

Thomas Bonis
CNRS, LAMA
Univ. Gustave Eiffel,
Univ Paris Est Creteil
F-77420 Marne-la-Vallée, France
thomas.bonis@univ-eiffel.fr

Valerie Renaudin
AME-GEOLOC
Univ. Gustave Eiffel
F-44344 Bouguenais, France
valerie.renaudin@univ-eiffel.fr

Ni Zhu
AME-GEOLOC
Univ. Gustave Eiffel
F-44344 Bouguenais, France
ni.zhu@univ-eiffel.fr

***Abstract*—In this paper, we propose an image-based neural network approach (PatternNet) for walking direction estimation with wearable inertial sensors. Gait event segmentation and projection are used to convert the inertial signals to image-like tabular samples, from which a Convolutional Neural Network (CNN) extracts geometrical features for walking direction inference. To embrace the diversity of individual walking characteristics and different ways to carry the device, tailor-made models are constructed based on individual users' gait characteristics and the device-carrying mode. Experimental assessments of the proposed method and a competing method (RoNIN) are carried out in real-life situations and over 3 km total walking distance, covering indoor and outdoor environments, involving both sighted and visually impaired volunteers carrying the device in three different ways: texting, swinging and in a jacket pocket. PatternNet estimates the walking directions with a mean accuracy between 7 to 10 degrees for the three test persons and is 1.5 times better than RONIN estimates.**

***Index Terms*—Indoor positioning, inertial sensors, pedestrian navigation, pedestrian dead reckoning, walking direction, deep learning**

I. Introduction

Wearable inertial sensors enable autonomous and low-cost pedestrian positioning solutions. Appealing it is, estimating the user position from the inertial signal is difficult due to sensor noise, diverse user dynamics (slow/normal/fast walking, going up/downstairs, stepping, ...), device handling modes (in texting/swinging hand, in a pocket, on the armband) and individual walking characteristics. A popular strategy is to use the inertial signal to detect the user's gait events (steps or strides) and estimate the user's displacement (step/stride length and direction) during a gait cycle [1]. This paper focuses on the walking direction estimation task. Estimating the user's walking direction is challenging because of the changeable misalignment between the device's pointing direction and the user's walking direction.

A first attempt to infer the walking direction from inertial signals consists in projecting the acceleration of the device on a horizontal plane and estimating the main direction of the accelerations through principal component analysis (PCA) [2]. However, the performance of such an approach depends on the device handling mode and on the user's walking characteristics. Another method [3] uses GMMs (Gaussian Mixture Model) to approximate the horizontal acceleration point cloud during walking. A GMM is learned for each handling mode and for each user. To infer the walking direction, we rotate the test sample of the horizontal acceleration point cloud to maximize the log-likelihood of such observation given the learned GMM. Recent approaches thus use deep learning to cope with the diversity of motion patterns. This kind of method uses a full deep learning approach with the time series of sensor data to regress the walking directions ([4], [5]) or velocity vectors ([6]) of the user. An example is RoNIN [6] (Robust Neural Inertial Navigation), a deep learning model trained with inertial data collected by 100 human subjects carrying the device in several ways.

The state-of-the-art methods face limitations. The bias between the walking direction and the PCA direction varies with different user dynamics (fast/normal walking), device handling modes, and user profiles, however, PCA is not able to distinguish any of these contexts. Learned from the weakness of PCA, GMM is dynamics-normalized, user-customized, and device-handling-mode-wise. However, as we can see in Figures 1 and 2, acceleration patterns are not Gaussian mixtures and thus are poorly described by GMM. Furthermore, both PCA and GMM suffer from 180° ambiguity which is difficult to eliminate completely. In some cases, the ambiguity is partly solved by GMM if the model is strongly asymmetric. RoNIN [6] achieves robust performance in diverse scenarios and on unseen users, without retraining. Instead of segmenting the inertial signal by gait events to obtain samples for training and testing, RoNIN's samples are fixed-length sequences of 200 frames (1 second), and it outputs a velocity vector prediction every 5 frames. As analyzed and evaluated in the state-of-the-art survey paper [1], ignoring gait events results in noisy velocity inferences, which contribute to an accumulated drift

This work was supported by the French National Research Agency under Grant ANR-20-LCV1-0002 and the French company Okeenea Digital.

in positioning.

This paper proposes a walking direction estimation framework–PatternNet, that exploits the characteristics of inertial signals recorded on different individuals and in diverse scenarios. The main contributions of this paper are:

- Unlike existing deep learning methods for user walking direction inference that use time series, we propose a geometrical and image processing deep learning approach to fully exploit the acceleration patterns.
- Our models are trained with small-size datasets and valid over time.

The performance of the proposed image-based approach is assessed against the ground truth walking direction given by a foot-mounted device. Experimental assessments are carried out in real-life situations and over 3 km total walking distance, covering indoor and outdoor environments, involving both sighted and visually impaired volunteers carrying the device in three different ways: texting, swinging, and in a jacket pocket.

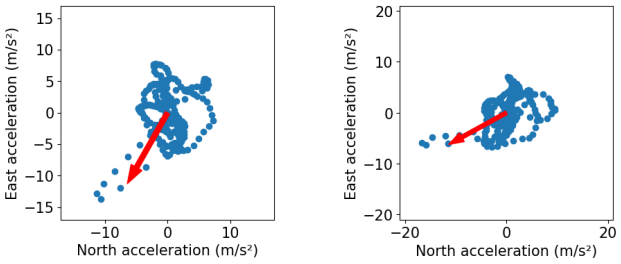
Section II describes the observation of the relationship between the device's horizontal acceleration and the user's walking direction, which is the basis of our framework. Section III presents the proposed computer vision approach for walking direction estimation – PatternNet. Section IV is dedicated to the experimental assessment, including the evaluation and comparison of our method and RoNIN [6] on pedestrian tracks. Section V concludes the paper.

II. Observing the Relationship Between the Device's Horizontal Acceleration and the User's Walking Direction

Inspired by PCA and GMM, our approach extracts walking direction-related geometrical information from the device's horizontal acceleration.

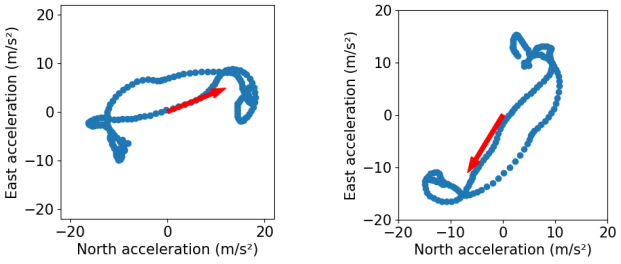
Our volunteers are asked to walk naturally and carry the device in several ways. Segments of raw acceleration readings over a gait cycle are projected on the local North-East plane. Details about the segmentation and projection procedures can be found in section III. Stride events are adopted to segment the inertial signals because a stride corresponds to one complete cycle of the human walking gait whereas a step is only half of it.

A stride segment of horizontal accelerations is considered a sample. Fig.1 shows 2 samples of a user keeping the device in his jacket pocket and Fig.2 shows 2 samples of a user holding the device in his swinging hand. The red arrows indicate the users' walking directions. We observe repetitive acceleration patterns during the walks. In Fig.1, the patterns feature a zone of higher density of acceleration points and a long tail that is approximately parallel to the walking direction. Patterns in Fig 2 have "banana" shape and the PCA of the "banana" indicates the line of walk. In both cases the patterns aren't 180° ambiguous, i.e. if we rotate the pattern by 180° around the origin, the rotated pattern doesn't coincide with the original one. In another word, some horizontal acceleration patterns can indicate the user's walking direction without ambiguity.



(a) (b)

Fig. 1: Two horizontal acceleration samples of a user keeping the device in his jacket right pocket. The red arrows indicate the user's walking directions. (a) Sample 1; (b) Sample 2



(a) (b)

Fig. 2: Two horizontal acceleration samples of a user holding the device in his swinging left hand. The red arrows indicate the user's walking directions. (a) Sample 1; (b) Sample 2

We observe the diversity of acceleration patterns of different users carrying the device in the same way. Fig.2 and Fig.3 show the swinging patterns of 2 persons. Patterns in Fig.3 have an "8" shape, which is more 180°-ambiguous than the "banana" shape in Fig.2.

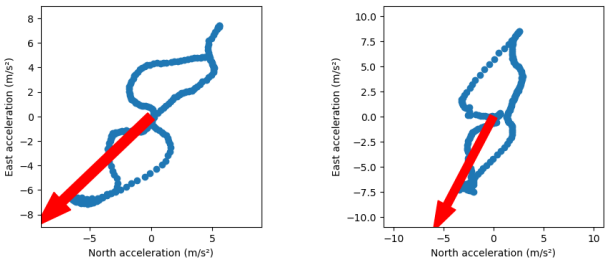
Convolutional neural networks (CNN) seem to fit the job of extracting informative geometric features from horizontal acceleration patterns for walking direction estimation. To embrace the diversity of patterns due to different carrying modes and different individual walking characteristics, each one of our models is tailor-made for one user and one device carrying mode.

III. PatternNet Framework

A. Preprocessing

Our framework takes only 3 inputs: 3D raw acceleration, angular rate, and magnetic field readings from a wearable IMU device. The preprocessing contains 3 steps:

1) Segmentation: Raw inertial signals are segmented with stride events (2 consecutive steps) detected by a Machine learning-based step detection algorithm SMARTStep [7]. SMARTStep considers a peak in the acceleration norm or a valley in the angular rate norm as a step instant, disregarding device-carrying modes. 2 parallel models are trained, one detects acceleration peaks and the other angular velocity valleys. A decision process is designed to combine the output of the 2 models.



(a) (b)

Fig. 3: Two horizontal acceleration samples of a user holding the device in her swinging left hand. The red arrows indicate the user's walking directions. (a) Sample 1; (b) Sample 2

2) Projection: Acceleration readings in the device body frame are projected on the local North-East-Down (NED) frame via MAGYQ [8], a real-time EKF-based device orientation tracking algorithm fusing acceleration, angular rate, and magnetometer readings, even under magnetic disturbances.

3) Converting horizontal acceleration signal to image-like tabular data: After the first 2 steps, we obtain stride segments of horizontal accelerations. The next step computes the 2D histogram of the sample: the horizontal acceleration point cloud is centered to [0, 0] by subtracting the mean value of each axis. To represent loyally the sample with a histogram, both axes (North = x-axis, East = y-axis) of the histogram have the same range and the same bin number. The range [-a, a] is defined in such a way that the histogram is big enough to count all points within the sample.

$$a = round(1.2 \times max(max(abs(a_x), max(abs(a_y))))) \quad (1)$$

where a_x is the segment of North accelerations and a_y, East accelerations. The bin number is set to 100 on each axis. Since the histogram is not Cartesian, it is then transposed and its columns are reversed, in such a way that the histogram has the same orientation as the original point cloud.

As a result, we obtain image-like tabular samples of size 100 * 100.

B. PatternNet

1) Converting regression problem to classification problem: The most straightforward way is to train our Network to regress an angle representing the user's walking direction, given an image sample. However, according to [9], a regression problem can be converted to a classification problem by quantization and CNN are better at classification than regression. Following this idea, our problem is defined as a 360-classes classification, each class represents an interval of 1 degree. The ground truth walking angles are one-hot coded into 360-entries binary vectors. This setup allows us to use the cross entropy loss for training.

2) Network architecture: The network architecture in [9] is adapted to our problem. The processing pipeline of the network is illustrated in Fig.4: the network contains 2 convolutional (conv) layers, a max pooling layer, and 2 fully connected

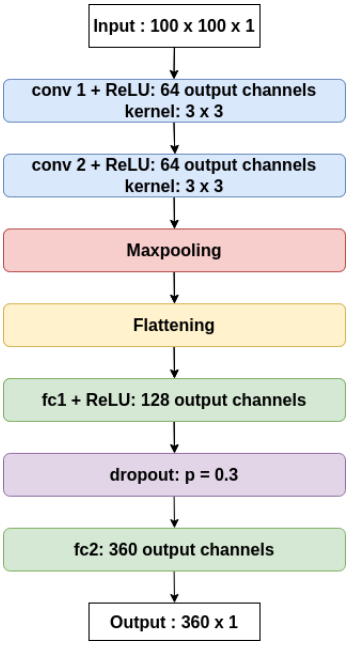


Fig. 4: Network architecture

(fc) layers. A 30% dropout is applied to the output of the first fc layer to reduce overfitting.

3) Implementation and training: Our models are implemented with the PyTorch framework and trained with the following parameters:

- Loss function: Cross Entropy Loss
- Optimizer: ADAM
- Learning rate: 0.001
- batch size : 50
- Initialization of model weights: xavier uniform [10] for convolutional layers
- Default epochs: 20
- Early stopping criteria: When training error $<$ validation error, every time the validation error is greater than it was in the last epoch, the trigger count increases by 1, when the trigger count exceeds 2, the training is stopped.

The training usually converges within 10 epochs.

C. Correction by moving average

The raw outputs from PatternNet sometimes suffer from noises in horizontal acceleration patterns or even 180° ambiguity. Luckily, noisy or ambiguous outputs are few enough to be smoothed out by moving average. The average of n angles is given by:

$$\bar{\alpha} = atan(\frac{\frac{1}{n}\sum_{j=1}^{n}\sin\alpha_j}{\frac{1}{n}\sum_{j=1}^{n}\cos\alpha_j}) \quad (2)$$

A moving average of 6 consecutive stride samples in the same track is applied, considering the continuity and smoothness in human locomotion. The window size is carefully chosen: if the window size is too small, the correction is not efficient enough and in the other extreme case, we wipe out the direction changes. The window size of 6 roughly

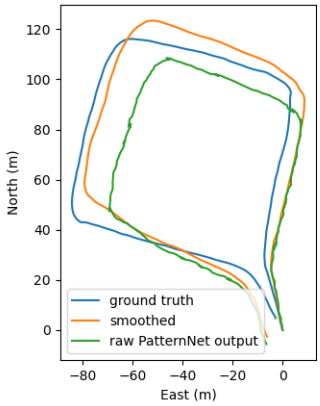


Fig. 5: Walking direction estimation before (in green) and after smoothing (in orange)

introduces a lag of 3 to 4 seconds (3 strides) depending on the walking speed. The efficiency of the smoothing process is shown in fig 5: the blue curve represents the ground truth user trajectory. The green curve is the estimated trajectory with the raw PatternNet outputs and the ground truth stride length and in orange, the raw PatternNet outputs are replaced by their moving averages.

D. Accuracy metric

The accuracy of the algorithm is measured as the absolute difference between a predicted walking angle α and the corresponding ground truth β. The distance δ between 2 angles α, β is computed as follows:

$$\delta(\alpha, \beta) = \min_{k \in \mathbb{Z}} |\alpha - \beta + 2k\pi|.$$

IV. Experiments

A. Experimental setup

We use 2 ULISS sensors [11], one as a wearable device to collect IMU and magnetometer readings, the other is attached to the user's foot to obtain ground truth trajectories [12], from which we compute the user's ground truth walking directions. ULISS (Fig 6(a)) is a state-of-the-art Inertial Navigation System containing an Xsens Mit-7 IMU-Mag sensor and a GNSS receiver, providing triaxial accelerometer, gyroscope, and magnetometer readings at 200Hz, GNSS readings at 5Hz, using GPS timestamps.

All our recordings start in an outdoor environment as requested by MAGYQ and the foot-mounted ULISS algorithm: both algorithms require a magnetometer calibration phase and an initial pointing direction estimation in a clean magnetic environment.

B. Datasets

Our volunteers walk naturally in indoor and outdoor environments, carrying the device either in texting, swinging, or pocket mode. To guarantee the repeatability of the experiment

(a) (b)

Fig. 6: (a) ULISS sensor; (b) experimental setup: one ULISS sensor is held in the user's right hand and the other on the user's right foot

Fig. 7: ″dress code″ for experiments: for the pocket mode, our volunteers are asked not to wear a coat (left) and wear a jacket (right) with zipper or buttons instead

of the pocket mode, our volunteers are asked to wear a jacket with zipper or buttons, with pockets near the hip (Fig.7, right), instead of a coat (Fig.7, left). With a coat, signals recorded in the pocket location sense the volatile movements of the coat instead of the periodic body movements of the user.

Noisy or corrupted samples are removed. Each original stride sample in a training set generates 10 randomly rotated duplicates, thus the quantity of training samples is 10-folded. However, if we add too many duplicates to the training set, we increase the chance of overfitting.

Here is some essential information about three of our volunteers and their training sets:

1) Volunteer F1: Female, visually impaired, walks her white cane, 1.58m: All her 3 datasets are collected in December 2022 and in January 2023.

mode	total walking distance (m)	number of samples
texting	648	8650
swinging	641	8620
pocket	760	9840

TABLE I: Datasets collected by volunteer F1

2) Volunteer M1: Male, sighted, 1.83m: Dataset collected between November 2022 and mid-January 2023.

3) Volunteer M2: Male, sighted, 1.80m: Dataset collected in November and December 2022.

mode	total walking distance (m)	number of samples
pocket	1467	15280

TABLE II: Dataset collected by volunteer M1

mode	total walking distance (m)	number of samples
swinging	964	11210

TABLE III: Dataset collected by volunteer M2

Each dataset is randomly split into 90% for training and 10% for validation.

C. Experimental results

We evaluated 5 PatternNet models each trained on a dataset described in IV-B. Among state-of-the-art pedestrian positioning or walking direction estimating solutions, we chose RoNIN [6] as the competing method. RoNIN implementation and trained model weights are available here [13]. We only evaluate RoNIN ResNet, since it yields the best performance according to the paper. To run RoNIN with ULISS data, we replace the Game Rotation Vector (GRV) required by RoNIN with the orientation quaternion estimated by MAGYQ. Since we only evaluate the walking direction, we scale the RoNIN trajectory length to the real walking distance.

The evaluation tracks are recorded in the same condition as the training sets. For illustration's purpose, we plot the ground truth trajectory, estimated trajectory with PatternNet predictions and the ground truth stride length, and the RoNIN trajectory in the same East-North frame. For each test, we report the test environment, path length, mean angular errors, and error standard deviations of all stride segments' orientation inferred by PatternNet and by RoNIN.

1) Volunteer F1, texting: Test track recorded in the same period of the training set. See results in Fig.8 and table IV.

			PatterNet		RoNIN	
F1, texting	environment	length (m)	mean (°)	std (°)	mean (°)	std (°)
Test 1	outdoor	330	11.9	14.4	22.3	30.4

TABLE IV: Volunteer F1, texting: Performance evaluation of PatternNet and RoNIN

2) Volunteer F1, swinging: Test tracks recorded in the same period of the training set. See results in Fig.9 and table V.

			PatterNet		RoNIN	
F1, swinging	environment	length (m)	mean (°)	std (°)	mean (°)	std (°)
Test 1	outdoor	325	7.5	9.5	19.5	21.8
Test 2	outdoor	328	15.4	27.4	23.8	27.5

TABLE V: Volunteer F1, swinging: Performance evaluation of PatternNet and RoNIN

3) Volunteer F1, pocket: Test tracks recorded in the same period of the training set. See results in Fig.10 and table VI

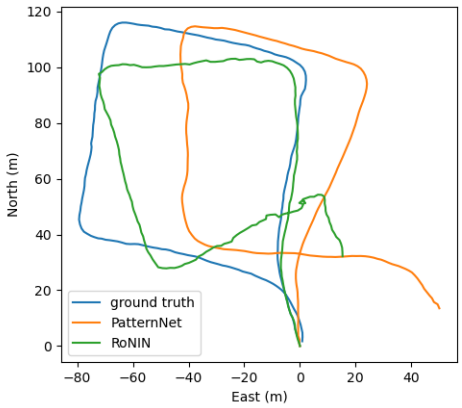


Fig. 8: volunteer F1 texting: Estimated trajectories by PatternNet (orange), RONIN (green) and the ground truth (blue)

			PatterNet		RoNIN	
F1, pocket	environment	length (m)	mean (°)	std (°)	mean (°)	std (°)
Test 1	outdoor	325	7.7	11.7	8.4	10.3
Test 2	outdoor	343	7.7	9.8	13.5	15.5

TABLE VI: Volunteer F1, pocket: Performance evaluation of PatternNet and RoNIN

4) Volunteer M1, pocket: Test 1 recorded in the same period of the training set, Test 2 and 3 were recorded 2 weeks after the collection of the training set. See results in Fig.11 and table VII

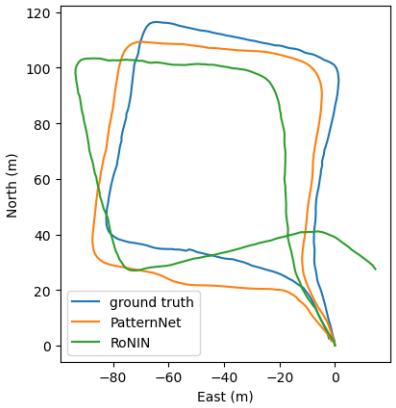
			PatterNet		RoNIN	
M1, pocket	environment	length (m)	mean (°)	std (°)	mean (°)	std (°)
Test 1	outdoor	323	6.6	8.7	9.1	11.5
Test 2	outdoor, indoor	160	7.5	13.2	6.3	13.3
Test 3	outdoor, indoor	286	12.6	15.9	11.7	16.3

TABLE VII: Volunteer M1, pocket: Performance evaluation of PatternNet and RoNIN

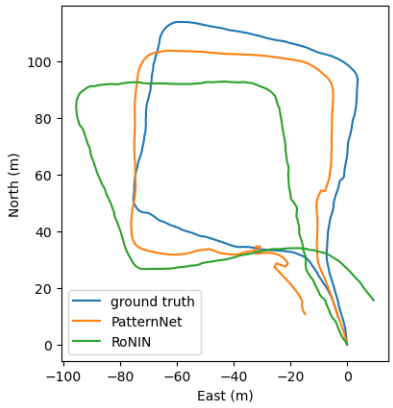
5) Volunteer M2, swinging: Test tracks recorded more than one month after the collection of the training set. See results in Fig.12 and table VIII

			PatterNet		RoNIN	
M2, swinging	environment	length (m)	mean (°)	std (°)	mean (°)	std (°)
Test 1	outdoor	202	6.1	9.4	8.7	14.5
Test 2	outdoor	113	6.3	8.9	8.6	14.7
Test 3	outdoor, indoor	150	9.1	11.6	10.5	15.6
Test 4	outdoor, indoor	250	5.9	9.8	9.5	12.2

TABLE VIII: Volunteer M2, swinging: Performance evaluation of PatternNet and RoNIN

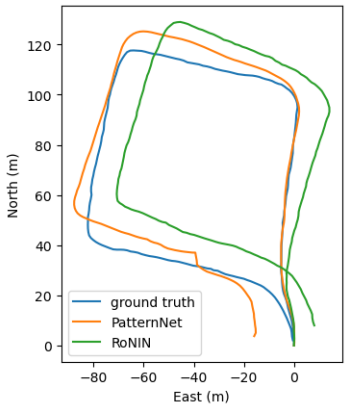


(a)

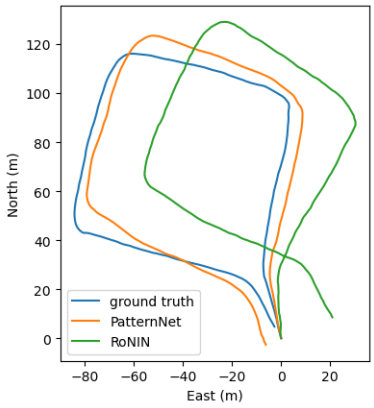


(b)

Fig. 9: volunteer F1 swinging: Performance evaluation of PatternNet and RoNIN: (a) Test 1; (b) Test 2



(a)



(b)

Fig. 10: volunteer F1 pocket: Performance evaluation of PatternNet and RoNIN: (a) Test 1; (b) Test 2

D. Analysis

The overall average performances of PatternNet and RoNIN for the three volunteers are reported in table IX. On average,

	PatterNet		RoNIN	
Volunteer	mean (°)	std (°)	mean (°)	std (°)
F1	10.4	14.6	17.5	21.1
M1	8.9	12.6	9	13.7
M2	6.9	9.9	9.3	14.3

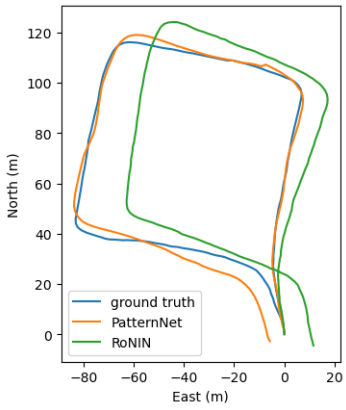
TABLE IX: Overall performances of PatternNet and RoNIN for the three volunteers

PatternNet estimates the walking directions with a mean accuracy between 7 to 10 degrees for the three test persons and is 1.5 times better than RONIN estimates. Four trained PatternNet models out of five perform better than RoNIN in our experiments. PatternNet provides better initial walking direction estimates. It is trained with a dataset that is 10^4 smaller than RoNIN's training dataset and achieves better mean average direction estimation for standard deviations in the same order of magnitude. PatternNet's strength is to achieve high accuracy angle estimation with a very small dataset. But contrary to RoNIN, the training is performed individually and cannot be generalized to unseen users. Both PatternNet and RoNIN perform poorly in F1's texting case. Errors of PatternNet are due to several reasons:

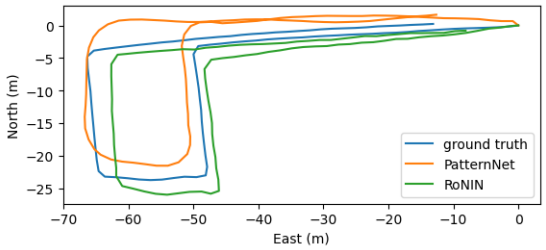
1) The irregularity of acceleration patterns: Important variations in the acceleration patterns are observed for certain users. Fig. 13 shows 4 samples of volunteer M1, pocket mode. As described in section II, this volunteer's pattern features a tail indicating roughly the walking direction. Sometimes, this tail is "normal" (a), sometimes it's sharp (b), sometimes it's "fat" (d), and sometimes it just disappears (c).

This can be especially problematic with modes for which the signal is already feint, namely the texting mode, leading to low signal-to-noise ratios and unreliable PatternNet outputs. For instance, in the texting mode, the dominant movements recorded are up-and-down, which is not informative about the walking direction. On the other hand, the horizontal movements are noisy and have a small amplitude (Fig. 14). This observation explains the poor performance of PatternNet in the texting case.

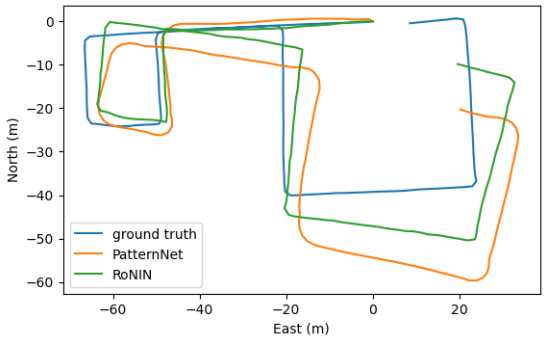
Although learning about individual human gaits aims at increasing the robustness of intra-individual gait variations, it is not canceled. Many factors can affect the regularity of the patterns: walking speed, hesitations, fatigue, different outfits,



(a)



(b)



(c)

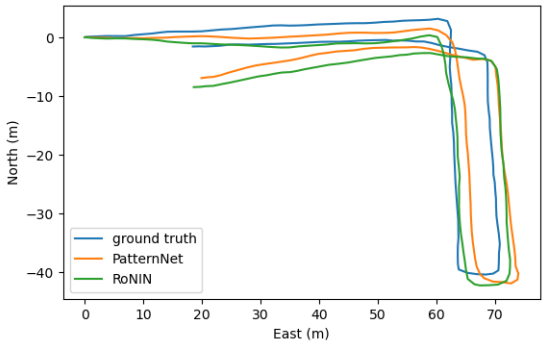
Fig. 11: volunteer M1 pocket: Performance evaluation of PatternNet and RoNIN: (a) Test 1; (b) Test 2; (c) Test 3

etc... Further studies are needed to identify these factors and characterize their impacts.

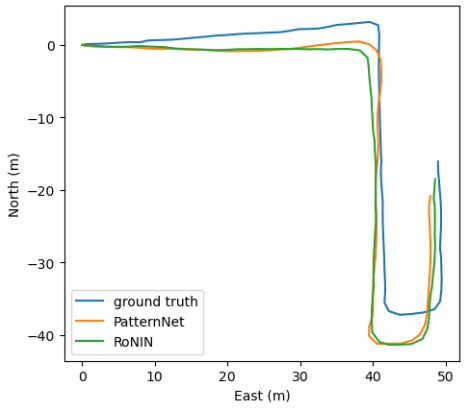
2) Errors propagated from MAGYQ: MAGYQ is an attitude estimation EKF that is not exempt from drift when the magnetic QSF (quasi-static-field) updates are not always available, especially in an indoor environment. As we use the output of this filter to project inertial measurements on the horizontal plane, attitude estimation error from MAGYQ can lead to deformed or badly rotated horizontal patterns.

V. Conclusion

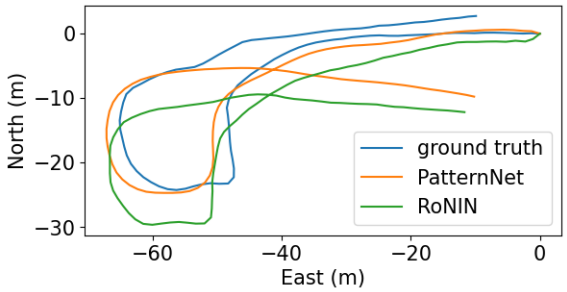
The paper proposes the computer vision approach PatternNet to extract geometrical features from the wearable device's horizontal accelerations, for walking direction inference. Our user-and-carrying-mode-customized strategy allows us to handle diversities in human walking and learn from very small



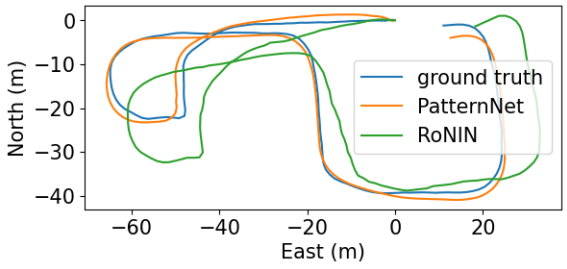
(a)



(b)



(c)



(d)

Fig. 12: volunteer M2, swinging: Performance evaluation of PatternNet and RoNIN: (a) Test 1; (b) Test 2; (c) Test 3; (d) Test 4

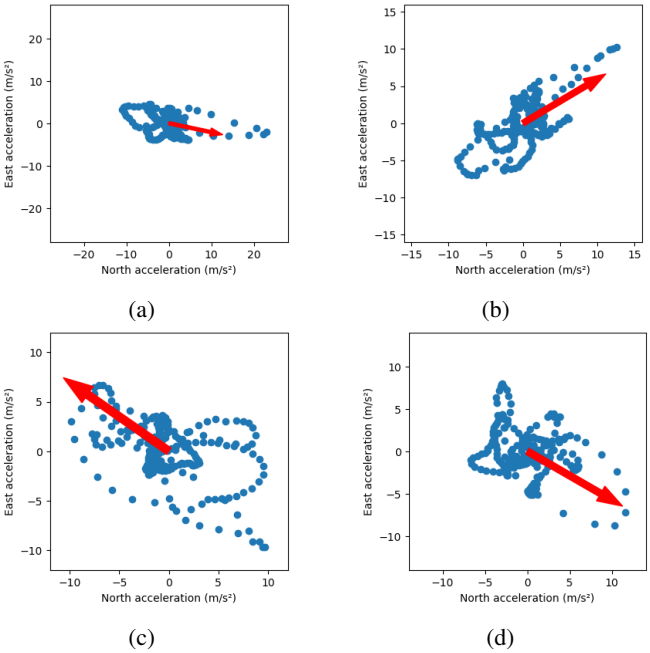


Fig. 13: volunteer M1 pocket: variations in the acceleration patterns. The user's walking direction is indicated by red arrows: (a) sample 1; (b) sample 2; (c) sample 3; (4) sample 4

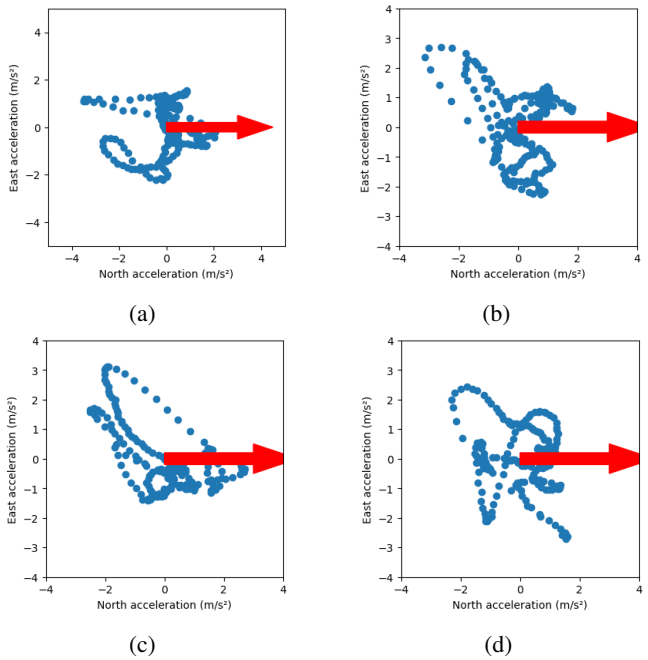


Fig. 14: volunteer F1 texting. variations in the acceleration patterns. The user's walking direction is represented by red arrows. : (a) sample 1; (b) sample 2; (c) sample 3; (d) sample 4;

datasets for an 8.5° mean accuracy with a mean 12° standard deviation. Experimental assessments are carried out in outdoor and indoor environments with three persons: one visually impaired lady and two sighted gentlemen walking naturally either in texting, swinging, or in-pocket mode. Performances of the five trained PatternNet models are compared to the state-of-the-art RoNIN's. The best performance achieved by PatternNet is a mean angular error of 5.9° over a 250m path including indoor segments. PatternNet's performance is limited by the need for (1) regular horizontal acceleration patterns and (2) highly accurate attitude estimation of the wearable device. Further studies will be conducted to improve the robustness against irregularities in the patterns and errors in the device attitude estimation.

ACKNOWLEDGMENT

This work was supported by the French National Research Agency under Grant ANR-20-LCV1-0002 and the French company Okeenea Digital.

REFERENCES

[1] H. Fu, Y. Kone, V. Renaudin, and N. Zhu, "A survey on artificial intelligence for pedestrian navigation with wearable inertial sensors," in *2022 IEEE 12th International Conference on Indoor Positioning and Indoor Navigation (IPIN)*, 2022, pp. 1–8.

[2] Z.-A. Deng, G. Wang, Y. Hu, and D. Wu, "Heading estimation for indoor pedestrian navigation using a smartphone in the pocket," *Sensors*, vol. 15, no. 9, pp. 21 518–21 536, 2015. [Online]. Available: https://www.mdpi.com/1424-8220/15/9/21518

[3] J. Perul and V. Renaudin, "Learning individual models to estimate the walking direction of mobile phone users," *IEEE Sensors Journal*, no. 10 p, pp. –, Jan. 2019. [Online]. Available: https://hal.archives-ouvertes.fr/hal-02285791

[4] Q. Wang, H. Luo, L. Ye, A. Men, F. Zhao, Y. Huang, and C. Ou, "Pedestrian heading estimation based on spatial transformer networks and hierarchical lstm," *IEEE Access*, vol. 7, pp. 162 309–162 322, 2019.

[5] J. Lin, C. Zou, L. Lan, S. Gu, and X. An, "Deep heading estimation for pedestrian dead reckoning," *Journal of Physics: Conference Series*, vol. 1656, no. 1, p. 012009, sep 2020. [Online]. Available: https://dx.doi.org/10.1088/1742-6596/1656/1/012009

[6] S. Herath, H. Yan, and Y. Furukawa, "Ronin: Robust neural inertial navigation in the wild: Benchmark, evaluations, amp; new methods," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 3146–3152.

[7] N. Al Abiad, Y. Kone, V. Renaudin, and T. Robert, "Smartphone inertial sensors based step detection driven by human gait learning," in *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2021, p. 8. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9662513

[8] V. Renaudin and C. Combettes, "Magnetic, acceleration fields and gyroscope quaternion (magyq)-based attitude estimation with smartphone sensors for indoor pedestrian navigation," *Sensors*, vol. 14, no. 12, pp. 22 864–22 890, 2014. [Online]. Available: https://www.mdpi.com/1424-8220/14/12/22864

[9] D. Saez, "Correcting image orientation using convolutional neural networks," Available: https://d4nst.github.io/2017/01/12/image-orientation/, Accessed Jan. 23, 2023 [Online], Jan. 12 2017.

[10] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, ser. Proceedings of Machine Learning Research, Y. W. Teh and M. Titterington, Eds., vol. 9. Chia Laguna Resort, Sardinia, Italy: PMLR, 13–15 May 2010, pp. 249–256. [Online]. Available: https://proceedings.mlr.press/v9/glorot10a.html

[11] M. Ortiz, M. De Sousa, and V. Renaudin, "A new pdr navigation device for challenging urban environments," *Journal of Sensors*, vol. 2017, pp. 1–11, 2017.

[12] N. Zhu, V. Renaudin, M. Ortiz, Y. Kone, C. Ichard, S. Ricou, and F. Gueit, "Foot-mounted ins for resilient real-time positioning of soldiers in non-collaborative indoor surroundings," in *2021 International Conference on Indoor Positioning and Indoor Navigation (IPIN)*. IEEE, 2021, pp. 1–8.

[13] S. Herath and H. Yan, "Ronin implementation and dataset," Available: https://github.com/Sachini/ronin, Accessed Apr. 25, 2022 [Online], Jan. 13 2022.